

Academic Press is an imprint of Elsevier  
The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, UK  
225 Wyman Street, Waltham, MA 02451, USA

First edition 2014

Copyright © 2014 Elsevier Ltd. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means electronic, mechanical, photocopying, recording or otherwise without the prior written permission of the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone (+44) (0) 1865 843830; fax (+44) (0) 1865 853333; email: [permissions@elsevier.com](mailto:permissions@elsevier.com). Alternatively you can submit your request online by visiting the Elsevier web site at <http://elsevier.com/locate/permissions>, and selecting Obtaining permission to use Elsevier material.

#### **Notice**

No responsibility is assumed by the publisher for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein. Because of rapid advances in the medical sciences, in particular, independent verification of diagnoses and drug dosages should be made.

#### **Library of Congress Cataloging in Publication Data**

A catalog record for this book is available from the Library of Congress

#### **British Library Cataloguing in Publication Data**

A catalogue record for this book is available from the British Library

ISBN: 978-0-12-411597-2

For information on all Elsevier publications  
visit our website at [www.store.elsevier.com](http://www.store.elsevier.com)

Printed and bound in Poland.

14 15 16 17 10 9 8 7 6 5 4 3 2 1



Working together  
to grow libraries in  
developing countries

[www.elsevier.com](http://www.elsevier.com) • [www.bookaid.org](http://www.bookaid.org)

# Introduction

## Signal Processing at Your Fingertips!

Let us flash back to the 1970s when the editors-in-chief of this e-reference were graduate students. One of the time-honored traditions then was to visit the libraries several times a week to keep track of the latest research findings. After your advisor and teachers, the librarians were your best friends. We visited the engineering and mathematics libraries of our Universities every Friday afternoon and poured over the IEEE Transactions, Annals of Statistics, the Journal of Royal Statistical Society, Biometrika, and other journals so that we could keep track of the recent results published in these journals. Another ritual that was part of these outings was to take sufficient number of coins so that papers of interest could be xeroxed. As there was no Internet, one would often request copies of reprints from authors by mailing postcards and most authors would oblige. Our generation maintained thick folders of hard-copies of papers. Prof. Azriel Rosenfeld (one of RC's mentors) maintained a library of over 30,000 papers going back to the early 1950s!

Another fact to recall is that in the absence of Internet, research results were not so widely disseminated then and even if they were, there was a delay between when the results were published in technologically advanced western countries and when these results were known to scientists in third world countries. For example, till the late 1990s, scientists in US and most countries in Europe had a lead time of at least a year to 18 months since it took that much time for papers to appear in journals after submission. Add to this the time it took for the Transactions to go by surface mails to various libraries in the world. Scientists who lived and worked in the more prosperous countries were aware of the progress in their fields by visiting each other or attending conferences.

Let us race back to 21st century! We live and experience a world which is fast changing with rates unseen before in the human history. The era of Information and Knowledge societies had an impact on all aspects of our social as well as personal lives. In many ways, it has changed the way we experience and understand the world around us; that is, the way we learn. Such a change is much more obvious to the younger generation, which carries much less momentum from the past, compared to us, the older generation. A generation which has grew up in the Internet age, the age of Images and Video games, the age of IPAD and Kindle, the age of the fast exchange of information. These new technologies comprise a part of their "real" world, and Education and Learning can no more ignore this reality. Although many questions are still open for discussions among sociologists, one thing is certain. Electronic publishing and dissemination, embodying new technologies, is here to stay. This is the only way that effective pedagogic tools can be developed and used to assist the learning process from now on. Many kids in the early school or even preschool years have their own IPADs to access information in the Internet. When they grow up to study engineering, science, or medicine or law, we doubt if they ever will visit a library as they would by then expect all information to be available at their fingertips, literally!

Another consequence of this development is the leveling of the playing field. Many institutions in lesser developed countries could not afford to buy the IEEE Transactions and other journals of repute. Even if they did, given the time between submission and publication of papers in journals and the time it took for the Transactions to be sent over surface mails, scientists and engineers in lesser developed countries were behind by two years or so. Also, most libraries did not acquire the proceedings of conferences and so there was a huge gap in the awareness of what was going on in technologically advanced

countries. The lucky few who could visit US and some countries in Europe were able to keep up with the progress in these countries. This has changed. Anyone with an Internet connection can request or download papers from the sites of scientists. Thus there is a leveling of the playing field which will lead to more scientist and engineers being groomed all over the world.

The aim of Online Reference for Signal Processing project is to implement such a vision. We all know that asking any of our students to search for information, the first step for him/her will be to click on the web and possibly in the Wikipedia. This was the inspiration for our project. To develop a site, related to the Signal Processing, where a selected set of reviewed articles will become available at a first “click.” However, these articles are fully refereed and written by experts in the respected topic. Moreover, the authors will have the “luxury” to update their articles regularly, so that to keep up with the advances that take place as time evolves. This will have a double benefit. Such articles, besides the more classical material, will also convey the most recent results providing the students/researchers with up-to-date information. In addition, the authors will have the chance of making their article a more “permanent” source of reference, that keeps up its freshness in spite of the passing time.

The other major advantage is that authors have the chance to provide, alongside their chapters, any multimedia tool in order to clarify concepts as well as to demonstrate more vividly the performance of various methods, in addition to the static figures and tables. Such tools can be updated at the author’s will, building upon previous experience and comments. We do hope that, in future editions, this aspect of this project will be further enriched and strengthened.

In the previously stated context, the Online Reference in Signal Processing provides a revolutionary way of accessing, updating and interacting with online content. In particular, the Online Reference will be a living, highly structured, and searchable peer-reviewed electronic reference in signal/image/video Processing and related applications, using existing books and newly commissioned content, which gives tutorial overviews of the latest technologies and research, key equations, algorithms, applications, standards, code, core principles, and links to key Elsevier journal articles and abstracts of non-Elsevier journals.

The audience of the Online Reference in Signal Processing is intended to include practicing engineers in signal/image processing and applications, researchers, PhD students, post Docs, consultants, and policy makers in governments. In particular, the readers can be benefited in the following needs:

- To learn about new areas outside their own expertise.
- To understand how their area of research is connected to other areas outside their expertise.
- To learn how different areas are interconnected and impact on each other: the need for a “helicopter” perspective that shows the “wood for the trees.”
- To keep up-to-date with new technologies as they develop: what they are about, what is their potential, what are the research issues that need to be resolved, and how can they be used.
- To find the best and most appropriate journal papers and keeping up-to-date with the newest, best papers as they are written.
- To link principles to the new technologies.

The Signal Processing topics have been divided into a number of subtopics, which have also dictated the way the different articles have been compiled together. Each one of the subtopics has been coordinated by an AE (Associate Editor). In particular:

1. Signal Processing Theory (Prof. P. Diniz)
2. Machine Learning (Prof. J. Suykens)
3. DSP for Communications (Prof. N. Sidiropoulos)
4. Radar Signal Processing (Prof. F. Gini)
5. Statistical SP (Prof. A. Zoubir)
6. Array Signal Processing (Prof. M. Viberg)
7. Image Enhancement and Restoration (Prof. H. J. Trussell)
8. Image Analysis and Recognition (Prof. Anuj Srivastava)
9. Video Processing (other than compression), Tracking, Super Resolution, Motion Estimation, etc. (Prof. A. R. Chowdhury)
10. Hardware and Software for Signal Processing Applications (Prof. Ankur Srivastava)
11. Speech Processing/Audio Processing (Prof. P. Naylor)
12. Still Image Compression
13. Video Compression

We would like to thank all the Associate Editors for all the time and effort in inviting authors as well as coordinating the reviewing process. The Associate Editors have also provided succinct summaries of their areas.

The articles included in the current editions comprise the first phase of the project. In the second phase, besides the updates of the current articles, more articles will be included to further enrich the existing number of topics. Also, we envisage that, in the future editions, besides the scientific articles we are going to be able to include articles of historical value. Signal Processing has now reached an age that its history has to be traced back and written.

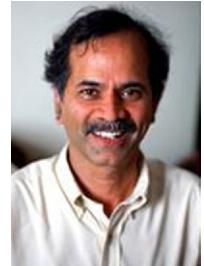
Last but not least, we would like to thank all the authors for their effort to contribute in this new and exciting project. We earnestly hope that in the area of Signal Processing, this reference will help level the playing field by highlighting the research progress made in a timely and accessible manner to anyone who has access to the Internet. With this effort the next breakthrough advances may be coming from all around the world.

The companion site for this work: <http://booksite.elsevier.com/9780124166165> includes multimedia files (Video/Audio) and MATLAB codes for selected chapters.

Rama Chellappa  
Sergios Theodoridis

# About the Editors

**Rama Chellappa** received the B.E. (Hons.) degree in Electronics and Communication Engineering from the University of Madras, India in 1975 and the M.E. (with Distinction) degree from the Indian Institute of Science, Bangalore, India in 1977. He received the M.S.E.E. and Ph.D. Degrees in Electrical Engineering from Purdue University, West Lafayette, IN, in 1978 and 1981, respectively. During 1981–1991, he was a faculty member in the department of EE-Systems at University of Southern California (USC). Since 1991, he has been a Professor of Electrical and Computer Engineering (ECE) and an affiliate Professor of Computer Science at University of Maryland (UMD), College Park. He is also affiliated with the Center for Automation Research, the Institute for Advanced Computer Studies (Permanent Member) and is serving as the Chair of the ECE department. In 2005, he was named a Minta Martin Professor of Engineering. His current research interests are face recognition, clustering and video summarization, 3D modeling from video, image and video-based recognition of objects, events and activities, dictionary-based inference, compressive sensing, domain adaptation and hyper spectral processing.



Prof. Chellappa received an NSF Presidential Young Investigator Award, four IBM Faculty Development Awards, an Excellence in Teaching Award from the School of Engineering at USC, and two paper awards from the International Association of Pattern Recognition (IAPR). He is a recipient of the K.S. Fu Prize from IAPR. He received the Society, Technical Achievement, and Meritorious Service Awards from the IEEE Signal Processing Society. He also received the Technical Achievement and Meritorious Service Awards from the IEEE Computer Society. At UMD, he was elected as a Distinguished Faculty Research Fellow, as a Distinguished Scholar-Teacher, received an Outstanding Innovator Award from the Office of Technology Commercialization, and an Outstanding GEMSTONE Mentor Award from the Honors College. He received the Outstanding Faculty Research Award and the Poole and Kent Teaching Award for Senior Faculty from the College of Engineering. In 2010, he was recognized as an Outstanding ECE by Purdue University. He is a Fellow of IEEE, IAPR, OSA, and AAAS. He holds four patents.

Prof. Chellappa served as the Editor-in-Chief of IEEE Transactions on Pattern Analysis and Machine Intelligence. He has served as a General and Technical Program Chair for several IEEE international and national conferences and workshops. He is a Golden Core Member of the IEEE Computer Society and served as a Distinguished Lecturer of the IEEE Signal Processing Society. Recently, he completed a two-year term as the President of the IEEE Biometrics Council.

**Sergios Theodoridis** is currently Professor of Signal Processing and Communications in the Department of Informatics and Telecommunications of the University of Athens. His research interests lie in the areas of Adaptive Algorithms and Communications, Machine Learning and Pattern Recognition, Signal Processing for Audio Processing and Retrieval. He is the co-editor of the book “Efficient Algorithms for Signal Processing and System Identification,” Prentice Hall 1993, the co-author of the best selling book “Pattern Recognition,” Academic Press, 4th ed. 2008, the co-author of the book “Introduction to Pattern Recognition: A MATLAB Approach,” Academic Press, 2009, and the co-author of three books in Greek, two of them for the Greek Open University. He is Editor-in-Chief for the Signal Processing Book Series, Academic Press and for the E-Reference Signal Processing, Elsevier.



He is the co-author of six papers that have received best paper awards including the 2009 IEEE Computational Intelligence Society Transactions on Neural Networks Outstanding paper Award. He has served as an IEEE Signal Processing Society Distinguished Lecturer. He was *Otto Monstead Guest Professor*, Technical University of Denmark, 2012, and holder of the *Excellence Chair*, Department of Signal Processing and Communications, University Carlos III, Madrid, Spain, 2011.

He was the General Chairman of EUSIPCO-98, the Technical Program co-Chair for ISCAS-2006 and ISCAS-2013, and co-Chairman and co-Founder of CIP-2008 and co-Chairman of CIP-2010. He has served as President of the European Association for Signal Processing (EURASIP) and as member of the Board of Governors for the IEEE CAS Society. He currently serves as member of the Board of Governors (Member-at-Large) of the IEEE SP Society.

He has served as a member of the Greek National Council for Research and Technology and he was Chairman of the SP advisory committee for the Edinburgh Research Partnership (ERP). He has served as Vice Chairman of the Greek Pedagogical Institute and he was for 4 years member of the Board of Directors of COSMOTE (the Greek mobile phone operating company). He is Fellow of IET, a Corresponding Fellow of the Royal Society of Edinburgh (RSE), a Fellow of EURASIP, and a Fellow of IEEE.

# Section Editors

## Section 1

**Abdelhak M. Zoubir** is a Fellow of the IEEE for contributions to statistical signal processing and an IEEE Distinguished Lecturer (Class 2010–2011). He received the Dipl.-Ing degree (B.Sc./B.Eng.) from Fachhochschule Niederrhein, Germany, in 1983, the Dipl.-Ing. (M.Sc./M.Eng.) and the Dr.-Ing. (Ph.D.) degree from Ruhr-Universität Bochum, Germany, in 1987 and 1992, respectively, all in Electrical Engineering. Early placement in industry (Klöckner-Moeller & Siempelkamp AG) was then followed by Associate Lectureship in the Division for Signal Theory at Ruhr-Universität Bochum, Germany. In June 1992, he joined Queensland University of Technology where he was Lecturer, Senior Lecturer, and then Associate Professor in the School of Electrical and Electronic Systems Engineering. In March 1999, he took up the position of Professor of Telecommunications at Curtin University of Technology, where he was Head of the School of Electrical & Computer Engineering from November 2001 until February 2003. He has been Professor of Signal Processing at Technische Universität Darmstadt since February 2003 and October 2012, respectively.



His research interest lies in statistical methods for signal processing with emphasis on bootstrap techniques, robust detection and estimation, and array processing applied to telecommunications, radar, sonar, automotive monitoring and safety, and biomedicine. He published extensively on these areas. He was/is General Co-Chair or Technical Co-Chair of numerous international conferences. His current editorial work includes member of the Editorial Board of the EURASIP journal Signal Processing and Editor-In-Chief of the IEEE Signal Processing Magazine (2012–2014). He was Chair of the IEEE SPS Technical Committee Signal Processing Theory and Methods (SPTM). He currently serves on the Board of Directors of the European Association of Signal Processing (EURASIP).

## Section 2

**Mats Viberg** received the Ph.D. degree in Automatic Control from Linköping University, Sweden in 1989. He has held academic positions at Linköping University and visiting Scholarships at Stanford University and Brigham Young University, USA. Since 1993, he is a Professor of Signal Processing at Chalmers University of Technology, Sweden. During 1999–2004 he served as the Chair of the Department of Signals and Systems. Since 2011, he holds a position as First Vice President at Chalmers University of Technology.



His research interests are in Statistical Signal Processing and its various applications, including Antenna Array Signal Processing, System Identification, Wireless Communications, Radar Systems, and Automotive Signal Processing.

He has served in various capacities in the IEEE Signal Processing Society, including Chair of the Technical Committee (TC) on Signal Processing Theory and Methods (2001–2003) Chair of the TC on Sensor Array and Multichannel (2011–2012), Associate Editor of the Transactions

on Signal Processing (2004–2005), member of the Awards Board (2005–2007), and member at large of the Board of Governors (2010–2012).

He has received two Paper Awards from the IEEE Signal Processing Society (1993 and 1999 respectively), and the Excellent Research Award from the Swedish Research Council (VR) in 2002. He is a Fellow of the IEEE since 2003, and his research group received the 2007 EURASIP European Group Technical Achievement Award. In 2008, he was elected into the Royal Swedish Academy of Sciences (KVA). He has published more than 45 journal papers, together cited more than 6800 times, and his H-index is 32 (Google Scholar, April 2013).

# Authors Biography

## CHAPTER 2

**Visa Koivunen** received the D.Sc. degree from the University of Oulu, Finland. He was a visiting researcher at the University of Pennsylvania from 1992 to 1995. Since 1999, he has been a Professor at Aalto University (Helsinki University of Technology, Finland), where he is currently Academy Professor. He is Vice Chair and one of the principal investigators in the Smart Radios and Wireless Systems Centre of Excellence in Research nominated by the Academy of Finland. He has been an adjunct faculty member at the University of Pennsylvania and a visiting fellow at Nokia Research Center. He spent his sabbatical term at Princeton University and makes frequent research visits there.



His research interests include statistical, communications, and array signal processing. He has published over 350 journal and conference papers in these areas. He received the 2007 IEEE Signal Processing Society Best Paper Award. He is an Editorial Board Member of the IEEE Signal Processing Magazine and a Fellow of the IEEE.

**Esa Ollila** received the M.Sc. degree in mathematics from the University of Oulu, in 1998, Ph.D. degree in statistics with honors from the University of Jyvaskyla, in 2002, and the D.Sc. (Tech) degree with honors in signal processing from Aalto University, in 2010. From 2004 to 2007 he was a post-doctoral fellow of the Academy of Finland. He has also been a Senior Researcher and a Senior Lecturer at Aalto University and University of Oulu, respectively. Currently, from August 2010, he is appointed as an Academy Research Fellow of the Academy of Finland at the Department of Signal Processing and Acoustics, Aalto University, Finland. He is also Adjunct Professor (statistics) of University of Oulu.



During the Fall-term 2001 he was a Visiting Researcher with the Department of Statistics, Pennsylvania State University, State College, PA while the academic year 2010–2011 he spent as a Visiting Post-doctoral Research Associate with the Department of Electrical Engineering, Princeton University, Princeton, NJ. His research interests focus on theory and methods of statistical signal processing, independent component analysis and blind source separation, complex-valued signal processing, array and radar signal processing, and robust and non-parametric statistical methods.

## CHAPTER 3

**Ljubiša Stanković** was born in Montenegro on June 1, 1960. He received a B.S. in electrical engineering from the University of Montenegro in 1982, with the award as the best student at the University. In 1984, he received an M.S. in communications from the University of Belgrade, and a Ph.D. in theory of electromagnetic waves from the University of Montenegro in 1988. As a Fulbright grantee, he spent the 1984–1985 academic year at the Worcester Polytechnic Institute in Worcester, MA. Since 1982, he has been on the faculty at the University of

Montenegro, where he has been a Full Professor since 1995. In 1997–1999, he was on leave at the Ruhr University Bochum in Germany, supported by the Alexander von Humboldt Foundation. At the beginning of 2001, he was at the Technische Universiteit Eindhoven in the Netherlands as a Visiting Professor. During 2003–2008, he was the Rector of the University of Montenegro. He is the Ambassador of Montenegro to the United Kingdom, Ireland, and Iceland. His current interests are in signal processing. He published about 350 technical papers, more than 120 of them in the leading journals, mainly the IEEE editions. He received the highest state award of Montenegro in 1997 for scientific achievements. He was a member of the IEEE Signal Processing Society's Technical Committee on Theory and Methods, an associate editor of the IEEE Transactions on Image Processing, and an associate editor of the IEEE Signal Processing Letters. He is an associate editor of the IEEE Transactions on Signal Processing. He has been a member of the National Academy of Science and Arts of Montenegro (CANU) since 1996 and a member of the European Academy of Sciences and Arts. He is a Fellow of the IEEE for contributions to time-frequency signal analysis.



**Miloš Dakovic** was born in 1970 in Nikšić, Montenegro. He received a B.S. in 1996, an M.S. in 2001, and a Ph.D. in 2005, all in electrical engineering from the University of Montenegro. He is an Associate Professor at the University of Montenegro. His research interests are in signal processing, time-frequency signal analysis, and radar signal processing. He is a member of the Time-Frequency Signal Analysis Group ([www.tfsa.ac.me](http://www.tfsa.ac.me)) at the University of Montenegro, where he is involved in several research projects.



**Thayananthan Thayaparan** earned a B.S. (Hons.) in physics at the University of Jaffna, Sri-Lanka, an M.S. in physics at the University of Oslo, Norway in 1991, and a Ph.D. in atmospheric physics at the University of Western Ontario, Canada in 1996. From 1996 to 1997, he was employed as a post-doctoral fellow at the University of Western Ontario. In 1997, he joined the Defence Research and Development Canada-Ottawa, Department of National Defence, Canada, as a defense scientist. His research interests include advanced radar signal and image processing methodologies and techniques against SAR/ISAR and HFSWR problems, such as detection, classification, recognition, and identification. His current research includes synthetic aperture radar imaging algorithms, time-frequency analysis for radar imaging and signal analysis, radar micro-Doppler analysis, and noise radar technology. Thayaparan is a Fellow of the IET (Institute of Engineering & Technology). Currently, he is an Adjunct Professor at McMaster University. He received IET Premium Award for Signal Processing for the best paper published in 2009–2010. He is currently serving in the editorial board of IET Signal Processing. He has authored or coauthored over 174 publications in journals, proceedings, and internal distribution reports.



## CHAPTER 4

**Simon Godsill** is Professor of Statistical Signal Processing in the Engineering Department at Cambridge University and a Professorial Fellow and tutor at Corpus Christi College, Cambridge. He coordinates an active research group in Signal Inference and its Applications within the Signal Processing and Communications Laboratory at Cambridge, specializing in Bayesian computational methodology, multiple object tracking, audio and music processing, and financial time series modeling. A particular methodological theme over recent years has been the development of novel techniques for optimal Bayesian filtering and smoothing, using Sequential Monte Carlo or Particle Filtering methods. He has published extensively in journals, books, and international conference proceedings. He was technical chair of the IEEE NSSPW workshop in 2006 on sequential and nonlinear filtering methods, was Technical Chair for Fusion 2010 in Edinburgh, and has been on the conference panel for numerous other conferences workshops. He has served as Associate Editor for IEEE Tr. Signal Processing and the journal Bayesian Analysis. He was Theme Leader in Tracking and Reasoning over Time for the UKâs Data and Information Fusion Defence Technology Centre (DIF-DTC) and Principal Investigator on grants funded by the EU, EPSRC, QinetiQ, General Dynamics, MOD, Microsoft UK, Citibank, and Mastercard. In 2009–2010 he was co-organizer of an 18-month research program in Sequential Monte Carlo Methods at the SAMSI Institute in North Carolina, and will coorganize an Isaac Newton Institute Programme on Sequential Monte Carlo in 2014. He is a Director of CEDAR Audio Ltd. (which has received a technical Oscar for its audio processing work), and Input Dynamics Ltd., both companies which utilize his research work in the audio area.



## CHAPTER 5

**Pramod K. Varshney** (S'72–M'77–SM'82–F'97) was born in Allahabad, India, on July 1, 1952. He received the B.S. degree in electrical engineering and computer science (with highest honors), and the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois at Urbana-Champaign in 1972, 1974, and 1976 respectively.

From 1972 to 1976, he held teaching and research assistantships at the University of Illinois. Since 1976, he has been with Syracuse University, Syracuse, NY, where he is currently a Distinguished Professor of Electrical Engineering and Computer Science and the Director of CASE: Center for Advanced Systems and Engineering. He served as the Associate Chair of the department from 1993 to 1996. He is also an Adjunct Professor of Radiology at Upstate Medical University, Syracuse. His current research interests are in distributed sensor networks and data fusion, detection and estimation theory, wireless communications, image processing, radar signal processing, and remote sensing. He has published extensively. He is the author of *Distributed Detection and Data Fusion* (New York: Springer-Verlag, 1997). He has served as a consultant to several major companies.



He was a James Scholar, a Bronze Tablet Senior, and a Fellow while with the University of Illinois. He is a member of Tau Beta Pi and is the recipient of the 1981 ASEE Dow Outstanding Young Faculty Award. He was elected to the grade of Fellow of the IEEE in 1997 for his

contributions in the area of distributed detection and data fusion. He was the Guest Editor of the Special Issue on Data Fusion of the Proceedings of the IEEE, January 1997. In 2000, he received the Third Millennium Medal from the IEEE and Chancellor's Citation for exceptional academic achievement at Syracuse University. He is the recipient of the IEEE 2012 Judith A. Resnik Award. He serves as a Distinguished Lecturer for the IEEE Aerospace and Electronic Systems (AES) Society. He is on the Editorial Board of the Journal on Advances in Information Fusion. He was the President of International Society of Information Fusion during 2001.

**Engin Masazade** (S'03–M'10) got his B.S. degree from Electronics and Communications Engineering Department from Istanbul Technical University in 2003. He then obtained his M.S. and Ph.D. degrees from Sabanci University, Electronics Engineering Program, Istanbul, Turkey in 2006 and 2010 respectively. He is currently an Assistant Professor with the Department of Electrical and Electronics Engineering, Yeditepe University, Istanbul, Turkey. Before joining Yeditepe University, he was a Postdoctoral Research Associate with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, USA. His research interests include distributed detection, localization, and tracking for wireless sensor networks, dynamic resource management in sensor/communication networks.



## CHAPTER 6

**Venugopal V. Veeravalli** (M'92–SM'98–F'06) received the B.Tech. degree (SilverMedal Honors) from the Indian Institute of Technology, Bombay, in 1985, the M.S. degree from Carnegie Mellon University, Pittsburgh, PA, in 1987, and the Ph.D. degree from the University of Illinois at Urbana-Champaign, in 1992, all in electrical engineering.

He joined the University of Illinois at Urbana-Champaign in 2000, where he is currently a Professor in the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory. He served as a Program Director for communications research at the US. National Science Foundation in Arlington, VA from 2003 to 2005. He has previously held academic positions at Harvard University, Rice University, and Cornell University, and has been on sabbatical at MIT, IISc Bangalore, and Qualcomm, Inc. His research interests include wireless communications, distributed sensor systems and networks, detection and estimation theory, and information theory.



He was a Distinguished Lecturer for the IEEE Signal Processing Society during 2010–2011. He has been on the Board of Governors of the IEEE Information Theory Society. He has been an Associate Editor for Detection and Estimation for the IEEE Transactions on Information Theory and for the IEEE Transactions on Wireless Communications. Among the awards he has received for research and teaching are the IEEE Browder J. Thompson Best Paper Award, the National Science Foundation CAREER Award, and the Presidential Early Career Award for Scientists and Engineers (PECASE).

**Taposh Banerjee** received an M.E. in Telecommunications from the ECE Department of the Indian Institute of Science. He is now pursuing his Ph.D. at the Coordinated Science Laboratory and the ECE Department at the University of Illinois at Urbana-Champaign. His research interests are in detection and estimation theory, sequential analysis, and wireless communications and networks.



## CHAPTER 7

**Fredrik Gustafsson** is Professor in Sensor Informatics at Department of Electrical Engineering, Linköping University, since 2005. He received the M.Sc. degree in electrical engineering 1988 and the Ph.D. degree in Automatic Control, 1992, both from Linköping University. During 1992–1999 he held various positions in automatic control, and 1999–2005 he had a professorship in Communication Systems. His research interests are in stochastic signal processing, adaptive filtering and change detection, with applications to communication, vehicular, airborne, and audio systems. He is a co-founder of the companies NIRA Dynamics (automotive safety systems), Softube (audio effects), and SenionLab (indoor positioning systems).



He was an associate editor for IEEE Transactions of Signal Processing 2000–2006, IEEE Transactions on Aerospace and Electronic Systems 2010–2012, and EURASIP Journal on Applied Signal Processing 2007–2012. He was awarded the Arnberg prize by the Royal Swedish Academy of Science (KVA) 2004, elected member of the Royal Academy of Engineering Sciences (IVA) 2007, elevated to IEEE Fellow 2011, and awarded the Harry Rowe Mimno Award 2011 for the tutorial “Particle Filter Theory and Practice with Positioning Applications,” which was published in the AESS Magazine in July 2010.

## CHAPTER 8

**Brian M. Sadler** received the B.S. and M.S. degrees from the University of Maryland, College Park, and the Ph.D. degree from the University of Virginia, Charlottesville, all in electrical engineering. He is a Fellow of the Army Research Laboratory (ARL) in Adelphi, MD, and a Fellow of the IEEE. He has been an Associate or Guest Editor for many journals, including the IEEE Transactions on Signal Processing and IEEE Signal Processing Letters, and he has served on three IEEE Technical Committees in signal processing and networking. He received Best Paper Awards from the IEEE Signal Processing Society in 2006 and 2010. He has received several ARL and Army R&D awards, as well as a 2008 Outstanding Invention of the Year Award from the University of Maryland. His research interests include information science, networked and autonomous systems, sensing, and mixed-signal integrated circuit architectures.



**Terrence J. Moore** received his B.S. and M.A. in Mathematics from American in 1998 and 2000, respectively, and his Ph.D. in Mathematics from the University of Maryland, College Park, in 2010. He is a mathematician at the U.S. Army Research Laboratory in Adelphi, MD. His research interests include sampling theory, constrained statistical inference, stochastic optimization, and graph theory. He is a member of the IEEE.



## CHAPTER 9

**Ali H. Sayed** is Professor of electrical engineering at the University of California, Los Angeles, where he directs the UCLA Adaptive Systems Laboratory (<http://www.ee.ucla.edu/asl>). An author of over 400 scholarly publications and five books, his research involves several areas including adaptation and learning, network science, information processing theories, and biologically inspired designs. His work received several recognitions including the 2012 Technical Achievement Award from the IEEE Signal Processing Society, the 2005 Terman Award from the American Society for Engineering Education, a 2005 Distinguished Lecturer from the IEEE Signal Processing Society, the 2003 Kuwait Prize, and the 1996 IEEE Donald Fink Prize. His work has also been awarded several Best Paper Awards in 2002, 2005, and 2012 from the IEEE. He is a Fellow of both the IEEE and the American Association for the Advancement of Science (AAAS). He is the author of the textbooks *Fundamentals of Adaptive Filtering* (Wiley, 2003) and *Adaptive Filters* (Wiley, 2008). He is also co-author of the textbook *Linear Estimation* (Prentice Hall, 2000).



## CHAPTER 12

**Sergiy A. Vorobyov** received the M.Sc. and Ph.D. degrees in systems and control from Kharkiv National University of Radio Electronics, Ukraine, in 1994 and 1997, respectively. He is a Professor with the Department of Signal Processing and Acoustics, Aalto University, Finland and currently on leave from the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada, where he has been an Assistant Professor in 2006, Associate Professor in 2010, and Full Professor in 2012. Since his graduation, he also held various research and faculty positions at Kharkiv National University of Radio Electronics, Ukraine; the Institute of Physical and Chemical Research (RIKEN), Japan; McMaster University, Canada; Duisburg-Essen University and Darmstadt University of Technology, Germany; and the Joint Research Institute between Heriot-Watt University and Edinburgh University, UK. He has also held visiting positions at Technion, Haifa, Israel and Ilmenau University of Technology, Ilmenau, Germany. His research interests include statistical and array signal processing, applications of linear algebra,



optimization, and game theory methods in signal processing and communications, estimation, detection, and sampling theories, and cognitive systems.

He is a recipient of the 2004 IEEE Signal Processing Society Best Paper Award, the 2007 Alberta Ingenuity New Faculty Award, the 2011 Carl Zeiss Award (Germany), and other awards. He has served as the Track Chair for Asilomar 2011, Pacific Grove, CA, the Technical Co-Chair for IEEE CAMSAP 2011, Puerto Rico, and the Plenary Chair of ISWCS 2013, Ilmenau, Germany.

## CHAPTER 13

**Sven E. Nordholm** received his Ph.D., Licentiate of Engineering, and MScEE degrees from Lund University, Sweden. He was one of the founders of the Department of Signal Processing in Blekinge Institute of Technology, Sweden. In 1999, he was appointed as a Professor and Director of the Australian Telecommunications Research Institute, Curtin University of Technology, Perth. From 2009 he is a Professor of Signal Processing with Department of Electrical and Computer Engineering, Curtin University, Perth, Australia. From 2012 he is Head of the Department. He is also the co-founder and Chief Scientist of Sensear Pty Ltd.



He has more than 25 years of experience in acoustic signal processing. His work has been a mix of commercial and academic contributions. His scientific interests are in optimization of filters and broadband beamformers, blind signal separation, and speech enhancement.

**Hai H. Dam** received the Bachelor (with first-class Honors) and Ph.D. degrees (with distinction) from Curtin University of Technology, Perth, Australia in 1996 and 2001, respectively. From 1999 to 2000, she was at the Blekinge Institute of Technology, Karlskrona, Sweden, as a Visiting Research Associate. From 2001 to 2005, she was a Research Fellow/Senior Research Fellow with the Western Australian Telecommunications Research Institute (WATRI), Curtin University of Technology, Australia. Currently, she is a Senior Lecturer with the Department of Mathematics and Statistics, Curtin University of Technology.



Her research interests are adaptive array processing, optimization methods, equalization, and filter design.

**Chiong C. Lai** received his B.Eng. (First Class Honors) in Electronic and Communication Engineering from Curtin University of Technology, Perth in 2008. He continues onto his Ph.D. study in acoustic array signal processing at the same university. Currently, he is working as a Research Engineer for Electrical and Computer Engineering at Curtin University and at the same time, pursuing his Ph.D. study.



His research interests include robust steerable broadband beamformer, array signal processing, and acoustic source tracking.

**Eric A. Lehmann** graduated in 1999 from the Swiss Federal Institute of Technology (ETHZ) in Zurich, Switzerland, with a Dipl.El.-Ing. degree (Electrical Engineering). He received the M.Phil. and Ph.D. degrees, both in electrical engineering, from the Australian National University, Canberra, in 2000 and 2004, respectively. From 2004 to 2008, he held research positions with National ICT Australia (NICTA) in Canberra, as well as the Western Australian Telecommunications Research Institute (WATRI) in Perth, Australia. He is now working as Research Scientist for the Commonwealth Scientific and Industrial Research Organisation (CSIRO), within the Division of Mathematics, Informatics and Statistics in Perth. His current work involves the development of statistical image processing techniques for remote sensing data analysis, with a focus on model-data fusion and integration of multisensor data for environmental resource management and monitoring. His scientific interests also include various aspects of signal processing (such as acoustics, speech, and array processing) as well as Bayesian estimation and tracking (sequential Monte Carlo methods).



## CHAPTER 14

**Pei-Jung Chung** received Dr.-Ing. in 2002 from Ruhr-University at Bochum, Germany with distinction. From 2002 to 2004 she held a post-doctoral position at Carnegie Mellon University and University of Michigan, Ann Arbor, USA, respectively. From 2004 to 2006 she was Assistant Professor with National Chiao Tung University, HsinChu, Taiwan. In 2006 she joined the Institute for Digital Communications, School of Engineering, the University of Edinburgh, UK as Lecturer. Currently, she is Associate Member of IEEE Signal Processing Society Sensor Array Multichannel Technical Committee and serves for IEEE Communications Society, Multimedia Communications Technical Committee as Vice Chair of Interest Group on Acoustic and Speech Processing for Communications. Her research interests include array processing, statistical signal processing, wireless MIMO communications, and distributed processing in wireless sensor networks.



**Jia Yu** received the B.Eng. degree in electronic information engineering and the MSc degree in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2007 and 2009, respectively. From 2010 to 2012, he was a system engineer at Huawei Technologies Co., Shenzhen, China. He is currently working toward a Ph.D. degree at the University of Edinburgh. His research interest lies in statistical signal processing and wireless sensor networks.



## CHAPTER 15

**Martin Haardt** has been a Full Professor in the Department of Electrical Engineering and Information Technology and Head of the Communications Research Laboratory at Ilmenau University of Technology, Germany, since 2001. Since 2012, he has also served as an Honorary Visiting Professor in the Department of Electronics at the University of York, UK.

After studying electrical engineering at the Ruhr-University Bochum, Germany, and at Purdue University, USA, he received his Diplom-Ingenieur (M.S.) degree from the Ruhr-University Bochum in 1991 and his Doktor-Ingenieur (Ph.D.) degree from Munich University of Technology in 1996.



In 1997 he joined Siemens Mobile Networks in Munich, Germany, where he was responsible for strategic research for third generation mobile radio systems. From 1998 to 2001 he was the Director for International Projects and University Cooperations in the mobile infrastructure business of Siemens in Munich, where his work focused on mobile communications beyond the third generation. During his time at Siemens, he also taught in the international Master of Science in Communications Engineering program at Munich University of Technology.

He has received the 2009 Best Paper Award from the *IEEE Signal Processing Society*, the Vodafone (formerly Mannesmann Mobilfunk) Innovations-Award for outstanding research in mobile communications, the ITG Best Paper Award from the Association of Electrical Engineering, Electronics, and Information Technology (VDE), and the Rohde & Schwarz Outstanding Dissertation Award. He is a Senior Member of the IEEE. In the fall of 2006 and the fall of 2007 he was a Visiting Professor at the University of Nice in Sophia-Antipolis, France, and at the University of York, UK, respectively. His research interests include wireless communications, array signal processing, high-resolution parameter estimation, as well as numerical linear and multilinear algebra.

He has served as an Associate Editor for the *IEEE Transactions on Signal Processing* (2002–2006 and since 2011), the *IEEE Signal Processing Letters* (2006–2010), the *Research Letters in Signal Processing* (2007–2009), the *Hindawi Journal of Electrical and Computer Engineering* (since 2009), the *EURASIP Signal Processing Journal* (since 2011), and as a guest editor for the *EURASIP Journal on Wireless Communications and Networking*. He has also served as an elected member of the Sensor Array and Multichannel (SAM) technical committee of the *IEEE Signal Processing Society* (since 2011), as the technical co-chair of the *IEEE International Symposiums on Personal Indoor and Mobile Radio Communications (PIMRC)* 2005 in Berlin, Germany, as the technical program chair of the *IEEE International Symposium on Wireless Communication Systems (ISWCS)* 2010 in York, UK, as the general chair of *ISWCS* 2013 in Ilmenau, Germany, and as the general co-chair of the 5th *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)* 2013 in Saint Martin, French Caribbean.

**Marius Pesavento** received the Dipl.-Ing. degree from Ruhr-Universität Bochum, Germany, in 1999 and the M.Eng. degree from McMaster University, Hamilton, ON, Canada, in 2000, and

## xxxviii Authors Biography

the Dr.-Ing. degree in electrical engineering from Ruhr-Universität Bochum, Germany, in 2005. Between 2005 and 2007, he was a Research Engineer at FAG Industrial Services GmbH, Aachen, Germany. From 2007 to 2009, he was the Director of the Signal Processing Section at mimoOn GmbH, Duisburg, Germany. In 2010, he became a Professor for Robust Signal Processing at the Department of Electrical Engineering and Information Technology, Darmstadt University of Technology, Darmstadt, Germany, where he is currently the Head of the Communication Systems Group.



His research interests are in the area of robust signal processing and adaptive beamforming, high-resolution sensor array processing, transceiver design for cognitive radio systems, cooperative communications in relay networks, MIMO and multiantenna communications, space-time coding, multiuser and multicarrier wireless communication systems, convex optimization for signal processing and communications, statistical signal processing, spectral analysis, parameter estimation, and detection theory.

He was a recipient of the 2003 ITG/VDE Best Paper Award, the 2005 Young Author Best Paper Award of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the 2010 Best Paper Award of the CROWNCOM conference. He is a member of the Editorial board of the EURASIP Signal Processing Journal, an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and a member of the Sensor Array and Multichannel (SAM) Technical Committee of the IEEE Signal Processing Society (SPS).

**Florian Roemer** has studied computer engineering at Ilmenau University of Technology, Germany, and McMaster University, Canada. He received the Diplom-Ingenieur (M.S.) degree in communications engineering from Ilmenau University of Technology in October 2006. He received the Siemens Communications Academic Award in 2006 for his diploma thesis. Since December 2006, he has been a Research Assistant in the Communications Research Laboratory at the Ilmenau University of Technology. His research interests include multidimensional signal processing, high-resolution parameter estimation, as well as multi-user MIMO precoding and relaying.



**Mohammed Nabil El Korso** was born in Oran, Algeria in 1983. He received the BSc degree in Mathematics and physics from ENPEI-National Preparatory School, Algeria in 2004. The M.Sc. in Electrical Engineering from the National Polytechnic School, Algeria in 2007. He obtained a Master Research degree in Signal and Image Processing from Paris-Sud XI University/Supélec, France in 2008. In 2011, he obtained a Ph.D. degree from Paris-Sud XI University/Supélec in the Modeling and Statistical Signal Processing team of the Laboratory of Signals and Systems. From 2011 to 2012, he was a postdoctoral research associate in the Communication Systems Group at Technische Universität Darmstadt, Germany. Since 2012, he has been temporary Assistant Professor at École Normale Supérieure de Cachan. His research interests include lower bounds on the mean-square



error applied to statistical signal processing and estimation/detection theory with applications to array signal processing.

## CHAPTER 16

**Jean Pierre Delmas** was born in France in 1950. He received the Engineering Degree from Ecole Centrale de Lyon, France in 1973, the Certificat d'Etudes Supérieures from Ecole Nationale Supérieure des Télécommunications, Paris, France in 1982 and the Habilitation à diriger des recherches degree from the University of Paris XI, Orsay, France in 2001. Since 1980, he has been with Telecom SudParis (formerly INT), where he is presently a Professor in the CITI department and Director of UMR CNRS 5157 (SAMOVAR) laboratory. His teaching and research interests are in the areas of statistical signal processing with application to communications and antenna array. He has served as an Associate Editor for the IEEE Transactions on Signal Processing (2002–2006) and presently for Signal Processing (Elsevier) and IEEE Transactions on Signal Processing. He is a member of the IEEE Sensor Array and Multichannel (SAM) Technical Committee and a IEEE Senior Member. He is author and co-author of more than 110 papers (journal, conference, and chapter of book).

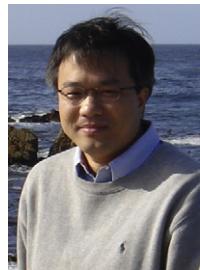


## CHAPTER 17

**Moeeness G. Amin** received his Ph.D. degree from the University of Colorado, Boulder, in 1984. He has been on the Faculty of Villanova University, Villanova, PA, since 1985, where he is now a Professor in the Department of Electrical and Computer Engineering and the Director of the Center for Advanced Communications. He has over 600 publications in the areas of wireless communications, time-frequency analysis, smart antennas, interference cancelation in broadband communication platforms, direction finding, GPS technologies, over-the-horizon radar, and radar imaging. He is the recipient of the 2009 Individual Technical Achievement Award from the European Association of Signal Processing. He is a Fellow of the Institute of Electrical and Electronics Engineers (IEEE); a Fellow of the International Society of Optical Engineering; recipient of the IEEE Third Millennium Medal; Distinguished Lecturer of the IEEE Signal Processing Society for 2003 and 2004. He was a Guest Editor of the Journal of Franklin Institute September 2008 Special Issue on Advances in Indoor Radar Imaging. He was a Guest Editor of the IEEE Transactions on Geoscience and Remote Sensing May 2009 Special Issue on Remote Sensing of Building Interior. He was the Co-Guest Editor of IET Signal Processing upcoming Special Issue on time-frequency Approach to Radar Detection, Imaging, and Classification. He serves on the Editorial Board of the Signal Processing Journal and the IEEE Signal Processing Magazine.



**Yimin D. Zhang** received his Ph.D. degree from the University of Tsukuba, Tsukuba, Japan, in 1988. He joined the faculty of the Department of Radio Engineering, Southeast University, Nanjing, China, in 1988. He served as a Technical Manager at the Communication Laboratory Japan, Kawasaki, Japan, from 1995 to 1997, and was a Visiting Researcher at ATR Adaptive Communications Research Laboratories, Kyoto, Japan, from 1997 to 1998. Since 1998, he has been with the Villanova University, Villanova, PA, where he is currently a Research Professor with the Center for Advanced Communications and the director of the Wireless Communications and Positioning Laboratory. He has more than 200 publications in the area of statistical signal and array processing for communications, radar, and navigation, including digital mobile communications, wireless networks, MIMO systems, time-frequency analysis, source localization and target tracking, over-the-horizon radar, radar imaging, cooperative communications, blind signal processing, jammer suppression, radio frequency identification (RFID), and image processing. He is an Associate Editor for the IEEE Transactions on Signal Processing and the Journal of the Franklin Institute, and serves on the editorial board of the Signal Processing Journal. He was an Associate Editor for the IEEE Signal Processing Letters from 2006 to 2010.



## CHAPTER 18

**Yu Hen Hu** received BSEE from National Taiwan University, Taiwan ROC in 1976, and MSEE and Ph.D. degrees from University of Southern California, Los Angeles, CA, USA in 1982. He was in the faculty of the Electrical Engineering Department of Southern Methodist University, Dallas, Texas. Since 1987, he has been with the Department of Electrical and Computer Engineering, University of Wisconsin, Madison where he is currently a professor.



He has broad research interests ranging from design and implementation of signal processing algorithms, computer-aided design and physical design of VLSI, pattern classification and machine learning algorithms, and image and signal processing in general. He has published more than 300 technical papers, edited or co-authored four books and many book chapters in these areas.

He has served as an associate editor for the IEEE Transaction of Acoustic, Speech, and Signal Processing, IEEE signal processing letters, European Journal of Applied signal Processing, Journal of VLSI Signal Processing, and IEEE Multimedia magazine. He has served as the secretary and an executive committee member of the IEEE signal processing society, a board of governor of IEEE neural network council representing the signal processing society, the chair of signal processing society neural network for signal processing technical committee, and the chair of IEEE signal processing society multimedia signal processing technical committee. He was also a steering committee member of the international conference of Multimedia and Expo on behalf of IEEE Signal processing society. He is a fellow of IEEE.

## CHAPTER 19

**Mário Costa** was born in Portugal in 1984. He received the M.Sc. degree with distinction in communications engineering from the Universidade do Minho, Portugal, in 2008. He has been with the Department of Signal Processing and Acoustics, Aalto University, Finland, since 2007, first as a Research Assistant and since 2008, as a Researcher working toward the Doctor of Science degree in technology. From January to July 2011, he was an External Researcher at the Connectivity Solutions Team, Nokia Research Center. His current research interests include sensor array and statistical signal processing as well as wireless communications.



## CHAPTER 20

**A. Lee Swindlehurst** received the B.S., summa cum laude, and M.S. degrees in Electrical Engineering from Brigham Young University, Provo, Utah, in 1985 and 1986, respectively, and the Ph.D. degree in Electrical Engineering from Stanford University in 1991. From 1986 to 1990, he was employed at ESL, Inc., of Sunnyvale, CA, where he was involved in the design of algorithms and architectures for several radar and sonar signal processing systems. He was on the faculty of the Department of Electrical and Computer Engineering at Brigham Young University from 1990 to 2007, where he was a Full Professor and served as Department Chair from 2003 to 2006. During 1996–1997, he held a joint appointment as a visiting scholar at both Uppsala University, Uppsala, Sweden, and at the Royal Institute of Technology, Stockholm, Sweden. From 2006 to 07, he was on leave working as Vice President of Research for ArrayComm LLC in San Jose, California. He is currently a Professor and Associate Chair of the Electrical Engineering and Computer Science Department at the University of California, Irvine. His research interests include sensor array signal processing for radar and wireless communications, detection and estimation theory, and system identification, and he has over 230 publications in these areas.



He is a Fellow of the IEEE, a past Secretary of the IEEE Signal Processing Society, past Editor-in-Chief of the *IEEE Journal of Selected Topics in Signal Processing*, and past member of the Editorial Boards for the *EURASIP Journal on Wireless Communications and Networking*, *IEEE Signal Processing Magazine*, and the *IEEE Transactions on Signal Processing*. He is a recipient of several paper awards: the 2000 IEEE W. R. G. Baker Prize Paper Award, the 2006 and 2010 IEEE Signal Processing Society's Best Paper Awards, the 2006 IEEE Communications Society Stephen O. Rice Prize in the Field of Communication Theory, and is co-author of a paper that received the IEEE Signal Processing Society Young Author Best Paper Award in 2001.

**Brian D. Jeffs** received B.S. (magna cum laude) and M.S. degrees in electrical engineering from Brigham Young University in 1978 and 1982, respectively. He received the Ph.D. degree from the University of Southern California in 1989, also in electrical engineering. He currently holds the rank of Professor in the Department of Electrical and Computer Engineering at Brigham Young University, where he lectures in the areas of signals and systems, digital signal processing, probability theory, and stochastic processes. Current research activity includes array signal processing for radio astronomy and radio frequency interference mitigation.



Previous employment includes Hughes Aircraft Company where he served as a sonar signal processing systems engineer in the anti-submarine warfare group. Projects there included algorithm development and system design for digital sonars in torpedo, surface ship towed array, and helicopter dipping array platforms.

He was a Vice General Chair for IEEE ICASSP-2001 held in Salt Lake City, Utah. He was a member of the executive organizing committee for the 1998 IEEE DSP Workshop, co-organized the 2010 Workshop on Phased Array Antennas Systems for Radio Astronomy, and served several years as chair of the Utah Chapter of the IEEE Communications and Signal Processing Societies.

**Gonzalo Seco-Granados** received the Ph.D. degree in telecommunication engineering from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 2000, and the M.B.A. degree from IESE-University of Navarra, Barcelona, in 2002. During 2002–2005, he was a member of the technical staff within the RF Payload Division, European Space Research and Technology Center (ESTEC), European Space Agency, Noordwijk, The Netherlands, where he was involved in the Galileo project. He led the activities concerning navigation receivers and indoor positioning for GPS and Galileo. Since 2006, he has been an Associate Professor with the Department of Telecommunications and Systems Engineering, Universitat Autònoma de Barcelona, Barcelona. From March 2007 to April 2011, he was coordinator of the Telecommunications Engineering degree and, since May 2011, he has been Vice Director of the UAB Engineering School.



He has been principal investigator of more than 12 national and international research projects, and acts often as an advisor of the European Commission in topics related to communications and navigation. He has had several visiting appointments at Brigham Young University, Provo, UT, and the University of California at Irvine. He has published more than 20 journal papers and more than 80 conference contributions. He holds two patents under exploitation. His research interests include signal processing for wireless communications and navigation, estimation theory, synchronization, location-based communications and optimization. He was appointed Director of one of the six Chairs of Technology and Knowledge Transfer “UAB Research Park-Santander” in March 2008.

**Jian Li** (S'87-M'91-SM'97-F'5) received the M.Sc. and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, in 1987 and 1991, respectively.

From April 1991 to June 1991, she was an Adjunct Assistant Professor with the Department of Electrical Engineering, The Ohio State University, Columbus. From July 1991 to June 1993, she was an Assistant Professor with the Department of Electrical Engineering, University of Kentucky, Lexington. Since August 1993, she has been with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, where she is currently a Professor. In Fall 2007, she was on sabbatical leave at MIT, Cambridge, Massachusetts. Her current research interests include spectral estimation, statistical and array signal processing, and their applications.

She is a Fellow of IEEE and a Fellow of IET. She is a member of Sigma Xi and Phi Kappa Phi. She received the 1994 National Science Foundation Young Investigator Award and the 1996 Office of Naval Research Young Investigator Award. She was an Executive Committee Member of the 2002 International Conference on Acoustics, Speech, and Signal Processing, Orlando, Florida, May 2002. She was an Associate Editor of the IEEE Transactions on Signal Processing from 1999 to 2005, an Associate Editor of the IEEE Signal Processing Magazine from 2003 to 2005, and a member of the Editorial Board of Signal Processing, a publication of the European Association for Signal Processing (EURASIP), from 2005 to 2007. She was a member of the Editorial Board of the IEEE Signal Processing Magazine from 2010 to 2012. She was a member of the Editorial Board of Digital Signal Processing—A Review Journal, a publication of Elsevier, from 2006 to 2012. She is a co-author of the papers that have received the First and Second Place Best Student Paper Awards, respectively, at the 2005 and 2007 Annual Asilomar Conferences on Signals, Systems, and Computers in Pacific Grove, California. She is a co-author of the paper that has received the M. Barry Carlton Award for the best paper published in IEEE Transactions on Aerospace and Electronic Systems in 2005. She is also a co-author of the paper that has received the Lockheed Martin Best Student Paper Award at the 2009 SPIE Defense, Security, and Sensing Conference in Orlando, Florida.



# Introduction to Statistical Signal Processing

# 1

**Abdelhak M. Zoubir**

*Signal processing Group, Technische Universität Darmstadt, Germany*

---

### 3.01.1 A brief historical recount

Signals are either random in nature or deterministic, but subject to random measurement errors. Therefore, Statistical Methods for Signal Processing or in short *Statistical Signal Processing* are the signal processing practitioner's choice to extracting useful information from measurements. Today, one can confidently claim that statistical signal processing is performed in every specialization of signal processing. Early stages of statistical signal processing concerned parameter (signal) estimation, signal detection, and signal classification. These have their roots in probability theory and mathematical statistics. For example, parameter estimation and signal detection are what is known under, respectively, point estimation [1] and statistical hypothesis testing [2,3] in mathematical statistics while fundamental concepts of classification were treated in classical texts such as [4].

