Simulating, Analyzing, and Animating Dynamical Systems

Software • Environments • Tools

The series includes handbooks and software guides as well as monographs on practical implementation of computational methods, environments, and tools. The focus is on making recent developments available in a practical format to researchers and other users of these methods and tools.

Software, Environments, and Tools

Bard Ermentrout, *Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students*

V. A. Barker, L. S. Blackford, J. Dongarra, J. Du Croz, S. Hammarling, M. Marinova, J. Waśniewski, and P. Yalamov, *LAPACK95 Users' Guide*

Stefan Goedecker and Adolfy Hoisie, *Performance Optimization of Numerically Intensive Codes*

Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, and Henk van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*

Lloyd N. Trefethen, *Spectral Methods in MATLAB*

E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide, Third Edition*

Michael W. Berry and Murray Browne, *Understanding Search Engines: Mathematical Modeling and Text Retrieval*

Jack J. Dongarra, Iain S. Duff, Danny C. Sorensen, and Henk A. van der Vorst, *Numerical Linear Algebra for High-Performance Computers*

R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*

Randolph E. Bank, *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations, Users' Guide 8.0*

L. S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley, *ScaLAPACK Users' Guide*

Greg Astfalk, editor, *Applications on Advanced Architecture Computers*

Françoise Chaitin-Chatelin and Valérie Frayssé, *Lectures on Finite Precision Computations*

Roger W. Hockney, *The Science of Computer Benchmarking*

Richard Barrett, Michael Berry, Tony F. Chan, James Demmel, June Donato, Jack Dongarra, Victor Eijkhout, Roldan Pozo, Charles Romine, and Henk van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*

E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide, Second Edition*

Jack J. Dongarra, Iain S. Duff, Danny C. Sorensen, and Henk van der Vorst, *Solving Linear Systems on Vector and Shared Memory Computers*

J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart, *Linpack Users' Guide*

Simulating, Analyzing, and Animating Dynamical Systems

A Guide to XPPAUT for Researchers and Students

Bard Ermentrout
University of Pittsburgh
Pittsburgh, Pennsylvania

siam
Society for Industrial and Applied Mathematics
Philadelphia

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data

Ermentrout, Bard.
Simulating, analyzing, and animating dynamical systems : a guide to XPPAUT for researchers and students / Bard Ermentrout.
p. cm. — (Software, environments, tools)
Includes bibliographical references and index.
ISBN 0-89871-506-7 (pbk.)
Differential equations—Numerical solutions—Data processing. I. Title. II. Series.

QA371.5.D37 E76 2002
515′.352—dc21 2002017639

About the cover: Pictured in the foreground is the manufacturer’s schematic of the design for the Zipper, a popular carnival ride. This drawing also appears in Figure 8.3 of Chapter 8, “Animation.” The background photograph shows the actual Zipper ride. Used with permission of Chance Rides, Wichita, KS.

Contents

List of Figures

Preface

XPPAUT is a tool for simulating, animating, and analyzing dynamical systems. The program evolved from a DOS program that was originally written so that John Rinzel and I could easily illustrate the dynamics of a simple model for an excitable membrane. The DOS program, PHASEPLANE, became a commercial project and was used for many years by a number of patient folks. In the early 1990s, I spent a month in a beautiful office at the Mathematical Sciences Research Institute as part of a Mathematical Biology program. During the evenings, I ported the DOS program to X Window on a UNIX environment while enjoying the sunset and listening to the same cassette tape over and over. (I forget what it was.) The program has evolved a great deal from those early years and is available at no cost to anyone who wishes to download it. I have also successfully compiled the X version to run under various 32-bit flavors of Windows and also under the new Mac OS X.

I have added lots of integrators and tools as well as my own idiosyncratic interface to the amazing continuation package AUTO. Most things that you might want to do that concern dynamics—either discrete or continuous—can probably be done with *XPPAUT* if you know a few of the tricks. That is the point of this book; I suspect many users do not take full advantage of the features of the program. This is mostly my fault, as the users' manual that is distributed with the program, while comprehensive in its description of all the features, is hopelessly baroque in its organization.

Why should anyone want to use *XPPAUT*? There are plenty of packages that will integrate differential equations for you. Many people use MATLAB, MAPLE, or MATHEMATICA to study and analyze dynamical systems. These are all general purpose packages that have the capability to do most everything that is described in this book. However, the latter two symbolic packages are extremely slow when it comes to numerically solving differential equations. Furthermore, they do not offer much flexibility in the choice of integration methods, and the integration is not done interactively. That is, you cannot see the progress of the solution until it is computed. Standard qualitative tools such as direction fields and nullclines require running additional packages or writing by hand. MATLAB has great flexibility and can even integrate differential equations with discontinuities such as the integrate-and-fire equations. However, the numerical integration is generally slower than can be achieved with *XPPAUT*. None of the packages offers an interface to AUTO, the main reason that some people use *XPPAUT*. The syntax of *XPPAUT* for setting up differential equations is pretty simple compared to the other programs. Finally, *XPPAUT* is free—no license demons crashing once a year, no guilt copying to another computer, and the source code is always there for the taking. To download your copy, go to the *XPPAUT* homepage at http://www.math.pitt.edu/~bard/xpp/xpp.html.

How to use this book. This book is written to be used by either a researcher or modeler who wants to simulate and analyze a particular system or by students as an adjunct to a modeling class or a class in differential equations. I have used it in many such courses both at the sophomore engineering level up through the graduate level in a dynamical systems class. I have used *XPPAUT* in applied courses for students in neuroscience and physiology. The present book contains many examples and many exercises. Along the way, I hope that it can aid in teaching certain concepts in the analysis of the behavior of differential equations. Most of the problems and examples are taken from research papers. The emphasis, I am afraid, is skewed toward biological applications, as that is what I do.

If all you want to do is solve differential equations and graph the solutions, then most of the information you will need can be found in Chapter 3. Suggestions for how you can use *XPPAUT* in a classroom setting are found in Chapter 4. Research problems involve a more complete set of tools: see Chapter 5 for functional and stochastic differential equations, and Chapter 6 for how to discretize and solve partial differential equations. Boundary value problems are also covered in Chapter 6. Chapter 7 introduces bifurcation theory and the use of the AUTO interface in *XPPAUT*. Chapter 8 shows you how to make animations with the built-in animator, and Chapter 9 shows other ways to make animations. Tricks and special classes of differential equations also are described in Chapter 9.

Acknowledgments. I have benefitted a great deal from the many users of various versions of the program. To those one or two of you who sent me a note about how useful the program is rather than what new features you wished I'd put into it or which ones didn't work, I salute you. To the others, well, I reluctantly thank you, as your comments motivated new features and bugs. Mostly, I thank John Rinzel and Artie Sherman for being guinea pigs for many versions of the program that have appeared throughout the years. Finally, I want to thank my wife, Ellen, for her patience, and the boys, Kyle and Jordan, without whom this book would have a 2000 copyright.

Bard Ermentrout