It is difficult to trace back the time when the term *statistical signal processing* was established. Undoubtedly, statistical signal processing was performed even before the birth of digital signal processing, which started soon after the discovery of the Fast Fourier Transform by Cooley and Tukey [5]. Themes on statistical signal processing started as part of broader workshops in the early 1980s, such as the IEEE Workshop on Spectrum Estimation and Modeling, the IEEE International Workshop on Statistical Signal and Array Processing, or the IEEE International Workshop on Higher-Order Statistics, to mention a few. The year 2001 gave birth to the IEEE International Workshop on Statistical Signal Processing, which is organized biannually. Textbooks on the subject have been around since the 1990s. They include, for example [6–8].

---

### 3.01.2 Content

Today, statistical signal processing finds a wide range of cross-fertilization and applications, far beyond signal estimation, detection, or classification. Examples include communications and networking, target tracking, (adaptive) filtering, multi-dimensional signal processing, and machine learning, to mention a few. Recently, I came across a few newly edited books with the content statistical signal processing for neuroscience. The authors of the chapters from both signal processing and neural computation aim at promoting interaction between the two disciplines of signal processing and neural sciences. This is surely not the first and not the last attempt to bring two communities together. Signal Processing, and in particular statistical signal processing, is such a dynamic discipline that calls for cross-fertilization

between disciplines. At no surprise, you will find a wide range of statistical signal processing treatment in other sections of this e-reference, such as in *Signal processing for Communications* or *Signal Processing for Machine Learning*.

I was honored and delighted when Sergios Theodoridis approached me to be the area editor for this section. I first thought of key advances in statistical signal processing. They are numerous, but space did not allow for all of these areas to be covered in this section. I had to select a subset and I approached not only the very best researchers in these areas, but who also are experienced in writing tutorial-style articles. Some of the contributors preferred to wait for the second edition of e-reference for their manuscript to appear.

---

### 3.01.3 Contributions

The contributions in this section cover chapters on recent advances in detection, estimation and applications of statistical signal processing. All of these chapters are written in a tutorial style.

#### 3.01.3.1 Quickest change detection

An important problem in engineering practice, such as engine monitoring, is to detect anomalies or changes in the environment as quickly as possibly, subject to false alarm constraints. These changes can be captured by a change in distribution of the observed data. The authors provide two formulations of quickest change detection, a Bayesian and a minimax approach, along with their (asymptotically) optimal solutions. The authors also discuss decentralized quickest change detection and provide various algorithms that are asymptotically optimal. Several open problems in the context of quickest change detection have been given by the authors. Among these, the authors mention an old, but to date an unsatisfactorily solved problem with a large impact in statistical signal processing practice, that is, transient detection.

#### 3.01.3.2 Distributed signal detection

Signal detection with a single sensor is a well-established theory with a wide range of applications, such as in radar, sonar, communications, or biomedicine. Today, we encounter an enormous growth of multi-sensor based detection. For example, in wireless sensor networks, one aims at making a global decision based on local decisions. The deployment of multiple sensors for signal detection improves system survivability, results in improved detection performance or in a shorter decision time to attain a preset performance level. In classical multi-sensor detection, local sensors transmit their raw data to a central processor where optimal detection is carried out. This has its drawbacks, including high communication bandwidth. Distributed processing has its advantage in that local sensors with low energy consumption carry out preliminary processing and communicate only the information relevant to the global objective, such as the decision on the presence or absence of a target in radar. This leads to low energy consumption, reduced communication bandwidth, and increases system reliability. The chapter on distributed signal detection provides a survey and most recent advances in distributed detection, such as distributed detection in the presence of dependent observations, using copula theory.

### 3.01.3.3 Diffusion adaptation over networks

Wireless sensor networks, which consist of spatially distributed autonomous sensors, are becoming fundamental to engineering practice. The sensors or agents in the network have the task to monitor physical or environmental conditions, such as temperature, pressure, or vibrations. Cooperatively, these sensors reach a global decision. The chapter on *Diffusion Adaptation over Networks* approaches the problem of global inference using adaptation and learning, which are important abilities of the collection of agents that are linked together through a connection topology. Adaptive networks are well suited to performing decentralized information processing and optimization in real time. One of the advantages of such networks is the continuous diffusion of information across the network that enables adaptation of performance in relation to changing data and network conditions. This overview article on diffusion strategies for adaptation and learning over networks provides fundamental principles and articulates the improved adaptation and learning performance of such networks relative to non-cooperative networks.

### 3.01.3.4 Non-stationary signal analysis—a time-frequency approach

Non-stationary signal analysis plays an important role in statistical signal processing. For example, in analyzing automotive engine signals, classical spectral analysis approaches fail as they do not capture the non-stationary nature of signals due to motion of the piston. Linear time-frequency approaches, such as the spectrogram, capture the non-stationary nature of signals whose spectral contents vary with time. This class of time-frequency representation has its advantages, but also its drawbacks. Quadratic time-frequency representations, although they lose the linearity property, have the advantage of providing a higher time-frequency concentration as compared to linear methods. Also, higher-order time-frequency representations have been proposed as they further improve the time-frequency concentration for a certain class of non-stationary signals. In this chapter, Ljubiša Stanković et al. provide an overview of state-of-the-art methods for non-stationary signal analysis and their applications to real-life problems, including inverse synthetic aperture radar.

### 3.01.3.5 Bayesian computational methods in signal processing

There are two schools of thoughts in statistical inference, i.e., the frequentist and the Bayesian approaches. This chapter deals with Bayesian inference. The author first illustrates Bayesian inference through the linear Gaussian model. This model makes many of the required calculations straightforward and analytically computable. He then considers the practically more relevant problem where there are intractable elements in the models. This problem can only be solved numerically. There is a wide range of computational tools available for solving complex Bayesian inference problems, ranging from simple Laplace approximations to posterior densities, through variational Bayes' methods to highly sophisticated Monte Carlo schemes. The author gives a flavor of some of the techniques available today, starting with one of the simplest and most effective: the Expectation-Maximization algorithm. He then describes Markov Chain Monte Carlo (MCMC) methods, which have gained much importance in solving complicated problems, and concludes with the emerging topic of particle filtering in statistical signal processing.

### 3.01.3.6 Model order selection

A problem encountered again and again in statistical signal processing is model selection. The signal processing practitioners require a simple, but effective means to deciding on a model within a family of models, given measurements. This problem is known as model selection. There exist a wealth of methods available to solving this problem. However, for a given set of data different model selection procedures give different results. For this reason, model selection and model order selection are still an active area of research in statistical science as well as statistical signal processing. The authors of this chapter describe the basic principles, challenges, and the complexity of the model selection problem. They treat statistical inference-based methods in detail, and those techniques as well as their variants, widely used by engineers. The chapter concludes with a practical engineering example in determining the dimension of the signal subspace, a problem encountered in sensor array processing and harmonic retrieval.

### 3.01.3.7 Performance analysis and bounds

Bounds provide fundamental limits on estimation given some assumptions on the probability laws and a model for the parameters of interest. In his chapter, Brian Sadler considers performance analysis of estimators, as well as bounds on estimation performance. He introduces key ideas and avenues for analysis, referring to the literature for detailed examples. He seeks to provide a description of the analytical procedure, as well as to provide some insight, intuition, and guidelines on applicability and results.

### 3.01.3.8 Geolocation

Geolocation denotes the position of an object in a geographical context. As Frederik describes it, geolocation is characterized by the four Ms, which are the Measurements used, the Map, the Motion model used for describing the motion of the object, and the filtering Method. He describes a general framework for geolocation based on the particle filter. He generalizes the concept of fingerprinting for describing the procedure of fitting measurements (along a trajectory) to the map. Several examples based on real data are used to illustrate various combinations of sensors and maps for geolocation. He finally discusses different ways as to show how the tedious mapping steps can be automated.

---

## 3.01.4 Suggested further reading

Some readers will be inspired by the collection of chapters in this section and would want to deepen further their knowledge and apply some of the techniques to their own signal processing problems. Those readers are encouraged to consult textbooks, such as the ones provided in the list of references in this introduction or the ones specially tailored to the subject contained in this section for further reading. I also hope that the readers will find inspirations in other tutorials on the above topics published earlier in the *IEEE Signal Processing Magazine* or *The Proceedings of the IEEE*.

---

## Acknowledgments

I am grateful to the contributors as well as the colleagues who provided a critical feedback on those contributions.

## References

- [1] E.L. Lehmann, Theory of Point Estimation, Wadsworth & Brooks/Cole Advanced Books & Software, 1983.
- [2] E.L. Lehmann, Testing Statistical Hypotheses, John Wiley & Sons, Inc., New York, 1959.
- [3] J. Neyman, E.S. Pearson, On the problem of the most efficient tests of statistical hypotheses, *Philos. Trans. R. Soc., Ser. A* 231 (1933) 289–337.
- [4] T.W. Anderson, An Introduction to Multivariate Statistical Analysis, John Wiley & Sons, Inc., New York, 1958.
- [5] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex Fourier series, *Math. Comput.* 19 (90) (1965) 297–301.
- [6] S.M. Kay, Fundamentals of Statistical Signal Processing, Estimation Theory, vol. I, Prentice-Hall, 1993.
- [7] S.M. Kay, Fundamentals of Statistical Signal Processing, Detection Theory, vol. II, Prentice-Hall, 1998.
- [8] L.L. Scharf, Statistical Signal Processing: Detection, Estimation, and Time Series Analysis, Addison Wesley, Boston, 1991.

# Model Order Selection

# 2

Visa Koivunen and Esa Ollila

Department of Signal Processing and Acoustics, School of Electrical Engineering, Aalto University, Finland

## 3.02.1 Introduction

In model selection the main task is to choose a mathematical model for a phenomenon from a collection of models given some evidence such as the observed data at hand. Informally, one aims at identifying a proper complexity for the model. Typical application examples include finding the number of radio frequency signals impinging an antenna array, choosing the order of an ARMA model used for analyzing time series data, variable selection in the regression model in statistics, estimating the channel order in wireless communication receiver and selecting the order of a state-space model used in tracking a maneuvering target in radar.

Model order selection uses a set of observed data and seeks the best dimension for the parametric model used. This is related to the idea of the inverse experiment and the maximum likelihood estimation where one looks for the parameters that most likely have produced the observed data. Various methods for model selection or model order estimation stem from the detection and estimation theory, statistical inference, information and coding theory and machine learning. The outcome of a model order selection is an integer-value describing the dimension of the model.

One of the well known *heuristic* approach used in model selection is Occam's razor (e.g., [1]). It usually favors parsimony, i.e., simpler explanations instead of more complex ones while retaining the explanatory power of the model. Hence, Occam's razor chooses the simplest model among the candidate models. Simpler models tend to capture the underlying structure of the phenomenon (such as signal of interest) better, and consequently provide better performance in predicting the output when new observations are processed. More complex models tend to overfit the observed data and to be more sensitive to noise. Moreover, the model may not generalize well to new data. The same problem is addressed in statistical inference where there is a trade-off between bias and variance when explaining the observed data. Too simple models lead to a bigger systematic error (bias), while too complex models may lead to an increased variance of estimated parameters and higher Cramer-Rao Bound as well as to finding structures in the data that do not actually exist. Parsimony is also favored in the probability theory, since the probability of making an error will increase by introducing additional assumptions that may be invalid or unnecessary.

Traditionally it is assumed that there is a single correct hypothesis within the set of hypotheses considered that describe the (true) model generating the given data. In practice, any model that we assume will be only an approximation to the truth: for example, a data does not necessarily follow

an exactly normal distribution. Yet the described criteria are still useful in selecting the “best model” from the set which is most supported by the data within the mathematical paradigm of the selected information criteria. There are many practical consequences for a misspecified model order. In state-space models and Kalman filtering, models with too low order make the innovation sequence correlated and the optimality of the filter is lost. On the other hand, if the order is too high, one may end up with tracking the noise that can lead to divergence of the Kalman filter. In regression, models with too low order may lead to biased estimates since they do not have enough flexibility to fit the data with high fidelity. On the other hand, too many parameters increase the variance of the estimates unnecessarily and cause the loss of optimality.

Model selection is of fundamental interest to the investigator seeking a better understanding of the phenomenon under study and the literature on this subject is considerable. Several review articles [2–4] are noticeable among others as well as the full length text-books [1, 5, 6]. In this overview, we first review the classical model selection paradigms: the Bayesian Information Criterion (BIC) with an emphasis on its asymptotic approximation due to Schwarz [7], Akaike’s Information Criterion (AIC) [8, 9], stochastic complexity [10] and its early development, the Minimum Description Length (MDL) criterion due to Rissanen [11] and the generalized likelihood ratio test (GLRT-) based sequential hypothesis testing (see, e.g., [12]). Among the methods considered, the MDL and AIC have their foundations in the information theory and coding whereas BIC and GLRT methods have their roots in major statistical inference paradigms (Bayesian and likelihood principle). The AIC is based on the asymptotic approximation of the Kullback-Leibler (K-L) divergence, a fundamental notion in information theory, whereas Rissanen’s MDL is based on the coding theory. However, both of these methods can also be linked with the likelihood principle and the classification of the statistical inference and the information theory based methods is provided mainly for pedagogic and informative purpose. There are some more recent contributions such as the exponentially embedded family (EEF) method by Kay [13] and Bozdogan’s extensions of AIC [14] among others which are not reviewed herein.

This chapter is organized as follows. First, Section 3.02.2 describes the basic principles, challenges and the complexity of the model selection problems with a simple example: variable selection in the multiple linear regression model. Especially, the need for compromise with the goodness of fit and concision (“principle of parsimony”), which is the key idea of model selection, is illustrated and the forward stepwise regression variable selection principle using the AIC is explained along with the cross-validation principle. Section 3.02.3 addresses statistical inference based methods (BIC, GLRT) in detail. Section 3.02.4 introduces the AIC, the MDL and its variants and enhancements. In Section 3.02.5, the model selection methods are derived and illustrated using a practical engineering example in determining the dimension of the signal subspace; such problems arise commonly in sensor array processing, and in harmonic retrieval; see, e.g., [15–18].

**Notations.** More specifically, model selection refers to the multihypothesis testing problem of determining which model best describes a given *data (measurement)*  $\mathbf{y}$  of size  $N$ . Let  $\mathcal{H}_m$ ,  $m = 1, \dots, M$ , denote the set of hypotheses which may not be nested (i.e.,  $\mathcal{H}_1 \not\subset \mathcal{H}_2 \subset \dots \not\subset \mathcal{H}_M$ ), and that each hypothesis specifies a probability density function (p.d.f.)  $f_m$  with a parameter vector  $\theta$  of dimension  $K = \dim(\theta|\mathcal{H}_m)$  in the parameter space  $\Theta$ . For the notational convenience we often suppress the subscript  $m$  (the model index) from  $\theta$ ,  $K$ , and  $\Theta$  but the reader should keep in mind that these differ for each model. *Model order selection* refers to finding the true dimension  $K$  of the model commonly in

the nested set of hypotheses. Let

$$\ell_m(\theta|\mathbf{y}) \triangleq \ln f_m(\mathbf{y}|\theta) \quad \text{and} \quad p_m \triangleq \mathbb{P}(\mathcal{H}_m) \equiv \mathbb{P}(\mathbf{y} \sim f_m)$$

denote the *log-likelihood* function of the data under the  $m$ th model and the *a priori probability* of the  $m$ th hypothesis ( $\sum_{m=1}^M p_m = 1$ ), respectively. Usually, all models are a priori equally likely, in which case  $p_m = 1/M$  for  $m = 1, \dots, M$ . In the likelihood theory, the natural point estimate of  $\theta$  is the *maximum likelihood estimate* (MLE)  $\hat{\theta} \triangleq \arg \max_{\theta \in \Theta} \ell_m(\theta|\mathbf{y})$  which is assumed to be unique. Let us denote by

$$\mathcal{J}_m \equiv \mathcal{J}_m(\theta) \triangleq -E \left[ \frac{\partial^2 \ell_m(\theta|\mathbf{y})}{\partial \theta \partial \theta^T} \right] \quad \text{and} \quad \widehat{\mathcal{J}}_m \triangleq -\left. \frac{\partial^2 \ell_m}{\partial \theta \partial \theta^T} (\theta|\mathbf{y}) \right|_{\theta=\hat{\theta}} \quad (2.1)$$

the (expected) *Fisher information matrix (FIM)* and the *observed FIM*, respectively.

### 3.02.2 Example: variable selection in regression

In the multiple linear regression model the data  $\mathbf{y}$  is modeled as  $\mathbf{y} = X\beta + \boldsymbol{\varepsilon}$ , where  $\beta \in \mathbb{R}^k$  denotes the unknown vector of regression coefficients,  $X$  is the known  $N \times k$  matrix of explanatory variables ( $k < N$ ) and  $\boldsymbol{\varepsilon}$  is the  $N$ -variate *noise vector* with i.i.d. components from normal distribution  $\mathcal{N}(0, \sigma^2)$  with an unknown variance  $\sigma^2 > 0$ . Let us call the columns of  $X$  (explanatory) *variables* following the convention in statistics, i.e., we use  $X_i$  to denote both the column of  $X$  and the  $i$ th variable of the model. Thus,  $\mathbf{y} \sim \mathcal{N}_N(X\beta, \sigma^2 I)$  and the p.d.f. and the log-likelihood are

$$f(\mathbf{y}|\theta) = (2\pi)^{-N/2} (\sigma^2)^{-N/2} \exp \left\{ -\frac{1}{2\sigma^2} \|\mathbf{y} - X\beta\|^2 \right\},$$

$$\ell(\theta|\mathbf{y}) = -\frac{N}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} \|\mathbf{y} - X\beta\|^2,$$

where  $\theta = (\beta^T, \sigma^2)^T$  denotes the parameter vector. Commonly there is a large set of explanatory variables and the main interest is choosing the subset of explanatory variables that significantly contribute to the data (response variable)  $\mathbf{y}$ ; see [19, 20]. Thus, we consider the hypotheses

$$\mathcal{H}_I : \text{Only the subset of variables, say } X_{i_1}, \dots, X_{i_m}, \text{ contribute to } \mathbf{y},$$

where  $I = (i_1, \dots, i_m)$ ,  $1 \leq i_1 < i_2 < \dots < i_m \leq k$ , denotes the index set of the contributing variables. Then under  $\mathcal{H}_I$ , the MLEs of the regression coefficient and the noise variance correspond to the classic least squares (LS-)estimates  $\hat{\beta} = (X_I^T X_I)^{-1} X_I^T \mathbf{y}$  and the residual sample variance,  $\hat{\sigma}^2 = (1/N)\text{RSS}(\hat{\beta})$ , respectively, where  $\text{RSS}(\hat{\beta}) = \|\mathbf{y} - X_I \hat{\beta}\|^2$  is the residual sum of squares. Hence the  $-2 \times$  (maximum log-likelihood) under hypothesis  $\mathcal{H}_I$  becomes

$$-2 \times \ell(\hat{\theta}|\mathbf{y}) = N \ln \hat{\sigma}^2 + \frac{1}{\hat{\sigma}^2} \|\mathbf{y} - X_I \hat{\beta}\|^2 = N \ln \hat{\sigma}^2 + N, \quad (2.2)$$

which always decrease when we include additional variables in the model. This is because  $\hat{\sigma}_{+1}^2 \leq \hat{\sigma}^2$ , where  $\hat{\sigma}_{+1}^2$  denotes the residual variance with one additional variable in the model. Then,  $\hat{\sigma}^2 = 0$  if we introduce as many explanatory variables as responses. More generally, if two models that are compared are nested, then the one with more parameters will always yield a larger maximum log-likelihood, even though it is not necessarily a better model.

### 3.02.2.1 AIC and the stepwise regression

Previously, we noticed that minimizing  $-2 \times \ell(\hat{\theta}|\mathbf{y})$  cannot be used as the criterion. An intuitive approach is to introduce an additive term that penalizes *overfitting*. The penalty term needs to be an increasing function of  $K = \dim(\theta|\mathcal{H}_I) = m + 1$ , so that the criterion  $-2 \times (\text{maximum log-likelihood})$  decreases as the penalty term increases yielding a trade-off between goodness of fit and simplicity. This *principle of parsimony* is the fundamental idea of model selection. As we shall see, the classic approximations of information criteria reduce to such forms. For example, the AIC can be stated as

$$\text{AIC} = -2 \times (\text{maximum log-likelihood}) + 2 \times (\text{number of estimable parameters}),$$

that is,  $\text{AIC} = -2\ell(\hat{\theta}|\mathbf{y}) + 2K$ , and the regression model (hypothesis  $\mathcal{H}_I$ ) that minimize the criterion is chosen. Here, the penalty term is  $2K$  which increases with  $K$  and hence penalizes selecting too many variables. In the regression setting,

$$\text{AIC}(m) = N \ln \hat{\sigma}^2 + 2m,$$

where we have suppressed the irrelevant additive constant  $N$  from (2.2) and the “plus one” term from  $K = m + 1$  which does not depend on the number of explanatory variables  $m = |I|$  in the model. Note that the robust versions of AIC have been proposed, e.g., [21] which utilizes robust  $M$ -estimation.

The AIC is not computationally feasible for variable selection when the number of explanatory variable candidates  $k$  is large. We need to consider all index sets  $I$  of size  $|I| = m$  for each  $m = 1, \dots, k$  and for each  $m$ , there are  $\binom{k}{m}$  possible index sets. More practical and commonly used approach is to use *stepwise regression* variable selection method. In the forward stepwise search strategy one starts with a model containing only an intercept term (or a subset of explanatory variables), and then includes the variable that yields the largest reduction in the AIC, i.e., the largest reduction in the RSS (and thus improving the fit the most). One stops when no variable produces a normalized reduction greater than a threshold. There are several variants of stepwise regression which differ based on the criterion used for inclusion/exclusion of variables. For example, instead of AIC, Mallows Cp, BIC, etc. can be used. One particularly simple stepwise regression is to include in each step the variable that is most correlated with the current residual  $\mathbf{r} = \mathbf{y} - \hat{\mathbf{y}}$ , where  $\hat{\mathbf{y}}$  is the LS fit based on the current selection of variables. This strategy is called the *orthogonal matching pursuit* [22] in the engineering literature. Some modern approaches for variable selection in regression that can be related to the forward stepwise regression are the LASSO (least absolute shrinkage and selection operator) [23] and the LARS (least angle regression) [24]. LASSO minimizes the usual LS criterion (the sum of squared residuals) with an additional  $L_1$ -norm constraint on the regression coefficient. As a result, some regression coefficients are shrunk to(wards) zero. LARS is a forward stepwise regression method where the selection of variables is done differently: namely a regression coefficient of the variable is increased until that variable is no longer the one most correlated with the residual  $r$ . A simple modification of the LARS algorithm can be used to compute the LASSO solution [24].

### 3.02.2.2 Cross-validation and bootstrap methods

Cross-validation (CV) is a widely used technique for model order selection in machine learning, regression analysis, pattern recognition, and density estimation; see [25]. In CV one is interested in not only

finding a statistical model for the observed data but also in prediction performance for other independent data sets. The goal is to ensure that the model generalizes well, not adapting too much to the particular data set at hand. Unlike many other model selection techniques, CV can be considered a computer-based heuristic method that replaces some statistical analyses by repeated computations over subsets of data.

In CV, one splits the data set into subsets, one *training* set for the initial statistical analysis and the other (potentially multiple) subsets for *testing* or validating the analysis. If the test data are independent and identically distributed (i.i.d.), they can be viewed as new data in prediction. If the sample size for the training set is small or the dimension of the initial model is high, it is likely that the obtained model will fit the set of observations with too high fidelity. Obviously, this specific model will not fit the other test data sets equally well and the prediction estimates produced by the initial model using the training set may contain significantly off-values than expected ones.

The prediction results from the testing (validation) sets are used to evaluate the goodness of the model optimized for the training set. The original model may then be adjusted by combining with the models estimated from the test sets to produce a single model, for example by averaging. CV may also be used in cases where the response is indicator of a class membership instead of being continuous valued. This is the case in  $M$ -ary detection problems and pattern recognition, for example.

There are many ways to subdivide the data into training and test sets. The assignment may be random such that the training set and the test set have equal number of observations. These two data sets may be used for training and testing in an alternating manner. A very commonly used approach is so-called leave-one-out (LOO) cross-validation. Each individual observation in the data set is serving a test set at its turn while the remaining data form the training set. This is done exhaustively until all the observations have been used as a test set. It is obvious that LOO has a high computational complexity since the training set is different and training process is repeated every time.

Cross-validation can also be used for model order selection in regression. Let us assume that we have a “supermodel” of large number of explaining variables and we want to select which subset of explaining or predictor variables gives the best model in terms of its prediction performance. By using the cross-validation approach with the test sets of observations one may find a subset of explaining variables that gives the best prediction performance measured, for example, in terms of mean square error. If cross-validation is not used, adding more predictor variables in the model decreases the risk or error measure, such as mean square error, sum of squared residuals or classification error for the training set but the predictive performance may be very different.

A few remarks on the statistical performance of the CV method are in place. CV gives a too positive view on the quality of the model optimized for the training set. The risk or error measure such as MSE will be smaller for the training set than the one for the test set. The variance of the cross-validation estimates depends on the data splitting scheme and it is difficult to analyze the performance of the method in a general case. However, it has been established in some special cases, see [25]. If the model is estimated for each subset of test data, the values of estimates will vary (estimator is a random variable).

The CV estimators often have some bias that depends on the size of the training sample relative to the whole sample size. It tends to decrease as the training sample size increases. For the same size of training set the bias remains the same but differences may be attested in variance (accuracy). The overall performance of the CV estimators depends on many factors including the size of training data set, validation data set, and the number of data subsets.

Bootstrapping [26] is a resampling method that is typically used for characterizing confidence intervals. It has found applications in hypothesis testing and model order estimation as well, see, e.g., [27]. Assuming independent and identically distributed (i.i.d.) data samples, it resamples the observed data set with replacement to simulate new data called bootstrap data. Then the parameter of interest such as the variance or the mean is computed for the bootstrap sample. It is a computer-based method since it requires that a very large number of resamples are drawn, and bootstrap estimates for each bootstrap sample are computed. Availability of highly powerful computing machinery has made the use of bootstrap more feasible in different applications as well as allowed to increase the sample sizes. The bootstrap method does not make any assumptions on the underlying distribution of the observed data. It is a nonparametric method and allows for characterizing the distribution of the observed data. This is particularly useful for non-Gaussian data where the distribution may not be symmetric or unimodal. An example of model order estimation in a sensor array signal processing application is provided in [27].

### 3.02.3 Methods based on statistical inference paradigms

In this section, we review the model selection methods based on two main stream statistical inference paradigms (Bayesian and likelihood principles). The widely used asymptotic approximation of the Bayesian Information Criterion (BIC) due to Schwarz [7] is reviewed first, followed by the GLRT-based model selection method which is based on the sequential use of the generalized likelihood ratio test (GLRT). The latter method is one of the oldest approaches in model selection.

#### 3.02.3.1 Bayesian Information Criterion (BIC)

We start by recalling the steps and assumptions that lead to the classic asymptotic approximation of BIC due to Schwarz [7] which can be stated as follows:

$$\text{BIC} = -2 \times (\text{maximum log-likelihood}) + \ln N \times (\text{number of estimable parameters})$$

so that  $\text{BIC} = -2\ell(\hat{\theta}|\mathbf{y}) + (\ln N)K$  and the hypothesis  $\mathcal{H}_m$  ( $m$ th model) that minimizes the criterion is to be chosen. This approximation is based on the second-order Taylor expansion of the log posterior density and some regularity assumptions on the log-likelihood and log posterior. Recall from earlier discussion on the nested hypotheses, the term  $-2 \times (\text{maximum log-likelihood})$  monotonically decreases with an increasing  $m$  (model index), and cannot be used alone as it leads to the choice of the model with the largest number of parameters. The second term is the penalty term that increases with  $K = \dim(\theta|\mathcal{H}_m)$  and hence penalizes the use of too many parameters.

In the Bayesian framework, the parameter vector is taken to be a random vector (r.v.) with a prior density  $q_m(\theta)$ ,  $\theta \in \Theta$ . For the  $m$ th model, let

$$g_m(\theta|\mathbf{y}) \triangleq \ell_m(\theta|\mathbf{y}) + \ln q_m(\theta)$$

denote the *log posterior density* (neglecting the irrelevant additive constant term). The full BIC is obtained by integrating out the nuisance variable  $\theta$  in the posterior density of  $\theta$  given the data  $\mathbf{y}$  and the model  $m$ :

$$\text{BIC}(m) \triangleq p_m \int_{\Theta} \exp\{g_m(\theta|\mathbf{y})\} d\theta. \quad (2.3)$$

The  $m$ th model that maximizes  $\text{BIC}(m)$  over  $m = 1, \dots, M$  is selected for the data. In most cases, the integration cannot be solved in a closed-form and approximations are sought for.

In Bayesian approach, the maximum-a-posteriori probability (MAP) estimate  $\hat{\theta}_B \triangleq \arg \max_{\theta \in \Theta} g_m(\theta|\mathbf{y})$  is a natural point estimate of  $\theta$  and assume that it is unique. An approximation of  $g(\cdot|\mathbf{y})$  around the MAP estimate based on second-order Taylor expansion yields

$$g_m(\theta|\mathbf{y}) \approx g_m(\hat{\theta}_B|\mathbf{y}) - \frac{1}{2}(\theta - \hat{\theta}_B)^T \widehat{H}_m(\theta - \hat{\theta}_B); \quad \widehat{H}_m \triangleq \frac{\partial^2 g_m}{\partial \theta \partial \theta^T}(\theta|\mathbf{y})|_{\theta=\hat{\theta}_B}.$$

Note that the first-order term vanishes when evaluated at the maximum. Let us assume that  $g_m$  is twice continuously differentiable ( $g_m \in C^2(\Theta)$ ). This means that its second-order partial derivatives w.r.t.  $\theta$  exist on  $\Theta$  and that the resulting  $K \times K$  (negative) Hessian matrix  $\widehat{H}_m$  is symmetric. Substituting the approximation into (2.3) yields

$$\begin{aligned} \text{BIC}(m) &\approx p_m \exp\{g_m(\hat{\theta}_B|\mathbf{y})\} \int_{\theta_m} \exp\left\{-\frac{1}{2}(\theta - \hat{\theta}_B)^T \widehat{H}_m(\theta - \hat{\theta}_B)\right\} d\theta \\ &= p_m \exp\{g_m(\hat{\theta}_B|\mathbf{y})\} (2\pi)^{K/2} |\widehat{H}_m|^{-1/2}, \end{aligned}$$

where the last equality follows by noticing that the integrand is the density of  $K$ -variate  $N_K(\hat{\theta}_B, \widehat{H}_m^{-1})$  distribution up to a constant term  $(2\pi)^{-K/2} |\widehat{H}_m|^{1/2}$ , where  $|\cdot|$  denotes the matrix determinant. Equivalently, we can minimize the log of BIC. In logarithmic form, the BIC becomes

$$\text{BIC}(m) \approx \ln p_m + g_m(\hat{\theta}_B|\mathbf{y}) + \frac{K}{2} \ln(2\pi) - \frac{1}{2} \ln |\widehat{H}_m|. \quad (2.4)$$

An alternative approximation for the BIC can be derived with the following assumption:

- (A)** the prior distribution  $q_m(\theta)$  is sufficiently flat around the MAP estimate  $\hat{\theta}_B$  and does not depend on  $N$  for  $m = 1, \dots, M$ .

Due to (A), when evaluating the integral in (2.3), we may extract the  $q_m(\theta)$  term from the integrand as a constant proportional to  $q_m(\hat{\theta}_B)$ , yielding the approximation

$$\text{BIC}(m) \approx p_m q_m(\hat{\theta}_B) \int_{\theta_m} \exp\{\ell_m(\theta|\mathbf{y})\} d\theta, \quad (2.5)$$

$$\approx p_m q_m(\hat{\theta}_B) \exp\{\ell_m(\hat{\theta}|\mathbf{y})\} (2\pi)^{K/2} |\widehat{\mathcal{J}}_m|^{-1/2}, \quad (2.6)$$

where  $\widehat{\mathcal{J}}_m$  is the Fisher information matrix of the  $m$ th model evaluated at the unique MLE  $\hat{\theta}$ . Note that (2.6) is an approximation of (2.5) using (similarly as earlier) the second-order Taylor expansion on  $\ell_m(\theta|\mathbf{y})$  around the MLE. Thus in terms of logarithms, we have

$$\text{BIC}(m) \approx \ln p_m + \ln q_m(\hat{\theta}_B) + \frac{K}{2} \ln(2\pi) + \ell_m(\hat{\theta}|\mathbf{y}) - \frac{1}{2} \ln |\widehat{\mathcal{J}}_m|. \quad (2.7)$$

Note that approximations (2.4) and (2.7) differ as the former (resp. the latter) is obtained for the case that the full log posterior (resp. log-likelihood) is expanded to the second-order terms. Furthermore, (2.7) relies upon (A).

Next, note that under some regularity conditions and due to consistency of the MLE  $\hat{\theta}$ , we can in most cases approximate  $\ln |\widehat{\mathcal{J}}_m|$  for a sufficiently large  $N$  as

$$\ln |\widehat{\mathcal{J}}_m| \approx K \ln N. \quad (2.8)$$

Since the three first terms in (2.7) are independent of  $N$  [recall assumption (A)], we may neglect the first three terms in (2.7) for large  $N$ , yielding after multiplying by  $-2$  the earlier stated classic approximation due to Schwarz,

$$\text{BIC}(m) = -2\ell_m(\hat{\theta}|\mathbf{y}) + K \ln N. \quad (2.9)$$

Let us recall the assumptions for the above result. First of all, the log-likelihood function  $\ell$  ought to be a  $C^2$ -function around the neighborhood of the unique MLE  $\hat{\theta}$ . Second, the prior density is assumed to verify Assumption (A). Third, sufficient regularity conditions based on likelihood theory, e.g., consistency of  $\hat{\theta}$ , are needed for the approximation (2.8) to hold. Note that the classic BIC approximation, in contrast to approximations (2.4) or (2.7), is based on asymptotic arguments, i.e.,  $N$  is assumed to be sufficiently large. Naturally, how large  $N$  is required, depends on the application at hand. The benefit of such an approximation is its simplicity and good performance observed in many applications. As we shall see in Section 3.02.4, Rissanen's MDL derived using arguments in coding theory can be asymptotically approximated in the same form as BIC in (2.9). This sometimes leads to misunderstanding that MDL and BIC are the same.

If we assume that the set of hypotheses  $\{\mathcal{H}_m\}$  includes the true model  $\mathcal{H}_{m^*}$  that generated the data, then the probability that the true  $m^*$ th model is selected can often be calculated analytically. For example, in the regression setting, the BIC estimate has proven to be consistent [28], that is,  $\mathbb{P}(\text{"correctly choosing } \mathcal{H}_{m^*}\text{"}) \rightarrow 1$  as  $N \rightarrow \infty$ . This can be contrasted with the AIC estimate that fails to be consistent, due to a tendency of overfitting [28].

### 3.02.3.2 GLRT-based sequential hypothesis testing

Let us assume that sequence of model p.d.f.'s  $f_m(\mathbf{y}|\theta)$ ,  $\theta \in \Theta_m$ , belong to the same parametric family (i.e., share the same functional form, so  $f = f_m$  for  $m = 1, \dots, M$ ) and are ordered in the sense that  $\Theta_m \subset \Theta_{m+1}$ . Hence, we have a nested set of hypotheses  $\mathcal{H}_1 \subset \mathcal{H}_2 \subset \dots \subset \mathcal{H}_M$ . We can consider  $\mathcal{H}_M$  as the “full model” and  $\mathcal{H}_m$ ,  $m < M$ , a parsimonious model presuming that a subset of the parameters of the full model are equal to zero. This type of situation can arise in many engineering applications; a practical application is given in Section 3.02.5. In such a scenario, the GLRT statistic, defined as the  $-2 \times$  the log of the ratio of the likelihood functions maximized under  $\mathcal{H}_m$  and under the full model  $\mathcal{H}_M$ :

$$\Lambda_m = -2\{\ell(\hat{\theta}_m|\mathbf{y}) - \ell(\hat{\theta}_M|\mathbf{y})\} = -2 \ln \left( \frac{\max_{\theta \in \Theta_m} f(\mathbf{y}|\theta)}{\max_{\theta \in \Theta_M} f(\mathbf{y}|\theta)} \right), \quad (2.10)$$

where  $\hat{\theta}_m = \arg \max_{\theta \in \Theta_m} \ell(\theta|\mathbf{y})$  and  $\hat{\theta}_M = \arg \max_{\theta \in \Theta_M} \ell(\theta|\mathbf{y})$  and  $\ell(\theta|\mathbf{y}) = \ln f(\mathbf{y}|\theta)$  denotes the log-likelihood. By the general maximum likelihood theory,  $\Lambda_m$  has a  $\chi^2_{\text{DOF}(m)}$  distribution asymptotically (i.e., chi-squared distribution with  $\text{DOF}(m)$  degrees of freedom), where

$$\text{DOF}(m) = \dim(\theta|\mathcal{H}_M) - \dim(\theta|\mathcal{H}_m).$$

The null hypothesis  $\mathcal{H}_m$  is rejected if  $\Lambda_m$  exceeds a rejection threshold  $\gamma_m$ , where  $\gamma_m$  is  $(1-\alpha)$ th quantile of the  $\chi^2_{\text{DOF}(m)}$  distribution (i.e.,  $\mathbb{P}(\Lambda_m > \gamma_m) = \alpha$  when  $\Lambda_m \sim \chi^2_{\text{DOF}(m)}$ ). Such a detector has an asymptotic probability of false alarm (PFA) equal to  $\alpha$ . For example, choosing PFA equal to  $\alpha = 0.05$  is quite reasonable for many applications; however, in radar applications, smaller PFA is often desired.

The sequential GLRT-based model selection procedure chooses the first hypothesis  $\mathcal{H}_m$  (and hence  $\dim(\theta|\mathcal{H}_m)$  as the model order) that is not rejected by the GLRT statistic. In other words, we first test  $\mathcal{H}_1$  and if it is rejected by the GLRT statistic (2.10) ( $\Lambda_1 > \gamma_1$  for some predetermined PFA  $\alpha$ ), then test  $\mathcal{H}_2$ , etc. until acceptance, i.e., until  $\Lambda_m \leq \gamma_m$ . In case that all hypotheses are rejected, then we accept the full model  $\mathcal{H}_M$ . Note that the above decision sequence is not statistically independent, even asymptotically. Hence, even though the asymptotic PFA of each individual GLRT for testing  $\mathcal{H}_m$  is  $\alpha$ , it does not mean that the asymptotic PFA of the GLRT-based model selection is  $\alpha$ . It is also possible to use a backward testing strategy, i.e., test the sequence of hypotheses  $\mathcal{H}_1, \dots, \mathcal{H}_{M-2}, \mathcal{H}_{M-1}$  from the reverse direction. Then we select the model in the sequence before the first rejection. That is, we first test  $\mathcal{H}_{M-1}$  and if it is accepted, then test  $\mathcal{H}_{M-2}$ . Continue until the first rejection and select the hypothesis tested prior the rejection. If it is known *a priori* that a large model order is more likely than a small model order, then the backward strategy might be preferred from pure computational point of view.

### 3.02.4 Information and coding theory based methods

The information and coding theory based approaches for model order selection are frequently used. We will consider two such techniques, Information Criterion proposed by Akaike [8,9] and Minimum Description Length (MDL) methods proposed by Rissanen [10,11]. The enhanced versions of both methods have been introduced in order to provide more accurate results for finite sample sizes and in some cases to relax unnecessary assumptions on underlying distribution models.

#### 3.02.4.1 Akaike Information Criterion (AIC)

The AIC criterion is based on an asymptotic approximation of the K-L divergence. Let  $f_0(\mathbf{y})$  denote the true p.d.f. of the data and let  $f(\mathbf{y}|\theta)$  denote the approximating model. The *Kullback-Leibler (K-L) divergence* (or relative entropy), can be defined as the integral (as  $f$  and  $f_0$  are continuous functions)

$$D(f_0\|f) = \mathbb{E}_0 \left[ \ln \left( \frac{f_0(\mathbf{y})}{f(\mathbf{y}|\theta)} \right) \right] = \int f_0(\mathbf{y}) \ln \left( \frac{f_0(\mathbf{y})}{f(\mathbf{y}|\theta)} \right) d\mathbf{y}$$

and it can be viewed as the information loss when the model  $f$  is used to approximate  $f_0$  or as a “distance” (discrepancy) between  $f$  and  $f_0$  since it verifies

$$D(f_0\|f) \geq 0 \quad \text{with equality if and only if } f_0 = f.$$

Clearly, among the models in the set of hypotheses  $\mathcal{H}_m : \mathbf{y} \sim f_m(\mathbf{y}|\theta_m)$ , the best one loses the least information (have smallest value of  $D(f_m\|f_0)$ ) relative to the other models. Note that the K-L divergence can be expressed as

$$D(f_0\|f) = \mathbb{E}_0[\ln f_0(\mathbf{y})] - \mathbb{E}_0[\ln f(\mathbf{y}|\theta)],$$

where the model dependent part is  $-\mathbb{E}_0[\ln f(\mathbf{y}|\theta)]$ . When finding the minimum of  $D(f_0\|f)$  over the set of models considered is equivalent to finding the maximum of the function

$$I(f_0, f) = \mathbb{E}_0[\ln f(\mathbf{y}|\theta)] = \mathbb{E}_0[\ell(\theta|\mathbf{y})]$$

among all the models. The function  $I(f_0, f)$ , sometimes referred to as *relative K-L information*, cannot be used directly in model selection because it requires the full knowledge of the true data p.d.f.  $f_0$  and the true value of the parameter  $\theta$  in the approximate model  $f(\mathbf{y}|\theta)$ , both of which are unknown. Hence we settle with its estimate

$$\hat{I}(f_0, f) = \mathbb{E}_{\mathbf{y}} \left[ \mathbb{E}_{\mathbf{w}} [\ln f(\mathbf{w}|\hat{\theta}(\mathbf{y}))] \right], \quad (2.11)$$

where  $w$  is an (imaginary) independent data vector from the same unknown true distribution  $f_0$  as the observed data  $y$  and  $\hat{\theta} = \hat{\theta}(\mathbf{y})$  denotes the MLE of  $\theta$  based on the assumed model  $f$  and the data  $\mathbf{y}$ . Note that the statistical expectations above are w.r.t. to the true (unknown) p.d.f.  $f_0$  and we maximize (2.11) instead of  $I(f_0, f)$ . Key finding is that for a large sample size  $N$ , the maximized log-likelihood  $\ell(\hat{\theta}|\mathbf{y}) = \ln f(\mathbf{y}|\hat{\theta})$  is a biased estimate of (2.11) where the bias is approximately equal to  $K$ , the number of the estimable parameters in the model. The maximum log-likelihood corrected by the asymptotic bias, i.e.,  $\ell(\hat{\theta}|\mathbf{y}) - K$ , can be viewed (under regularity conditions) as an unbiased estimator of  $\hat{I}(f, f_0)$  when  $N$  is large. Multiplying by  $-2$  gives the famous Akaike's Information Criterion

$$\text{AIC} = -2\ell(\hat{\theta}|\mathbf{y}) + 2K.$$

Unlike the BIC, the AIC is not consistent, namely the probability of correctly identifying the true model does not converge to one with the sample length. It can be shown that for nested hypotheses and under fairly general conditions,

$$\begin{aligned} \mathbb{P}(\text{"choosing } \mathcal{H}_k \subset \mathcal{H}_{\text{true}}") &\rightarrow 0, \\ \mathbb{P}(\text{"choosing } \mathcal{H}_k \supset \mathcal{H}_{\text{true}}") &\rightarrow c > 0 \quad \text{as } N \rightarrow \infty, \end{aligned}$$

that is, AIC tends to overfit, i.e., choosing model order that is larger than the true model order. See, e.g., [15].

Some enhanced versions have been introduced where the penalty term is larger to reduce the chance of overfitting. The following corrected AIC rule [29,30]

$$\text{AIC}_c = -2\ell(\hat{\theta}|\mathbf{y}) + 2K + \frac{2K(K+1)}{N-K-1}, \quad (2.12)$$

$$= -2\ell(\hat{\theta}|\mathbf{y}) + C_N \cdot 2K, \quad \text{where } C_N = \frac{N}{N-K-1} \quad (2.13)$$

provides a finite sample (second-order bias correction) adjustment to AIC; asymptotically they are the same since  $C_N \rightarrow 1$  when  $N \rightarrow \infty$ , but  $C_N > 1$  for finite  $N$ . Consequently the penalty term of  $\text{AIC}_c$  is always larger than that of AIC, which means that the criterion is less likely to overfit. On the other hand, this comes with an increased risk of underfitting, i.e., choosing a smaller model order than the true one. Since the underfitting risk does not seem to be unduly large in many practical experiments, many researchers recommend the use of  $\text{AIC}_c$  instead of AIC. It should be noted, however, that  $\text{AIC}_c$  possess

a sound theoretical justification only in the regression model with normal errors, where it is an unbiased estimate of the relative K-L information; note that AIC is by its construction only an asymptotically unbiased estimate of the relative K-L information. Besides the regression model, the reasonings for  $C_N$ , lay merely on its property of reducing the risk of overfitting. The so-called generalized information criteria (GIC) [2], defined as in (2.13) as

$$\text{GIC} = -2\ell(\hat{\theta}|\mathbf{y}) + C'_N \cdot 2K$$

can often outperform AIC for  $C'_N > 1$ . For example, when  $C'_N = C_N$  the  $\text{AIC}_c$  is obtained and the choice  $C'_N = (1/2) \log N$  yields the MDL criterion.

### 3.02.4.2 Minimum Description Length

The Minimum Description Length [11] method stems from the algorithmic information and coding theory. The basic idea is to find any regularity in the observed data that can be used to compress the data. This allows for condensing the data so that less symbols are needed to present it. The code used for compressing the data should be such that the total length (in bits) of description of the code and the description of the data is minimal. There exists a simple and refined version of the MDL method. The concept of *stochastic complexity* will be employed in estimating the model order. It defines the code length of the data with respect to the employed probability distribution model. It also establishes a connection to statistical model order estimation methods such as BIC and MDL. In some special cases, the approximation of stochastic complexity and BIC leads to the same criterion. Detailed descriptions of the MDL method can be found in [1,4].

The MDL method does not make any assumptions on the underlying distribution of the data. In fact, it is considered to be less important since the task at hand is not to estimate the distribution that generated the data but rather to extract useful information or model from the data that facilitates by compressing it efficiently. However, codes used for compressing data and the probability mass functions are related. The Kraft inequality establishes a correspondence between code lengths and probabilities whereas the information inequality relates the optimal code in terms of the minimum expected code length with the distribution of the data. If the data obeys the distribution  $f(\mathbf{y})$ , then the code with the length  $-\log f(\mathbf{y})$  achieves the minimum code length on average.

A simple version of the MDL principle may now be stated as follows. Let  $\mathcal{H}$  be a class of candidate models or codes used to explain the data  $\mathbf{y}$ . The best model  $H \in \mathcal{H}$  is the one that minimizes

$$\mathcal{L}(H) + \mathcal{L}(\mathbf{y}|H),$$

i.e., the sum of the length of the description of the model and the length of the data encoded using the model. Trading-off between complexity of the code and complexity of the data is built in the MDL method. Hence, it is less likely to overestimate the model order and overfit the data than AIC, for example.

A probability distribution where the parameter  $\theta$  may vary defines a family of distributions. The task of interest is to optimally encode the data and to use the whole family of distributions in order to make the process efficient. An obvious choice from the distribution family would be to use the code corresponding to  $f(\mathbf{y}|\hat{\theta})$  where  $\hat{\theta}$  is the MLE of  $\theta$  for the observed data, i.e.,  $\theta$  that most likely would have produced the data set. Consequently, the code corresponding to the maximum likelihood (or the probability model) depends on the data and the code cannot be specified before observing the data.

This makes this approach less practical. One could avoid this problem by using an universal distribution model that is representative of the entire family of distributions and would allow coding the data set almost as efficiently as the code associated with the maximum likelihood function. The refined MDL uses *normalized maximum likelihood* approach for finding an universal distribution model used to assist the coding. This will lead to an excessive code length that is often called *regret*. The Kullback-Leibler divergence  $D(p\|f)$  may be used as a starting point where  $p$  is employed to approximate the distribution  $f$ . Finding the optimal universal distribution  $p^*$  can be formulated as a minimax problem:

$$p^* = \arg \min_p \max_q \left[ \ln \frac{f(\mathbf{y}|\hat{\theta})}{p(\mathbf{y})} \right],$$

where  $q$  denotes the set of all feasible distributions. The distribution  $q(\mathbf{y})$  represents the worst case approximation of the maximum likelihood code and it is allowed to be almost any distribution within a model class of feasible distributions. Hence, the assumption on the true distribution generating the data may be considered to be relaxed. The so-called normalized maximum likelihood (NML) distribution satisfies the above optimization problem. It may be written as [31]

$$p^*(\mathbf{y}) = \frac{f(\mathbf{y}|\hat{\theta})}{\int f(\mathbf{Y}|\hat{\theta})d\mathbf{Y}},$$

where  $\hat{\theta}$  denotes the maximum likelihood estimate of  $\theta$  over the observation set. The denominator is a normalizing factor that is the sum of the maximum likelihoods of all potential observation sets  $\mathbf{Y}$  acquired in experiments, and the numerator represents the maximized likelihood value. With this solution, the worst case regret is minimized. It has been further shown that by solving a related minimax problem

$$p^* = \arg \min_p \max_q \mathbb{E}_q \left[ \ln \frac{f(\mathbf{y}|\hat{\theta})}{p(\mathbf{y})} \right],$$

the worst case of the expected regret is considered instead of the regret for the observed data set, see [32].

The code length associated with normalized maximum likelihood is called the *stochastic complexity* of the data for a distribution family. It minimizes the worst case expected regret

$$\mathbb{E}_q \left[ \ln \frac{f(\mathbf{y}|\hat{\theta})}{p^*(\mathbf{y})} \right].$$

The MDL method employs first the normalized maximum likelihood distributions to order the candidate models and chooses the model that gives minimal stochastic complexity. An early result by Rissanen [11] and a frequently used approximation for the stochastic complexity is given as follows:

$$\text{MDL}(K) = -2 \log f(\mathbf{y}|\hat{\theta}) + K \log N,$$

which is the same result as the approximation of BIC by Schwarz given earlier in this section. Other approximations of the stochastic complexity that are more accurate avoid some pitfalls of the above approximation, for example:

$$\text{MDL}(K) = -2 \log f(\mathbf{y}|\hat{\theta}) + K \log N + \log \int_{\theta} \sqrt{|I(\theta)|} d\theta + o(1),$$

where  $I(\theta)$  denotes the Fischer information matrix of sample size 1. The term  $o(1)$  contains the higher order terms and vanishes as  $N \rightarrow \infty$ . The integral term with Fischer information needs to be finite. See [33] for more detail on other approximations and their computation. The MDL principle and the normalized maximum likelihood (NML) lead to the same expression as the Bayesian Information Criterion (BIC) in some special cases, for example, when Jeffrey's noninformative prior is used in a Bayesian model and  $N \rightarrow \infty$ . In general, however, the NML and BIC have different formulations. The MDL has been shown to provide consistent estimates of the model order, e.g., for estimating the dimensionality of the signal subspace in the presence of white noise [34]. This problem is reviewed in detail in the next section.

### 3.02.5 Example: estimating number of signals in subspace methods

We consider the problem of estimating the dimensionality of the signal subspace. Such a problem commonly arises in array processing, time-delay estimation, frequency estimation, channel order estimation, for example. In these applications, if the signals are not coherent and signals are narrow-band, the dimension of the signal subspace defines the number of signals.

We consider the model, where the received data  $\mathbf{y} \in \mathbb{C}^M$  consists of (unobservable) random zero mean signal vector  $\mathbf{v} \in \mathbb{C}^M$  plus additive zero mean white noise  $\mathbf{n} \in \mathbb{C}^M$  ( $\mathbb{E}[\mathbf{n}\mathbf{n}^H] = \sigma^2 I$  with  $\sigma^2 > 0$  unknown) uncorrelated with the signal  $\mathbf{v}$  which is supposed to lie in an  $m$ -dimensional subspace of  $\mathbb{C}^M$  ( $m \leq M$ ), called the *signal subspace*. The covariance of the data  $\mathbf{y} = \mathbf{v} + \mathbf{n}$  is then  $\Sigma = \mathbb{E}[\mathbf{y}\mathbf{y}^H] = \Omega + \sigma^2 I$ , where the signal covariance matrix  $\Omega = \mathbb{E}[\mathbf{v}\mathbf{v}^H]$  is of rank( $\Omega$ ) =  $m \leq M$ . Let  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M \geq 0$  denote the ordered eigenvalues of  $\Sigma$ . If rank( $\Omega$ ) =  $m$ , then the smallest eigenvalue  $\lambda_M$  equals  $\sigma^2$  and has multiplicity  $M - m$ , whereas the  $m$  largest eigenvalues are equal to the nonzero eigenvalues of  $\Omega$  plus  $\sigma^2$ . Hence, testing that the signal subspace has dimensionality  $m$  ( $m \in \{0, 1, \dots, M - 1\}$ ) can be formulated as a hypothesis of the equality of the eigenvalues

$$\mathcal{H}_m : \lambda_{m+1} = \lambda_{m+2} = \dots = \lambda_M \quad (2.14)$$

and a GLRT can be formed against a general alternative  $\mathcal{H}_M : \Sigma$  arbitrary. Thus, we have a set of hypotheses  $\{\mathcal{H}_m, m = 0, 1, \dots, M\}$  and a model selection method can be used to determine the best model given the data. To this end, note that  $\mathcal{H}_0$  corresponds to the noise-only hypothesis. Let us assume that  $\mathbf{v}$  and  $\mathbf{n}$  possess  $M$ -variate circular complex normal (CN) distribution in which case the received data  $\mathbf{y}$  has a zero mean CN distribution with covariance matrix  $\Sigma$ . Further suppose that an i.i.d. random sample  $\mathbf{y}_1, \dots, \mathbf{y}_N$  is observed from  $\mathbf{y} \sim \mathcal{CN}_M(0, \Sigma)$ .

For the above problem, GLRT-based sequential test or the MDL and AIC criteria have been widely in sensor array signal processing and subspace estimation methods; see, e.g., [15–17], where the task at hand is to determine the number of signals  $m$  impinging on the sensor array. In this case, the signal  $\mathbf{v}$  has a representation  $\mathbf{v} = \mathbf{a}(\vartheta_1)s_1 + \dots + \mathbf{a}(\vartheta_m)s_m$ , where  $\mathbf{a}(\vartheta_i)$  denote the array response (steering vector) for a random source signal  $s_i$  impinging on the array from the direction-of-arrival (DOA)  $\vartheta_i$  ( $i = 1, \dots, m$ ). In this application, the sample  $\mathbf{y}_1, \dots, \mathbf{y}_N$  represents the array snapshots at time instants  $t_1, \dots, t_N$  and the SCM  $\widehat{\Sigma}$  is the conventional estimate of the array covariance matrix  $\Sigma$ , where the signal covariance matrix  $\Omega$  has a representation  $\Omega = A\mathbb{E}[\mathbf{s}\mathbf{s}^H]A^H$ , where  $A = (\mathbf{a}(\vartheta_1) \cdots \mathbf{a}(\vartheta_m))$  denotes the array

steering matrix and  $\mathbf{s} = (s_1, \dots, s_m)^T$  the vector of source signals. Another type of the problem such as determining the degree of non-circularity of complex random signals is illustrated in detail in [35].

Let us proceed by first constructing the sequential GLRT-based model order selection procedure for the problem at hand. The p.d.f. of  $\mathbf{y} \sim \mathbb{C}N_M(0, \Sigma)$  is  $f(\mathbf{y}|\Sigma) = \pi^{-M} |\Sigma|^{-1} \exp(-\mathbf{y}^H \Sigma^{-1} \mathbf{y})$ , and thus the log-likelihood function of the sample is

$$\ell(\theta) = \sum_{i=1}^N \ln f(y_i|\Sigma) \propto -N \ln |\Sigma| - N \text{Tr}(\Sigma^{-1} \widehat{\Sigma}),$$

where  $\widehat{\Sigma} = \frac{1}{N} \sum_{i=1}^N \mathbf{y}_i \mathbf{y}_i^H$  denotes the *sample covariance matrix* (SCM),  $\text{Tr}(\cdot)$  denotes matrix trace and  $\theta$  denotes the vector consisting of functionally independent real-valued parameters in  $\Sigma$ . Naturally,  $\theta$  has different parameter space  $\Theta_m$  under each hypothesis  $\mathcal{H}_m$  dimension and hence also different order  $\dim(\theta|\mathcal{H}_m)$ . Under no-structure hypothesis  $\mathcal{H}_M$ ,  $\dim(\theta|\mathcal{H}_M) = M^2$ . This follows as  $\Sigma$  is Hermitian and hence there are  $M(M-1)$  estimable parameters due to  $\text{Re}(\Sigma_{ij})$  and  $\text{Im}(\Sigma_{ij})$  for  $i < j \in \{1, \dots, M\}$  and  $M$  estimable parameters  $\Sigma_{ii} \geq 0, i = 1, \dots, M$ . For  $\mathcal{H}_m$ , where  $m < M$ ,  $K$  is the number of functionally independent real parameters in  $\Omega$  with rank  $m$  plus 1 due to  $\sigma^2 > 0$ . The eigenvalue decomposition of  $\Omega$  is  $\Omega = \sum_{i=1}^m \psi_i \mathbf{u}_i \mathbf{u}_i^H$  where  $\psi_i = \lambda_i - \sigma^2$  represent its nonzero eigenvalues and  $\mathbf{u}_i$ s the corresponding eigenvectors. There are  $2Mm$  real parameters in the eigenvectors but they are tied by  $2m$  constraints due to the scale and phase indeterminacy of each eigenvector<sup>1</sup> and an additional  $m(m-1)$  constraints due to orthogonality of the eigenvectors,  $\mathbf{u}_i^H \mathbf{u}_j = 0 + j0$  for  $i < j \in \{1, \dots, m\}$ . Since there are  $m$  eigenvalues  $\psi_i$  and  $\sigma^2$ , the total number of estimable parameters in  $\theta$  is

$$\dim(\theta|\mathcal{H}_m) = 2Mm + m + 1 - 2m - m(m-1) = m(2M-m) + 1 \quad (2.15)$$

when  $m < M$ .

Let us then derive the GLRT statistic  $\Lambda_m$  as defined in (2.10). Under  $\mathcal{H}_m$  for  $m \leq M$ , the maximum log-likelihood is

$$\ell(\hat{\theta}_m) = \max_{\theta \in \Theta_m} \ell(\theta) = -N \ln \prod_{i=1}^m \hat{\lambda}_i - N(M-m) \ln \hat{\sigma}^2 - NM, \quad (2.16)$$

where  $\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_M$  denote the eigenvalues of the SCM  $\widehat{\Sigma}$  and  $\hat{\sigma}^2 = \frac{1}{M-m} \sum_{i=m+1}^M \hat{\lambda}_i$ . The proof of (2.16) follows similarly as in Anderson [36, Theorem 2] in the real case. Then given the expression (2.16), it is easy to verify the GLRT statistic  $\Lambda_m = -2\{\ell(\hat{\theta}_m) - \ell(\hat{\theta}_M)\}$ , simplifies into the form

$$\Lambda_m = -2N(M-m) \ln \varrho(m), \quad (2.17)$$

where

$$\varrho(m) = \frac{1}{\hat{\sigma}^2} (\hat{\lambda}_{m+1} \hat{\lambda}_{m+2} \cdots \hat{\lambda}_M)^{\frac{1}{M-m}} \quad (2.18)$$

is the ratio of the geometric to the arithmetic mean of the noise subspace eigenvalues of the SCM. If  $\mathcal{H}_m$  holds, this ratio tends to unity as  $N \rightarrow \infty$ . Now recall from Section 3.02.3.2 that the GLRT statistic

<sup>1</sup>Recall that scale indeterminacy is fixed by assuming  $u_i^H u_i = 1$  and the phase of  $u_i$  can be fixed similarly by any suitable convention.

$\Lambda_m$  in (2.17) has a limiting  $\chi^2_{\text{DOF}(m)}$  distribution, where

$$\text{DOF}(m) = \dim(\theta|\mathcal{H}_M) - \dim(\theta|\mathcal{H}_m) = M^2 - \{m(2M-m)+1\} = (M-m)^2 - 1.$$

The (forward) GLRT-based model selection procedure then chooses the first  $\mathcal{H}_m$  not rejected by the GLRT, i.e., the first  $m$  which verifies  $\Lambda_m \leq \gamma_m$ , where (as discussed in Section 3.02.3.2) the detection threshold  $\gamma_m$  can be set as the  $(1-\alpha)$ th quantile of the  $\chi^2_{\text{DOF}(m)}$ . This choice yields the asymptotic PFA  $\alpha$  for testing hypothesis (2.14).

Let us now derive the AIC and MDL criteria. Let us first point out that the  $-NM$  term in (2.16) can be dropped as it does not depend on  $m$  whereas any constant term (namely,  $c = \ln \prod_{i=1}^M \hat{\lambda}_i$ ) not dependent on  $m$  can be added, thus showing that  $\ell(\hat{\theta}_m)$  in (2.16) can be written compactly as  $n(M-m) \ln \varrho(m)$  up to irrelevant additive constant not dependent on  $m$ . Further noting that the +1 can be dropped from the penalty term  $\dim(\theta|\mathcal{H}_m)$  in (2.15), we have the result that the AIC and the MDL criteria can be written in the following forms:

$$\begin{aligned} \text{AIC}(m) &= -2n(M-m) \ln \varrho(m) + 2m(2M-m), \\ \text{MDL}(m) &= -2n(M-m) \ln \varrho(m) + 2m(2M-m) \cdot \frac{\ln N}{2}. \end{aligned}$$

An estimate of the AIC/MDL model order, i.e., the dimensionality of the signal subspace  $m$ , is found by choosing  $m \in \{0, 1, \dots, M-1\}$  that minimizes the AIC/MDL criterion above. As already pointed out, the penalty term of the MDL is  $(1/2) \ln N$  times the penalty term of AIC. Since  $(1/2) \ln N > 1$  already at small sample lengths ( $N \geq 8$ ), the MDL method is less prone to overfitting the data compared to the AIC. Note that first term in the MDL and AIC criterions above is simply the GLRT statistic  $\Lambda_m$ . Wax and Kailath [15] showed the MDL rule consistent in choosing the true dimensionality whereas the AIC tends to overestimate the dimensionality (in the large sample limit). Later [34] proved the consistency of the MDL rule when the CN assumption does not hold.

### 3.02.6 Conclusions

In this section, we provided an overview of model order estimation methods. The majority of well-defined methods stem from statistical inference or coding and information theory. The likelihood function plays a crucial role in most methods based on solid theoretical foundation. Statistical and information theoretical methods stem from fundamentally different theoretical background but have been shown to lead to identical results under certain restrictive assumptions and model classes. A practical example of model order estimation in case of low-rank model that is commonly used in sensor array signal processing and many other tasks that require high-resolution capability was provided.

---

## References

- [1] P.D. Grünwald, The Minimum Description Length Principle, MIT Press, 2007.
- [2] P. Stoica, Y. Selen, Model-order selection: a review of information criterion rules, IEEE Signal Process. Mag. 21 (4) (2004) 36–47.

- [3] A.D. Lanterman, Schwarz, Wallace and Rissanen: intertwining themes in theories of model selection, *Int. Stat. Rev.* 69 (2) (2001) 185–212.
- [4] P.D. Grünwald, I. Myung, M. Pitt (Eds.), *Advances in Minimum Description Length: Theory and Applications*, MIT Press, 2005.
- [5] H. Linhart, W. Zucchini, *Model Selection*, Wiley, New York.
- [6] K. Burnham, D. Anderson, *Model Selection and Inference: A Practical Information-Theoretic Approach*, Springer, New York, 1998.
- [7] G. Schwarz, Estimating the dimension of a model, *Ann. Stat.* 6 (2) (1978) 461–464.
- [8] H. Akaike, A new look at the statistical model identification, *IEEE Trans. Automat. Control* 19 (1974) 716–723.
- [9] H. Akaike, On the likelihood of a time series model, *The Statistician* 27 (3) (1978) 217–235.
- [10] J. Rissanen, Stochastic complexity and modeling, *Ann. Stat.* 14 (3) (1986) 1080–1100.
- [11] J. Rissanen, Modeling by the shortest data description, *Automatica* 14 (1978) 465–471.
- [12] S.M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*, Prentice Hall, 1998.
- [13] S. Kay, Exponentially embedded families—new approaches to model order estimation, *IEEE Trans. Aerosp. Electron. Syst.* 41 (1) (2005) 333–344.
- [14] H. Bozdogan, Model selection and Akaike’s information criterion (AIC): the general theory and its analytical extensions, *Psychometrika* 52 (3) (1987) 345–370.
- [15] T. Wax, T. Kailath, Detection of signals by information theoretic criteria, *IEEE Trans. Acoust. Speech Signal Process.* 33 (2) (1985) 387–392.
- [16] Q.T. Zhang, K.M. Wong, Information theoretic criteria for the determination of the number of signals in spatially correlated noise, *IEEE Trans. Signal Process.* 41 (4) (1993) 1652–1663.
- [17] G. Xu, R.H.I. Roy, T. Kailath, Detection of number of sources via exploitation of centro-symmetry property, *IEEE Trans. Signal Process.* 42 (1) (1994) 102–112.
- [18] A.A. Shah, D.W. Tufts, Determination of the dimension of a signal subspace from short data records, *IEEE Trans. Signal Process.* 42 (9) (1994) 2531–2535.
- [19] S. Weisberg, *Applied Linear Regression*, Wiley, New York, 1980.
- [20] N.R. Draper, H. Smith, *Applied Regression Analysis*, third ed., Wiley, New York, 1998.
- [21] E. Ronchetti, Robust model selection in regression, *Stat. Prob. Lett.* 3 (1985) 21–23.
- [22] J.A. Tropp, A.C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Trans. Inform. Theory* 53 (12) (2007) 4655–4666.
- [23] R. Tibshirani, Regression shrinkage and selection via the lasso, *J. Roy. Stat. Soc. Ser. B* 58 (1996) 267–288.
- [24] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, *Ann. Stat.* 32 (2) (2004) 407–451.
- [25] S. Arlot, A. Celisse, A survey of cross-validation procedures for model selection, *Stat. Surv.* 4 (2010) 40–79.
- [26] B. Efron, Bootstrap methods: another look at the jackknife, *Ann. Stat.* 7 (1) (1979) 1–26.
- [27] R. Brich, A.M. Zoubir, P. Pelin, Detection of sources using bootstrap techniques, *IEEE Trans. Signal Process.* 50 (2) 206–215.
- [28] R. Nishii, Asymptotic properties of criteria for selection of variables in multiple regression, *Ann. Stat.* 2 (1984) 758–765.
- [29] J.E. Cavanaugh, Unifying the derivations for the Akaike and corrected Akaike information criteria, *Stat. Prob. Lett.* 33 (1977) 201–208.
- [30] N. Sugiura, Further analysis of the data by Akaike information criterion and the finite corrections, *Commun. Stat. Theory Methods* 7 (1978) 13–26.
- [31] Y. Shtarkov, Block universal sequential coding of single messages, *Prob. Inform. Transmission* 23 (3) (1987) 175–186.
- [32] J. Myung, D. Navarro, M. Pitt, Model selection by normalized maximum likelihood, *J. Math. Psychol.* 50 (2006) 167–179.

- [33] P. Kontkanen, W. Buntine, P. Myllymaki, J. Rissanen, H. Tirri, Efficient computation of stochastic complexity, in: Proceedings of International Conference on Artificial Intelligence and, Statistics, 2003, pp. 233–238.
- [34] L.C. Zhao, P.R. Krishnaiah, Z.D. Bai, On detection of the number of signals in presence of white noise, *J. Multivar. Anal.* 20 (1) (1986) 1–25.
- [35] M. Novey, E. Ollila, T. Adali, On testing the extent of noncircularity, *IEEE Trans. Signal Process.* 59 (11) (2011) 5632–5637.
- [36] T.W. Anderson, Asymptotic theory for principal component analysis, *Ann. Math. Stat.* 1 (1963) 122–148.

# Non-Stationary Signal Analysis Time-Frequency Approach

# 3

Ljubiša Stanković\*, Miloš Daković\*, and Thayananthan Thayaparan†

\*Electrical Engineering Department, University of Montenegro, Montenegro  
†Defense Scientist, Defence R&D, Ottawa, Canada

## 3.03.1 Introduction

The Fourier transform (FT) provides a unique mapping of a signal from the time domain to the frequency domain. The frequency domain representation provides the signal's spectral content. Although the phase characteristic of the FT contains information about the time distribution of the spectral content, it is very difficult to use this information. Therefore, one may say that the FT is practically useless for this purpose, i.e., that the FT does not provide a time distribution of the spectral components.

Depending on problems encountered in practice, various representations have been proposed to analyze non-stationary signals in order to provide time-varying spectral description. The field of the time-frequency signal analysis deals with these representations of non-stationary signals and their properties. Time-frequency representations may roughly be classified as linear, quadratic or higher order representations.

Linear time-frequency representations exhibit linearity, i.e., the representation of a linear combination of signals equals the linear combination of the individual representations. From this class, the most important one is the short-time Fourier transform (STFT) and its variations. A specific form of the STFT was originally introduced by Gabor in mid 1940s. The energetic version of the STFT is called spectrogram. It is the most frequently used tool in time-frequency signal analysis [1–6].

The second class of time-frequency representations are the quadratic ones. The most interesting representations of this class are those which provide a distribution of signal energy in the time-frequency plane. They will be referred to as distributions. The concept of a distribution is borrowed from the probability theory, although there is a fundamental difference. For example, in time-frequency analysis, distributions may take negative values. Other possible domains for quadratic signal representations are the ambiguity domain, the time-lag domain and the frequency-Doppler frequency domain.

Despite the loss of the linearity, the quadratic representations are commonly used due to higher time-frequency concentration compared to linear transforms. A quadratic time-frequency representation known as the Wigner distribution was the first representation introduced in 1932. It is interesting to note that the motivation for definition of this distribution, as well as for some others, was found in quantum mechanics. The Wigner distribution was introduced into the signal theory by Ville in 1948. Therefore, it is often called the Wigner-Ville distribution. In order to reduce undesirable effects, other quadratic time-frequency distributions have been introduced. A general form of all quadratic time-frequency

distributions has been defined by Cohen (1966) and introduced in time-frequency analysis by Claassen and Mecklenbräuker (1981). This generalization prompted the introduction of new time-frequency distributions, including the Choi-Williams distribution, Zhao-Atlas Marks distribution and many other distributions referred to as the reduced interference distributions [1,2,6–18].

Higher order representations have been introduced in order to further improve the concentration of time-frequency representations [6,19–22].

### 3.03.2 Linear signal transforms

A transform is linear if a linear combination of signals is equal to the linear combination of the transforms. Various complex forms of signal representations satisfy this property, starting from the short-time Fourier transform, via local polynomial Fourier transform and wavelet transform, up to general signal decomposition forms, including chirplet transform. Although energetic versions of the linear transforms, calculated as their squared moduli, do not preserve the linearity, they will be considered within this section as well.

#### 3.03.2.1 Short-time fourier transform

The Fourier transform (FT) of a signal  $x(t)$  and its inverse are defined by

$$X(\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt, \quad (3.1)$$

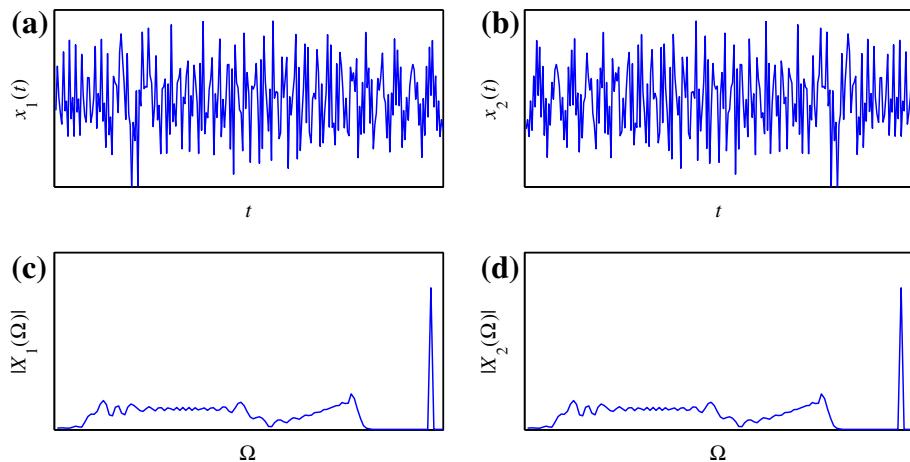
$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) e^{j\Omega t} d\Omega. \quad (3.2)$$

The FT of a signal  $x(t)$  shifted in time for  $t_0$ , i.e.,  $x(t - t_0)$ , is equal to  $X(\Omega) \exp(-j\Omega t_0)$ . The amplitude characteristics of  $x(t)$  and  $x(t - t_0)$  are the same and equal to  $|X(\Omega)|$ . The same holds for a real valued signal  $x(t)$  and its shifted and reversed version  $x(t_0 - t)$ . We will illustrate this with two different signals  $x_1(t)$  and  $x_2(t)$  (distributed over time in a different manner) producing the same amplitude of the FT (see Figure 3.1)

$$\begin{aligned} x_1(t) = & \sin\left(122\pi \frac{t}{128}\right) - \cos\left(42\pi \frac{t}{128} - \frac{16}{11}\pi \left(\frac{t-128}{64}\right)^2\right) \\ & - 1.2 \cos\left(94\pi \frac{t}{128} - 2\pi \left(\frac{t-128}{64}\right)^2 - \pi \left(\frac{t-120}{64}\right)^3\right) e^{-\left(\frac{t-140}{75}\right)^2} \\ & - 1.6 \cos\left(15\pi \frac{t}{128} - 2\pi \left(\frac{t-50}{64}\right)^2\right) e^{-\left(\frac{t-50}{16}\right)^2}, \end{aligned} \quad (3.3)$$

$$x_2(t) = x_1(255 - t).$$

The idea behind the short-time Fourier transform (STFT) is to apply the FT to a portion of the original signal, obtained by introducing a sliding window function  $w(t)$  which will localize, truncate (and weight), the analyzed signal  $x(t)$ . The FT is calculated for the localized part of the signal. It produces

**FIGURE 3.1**

Two different signals  $x_1(t) \neq x_2(t)$  with the same amplitudes of their Fourier transforms,  $|X_1(\Omega)| = |X_2(\Omega)|$ .

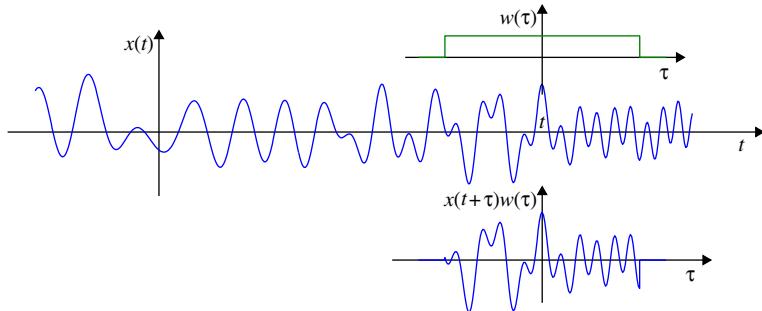
**FIGURE 3.2**

Illustration of the signal localization in the STFT calculation.

the spectral content of the portion of the analyzed signal within the time interval defined by the width of the window function. The STFT (a time-frequency representation of the signal) is then obtained by sliding the window along the signal. Illustration of the STFT calculation is presented in Figure 3.2.

Analytic formulation of the STFT is

$$\text{STFT}(t, \Omega) = \int_{-\infty}^{\infty} x(t + \tau)w(\tau)e^{-j\Omega\tau} d\tau. \quad (3.4)$$

From (3.4) it is apparent that the STFT actually represents the FT of a signal  $x(t)$ , truncated by the window  $w(\tau)$  centered at instant  $t$  (see Figure 3.2). From the definition, it is clear that the STFT satisfies properties inherited from the FT (e.g., linearity).

By denoting  $x_t(\tau) = x(t + \tau)$  we can conclude that the STFT is the FT of the signal  $x_t(\tau)w(\tau)$ ,  $\text{STFT}(t, \Omega) = \text{FT}_\tau\{x_t(\tau)w(\tau)\}$ .

Another form of the STFT, with the same time-frequency performance, is

$$\text{STFT}_{II}(t, \Omega) = \int_{-\infty}^{\infty} x(\tau)w^*(\tau - t)e^{-j\Omega\tau} d\tau, \quad (3.5)$$

where  $w^*(t)$  denotes the conjugated window function.

It is obvious that definitions (3.4) and (3.5) differ only in phase, i.e.,  $\text{STFT}_{II}(t, \Omega) = e^{-j\Omega t} \text{STFT}(t, \Omega)$  for real valued windows  $w(\tau)$ . In the sequel we will mainly use the first definition of the STFT [23].

**Example 1.** To illustrate the STFT application, let us perform the time-frequency analysis of the following signal:

$$x(t) = \delta(t - t_1) + \delta(t - t_2) + e^{j\Omega_1 t} + e^{j\Omega_2 t}. \quad (3.6)$$

The STFT of this signal equals

$$\begin{aligned} \text{STFT}(t, \Omega) &= w(t_1 - t)e^{-j\Omega(t_1-t)} + w(t_2 - t)e^{-j\Omega(t_2-t)} \\ &\quad + W(\Omega - \Omega_1)e^{j\Omega_1 t} + W(\Omega - \Omega_2)e^{j\Omega_2 t}, \end{aligned} \quad (3.7)$$

where  $W(\Omega)$  is the FT of the used window. The STFT is depicted in Figure 3.3 for various window lengths, along with the ideal representation.

The STFT can be expressed in terms of the signal's FT

$$\begin{aligned} \text{STFT}(t, \Omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(\theta)e^{j(t+\tau)\theta} w(\tau)e^{-j\Omega\tau} d\theta d\tau \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\theta)W(\Omega - \theta)e^{jt\theta} d\theta = [X(\Omega)e^{jt\Omega}] *_{\Omega} W(\Omega), \end{aligned} \quad (3.8)$$

where  $*_{\Omega}$  denotes convolution in  $\Omega$ . It may be interpreted as an inverse FT of the frequency localized version of  $X(\Omega)$ , with localization window  $W(\Omega) = \text{FT}\{w(\tau)\}$ .

### 3.03.2.1.1 Windows

It is obvious that the window function plays a critical role in the localization of the signal in the time-frequency plane. Thus, we will briefly review windows commonly used for localization of non-stationary signals.

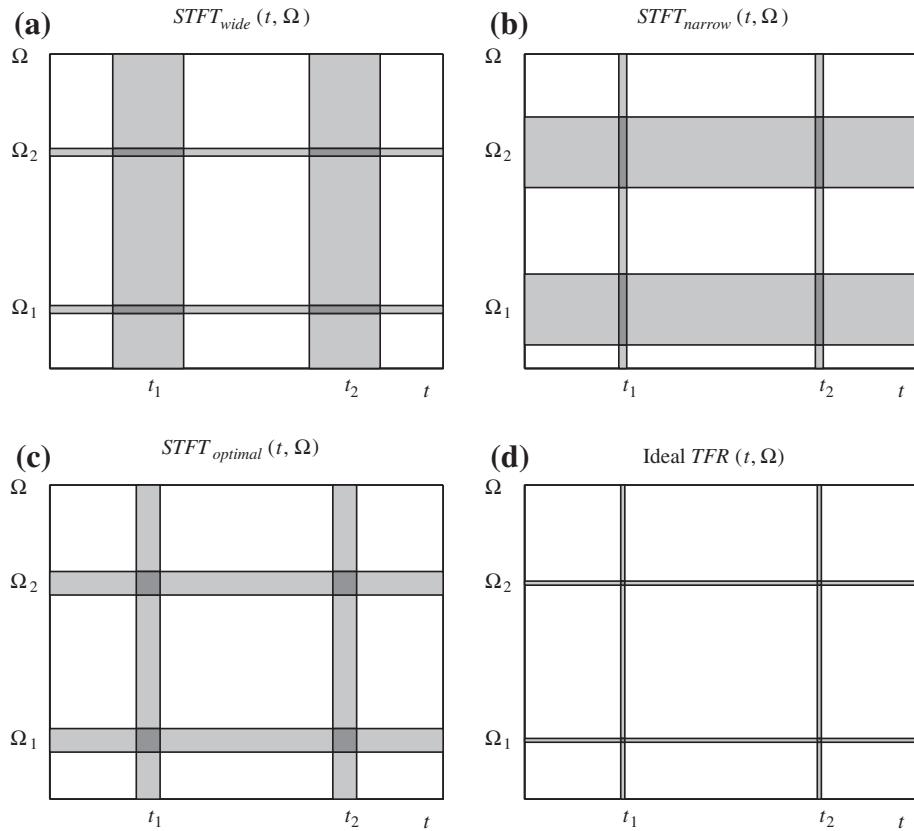
*Rectangular window:* The simplest window is the rectangular one, defined by

$$w(\tau) = \begin{cases} 1 & \text{for } |\tau| < T, \\ 0 & \text{elsewhere,} \end{cases}$$

whose FT is

$$W_R(\Omega) = \int_{-T}^T e^{-j\Omega\tau} d\tau = \frac{2 \sin(\Omega T)}{\Omega}.$$

The rectangular window function has very strong side lobes in the frequency domain, since the function  $\sin(\Omega T)/\Omega$  converges very slowly as  $\Omega \rightarrow \infty$ . Thus, in order to enhance signal localization in the frequency domain, other window functions have been introduced.

**FIGURE 3.3**

Time-frequency representation of a sum of two delta pulses and two sinusoids obtained by using: (a) a wide window, (b) a narrow window, (c) a medium width window, and (d) the ideal time-frequency representation.

*Hann(ing) window:* This window is of the form

$$w(\tau) = \begin{cases} \frac{1}{2}(1 + \cos(\tau\pi/T)) & \text{for } |\tau| < T, \\ 0 & \text{elsewhere.} \end{cases}$$

Since  $\cos(\tau\pi/T) = [\exp(j\pi\tau/T) + \exp(-j\pi\tau/T)]/2$ , the FT of this window is related to the FT of the rectangular window of the same width as

$$W_H(\Omega) = \frac{1}{2}W_R(\Omega) + \frac{1}{4}W_R(\Omega - \pi/T) + \frac{1}{4}W_R(\Omega + \pi/T) = \frac{\pi^2 \sin(\Omega T)}{\Omega(\pi^2 - \Omega^2 T^2)}.$$

Function  $W_H(\Omega)$  decays in frequency much faster than  $W_R(\Omega)$ . The previous relation also implies the relationship between the STFTs of the signal  $x(t)$  calculated using the rectangular and Hann(ing)

windows,  $\text{STFT}_R(t, \Omega)$  and  $\text{STFT}_H(t, \Omega)$ , given as

$$\text{STFT}_H(t, \Omega) = \frac{1}{2} \text{STFT}_R(t, \Omega) + \frac{1}{4} \text{STFT}_R(t, \Omega - \pi/T) + \frac{1}{4} \text{STFT}_R(t, \Omega + \pi/T). \quad (3.9)$$

For the Hann(ing) window  $w(\tau)$  of the width  $2T$ , in the analysis that follows, we may roughly assume that its FT  $W_H(\Omega)$  is non-zero only within the main lattice  $|\Omega| < 2\pi/T$ , since the side lobes decay very fast. It means that the STFT is non-zero-valued in the shaded regions in Figure 3.3a–c. We see that the duration in time of the STFT of a delta pulse is equal to the widow width  $d_t = 2T$ . The STFTs of two delta pulses  $\delta(t - t_1)$  and  $\delta(t - t_2)$  (very short duration signals) do not overlap in time-frequency domain if their distance is greater than the window duration  $|t_1 - t_2| > d_t$ . Since the FT of the Hann(ing) window converges fast, we can intuitively assume that a measure of duration in frequency is the width of its main lobe,  $d_\Omega = 4\pi/T$ . Then, we may say that two (complex) sine waves  $e^{j\Omega_1 t}$  and  $e^{j\Omega_2 t}$  do not overlap in frequency if the condition  $|\Omega_1 - \Omega_2| > d_\Omega$  holds. It is important to observe that the product of the window durations in time and frequency is a constant. In this example, considering time domain duration of the Hann(ing) window and the width of its main lobe in the frequency domain, this product is  $d_t d_\Omega = 8\pi$ . Therefore, if we improve the resolution in the time domain  $d_t$ , by decreasing  $T$ , we inherently increase value of  $d_\Omega$  in the frequency domain. This essentially prevents us from achieving the ideal resolution in both domains. A general formulation of this principle, stating that product of window durations in time and in frequency cannot be arbitrary small, will be presented later.

*Hamming window:* This window has the form

$$w(\tau) = \begin{cases} 0.54 + 0.46 \cos(\tau\pi/T) & \text{for } |\tau| < T, \\ 0 & \text{elsewhere.} \end{cases}$$

Similar relations between the Hamming and the rectangular window transforms hold as in the case of Hann(ing) window. This window has lower first side lobe than the Hann(ing) window. However, since it has a discontinuity at  $t = \pm T$ , its convergence as  $\Omega \rightarrow \infty$  is not faster in frequency than in the case of a Hann(ing) window.

*Gaussian window:* This window localizes signal in time, although it is not time-limited. Its form is

$$w(\tau) = e^{-\tau^2/\alpha^2}.$$

In some applications it is crucial that the nearest side lobes are suppressed as much as possible. This is achieved by using windows of more complicated forms, like the **Blackman window** and the **Kaiser window** [6].

### 3.03.2.1.2 Duration measures and uncertainty principle

We started discussion about the signal concentration (window duration) and resolution in the Hann(ing) window case, with illustration in Figure 3.3. In general, window (any signal) duration in time or/and in frequency is not obvious from its definition or form. Then, the effective duration is used as a measure of window (signal) duration. In time domain the effective duration measure is defined by

$$\sigma_t^2 = \frac{\int_{-\infty}^{\infty} \tau^2 |w(\tau)|^2 d\tau}{\int_{-\infty}^{\infty} |w(\tau)|^2 d\tau}.$$

Similarly, the measure of duration in frequency is

$$\sigma_{\Omega}^2 = \frac{\int_{-\infty}^{\infty} \Omega^2 |W(\Omega)|^2 d\Omega}{\int_{-\infty}^{\infty} |W(\Omega)|^2 d\Omega}.$$

Here, it has been assumed that the time and frequency domain forms of the window (signal) are centered, i.e.,  $\int_{-\infty}^{\infty} \tau |w(\tau)|^2 d\tau = 0$  and  $\int_{-\infty}^{\infty} \Omega |W(\Omega)|^2 d\Omega = 0$ . If this were not the case, then the widths of centered forms in time and frequency would be calculated.

The uncertainty principle in signal processing states that the product of effective measures of duration in time and frequency, for any function satisfying  $w(\tau)\sqrt{\tau} \rightarrow 0$  as  $\tau \rightarrow \pm\infty$ , is

$$\sigma_t^2 \sigma_{\Omega}^2 \geq 1/4. \quad (3.10)$$

Since this principle will be used in the further analysis, we will present its short proof. Since  $w'(\tau) = \frac{d}{d\tau} w(\tau)$  is the inverse FT of  $j\Omega W(\Omega)$ , according to the Parseval's theorem we have

$$\int_{-\infty}^{\infty} |j\Omega W(\Omega)|^2 d\Omega / 2\pi = \int_{-\infty}^{\infty} |w'(\tau)|^2 d\tau$$

and the product  $\sigma_t^2 \sigma_{\Omega}^2$  may be written as

$$\sigma_t^2 \sigma_{\Omega}^2 = \frac{1}{E_w^2} \int_{-\infty}^{\infty} \tau^2 |w(\tau)|^2 d\tau \int_{-\infty}^{\infty} |w'(\tau)|^2 d\tau,$$

where  $E_w$  is the energy of the window (signal),

$$E_w = \int_{-\infty}^{\infty} |w(\tau)|^2 d\tau = \frac{1}{2\pi} \int_{-\infty}^{\infty} |W(\Omega)|^2 d\Omega.$$

For any two integrable functions  $x_1(\tau)$  and  $x_2(\tau)$ , the Cauchy-Schwartz inequality

$$\left| \int_{-\infty}^{\infty} x_1(\tau) x_2^*(\tau) d\tau \right|^2 \leq \int_{-\infty}^{\infty} |x_1(\tau)|^2 d\tau \int_{-\infty}^{\infty} |x_2(\tau)|^2 d\tau$$

holds. The equality holds for

$$x_1(\tau) = \pm \gamma x_2^*(\tau),$$

where  $\gamma$  is a positive constant.

In our case, the equality holds for

$$\tau w(\tau) = \pm \gamma w'(\tau).$$

The finite energy solution of this differential equation, with  $w(0) = 1$ , is the Gaussian function

$$w(\tau) = \exp(-\tau^2/2\gamma).$$

For the Gaussian window (signal) it may be shown that this product is equal to  $\sigma_t^2 \sigma_{\Omega}^2 = 1/4$ , meaning that the Gaussian window (signal) is the best localized window (signal) in the sense of effective durations. In the sense of illustration in Figure 3.3, this fact also means that, for a given width of the STFT of a pulse  $\delta(t)$  in time direction, the narrowest presentation of a sinusoid in frequency direction is achieved by using the Gaussian window.

### 3.03.2.1.3 Continuous STFT inversion

The original signal  $x(t)$  may be easily reconstructed from its STFT (3.4) by applying the inverse FT, i.e.,

$$x(t + \tau) = \frac{1}{2\pi w(\tau)} \int_{-\infty}^{\infty} \text{STFT}(t, \Omega) e^{j\Omega\tau} d\Omega. \quad (3.11)$$

In this way, we can calculate the values of  $x(t + \tau)$  for a given instant  $t$  ( $\tau = 0$ ) and for the values of  $\tau$  where  $w(\tau)$  is non-zero. Then, we may skip the window width, take the time instant  $t + 2T$ , and calculate the inverse of  $\text{STFT}(t + 2T, \Omega)$ , and so on.

Theoretically, for a window of the width  $2T$ , it is sufficient to know the STFT calculated at  $t = 2kR$ ,  $k = 0, \pm 1, \pm 2, \dots$ , with  $R \leq T$ , in order to reconstruct signal for any  $t$  (reconstruction conditions will be discussed in details later, within the discrete forms).

A special case for  $\tau = 0$  gives

$$x(t) = \frac{1}{2\pi w(0)} \int_{-\infty}^{\infty} \text{STFT}(t, \Omega) d\Omega. \quad (3.12)$$

For the STFT defined by (3.5) the signal can be obtained as

$$x(\tau) = \frac{1}{2\pi w^*(\tau - t)} \int_{-\infty}^{\infty} \text{STFT}_{II}(t, \Omega) e^{j\Omega\tau} d\Omega.$$

In order to reconstruct the signal from its STFT, we may skip the window width at  $t$  and take as the time instant  $t + 2T$ . If we calculate  $\text{STFT}(t, \Omega)$  for all values of  $t$  (which is a common case in the analysis of highly non-stationary signals), the inversion results in multiple values of signal for a given instant, which all can be used for better signal reconstruction as follows:

$$x(\tau) = \frac{1}{2\pi W^*(0)} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{STFT}_{II}(\Omega, t) e^{j\Omega\tau} d\Omega dt.$$

In the case that we are interested only in a part of the time-frequency plane, relation (3.12) can be used for the time-varying signal filtering. The STFT, for a given  $t$ , can be modified by  $B(t, \Omega)$  and the filtered signal obtained as

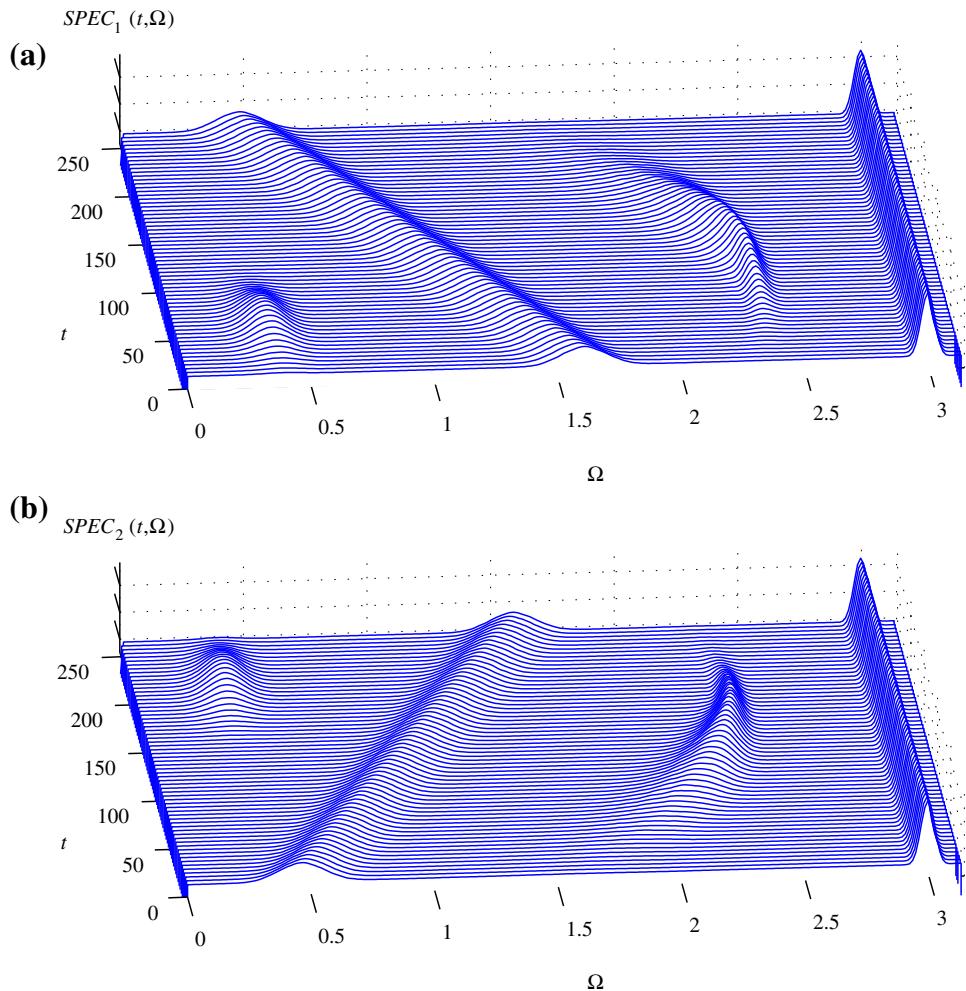
$$y(t) = \frac{1}{2\pi w(0)} \int_{-\infty}^{\infty} B(t, \Omega) \text{STFT}(t, \Omega) d\Omega.$$

For example we can use  $B(t, \Omega) = 1$  within the time-frequency region of interest and  $B(t, \Omega) = 0$  elsewhere.

The energetic version of the STFT, called the spectrogram, is defined by

$$\begin{aligned} \text{SPEC}(t, \Omega) &= |\text{STFT}(t, \Omega)|^2 \\ &= \left| \int_{-\infty}^{\infty} x(\tau) w^*(\tau - t) e^{-j\Omega\tau} d\tau \right|^2 = \left| \int_{-\infty}^{\infty} x(t + \tau) w^*(\tau) e^{-j\Omega\tau} d\tau \right|^2. \end{aligned}$$

Obviously, the linearity property is lost in the spectrogram. The spectrograms of the signals from Figure 3.1 are presented in Figure 3.4.

**FIGURE 3.4**

The spectrograms of the signals presented in Figure 3.1.

#### 3.03.2.1.4 STFT of multi-component signals

Let us introduce multi-component signal  $x(t)$  as the sum of  $M$  components  $x_m(t)$ ,

$$x(t) = \sum_{m=1}^M x_m(t) = \sum_{m=1}^M A_m(t) e^{j\phi_m(t)}. \quad (3.13)$$

The STFT of this signal is equal to the sum of the STFTs of individual components,

$$\text{STFT}(t, \Omega) = \sum_{m=1}^M \text{STFT}_m(t, \Omega) \quad (3.14)$$

that will be referred to as the auto-terms. This is one of very appealing properties of the STFT, which will be lost in the quadratic and higher order distributions.

The spectrogram of multi-component signal (3.13) is of the form:

$$\text{SPEC}(t, \Omega) = |\text{STFT}(t, \Omega)|^2 = \sum_{m=1}^M |\text{STFT}_m(t, \Omega)|^2$$

only if the STFTs of signal components,  $\text{STFT}_m(t, \Omega)$ ,  $m = 1, 2, \dots, M$ , do not overlap in the time-frequency plane, i.e., if

$$\text{STFT}_m(t, \Omega)\text{STFT}_n^*(t, \Omega) = 0 \quad \text{for all } (t, \Omega) \text{ if } m \neq n.$$

In general

$$\text{SPEC}(t, \Omega) = \sum_{m=1}^M |\text{STFT}_m(t, \Omega)|^2 + \sum_{m=1}^M \sum_{\substack{n=1 \\ n \neq m}}^M \text{STFT}_m(t, \Omega)\text{STFT}_n^*(t, \Omega),$$

where the second term on the right side represents the terms resulting from the interaction between two signal components. They are called cross-terms. The cross-terms are undesirable components, arising due to non-linear structure of the spectrogram. Here, they appear only at the time-frequency points where the auto-terms overlap. We will see that in other quadratic time-frequency representations they may appear even if the components do not overlap.

### 3.03.2.2 Discrete form and realizations of the STFT

In numerical calculations the integral form of the STFT should be discretized. By sampling the signal with sampling interval  $\Delta t$  we get

$$\begin{aligned} \text{STFT}(t, \Omega) &= \int_{-\infty}^{\infty} x(t + \tau)w(\tau)e^{-j\Omega\tau}d\tau \\ &\simeq \sum_{m=-\infty}^{\infty} x((n+m)\Delta t)w(m\Delta t)e^{-jm\Delta t\Omega}\Delta t. \end{aligned}$$

By denoting

$$x(n) = x(n\Delta t)\Delta t$$

and normalizing the frequency  $\Omega$  by  $\Delta t$ ,  $\omega = \Delta t\Omega$ , we get the time-discrete form of the STFT as

$$\text{STFT}(n, \omega) = \sum_{m=-\infty}^{\infty} w(m)x(n+m)e^{-jm\omega}. \quad (3.15)$$

We will use the same notation for continuous-time and discrete-time signals,  $x(t)$  and  $x(n)$ . However, we hope that this will not cause any confusion since we will use different sets of variables, for example  $t$  and  $\tau$  for continuous time and  $n$  and  $m$  for discrete time. Also, we hope that the context will be always clear, so that there is no doubt what kind of signal is considered.

It is important to note that  $\text{STFT}(n, \omega)$  is periodic in frequency with period  $2\pi$ . The relation between the analog and the discrete-time form is

$$\text{STFT}(n, \omega) = \sum_{k=-\infty}^{\infty} \text{STFT}(n\Delta t, \Omega + 2k\Omega_0) \quad \text{with } \omega = \Delta t \Omega.$$

The sampling interval  $\Delta t$  is related to the period in frequency as  $\Delta t = \pi / \Omega_0$ . According to the sampling theorem, in order to avoid the overlapping of the STFT periods (aliasing), we should take

$$\Delta t = \frac{\pi}{\Omega_0} \leq \frac{\pi}{\Omega_m},$$

where  $\Omega_m$  is the maximal frequency in the STFT. Strictly speaking, the windowed signal  $x(t + \tau)w(\tau)$  is time limited, thus it is not frequency limited. Theoretically, there is no maximal frequency since the width of the window's FT is infinite. However, in practice we can always assume that the value of spectral content of  $x(t + \tau)w(\tau)$  above frequency  $\Omega_m$ , i.e., for  $|\Omega| > \Omega_m$ , can be neglected, and that overlapping of the frequency content above  $\Omega_m$  does not degrade the basic frequency period.

The discretization in frequency should be done by a number of samples greater than or equal to the window length  $N$ . If we assume that the number of discrete frequency points is equal to the window length, then

$$\text{STFT}(n, k) = \text{STFT}(n, \omega)|_{\omega=\frac{2\pi}{N}k} = \sum_{m=-N/2}^{N/2-1} w(m)x(n+m)e^{-j2\pi mk/N} \quad (3.16)$$

and it can be efficiently calculated using the fast DFT routines

$$\text{STFT}(n, k) = \text{DFT}_m\{w(m)x(n+m)\}$$

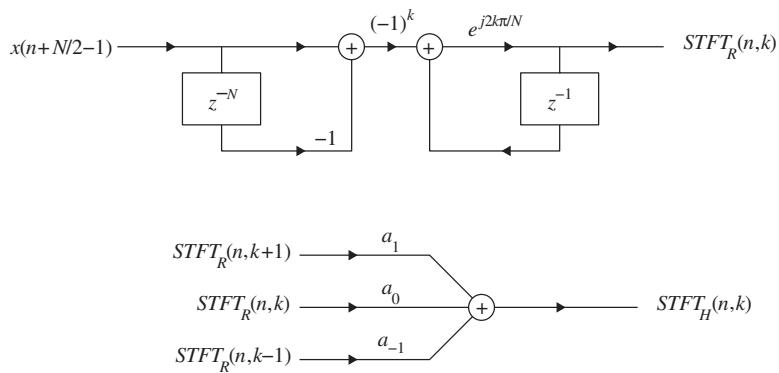
for a given instant  $n$ . When the DFT routines with indices from 0 to  $N - 1$  are used, then a shifted version of  $w(m)x(n+m)$  should be formed for the calculation for  $N/2 \leq m \leq N - 1$ . It is obtained as  $w(m-N)x(n+m-N)$ , since in the DFT calculation periodicity of the signal  $w(m)x(n+m)$ , with period  $N$ , is inherently assumed.

For the rectangular window, the STFT values at an instant  $n$  can be calculated recursively from the STFT values at  $n - 1$ , as

$$\begin{aligned} \text{STFT}_R(n, k) &= [x(n+N/2-1) - x(n-N/2-1)](-1)^k e^{j2\pi k/N} \\ &\quad + \text{STFT}_R(n-1, k)e^{j2\pi k/N}. \end{aligned}$$

This recursive formula follows easily from the STFT definition (3.16).

For other window forms, the STFT can be obtained from the STFT obtained by using the rectangular window. For example, according to (3.9) the STFT with Hann(ing) window  $\text{STFT}_H(n, k)$  is related to

**FIGURE 3.5**

A recursive implementation of the STFT for the rectangular and other windows.

the STFT with rectangular window  $\text{STFT}_R(n, k)$  as

$$\text{STFT}_H(n, k) = \frac{1}{2} \text{STFT}_R(n, k) + \frac{1}{4} \text{STFT}_R(n, k - 1) + \frac{1}{4} \text{STFT}_R(n, k + 1).$$

This recursive calculation is important for hardware implementation of the STFT and other related time-frequency representations (e.g., the higher order representations implementations based on the STFT).

A system for the recursive implementation of the STFT is shown in Figure 3.5. The STFT obtained by using the rectangular window is denoted by  $\text{STFT}_R(n, k)$ , Figure 3.5, while the values of coefficients are

$$(a_{-1}, a_0, a_1) = \left( \frac{1}{4}, \frac{1}{2}, \frac{1}{4} \right),$$

$$(a_{-1}, a_0, a_1) = (0.23, 0.54, 0.23),$$

$$(a_{-2}, a_{-1}, a_0, a_1, a_2) = (0.04, 0.25, 0.42, 0.25, 0.04)$$

for the Hann(ing), Hamming and Blackman windows, respectively.

### 3.03.2.2.1 Filter bank STFT implementation

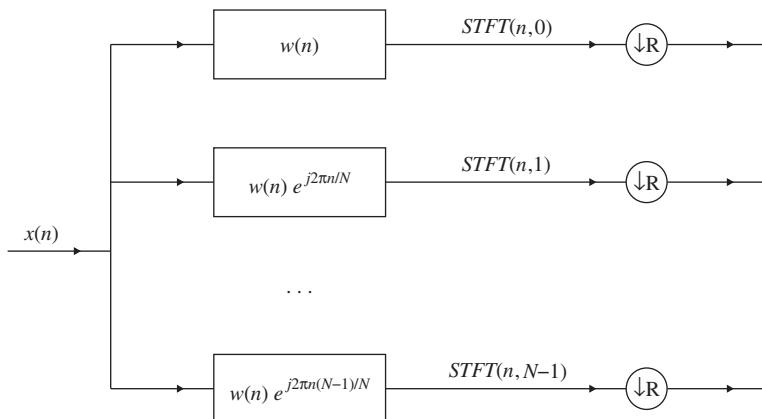
According to (3.4), the STFT can be written as a convolution

$$\text{STF}(t, \Omega) = \int_{-\infty}^{\infty} x(t - \tau) w(\tau) e^{j\Omega\tau} d\tau = x(t) *_t [w(t) e^{j\Omega t}],$$

where an even, real valued, window function is assumed,  $w(\tau) = w(-\tau)$ . For a discrete set of frequencies  $\Omega_k = k\Delta\Omega = 2\pi k/(N\Delta t)$ ,  $k = 0, 1, 2, \dots, N - 1$ , and discrete values of signal, we get that the discrete STFT, (3.16), is an output of the filter bank with impulse responses

$$h_k(n) = w(n) e^{j2\pi kn/N}, \quad k = 0, 1, \dots, N - 1$$

what is illustrated in Figure 3.6.

**FIGURE 3.6**

Filter bank realization of the STFT.

*Illustrative example:* In order to additionally explain this form of realization, as well as to introduce various possibilities for splitting the whole time-frequency plane, let us assume that the total length of discrete signal  $x(n)$  is  $M = KN$ , where  $N$  is the length of the window used for the STFT analysis. If the signal was sampled by  $\Delta t$ , then the time-frequency region of interest in the analog domain is  $t \in [0, KN\Delta t]$  and  $\Omega \in [0, \Omega_m]$ , with  $\Omega_m = \pi/\Delta t$ , or  $\omega \in [0, \pi]$  and  $n \in [0, KN]$  in the discrete time domain. For the illustration we will assume  $M = 16$ .

The first special case of the STFT is the signal itself (in discrete time domain). This case corresponds to window  $w(n) = \delta(n)$ . Here, there is no information about the frequency content, since the STFT of one sample  $x(n)$  is the sample itself, i.e.,  $\text{STFT}(n, \omega) = x(n)$ , for the whole frequency range. The whole considered time-frequency plane is divided as in Figure 3.7a.

Let us now consider a two samples rectangular window,  $w(n) = \delta(n) + \delta(n+1)$ , with  $N = 2$ . The corresponding two samples STFT is

$$\text{STFT}(n, 0) = x(n) + x(n-1)$$

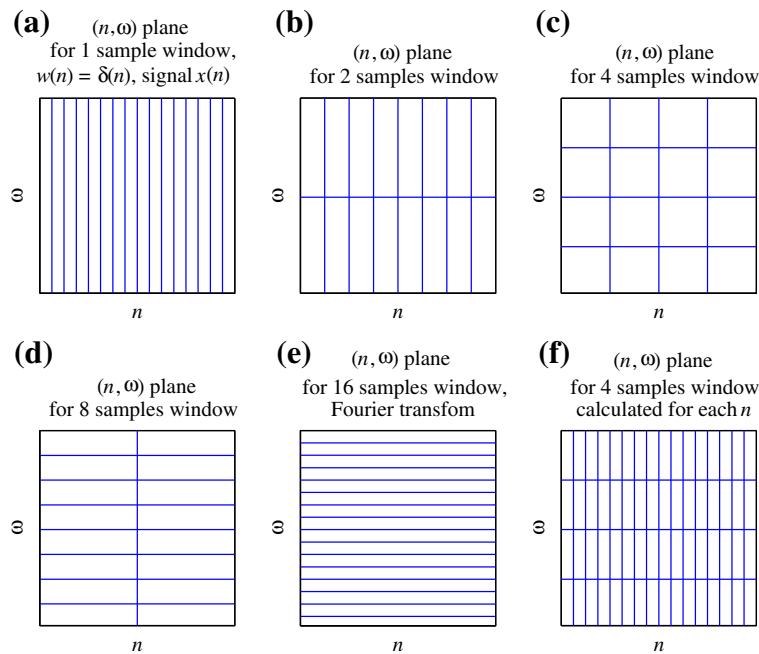
for  $k = 0$  (corresponding to  $\omega = 0$ ) and

$$\text{STFT}(n, 1) = x(n) - x(n-1)$$

for  $k = 1$  (corresponding to  $\omega = \pi$ ). Thus, the whole frequency interval is represented by the low frequency value  $\text{STFT}(n, 0)$  and the high frequency value  $\text{STFT}(n, 1)$ . From the signal reconstruction point of view, we can skip one sample in the STFT calculation and calculate  $\text{STFT}(n+2, 0) = x(n+2) + x(n+1)$  and  $\text{STFT}(n+2, 1) = x(n+2) - x(n+1)$ , and so on. It means that  $\text{STFT}(n, k)$  could be down-sampled in discrete time  $n$  by a factor of 2. The signal reconstruction, in this case, is based on

$$x(n) = [\text{STFT}(n, 0) + \text{STFT}(n, 1)]/2,$$

$$x(n-1) = [\text{STFT}(n, 0) - \text{STFT}(n, 1)]/2,$$

**FIGURE 3.7**

Time-frequency plane for a signal  $x(n)$  having 16 samples: (a) The STFT of signal  $x(n)$  using one-sample window. (Note that  $\text{STFT}(n, k) = x(n)$ , for  $w(n) = \delta(n)$ .) (b) The STFT obtained by using a two-samples window  $w(n)$ , without overlapping. (c) The STFT obtained by using a four-samples window  $w(n)$ , without overlapping. (d) The STFT obtained by using an eight-samples window  $w(n)$ , without overlapping. (e) The STFT obtained by using a 16-samples window  $w(n)$ , without overlapping. (Note that  $\text{STFT}(n, k) = X(k)$  for  $w(n) = 1$  for all  $n$ .) (f) The STFT obtained by using a four-samples window  $w(n)$ , calculated for each  $n$ . Overlapping is present in this case.

where  $\text{STFT}(n, k)$  is calculated for every other  $n$  (every even or every odd  $n$ ). In the time-frequency plane, the time resolution is now  $2\Delta t$  corresponding to the two samples, and the whole frequency interval is divided into two parts (low-pass and high-pass), Figure 3.7b. In this way, we can proceed and divide the low-pass part of the STFT, i.e., signal  $x_l(n) = x(n) + x(n-1)$ , into two parts, its low-pass and high-pass parts, according to  $\text{STFT}_l(n, 0) = x_l(n) + x_l(n-1)$  and  $\text{STFT}_l(n, 1) = x_l(n) - x_l(n-1)$ . The same can be done to the high pass part  $x_h(n) = x(n) - x(n-1)$ . In this way, we divide the frequency range into four parts and the STFT can be down-sampled in time by 4 (time resolution corresponding to the four sampling intervals), Figure 3.7b. This may be continued, until we split the frequency region into  $KN$  intervals, and down-sample the STFT in time by a factor of  $M = KN$ , thus producing the spectral content with high resolution, without any time-resolution (time resolution is equal to the whole considered time interval), Figure 3.7c–e.

The second special case is the FT of the whole signal,  $X(k)$ ,  $k = 0, 1, 2, \dots, KN - 1$ . Its contains  $KN$  frequency points, but there is no time resolution, since it is calculated over the entire time interval,

Figure 3.7e. Let us split the signal into two parts  $x_1(n)$  for  $n = 0, 1, 2, \dots, M/2 - 1$  and  $x_2(n)$  for  $n = M/2, M/2 + 1, \dots, M - 1$  (“lower” time and “higher” time intervals). By calculating the FT of  $x_1(n)$  we get a half of the frequency samples within the whole frequency interval. In the time domain, these samples correspond to the half of the original signal duration, i.e., to the lower time interval  $n = 0, 1, 2, \dots, M/2 - 1$ . The same holds for signal  $x_2(n)$ , Figure 3.7d. In this way, we may continue and split the signal into four parts, Figure 3.7c, and so on.

### 3.03.2.2.2 Time and frequency varying windows

In general, we may split the original signal into  $K$  signals of duration  $N$ :  $x_1(n)$  for  $n = 0, 1, 2, \dots, N-1$ ,  $x_2(n)$  for  $n = N, N+1, \dots, 2N-1$ , and so on until  $x_K(n)$  for  $n = (K-1)N, (K-1)N+1, \dots, KN-1$ . Obviously by each signal  $x_i(n)$  we cover  $N$  samples in time, while corresponding STFT covering  $N$  samples of the whole frequency interval. Thus the time-frequency interval is divided as in Figure 3.7.

Consider a discrete-time signal  $x(n)$  of the length  $N$  and its discrete Fourier transform (DFT)  $X(k)$ . The STFT, with a rectangular window of the width  $M$ , is:

$$\text{STFT}(n, k) = \sum_{m=0}^{M-1} x(n+m)e^{-j2\pi mk/M}. \quad (3.17)$$

In a matrix form, it can be written as:

$$\text{STFT}_M(n) = \mathbf{W}_M \mathbf{x}(n), \quad (3.18)$$

where  $\text{STFT}_M(n)$  and  $\mathbf{x}(n)$  are vectors:

$$\begin{aligned} \text{STFT}_M(n) &= [\text{STFT}(n, 0), \text{STFT}(n, 1), \dots, \text{STFT}(n, M-1)]^T, \\ \mathbf{x}(n) &= [x(n), x(n+1), \dots, x(n+M-1)]^T, \end{aligned} \quad (3.19)$$

and  $\mathbf{W}_M$  is the  $M \times M$  DFT matrix with coefficients:

$$W_M(m, k) = \exp(-j2\pi km/M).$$

Considering non-overlapping contiguous data segments, the next STFT will be calculated at instant  $n + M$ , as follows:

$$\text{STFT}_M(n+M) = \mathbf{W}_M \mathbf{x}(n+M).$$

The last STFT at instant  $n + N - M$ , (assuming that  $N/M$  is an integer) is:

$$\text{STFT}_{N-M}(n+N-M) = \mathbf{W}_M \mathbf{x}(n+N-M).$$

Combining all STFT vectors in a single vector, we obtain:

$$\begin{bmatrix} \text{STFT}_M(0) \\ \text{STFT}_M(M) \\ \vdots \\ \text{STFT}_M(N-M) \end{bmatrix} = \begin{bmatrix} \mathbf{W}_M & \mathbf{0}_M & \cdots & \mathbf{0}_M \\ \mathbf{0}_M & \mathbf{W}_M & \cdots & \mathbf{0}_M \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_M & \mathbf{0}_M & \cdots & \mathbf{W}_M \end{bmatrix} \begin{bmatrix} \mathbf{x}(0) \\ \mathbf{x}(M) \\ \vdots \\ \mathbf{x}(N-M) \end{bmatrix}, \quad (3.20)$$

where  $\mathbf{0}_M$  is a  $M \times M$  zero matrix. The vector  $[\mathbf{x}(0), \mathbf{x}(M), \dots, \mathbf{x}(N-M)]^T$  is the signal vector  $\mathbf{x}$ , since

$$\mathbf{x} = [\mathbf{x}(0), \mathbf{x}(M), \dots, \mathbf{x}(N-M)]^T = [x(0), x(1), \dots, x(N-1)]^T. \quad (3.21)$$

### Time varying window

A similar relations can be obtained if the STFT with a varying window width (for each time instant  $n$ ) is considered. Assume that we use the window width  $M_0$  for the instant  $n = 0$  and calculate  $\text{STFT}_{M_0}(0) = \mathbf{W}_{M_0}\mathbf{x}(0)$ . Then, we skip  $M_0$  signal samples. At  $n = M_0$ , a window of  $M_1$  width is used to calculate  $\text{STFT}_{M_1}(M_0) = \mathbf{W}_{M_1}\mathbf{x}(M_0)$ , and so on, until the last one  $\text{STFT}_{M_L}(M_L) = \mathbf{W}_{M_L}\mathbf{x}(N - M_L)$  is obtained. Assuming that  $M_0 + M_1 + \dots + M_L = N$ , we can write:

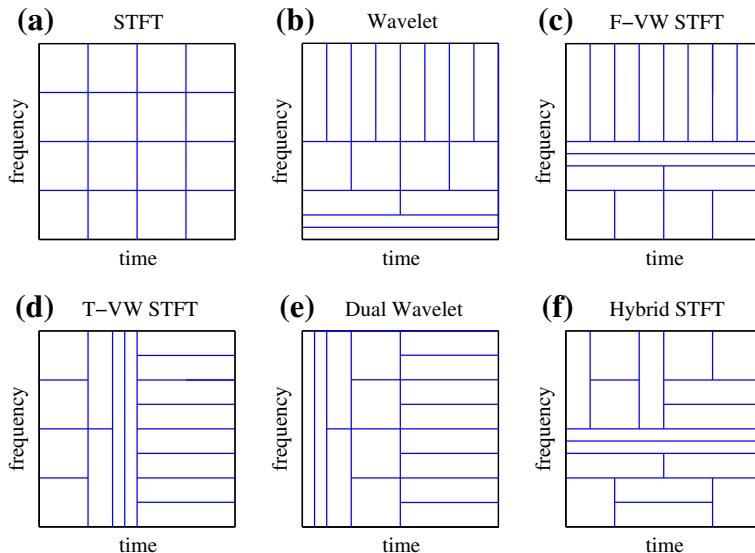
$$\begin{bmatrix} \text{STFT}_{M_0}(0) \\ \text{STFT}_{M_1}(M_0) \\ \vdots \\ \text{STFT}_{M_L}(N - M_L) \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{M_0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_{M_1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{W}_{M_L} \end{bmatrix} \mathbf{x}. \quad (3.22)$$

As a special case of time-varying windows, consider a dual wavelet form (see Figure 3.8). It means that for a “low time” we have the best time-resolution, without frequency resolution. This is achieved with a one sample window. So for “low time,” at  $n = 0$ , the best time resolution is obtained with  $M_0 = 1$ ,

$$\text{STFT}_1(0) = \mathbf{W}_1\mathbf{x}(0) = x(0).$$

For an even number  $N$ , the same should be repeated for the next lowest time,  $n = 1$ , when:

$$\text{STFT}_1(1) = \mathbf{W}_1\mathbf{x}(1) = x(1).$$



**FIGURE 3.8**

Time and frequency lattice illustration for: (a) the STFT with a constant window, (b) the wavelet transform, (c) a frequency-varying window (F-VW) STFT, (d) a time-varying window (T-VW) STFT, (e) the dual wavelet transform, and (f) a hybrid STFT with time and frequency varying window.

At the time instant  $n = 2$ , we now decrease time resolution and increase frequency resolution by factor of 2. It is done by using a two samples window in the STFT,  $M_2 = 2$ , so we have:

$$\text{STFT}_2(2) = \mathbf{W}_2 \mathbf{x}(2) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x(2) \\ x(3) \end{bmatrix}.$$

The next instant, for the non-overlapping STFT calculation is  $n = 4$ , when we again increase the frequency resolution and decrease time resolution by using window of the width  $M_4 = 4$ ,  $\text{STFT}_4(4) = \mathbf{W}_4 \mathbf{x}(4)$ . Continuing in this way, until the end of signal, we get a resulting matrix,

$$\begin{bmatrix} \text{STFT}_1(0) \\ \text{STFT}_1(1) \\ \text{STFT}_2(2) \\ \text{STFT}_4(4) \\ \vdots \\ \text{STFT}_{N/2}(N/2) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{W}_2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \mathbf{W}_4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathbf{W}_{N/2} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ \vdots \\ x(N-1) \end{bmatrix}. \quad (3.23)$$

This matrix corresponds to the dual wavelet transform, since we used the wavelet transform reasoning, but in the time instead of frequency.

#### *Frequency varying window*

The STFT can be calculated by using the signal's DFT instead of the signal. There is a direct relation between the time and the frequency domain STFT via coefficients of the form  $\exp(j2\pi nk/M)$ . A dual form of the STFT is:

$$\begin{aligned} \text{STFT}(n, k) &= \frac{1}{M} \sum_{m=0}^{M-1} X(k+m)e^{j2\pi mn/M}, \\ \text{STFT}_M(k) &= \mathbf{W}_M^{-1} \mathbf{X}(k). \end{aligned} \quad (3.24)$$

Frequency domain window may be of frequency varying width (Figure 3.8). This form is dual to the time-varying form.

#### *Hybrid time and frequency varying windows*

In general, spectral content of signal changes in time and frequency in an arbitrary manner. There are several methods in the literature that adapt windows or basis functions to the signal form for each time instant or even for every considered time and frequency point in the time-frequency plane (e.g., as in Figure 3.8). Selection of the most appropriate form of the basis functions (windows) for each time-frequency point includes a criterion for selecting the optimal window width (basis function scale) for each point.

If we consider a signal with  $N$  samples, then its time-frequency plane can be split in a large number of different grids for the non-overlapping STFT calculation. All possible variations of time-varying or frequency-varying windows, are just special cases of general hybrid time-varying grid. Covering a time-frequency  $N \times N$  plane, by any combination of non-overlapping rectangular areas, whose individual area is  $N$ , corresponds to a valid non-overlapping STFT calculation scheme. The total number of ways  $F(N)$ , how an  $N \times N$  plane can be split (into non-overlapping STFTs of area  $N$  with dyadic time-varying windows) is:

$N$	1	4	6	8	12	14	16	...
$F(N)$	1	6	18	56	542	1690	5272	

The approximative formula for  $F(N)$  can be written in the form, [6]:

$$F(N) \approx \left\lfloor 1.0366(1.7664)^{N-1} \right\rfloor, \quad (3.25)$$

where  $\lfloor \cdot \rfloor$  stands for an integer part of the argument. It holds with relative error smaller than 0.4% for  $1 \leq N \leq 1024$ . For example, for  $N = 16$  we have **5272** different ways to split time-frequency plane into non-overlapping time-frequency regions with dyadic time-varying windows. Of course, most of them cannot be considered within the either time-varying or frequency-varying window case, since they are time-frequency varying (hybrid) in general.

### 3.03.2.2.3 Signal reconstruction form the discrete STFT

Signal reconstruction from non-overlapping STFT values is obvious, according to (3.20), (3.22), or (3.23).

Signal can be reconstructed from the STFT calculated with  $N$  signal samples, if the calculated STFT is overlapped, i.e., down-sampled in time by  $R \leq N$ . Here, the signal general reconstruction scheme from the STFT values, overlapped in time, will be presented. Consider the STFT, (3.16), written in a vector form, as

$$\begin{aligned} \text{STFT}(n, k) &= \sum_{m=-N/2}^{N/2-1} w(m)x(n+m)e^{-j2\pi mk/N} \\ &= e^{j2\pi nk/N} \sum_{m=-N/2}^{N/2-1} w(n-m)x(m)e^{-j2\pi mk/N}, \\ \text{STFT}(n) &= \mathbf{W}_N \mathbf{H}_w \mathbf{x}(n) \end{aligned} \quad (3.26)$$

where the vector **STFT**( $n$ ) contains all frequency values of the STFT, for a given  $n$ ,

$$\text{STFT}(n) = [STFT(n, 0), STFT(n, 1), \dots, STFT(n, N - 1)]^T.$$

Signal vector is

$$\mathbf{x}(n) = [x(n - N/2), x(n - N/2 + 1), \dots, x(n + N/2 - 1)]^T,$$

while  $\mathbf{W}_N$  is the DFT matrix with coefficients  $W_N(n, k) = e^{-j2\pi mk/N}$ . A diagonal matrix  $\mathbf{H}_w$  is the window matrix  $H_w(m, m) = w(m)$  and  $H_w(m, n) = 0$  for  $m \neq n$ .

It has been assumed that the STFTs are calculated with a step  $1 \leq R \leq N$  in time. So they are overlapped for  $R < N$ . Available STFT values are

$$\begin{gathered} \dots \\ \text{STFT}(n - 2R), \\ \text{STFT}(n - R), \\ \text{STFT}(n), \\ \text{STFT}(n + R), \\ \text{STFT}(n + 2R), \\ \dots \end{gathered}$$

Based on the available STFT values (3.26), the windowed signal values can be reconstructed as

$$\mathbf{H}_w \mathbf{x}(n + iR) = \mathbf{W}_N^{-1} \text{STFT}(n + iR), \quad i = \dots - 2, -1, 0, 1, 2, \dots \quad (3.27)$$

For  $m = -N/2, -N/2 + 1, \dots, N/2 - 1$  we get

$$w(m)x(n + iR + m) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \text{STFT}(n + iR, k) e^{j2\pi mk/N} \quad (3.28)$$

Let us reindex the reconstructed signal value (3.28) by substitution  $m = l - iR$

$$w(l - iR)x(n + 1) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \text{STFT}(n + iR, k) e^{j2\pi lk/N} e^{-j2\pi Rk/N} \\ -N/2 \leq l - iR \leq N/2 - 1.$$

By summing over  $i$  satisfying  $-N/2 \leq l - iR \leq N/2 - 1$  we get that the reconstructed signal is undistorted (up to a constant) if

$$c(l) = \sum_i w(l - iR) = \text{const.} \quad (3.29)$$

*Special cases:*

1. For  $R = N$  (non-overlapping), relation (3.29) is satisfied for the rectangular window, only.
2. For a half of the overlapping period,  $R = N/2$ , condition (3.29) is met for the rectangular, Hann(ing), Hamming, triangular, ..., windows.
3. The same holds for  $R = N/2, N/4, N/8$ , if the values of  $R$  are integers.
4. For  $R = 1$  (the STFT calculation in each available time instant), any window satisfies the inversion relation.

In analysis of non-stationary signals our primary interest is not in signal reconstruction with the fewest number of calculation points. Rather, we are interested in tracking signals' non-stationary parameters, like for example, instantaneous frequency. These parameters may significantly vary between neighboring time instants  $n$  and  $n + 1$ . Quasi-stationarity of signal within  $R$  samples (implicitly assumed when down-sampling by factor of  $R$  is done) in this case is not a good starting point for the analysis. Here, we have to use the time-frequency analysis of signal at each instant  $n$ , without any down-sampling. Very efficient realizations, for this case, are the recursive ones.

### 3.03.2.3 Gabor transform

The Gabor transform is the oldest time-frequency form applied in the signal processing field (since the Wigner distribution remained for a long time within the quantum mechanics, only). It has been introduced with the aim to expand a signal  $x(t)$  into a series of time-frequency shifted elementary

functions  $w(t - nT)e^{jk\Delta\Omega t}$  (logons)

$$x(t) = \sum_{k=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} a(n, k) w(t - nT)e^{jk\Delta\Omega t}. \quad (3.30)$$

The Gabor's original choice was the Gaussian window

$$w(\tau) = \exp\left(-\pi \frac{\tau^2}{2T^2}\right),$$

due to its best concentration in the time-frequency plane. Gabor also used  $\Delta\Omega = 2\pi/T$ .

In the time frequency-domain, the elementary functions  $w(\tau - nT)e^{jk\Delta\Omega\tau}$  are shifted in time and frequency for  $nT$  and  $k\Delta\Omega$ , respectively.

For the analysis of signal, Gabor has divided the whole information (time-frequency) plane by a grid at  $t = nT$  and  $\Omega = k\Delta\Omega$ , with area of elementary cell  $T\Delta\Omega/2\pi = 1$ . Then, the signal is expanded at the central points of the grid ( $nT, k\Delta\Omega$ ) using the elementary atom functions  $\exp(-\pi(t - nT)^2/(2T^2)) \exp(jk\Delta\Omega t)$ .

If the elementary functions  $w(\tau - nT)e^{jk\Delta\Omega\tau}$  were orthogonal to each other, i.e.,

$$\int_{-\infty}^{\infty} w(\tau - nT)e^{jk\Delta\Omega\tau} w^*(\tau - mT)e^{-jl\Delta\Omega\tau} d\tau = \delta(n - m)\delta(k - l),$$

then by multiplying (3.30) by  $w^*(t - mT)e^{-jl\Delta\Omega t}$  and integrating over  $t$  we could get

$$a(n, k) = \int_{-\infty}^{\infty} x(\tau) w^*(\tau - nT)e^{-jk\Delta\Omega\tau} d\tau,$$

which corresponds to the STFT at  $t = nT$ . However, the elementary logons do not satisfy the orthogonality property. Gabor originally proposed an iterative procedure for the calculation of  $a(n, k)$ .

Interest in the Gabor transform, had been lost for decades, until a simplified procedure for the calculation of coefficients has been developed. This procedure is based on introducing elementary signal  $\gamma(\tau)$  dual to  $w(\tau)$  such that

$$\int_{-\infty}^{\infty} w(\tau - nT)e^{jk\Delta\Omega\tau} \gamma^*(\tau - mT)e^{-jl\Delta\Omega\tau} d\tau = \delta(n - m)\delta(k - l)$$

holds (Bastiaans logons). Then

$$a(n, k) = \int_{-\infty}^{\infty} x(\tau) \gamma^*(\tau - nT)e^{-jk\Delta\Omega\tau} d\tau.$$

However, the dual function  $\gamma(\tau)$  has a poor time-frequency localization. In addition, there is no stable algorithm to reconstruct the signal with the critical sampling condition  $\Delta\Omega T = 2\pi$ . One solution is to use an oversampled set of functions with  $\Delta\Omega T < 2\pi$ .

### 3.03.2.4 Stationary phase method

When the signal

$$x(t) = A(t)e^{j\phi(t)}$$

is not of simple analytic form, it may be possible, in some cases, to obtain an approximative expression for the FT by using the method of stationary phase [24, 25].

The **method of stationary phase** states:

If the function  $\phi(t)$  is monotonous and  $A(t)$  is sufficiently smooth function, then

$$\int_{-\infty}^{\infty} A(t) e^{j\phi(t)} e^{-j\Omega t} dt \simeq A(t_0) e^{j\phi(t_0)} e^{-j\Omega t_0} \sqrt{\frac{2\pi j}{\phi''(t_0)}}, \quad (3.31)$$

where  $t_0$  is the solution of

$$\phi'(t_0) = \Omega.$$

The most significant contribution to the integral on the left side of (3.31) comes from the region where the phase of the exponential function  $\exp(j(\phi(t) - \Omega t))$  is stationary in time, since the contribution of the intervals with fast varying  $\phi(t) - \Omega t$  tends to zero. In other words, in the considered time region, the signal's phase  $\phi(t)$  behaves as  $\Omega t$ . Thus, we may say that the rate of the phase change,  $\phi'(t)$ , for that particular instant is its instantaneous frequency corresponding to frequency  $\Omega$ . The stationary point  $t_0$  of phase  $\phi(t) - \Omega t$ , of signal  $A(t)e^{j\phi(t)-j\Omega t}$ , is obtained as a solution of

$$\left. \frac{d(\phi(t) - \Omega t)}{dt} \right|_{t=t_0} = 0.$$

By expanding  $\phi(t) - \Omega t$  into a Taylor series up to the second order term, around the stationary point  $t_0$ , we have

$$\begin{aligned} \phi(t) - \Omega t &\simeq \phi(t_0) - \Omega t_0 + \frac{1}{2} \left. \frac{d^2\phi(t)}{dt^2} \right|_{t=t_0} (t - t_0)^2, \\ \int_{-\infty}^{\infty} A(t) e^{j\phi(t)} e^{-j\Omega t} dt &\simeq \int_{-\infty}^{\infty} A(t) e^{j(\phi(t_0) - \Omega t_0 + \frac{1}{2}\phi''(t_0)(t - t_0)^2)} dt. \end{aligned} \quad (3.32)$$

Using the FT pair

$$\exp(j\alpha t^2/2) \longleftrightarrow \sqrt{\frac{2\pi j}{\alpha}} \exp\left(-j\frac{\Omega^2}{2\alpha}\right),$$

for a large  $\alpha = \phi''(t_0)$  it follows FT  $\{\sqrt{\alpha/(2\pi j)} \exp(j\alpha t^2/2)\} \rightarrow 1$ , i.e.,

$$\lim_{\alpha \rightarrow \infty} \sqrt{\frac{\alpha}{2\pi j}} \exp(j\alpha t^2/2) = \delta(t). \quad (3.33)$$

Relation (3.31) is now easily obtained from (3.32) with (3.33), for large  $\phi''(t_0)$ .

If the equation  $\phi'(t_0) = \Omega$  has two (or more) solutions  $t_0^+$  and  $t_0^-$  then the integral on the left side of (3.31) is equal to the sum of functions at both (or more) stationary phase points. Finally, this relation holds for  $\phi''(t_0) \neq 0$ . If  $\phi''(t_0) = 0$  then similar analysis may be performed, using the lowest non-zero phase derivative at the stationary phase point.

**Example 2.** Consider a frequency modulated signal

$$x(t) = \exp(jat^{2N}).$$

By using the stationary phase method we get that the stationary phase point is  $2N\alpha t_0^{2N-1} = \Omega$  with  $t_0 = (\frac{\Omega}{2Na})^{1/(2N-1)}$  and  $\phi''(t_0) = 2N(2N-1)\alpha(\frac{\Omega}{2Na})^{(2N-2)/(2N-1)}$ . The amplitude and phase of  $X(\Omega)$ , according to (3.31), are

$$|X(\Omega)|^2 \simeq \left| \frac{2\pi}{\phi''(t_0)} \right| = \left| \frac{2\pi}{(2N-1)\Omega} \left( \frac{\Omega}{2aN} \right)^{1/(2N-1)} \right|,$$

$$\arg\{X(\Omega)\} \simeq \phi(t_0) - \Omega t_0 + \pi/4 = \frac{(1-2N)}{2N} \Omega \left( \frac{\Omega}{2aN} \right)^{1/(2N-1)} + \pi/4$$

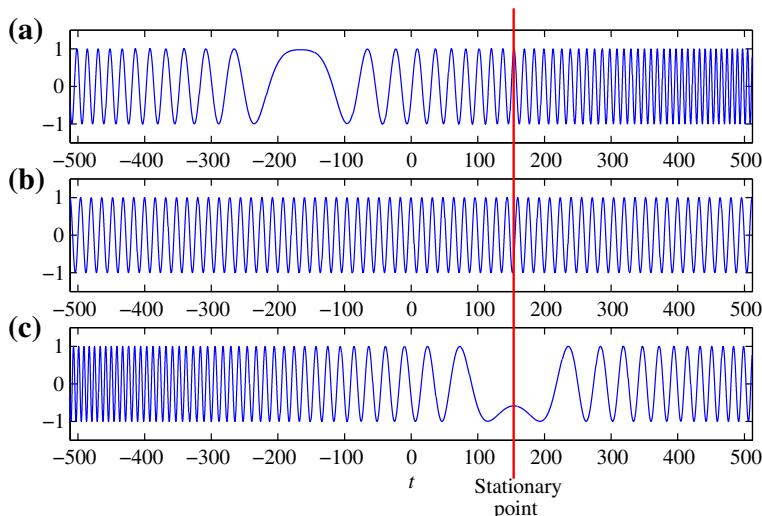
for a large value of  $a$ . The integrand in (3.31) is illustrated in Figure 3.9, for  $N = 1$ , when  $|X(\Omega)|^2 = |\pi/a|$  and  $\arg\{X(\Omega)\} = -\Omega^2/(4a) + \pi/4$ .

### 3.03.2.4.1 Instantaneous frequency

Here, we present a simple instantaneous frequency (IF) interpretation when signal may be considered as stationary within the localization window (quasistationary signal). Consider a signal  $x(t) = A(t)e^{j\phi(t)}$ , within the window  $w(\tau)$  of the width  $2T$ . If we can assume that the amplitude variations are small and the phase variations are almost linear within  $w(\tau)$ , i.e.,

$$A(t + \tau) \simeq A(t),$$

$$\phi(t + \tau) \simeq \phi(t) + \phi'(t)\tau,$$



**FIGURE 3.9**

Stationary phase method illustration: (a) a real part of a frequency modulated signal, (b) a real part of the demodulation signal, and (c) a real part of the stationary phase method integrand.

then it holds

$$x(t + \tau) \simeq A(t)e^{j\phi(t)}e^{j\phi'(t)\tau}.$$

Thus, around a given instant  $t$ , the signal behaves as a sinusoid in the  $\tau$  domain with amplitude  $A(t)$ , phase  $\phi(t)$ , and frequency  $\phi'(t)$ . The first derivative of phase,  $\phi'(t)$ , plays the role of frequency within the considered lag interval around  $t$ .

The stationary phase method relates the spectral content at the frequency  $\Omega$  with the signal's value at time instant  $t$ , such that  $\phi'(t) = \Omega$ . A signal in time domain, that satisfies stationary phase method conditions, contributes at the considered instant  $t$  to the FT at the corresponding frequency

$$\Omega(t) = \phi'(t).$$

Additional comments on this relation are given within the stationary phase method presentation subsection.

The instantaneous frequency is not so clearly defined as the frequency in the FT. For example, the frequency in the FT has the physical interpretation as the number of signal periods within the considered time interval, while this interpretation is not possible if a single time instant is considered. Thus, a significant caution has to be taken in using this notion. Various definitions and interpretations of the IF are given in the literature, with the most comprehensive review presented by Boashash.

**Example 3.** The STFT of the signal

$$x(t) = e^{jat^2}$$

can be approximately calculated for a large  $a$ , by using the method of stationary phase, as

$$\begin{aligned} \text{STFT}(t, \Omega) &= \int_{-\infty}^{\infty} e^{ja(t+\tau)^2} w(\tau) e^{-j\Omega\tau} d\tau \simeq e^{jat^2} e^{j(2at-\Omega)\tau_0} e^{jat_0^2} w(\tau_0) \sqrt{\frac{2\pi j}{2a}} \\ &= e^{jat^2} e^{-j(2at-\Omega)^2/4a} w\left(\frac{\Omega - 2at}{2a}\right) \sqrt{\frac{\pi j}{a}}, \end{aligned}$$

where the stationary point  $\tau_0$  is obtained from

$$2a(t + \tau_0) = \Omega.$$

Note that the absolute value of the STFT reduces to

$$|\text{STFT}(t, \Omega)| \simeq \left| w\left(\frac{\Omega - 2at}{2a}\right) \right| \sqrt{\frac{\pi}{a}}. \quad (3.34)$$

In this case, the width of  $|\text{STFT}(t, \Omega)|$  in frequency does not decrease with the increase of the window  $w(\tau)$  width. The width of  $|\text{STFT}(\Omega, t)|$  around the central frequency  $\Omega = 2at$  is

$$D = 4aT,$$

where  $2T$  is the window width in time domain. Note that this relation holds for a wide window  $w(\tau)$  such that the stationary phase method may be applied. If the window is narrow with respect to the phase variations of the signal, the STFT width is defined by the width of the FT of window, being proportional to  $1/T$ . Thus, the overall STFT width is equal to the sum of the frequency variation caused width and the window's FT width, i.e.,

$$D_o = 4aT + 2c/T,$$

where  $c$  is a constant defined by the window shape. Therefore, there is a window width  $T$  producing the narrowest possible STFT for this signal. It is obtained by equating the derivative of the overall width to zero,  $2a - c/T^2 = 0$ , which results in

$$T_o = \sqrt{c/(2a)}.$$

As expected, for a sinusoid,  $a \rightarrow 0$ ,  $T_o \rightarrow \infty$ .

Consider now the general form of FM signal

$$x(t) = Ae^{j\phi(t)},$$

where  $\phi(t)$  is a differentiable function. Its STFT is of the form

$$\begin{aligned} \text{STFT}(t, \Omega) &= \int_{-\infty}^{\infty} Ae^{j\phi(t+\tau)} w(\tau) e^{-j\Omega\tau} d\tau \\ &= \int_{-\infty}^{\infty} Ae^{j[\phi(t)+\phi'(t)\tau+\phi''(t)\tau^2/2+\dots]} w(\tau) e^{-j\Omega\tau} d\tau \\ &= Ae^{j\phi(t)} \text{FT} \left\{ e^{j\phi'(t)\tau} \right\} *_{\Omega} \text{FT}\{w(\tau)\} *_{\Omega} \text{FT} \left\{ \sum_{k=2}^{\infty} e^{j\phi^{(k)}(t)\tau^k/k!} \right\}, \end{aligned}$$

where  $\phi(t + \tau)$  is expanded into the Taylor series around  $t$  as

$$\phi(t + \tau) = \phi(t) + \phi'(t)\tau + \phi''(t)\tau^2/2 + \dots + \phi^{(k)}(t)\tau^k/k! + \dots$$

Neglecting the higher order terms in the Taylor series we can write

$$\begin{aligned} \text{STFT}(t, \Omega) &= Ae^{j\phi(t)} \text{FT} \left\{ e^{j\phi'(t)\tau} \right\} *_{\Omega} \text{FT}\{w(\tau)\} *_{\Omega} \text{FT} \left\{ e^{j\phi''(t)\tau^2/2} \right\} \\ &= 2\pi Ae^{j\phi(t)} \delta(\Omega - \phi'(t)) *_{\Omega} W(\Omega) *_{\Omega} e^{-j\Omega^2/(2\phi''(t))} \sqrt{\frac{2\pi j}{\phi''(t)}}, \end{aligned}$$

where  $*_{\Omega}$  denotes the convolution in frequency. As expected, the influence of the window is manifested as a spread of ideal concentration  $\delta(\Omega - \phi'(t))$ . In addition, the term due to the frequency non-stationarity  $\phi''(t)$  causes an additional spread. This relation confirms our previous conclusion that the overall STFT width is the sum of the width of  $W(\Omega)$  and the width caused by the signal's non-stationarity.

### 3.03.2.5 Local polynomial Fourier transform

There are signals whose form is known up to an unknown set of parameters. For example, many signals could be expressed as polynomial-phase signal

$$x(t) = Ae^{j(\Omega_0 t + a_1 t^2 + a_2 t^3 + \dots)}$$

with (unknown) parameters  $\Omega_0, a_1, a_2, \dots$ . High concentration of such signals in the frequency domain is achieved by the polynomial FT defined by

$$\text{PFT}_{\Omega_1, \Omega_2, \dots}(t, \Omega) = \int_{-\infty}^{\infty} x(t) e^{-j(\Omega t + \Omega_1 t^2 + \Omega_2 t^3 + \dots)} dt$$

when parameters  $\Omega_1, \Omega_2, \dots$  are equal to the signal parameters  $a_1, a_2, \dots$ . Finding values of unknown parameters  $\Omega_1, \Omega_2, \dots$  that match signal parameters can be done by a simple search over a possible set of values for  $\Omega_1, \Omega_2, \dots$  and stopping the search when the maximally concentrated distribution is achieved (in ideal case, the delta function at  $\Omega = \Omega_0$ , for  $\Omega_1 = a_1, \Omega_2 = a_2, \dots$  is obtained). This procedure may be time consuming.

For non-stationary signals, this approach may be used if the non-stationary signal could be considered as a signal with constant parameters within the analysis window. In that case, the local polynomial Fourier transform (LPFT), proposed by Katkovnik, may be used [26]. It is defined as

$$\text{LPFT}_{\Omega_1, \Omega_2, \dots}(t, \Omega) = \int_{-\infty}^{\infty} x(t + \tau) w(\tau) e^{-j(\Omega\tau + \Omega_1\tau^2 + \Omega_2\tau^3 + \dots)} d\tau.$$

**Example 4.** Consider the second order polynomial-phase signal

$$x(t) = e^{j(\Omega_0 t + a_1 t^2)}.$$

Its LPFT has the form

$$\begin{aligned} \text{LPFT}_{\Omega_1}(t, \Omega) &= \int_{-\infty}^{\infty} x(t + \tau) w(\tau) e^{-j(\Omega\tau + \Omega_1\tau^2)} d\tau. \\ &= e^{j(\Omega_0 t + \Omega_1 t^2)} \int_{-\infty}^{\infty} w(\tau) e^{-j(\Omega - \Omega_0 - 2a_1 t)\tau} e^{j(\Omega_1 - a_1)\tau^2} d\tau. \end{aligned}$$

For  $\Omega_1 = a_1$ , the second-order phase term does not introduce any distortion to the local polynomial spectrogram,

$$|\text{LPFT}_{\Omega_1=a_1}(t, \Omega)|^2 = |W(\Omega - \Omega_0 - 2a_1 t)|^2$$

with respect to the spectrogram of a sinusoid with constant frequency. For a wide window  $w(\tau)$ , like in the case of the STFT of a pure sinusoid, we achieve high concentration.

The LPFT could be considered as the FT of signal demodulated with  $e^{-j(\Omega_1\tau^2 + \Omega_2\tau^3 + \dots)}$ . Thus, if we are interested in signal filtering, we can find the coefficients  $\Omega_1, \Omega_2, \dots$ , demodulate the signal by multiplying it with  $e^{-j(\Omega_1\tau^2 + \Omega_2\tau^3 + \dots)}$  and use a standard filter for a pure sinusoid.

In general, we can extend this approach to any signal  $x(t) = e^{j\phi(t)}$  by estimating its phase  $\phi(t)$  with  $\hat{\phi}(t)$  (using the instantaneous frequency estimation that will be discussed later) and filtering demodulated signal  $x(t)e^{-j\hat{\phi}(t)}$  by a low-pass filter. The resulting signal is obtained when the filtered signal is returned back to the original frequencies, by modulation with  $e^{j\hat{\phi}(t)}$ .

The filtering of signal can be modeled by the following expression:

$$x_f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} B_t(t, \Omega) \text{LPFT}(t, \Omega) d\Omega, \quad (3.35)$$

where  $\text{LPFT}(t, \Omega)$  is the LPFT of  $x(t)$ ,  $x_f(t)$  is the filtered signal,  $B_t(t, \Omega)$  is a support function used for filtering. It could be 1 within the time-frequency region where we assume that the signal of interest exists, and 0 elsewhere.

Note that the sufficient order of the LPFT can be obtained recursively. We start from the STFT and check whether its auto-term's width is equal to the width of the FT of the used window. If true, it means that a signal is a pure sinusoid and the STFT provides its best possible concentration. We should not calculate the LPFT. If the auto-term is wider, it means that there are signal non-stationarities within the window and the first-order LPFT should be calculated. The auto-term's width is again compared to the width of the window's FT and if they do not coincide we should increase the LPFT order.

In case of multi-component signals, the distribution will be optimized to the strongest component first. Then, the strongest component is filtered out and the procedure is repeated for the next component in the same manner, until the energy of the remaining signal is negligible, i.e., until all the components are processed.

### 3.03.2.6 Relation between the STFT and the continuous wavelet transform

The first form of functions having the basic property of wavelets was used by Haar at the beginning of the 20th century. At the beginning of 1980s, Morlet introduced a form of basis functions for analysis of seismic signals, naming them “wavelets.” Theory of wavelets was linked to the image processing by Mallat in the following years. In late 1980s Daubechies presented a whole new class of wavelets that, in addition to the orthogonality property, can be implemented in a simple way, by using digital filtering ideas. The most important applications of the wavelets are found in image processing and compression, pattern recognition and signal denoising. As such they will be a separate topic of this book. Here, we will only link continuous wavelet transform to the time-frequency analysis [5,27,28].

The STFT is characterized by constant time and frequency resolutions for both low and high frequencies. The basic idea behind the wavelet transform is to vary the resolutions with scale (being related to frequency), so that a high frequency resolution is obtained for low frequencies, whereas a high time resolution is obtained for high frequencies, which could be relevant for some practical applications. It is achieved by introducing a variable window width, such that it is decreased for higher frequencies. The basic idea of the wavelet transform and its comparison with the STFT is illustrated in Figure 3.10.

Time and frequency resolution is schematically illustrated in Figure 3.8.

When the above idea is translated into the mathematical form, one gets the definition of a continuous wavelet transform

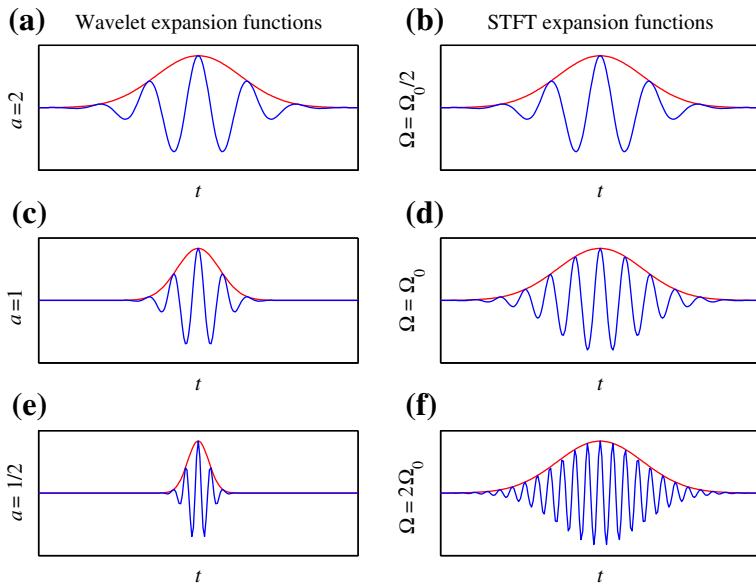
$$\text{WT}(t, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(\tau) h^* \left( \frac{\tau - t}{a} \right) d\tau, \quad (3.36)$$

where  $h(t)$  is a band-pass signal, and the parameter  $a$  is the scale. This transform produces a time-scale, rather than the time-frequency signal representation. For the Morlet wavelet (that will be used for illustrations in this short presentation) the relation between the scale and the frequency is  $a = \Omega_0 / \Omega$ . In order to establish a strong formal relationship between the WT and the STFT, we will choose the basic wavelet  $h(t)$  in the form

$$h(t) = w(t) e^{j\Omega_0 t}, \quad (3.37)$$

where  $w(t)$  is a window function and  $\Omega_0$  is a constant frequency. For example, for the Morlet wavelet we have a modulated Gaussian function

$$h(t) = \sqrt{\frac{1}{2\pi}} e^{-\alpha t^2} e^{j\Omega_0 t},$$

**FIGURE 3.10**

Expansion functions for the wavelet transform (left) and the short-time Fourier transform (right). Top row presents high scale (low frequency), middle row is for a medium scale (medium frequency) and bottom row is for a low scale (high frequency).

where the values of  $\alpha$  and  $\Omega_0$  are chosen such that the ratio of  $h(0)$  and the first maximum is  $1/2$ ,  $\Omega_0 = 2\pi\sqrt{\alpha}/\ln 2$ . From the definition of  $h(t)$  it is obvious that small  $\Omega$  (i.e., large  $a$ ) corresponds to a wide wavelet, i.e., a wide window, and vice versa.

Substitution of (3.37) into (3.36) leads to a continuous wavelet transform form suitable for a direct comparison with the STFT:

$$\text{WT}(t, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(\tau) w^* \left( \frac{\tau - t}{a} \right) e^{-j\Omega_0 \frac{\tau-t}{a}} d\tau. \quad (3.38)$$

From the filter theory point of view the wavelet transform, for a given scale  $a$ , could be considered as the output of system with impulse response  $h^*(-t/a)\sqrt{|a|}$ , i.e.,  $\text{WT}(t, a) = x(t) *_t h^*(-t/a)\sqrt{|a|}$ , where  $*_t$  denotes a convolution in time. Similarly the STFT, for a given  $\Omega$ , may be considered as  $\text{STFT}_{II}(t, \Omega) = x(t) *_t [w^*(-t)e^{j\Omega t}]$ . If we consider these two band-pass filters from the bandwidth point of view we can see that, in the case of STFT, the filtering is done by a system whose impulse response  $w^*(-t)e^{j\Omega t}$  has a constant bandwidth, being equal to the width of the FT of  $w(t)$ .

The *S*-transform (or the Stockwell transform) is conceptually a hybrid of short-time Fourier analysis and wavelet analysis. It employs a variable window length but preserves the phase information by using

the STFT form in the signal decomposition. As a result, the phase spectrum is absolute in the sense that it is always referred to a fixed time reference. The real and imaginary spectrum can be localized independently with resolution in time in terms of basis functions. The changes in the absolute phase of a certain frequency can be tracked along the time axis and useful information can be extracted. In contrast to wavelet transform, the phase information provided by the *S*-transform is referenced to the time origin, and therefore provides supplementary information about spectra which is not available from locally referenced phase information obtained by the continuous wavelet transform. The frequency dependent window function produces higher frequency resolution at lower frequencies, while at higher frequencies sharper time localization can be achieved.

#### *Constant Q-factor transform*

The quality factor  $Q$  for a band-pass filter, as measure of the filter selectivity, is defined as

$$Q = \frac{\text{Central Frequency}}{\text{Bandwidth}}.$$

In the STFT the bandwidth is constant, equal to the window FT width,  $B_w$ . Thus, factor  $Q$  is proportional to the considered frequency,

$$Q = \frac{\Omega}{B_w}.$$

In the case of the wavelet transform the bandwidth of impulse response is the width of the FT of  $w(t/a)$ . It is equal to  $B_0/a$ , where  $B_0$  is the constant bandwidth corresponding to the mother wavelet. It follows

$$Q = \frac{\Omega}{B_0/a} = \frac{\Omega_0}{B_0} = \text{const.}$$

Therefore, the continuous wavelet transform corresponds to the passing a signal through a series of band-pass filters centered at  $\Omega$ , with constant factor  $Q$ . Again we can conclude that the filtering, that produces WT, results in a small bandwidth (high frequency resolution and low time resolution) at low frequencies and wide bandwidth (low frequency and high time resolution) at high frequencies.

#### *Affine transforms*

A whole class of signal representations, including the quadratic ones, is defined with the aim to preserve the constant  $Q$  property. They belong to the area of the so called time-scale signal analysis or affine time-frequency representations [5, 14, 28–31]. The basic property of an affine time-frequency representation is that the representation of time shifted and scaled version of signal

$$y(t) = \frac{1}{\sqrt{\gamma}}x\left(\frac{t - t_0}{\gamma}\right),$$

whose FT is  $Y(\Omega) = \sqrt{\gamma}X(\gamma\Omega)e^{-j\Omega t_0}$ , results in a time-frequency representation

$$\text{TFR}_y(t, \Omega) = \text{TFR}_x\left(\frac{t - t_0}{\gamma}, \gamma\Omega\right).$$

The name affine comes from the affine transformation of time, that is, in general a transformation of the form  $t \rightarrow \alpha t + \beta$ . It is easy to verify that continuous wavelet transform satisfies this property.

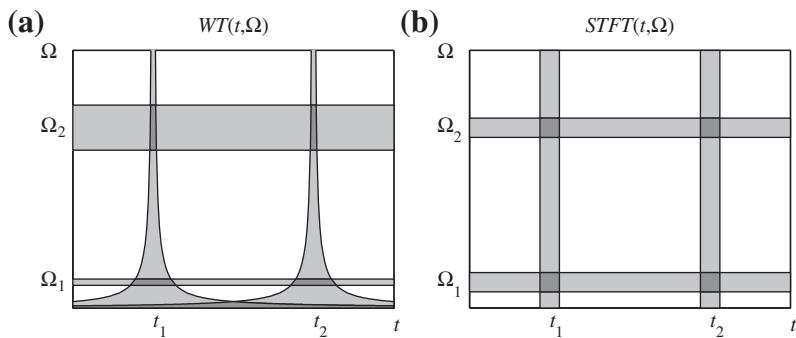
**FIGURE 3.11**

Illustration of the wavelet transform (a) of a sum of two delta pulses and two sinusoids compared with STFT in (b).

**Example 5.** Consider signal (3.6). Its continuous wavelet transform is

$$\begin{aligned} WT(t, a) = & \frac{1}{\sqrt{|a|}} \left[ w((t_1 - t)/a) e^{-j\Omega_0(t_1-t)/a} + w((t_2 - t)/a) e^{-j\Omega_0(t_2-t)/a} \right] \\ & + \sqrt{a} \left[ e^{j\Omega_1 t} W[a(\Omega_0/a - \Omega_1)] + e^{j\Omega_2 t} W[a(\Omega_0/a - \Omega_2)] \right]. \end{aligned} \quad (3.39)$$

The transform (3.39) has non-zero values in the region depicted in Figure 3.11a.

*Scalogram:*

In analogy with spectrogram, the scalogram is defined as the squared magnitude of a wavelet transform:

$$SCAL(t, a) = |WT(t, a)|^2. \quad (3.40)$$

The scalogram obviously loses the linearity property, and fits into the category of quadratic transforms.

*A Simple filter bank formulation*

Time-frequency grid for wavelet transform is presented in Figure 3.8. Within the filter bank framework it means that the original signal is processed in the following way. The signal's spectral content is divided into high frequency and low frequency part. An example, how to achieve this is presented in the STFT analysis by using a two samples rectangular window  $w(n) = \delta(n) + \delta(n + 1)$ , with  $N = 2$ . Then, its two samples WT is  $x_L(n, 0) = x(n) + x(n - 1)$ , for  $k = 0$ , corresponding to low frequency  $\omega = 0$  and  $x_H(n, 1) = x(n) - x(n - 1)$  for  $k = 1$  corresponding to high frequency  $\omega = \pi$ . The high frequency part,  $x_H(n, 1)$ , having high resolution in time, is not processed any more. It is kept with this high resolution in time, expecting that this kind of resolution will be needed for a signal. The low pass part  $x_L(n, 0) = x(n) + x(n - 1)$  is further processed, by dividing it into its low frequency part,

$$\begin{aligned} x_{LL}(n, 0, 0) &= x_L(n, 0) + x_L(n - 2, 0) \\ &= x(n) + x(n - 1) + x(n - 2) + x(n - 3) \end{aligned}$$

and its high frequency part

$$\begin{aligned} x_{LH}(n, 0, 1) &= x_L(n, 0) - x_L(n - 2, 0) \\ &= x(n) + x(n - 1) - [x(n - 2) + x(n - 3)]. \end{aligned}$$

The high pass of this part is left with resolution four in time, while the low pass part is further processed, by dividing it into its low and high frequency part, until the full length of signal is achieved, Figure 3.8b.

#### *Chirplet transform*

An extension of the wavelet transform, for time-frequency analysis, is the chirplet transform. By using linear frequency modulated forms instead of the constant frequency ones, the chirplet is formed. Here we will present a Gaussian chirplet atom that is a four parameter function

$$\varphi(\tau; [t, \Omega, \Omega_1, \sigma]) = \frac{1}{\sqrt{\sqrt{\pi}\sigma}} \exp\left(-\frac{1}{2} \left(\frac{\tau - t}{\sigma}\right)^2 + j(\Omega_1(\tau - t)^2 + j\Omega(\tau - t))\right),$$

where the parameter  $\sigma$  controls the width of the chirplet in time, parameter  $\Omega_1$  stands for the chirplet rate in time-frequency plane, while  $t$  and  $\Omega$  are the coordinates of the central time and frequency point in the time-frequency plane. In this way, for a given parameters  $[t, \Omega, \Omega_1, \sigma]$  we project signal onto a Gaussian chirp, centered at  $t, \Omega$  whose width is defined by  $\sigma$  and rate is  $\Omega_1$ :

$$c(t, \Omega, \Omega_1, \sigma) = \int_{-\infty}^{\infty} x(\tau)\varphi^*(\tau; [t, \Omega, \Omega_1, \sigma])d\tau.$$

In general, projection procedure should be performed for each point in time-frequency plane, for all possible parameter values. Interest in using a Gaussian chirplet atom stems from to the fact that it provides the highest joint time-frequency concentration. In practice, all four parameters should be discretized. The set of the parameter discretized atoms are called a dictionary. In contrast to the second order local polynomial FT, here the window width is parametrized and varies, as well. Since we have a multiparameter problem, computational requirements for this transform are very high.

In order to improve efficiency of the chirplet transform calculation, various adaptive forms of the chirplet transform were proposed. The matching pursuit procedure is a typical example. The first step of this procedure is to choose a chirplet atom from the dictionary yielding the largest amplitude of the inner product between the atom and the signal. Then the residual signal, obtained after extracting the first atom, is decomposed in the same way. Consequently, the signal is decomposed into a sum of chirplet atoms.

#### 3.03.2.7 Generalization

In general, any set of well localized functions in time and frequency can be used for the time-frequency analysis of a signal. Let us denote signal as  $x(\tau)$  and the set of such functions with  $\varphi(\tau; [\text{Parameters}])$ , then the projection of the signal  $x(\tau)$  onto such functions,

$$c([\text{Parameters}]) = \int_{-\infty}^{\infty} x(\tau)\varphi^*(\tau; [\text{Parameters}])d\tau$$

represents similarity between  $x(t)$  and  $\varphi(t; [\text{Parameters}])$ , at a given point with parameter values defined by  $[\text{Parameters}]$ .

We may have the following cases:

- Frequency as the only one parameter. Then, we have projection onto complex harmonics with changing frequency, and  $c([\Omega])$  is the FT of signal  $x(\tau)$  with

$$\varphi(\tau; [\Omega]) = e^{j\Omega\tau}.$$

- Time and frequency as parameters. Varying  $t$  and  $\Omega$  and calculating projections of signal  $x(\tau)$  we get the STFT. In this case we use  $w(\tau - t)$  as a localization function around parameter  $t$  and

$$\varphi(\tau; [t, \Omega]) = w(\tau - t)e^{j\Omega\tau}.$$

- Time and frequency as parameters with a frequency dependent localization in time, we get wavelet transform. It is more often expressed as function of scale parameter  $a = \Omega_0/\Omega$ , than the frequency  $\Omega$ . The S-transform belongs to this class. For the continuous wavelet transform with mother wavelet  $w(\tau)e^{j\Omega_0\tau}$  we have

$$\varphi(\tau; [t, a]) = w((\tau - t)/a)e^{j\Omega_0(\tau-t)/a}.$$

- Frequency and signal phase rate. We get the polynomial FT of the second order, with

$$\varphi(\tau; [\Omega, \Omega_1]) = e^{j(\Omega\tau + \Omega_1\tau^2)}.$$

- Time, frequency, and signal phase rate as parameters results in a form of the local polynomial Fourier transform with

$$\varphi(\tau; [t, \Omega, \Omega_1]) = w(\tau - t)e^{j(\Omega\tau + \Omega_1\tau^2)}.$$

- Time, frequency, and signal phase rate as parameters, with a varying time localization, as parameters results in the chirplets with localization function with

$$\varphi(\tau; [t, \Omega, \Omega_1, \sigma]) = w((\tau - t)/\sigma) \exp(j(\Omega_1(\tau - t)^2 + j\Omega(\tau - t))).$$

- Frequency, signal phase rate, and other higher order coefficients, we get the polynomial FT of the  $N$ th order, with

$$\varphi(\tau; [\Omega, \Omega_1, \Omega_2, \dots, \Omega_N]) = e^{j(\Omega\tau + \Omega_1\tau^2 + \Omega_2\tau^3 + \dots + \Omega_N\tau^N)}.$$

- Time, frequency, signal phase rate, and other higher order coefficients, we get the local polynomial FT of the  $N$ th order, with

$$\varphi(\tau; [t, \Omega, \Omega_1, \Omega_2, \dots, \Omega_N]) = w(\tau - t)e^{j(\Omega\tau + \Omega_1\tau^2 + \Omega_2\tau^3 + \dots + \Omega_N\tau^N)}.$$

- Time, frequency, signal phase rate, and other higher order coefficients, with a variable window width we would get the  $N$ th order-lets, with

$$\varphi(\tau; [t, \Omega, \Omega_1, \Omega_2, \dots, \Omega_N, \sigma]) = w((\tau - t)/\sigma)e^{j(\Omega\tau + \Omega_1\tau^2 + \Omega_2\tau^3 + \dots + \Omega_N\tau^N)}.$$

- Time, frequency, and any other parametrized phase function form, like sinusoidal ones, with constant or variable window width...

### 3.03.3 Quadratic time-frequency distributions

In order to provide additional insight into the field of joint time-frequency analysis, as well as to improve concentration of time-frequency representation, energy distributions of signals were introduced. We have already mentioned the spectrogram which belongs to this class of representations and is a straightforward extension of the STFT. Here, we will discuss other distributions and their generalizations.

The basic condition for the definition of time-frequency energy distributions is that a two-dimensional function of time and frequency  $P(t, \Omega)$  represents the energy density of a signal in the time-frequency plane. Thus, the signal energy associated with the small time and frequency intervals  $\Delta t$  and  $\Delta\Omega$ , respectively, would be

$$\text{Signal energy within } [\Omega + \Delta\Omega, t + \Delta t] = P(t, \Omega)\Delta\Omega\Delta t.$$

However, point by point definition of time-frequency energy densities in the time-frequency plane is not possible, since the uncertainty principle prevents us from defining concept of energy at a specific instant and frequency. This is the reason why some more general conditions are being considered to derive time-frequency distributions of a signal. Namely, one requires that the integral of  $P(t, \Omega)$  over  $\Omega$ , for a particular instant of time should be equal to the instantaneous power of the signal  $|x(t)|^2$ , while the integral over time for a particular frequency should be equal to the spectral energy density  $|X(\Omega)|^2$ . These conditions are known as marginal conditions or marginal properties of time-frequency distributions.

Therefore, it is desirable that an energetic time-frequency distribution of a signal  $x(t)$  satisfies:

- Energy property

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(t, \Omega) d\Omega dt = E_x, \quad (3.41)$$

- Time marginal properties

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} P(t, \Omega) d\Omega = |x(t)|^2, \text{ and} \quad (3.42)$$

- Frequency marginal property

$$\int_{-\infty}^{\infty} P(t, \Omega) dt = |X(\Omega)|^2, \quad (3.43)$$

where  $E_x$  denotes the energy of  $x(t)$ . It is obvious that if either one of marginal properties (3.42), (3.43) is fulfilled, so is the energy property. Note that relations (3.41)–(3.43), do not reveal any information about the local distribution of energy at a point  $(t, \Omega)$ . The marginal properties are illustrated in Figure 3.12.

Next we will introduce some distributions satisfying these properties.

#### 3.03.3.1 Rihaczek distribution

The Rihaczek distribution satisfies the marginal properties (3.41)–(3.43). This distribution is of limited practical importance (some recent contributions show that it could be interesting in the phase synchrony and stochastic signal analysis). We will present one of its derivations with a simple electrical engineering foundation.

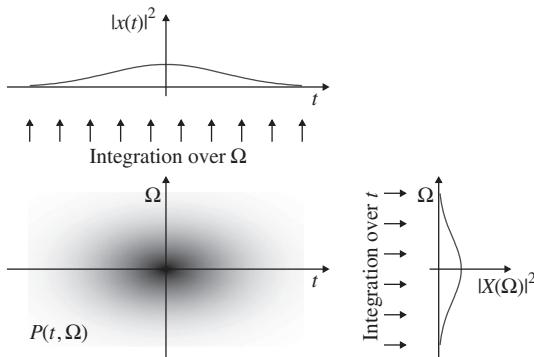
**FIGURE 3.12**

Illustration of the marginal properties as distribution projections.

Consider a simple electrical circuit analysis. Assume that a voltage  $v(t)$  is applied at the resistor whose resistance is  $R = 1[\Omega]$ , but only within a very narrow frequency band  $[\Omega, \Omega + \Delta\Omega]$

$$R(\theta) = \begin{cases} 1 & \text{for } \Omega \leq \theta < \Omega + \Delta\Omega, \\ \infty & \text{elsewhere.} \end{cases}$$

In that case, the energy dissipated at the resistor within a short time interval  $(t, t + \Delta t)$  is defined as:

$$E(t, \Omega) = \int_t^{t+\Delta t} v(\tau) i^*(\tau) d\tau, \quad (3.44)$$

where  $i(t)$  denotes the resulting current. It may be expressed in terms of the FT of the voltage:

$$i(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} I(\theta) e^{j\theta\tau} d\theta = \frac{1}{2\pi} \int_{\Omega}^{\Omega+\Delta\Omega} V(\theta) e^{j\theta\tau} d\theta, \quad (3.45)$$

where capital letters represent corresponding FTs of the current and voltage. Substitution of (3.45) into (3.44) produces

$$E(t, \Omega) = \frac{1}{2\pi} \int_t^{t+\Delta t} \int_{\Omega}^{\Omega+\Delta\Omega} v(\tau) V^*(\theta) e^{-j\theta\tau} d\theta d\tau. \quad (3.46)$$

Based on the above considerations, one may define a time-frequency energy distribution:

$$P(t, \Omega) = 2\pi \lim_{\substack{\Delta\Omega \rightarrow 0 \\ \Delta t \rightarrow 0}} \frac{E(t, \Omega)}{\Delta\Omega \Delta t} = v(t) V^*(\Omega) e^{-j\Omega t}. \quad (3.47)$$

The previous analysis may be generalized for an arbitrary signal  $x(t)$  with the associated FT  $X(\Omega)$ . The Rihaczek distribution is obtained in the following form:

$$\begin{aligned} RD(t, \Omega) &= x(t) X^*(\Omega) e^{-j\Omega t} \\ &= \int_{-\infty}^{\infty} x(t) x^*(t - \tau) e^{-j\Omega\tau} d\tau = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega + \theta) X^*(\Omega) e^{j\theta t} d\theta. \end{aligned} \quad (3.48)$$

It seems that the Rihaczek distribution is an ideal one, we have been looking for. However, energy is calculated over the intervals  $(t, t + \Delta t)$  and  $(\Omega, \Omega + \Delta\Omega)$ , while  $V(\Omega)$  was calculated over the entire interval  $(-\infty, \infty)$ . This introduces the influence of other time periods onto the interval  $(t, t + \Delta t)$ . Therefore, it is not as local as it may seem from the derivation. This distribution exhibits significant drawbacks for possible time-frequency analysis, as well. The most important one is that it is complex valued, despite the fact that it has been derived with the aim to represent signal energy density. In addition, its time-frequency concentration for non-stationary signals is quite low.

### 3.03.3.2 Wigner distribution

The other quadratic distributions cannot be easily derived as the Rihaczek distribution. Partially this is due to the lack of adequate simple physical interpretations. In order to derive some other quadratic time-frequency distributions, observe that the Rihaczek distribution may be interpreted as the FT (over  $\tau$ ) of the function

$$R(t, \tau) = x(t)x^*(t - \tau),$$

that will be referred to as the local autocorrelation function,

$$\text{RD}(t, \Omega) = \int_{-\infty}^{\infty} R(t, \tau)e^{-j\Omega\tau} d\tau. \quad (3.49)$$

This relation is in accordance with spectral density function for random signals. A general form of the local autocorrelation function may be written as

$$R(t, \tau) = x(t + (\alpha + 1/2)\tau)x^*(t + (\alpha - 1/2)\tau), \quad (3.50)$$

where  $\alpha$  is an arbitrary constant ( $\alpha = -1/2$  produces the RD; note that also  $\alpha = 1/2$  could be used as a variant of the RD). For  $\alpha = 0$ , the local autocorrelation function  $R_t(\tau)$  is Hermitian, i.e.,

$$R(t, \tau) = R^*(t, -\tau), \quad (3.51)$$

and its FT is real valued. The distribution that satisfies this property is called the Wigner distribution (or the Wigner-Ville distribution). It is defined as

$$\text{WD}(t, \Omega) = \int_{-\infty}^{\infty} x(t + \tau/2)x^*(t - \tau/2)e^{-j\Omega\tau} d\tau. \quad (3.52)$$

The Wigner distribution is originally introduced in quantum mechanics.

Expressing  $x(t)$  in terms of  $X(\Omega)$  and substituting it into (3.52) we get

$$\text{WD}(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega + \theta/2)X^*(\Omega - \theta/2)e^{j\theta t} d\theta \quad (3.53)$$

what represents a definition of the Wigner distribution in the frequency domain.

A distribution defined as the FT of (3.50) is called the Generalized Wigner Distribution (GWD). The name stems from the fact that this distribution is based on the Wigner distribution (for  $\alpha = 0$ ), which is the most important member of this class of distributions.

It is easy to show that the Wigner distribution and all the other distributions from the GWD class satisfy the marginal properties. From the Wigner distribution definition, it follows

$$x(t + \tau/2)x^*(t - \tau/2) = \text{IFT}\{\text{WD}(t, \Omega)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}(t, \Omega)e^{j\Omega\tau} d\Omega \quad (3.54)$$

which, for  $\tau = 0$ , produces (3.42)

$$|x(t)|^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}(t, \Omega)d\Omega. \quad (3.55)$$

Based on the definition of the Wigner distribution in the frequency domain, (3.53), one may easily prove the fulfillment of the frequency marginal. The marginal properties are satisfied for the whole class of GWD.

**Example 6.** The Wigner distribution of signals  $x_1(t) = \delta(t - t_1)$  and  $x_2(t) = e^{j\Omega_1 t}$  is given by

$$\text{WD}_1(\Omega, t) = \delta(t - t_1)$$

and

$$\text{WD}_2(\Omega, t) = 2\pi\delta(\Omega - \Omega_1),$$

respectively. The distribution concentration is very high, in both cases. Note that this fact does not mean that, for one signal component, we will be able to achieve an arbitrary high concentration simultaneously in both time and in frequency.

**Example 7.** Let us now assume a linear frequency modulated signal,  $x(t) = Ae^{jat^2/2}$ . In this case we have

$$x(t + \tau/2)x^*(t - \tau/2) = A^2e^{jtt\tau a}$$

with

$$\text{WD}(\Omega, t) = 2\pi A^2\delta(\Omega - at).$$

Again, a high concentration in the time-frequency plane is achieved.

These two examples demonstrated that the Wigner distribution can provide superior time-frequency representations in comparison to the STFT.

### 3.03.3.2.1 Distribution concentrated at the instantaneous frequency

For a general mono-component signal of the form  $x(t) = A(t) \exp(j\phi(t))$ , with slow varying amplitude comparing to the signal phase variations  $|A(t)|' \ll |\phi'(t)|$ , an ideal time-frequency (ITF) representation (fully concentrated along the instantaneous frequency) can be defined as:

$$\text{ITF}(t, \Omega) = 2\pi |A(t)|^2 \delta(\Omega - \phi'(t)). \quad (3.56)$$

Note that the ideal TFD defined by (3.56) satisfies the marginal properties for a wide class of frequency modulated signals. The time marginal is satisfied since:

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{ITF}(t, \Omega)d\Omega = |A(t)|^2,$$

where a monotonous function  $\phi'(t)$  is assumed, with  $t_0$  being the solution of  $\Omega - \phi'(t_0) = 0$ . Since the time marginal condition is satisfied, so is the energy condition. For signals satisfying the stationary phase method, the frequency marginal is satisfied, as well. From a similar analysis in the frequency domain one may define a distribution fully concentrated along the group delay, as well.

For a frequency modulated signal  $x(t) = A \exp(j\phi(t))$  the Wigner distribution (3.52) assumes the form:

$$\begin{aligned} \text{WD}(t, \Omega) &= A^2 \int_{-\infty}^{\infty} e^{j[\phi(t+\tau/2)-\phi(t-\tau/2)]} e^{-j\Omega\tau} d\tau \\ &= A^2 \int_{-\infty}^{\infty} e^{j[\phi(t+\tau/2)-\phi(t-\tau/2)]-j\phi'(t)\tau} e^{j\phi'(t)\tau} e^{-j\Omega\tau} d\tau. \end{aligned}$$

Factor  $A^2 \int_{-\infty}^{\infty} e^{j\phi'(t)\tau} e^{-j\Omega\tau} d\tau$  produces the ideal distribution concentration  $2\pi A^2 \delta(\Omega - \phi'(t))$ , while the term

$$Q = \left[ \phi\left(t + \frac{\tau}{2}\right) - \phi\left(t - \frac{\tau}{2}\right) \right] - \phi'(t)\tau = \frac{1}{24}\phi^{(3)}(t)\tau^3 + \dots \quad (3.57)$$

causes distribution spread around the instantaneous frequency. Factor  $Q$  will be referred to as the spread factor. It is equal to zero if instantaneous frequency  $\phi'(t)$  is a linear function, i.e., if  $\phi^{(n)}(t) \equiv 0$ , for  $n \geq 3$ .

**Example 8.** Let us consider signal of the form

$$x(t) = e^{-\frac{1}{2}(t/\alpha)^2}.$$

The Wigner distribution of  $x(t)$  is FT of

$$\begin{aligned} x\left(t + \frac{\tau}{2}\right)x^*\left(t - \frac{\tau}{2}\right) &= e^{-(t/\alpha)^2}e^{-\frac{1}{4}(\tau/\alpha)^2}, \\ \text{WD}(t, \Omega) &= 2\sqrt{\pi}e^{-(t/\alpha)^2}e^{-(\alpha\Omega)^2}. \end{aligned}$$

Note that the duration in time is proportional to  $\alpha$  while the duration in frequency is proportional to  $1/\alpha$ . Product of these durations is constant.

### 3.03.3.2.2 Signal reconstruction

The signal can be reconstructed from the Wigner distribution, Eq. (3.54) with  $\tau/2 = t$ , as:

$$x(t) = \frac{1}{2\pi x^*(0)} \int_{-\infty}^{\infty} \text{WD}(t/2, \Omega) e^{j\Omega t} d\Omega.$$

Due to the term  $x^*(0)$  ambiguity in the signal phase remains.

Since the Wigner distribution is a two-dimensional representation of a one-dimensional signal, obviously an arbitrary real valued two-dimensional function will not be a valid Wigner distribution. A two-dimensional real function  $P(t, \Omega)$  is the Wigner distribution of a signal if:

$$\frac{\partial^2 \ln[\rho(t_1, t_2)]}{\partial t_1 \partial t_2} = 0, \quad (3.58)$$

where

$$\rho(t_1, t_2) = \frac{1}{2\pi} \int_{-\infty}^{\infty} P\left(\frac{t_1 + t_2}{2}, \Omega\right) e^{j\Omega(t_1 - t_2)} d\Omega.$$

The solution of partial differential equation (3.58) is equal to  $\ln[\rho(t_1, t_2)] = \varphi_1(t_1) + \varphi_2(t_2)$ , where  $\varphi_1(t_1)$  and  $\varphi_2(t_2)$  are arbitrary functions of  $t_1$  and  $t_2$ . Therefore,  $\rho(t_1, t_2) = e^{\varphi_1(t_1)} e^{\varphi_2(t_2)} = f_1(t_1) f_2(t_2)$ .

With  $t_1 = t + \frac{\tau}{2}$  and  $t_2 = t - \frac{\tau}{2}$ , we get:

$$f_1\left(t + \frac{\tau}{2}\right) f_2\left(t - \frac{\tau}{2}\right) = \frac{1}{2\pi} \int_{-\infty}^{\infty} P(t, \Omega) e^{j\Omega\tau} d\Omega.$$

Since  $P(t, \Omega)$  is a real function, it follows that  $f_1(t) = f_2^*(t) = x(t)$ . Thus, for  $P(t, \Omega)$  satisfying (3.58), there exists function  $x(t)$  such that  $x(t + \frac{\tau}{2})x^*(t - \frac{\tau}{2})$  and  $P(t, \Omega)$  are the FT pair. A mean squared approximation of an arbitrary two-dimensional function by a valid Wigner distribution, or a sum of the Wigner distributions, will be discussed later.

### 3.03.3.2.3 Uncertainty principle and the Wigner distribution

*The uncertainty principle for the Wigner distribution* states that the product of effective durations of a signal  $x(t)$  in time  $\sigma_t$  and in frequency  $\sigma_\Omega$  cannot be arbitrary small. It satisfies the inequality:

$$\sigma_t^2 \sigma_\Omega^2 \geq 1/4, \quad (3.59)$$

where  $\sigma_t^2$  and  $\sigma_\Omega^2$  are defined by

$$\begin{aligned} \sigma_t^2 &= \frac{1}{2\pi E_x} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (t - t_c)^2 \text{WD}(t, \Omega) dt d\Omega = \frac{1}{E_x} \int_{-\infty}^{\infty} (t - t_c)^2 |x(t)|^2 dt, \\ \sigma_\Omega^2 &= \frac{1}{2\pi E_x} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (\Omega - \Omega_c)^2 \text{WD}(t, \Omega) dt d\Omega \\ &= \frac{1}{2\pi E_x} \int_{-\infty}^{\infty} (\Omega - \Omega_c)^2 |X(\Omega)|^2 d\Omega, \end{aligned} \quad (3.60)$$

where  $E_x$  is signal energy and

$$\begin{aligned} t_c &= \frac{1}{E_x} \int_{-\infty}^{\infty} t |x(t)|^2 dt, \\ \Omega_c &= \frac{1}{2\pi E_x} \int_{-\infty}^{\infty} \Omega |X(\Omega)|^2 d\Omega. \end{aligned}$$

The equality in (3.59) holds for the Gaussian signal  $x(t)$ , when we get  $\sigma_t^2 \sigma_\Omega^2 = 1/4$ . Thus, it is not possible to achieve arbitrary high resolution in both directions, simultaneously. The product of effective durations is higher than 1/4 for any other than the Gaussian signal.

The fact that the signal  $x(t)$  is located within  $[t_g - \sigma_t, t_g + \sigma_t]$  in time and within  $[\Omega_i - \sigma_\Omega, \Omega_i + \sigma_\Omega]$  in frequency does not provide any information about the local concentration of the signal within this time-frequency region. It can be spread all over the region or highly concentrated along a line within that

region. Thus, the conclusion that the Wigner distribution is highly concentrated along a line (that we made earlier for a linear FM signal), does not contradict the uncertainty principle. Local concentration measures are used to grade signal's concentration in the time-frequency domain.

The duration of signal  $x(t) = A(t) \exp(j\phi(t))$  in frequency domain is

$$\begin{aligned}\sigma_{\Omega}^2 &= \frac{1}{2\pi E_x} \int_{-\infty}^{\infty} \Omega^2 |X(\Omega)|^2 d\Omega = \frac{1}{E_x} \int_{-\infty}^{\infty} |x'(t)|^2 dt \\ &= \frac{1}{E_x} \int_{-\infty}^{\infty} \left( (A'(t))^2 + (A(t)\phi'(t))^2 \right) dt,\end{aligned}$$

where, without loss of generality,  $\Omega_i = 0$  is assumed. Note that the product  $\sigma_t^2 \sigma_{\Omega}^2$  has a lower limit  $1/4$ , but there is no upper limit. It can be very large. Signals whose product of durations in time and frequency is large,  $\sigma_t^2 \sigma_{\Omega}^2 \gg 1$ , are called asymptotic signals.

#### 3.03.3.2.4 Pseudo quantum signal representation

A distribution that parametrize the uncertainty, keeping the marginal properties and the location of the instantaneous frequency, is defined as a “pseudo quantum” signal representation:

$$\text{SD}_L(t, \wp) = \int_{-\infty}^{\infty} x^{[L]} \left( t + \frac{\tau}{2L} \right) x^{[L]*} \left( t - \frac{\tau}{2L} \right) e^{-j\wp\tau} d\tau \quad (3.61)$$

with

$$x^{[L]}(t) = A(t) e^{jL\phi(t)}.$$

The spreading factor in this representation is

$$Q(t, \tau) = \left[ L\phi \left( t + \frac{\tau}{2L} \right) - L\phi \left( t - \frac{\tau}{2L} \right) \right] - \phi'(t)\tau = \frac{1}{24L^2} \phi^{(3)}(t) \tau^3 + \dots$$

For the Gaussian chirp signal

$$x(t) = A e^{-at^2/2} e^{j(bt^2/2+ct)} \quad (3.62)$$

we get

$$\text{SD}_L(t, \wp) = 2|A|^2 e^{-at^2} \sqrt{\frac{\pi}{a}} L e^{-\frac{(\wp-bt-c)^2}{a/L^2}}.$$

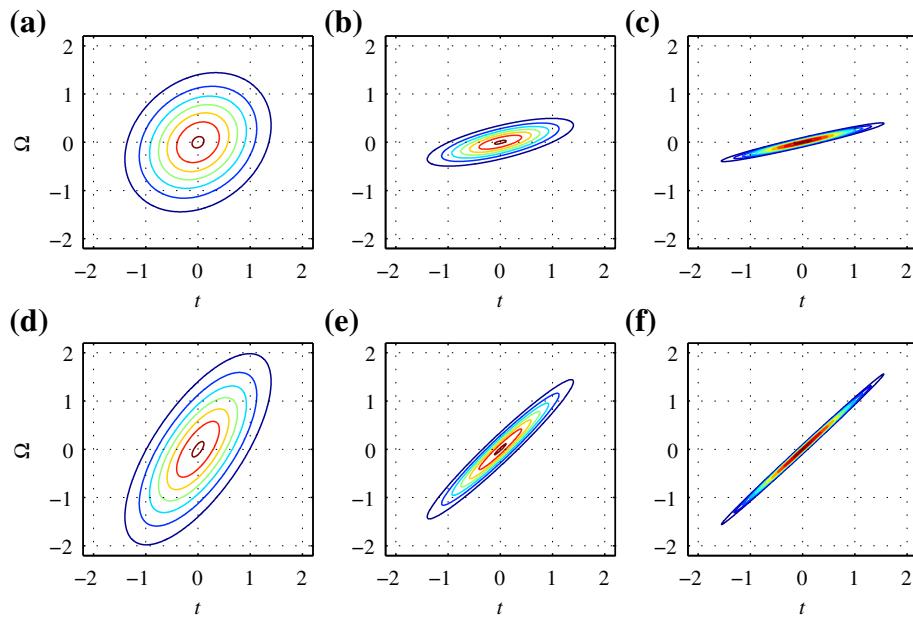
For a large parameter  $L$ , when  $L\sqrt{\frac{\pi}{a}} e^{-L^2\wp^2/a} \rightarrow \pi\delta(\wp)$ , we get

$$\text{SD}_L(t, \wp) = |A|^2 e^{-at^2} 2\pi\delta(\wp - bt - c),$$

being highly concentrated, simultaneously in time and in frequency  $\wp$  at  $(0, c)$  for a large  $a$ , if  $a/L^2 \rightarrow 0$ . The uncertainty principle for (3.61) is

$$\sigma_t^2 \sigma_{\wp}^2 \geq \frac{1}{4L^2}. \quad (3.63)$$

Note that the distribution  $|A|^2 e^{-at^2} 2\pi\delta(\wp - bt - c)$  satisfies the energy and the time marginal property for any set of parameters.

**FIGURE 3.13**

The pseudo quantum signal representation of a Gaussian chirp with a low (a–c) and a high (d–f) frequency rate for (a, d)  $L = 1$  (the Wigner distribution), (b, e)  $L = 4$ , and (c, f)  $L = 16$ .

The pseudo quantum signal representation of signal (3.62) with  $a = 1$ ,  $b = 1/2$ , and  $c = 0$  for  $L = 1$  (Wigner distribution),  $L = 4$  and  $L = 16$  is given in Figure 3.13. The pseudo quantum distribution may be illustrated through a physical experiment with a pendulum, for example, by changing the total acceleration of the pendulum system, as described in [6].

### 3.03.3.2.5 Instantaneous bandwidth

From the definition of the signal width in frequency (3.60) we can conclude that, at a given instant  $t$ , we may define similar local values

$$\sigma_{\Omega}^2(t) = \frac{1}{2\pi|x(t)|^2} \int_{-\infty}^{\infty} (\Omega - \Omega_i(t))^2 \text{WD}(t, \Omega) d\Omega \quad (3.64)$$

with

$$\Omega_i(t) = \frac{1}{2\pi|x(t)|^2} \int_{-\infty}^{\infty} \Omega \text{WD}(t, \Omega) d\Omega$$

playing a role of the instantaneous bandwidth  $\sigma_{\Omega}^2(t)$  and the mean frequency  $\Omega_i(t)$ . It is easy to show that, for a signal  $x(t) = A(t)e^{j\phi(t)}$ , the mean frequency is equal to the instantaneous frequency,

$$\Omega_i(t) = \phi'(t).$$

This relation is used for the instantaneous frequency estimation, in addition to the simple detection of maximum position, for a given  $t$ .

For the instantaneous bandwidth we easily get:

$$\sigma_{\Omega}^2(t) = \frac{1}{2} \left[ \left( \frac{A'(t)}{A(t)} \right)^2 - \frac{A''(t)}{A(t)} \right].$$

Note that both of these forms follow as special cases of conditional instantaneous moments. The  $n$ th conditional moment of the Wigner distribution, at an instant  $t$ , is defined as

$$m_i^n(t) = \frac{1}{2\pi|x(t)|^2} \int_{-\infty}^{\infty} \Omega^n \text{WD}(t, \Omega) d\Omega.$$

Using the fact that the Wigner distribution and the local autocorrelation function are the FT pair,  $\text{WD}(t, \Omega) \xleftrightarrow[\Omega, \tau]{} x(t + \tau/2)x^*(t - \tau/2)$ , resulting in

$$(j\Omega)^n \text{WD}(t, \Omega) \xleftrightarrow[\Omega, \tau]{} \frac{d^n}{d\tau^n} (x(t + \tau/2)x^*(t - \tau/2)),$$

the moments are calculated as

$$\begin{aligned} m_i^n(t) &= \frac{(-j)^n \frac{d^n}{d\tau^n} (x(t + \tau/2)x^*(t - \tau/2)) \Big|_{\tau=0}}{|x(t)|^2}, \\ &= \frac{(-j/2)^n \sum_{l=0}^n \binom{n}{l} (-1)^l x^{*(l)}(t) x^{(n-l)}(t)}{|x(t)|^2}. \end{aligned}$$

In a similar way we can define moments for other distributions from the generalized Wigner distribution form, including the Rihaczek distribution.

It is important to note that the instantaneous bandwidth is not a measure of the distribution spread around the instantaneous frequency, in contrast to the global parameters  $\sigma_t^2$  and  $\sigma_{\Omega}^2$ , which indicate a global region of the distribution spread. It is obtained with the Wigner distribution as a weighting function, that can assume negative values. It may result in small values of  $\sigma_{\Omega}^2(t)$  even in the cases when the Wigner distribution is quite spread.

**Example 9.** Consider the instantaneous bandwidth of linear modulated Gaussian function

$$x(t) = e^{-\frac{1}{2}(t/\alpha)^2} \exp(jat^2).$$

Its Wigner distribution is

$$\text{WD}(t, \Omega) = 2\alpha\sqrt{\pi} e^{-(t/\alpha)^2} e^{-(\alpha(\Omega - 2at))^2}$$

For a large value of  $\alpha$ , as compared to  $a$  (slow-varying amplitude with respect to phase variations), we get a very concentrated distribution along the IF  $\Omega_i(t) = 2at$ . It is in a full agreement with the instantaneous bandwidth definition which produces, for example for  $t = 0$ , value of  $\sigma_{\Omega}^2(0) = 1/(2\alpha)$ , what is very small for large  $\alpha$ . However, when the Wigner distribution assumes negative values, like in the cubic phase signal that will be presented later (see example in Section 3.03.3.2.9), we have to be very careful with the instantaneous bandwidth interpretation.

### 3.03.3.2.6 Properties of the Wigner distribution

A list of the properties satisfied by the Wigner distribution follows:

P<sub>1</sub>—Realness for any signal

$$\text{WD}_x^*(t, \Omega) = \text{WD}_x(t, \Omega).$$

P<sub>2</sub>—Time-shift property

$$\text{WD}_y(t, \Omega) = \text{WD}_x(t - t_0, \Omega)$$

for

$$y(t) = x(t - t_0).$$

P<sub>3</sub>—Frequency shift property

$$\text{WD}_y(t, \Omega) = \text{WD}_x(t, \Omega - \Omega_0)$$

for

$$y(t) = x(t)e^{j\Omega_0 t}.$$

P<sub>4</sub>—Time marginal property

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}_x(t, \Omega) d\Omega = |x(t)|^2.$$

P<sub>5</sub>—Frequency marginal property

$$\int_{-\infty}^{\infty} \text{WD}_x(t, \Omega) dt = |X(\Omega)|^2.$$

P<sub>6</sub>—Time moments

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} t^n \text{WD}_x(t, \Omega) dt d\Omega = \int_{-\infty}^{\infty} t^n |x(t)|^2 dt.$$

P<sub>7</sub>—Frequency moments

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Omega^n \text{WD}_x(t, \Omega) dt d\Omega = \int_{-\infty}^{\infty} \Omega^n |X(\Omega)|^2 d\Omega.$$

P<sub>8</sub>—Scaling

$$\text{WD}_y(t, \Omega) = \text{WD}_x(at, \Omega/a)$$

for

$$y(t) = \sqrt{|a|}x(at), \quad a \neq 0.$$

P<sub>9</sub>—Instantaneous frequency

$$\frac{\int_{-\infty}^{\infty} \Omega \text{WD}_x(t, \Omega) d\Omega}{\int_{-\infty}^{\infty} \text{WD}_x(t, \Omega) d\Omega} = \Omega_i(t) = \frac{d}{dt} \arg[x(t)].$$

P<sub>10</sub>—Group delay

$$\frac{\int_{-\infty}^{\infty} t \text{WD}_x(t, \Omega) dt}{\int_{-\infty}^{\infty} \text{WD}_x(t, \Omega) dt} = -t_g(\Omega) = -\frac{d}{d\Omega} \arg[X(\Omega)].$$

P<sub>11</sub>—Time constraint

If  $x(t) = 0$  for  $t$  outside  $[t_1, t_2]$  then, also  $\text{WD}_x(t, \Omega) = 0$  for  $t$  outside  $[t_1, t_2]$ .

P<sub>12</sub>—Frequency constraint

If  $X(\Omega) = 0$  for  $\Omega$  outside  $[\Omega_1, \Omega_2]$ , then, also  $\text{WD}_x(t, \Omega) = 0$  for  $\Omega$  outside  $[\Omega_1, \Omega_2]$ .

P<sub>13</sub>—Convolution

$$\text{WD}_y(t, \Omega) = \int_{-\infty}^{\infty} \text{WD}_h(t - t_0, \Omega) \text{WD}_x(t_0, \Omega) dt_0$$

if

$$y(t) = \int_{-\infty}^{\infty} h(t - t_0) x(t_0) dt_0.$$

P<sub>14</sub>—Product

$$\text{WD}_y(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}_h(t, \Omega - \Omega_0) \text{WD}_x(t, \Omega_0) d\Omega_0$$

for

$$y(t) = h(t)x(t).$$

P<sub>15</sub>—Fourier transform

$$\text{WD}_y(t, \Omega) = \text{WD}_x(-\Omega/c, ct)$$

for

$$y(t) = \frac{1}{\sqrt{(2\pi)}} \sqrt{|c|} X(ct), \quad c \neq 0.$$

P<sub>16</sub>—Chirp convolution

$$\text{WD}_y(t, \Omega) = \text{WD}_x\left(t - \frac{\Omega}{c}, \Omega\right)$$

for

$$y(t) = x(t) * \sqrt{|c|} e^{jct^2/2}.$$

P<sub>17</sub>—Chirp product

$$\text{WD}_y(t, \Omega) = \text{WD}_x(t, \Omega - ct)$$

for

$$y(t) = x(t) e^{jct^2/2}.$$

P<sub>18</sub>—Moyal property

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{WD}_x(t, \Omega) \text{WD}_y(t, \Omega) dt d\Omega = \left| \int_{-\infty}^{\infty} x(t) y(t) dt \right|^2.$$

Verifying of these properties is straightforward and it is left to the reader.

### 3.03.3.2.7 Linear coordinate transforms of the Wigner distribution

Here we derive a general form of the linear coordinate transformation of the Wigner distribution, with the coordinate rotation as a special case. From the property P<sub>17</sub> it is easy to conclude that multiplication of signal by a chirp,  $x(t)e^{jat^2/2}$ , leads to  $\text{WD}_x(t, \Omega - at)$ . Similarly, for the convolution with a linear FM signal,  $x(t) * \sqrt{|b|}e^{jbt^2/2}$ , the transformation, according to P<sub>16</sub>, is  $\text{WD}_y(t, \Omega) = \text{WD}_x(t - \Omega/b, \Omega)$ . Now, we can easily conclude that for the signal

$$x_L(t) = \left\{ \left[ x(t)e^{jct^2/2} \right] * \sqrt{|b|}e^{jbt^2/2} \right\} e^{jat^2/2}, \quad (3.65)$$

the coordinate transformation matrix is

$$\begin{aligned} \mathbf{L} &= \begin{bmatrix} 1 & 0 \\ -c & 1 \end{bmatrix} \begin{bmatrix} 1 & -1/b \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -a & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 + a/b & -1/b \\ -c - a(c/b + 1) & c/b + 1 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \end{aligned} \quad (3.66)$$

with  $b \neq 0$ . Thus, we get the signal given by (3.65) results in the linear coordinate transformation of the Wigner distribution:

$$\text{WD}_L(t, \Omega) = \text{WD}_x(At + B\Omega, Ct + D\Omega), \quad (3.67)$$

where  $\text{WD}_x(t, \Omega)$  is the Wigner distribution of  $x(t)$ ,  $\text{WD}_L(t, \Omega)$  is the Wigner distribution of  $x_L(t)$ , defined by (3.65), and transformation matrix  $\mathbf{L}$  has the form, with values of  $A$ ,  $B$ ,  $C$ , and  $D$ , defined by (3.66).

#### *Rotation of the time-frequency plane*

We may easily conclude that the fractional Fourier transform (FRFT) directly follows as a special case of linear coordinate transformation, with transformation matrix:

$$\mathbf{L} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \quad (3.68)$$

which corresponds to the coordinate rotation of the time-frequency plane. By comparing (3.66) and (3.68) we easily get  $b = 1/\sin(\alpha) = \csc(\alpha)$  and  $a = c = -\tan(\alpha/2)$ . Substituting these values into (3.65) we get:

$$\begin{aligned} x_L(t) &= \left\{ \left[ x(t)e^{-j \tan(\alpha/2)t^2/2} \right] * \sqrt{\csc(\alpha)}e^{j \csc(\alpha)t^2/2} \right\} e^{-j \tan(\alpha/2)t^2/2} \\ &= \sqrt{\csc(\alpha)}e^{-j \tan(\alpha/2)t^2/2} \int_{-\infty}^{\infty} x(\tau)e^{-j \tan(\alpha/2)\tau^2/2} e^{j \csc(\alpha)(t-\tau)^2/2} d\tau \\ &= \sqrt{2\pi j e^{-j\alpha}} \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j \cot(\alpha)t^2/2} \int_{-\infty}^{\infty} x(\tau)e^{j \cot(\alpha)\tau^2/2} e^{-j t \tau \csc(\alpha)} d\tau, \end{aligned}$$

which is exactly the fractional Fourier transform up to the constant factor  $\sqrt{2\pi j e^{-j\alpha}}$  [8,32–34].

The fractional Fourier transform was reintroduced in the signal processing by Almeida. For an angle  $\alpha$  ( $\alpha \neq k\pi$ ) the fractional Fourier transform is defined as

$$X_\alpha(u) = \int_{-\infty}^{\infty} x(\tau) K_\alpha(u, \tau) d\tau,$$

where

$$K_\alpha(u, \tau) = \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j(u^2/2) \cot \alpha} e^{j(\tau^2/2) \cot \alpha} e^{-ju\tau \csc \alpha}.$$

Its inverse can be considered as a rotation for angle  $-\alpha$ :

$$x(t) = \int_{-\infty}^{\infty} X_\alpha(u) K_{-\alpha}(u, t) du.$$

Thus, the fractional Fourier transform is a special form of the signal transform which produces linear coordinate transformation in the time-frequency domain. The windowed fractional Fourier transform is

$$X_{w,\alpha}(t, u) = \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j(u^2/2) \cot \alpha} \int_{-\infty}^{\infty} x_t(\tau) w(\tau) e^{j(\tau^2/2) \cot \alpha} e^{-ju\tau \csc \alpha} d\tau,$$

where the local signal is  $x_t(\tau) = x(t + \tau)$ . Relation between the windowed fractional Fourier transform and the second order LPFT is

$$X_{w,\alpha}(t, u) = \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j(u^2/2) \cot \alpha} \text{LPFT}_{\Omega_1}(t, \Omega),$$

where  $\Omega_1 = \cot(\alpha)/2$  and  $\Omega = u \csc(\alpha)$ . Thus, all results can be easily converted from the second order LPFT to the windowed fractional Fourier transform, and vice versa.

These relations, leading to the Wigner distribution linear coordinate transforms, may also be used to produce some other signal transformation schemes (different from the fractional Fourier transform), which may be interesting in signal processing.

### 3.03.3.2.8 Auto-terms and cross-terms in the Wigner distribution

A drawback of the Wigner distribution is the presence of cross-terms when the multi-component signals are analyzed. For the multi-component signal

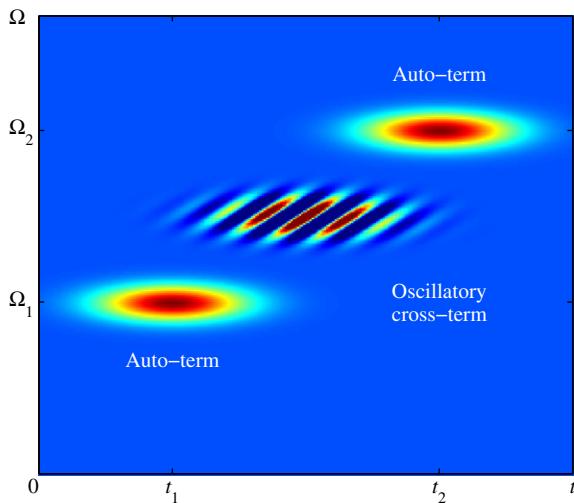
$$x(t) = \sum_{m=1}^M x_m(t)$$

the Wigner distribution has the form

$$\text{WD}(t, \Omega) = \sum_{m=1}^M \sum_{n=1}^M \int_{-\infty}^{\infty} x_m\left(t + \frac{\tau}{2}\right) x_n^*\left(t - \frac{\tau}{2}\right) e^{-j\Omega\tau} d\tau.$$

Besides the auto-terms

$$\text{WD}_{at}(t, \Omega) = \sum_{m=1}^M \int_{-\infty}^{\infty} x_m\left(t + \frac{\tau}{2}\right) x_m^*\left(t - \frac{\tau}{2}\right) e^{-j\Omega\tau} d\tau,$$

**FIGURE 3.14**

Wigner distribution of a two component signal.

the Wigner distribution contains a significant number of cross-terms,

$$\text{WD}_{ct}(t, \Omega) = \sum_{m=1}^M \sum_{\substack{n=1 \\ n \neq m}}^M \int_{-\infty}^{\infty} x_m \left( t + \frac{\tau}{2} \right) x_n^* \left( t - \frac{\tau}{2} \right) e^{-j\Omega\tau} d\tau.$$

Usually, they are not desirable in the time-frequency signal analysis. Cross-terms can mask the presence of auto-terms, which makes the Wigner distribution unsuitable for the time-frequency analysis of signals.

For a two-component signal with auto-terms located around  $(t_1, \Omega_1)$  and  $(t_2, \Omega_2)$  (see Figure 3.14) the oscillatory cross-terms are located around  $((t_1 + t_2)/2, (\Omega_1 + \Omega_2)/2)$ .

**Example 10.** For two-component signal of the form

$$x(t) = e^{-\frac{1}{2}(t-t_1)^2} e^{j\Omega_1 t} + e^{-\frac{1}{2}(t+t_1)^2} e^{-j\Omega_1 t}$$

we have

$$\begin{aligned} \text{WD}_x(t, \Omega) &= 2\sqrt{\pi} e^{-(t-t_1)^2 - (\Omega-\Omega_1)^2} + 2\sqrt{\pi} e^{-(t+t_1)^2 - (\Omega+\Omega_1)^2} \\ &\quad + 4\sqrt{\pi} e^{-t^2 - \Omega^2} \cos(2t_1\Omega - 2\Omega_1 t), \end{aligned}$$

where the first and second terms represent auto-terms while the third term is a cross-term. Note that the cross-term is oscillatory in both directions. The oscillation rate along the time axis is proportional to the frequency distance between components  $2\Omega_1$ , while the oscillation rate along frequency axis is proportional to the distance in time of components,  $2t_1$ . The oscillatory nature of cross-terms will be used for their suppression.

### 3.03.3.2.9 Inner interferences in the Wigner distribution

Another serious drawback of the Wigner distribution is in the presence of inner interferences for non-linear FM signals. Using the Taylor series expansion of the signal's  $x(t) = A \exp(j\phi(t))$  phase we get:

$$\begin{aligned} \text{WD}(t, \Omega) &= \int_{-\infty}^{\infty} A^2 e^{j\phi(t+\tau/2)} e^{-j\phi(t-\tau/2)} e^{-j\Omega\tau} d\tau \\ &= A^2 \int_{-\infty}^{\infty} \exp\left(j\phi'(t)\tau + 2j \sum_{k=1}^{\infty} \frac{\phi^{(2k+1)}(t)}{(2k+1)!} \left(\frac{\tau}{2}\right)^{2k+1} - j\Omega\tau\right) d\tau \\ &= 2\pi A^2 \delta(\Omega - \phi'(t)) *_{\Omega} \text{FT} \left\{ \exp\left(2j \sum_{k=1}^{\infty} \frac{\phi^{(2k+1)}(t)}{(2k+1)!} \left(\frac{\tau}{2}\right)^{2k+1}\right) \right\}, \end{aligned}$$

where  $\text{FT} \left\{ \exp\left(2j \sum_{k=1}^{\infty} \frac{\phi^{(2k+1)}(t)}{(2k+1)!} \left(\frac{\tau}{2}\right)^{2k+1}\right) \right\}$  is the term introducing interferences. The analytic form of this term can be obtained by using the stationary phase approximation.

For example, let us consider a cubic phase signal (quadratic frequency modulated) with Gaussian amplitude

$$x(t) = e^{-\frac{1}{2}(t/\alpha)^2} e^{jat^3}.$$

The Wigner distribution value is:

$$\text{WD}(t, \Omega) = e^{-t^2/\alpha^2} \int_{-\infty}^{\infty} e^{-\tau^2/(2\alpha)^2} e^{jat^3/4} e^{-j(\Omega-3at^2)\tau} d\tau.$$

The stationary phase points are

$$3a\tau_0^2/4 = \Omega - 3at^2$$

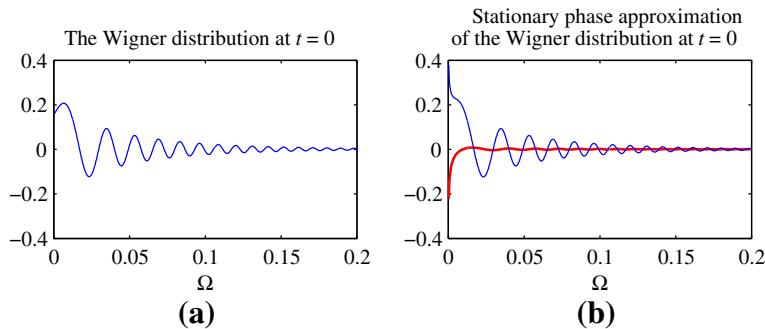
or  $\tau_{0+} = \sqrt{4(\Omega - 3at^2)/(3a)}$  and  $\tau_{0-} = -\sqrt{4(\Omega - 3at^2)/(3a)}$  for  $(\Omega - 3at^2) \geq 0$ , and

$$\phi''(\tau_0) = 3a\tau_0/2.$$

The resulting stationary phase approximation of the Wigner distribution is obtained by summing contribution from both stationary phase points,  $\tau_{0+}$  and  $\tau_{0-}$ , as

$$\begin{aligned} \text{WD}(t, \Omega) &= \sqrt{\frac{2\pi}{\sqrt{3a(\Omega - 3at^2)}}} e^{-t^2/\alpha^2} \\ &\times \exp\left(-\left[(\Omega - 3at^2)/(3a\alpha^2)\right]\right) \cos\left(\frac{4}{3\sqrt{3a}} \left[(\Omega - 3at^2)\right]^{3/2} - \pi/4\right) \end{aligned}$$

for  $\Omega - 3at^2 \geq 0$  and  $\text{WD}(t, \Omega) = 0$  for  $\Omega - 3at^2 < 0$ . For  $t = 0$ , significant oscillatory values are up to  $\Omega/(3a\alpha^2) \sim 1$ , since the attenuation in frequency is  $\exp(-[(\Omega - 3at^2)/(3a\alpha^2)])$ . Note that this is not in agreement with the expectation that the instantaneous bandwidth  $\sigma_{\Omega}^2(0) = 1/(2\alpha)$ , calculated according to (3.64), is small for a large  $\alpha$ .

**FIGURE 3.15**

Stationary phase approximation of the Wigner distribution of a cubic-phase signal. The approximation error is presented with thick red line. (For interpretation of the references to color in this Figure 3.15 legend, the reader is referred to the web version of this book.)

Note that the stationary phase is an approximation, producing accurate results for large arguments. In this case, exact Wigner distribution almost coincides with this approximation, already for  $\Omega - 3at^2 > \pi/4$  as presented in Figure 3.15.

If these terms are not reduced, they can reduce the accuracy of the time-frequency representation of a signal.

### 3.03.3.2.10 Pseudo and smoothed Wigner distribution

In practical realizations of the Wigner distribution, we are constrained with a finite time lag  $\tau$ . A pseudo form of the Wigner distribution is then used [2, 9, 10, 13, 18, 23, 35]. It is defined as

$$\text{PWD}(t, \Omega) = \int_{-\infty}^{\infty} w(\tau/2)w^*(-\tau/2)x(t + \tau/2)x^*(t - \tau/2)e^{-j\Omega\tau} d\tau, \quad (3.69)$$

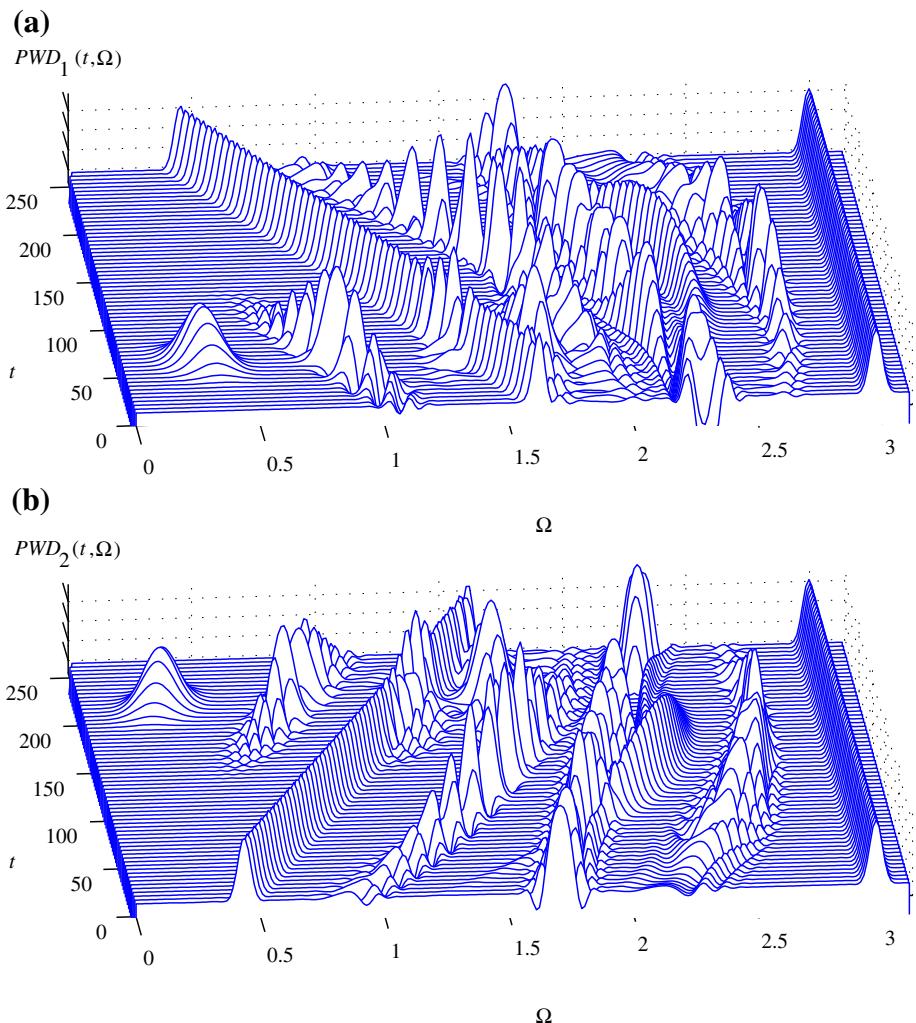
where window  $w(\tau)$  localizes the considered lag interval. If  $w(0) = 1$ , the pseudo Wigner distribution satisfies the time marginal property. Note that the pseudo Wigner distribution is smoothed in the frequency direction with respect to the Wigner distribution

$$\text{PWD}(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}(t, \theta)W_e(\Omega - \theta)d\theta,$$

where  $W_e(\Omega)$  is a FT of  $w(\tau/2)w^*(-\tau/2)$ . The pseudo Wigner distribution example for multi-component signals is presented in Figure 3.16. Mono-component case with sinusoidally frequency modulated signal is presented in Figure 3.17. Note that significant inner interferences are present.

In order to reduce the interferences in the Wigner distribution, it is sometimes smoothed not only in the frequency axis direction, but also in time, by using time-smoothing window  $G(t)$ . This form is called the smoothed Wigner distribution

$$\text{SWD}(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(t - u)W_e(\Omega - \theta)\text{WD}(u, \theta)du d\theta. \quad (3.70)$$

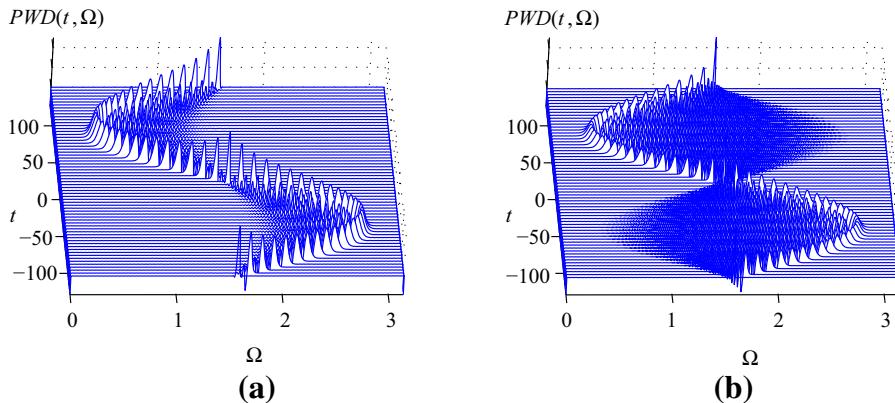
**FIGURE 3.16**

The pseudo Wigner distribution of signals from Figure 3.1.

The smoothed Wigner distribution with

$$G(t)W_e(\Omega) = \alpha e^{-\gamma t^2} e^{-\beta \Omega^2} = \alpha e^{-(\gamma t^2 + \beta \Omega^2)}$$

is equal to the spectrogram with window  $w(\tau) = e^{-\gamma \tau^2/2}$  if  $\gamma = 1/\beta$ . This smoothed Wigner distribution is always positive.

**FIGURE 3.17**

The pseudo Wigner distribution for a sinusoidally frequency modulated signal. A narrow window (left) and a wide window (right).

### 3.03.3.2.11 Discrete pseudo Wigner distribution

The pseudo Wigner distribution of a discrete-time signal, with a finite length lag window, is given by

$$\text{PWD}(n, \omega) = \sum_{m=-N/2}^{N/2} w(m)w^*(-m)x(n+m)x^*(n-m)e^{-j2m\omega}. \quad (3.71)$$

Note that the pseudo Wigner distribution is periodic in  $\omega$  with period  $\pi$ . The signal should be sampled at a twice higher sampling rate than it is required by the sampling theorem,  $\Delta t \leq \pi/(2\Omega_m)$ . Thus, with the same lag window length the pseudo Wigner distribution will have twice more samples than the STFT. In order to produce an unbiased approximation of the analog form (3.69), the sampled signal in (3.71) should be formed as  $x(n) = x(n\Delta t)\sqrt{2\Delta t}$ .

The discrete time and frequency form is given by

$$\text{PWD}(n, k) = \sum_{m=-N/2}^{N/2} w(m)w^*(-m)x(n+m)x^*(n-m)e^{-j4\pi mk/(N+1)}$$

and may also be efficiently calculated by using the FFT routines. Note that discrete frequency  $\omega$  is related to frequency index  $k$  as  $\omega = 2k\pi/(N+1)$ .

In order to avoid the need for oversampling, as well as to eliminate cross-terms between positive and negative frequency components in real signals, the real valued signals are usually transformed into their analytic forms  $x_a(t) = x(t) + jx_h(t)$ , where  $x_h(t)$  is the signal's Hilbert transform. In the frequency domain  $X_a(\Omega) = 2X(\Omega)$ , for  $\Omega > 0$  and  $X_a(\Omega) = 0$  for  $\Omega < 0$ , while  $X_a(\Omega) = X(\Omega)$ , for  $\Omega = 0$ . Pseudo Wigner distribution is then calculated based on the analytic form of a signal. A STFT-based approach for creating the alias-free Wigner distribution will be also described later in the text.

### 3.03.3.2.12 Wigner distribution based inversion and synthesis

In order to define an efficient algorithm for the synthesis of a signal with specified time-frequency distribution, we will restate the Wigner distribution inversion within the eigenvalue and eigenvectors decomposition framework. A discrete form of the Wigner distribution is defined by

$$\text{WD}(n, k) = \sum_{m=-N/2}^{N/2} x(n+m)x^*(n-m)e^{-j\frac{2\pi}{N+1}2mk}, \quad (3.72)$$

where we assume that the signal  $x(n)$  is time limited within  $|n| \leq N/2$ . Inversion relation for the Wigner distribution reads

$$x(n+m)x^*(n-m) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{WD}(n, k)e^{j\frac{2\pi}{N+1}m(2k)}.$$

After substitutions  $n_1 = n + m$  and  $n_2 = n - m$  we get

$$x(n_1)x^*(n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{WD}\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)}. \quad (3.73)$$

For cases when  $(n_1 + n_2)/2$  is not an integer, an appropriate interpolation is performed in order to calculate  $\text{WD}((n_1 + n_2)/2, k)$ .

Note that relation (3.73) is a discrete counterpart of the Wigner distribution inversion in analog domain, that reads:

$$x(t_1)x^*(t_2) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{WD}((t_1 + t_2)/2, \Omega)e^{j\Omega(t_1-t_2)}d\Omega.$$

By discretization of angular frequency  $\Omega = k\Delta\Omega$  and time  $t_1 = n_1\Delta t$ ,  $t_2 = n_2\Delta t$ , with appropriate definition of discrete values, we easily obtain (3.73).

Introducing the notation,

$$R(n_1, n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{WD}\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)}, \quad (3.74)$$

we get

$$R(n_1, n_2) = x(n_1)x^*(n_2). \quad (3.75)$$

Matrix form of (3.75) reads

$$\mathbf{R} = \mathbf{x}(n)\mathbf{x}^H(n), \quad (3.76)$$

where  $\mathbf{x}(n)$  is a column vector whose elements are the signal values,  $\mathbf{x}^H(n)$  is a row vector (Hermitian transpose of  $\mathbf{x}(n)$ ), and  $\mathbf{R}$  is a matrix with the elements  $R(n_1, n_2)$ , defined by (3.74).

The eigenvalue decomposition of  $\mathbf{R}$  reads

$$\mathbf{R} = \mathbf{Q}\Lambda\mathbf{Q}^T = \sum_{i=1}^{N+1} \lambda_i \mathbf{q}_i(n)\mathbf{q}_i^H(n), \quad (3.77)$$

where  $\lambda_i$  are eigenvalues and  $\mathbf{q}_i(n)$  are corresponding eigenvectors of  $\mathbf{R}$ . By comparing (3.76) and (3.77), it follows that the matrix with elements of form (3.74) can be decomposed by using only one non-zero eigenvalue. Note that the energy of the corresponding eigenvector is equal to 1, by definition  $\|\mathbf{q}_1(n)\| = 1$ . By comparing (3.76) and (3.77), having in mind that there is only one non-zero eigenvalue  $\lambda_1$ , we have

$$\mathbf{x}(n)\mathbf{x}^H(n) = \lambda_1 \mathbf{q}_1(n) \mathbf{q}_1^H(n) = (\sqrt{\lambda_1} \mathbf{q}_1(n))(\sqrt{\lambda_1} \mathbf{q}_1(n))^H$$

and

$$\lambda_1 = \left\| \sqrt{\lambda_1} \mathbf{q}_1(n) \right\|^2 = \|\mathbf{x}(n)\|^2 = \sum_{n=-N/2}^{N/2} x^2(n) = E_x. \quad (3.78)$$

The eigenvector  $\mathbf{q}_1(n)$  is equal to the signal vector  $\mathbf{x}(n)$ , up to the constant amplitude and phase factor. Therefore, an eigenvalue decomposition of the matrix, formed according to (3.74), can be used to check if an arbitrary 2D function  $D(n, k)$  is a valid Wigner distribution.

These relations can be used in signal synthesis. Assume that we have a given function  $D(n, k)$ , calculate (3.74) and perform eigenvalue decomposition (3.77). If the given function is the Wigner distribution of a signal it will result in one non-zero eigenvalue and corresponding eigenvector. If that is not the case then the first (largest) eigenvalue and corresponding eigenvector produce a signal such that its Wigner distribution will be the closest possible Wigner distribution (in the LMS sense) to the given arbitrary function  $D(n, k)$ . This conclusion follows from the eigenvalue/eigenvectors decomposition properties.

### 3.03.3.3 Ambiguity function

To analyze auto-terms and cross-terms, the well-known ambiguity function can be used as well. It is defined as:

$$\text{AF}(\theta, \tau) = \int_{-\infty}^{\infty} x\left(t + \frac{\tau}{2}\right) x^*\left(t - \frac{\tau}{2}\right) e^{-j\theta t} dt. \quad (3.79)$$

It is already a classical tool in optics as well as in radar and sonar signal analysis.

The ambiguity function and the Wigner distribution form a two-dimensional FT pair

$$\text{AF}(\theta, \tau) = \text{FT}_{t, \Omega}^{2D}\{\text{WD}(t, \Omega)\}.$$

Consider a signal whose components are limited in time to

$$x_m(t) \neq 0 \quad \text{only for } |t - t_m| < T_m.$$

In the ambiguity  $(\theta, \tau)$  domain we have  $x_m(t + \tau/2)x_m^*(t - \tau/2) \neq 0$  only for

$$\begin{aligned} -T_m &< t - t_m + \tau/2 < T_m, \\ -T_m &< t - t_m - \tau/2 < T_m. \end{aligned}$$

It means that  $x_m(t + \tau/2)x_m^*(t - \tau/2)$  is located within  $|\tau| < 2T_m$ , i.e., around the  $\theta$ -axis independently of the signal's position  $t_m$ . Cross-term between signal's  $m$ th and  $n$ th component is located within  $|\tau + t_n - t_m| < T_m + T_n$ . It is dislocated from  $\tau = 0$  for two-components that do not occur simultaneously, i.e., when  $t_m \neq t_n$ .

From the frequency domain definition of the Wigner distribution a corresponding ambiguity function form follows:

$$\text{AF}(\theta, \tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X\left(\Omega + \frac{\theta}{2}\right) X^*\left(\Omega - \frac{\theta}{2}\right) e^{j\Omega\tau} d\Omega. \quad (3.80)$$

From this form we can conclude that the auto-terms of the components, limited in frequency to  $X_m(\Omega) \neq 0$  only for  $|\Omega - \Omega_m| < W_m$ , are located in the ambiguity domain around  $\tau$ -axis within the region  $|\theta/2| < W_m$ . The cross-terms are within

$$|\theta + \Omega_n - \Omega_m| < W_m + W_n,$$

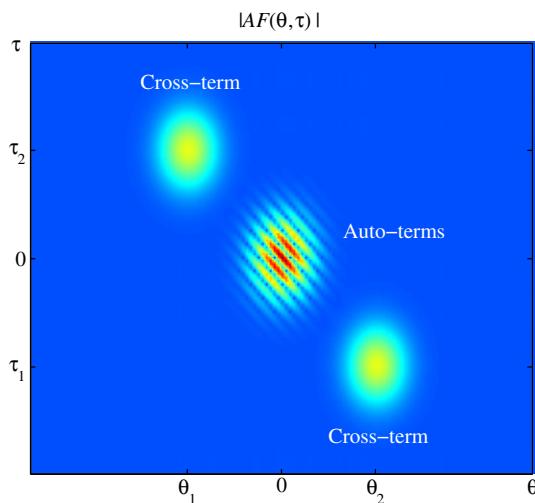
where  $\Omega_m$  and  $\Omega_n$  are the frequencies around which the FT of each component lies.

Therefore, all auto-terms are located along and around the ambiguity domain axis. The cross-terms, for the components which do not overlap in the time and frequency, simultaneously, are dislocated from the ambiguity axes, Figure 3.18. This property will be used in the definition of the reduced interference time-frequency distributions.

The ambiguity function of a four-component signal consisting of two Gaussian pulses, one sinusoidal and one linear frequency modulated component is presented in Figure 3.19.

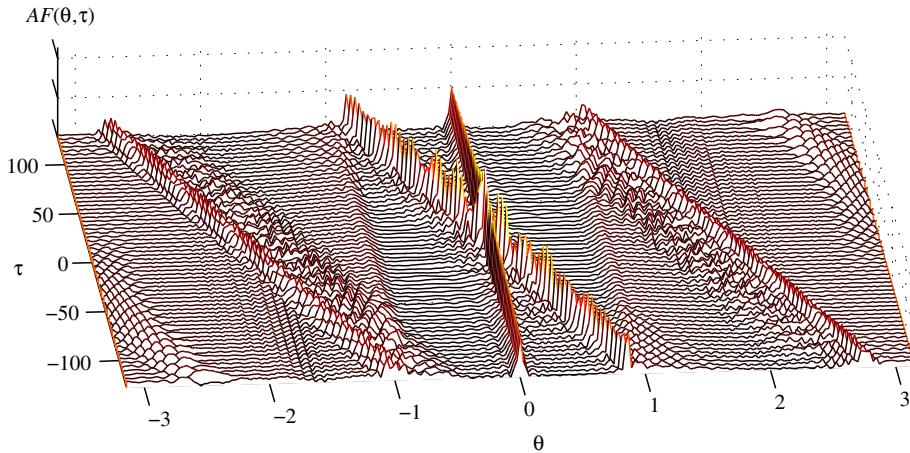
**Example 11.** Let us consider signals of the form

$$\begin{aligned} x_1(t) &= e^{-\frac{1}{2}t^2}, \\ x_2(t) &= e^{-\frac{1}{2}(t-t_1)^2} e^{j\Omega_1 t} + e^{-\frac{1}{2}(t+t_1)^2} e^{-j\Omega_1 t}. \end{aligned}$$



**FIGURE 3.18**

Auto and cross-terms in a two-component signal in the ambiguity domain.

**FIGURE 3.19**

Ambiguity function of the signal from Figure 3.1.

The ambiguity function of  $x_1(t)$  is

$$\text{AF}_{x_1}(\theta, \tau) = \sqrt{\pi} e^{-\frac{1}{4}\tau^2 - \frac{1}{4}\theta^2}$$

while the ambiguity function of two-component signal  $x_2(t)$  is

$$\begin{aligned} \text{AF}_{x_2}(\theta, \tau) = & \sqrt{\pi} e^{-\frac{1}{4}\tau^2 - \frac{1}{4}\theta^2} e^{j\Omega_1 \tau} e^{-jt_1 \theta} + \sqrt{\pi} e^{-\frac{1}{4}\tau^2 - \frac{1}{4}\theta^2} e^{-j\Omega_1 \tau} e^{jt_1 \theta} \\ & + \sqrt{\pi} e^{-\frac{1}{4}(\tau - 2t_1)^2 - \frac{1}{4}(\theta - 2\Omega_1)^2} + \sqrt{\pi} e^{-\frac{1}{4}(\tau + 2t_1)^2 - \frac{1}{4}(\theta + 2\Omega_1)^2}. \end{aligned}$$

In the ambiguity domain  $(\theta, \tau)$  auto-terms are located around  $(0, 0)$  while cross-terms are located around  $(2\Omega_1, 2t_1)$  and  $(-2\Omega_1, -2t_1)$  as presented in Figure 3.18.

**Example 12.** Show that the second order moment of the signal

$$x_L(t) = \left( (x(t)e^{jct^2/2}) * \sqrt{|b|} e^{jbt^2/2} \right) e^{jat^2/2}$$

may be calculated based on the signal's and the Fourier transform's second order moments and the joint first order moment.

For signal  $x_L(t)$ , the Wigner distribution is obtained by linear coordinate transformation of the Wigner distribution of a signal  $x(t)$ ,

$$\text{WD}_{x_L}(t, \Omega) = \text{WD}_x(u, v) = \text{WD}_x(At + B\Omega, Ct + D\Omega). \quad (3.81)$$

The coordinate transformation matrix has the form

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 1 + a/b & -1/b \\ -c - a(c/b + 1) & c/b + 1 \end{bmatrix} \begin{bmatrix} t \\ \Omega \end{bmatrix}$$

with  $A, B, C$ , and  $D$  being related to  $a, b, c$  by the expressions in the transformation matrix.

The second order moment of  $x_L(t)$  is

$$\begin{aligned} m_2(a, b, c) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} t^2 \text{WD}_{x_L}(t, \Omega) dt d\Omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} t^2 \text{WD}_x(At + B\Omega, Ct + D\Omega) dt d\Omega. \end{aligned}$$

With a change of variables  $At + B\Omega = u$  and  $Ct + D\Omega = v$ ,  $t = Du - Bv$ , having in mind that the transformation is unitary,  $AD - BC = 1$ ,

$$\begin{aligned} m_2(a, b, c) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (D^2u^2 - 2BDuv + B^2v^2) \text{WD}_x(u, v) du dv \\ &= D^2 \int_{-\infty}^{\infty} t^2 |x(t)|^2 dt - 2BD\mu_{11} + B^2 \frac{1}{2\pi} \int_{-\infty}^{\infty} \Omega^2 |X(\Omega)|^2 d\Omega \\ &= D^2 m_2 - 2BD\mu_{11} + B^2 M_2, \end{aligned} \quad (3.82)$$

where  $M_2$  could be calculated as  $M_2 = \int_{-\infty}^{\infty} |x'(t)|^2 dt$  and

$$\begin{aligned} \mu_{11} &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} t\Omega \text{WD}(t, \Omega) dt d\Omega \\ &= \left. \frac{\partial^2 A(\theta, \tau)}{\partial \theta \partial \tau} \right|_{\theta=0, \tau=0} = -\frac{j}{2} \int_{-\infty}^{\infty} t(x'(t)x^*(t) - x(t)x^{*\prime}(t)) dt. \end{aligned}$$

This relation is useful for multiparameter optimization in order to find time-frequency representation (with distribution coordinate transformation) that would produce the best concentrated signal, with minimal moment  $m_2(a, b, c)$ . Similar relation was obtained in the local polynomial Fourier transform analysis. A special case, that reduces to the time-frequency plane rotation with  $-1/b = \sin(\alpha) = 1/\csc(\alpha)$  and  $a = c = -\tan(\alpha/2)$  is used in practice by fractional Fourier transforms [8,33].

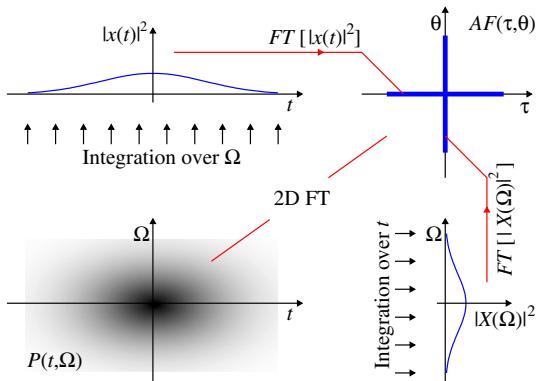
### 3.03.3.4 Cohen class of distributions

Time and frequency marginal properties (3.42) and (3.43) may be considered as the projections of the distribution  $P(t, \Omega)$  along the time and frequency axes, i.e., as the Radon transform of  $P(t, \Omega)$  along these two directions. It is known that the FT of the projection of a two-dimensional function on a given line is equal to the value of the two-dimensional FT of  $P(t, \Omega)$ , denoted by  $\text{AF}(\theta, \tau)$ , along the same direction (inverse Radon transform property). Therefore, if  $P(t, \Omega)$  satisfies marginal properties then any other function having two-dimensional FT equals to  $\text{AF}(\theta, \tau)$  along the axes lines  $\theta = 0$  and  $\tau = 0$ , and arbitrary values elsewhere, will satisfy marginal properties, Figure 3.20.

Assuming that the Wigner distribution is a basic distribution which satisfies the marginal properties (any other distribution satisfying marginals can be used as the basic one), then any other distribution with two-dimensional FT

$$\text{AF}_g(\theta, \tau) = c(\theta, \tau) \text{FT}_{t, \Omega}^{2D}\{\text{WD}(t, \Omega)\} = c(\theta, \tau) \text{AF}(\theta, \tau), \quad (3.83)$$

where  $c(0, \tau) = 1$  and  $c(\theta, 0) = 1$ , satisfies marginal properties as well.

**FIGURE 3.20**

Marginal properties and their relation to the ambiguity function.

The inverse two-dimensional FT of  $AF_g(\theta, \tau)$  produces the Cohen class of distributions, introduced from quantum mechanics into the time-frequency analysis by Claassen and Mecklenbäuker, in the form

$$CD(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\theta, \tau) x(u + \tau/2) x^*(u - \tau/2) e^{j\theta t - j\Omega\tau - j\theta u} du d\tau d\theta, \quad (3.84)$$

where  $c(\theta, \tau)$  is called the kernel in the ambiguity domain.

Alternatively, the frequency domain definition of the Cohen class of distributions is

$$CD(t, \Omega) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(u - \theta/2) X^*(u + \theta/2) c(\theta, \tau) e^{j\theta t - j\tau\Omega + j\tau u} du d\tau d\theta. \quad (3.85)$$

Various distributions can be obtained by altering the kernel function  $c(\theta, \tau)$ . For example,  $c(\theta, \tau) = 1$  produces the Wigner distribution, while for  $c(\theta, \tau) = e^{j\theta\tau/2}$  the Rihaczek distribution follows.

The Cohen class of distributions, defined in the ambiguity domain:

$$CD(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\theta, \tau) AF(\theta, \tau) e^{j\theta t - j\Omega\tau} d\tau d\theta \quad (3.86)$$

can be written in other domains, as well. The time-lag domain form is obtained from (3.84), after integration on  $\theta$ , as:

$$CD(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c_T(t - u, \tau) x(u + \tau/2) x^*(u - \tau/2) e^{-j\Omega\tau} d\tau du. \quad (3.87)$$

The frequency-Doppler frequency domain form follows from (3.85), after integration on  $\tau$ , as:

$$CD(t, \Omega) = \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} C_{\Omega}(\theta, \Omega - u) X(u + \theta/2) X^*(u - \theta/2) e^{j\theta t} d\theta du. \quad (3.88)$$

Finally, the time-frequency domain form is obtained as a two-dimensional convolution of the two-dimensional FTs, from (3.86), as:

$$\text{CD}(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pi(t-u, \Omega-\xi) \text{WD}(u, \xi) du d\xi. \quad (3.89)$$

Kernel functions in the respective time-lag, Doppler frequency-frequency and time-frequency domains are related to the ambiguity domain kernel  $c(\theta, \tau)$  as:

$$c_T(t, \tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} c(\theta, \tau) e^{j\theta t} d\theta, \quad (3.90)$$

$$C_{\Omega}(\theta, \Omega) = \int_{-\infty}^{\infty} c(\theta, \tau) e^{-j\Omega\tau} d\tau, \quad (3.91)$$

$$\Pi(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\theta, \tau) e^{j\theta t - j\Omega\tau} d\tau d\theta. \quad (3.92)$$

According to (3.89) all distributions from the Cohen class may be considered as 2D filtered versions of the Wigner distribution. Although any distribution could be taken as a basis for the Cohen class derivation, the form with the Wigner distribution is used because it is the best concentrated distribution

**Table 3.1** Properties of the Distributions from the Cohen Class

	Distribution property	Kernel constraint
P <sub>1</sub>	Realness	$c(\theta, \tau) = c^*(-\theta, -\tau)$
P <sub>2</sub>	Time shift	Any kernel
P <sub>3</sub>	Frequency shift	Any kernel
P <sub>4</sub>	Time marginal	$c(\theta, 0) = 1$
P <sub>5</sub>	Frequency marginal	$c(0, \tau) = 1$
P <sub>6</sub>	Time moments	$c(\theta, 0) = 1$
P <sub>7</sub>	Frequency moments	$c(0, \tau) = 1$
P <sub>8</sub>	Scaling	$c(\theta, \tau) = c(a\theta, \tau/a)$
P <sub>9</sub>	Instantaneous frequency	$c(\theta, 0) = 1, \left. \frac{\partial c(\theta, \tau)}{\partial \tau} \right _{\tau=0} = 0$
P <sub>10</sub>	Group delay	$c(0, \tau) = 1, \left. \frac{\partial c(\theta, \tau)}{\partial \theta} \right _{\theta=0} = 0$
P <sub>11</sub>	Time constraint	$c_T(t, \tau) = 0 \text{ for }  t/\tau  > 1/2$
P <sub>12</sub>	Frequency constraint	$C_{\Omega}(\theta, \Omega) = 0 \text{ for }  \Omega/\theta  > 1/2$
P <sub>13</sub>	Convolution	$c(\theta, \tau_1)c(\theta, \tau_2) = c(\theta, \tau_1 + \tau_2)$
P <sub>14</sub>	Product	$c(\theta_1, \tau)c(\theta_2, \tau) = c(\theta_1 + \theta_2, \tau)$
P <sub>15</sub>	Fourier transform	$c(\theta, \tau) = c(c\tau, -\theta/c)$
P <sub>16</sub>	Chirp convolution	$c(\theta, \tau) = c(\theta, \tau - \theta/c)$
P <sub>17</sub>	Chirp product	$c(\theta, \tau) = c(\theta + c\tau, \tau)$
P <sub>18</sub>	Moyal property	$ c(\theta, \tau) ^2 = 1$

from the Cohen class with the signal independent kernels. Note that the Cohen class of distributions is more general than the class of distributions, in the literature referred to as the smoothed Wigner distributions (3.70), since generally  $\Pi(t - u, \Omega - v)$  is not a separable function.

Desired properties of the time-frequency representations, presented in the case of the Wigner distribution, are satisfied for a distribution from the Cohen class under the kernel constraints presented in Table 3.1 [2,4,6,10,30].

#### 3.03.3.4.1 Reduced interference distributions

The analysis performed on ambiguity function and Cohen class of time-frequency distributions leads to the conclusion that the cross-terms may be suppressed or eliminated, if a kernel  $c(\theta, \tau)$  is a function of a two-dimensional low-pass type. In order to preserve the marginal properties  $c(\theta, \tau)$  values along the axis should be  $c(\theta, 0) = 1$  and  $c(0, \tau) = 1$ .

Choi and Williams exploited one of the possibilities defining the distribution with the kernel of the form

$$c(\theta, \tau) = e^{-\theta^2 \tau^2 / \sigma^2}.$$

The parameter  $\sigma$  controls the slope of the kernel function which affects the influence of cross-terms. Small  $\sigma$  causes the elimination of cross-terms but it should not be too small because, for the finite width of the auto-terms around  $\theta$  and  $\tau$  coordinates, the kernel will cause their distortion, as well. Thus, there should be a trade-off in the selection of  $\sigma$ .

Here we will mention some other interesting kernel functions, producing corresponding distributions, Figure 3.21 [2,4,10,13,18,36,37]:

Born-Jordan distribution

$$c(\theta, \tau) = \frac{\sin\left(\frac{\theta\tau}{2}\right)}{\frac{\theta\tau}{2}},$$

Zhao-Atlas-Marks distribution

$$c(\theta, \tau) = w(\tau) |\tau| \frac{\sin\left(\frac{\theta\tau}{2}\right)}{\frac{\theta\tau}{2}},$$

Sinc distribution

$$c(\theta, \tau) = \text{rect}\left(\frac{\theta\tau}{\alpha}\right) = \begin{cases} 1 & \text{for } |\theta\tau/\alpha| < 1/2, \\ 0 & \text{otherwise.} \end{cases}$$

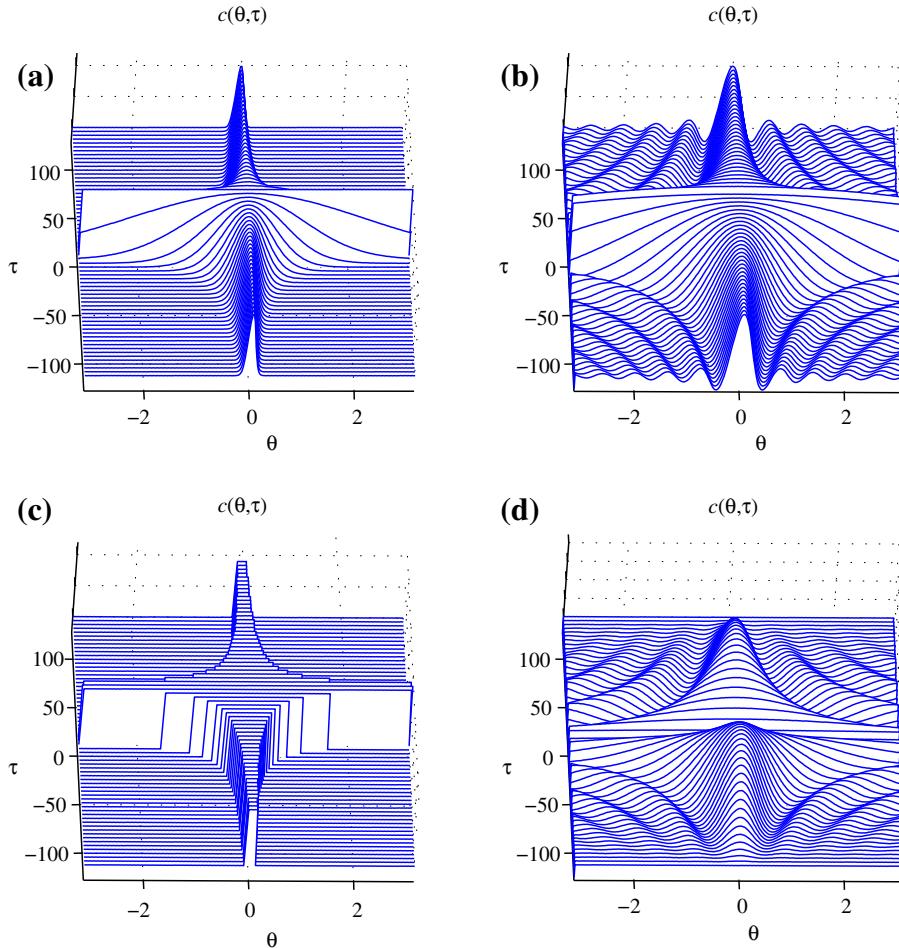
Butterworth distribution

$$c(\theta, \tau) = \frac{1}{1 + \left(\frac{\theta\tau}{\theta_c \tau_c}\right)^{2N}},$$

where  $w(\tau)$  is a function corresponding to a lag window and  $\alpha$ ,  $N$ ,  $\theta_c$ , and  $\tau_c$  are constants in the above kernel definitions.

The spectrogram belongs to this class of distributions. Its kernel in  $(\theta, \tau)$  domain is the ambiguity function of the window

$$c(\theta, \tau) = \int_{-\infty}^{\infty} w\left(t - \frac{\tau}{2}\right) w\left(t + \frac{\tau}{2}\right) e^{-j\theta t} dt = \text{AF}_w(\theta, \tau).$$

**FIGURE 3.21**

Kernel functions in the ambiguity domain for: (a) the Choi-Williams distribution, (b) the Born-Jordan distribution, (c) the sinc distribution, and (d) the Zhao-Atlas-Marks distribution.

Since the Cohen class is linear with respect to the kernel, it is easy to conclude that a distribution from the Cohen class is positive if its kernel can be written as

$$c(\theta, \tau) = \sum_{i=1}^M a_i \text{AF}_{w_i}(\theta, \tau),$$

where  $a_i \geq 0, i = 1, 2, \dots, M$ .

There are several ways for calculation of the reduced interference distributions from the Cohen class. The first method is based on the ambiguity function (3.86):

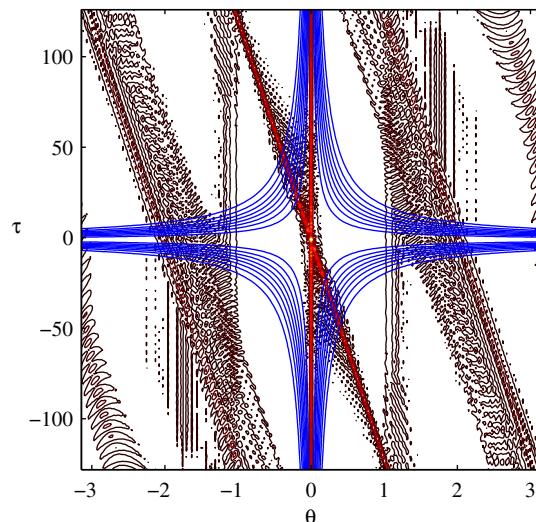
1. Calculation of the ambiguity function.
2. Multiplication with the kernel.
3. Calculation of the inverse two-dimensional FT of this product.

The reduced interference distribution may also be calculated by using (3.87) or (3.89) with appropriate kernel transformations defined by (3.90) and (3.92). All these methods assume signal oversampling in order to avoid aliasing effects. Figure 3.22 presents the ambiguity function along with kernel (Choi-Williams). Figure 3.23a presents Choi-Williams distribution calculated according to the presented procedure. In order to reduce high side lobes of the rectangular window, the Choi-Williams distribution is also calculated with the Hann(ing) window in the kernel definition  $c(\theta, \tau)w(\tau)$  and presented in Figure 3.23b. The pseudo Wigner distribution with Hann(ing) window is shown in Figure 3.16.

The optimal kernel form, introduce by Baraniuk and Jones, is based on the ambiguity function and the kernel form optimization in the ambiguity domain [38, 39]. For a given signal and its ambiguity function, the optimal kernel is obtained as a real, non-negative function obtained as a solution of the following optimization problem:

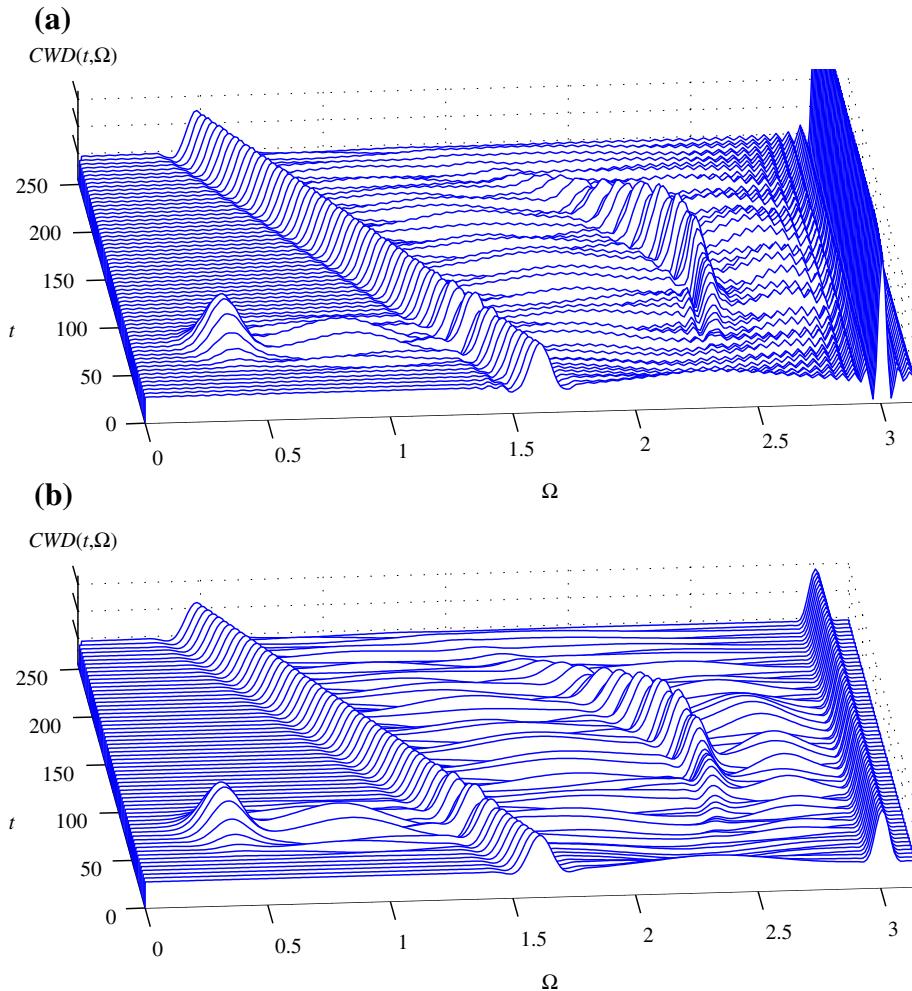
$$\max \iint |A(\theta, \tau)c(\theta, \tau)|^2 d\theta d\tau$$

AF( $\theta, \tau$ ) and CW kernel



**FIGURE 3.22**

Ambiguity function for the signal from Figure 3.1 the Choi-Williams kernel (thick blue contour lines). (For interpretation of the references to color in this Figure 3.22 legend, the reader is referred to the web version of this book.)

**FIGURE 3.23**

The Choi-Williams distribution by a: (a) direct calculation and (b) calculation with the kernel multiplied by a Hann(ing) lag window.

subject to

$$c(0, 0) = 1, \text{ unbiased signal energy condition,}$$

$$c(\theta, \tau) \text{ is radially non-increasing function,}$$

and kernel energy constraint:

$$\frac{1}{2\pi} \iint |c(\theta, \tau)|^2 d\theta d\tau < \alpha, \quad \alpha > 0.$$

In the derivation and the analysis these constraints were also expressed in the polar coordinate system with  $\theta = \rho \cos \psi$  and  $\tau = \rho \sin \psi$ . We can understand this optimization as the procedure to find the kernel that passes auto-terms and suppresses cross-terms. Since the auto-terms are centered about the ambiguity domain origin, while the cross-terms are dislocated from the origin, low-pass kernels are used. The constraints force the kernel to be a low-pass filter of fixed energy lower than  $\alpha$ . These constraints are quite general and do not dictate the exact shape of the kernel. It is determined by maximizing the performance measure. Without the monotonicity constraint, the optimal kernel would be large, regardless of whether the peaks correspond to auto-terms or cross-terms. However, assuming that the auto-terms and cross-terms are separated in the ambiguity plane, the monotonicity constraint imposes a penalty on kernels whose pass-bands extend over cross-terms.

#### 3.03.3.4.2 Auto-terms form in the Cohen class of distributions

An ideally concentrated distribution of a signal  $x(t) = Ae^{j\phi(t)}$ , having the form  $\text{ITF}(\Omega, t) = 2\pi A^2 \delta(\Omega - \phi'(t))$ , may be easily translated into the general form (taking an inverse 2-D FT of  $\text{ITF}(\Omega, t)$ ) as

$$\text{ITF}(\Omega, t) = \frac{A^2}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-j\phi'(u)\tau} e^{j\theta t - j\Omega\tau - j\theta u} du d\theta d\tau. \quad (3.93)$$

Comparing (3.93) with the Cohen class definition, while having in mind uniqueness of the FT, we get that signal  $x(t) = Ae^{j\phi(t)}$  has the distribution equal to the ideal one if

$$c(\theta, \tau) e^{j\phi(u+\tau/2) - j\phi(u-\tau/2)} = e^{j\phi'(u)\tau}.$$

Expanding  $\phi(u \pm \tau/2)$  into a Taylor series around  $u$ , up to the second order term, we get

$$c(\theta, \tau) = e^{-j\frac{\phi^{(3)}(u+\tau_1)+\phi^{(3)}(u-\tau_2)}{3!}\left(\frac{\tau}{2}\right)^3},$$

where  $\tau_1$  and  $\tau_2$  are variables ranging from 0 to  $\tau/2$ . From the last equation one may conclude that for any signal  $x(t)$  there exists a signal dependent kernel, such that the Cohen distribution is equal to the ideal one. With the assumption of signal independent kernel (which is of practical importance) we get that the ideal distribution may be obtained only if  $\phi^{(3)}(u) \equiv 0$ , i.e.,  $c(\theta, \tau) \equiv 1$ . The previous requirement ( $\phi^{(3)}(u) \equiv 0$ ) is met only if the signal is linear frequency modulated  $x(t) = Ae^{j(at^2/2+bt)}$ . The kernel  $c(\theta, \tau) \equiv 1$  corresponds to the Wigner distribution. Any other distribution will have auto-terms that are more or less distorted when compared with the ideal representation [35,37].

Let us consider how other members of the Cohen distribution behave for linear frequency modulated signal  $x(t) = Ae^{j(at^2/2+bt)}$ :

$$\begin{aligned} \text{CD}(\Omega, t) &= A^2 \int_{-\infty}^{\infty} c(a\tau, \tau) e^{j(at+b-\Omega)\tau} d\tau \\ &= A^2 C(\Omega - at - b) \end{aligned}$$

with

$$C(\Omega) = \text{FT}\{c(a\tau, \tau)\}. \quad (3.94)$$

The auto-term shape is determined by the function  $C(\Omega)$  which will be referred to as the **auto-term function**. According to (3.94), one is able to derive the auto-term function for any distribution from

the Cohen class. In addition, based on (3.94), one may construct a distribution with the desired auto-term shape, in the following way: If  $C(\Omega)$  is a given auto-term function, for the linear frequency modulated signal  $x(t) = Ae^{jat^2/2+bt}$  with an instantaneous frequency rate  $a$ , then the product kernel,  $c(\theta, \tau) = c(\theta\tau)$ , which will produce this auto-term form, can be determined as

$$c(\theta\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} C(\Omega) e^{j\Omega\tau} d\Omega \Big|_{a\tau^2=\theta\tau}. \quad (3.95)$$

**Example 13.** We have seen that the width of  $c(\tau, \tau)$  (i.e., the width of  $c(a\tau, \tau)$  for  $a = 1$ ) should be as small as possible in order to have high cross-terms suppression. However, at the same time, the width of auto-term function  $C(\Omega) = \text{FT}\{c(\tau, \tau)\}$  should be small (i.e.,  $c(\tau, \tau)$  wide) in order to produce a concentrated and sharp distribution in the time-frequency plane. Product of the measures of widths of  $c(\tau, \tau)$  (denoted by  $\sigma_\tau$ ) and its FT  $C(\Omega)$  (denoted by  $\sigma_\Omega$ ) is constant for a given kernel. It satisfies the uncertainty principle relation  $\sigma_\tau \sigma_\Omega \geq 1/2$ . Thus, if one fixes the value of  $\sigma_\Omega$  (the measure of the auto-term width) then the remaining value  $\sigma_\tau$  (being the measure of cross-terms suppression) will be minimal if  $\sigma_\tau \sigma_\Omega$  is minimal, i.e., equal to  $1/2$ . The same is valid if one fixes  $\sigma_\tau$ . A kernel defined by  $\sigma_\tau \sigma_\Omega = 1/2$  (optimal in the described way) is

$$c(\theta\tau) = e^{-|\theta\tau|/\sigma} \quad (3.96)$$

since its auto-term function is of the Gaussian form

$$c(\tau, \tau) = c(\tau^2) = e^{-\tau^2/\sigma}.$$

### 3.03.3.5 Kernel decomposition method

Distributions from the Cohen class can be calculated by using decomposition of the kernel function in the time-lag domain, introduced by Amin [40], Cunningham, and Williams. Starting from

$$\text{CD}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c_T(t-u, \tau) x(u + \tau/2) x^*(u - \tau/2) e^{-j\Omega\tau} d\tau du$$

with substitutions  $u + \tau/2 = t + v_1$  and  $u - \tau/2 = t + v_2$  we get  $t - u = -(v_1 + v_2)/2$  and  $\tau = v_1 - v_2$ , resulting in

$$\text{CD}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c_T\left(-\frac{v_1 + v_2}{2}, v_1 - v_2\right) x(t + v_1) x^*(t + v_2) e^{-j\Omega(v_1 - v_2)} dv_1 dv_2.$$

The discrete-time version of the Cohen class of distribution can be written, as

$$\text{CD}(n, \omega) = \sum_{n_1} \sum_{n_2} c_T\left(-\frac{n_1 + n_2}{2}, n_1 - n_2\right) [x(n + n_1) e^{-j\omega n_1}] [x(n + n_2) e^{-j\omega n_2}]^*.$$

Assuming that  $\mathbf{C}$  is a square matrix of finite dimension, with elements:

$$C(n_1, n_2) = c_T\left(-\frac{n_1 + n_2}{2}, n_1 - n_2\right),$$

we can write

$$\text{CD}(n, \omega) = \mathbf{x}_n \mathbf{C} \mathbf{x}_n^H,$$

where  $\mathbf{x}_n$  is a vector with elements  $x(n+n_1)e^{-j\omega n_1}$ . We can now perform the eigenvalue decomposition, finding solutions of  $\det(\mathbf{C} - \lambda\mathbf{I}) = 0$  and determining eigenvectors matrix  $\mathbf{Q}$  that satisfies  $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$  and  $\mathbf{C} = \mathbf{Q}\Lambda\mathbf{Q}^H$ , where  $\Lambda$  is a diagonal matrix containing the eigenvalues. It results in

$$\text{CD}(n, \omega) = (\mathbf{x}_n \mathbf{Q}) \Lambda (\mathbf{x}_n \mathbf{Q})^H.$$

Then, it is easy to conclude that the Cohen class of distribution can be written as a sum of spectrograms:

$$\text{CD}(n, \omega) = \sum_i \lambda_i |\text{STFT}_{\mathbf{q}_i}(n, \omega)|^2,$$

where  $\lambda_i$  represents eigenvalues, while  $\mathbf{q}_i$  are corresponding eigenvectors of  $\mathbf{C}$ , i.e., columns of  $\mathbf{Q}$ , used as windows in the STFT calculations. The eigenvalues and corresponding eigenvectors in the case of Choi-Williams kernel are presented in Figure 3.24.

Alternative decomposition matrix scheme, singular value decomposition, can be applied to the matrix of arbitrary shape.

### 3.03.3.6 S-method

The reduced interference distributions are derived in order to suppress cross-terms, while preserving the marginal properties. Another method is based on the idea of preserving the auto-terms as in the Wigner distribution, with elimination, or significant reduction, of the cross-terms. This method has been derived based on the relationship between the STFT and the pseudo Wigner distribution [23,35].

The pseudo Wigner distribution can be calculated as

$$\text{PWD}(t, \Omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} \text{STFT}(t, \Omega + \theta) \text{STFT}^*(t, \Omega - \theta) d\theta, \quad (3.97)$$

where

$$\text{STFT}(t, \Omega) = \int_{-\infty}^{\infty} x(t + \tau) w(\tau) e^{-j\Omega\tau} d\tau. \quad (3.98)$$

This can be proven by substituting (3.98) into (3.97).

Relation (3.97) has led to the definition of a time-frequency distribution

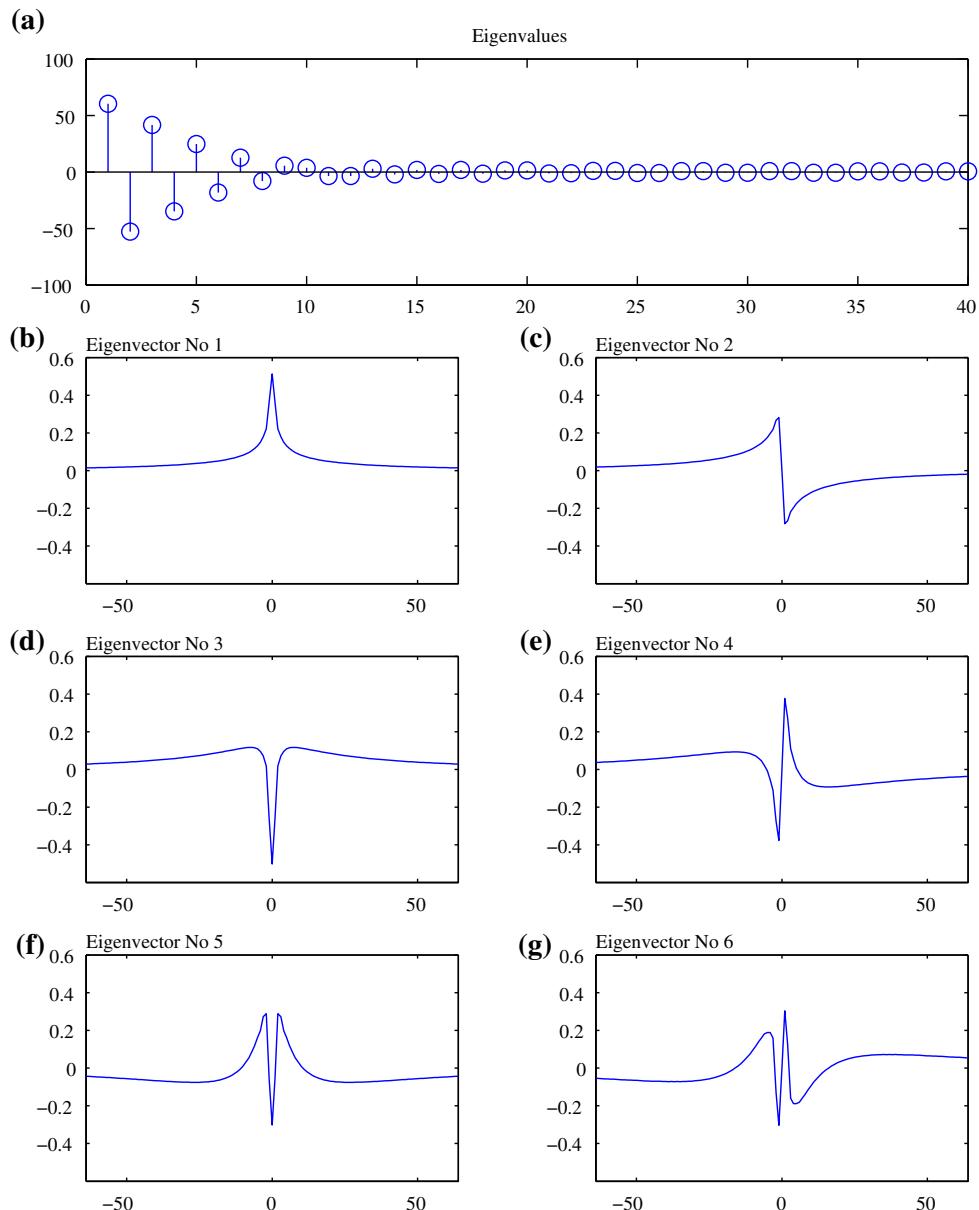
$$\text{SM}(t, \Omega) = \frac{1}{\pi} \int_{-L_P}^{L_P} P(\theta) \text{STFT}(t, \Omega + \theta) \text{STFT}^*(t, \Omega - \theta) d\theta, \quad (3.99)$$

where  $P(\theta)$  is a finite frequency domain window (we also assume rectangular form),  $P(\theta) = 0$  for  $|\theta| > L_P$ . Distribution obtained in this way is referred to as the S-method. Two special cases are: the spectrogram  $P(\theta) = \pi\delta(\theta)$  and the pseudo Wigner distribution  $P(\theta) = 1$ .

The S-method can produce a representation of a multi-component signal such that the distribution of each component is its Wigner distribution, avoiding cross-terms, if the STFTs of the components do not overlap in time-frequency plane.

Consider a signal

$$x(t) = \sum_{m=1}^M x_m(t),$$

**FIGURE 3.24**

Kernel decomposition example for the Choi-Williams distribution kernel: eigenvalues and six eigenvectors corresponding to the highest magnitude eigenvalues.

where  $x_m(t)$  are mono-component signals. Assume that the STFT of each component lies inside the region  $D_m(t, \Omega)$ ,  $m = 1, 2, \dots, M$  and assume that regions  $D_m(t, \Omega)$  do not overlap. Denote the length of the  $m$ th region along  $\Omega$ , for a given  $t$ , by  $2B_m(t)$ , and its central frequency by  $\Omega_{0m}(t)$ . Under this assumptions the  $S$ -method of  $x(t)$  produces the sum of the pseudo Wigner distributions of each signal component

$$\text{SM}_x(t, \Omega) = \sum_{m=1}^M \text{PWD}_{x_m}(t, \Omega), \quad (3.100)$$

if the width of the rectangular window  $P(\theta)$ , for a point  $(t, \Omega)$ , is defined by

$$L_P(t, \Omega) = \begin{cases} B_m(t) - |\Omega - \Omega_{0m}(t)| & \text{for } (t, \Omega) \in D_m(t, \Omega), \\ 0 & \text{elsewhere.} \end{cases}$$

To prove this consider a point  $(t, \Omega)$  inside a region  $D_m(t, \Omega)$ . The integration interval in (3.99), for the  $m$ th signal component is symmetrical with respect to  $\theta = 0$ . It is defined by the smallest absolute value of  $\theta$  for which  $\Omega + \theta$  or  $\Omega - \theta$  falls outside  $D_m(t, \Omega)$ , i.e.,

$$|\Omega \pm \theta - \Omega_{0m}(t)| \geq B_m(t).$$

For  $\Omega > \Omega_{0m}(t)$  and positive  $\theta$ , the integration limit is reached for  $\theta = B_m(t) - (\Omega - \Omega_{0m}(t))$ . For  $\Omega < \Omega_{0m}(t)$  and positive  $\theta$ , the limit is reached for  $\theta = B_m(t) + (\Omega - \Omega_{0m}(t))$ . Thus, having in mind the interval symmetry, an integration limit which produces the same value of integral (3.99) as the value of (3.97), over the region  $D_m(t, \Omega)$ , is given by  $L_P(t, \Omega)$ . Therefore, for  $(t, \Omega) \in D_m(t, \Omega)$  we have  $\text{SM}_x(t, \Omega) = \text{PWD}_{x_m}(t, \Omega)$ . Since regions  $D_m(t, \Omega)$  do not overlap we have

$$\text{SM}_x(t, \Omega) = \sum_{m=1}^M \text{PWD}_{x_m}(t, \Omega).$$

Note that any window  $P(\theta)$  with constant width  $L_P \geq \max_{(t, \Omega)}\{L_P(t, \Omega)\}$  produces  $\text{SM}_x(t, f) = \sum_{m=1}^M \text{PWD}_{x_m}(t, \Omega)$ , if the regions  $D_m(t, \Omega)$ ,  $m = 1, 2, \dots, M$ , are at least  $2L_P$  apart along the frequency axis, i.e.,  $|\Omega_{0p}(t) - \Omega_{0q}(t)| > B_p(t) + B_q(t) + 2L_P$ , for each  $p, q$ , and  $t$ . This is the  $S$ -method with constant window width. The best choice of  $L_P$  is the value when  $P(\theta)$  is wide enough to enable complete integration over the auto-terms, but narrower than the distance between the auto-terms, in order to avoid the cross-terms. *If two components overlap for some time instants  $t$ , then the cross-term will appear, but only between these two components and for that time instants.*

Kernel function of the  $S$ -method is given by  $c(\theta, \tau) = P(\theta/2) *_{\theta} \text{AF}_{ww}(\theta, \tau)/2\pi$ , where  $\text{AF}_{ww}(\theta, \tau)$  is the ambiguity function of window  $w(\tau)$ . It is generally a non-separable function.

### 3.03.3.6.1 Discrete $S$ -method

The discrete form of the  $S$ -method reads

$$\text{SM}(n, k) = \sum_{i=-L_d}^{L_d} P(i) \text{STFT}(n, k+i) \text{STFT}^*(n, k-i), \quad (3.101)$$

$$\text{SM}(n, k) = |\text{STFT}(n, k)|^2 + 2\text{Re} \left[ \sum_{i=1}^{L_d} P(i) \text{STFT}(n, k+i) \text{STFT}^*(n, k-i) \right], \quad (3.102)$$

where  $\text{STFT}(n, k) = \text{DFT}_{i \rightarrow k}\{x(n+i)w(i)\}$ . The terms in summation improve the quality of spectrogram  $|\text{STFT}(n, k)|^2$  toward the Wigner distribution quality.

A recursive relation for the *S*-method calculation with rectangular window  $P(i)$  is

$$\begin{aligned} \text{SM}(n, k; L_d) &= \text{SM}(n, k; L_d - 1) \\ &\quad + 2\text{Re}[\text{STFT}(n, k + L_d) \text{STFT}^*(n, k - L_d)], \end{aligned} \quad (3.103)$$

where  $\text{SM}(n, k; 0) = |\text{STFT}(n, k)|^2$ , and  $\text{SM}(n, k; L_d)$  denotes  $\text{SM}(n, k)$  in (3.102) calculated with  $L_d$  terms in the sum. In this way, we start from the spectrogram, and gradually make the transition toward the pseudo Wigner distribution (see Figure 3.25).

For the *S*-method realization we have to implement the STFT first, based either on the FFT routines or recursive approaches suitable for hardware realizations. After we get the STFT we have to “correct” the obtained values, according to (3.102), by adding few “correction” terms to the spectrogram values. Note that *S*-method is one of the rare quadratic time-frequency distributions allowing easy hardware realization, based on the hardware realization of the STFT, presented in the first part, and its “correction” according to (3.102).

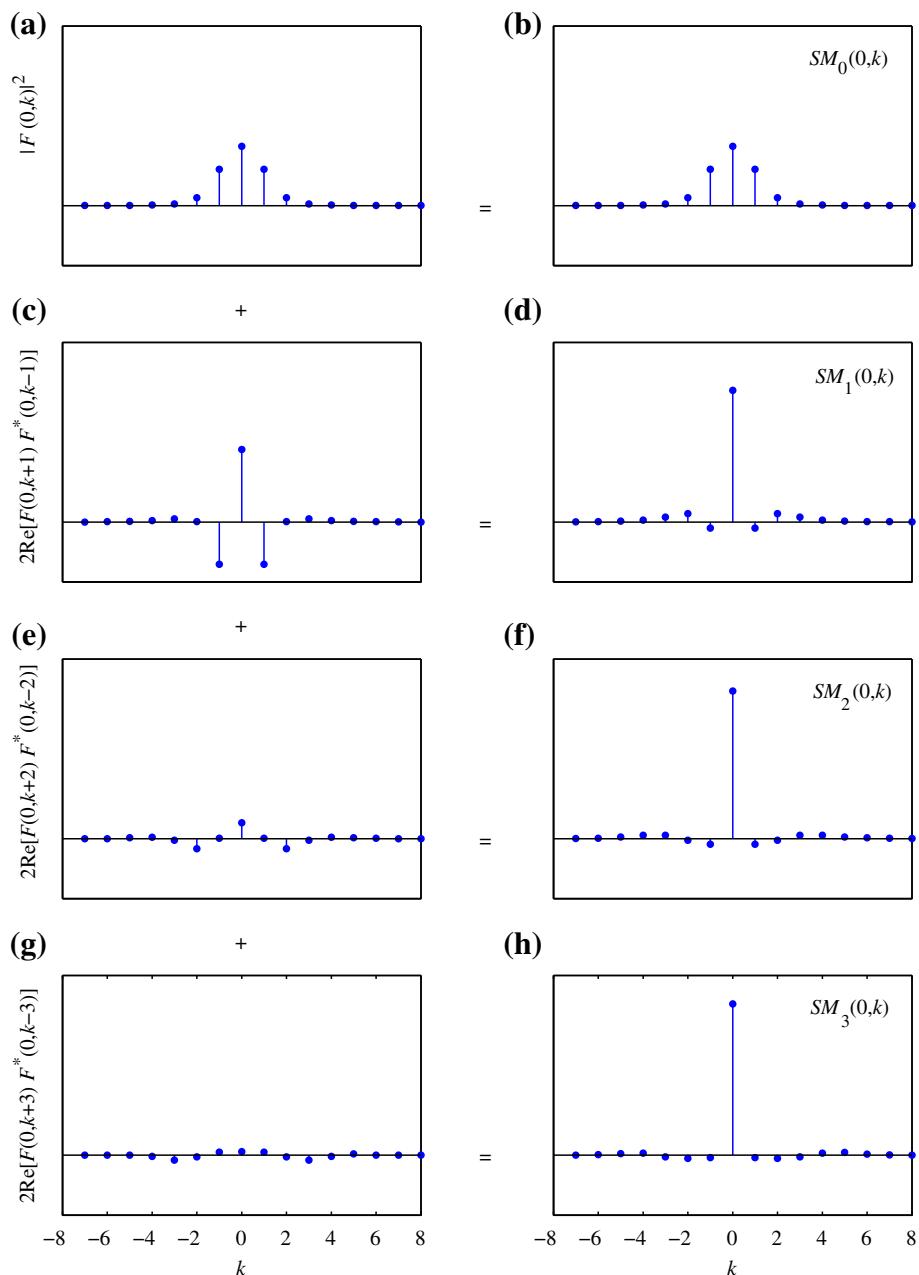
In the *S*-method calculation:

1. There is no need for analytic signal since the cross-terms between negative and positive frequency components are removed in the same way like the other cross-terms.
2. If we take that  $\text{STFT}(n, k) = 0$  outside the basic period, i.e., when  $k < -N/2$  or  $k > N/2 - 1$ , then there is no aliasing when the STFT is alias-free (in this way we can calculate the alias-free Wigner distribution by taking  $L_d = N/2$  and  $P(i) = 1$  in (3.102)).
3. The calculation in (3.102) and (3.103) does not need to be done for each point  $(n, k)$  separately. It can be performed for the whole matrix of the *S*-method and the STFT. This can significantly save time in some matrix based calculation tools.

There are two possibilities to implement the summation in (3.102):

1. With a signal independent  $L_d$ . Theoretically, in order to get the pseudo Wigner distribution for each individual component, we should use rectangular window with length  $2L_d + 1$  such that  $2L_d$  is equal to the width of the widest auto-term. This will guarantee cross-terms free distribution for all components which are at least  $2L_d$  samples apart. For components and time instants where this condition is not satisfied, the cross-terms will appear, but still in a reduced form.
2. With a signal dependent  $L_d = L_d(n, k)$  where the summation, for each point  $(n, k)$ , stops when the absolute square value of  $\text{STFT}(n, k+i)$  or  $\text{STFT}(n, k-i)$  is smaller than an assumed reference level  $R$ . If a zero value may be expected within a single auto-term, then the summation lasts until two subsequent values below reference level are detected. The reference level is defined as a few percent of the spectrogram’s maximal value at a considered instant  $n$

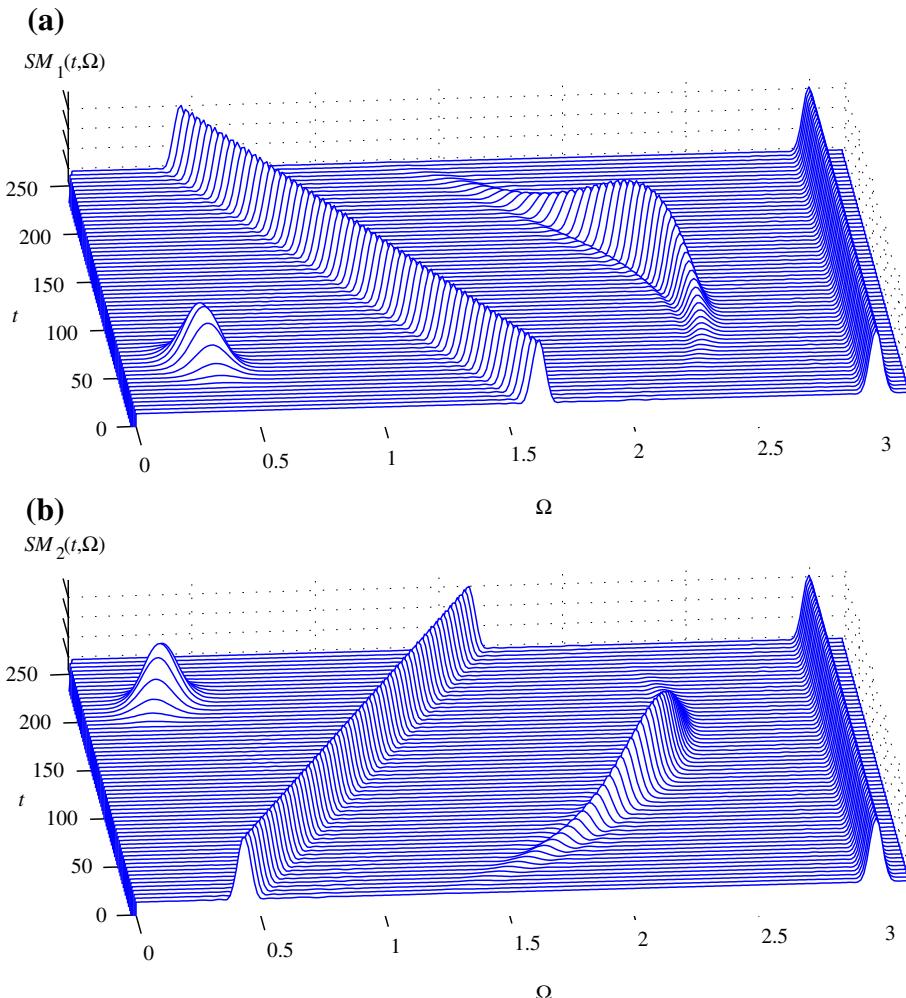
$$R_n = \frac{1}{Q^2} \max_k \{|\text{STFT}(n, k)|^2\},$$

**FIGURE 3.25**

The S-method illustration for 16 point linear frequency modulated signal. Notation  $\text{STFT}(n, k) = F(n, k)$  is used.

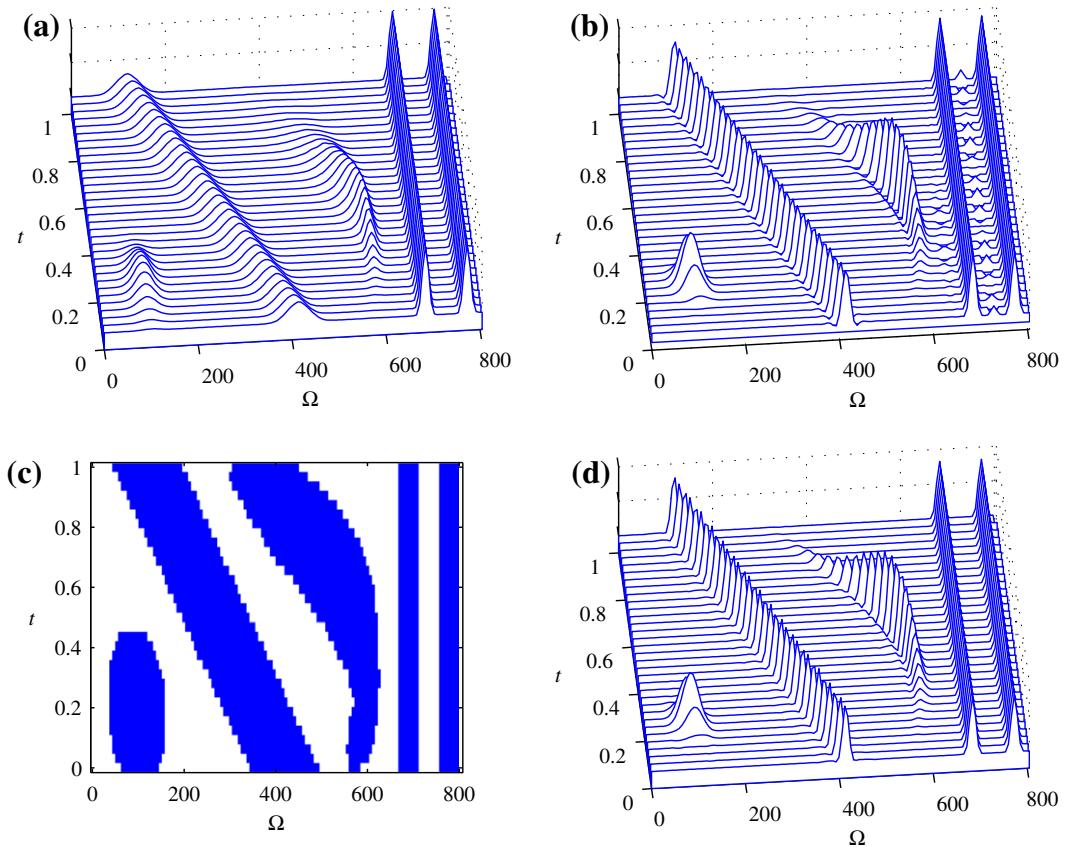
where  $Q \geq 1$  is a constant. Index  $n$  is added to show that the reference level  $R$  is time dependent. Note that if  $Q^2 \rightarrow \infty$ , the Wigner distribution will be obtained, while  $Q^2 = 1$  results in the spectrogram. A choice of an appropriate value for design parameter  $Q$  will be discussed in the next *Examples*. This is also known as adaptive  $S$ -method.

**Example 14.** Consider a real-valued multi-component signals presented in Figure 3.1. The  $S$ -method for  $L_d = 4$  is presented in Figure 3.26.



**FIGURE 3.26**

The  $S$ -method of the signals from Figure 3.1.

**FIGURE 3.27**

Time-frequency analysis of a multi-component signal: (a) The spectrogram. (b) The S-method with a constant window, with  $L_P = 3$ . (c) Regions of support for the S-method with a variable window width calculation, corresponding to  $Q^2 = 725$ . (d) The S-method with the variable window width.

**Example 15.** The adaptive S-method realization will be illustrated on a five-component signal  $x(t)$  defined for  $0 \leq t < 1$  and sampled with  $\Delta t = 1/256$ . The Hamming window of the width  $T_w = 1/2$  (128 samples) is used for STFT calculation. The spectrogram is presented in Figure 3.27a, while the S-method with the constant  $L_d = 3$  is shown in Figure 3.27b. The concentration improvement with respect to the case  $L_d = 0$ , Figure 3.27a, is evident. Further increasing of  $L_d$  would improve the concentration, but the cross-terms would also appear. Small changes are noticeable between the components with constant IF and between quadratic and constant IF component. An improved concentration, without cross-terms, can be achieved by using the variable window width  $L_d$ . The regions  $D_i(n, k)$ , determining the summation limit  $L_d(n, k)$  for each point  $(n, k)$ , are obtained by imposing the reference

level  $R_n$  corresponding to  $Q^2 = 725$ . They are defined as:

$$D_i(n, k) = \begin{cases} 1 & \text{when } |\text{STFT}_{x_i}(n, k)|^2 \geq R_n, \\ 0 & \text{elsewhere,} \end{cases}$$

and presented in Figure 3.27c. White regions mean that the value of spectrogram is below 0.14% of its maximal value at that time instant  $n$ , meaning that the concentration improvement is not performed at these points. The signal dependent S-method is given in Figure 3.27d. The method sensitivity, with respect to the value of  $Q^2$ , is low.

### 3.03.3.6.2 Smoothed spectrogram versus S-method

The S-method belongs to the general class of quadratic time-frequency distributions. Let us consider the general form

$$\text{CD}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c_T \left( -\frac{v_1 + v_2}{2}, v_1 - v_2 \right) x(t + v_1) x^*(t + v_2) e^{-j\Omega(v_1 - v_2)} dv_1 dv_2$$

and write it as

$$\text{CD}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} [x(t + v_1) e^{-j\Omega v_1}] G_T(v_1, v_2) [x(t + v_2) e^{-j\Omega v_2}]^* dv_1 dv_2.$$

If the inner product kernel  $G_T(v_1, v_2)$  is factorized in the Hankel form  $G_T(v_1, v_2) = 2w(v_1)p(v_1 + v_2)w(v_2)$ , then, by substituting its value into the previous relation, the S-method follows. The Toeplitz factorization of the kernel  $G_T(v_1, v_2) = 2w(v_1)p(v_1 - v_2)w(v_2)$  results in the smoothed spectrogram. The smoothed spectrogram composes two STFTs in the same direction,

$$\text{SSPEC}(n, k) = \sum_{i=-L_d}^{L_d} P(i) \text{STFT}(n, k + i) \text{STFT}^*(n, k + i),$$

resulting in the distribution spread, in contrast to the S-method, where two STFTs are composed in counterdirection,

$$\text{SM}(n, k) = \sum_{i=-L_d}^{L_d} P(i) \text{STFT}(n, k + i) \text{STFT}^*(n, k - i).$$

These forms led Scharf and Friedlander to divide all the estimators of discrete time-varying processes into two classes, one smoothed spectrogram based and the other S-method based [41].

In this sense we may conclude that it is possible to define various forms of the S-method and use it for various applications. It has been done in literature, for example, for the time-direction form of the S-method, for the S-method composed in both time and frequency (two dimensional S-method), for composing the windowed fractional Fourier transforms (fractional form of the S-method), and composing wavelets and other time-scale transforms (affine form of the S-method).

### 3.03.3.6.3 Decomposition of multi-component signals

Let us consider a multi-component signal

$$x(n) = \sum_{i=1}^M x_i(n),$$

where components  $x_i(n)$  are mutually orthogonal, i.e., the components do not overlap in the time-frequency plane.

For each signal component  $x_i(n)$  we can write its inversion formula, corresponding to (3.73), as

$$x_i(n_1)x_i^*(n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{WD}_i\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)},$$

$$i = 1, 2, \dots, M,$$

if the Wigner distribution  $\text{WD}_i(n, k)$  of this component were known. By summing the above relations for  $i = 1, 2, \dots, M$  we get

$$\sum_{i=1}^M x_i(n_1)x_i^*(n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \sum_{i=1}^M \text{WD}_i\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)}.$$

Having in mind (3.100), for the signals that satisfy the presented conditions, this relation reduces to:

$$\sum_{i=1}^M x_i(n_1)x_i^*(n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{SM}\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)}. \quad (3.104)$$

By denoting

$$R_{\text{SM}}(n_1, n_2) = \frac{1}{N+1} \sum_{k=-N/2}^{N/2} \text{SM}\left(\frac{n_1+n_2}{2}, k\right) e^{j\frac{2\pi}{N+1}k(n_1-n_2)} \quad (3.105)$$

and using the eigenvalue decomposition of matrix  $\mathbf{R}_{\text{SM}}$ , with the elements  $R_{\text{SM}}(n_1, n_2)$ , we get

$$\mathbf{R}_{\text{SM}} = \sum_{i=1}^{N+1} \lambda_i \mathbf{q}_i(n) \mathbf{q}_i^*(n).$$

As in the case of the Wigner distribution, we can conclude that  $\lambda_i = E_{x_i}$ ,  $i = 1, 2, \dots, M$ , and  $\lambda_i = 0$  for  $i = M+1, \dots, N$ .

The eigenvector  $\mathbf{q}_i(n)$  will be equal to the signal component  $\mathbf{x}_i(n)$ , up to the phase and amplitude constants, since the components orthogonality is assumed. Amplitude constants are again contained in the eigenvalues  $\lambda_i$ . Thus, the reconstructed signal can be written as

$$x_{\text{rec}}(n) = \sum_{i=1}^M \sqrt{\lambda_i} q_i(n).$$

It is equal to the original signal, up to the phase constants in each component. When we have several components of different energies  $x_1(n), x_2(n), \dots, x_M(n)$  and when they are of equal importance in

analysis, we can use normalized values of the signal components and calculate the time-frequency representation of

$$x_{\text{nor}}(n) = \sum_{i=1}^M k(\lambda_i) q_i(n)$$

by using the weights  $k(\lambda_i) = 1$  in the signal, i.e., when  $i = 1, 2, \dots, M$ .

#### *Illustrative example*

Consider a signal whose analog form reads

$$x(t) = e^{j\frac{\pi}{3200}t^2} e^{-(\frac{t}{96})^2} + \sum_{k=2}^5 \sqrt{\frac{26-k}{10}} e^{j\Omega_k t} e^{-\left(\frac{t-d_k}{16}\right)^2}$$

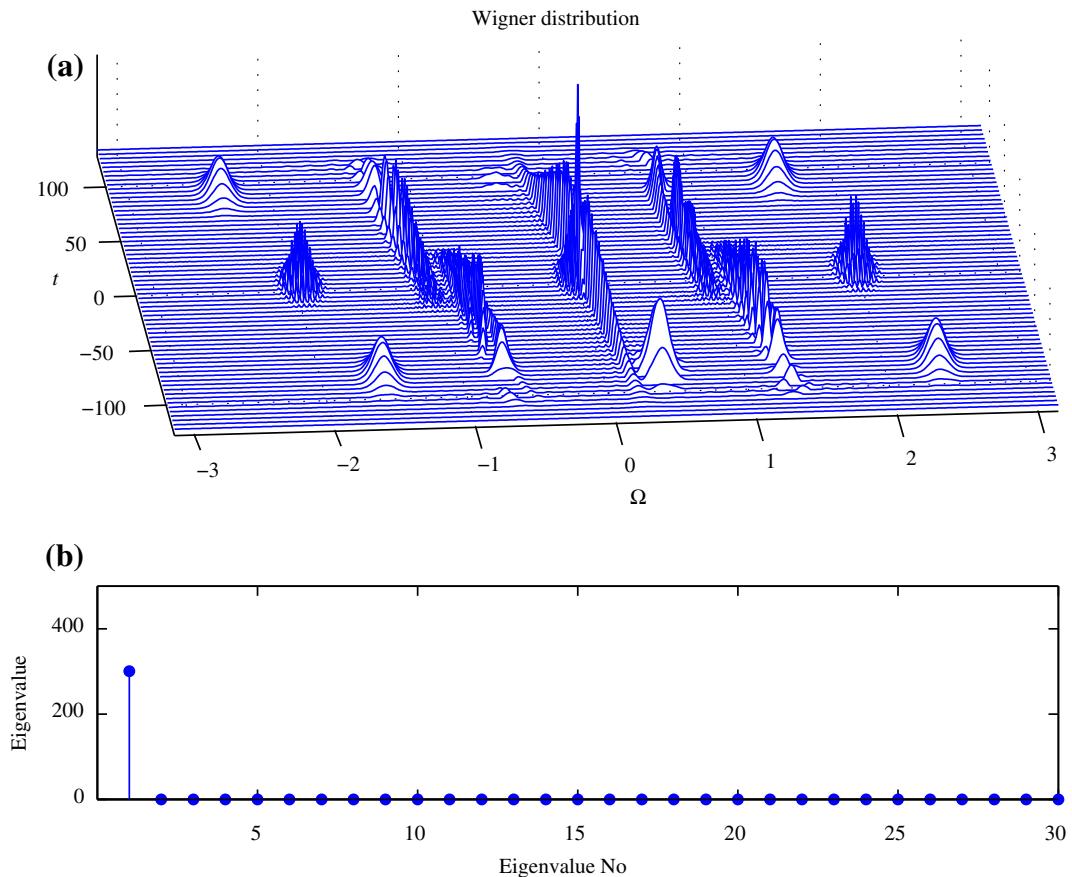
within the interval  $-128 \leq t \leq 128$ , where  $\Omega_2 = -\frac{3\pi}{4}$ ,  $\Omega_3 = -\frac{\pi}{2}$ ,  $\Omega_4 = \frac{\pi}{2}$ ,  $\Omega_5 = \frac{3\pi}{4}$ ,  $d_2 = d_4 = -85$ , and  $d_3 = d_5 = 85$ . The sampling interval is  $\Delta t = 1$ . The Wigner distribution is presented in Figure 3.28, upper subplot. Based on the Wigner distribution, the elements of matrix  $\mathbf{R}$  are calculated by using (3.74). Eigenvalue decomposition (3.77) of this matrix produces exactly one non-zero eigenvalue,  $\lambda_1 = 299.7$  ( $\lambda_2 = 0.00$ ,  $\lambda_3 = 0.00$ , ...), being equal (up to the numerical error) to the total signal energy  $E_x = 299.9$ , as expected from (3.74)–(3.78).

The *S*-method of the same signal is calculated by using (3.101) with  $L = 12$ . The obtained results are depicted in Figure 3.29. Matrix  $\mathbf{R}_{\text{SM}}$  is formed according to (3.105). Its eigenvalue decomposition results in the same number of non-zero eigenvalues as the number of signal components. Eigenvalues correspond to the components energies, while the eigenvectors correspond to the normalized signal components, up to the phase constants. First seven components correspond to the signal, while the remaining ones are with very small eigenvalues. Energies of discrete signal components are:  $E_1 = 119.40$ ,  $E_2 = 48.12$ ,  $E_3 = 46.12$ ,  $E_4 = 44.12$ , and  $E_5 = 42.11$ , while the obtained eigenvalues by using the *S*-method with  $L = 12$  are:  $\lambda_1 = 117.42$ ,  $\lambda_2 = 47.99$ ,  $\lambda_3 = 45.99$ ,  $\lambda_4 = 43.99$ ,  $\lambda_5 = 41.99$ ,  $\lambda_6 = 8.26$ , ...

Sensitivity of the results with respect to  $L$  is quite low within a wide region. We have repeated calculations with values of  $L$  from  $L = 10$  up to  $L = 20$  and obtained almost the same results. The error in components energy, estimated by corresponding eigenvalues, was within  $\pm 0.25\%$ .

A similar procedure can be used for signal synthesis from the given time-frequency distribution function  $D(n, k)$ . We should calculate  $\mathbf{R}_D$  matrix by substituting  $D(n, k)$  instead of  $\text{SM}(n, k)$  in (3.105) and calculate corresponding eigenvalues. If we obtain single non-zero eigenvalue then there exists a signal  $x(n)$  such that  $D(n, k)$  is its Wigner distribution. In the case when  $M$  non-zero eigenvalues is present ( $M > 1$ ), the function  $D(n, k)$  can be approximated as sum of Wigner distribution of several components. The approximation error can be estimated as a sum of the remaining eigenvalues ( $\lambda_{M+1} + \lambda_{M+2} + \dots$ ).

**Example 16.** A two dimensional function  $D(n, k)$ , representing the desired time-frequency distributions of signal energy, is given in Figure 3.30a. It consists of seven time-frequency regions. We will now find a seven component signals, such that the Wigner distribution of each component is the mean squared estimation of each desired region. Performing the decomposition approach, by using the *S*-method as an approximation of the Wigner distributions of individual components, we get the eigenvalues (Figure 3.30b) with corresponding eigenvectors. Keeping the largest seven eigenvalues,

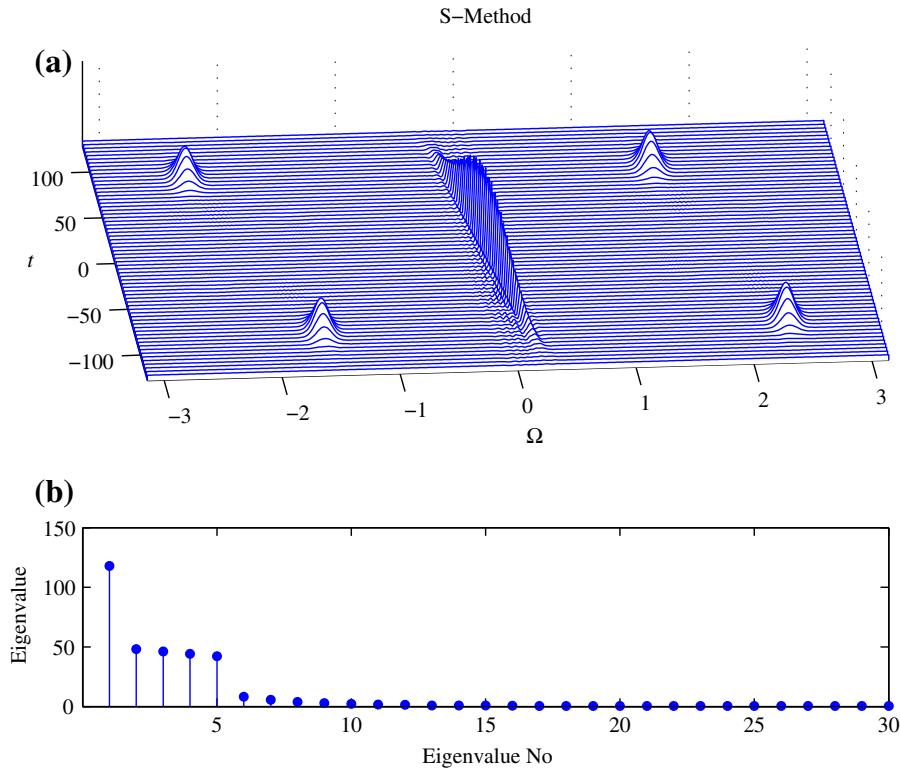
**FIGURE 3.28**

Decomposition of the Wigner distribution. Only one non-zero eigenvalue is obtained.

with corresponding eigenvectors, we form a signal that is best time-frequency approximation of the desired arbitrary function form Figure 3.30. The S-method, as a time-frequency representation, of the synthesized signal is shown in Figure 3.31.

### 3.03.3.7 Reassignment in time-frequency

The reassignment method is an approach for post-processing of the time-frequency representations. It was originally introduced by Kodera et al. to improve the readability of the spectrogram by using the phase information in the STFT to relocate (reassign) the distribution values closer to the instantaneous frequency or group delay.

**FIGURE 3.29**

Decomposition of the S-method. Number of non-zero eigenvalues coincide with number of the signal components.

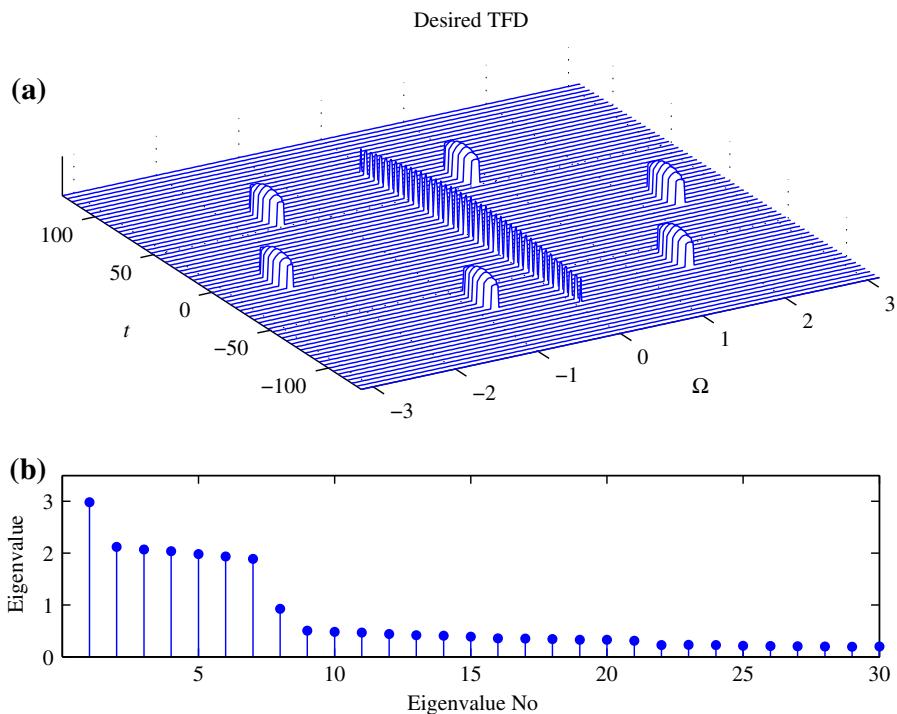
In order to explain the principle of reassignment let us consider the STFT definition, that we used in this book,

$$\begin{aligned} \text{STFT}_w(t, \Omega) &= \int_{-\infty}^{\infty} w(\tau)x(t + \tau)e^{-j\Omega\tau} d\tau \\ &= e^{j\Omega t} \int_{-\infty}^{\infty} w(\tau - t)x(\tau)e^{-j\Omega\tau} d\tau = |\text{STFT}(t, \Omega)| e^{j\Psi(t, \Omega)}. \end{aligned}$$

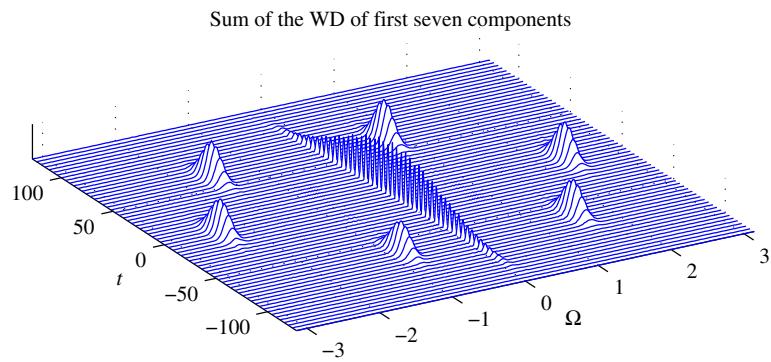
It may be understood as decomposing a localized signal  $x_t(\tau) = w(\tau)x(t + \tau)$  into the periodic functions  $e^{-j\Omega\tau}$ . Here index  $w$  in the  $\text{STFT}_w(t, \Omega)$  indicates that a window  $w(\tau)$  is used for the STFT calculation.

The signal can be reconstructed by

$$x(t) = \frac{1}{2\pi E_w} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{STFT}_w(v, \Omega)w(t - v)e^{-j\Omega(v-t)} d\Omega dv, \quad (3.106)$$

**FIGURE 3.30**

Desired time-frequency distribution and corresponding eigenvalues.

**FIGURE 3.31**

Resulting time-frequency distribution after signal synthesis from the most significant components.

where  $E_w$  is a window energy, since

$$\begin{aligned} & \frac{1}{2\pi E_w} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{j\Omega v} w(u-t) x(u) e^{-j\Omega u} w(t-v) e^{-j\Omega(v-t)} du d\Omega dv \\ &= \frac{1}{E_w} \int_{-\infty}^{\infty} w(0) x(t) w(t-v) dv = x(t). \end{aligned}$$

Relation (3.106) can be written as

$$x(t) = \frac{1}{2\pi E_w} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |\text{STFT}_w(v, \Omega)| w(t-v) e^{j(\Psi(v, \Omega) - \Omega v + \Omega t)} d\Omega dv. \quad (3.107)$$

According to the stationary phase method, the most significant contribution to the reconstructed value of  $x(t)$  is from the stationary point of the phase  $\Psi(v, \Omega) - \Omega v + \Omega t$ , in time and in frequency. The stationary phase point is obtained from the phase derivatives in the corresponding directions,

$$\begin{aligned} \frac{\partial \Psi(v, \Omega)}{\partial \Omega} - (v - t) &= 0, \\ \frac{\partial \Psi(v, \Omega)}{\partial v} - \Omega &= 0. \end{aligned} \quad (3.108)$$

It means that the calculated distribution value (in this case the spectrogram value) should be assigned not to the point  $(t, \Omega)$  where it is calculated, but to the point where it contributes the most to the signal reconstruction, according to the stationary phase principle. The shifts of the calculated values are

$$\begin{aligned} \hat{t}(t, \Omega) &= t - \frac{\partial \Psi(t, \Omega)}{\partial \Omega}, \\ \hat{\Omega}(t, \Omega) &= \Omega - \frac{\partial \Psi(t, \Omega)}{\partial t}. \end{aligned} \quad (3.109)$$

Note that the crucial role here is played by the STFT phase function, that is ignored in the spectrogram calculation and presentation.

In the case of the spectrogram, the reassigning shifts, are obtained as

$$\hat{t}(t, \Omega) = t + \text{Re} \left\{ \frac{\text{STFT}_{\tau w}(t, \Omega) \text{STFT}_w^*(t, \Omega)}{|\text{STFT}_w(t, \Omega)|^2} \right\}, \quad (3.110)$$

$$\hat{\Omega}(t, \Omega) = \Omega - \text{Im} \left\{ \frac{\text{STFT}_{Dw}(t, \Omega) \text{STFT}_w^*(t, \Omega)}{|\text{STFT}_w(t, \Omega)|^2} \right\}, \quad (3.111)$$

where  $\text{STFT}_{\tau w}(t, \Omega)$  and  $\text{STFT}_{Dw}(t, \Omega)$  are the STFTs calculated with windows  $\tau w(\tau)$  and  $Dw(\tau)/d\tau$ , respectively. For  $|\text{STFT}_w(t, \Omega)|^2 = 0$  there is nothing to reassign, so the expressions (3.110) are not used.

To prove this, rewrite

$$|\text{STFT}_w(t, \Omega)| e^{j\Psi(t, \Omega)} = \int_{-\infty}^{\infty} w(\tau) x(t+\tau) e^{-j\Omega\tau} d\tau.$$

Calculation of the  $\text{STFT}_{\tau w}(t, \Omega)$ , with  $\tau w(\tau)$  as the window function, corresponds to the derivative over  $\Omega$  of both sides of the previous equation. It results in

$$\begin{aligned} & \int_{-\infty}^{\infty} \tau w(\tau) x(t + \tau) e^{-\Omega \tau} d\tau \\ &= j \frac{\partial |\text{STFT}_w(t, \Omega)|}{\partial \Omega} e^{j\Psi(t, \Omega)} - \frac{\partial \Psi(v, \Omega)}{\partial \Omega} |\text{STFT}_w(t, \Omega)| e^{j\Psi(t, \Omega)}. \end{aligned}$$

Thus,

$$\begin{aligned} & \text{STFT}_{\tau w}(t, \Omega) \text{STFT}_w^*(t, \Omega) \\ &= j \frac{\partial |\text{STFT}_w(t, \Omega)|}{\partial \Omega} |\text{STFT}_w(t, \Omega)| - \frac{\partial \Psi(v, \Omega)}{\partial \Omega} |\text{STFT}_w(t, \Omega)|^2 \end{aligned}$$

with

$$\text{Re} \{ \text{STFT}_{\tau w}(t, \Omega) \text{STFT}_w^*(t, \Omega) \} = - \frac{\partial \Psi(v, \Omega)}{\partial \Omega} |\text{STFT}_w(t, \Omega)|^2 \quad (3.112)$$

producing the reassignment shift in time in (3.110).

In a similar way, using the frequency domain definitions of the STFT we obtain the reassignment shift in frequency.

The previous procedure, stating that a value of the spectrogram  $\text{SPEC}(t, \Omega)$  should not be placed at  $(t, \Omega)$  in the time-frequency plane but should be reassigned to the new positions  $\hat{t}(t, \Omega)$  and  $\hat{\Omega}(t, \Omega)$ , results in the reassigned spectrogram:

$$\text{SPEC}_{\text{reassign}}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{SPEC}(u, v) \delta(t - \hat{t}(u, v)) \delta(\Omega - \hat{\Omega}(u, v)) du dv. \quad (3.113)$$

The reassigned form of a distribution from the Cohen class  $\text{CD}(t, \Omega)$ , introduced by Flandrin et al. is defined by [42]

$$\text{RTF}(t, \Omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{CD}(u, v) \delta(t - \hat{t}(u, v)) \delta(\Omega - \hat{\Omega}(u, v)) du dv,$$

where  $\hat{t}(u, v)$  and  $\hat{\Omega}(u, v)$  are time and frequency displacements defined respectively as:

$$\begin{aligned} \hat{t}(t, \Omega) &= t - \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u \Pi(u, v) \text{WD}(t - u, \Omega - v) du dv}{\text{CD}(t, \Omega)}, \\ \hat{\Omega}(t, \Omega) &= \Omega - \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} v \Pi(u, v) \text{WD}(t - u, \Omega - v) du dv}{\text{CD}(t, \Omega)}. \end{aligned}$$

A reassigned distribution can be understood as the one with assigned values of the basic time-frequency representation to a center of gravity in the considered region.

The reassigned representations satisfy the following important properties.

1. *Time-frequency shift:* For a signal shifted in time and frequency  $y(t) = x(t - t_0)e^{j\Omega_0 t}$  the reassigned representation is shifted version of the reassigned distribution of the original signal  $x(t)$ :  $\text{RTF}_y(t, \Omega) = \text{RTF}_x(t - t_0, \Omega - \Omega_0)$ .

2. *Energy marginal:* For basic time-frequency representation satisfying

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pi(t, v) \Omega = 1,$$

the reassigned distribution satisfies the energy property:

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{RTF}(t, \Omega) dt d\Omega = \int_{-\infty}^{\infty} |x(t)|^2 dt.$$

3. *Reassigned representation is ideally concentrated for linear FM signal and delta pulse:* For linear FM signal  $x(t) = A \exp(jat^2/2 + jbt)$  the frequency displacement is  $\hat{\Omega}(t, \Omega) = b + a\hat{t}(t, \Omega)$ . Then, reassigned representation is

$$\begin{aligned} \text{RTF}(t, \Omega) &= \left( \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{CD}(u, v) \delta(t - \hat{t}(u, v)) du dv \right) \delta(\Omega - at - b) \\ &= G(t, \Omega) \delta(\Omega - at - b). \end{aligned}$$

For delta pulse  $x(t) = A\delta(t - t_0)$  the time displacement is  $\hat{t}(t, \Omega) = t_0$ . Reassigned representation is

$$\text{RTF}(t, \Omega) = \left( \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{CD}(u, v) \delta(t - \hat{\Omega}(u, v)) du dv \right) \delta(t - t_0).$$

4. *Reassigned Wigner distribution is the Wigner distribution itself:* Namely, for the Wigner distribution the time-frequency kernel is  $\Pi(t, \Omega) = 2\pi\delta(t)\delta(\Omega)$ . Then displacements are

$$\begin{aligned} \hat{t}(t, \Omega) &= t - \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u \Pi(u, v) \text{WD}(t-u, \Omega-v) du dv}{\text{WD}(t, \Omega)} = t, \\ \hat{\Omega}(t, \Omega) &= \Omega. \end{aligned}$$

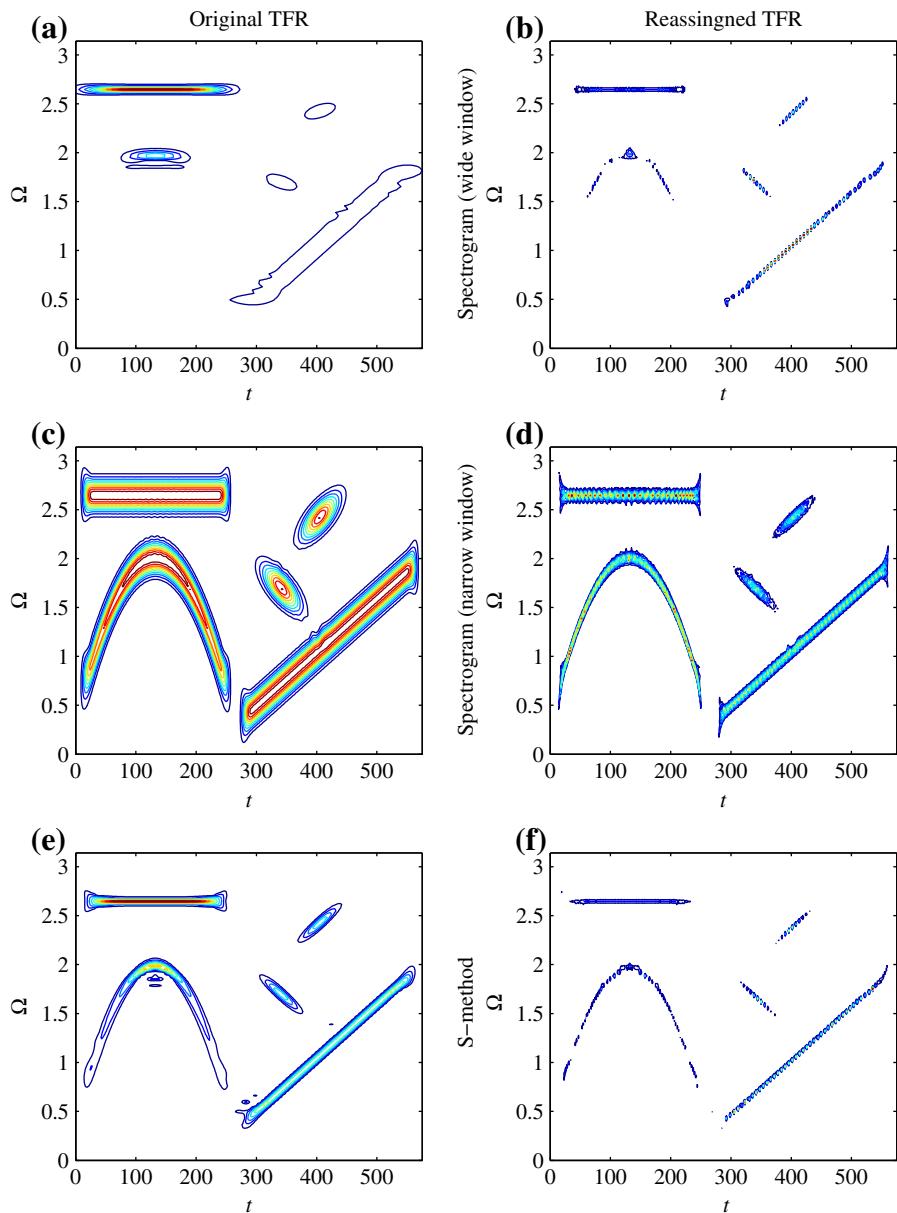
Substituting these displacement into the reassignment method, definition the Wigner distribution easily follows.

The reassigned version of spectrogram for wide and narrow analysis window along with the reassigned  $S$ -method are presented in Figure 3.32.

### 3.03.3.8 Affine class of time-frequency representations

Time-scale distributions or time-frequency representations that are covariant to scale changes and time translations,

$$\begin{aligned} y(t) &= \frac{1}{\sqrt{|a|}} x \left( \frac{\tau - t}{a} \right), \\ \text{TFD}_y(t, \Omega) &= \text{TFD}_x \left( \frac{\tau - t}{a}, \Omega a \right) \end{aligned}$$

**FIGURE 3.32**

The spectrogram with a wide window (a) and the reassigned spectrogram with wide window (b). The spectrogram with a narrow window (c) and the reassigned spectrogram with a narrow window (d). The S-method (e) and the reassigned S-method (f).

belong to the affine class of distributions. Representations from the affine class may be written in the forms similar to the Cohen class of distributions. The simplest time-scale representation is the continuous wavelet transform. It is a linear expansion of the signal onto a set of analyzing functions. However, in time-frequency analysis applications, the resolution of this transform limits its applications.

In order to improve concentration and to satisfy some other desirable properties of a time-frequency representation the quadratic affine distributions are introduced [14, 29–31, 35]. They can be expressed as a function of any time-frequency distribution (as in the Cohen class of time-frequency distributions). Taking the Wigner distribution as the central one, we can write

$$\begin{aligned} \text{AD}(t, \Omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{AF}(\tau, \theta) c(\Omega\tau, \theta/\Omega) e^{-j\theta t} d\theta d\tau \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x\left(u + \frac{\tau}{2}\right) x\left(u + \frac{\tau}{2}\right) c(\Omega\tau, \theta/\Omega) e^{ju\theta} e^{-j\theta t} d\theta d\tau du, \\ \text{AD}(t, \Omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{WD}(\lambda, \mu) \Pi(\Omega(\lambda - t), \mu/\Omega) d\lambda d\mu. \end{aligned}$$

The scalogram and the affine Wigner distributions belong to the affine class. Note that the scalogram in this sense is the Wigner distribution smoothed by the Wigner distribution of the basis function

$$\text{AD}(t, \Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \text{WD}(\lambda, \mu) \text{WD}_{\psi}(\Omega(\lambda - t), \mu/\Omega) d\lambda d\mu.$$

Because of the scale covariance property, many time-frequency representations in the Affine class exhibit constant- $Q$  behavior, permitting multi-resolution analysis.

The time-scale pseudo Wigner distribution is defined by

$$\text{WDT}(t, a) = \int_{-\infty}^{\infty} w_0(\tau/a) x\left(t + \frac{\tau}{2}\right) x^*\left(t - \frac{\tau}{2}\right) e^{j\Omega_0 \tau/a} d\tau$$

The presented Wigner distribution definition means that the local autocorrelation function  $r(x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}))$  is expanded onto the basis functions

$$h\left(\frac{\tau}{a}\right) = w\left(\frac{\tau}{a}\right) e^{j\Omega_0 \tau/a}.$$

Pseudo affine Wigner distribution can be calculated by using the  $S$ -method with reduced interferences.

Continuous WT can be written as  $\text{WT}(t, \Omega) = \int_{-\infty}^{\infty} x(\tau) h^*((\tau - t)\Omega/\Omega_0) d\tau / \sqrt{|\Omega_0/\Omega|}$ . Here, we used frequency instead of scale  $a = \Omega_0/\Omega$ . Consider again  $h(t)$  in the form  $h(t) = w(t) \exp(j\Omega_0 t)$ . Then, the pseudo affine Wigner distribution may be written as

$$\text{WDT}(t, \Omega) = \int_{-\infty}^{\infty} w\left(\frac{\tau}{2\Omega_0}\Omega\right) w\left(-\frac{\tau}{2\Omega_0}\Omega\right) x\left(t + \frac{\tau}{2}\right) x^*\left(t - \frac{\tau}{2}\right) e^{-j\Omega\tau} d\tau. \quad (3.114)$$

The affine  $S$ -method form reads:

$$\text{SM}(t, \Omega) = 2 \int_{-\infty}^{\infty} P(\theta) \text{WT}(t, \Omega; \Omega_0 + \theta) \text{WT}^*(t, \Omega; \Omega_0 - \theta) d\theta, \quad (3.115)$$

where  $\text{WT}(t, \Omega; \Omega_0 + \theta)$  is the WT calculated with  $h(t) = w(t) \exp(j2\pi(\Omega_0 + \theta)t)$ . If  $P(\theta) = \delta(\theta)/2$ , then  $\text{SM}(t, \Omega)$  is equal to the scalogram of  $x(t)$ , while for  $P(\theta) = 1$  it produces  $\text{WDT}(t, \Omega)$  defined by (3.114). This form of the  $S$ -method has been extended to other time-scale representations.

### 3.03.4 Higher order time-frequency representations

A higher order spectral analysis have found its applications in many fields during the last decades: radars, sonars, biomedicine, plasma physics, seismic data processing, image reconstruction, time-delay estimation, adaptive filtering, etc. Higher order statistics, known as cumulants, and its Fourier transforms, known as higher order spectra (polyspectra), are the basic forms in this analysis. Based on these forms, higher order time-varying spectra are introduced and analyzed. The basic representation in the time-varying higher order spectral analysis is the Wigner higher order spectra. After presenting full forms of higher order time-varying spectra, higher order forms reduced to the two-dimensional time-frequency plane are analyzed. The main representatives are the  $L$ -Wigner distributions (sliced version of the higher order spectra) and the polynomial Wigner-Ville distributions (a projection of higher order spectra). Due to slicing or projecting operation, these distributions will loose some of the basic properties of the higher order spectra, but will be able to enhance some other desirable properties for non-stationary signal analysis. A highly concentrated distribution based on complex argument function is presented, as well. The higher order ambiguity function analysis, as a tool for higher order polynomial frequency modulated signals, concludes this section.

#### 3.03.4.1 Wigner bispectrum

The third order moment  $m_3^x(\tau_1, \tau_2)$  is given by:

$$m_3^x(\tau_1, \tau_2) = E[x^*(t)x(t + \tau_1)x(t + \tau_2)].$$

Without loss of generality, we have assumed that  $m_1^x = 0$ , when the third order cumulant is equal to the third order moment  $c_3^x(\tau_1, \tau_2) = m_3^x(\tau_1, \tau_2)$ .

The two-dimensional FT of  $c_3^x(\tau_1, \tau_2)$  is called bispectrum:

$$B(\Omega_1, \Omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c_3^x(\tau_1, \tau_2) e^{-j(\Omega_1 \tau_1 + \Omega_2 \tau_2)} d\tau_1 d\tau_2. \quad (3.116)$$

For deterministic non-stationary signals, replacing  $E[x^*(t + \alpha)x(t + \tau_1 + \alpha)x(t + \tau_2 + \alpha)]$  with the local autocorrelation function  $R_2(t, \tau_1, \tau_2) = x^*(t + \alpha)x(t + \tau_1 + \alpha)x(t + \tau_2 + \alpha)$  we arrive at the Wigner bispectrum (WB):

$$\text{WB}(t, \Omega_1, \Omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^*(t + \alpha)x(t + \tau_1 + \alpha)x(t + \tau_2 + \alpha)e^{-j(\Omega_1 \tau_1 + \Omega_2 \tau_2)} d\tau_1 d\tau_2, \quad (3.117)$$

where the value of  $\alpha = -\frac{\tau_1}{3} - \frac{\tau_2}{3}$  is chosen such that the mean value of the signal's arguments in the above integral is equal to  $t$ . The Wigner bispectrum was introduced by Gerr. In terms of the signal's FT

it reads

$$\text{WB}(t, \Omega_1, \Omega_2) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X^* \left( \Omega_1 + \Omega_2 + \frac{\theta}{3} \right) X \left( \Omega_1 - \frac{\theta}{3} \right) X \left( \Omega_2 - \frac{\theta}{3} \right) e^{-j\theta t} d\theta.$$

Its marginal over time is the bispectrum (of a deterministic signal):

$$\int_{-\infty}^{\infty} \text{WB}(t, \Omega_1, \Omega_2) dt = X^*(\Omega_1 + \Omega_2) X(\Omega_1) X(\Omega_2) = B(\Omega_1, \Omega_2).$$

For a sinusoidal signal,  $x(t) = \exp(j\Omega_0 t)$  with  $X(\Omega) = 2\pi\delta(\Omega - \Omega_0)$ , bispectrum is zero for all frequencies,  $B(\Omega_1, \Omega_2) \equiv 0$ . Bispectrum will produce a peak in frequency-frequency domain in the case of a quadratic phase coupled signals (resulting when a sum of sinusoids passes through a non-linear system, like  $y(t) = x^2(t)$ ). For example, for a signal of the form  $\exp(j\Omega_0 t) + \exp(j2\Omega_0 t)$  there will be a non-zero value (a peak) in the bispectrum,  $B(\Omega_1, \Omega_2) = (2\pi)^3 \delta(\Omega_1 - \Omega_0) \delta(\Omega_2 - \Omega_0) \delta(\Omega_1 + \Omega_2 - 2\Omega_0)$  for  $\Omega_1 = \Omega_0$  and  $\Omega_2 = \Omega_0$ . A similar situation will appear for  $\exp(j\Omega_{01} t) + \exp(j\Omega_{02} t) + \exp(j(\Omega_{01} + \Omega_{02}) t)$ , indicating that a coupling of two sinusoidal signals with frequencies  $\Omega_{01}$  and  $\Omega_{02}$  has occurred.

The Wigner bispectrum behaves differently. For a sinusoidal signal,  $x(t) = \exp(j\Omega_0 t)$ , it will always produce a non-zero value, due to the varying  $\theta$ . Here we have

$$\begin{aligned} \text{WB}(t, \Omega_1, \Omega_2) &= (2\pi)^2 \int_{-\infty}^{\infty} \delta \left( \Omega_1 + \Omega_2 + \frac{\theta}{3} - \Omega_0 \right) \delta \left( \Omega_1 - \frac{\theta}{3} - \Omega_0 \right) \\ &\quad \times \delta \left( \Omega_2 - \frac{\theta}{3} - \Omega_0 \right) e^{-j\theta t} d\theta. \end{aligned}$$

The peak value of the Wigner bispectrum is at  $(\Omega_1, \Omega_2) = (2\Omega_0/3, 2\Omega_0/3)$  for  $\theta/3 = -\Omega_0/3$ . Note that this value is obtained with a shift of frequency lag for  $\theta/3 = -\Omega_0/3$ , which is signal dependent.

### 3.03.4.2 Wigner higher order spectra

Following the idea of (3.117), the Wigner higher order spectra of order  $k$ , of a deterministic signal  $x(t)$ , are defined as the  $k$ -dimensional FT of the local autocorrelation function by Fonolosa and Nikias, as:

$$\begin{aligned} W_k(t, \Omega_1, \Omega_2, \dots, \Omega_k) \\ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} x^*(t - \alpha) \prod_{i=1}^{L-1} x^*(t - \alpha + \tau_i) \prod_{i=L}^k x(t - \alpha + \tau_i) \prod_{i=1}^k e^{-j\Omega_i \tau_i} d\tau_i \end{aligned} \quad (3.118)$$

with  $L$  conjugated terms ( $1 \leq L \leq k$ ). The value of  $\alpha$  is

$$\alpha = \frac{1}{k+1} \sum_{i=1}^k \tau_i. \quad (3.119)$$

It is chosen such that the local moment function

$$R_k(t, \tau_1, \tau_2, \dots, \tau_k) = x^*(t - \alpha) \prod_{i=1}^{L-1} x^*(t - \alpha + \tau_i) \prod_{i=L}^k x(t - \alpha + \tau_i)$$

is centered at the time instant  $t$ , i.e., the mean value of all its arguments is  $[(t - \alpha) + \sum_{i=1}^k (t - \alpha + \tau_i)]/(k + 1) = t$ .

In terms of the signal's FT, the above equation becomes:

$$\begin{aligned} W_k(t, \Omega_1, \Omega_2, \dots, \Omega_k) \\ = \frac{1}{2\pi} \int_{-\infty}^{\infty} X^* \left( \sum_{i=1}^k \Omega_i + \frac{\theta}{k+1} \right) \prod_{i=1}^{L-1} X^* \left( -\Omega_i + \frac{\theta}{k+1} \right) \prod_{i=L}^k X \left( \Omega_i - \frac{\theta}{k+1} \right) e^{-j\theta t} d\theta. \end{aligned} \quad (3.120)$$

The mean frequency over the multi-frequency space is defined by:

$$\Omega_m(t) = \frac{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \Omega_m W_k(t, \Omega_1, \Omega_2, \dots, \Omega_k) \prod_{i=1}^k d\Omega_i}{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} W_k(t, \Omega_1, \Omega_2, \dots, \Omega_k) \prod_{i=1}^k d\Omega_i}, \quad m = 1, 2, \dots, k.$$

For a signal  $x(t) = A \exp(j\varphi(t))$ , having in mind the  $k$ -dimensional FT pair

$$R_k(t, \tau_1, \tau_2, \dots, \tau_k) \longleftrightarrow W_k(t, \Omega_1, \Omega_2, \dots, \Omega_k),$$

it can be calculated as:

$$\Omega_m(t) = -j \frac{\frac{\partial}{\partial \tau_m} \left[ x^*(t - \alpha) \prod_{i=1}^{L-1} x^*(t - \alpha + \tau_i) \prod_{i=L}^k x(t - \alpha + \tau_i) \right]_{\tau_1=\tau_2=\dots=\tau_k=0}}{\left[ x^*(t - \alpha) \prod_{i=1}^{L-1} x^*(t - \alpha + \tau_i) \prod_{i=L}^k x(t - \alpha + \tau_i) \right]_{\tau_1=\tau_2=\dots=\tau_k=0}}. \quad (3.121)$$

For  $\alpha$  being a linear function of  $\tau_i$  (3.119), we get

$$\Omega_m(t) = \varphi'(t) \left[ \frac{L}{k+1} \pm 1 - \frac{1}{k+1}(k-L+1) \right],$$

where  $-1$  stands for  $m \leq L-1$  and  $+1$  for  $L \leq m \leq k$ . For the special case of  $L=1$ , we get

$$\Omega_m(t) = \frac{2}{k+1} \varphi'(t).$$

This is in agreement with our previous Wigner bispectrum ( $k=2, L=1$ ) analysis of a sinusoidal signal, when we obtained that the Wigner bispectrum peak is located at  $(\Omega_1, \Omega_2) = (2\Omega_0/3, 2\Omega_0/3)$ . Locations of the peaks are biased with respect to the true IF position. An interesting case, that will be used later, is for  $L=(k+1)/2$ , when the IF positions are unbiased:

$$\Omega_m(t) = \pm \varphi'(t), \quad m = 1, 2, \dots, k. \quad (3.122)$$

In this case the number of conjugated and non-conjugated terms is equal. Sign  $-$  is for the  $m$  corresponding to the axis associated with the conjugated terms, while  $\Omega_m(t) = +\varphi'(t)$  stands for the axis corresponding to the non-conjugated terms.

### 3.03.4.3 Wigner multi-time distribution

A distribution dual to the Wigner higher order spectra (3.118) and (3.120) is introduced and defined as the multi-time Wigner higher order distribution (MTWD). The MTWD in terms of signal  $x(t)$  in time domain reads

$$\begin{aligned} W_k(\Omega, t_1, t_2, \dots, t_k) = & \int_{-\infty}^{\infty} x^* \left( \sum_{i=1}^k t_i + \frac{\tau}{k+1} \right) \prod_{i=1}^{L-1} x^* \left( -t_i + \frac{\tau}{k+1} \right) \\ & \times \prod_{i=L}^k x \left( t_i - \frac{\tau}{k+1} \right) e^{j\tau\Omega} d\tau. \end{aligned} \quad (3.123)$$

All properties of the multi-time Wigner distribution are dual to the ones for the Wigner higher order spectra. For example, the frequency marginal (i.e., integral of  $W_k(\Omega, t_1, t_2, \dots, t_k)$  over frequency), for  $k = 2$  and  $L = 1$ , is a value dual to bispectrum,  $x^*(t_1 + t_2)x(t_1)x(t_2)$ .

The multi-time Wigner distribution leads to the introduction of an efficient higher order distribution, the  $L$ -Wigner distribution, obtained as its slice. Consider a multi-component signal, formed as a sum of short duration (pulse) signals:

$$x(t) = \sum_{m=1}^M x_m(t - d_m), \quad (3.124)$$

where  $x_m(t)$  ( $m = 1, 2, \dots, M$ ) are such that  $x_m(t) = 0$  for  $|t| \geq \epsilon$ , with  $\epsilon$  being small as compared to the considered time interval.

As in the case of the instantaneous frequency, (3.122), we can show that the location of auto-terms along  $\tau$  (or group delay) depends on the signal's position  $d$ , for any  $L$ , except for  $L = (k+1)/2$ . This case was also preferred in cumulant analysis. When  $L = (k+1)/2$ , the auto-terms are located at the  $\tau$  axis origin and its vicinity. Also, we may easily conclude from (3.123) and (3.124) that for  $L = (k+1)/2$  the auto-terms lie, in the  $k$ -dimensional  $t_1, t_2, \dots, t_k$  space, along line  $s$  defined by:

$$\begin{aligned} s : t_1 &= -t, \quad t_2 = -t, \dots, t_{L-1} = -t, \\ t_L &= t, \quad t_{L+1} = t, \dots, t_k = t \end{aligned} \quad (3.125)$$

at the points  $t = d_m$ . The illustration of multi-time Wigner distribution of the second order with  $k = 2$ , dual to the Wigner bispectrum, which cannot satisfy the condition  $L = (k+1)/2$ , as well as the illustration of multi-time Wigner distribution of the third order (with  $k = 3$ , dual to the Wigner trispectrum), as the lowest one satisfying the previous condition (if one does not count the well-known Wigner distribution), are given in Figure 3.33.

For  $M > 1$  in (3.124), and for  $L = (k+1)/2$ , considering only line  $s$ , it can be shown that the regions corresponding to cross-terms, where the integrand in (3.123) is different from zero, are dislocated from the  $\tau$  axis origin.

The multi-time Wigner higher order distribution, with  $L = (k+1)/2$ , along the line  $s$ , given by (3.125), is equal to the  $L$ -Wigner distribution. It is defined as

$$\text{LWD}_L(t, \Omega) = \int_{-\infty}^{\infty} x^{*L} \left( t - \frac{\tau}{2L} \right) x^L \left( t + \frac{\tau}{2L} \right) e^{-j\Omega\tau} d\tau. \quad (3.126)$$

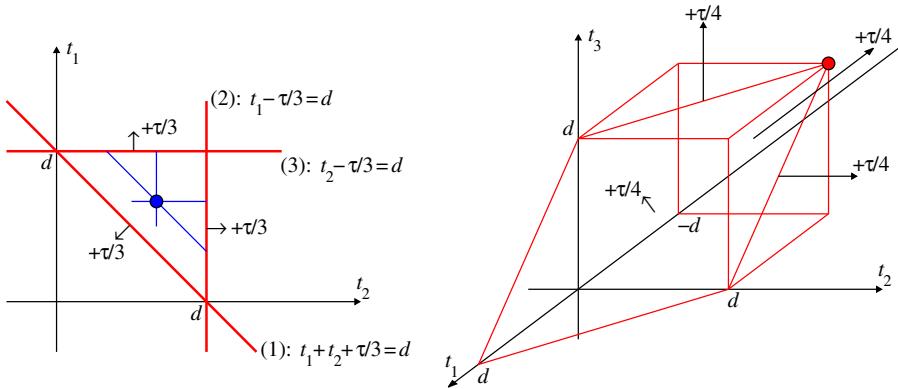


FIGURE 3.33

Illustration of the multi-time Wigner distribution of: the second order (left), the third order (right).

For  $L = 1$ , the  $L$ -Wigner distribution reduces to the Wigner distribution. The  $L$ -Wigner distribution was introduced before the time-varying higher order spectra, as a distribution that improves time-frequency concentration along the instantaneous frequency, since its spread factor is

$$Q(t, \tau) = \left[ L\phi\left(t + \frac{\tau}{2L}\right) - L\phi\left(t - \frac{\tau}{2L}\right) \right] - \phi'(t)\tau = \frac{1}{24L^2}\phi^{(3)}(t)\tau^3 + \dots$$

Its pseudo form, as it will be used in the sequel, is defined by introducing a lag localization window  $w_L(\tau)$ ,

$$\text{LWD}_L(t, \Omega) = \int_{-\infty}^{\infty} w_L(\tau)x^{*L}\left(t - \frac{\tau}{2L}\right)x^L\left(t + \frac{\tau}{2L}\right)e^{-j\Omega\tau}d\tau. \quad (3.127)$$

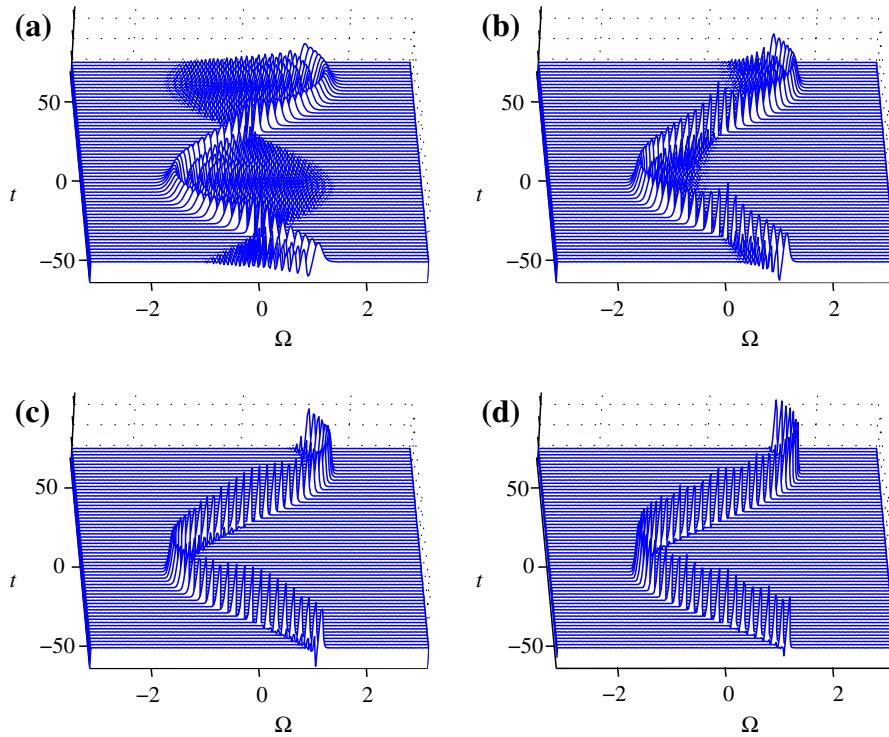
Since we will use only this form in the analysis and realizations, we will not introduce a new notation for it. For a signal  $x(t) = A \exp(j\phi(t))$ , expanding  $\phi(t \pm \tau/2L)$  into a Taylor series around  $t$ , up to the third order term, we get:

$$\text{LWD}_L(t, \Omega) = 2\pi A^{2L} \delta(\omega - \phi'(t)) *_{\Omega} W_L(\Omega) *_{\Omega} \text{FT} \left\{ e^{j\frac{\phi'''(t)}{24L^2}\tau^3} \right\}, \quad (3.128)$$

where  $*_{\Omega}$  denotes the frequency domain convolution and  $W_L(\Omega) = \text{FT}\{w_L(\tau)\}$ . From (3.128) one may conclude that the generalized power  $A^{2L}$  is concentrated at the instantaneous frequency  $\phi'(t)$ . The distortions caused by the shape of the phase function are due to the existence of its third and higher order derivatives. If the instantaneous frequency is a linear function of time, then the Wigner distribution ( $L = 1$ ) produces the ideal concentration. But, if that is not the case, then  $L > 1$  reduces the distortion. In other words, the pseudo  $L$ -Wigner distribution locally linearize the instantaneous frequency function, by reducing the spreading term  $\text{FT}\{\exp(j\phi'''(t)\tau^3/(24L^2))\}$  influence by  $L^2$ .

The pseudo  $L$ -Wigner distribution of an order  $L$  can be calculated based on the  $L$ -Wigner distribution of order  $L/2$  as

$$\text{LWD}_L(t, \Omega) = \text{LWD}_{L/2}(t, 2\Omega) *_{\Omega} \text{LWD}_{L/2}(t, 2\Omega). \quad (3.129)$$

**FIGURE 3.34**

Time-frequency representation of a sinusoidally (nonlinear) frequency modulated signal: (a) Wigner distribution. (b)  $L$ -Wigner distribution with  $L = 2$ . (c)  $L$ -Wigner distribution with  $L = 4$ . (d)  $L$ -Wigner distribution with  $L = 8$ .

**Example 17.** The pseudo Wigner distribution, along with the  $L$ -Wigner distributions with  $L = 2$ ,  $L = 4$ , and  $L = 8$  are shown in Figure 3.34 for a sinusoidally frequency modulated signal.

A form of the  $L$ -Wigner distribution, that preserves the signal energy property, is presented in the Section 3.03.3.2.4 as the pseudo quantum signal representation.

#### 3.03.4.4 Signal phase derivative and distributions definitions

Let us consider signal of the form

$$x(t) = A e^{j\varphi(t)}$$

with the instantaneous frequency

$$\omega(t) = \frac{d\varphi(t)}{dt}.$$

In order to estimate the instantaneous frequency from the phase function we can use the following approximative relations for the first derivative.

*Quadratic distributions:*

- First order backward estimation

$$\omega(t) \approx \frac{\varphi(t) - \varphi(t - \tau)}{\tau} = \frac{d\varphi(t)}{dt} + O(\varphi''(\tau))$$

with error of  $\varphi''(\tau)$  order. Time-frequency distribution corresponding to this estimation is the Richazek distribution

$$RD(t, \omega) = \int_{-\infty}^{\infty} x(t)x^*(t - \tau)e^{-j\omega\tau} d\tau.$$

It is the FT of  $A^2 \exp(j(\varphi(t) - \varphi(t - \tau)))$ , thus being concentrated at the  $\omega = d\varphi(t)/dt$  with the spread factor depending on  $\varphi''(\tau)$ .

- Symmetric derivative estimation

$$\omega(t) \approx \frac{\varphi(t + \tau/2) - \varphi(t - \tau/2)}{\tau} = \frac{d\varphi(t)}{dt} + O(\varphi'''(\tau))$$

obviously corresponds to the Wigner distribution with the spread factor depending on  $\varphi'''(\tau)$ . For a linear frequency modulated signal (quadratic phase) there is no estimation error in derivative. Therefore, in this case, the Wigner distribution is ideally concentrated, as it is well known.

*Higher order distributions*

We can further improve the estimation accuracy but at the cost of the estimator complexity. For example,

$$\omega(t) \approx \frac{-\varphi(t - \tau/6) + 8\varphi(t - \tau/12) - 8\varphi(t + \tau/12) + \varphi(t + \tau/6)}{\tau} = \frac{d\varphi(t)}{dt} + O(\varphi^{(5)}(\tau)) \quad (3.130)$$

corresponds to a distribution

$$PD(t, \Omega) = \int_{-\infty}^{\infty} x^*(t - \tau/6)x^8(t - \tau/12)x^{*8}(t + \tau/12)x(t + \tau/6)e^{-j\Omega\tau} d\tau$$

that is fully concentrated along the IF up to the fifth order polynomial phase of the signal. However, it is of a quite high order with all drawbacks from the increased order.

In general, an estimator (distribution), may be written as

$$\omega(t) \approx \frac{\sum_i b_i \varphi(t + c_i \tau)}{\tau} = \frac{d\varphi(t)}{dt} + O(\varphi^{(p)}(\tau)). \quad (3.131)$$

Coefficients  $b_i$  and  $c_i$  follow from the system of equations, obtained by expanding  $b_i \varphi(t + c_i \tau)$  into a Taylor series around  $t$ ,

$$b_i \varphi(t + c_i \tau) = b_i \varphi(t) + b_i \varphi'(t)c_i \tau + b_i \varphi''(t) \frac{(c_i \tau)^2}{2!} + b_i \varphi'''(t) \frac{(c_i \tau)^3}{3!} + b_i \varphi^{(4)}(t) \frac{(c_i \tau)^4}{4!} + \dots$$

and the conditions that:

1. The sum of coefficients with  $\varphi(t)$  is equal to 0.
2. The sum of coefficients with  $\varphi'(t)$  is equal to 1.
3. The sum of coefficients with  $\varphi^{(n)}(t)$  is equal to 0 up to the desired order.

For a fourth order estimators of the first derivative we get:

$$\begin{aligned} b_1 + b_2 + b_3 + b_4 &= 0, \\ b_1 c_1 + b_2 c_2 + b_3 c_3 + b_4 c_4 &= 1, \\ b_1 c_1^2 + b_2 c_2^2 + b_3 c_3^2 + b_4 c_4^2 &= 0, \\ b_1 c_1^3 + b_2 c_2^3 + b_3 c_3^3 + b_4 c_4^3 &= 0, \\ b_1 c_1^4 + b_2 c_2^4 + b_3 c_3^4 + b_4 c_4^4 &= 0. \end{aligned}$$

If we add the requirement that a distribution is real valued, i.e., that the estimator is symmetric, then

$$\begin{aligned} b_1 &= -b_2, & c_1 &= -c_2, \\ b_3 &= -b_4, & c_3 &= -c_4 \end{aligned}$$

resulting in the system of equations

$$\begin{aligned} b_1 c_1 + b_3 c_3 &= 1/2, \\ b_1 c_1^3 + b_3 c_3^3 &= 0. \end{aligned}$$

#### *Polynomial Wigner-Ville distribution*

Assuming the lowest possible integer values (signal powers)  $b_1 = 2$  and  $b_3 = -1$  (negative  $b$  corresponds to a conjugation of signal, i.e., for example for  $b = -2$ , we use with  $x^{*2}(t)$ ) we get the fourth order polynomial Wigner-Ville distributions (PWVD) coefficients  $c_1 \simeq 0.675$  and  $c_3 \simeq 0.85$ , defined (by Boashash et al.) as [24,43]

$$\text{PD}(t, \Omega) = \int_{-\infty}^{\infty} x^2(t + 0.675\tau) x^{*2}(t - 0.675\tau) x^*(t + 0.85\tau) x(t - 0.85\tau) e^{-j\Omega\tau} d\tau. \quad (3.132)$$

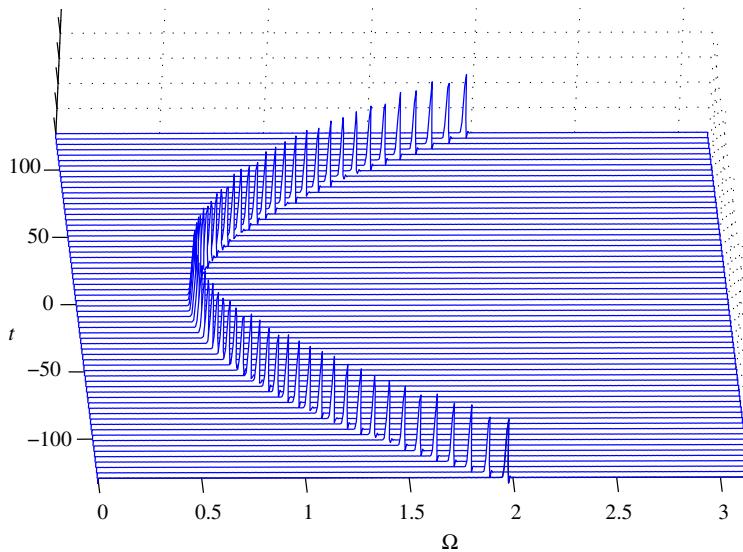
The polynomial Wigner-Ville distribution corresponds to the estimator:

$$\begin{aligned} \omega(t) &\approx \frac{2\varphi(t + 0.675\tau) - 2\varphi(t - 0.675\tau) - \varphi(t + 0.85\tau) + \varphi(t - 0.85\tau)}{\tau} \\ &= \frac{d\varphi(t)}{dt} + O(\varphi^{(5)}(\tau)) \end{aligned}$$

that is of lower order than (3.130), with the same error order.

The polynomial Wigner-Ville distribution of a fourth order polynomial phase signal is shown in Figure 3.35.

Another approach, that was used in literature to define polynomial Wigner-Ville distribution, was to assume values for  $c_1$  and  $c_2$ , appropriate for discrete realization without interpolation. It results in rational values of  $b_1$  and  $b_3$ , i.e., rational signal powers.

**FIGURE 3.35**

Polynomial Wigner-Ville distribution of a signal with the fourth order polynomial phase.

#### *Complex argument distribution*

A complex time distribution, that preserves energy and time marginal condition for frequency modulated signals, may be derived from a similar analysis, allowing complex arguments. For example, the fourth order form follows for  $b_1 = 1$  and  $b_3 = j$ , when we get  $c_1 = 1/4$  and  $c_3 = -j/4$ , with frequency (first derivative) estimator

$$\Omega(t) \approx \frac{\phi\left(t + \frac{\tau}{4}\right) - \phi\left(t - \frac{\tau}{4}\right) + j\phi\left(t - j\frac{\tau}{4}\right) - j\phi\left(t + j\frac{\tau}{4}\right)}{\tau}. \quad (3.133)$$

The corresponding distribution is

$$\text{CTD}(t, \Omega) = \int_{-\infty}^{\infty} x\left(t + \frac{\tau}{4}\right)x^*\left(t - \frac{\tau}{4}\right)x^j\left(t - j\frac{\tau}{4}\right)x^{-j}\left(t + j\frac{\tau}{4}\right)e^{-j\Omega\tau}d\tau. \quad (3.134)$$

In the derivation of (3.133), the series

$$\phi(t + j\tau) = \phi(t) + j\tau\phi'(t) - \tau^2\phi''(t)/2! - j\tau^3\phi'''(t)/3! + \dots$$

is used. It is interesting to note that a few years after the complex-time argument derivative estimation was proposed and used in signal analysis, the estimator based on the complex-argument, of the form

$$\tau\phi'(t) \approx \text{Im } \phi(t + j\tau)$$

was reintroduced in mathematical journals.

The spread factor  $Q(t, \tau)$  for this distribution is

$$Q(t, \tau) = \phi^{(5)}(t) \frac{\tau^5}{445!} + \phi^{(9)}(t) \frac{\tau^9}{489!} + \dots \quad (3.135)$$

The dominant term in  $Q(t, \tau)$  is of the fifth order. All existing terms are significantly reduced as compared to the respective ones in the Wigner distribution. Continuous form of the “complex-time” signal  $x(\varsigma)$  is dual to the Laplace transform of  $x(t)$ . Complex time is denoted by  $\varsigma = t + j\tau$ .

$$x(\varsigma) = x(t + j\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{-\Omega\tau} e^{j\Omega t} d\Omega.$$

The signal with a “complex-time” argument could be calculated in the same way as the Laplace transform is calculated from the FT. Value  $x(\varsigma)$  converges within the entire complex plane  $\varsigma$  if  $x(t)$  is a band limited signal. In practical realizations, the values of  $x(n)$  are available as a set of data along the real axis only. *The values of signal with complex argument are not known. They must be determined from the samples on the real time axis.* This problem is well studied mathematics. It is known as an analytical extension (continuation) of the real argument function. An analytic extension of the signal  $x(n)$  is defined as a sum of the analytic extensions of complex exponential functions. It is of the form

$$x(\eta) = x(n + jm) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} X(k) e^{-\frac{2\pi}{N} mk} e^{j\frac{2\pi}{N} nk}.$$

If we multiply  $X(k) = \text{FFT}[x(n)]$  by  $\exp(-2\pi mk/N)$ , for a given  $m$ , then  $x(n + jm)$  is obtained as  $x(n + jm) = \text{IFFT}[X(k) \exp(-2\pi mk/N)]$ . The presented form of  $x(n + jm)$  could directly be used for the realization of a complex-lag distribution. Real exponential functions  $\exp(-2\pi mk/N)$ , may be out of the computer precision range, significantly worsening the results. Thus, one should carefully use the above relations in the direct numerical realization. Procedures for the cross-terms reduced calculation, reducing the distribution sensitivity to the calculation precision, is based on the *S*-method.

### Real Time Distributions

An interesting approximation

$$\omega(t) \approx \frac{\varphi(t - \tau) - 4\varphi(t - \tau/2) + 3\varphi(t)}{\tau} = \frac{d\varphi(t)}{dt} + O(\varphi''(\tau))$$

leads to a distribution that is fully concentrated for linear frequency modulated signals, as the Wigner distribution. However, in contrast to the Wigner distribution that uses past and future signal values (arguments  $t + \tau/2$  and  $t - \tau/2$  are used), this distribution uses only past signal values:

$$\text{RTD}(t, \Omega) = \int_0^{\infty} x(t - \tau) x^{*4}(t - \tau/2) x^3(t) e^{-j\Omega\tau} d\tau.$$

Its pseudo form is

$$\text{RTD}(t, \Omega) = \int_0^T w(\tau) x(t - \tau) x^{*4}(t - \tau/2) x^3(t) e^{-j\Omega\tau} d\tau.$$

This distribution (although not real valued) can be efficiently used in many application when it is important to work in real time and to estimate the instantaneous frequency not in the middle point of the analyzed interval (as the Wigner distribution does for the middle point of the lag window) but at the current, last point of the considered time interval. An example of such importance is in radar signals, when the estimation of the target parameters at the middle of the considered interval (coherent integration time of the order of seconds) could be quite late and inappropriate for decision.

### 3.03.4.5 STFT based realization of higher order representations

Here, we will extend the  $S$ -method based approach to the realization of the higher order time-frequency forms, obtained by reducing the full higher order forms to the two-dimensional time-frequency plane. This approach will provide two substantial advantages over the direct calculation: (1) It produces the higher order distributions, without need for signal oversampling. (2) In the case of multi-component signals the cross-terms are reduced (eliminated).

#### 3.03.4.5.1 $L$ -Wigner distribution realization

The relationship between the  $L$ -Wigner distribution of an order  $2L$  and  $L$ -Wigner distribution of an order  $L$  is of the form

$$\text{LWD}_{2L}(t, \Omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} \text{LWD}_L(t, \Omega + \theta) \text{LWD}_L(t, \Omega - \theta) d\theta. \quad (3.136)$$

The realization of cross-terms and alias free version of the  $L$ -Wigner distribution may be efficiently done in the discrete domain, by using the  $S$ -method realization form, as

$$\text{LWD}_{2L}(n, k) = \text{LWD}_L^2(n, k) + 2 \sum_{i=1}^{L_P} \text{LWD}_L(n, k+i) \text{LWD}_L(n, k-i) \quad (3.137)$$

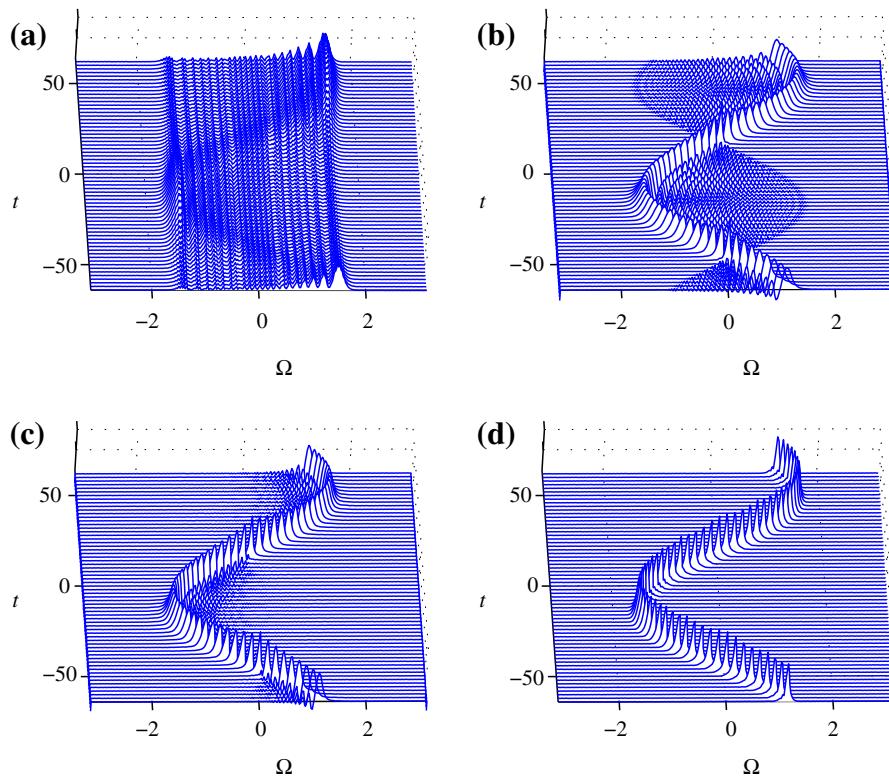
with  $\text{LWD}_1(n, k) = \text{SM}(n, k)$ , calculated according to the described procedure in Section 3.03.3.6.1,

$$\text{LWD}_1(n, k) = |\text{STFT}(n, k)|^2 + 2 \sum_{i=1}^{L_P} \text{Re}\{\text{STFT}(n, k+i) \text{STFT}^*(n, k-i)\}. \quad (3.138)$$

Form (3.137) is very convenient for software and hardware realizations since the same blocks, connected in cascade, can provide a simple and efficient system for higher order time-frequency analysis, based on the STFT in the initial step, with signal sampled at the Nyquist rate (see Figure 3.36).

**Example 18.** First we will present the  $L$ -Wigner distribution realization for the signal presented in Figure 3.34. The spectrogram is shown in Figure 3.36a. The Wigner distribution realized according to (3.138), without oversampling, is shown in Figure 3.36b. The  $L$ -Wigner distribution distributions, calculated by using (3.137), are presented in Figure 3.36c and d, for  $L = 2$  and  $L = 8$  respectively. The only difference from Figure 3.34 is that here we used a lag window  $w(\tau) = \exp(-|\tau/\sigma|)$  that is order invariant,  $w(\tau) = w(\tau/2)w(-\tau/2) = w^2(\tau/2)^2w(-\tau/2) = \dots$ , so that we can make fair comparisons of different order distributions.

**Example 19.** Similar calculations were repeated for a two-component signal. In this case, in order to provide a good graphical presentation at the point of intersection (where the distribution values increases

**FIGURE 3.36**

Time-frequency representation of a sinusoidally frequency modulated signal: (a) The spectrogram. (b) The Wigner distribution. (c) The  $L$ -Wigner distribution with  $L = 2$ . (d) The  $L$ -Wigner distribution with  $L = 8$ .

with a power of  $2L$ ) the distribution values are normalized, for each time instant  $n$ , with a maximal value for that instant over all  $k$ , i.e., just in Figure 3.37 the value  $LW_{2L}(n, k)/\max_k(LW_{2L}(n, k))$  for each  $n$ , is plotted.

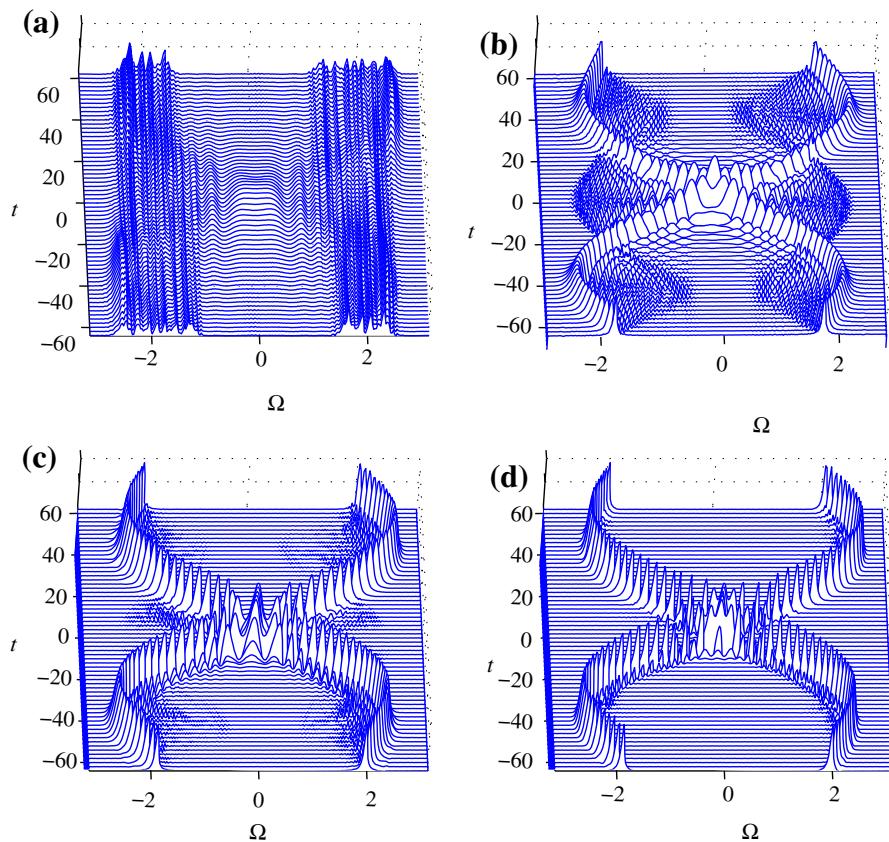
#### 3.03.4.5.2 Polynomial Wigner-Ville distribution realization

Modification of the presented method for the realization of the polynomial Wigner-Ville distribution is straightforward. The fourth order polynomial Wigner-Ville distribution

$$PD(t, \Omega) = \int_{-\infty}^{\infty} x^2(t + 0.675\tau)x^{*2}(t - 0.675\tau)x^*(t + 0.85\tau)x(t - 0.85\tau)e^{-j\Omega\tau} d\tau \quad (3.139)$$

can be written, by using the change of variables  $0.675\tau \rightarrow \tau/4$  (or  $\tau \rightarrow \tau/2.7$ ) as

$$PD(t, \Omega) = \int_{-\infty}^{\infty} x^2\left(t + \frac{\tau}{4}\right)x^{*2}\left(t + \frac{\tau}{4}\right)x^*\left(t + \frac{1.7\tau}{2.7}\right)x\left(t + \frac{1.7\tau}{2.7}\right)e^{-j\tau\Omega/2.7} d\tau.$$

**FIGURE 3.37**

Time-frequency representation of a multi-component signal: (a) the spectrogram, (b) the Wigner distribution, (c) the  $L$ -Wigner distribution with  $L = 2$ , and (d) the  $L$ -Wigner distribution with  $L = 8$ .

In a frequency scaled form

$$\text{PD}(t, \Omega) = \frac{1}{2.7} \int_{-\infty}^{\infty} x^2 \left( t + \frac{\tau}{4} \right) x^{*2} \left( t - \frac{\tau}{4} \right) x^* \left( t + A \frac{\tau}{2} \right) x \left( t - A \frac{\tau}{2} \right) e^{-j\tau\Omega_s} d\tau, \quad (3.140)$$

where  $A = 1.7/2.7$  and  $\Omega_s = \Omega/2.7$ . Note that

$$\text{PD}(t, \Omega_s) = \frac{1}{2.7} \text{LWD}_2(t, \Omega_s) *_{\Omega_s} \text{WD}^A(t, \Omega_s), \quad (3.141)$$

where

$$W^A(t, \Omega_s) = \text{FT} \left\{ x^* \left( t + A \frac{\tau}{2} \right) x \left( t - A \frac{\tau}{2} \right) \right\}$$

is a scaled and frequency reversed version (due to order of conjugate terms) of the pseudo Wigner distribution and

$$\text{LWD}_2(t, \Omega_s) = \text{FT} \left\{ x^2 \left( t + \frac{\tau}{4} \right) x^{*2} \left( t - \frac{\tau}{4} \right) \right\}$$

is the  $L$ -Wigner distribution with  $L = 2$ . The cross-terms free realization of the pseudo Wigner distribution and the pseudo  $L$ -Wigner distribution is already presented.

In the discrete implementation of the above relation the only remaining problem is the evaluation of  $\text{WD}^A(t, \Omega_s)$  on the discrete set of points on the frequency axis,  $\Omega_s = -k\Delta\Omega_s$ . Since  $\text{WD}^A(t, \Omega_s)$  is, by definition, a scaled and frequency reversed version of  $\text{WD}(t, \Omega)$ . Therefore the values of  $\text{WD}^A(t, \Omega_s)$  at  $\Omega_s = -k\Delta\Omega_s$  are the values of  $\text{WD}(t, \Omega)$  at  $\Omega_s = k\Delta\Omega_s/A$ . However, these points do not correspond to any sample location along the frequency axis grid. Thus, the interpolation of the pseudo Wigner distribution has to be done (one way of doing it is in an appropriate zero padding of the signal).

A discrete form of convolution (3.140), including rectangular window  $P(\theta)$  and the above considerations, is

$$\text{PD}(n, k) = \sum_{i=-L_P}^{L_P} L \text{W}_2(n, k+i) \widehat{\text{WD}}(n, k+i/A), \quad (3.142)$$

where  $2L_P + 1$  is the width of  $P(\theta)$  in the discrete domain, while  $\widehat{\text{WD}}(n, k+i/A)$  is the pseudo Wigner distribution approximation. We can simply use  $\widehat{\text{WD}}(n, k+i/A) = \text{SM}_x(n, k+[i/A])$  where  $[i/A]$  is the nearest integer to  $i/A$ , or use the linear interpolation of the pseudo Wigner distribution ( $S$ -method) values at two nearest integers. The terms in (3.142), when  $k+i$ , or  $k+[i/A]$  is outside the basic period, are considered as being zero in order to avoid aliasing.

**Example 20.** Consider a real-valued multi-component signal

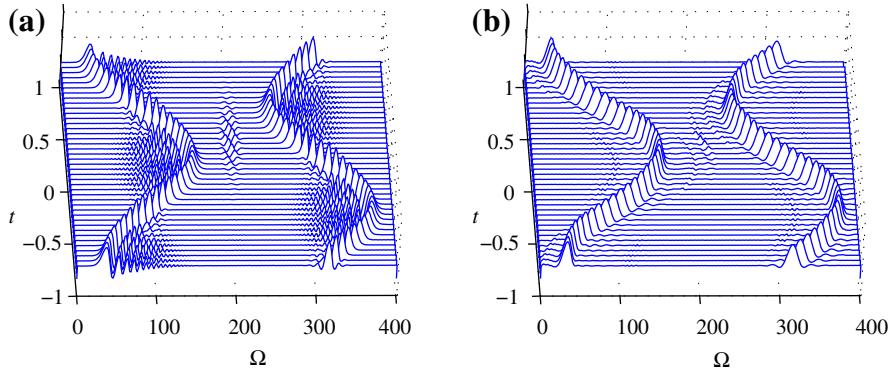
$$x(t) = \cos(20 \sin(\pi t) + 30\pi t) + \sin(20 \cos(\pi t) + 100\pi t)$$

within  $-1 \leq t < 1$ , with  $\Delta t = 1/128$ . In the realization, a Hann(ing) window of the width  $T_w = 2$  is used. Based on the STFT (using its positive frequencies), the cross-terms free form of the Wigner distribution is obtained from (3.138) with  $L_P = 15$ , by using the  $S$ -method, Figure 3.38a. Then the  $L$ -Wigner distribution, with  $L = 2$ , is calculated according to (3.137). It is combined with the linearly interpolated  $S$ -method value into the PWVD (3.142), shown in Figure 3.38b. For the precise implementation of  $[i/\chi]$  the lag window has been zero-padded by a factor of 2.

A similar approach can be used for the STFT based realization of the complex time distributions and all other higher order representations.

### 3.03.4.6 Higher order ambiguity functions

A specific class of non-linear frequency modulated signals are the polynomial phase signals. This case is of importance not only when the phase is of polynomial form, but also in quite general case, when signals within a smaller time intervals can be approximated by finite order polynomial functions, using the Taylor series. By using the multi-lag higher order instantaneous moments the polynomial order of the phase function is reduced, in several steps, to the linear one. The FT of the multi-lag instantaneous moments, known as the multi-lag higher order ambiguity function, is used to estimate the highest phase coefficient. After the original signal is demodulated, the procedure is repeated until all parameters of

**FIGURE 3.38**

Time-frequency representation of a real-valued multi-component signal: (a) The S-method (cross-terms and alias free version of the Wigner distribution). (b) Polynomial Wigner-Ville distribution realized based on the STFT by using the S-method and its order recursive form.

the phase are estimated. The product higher order ambiguity function (PHAF) is introduced to estimate the polynomial phase signal coefficients in the case of multi-component signals [21, 44–46].

Consider a deterministic polynomial phase signal

$$x(t) = A \exp(j\varphi(t)) = A \exp\left(j \sum_{p=0}^P \alpha_p t^p\right) \quad (3.143)$$

and see what will be the result of an operation corresponding the studied symmetric phase derivative estimation

$$\hat{\Omega}(t) = \frac{\varphi(t + \tau_1) - \varphi(t - \tau_1)}{2\tau_1}.$$

In terms of signal, it means

$$\begin{aligned} x(t + \tau_1)x^*(t - \tau_1) &= A^2 \exp(j(\varphi(t + \tau_1) - \varphi(t - \tau_1))) \\ &= A^2 \exp\left(j \sum_{p=0}^P \alpha_p (t + \tau_1)^p - j \sum_{p=0}^P \alpha_p (t - \tau_1)^p\right). \end{aligned} \quad (3.144)$$

The highest order phase term is transformed into

$$\alpha_P(t + \tau_1)^P - \alpha_P(t - \tau_1)^P = 2\alpha_P \tau_1 \left[ (t + \tau_1)^{P-1} + (t + \tau_1)^{P-2}(t - \tau_1) + \cdots + (t - \tau_1)^{P-1} \right].$$

Thus, the phase order in  $t$  is reduced in  $x(t + \tau_1)x^*(t - \tau_1)$  to  $P - 1$ ,

$$R(t, \tau_1) = x(t + \tau_1)x^*(t - \tau_1) = A^2 \exp \sum_{p=0}^{P-1} \beta_p(\tau_1) t^p. \quad (3.145)$$

The highest order coefficient, with  $t^{P-1}$ , in  $R(t, \tau_1)$  is

$$\beta_{P-1}(\tau_1) = 2\alpha_P \tau_1 P. \quad (3.146)$$

It is possible to continue in this way, by forming

$$R_2(t, \tau_1, \tau_2) = R(t + \tau_2, \tau_1) R^*(t - \tau_2, \tau_1) = A^4 \exp \sum_{p=0}^{P-2} \gamma_p(\tau_1, \tau_2) t^p.$$

The highest order coefficient in  $R_2(t, \tau_1, \tau_2)$ , according to (3.146), is

$$\gamma_{P-1}(\tau_1, \tau_2) = 2\beta_{P-1}(\tau_1)\tau_2(P-1) = 4\alpha_P \tau_1 \tau_2 P(P-1).$$

After  $P-1$  steps, a pure complex sinusoidal signal in  $R(t, \tau_1, \tau_2, \dots, \tau_{P-1})$  is obtained. The functions  $R_{P-1}(t, \tau_1, \tau_2, \dots, \tau_{P-1})$  are called the multi-lag higher order instantaneous moments. The coefficient with  $t$ , in the final sinusoid is obtained as

$$\Omega_0 = 2^{P-1} \alpha_P \tau_1 \tau_2 \dots \tau_{P-1} P! \quad (3.147)$$

It can be calculated by using the multi-lag higher order ambiguity function, defined as the FT of the multi-lag higher order instantaneous moment,

$$X_{P-1}(\Omega, \tau_1, \tau_2, \dots, \tau_{P-1}) = \int_{-\infty}^{\infty} R_{P-1}(t, \tau_1, \tau_2, \dots, \tau_{P-1}) e^{-j\Omega t} dt, \quad (3.148)$$

as

$$\begin{aligned} \Omega_0 &= \arg \left\{ \max_{\Omega} X_{P-1}(\Omega, \tau_1, \tau_2, \dots, \tau_{P-1}) \right\}, \\ \hat{\alpha}_P &= \frac{\Omega_0}{2^{P-1} \alpha_P \tau_1 \tau_2 \dots \tau_{P-1} P!}. \end{aligned} \quad (3.149)$$

Now, the original signal is demodulated by

$$x_1(t) = x(t) \exp(-j\hat{\alpha}_P t^P),$$

producing, in an ideal case, a signal with a lower,  $P-1$ , order of the phase,

$$x_1(t) = A \exp \left( j \sum_{p=0}^{P-1} \alpha_p t^p \right).$$

All previous steps are repeated on  $x_1(t)$  to produce the next highest coefficient  $\hat{\alpha}_{P-1}$ . Here,  $P-2$  steps are performed. In this way lower and lower order coefficients are estimated. Note that if an error in the estimation of a coefficient  $\hat{\alpha}_P$  occurs, it will propagate and cause inaccurate lower order coefficients estimation.

**Example 21.** Consider a third-order ( $P = 3$ ) polynomial phase signal:

$$x(t) = A e^{j(a_0 + a_1 t + a_2 t^2 + a_3 t^3)}.$$

Estimate its highest (third) order coefficient.

The multi-lag higher order instantaneous moments of the signal  $x(t)$  are:

$$\begin{aligned} x(t) &= A e^{j(a_0 + a_1 t + a_2 t^2 + a_3 t^3)}, \\ R(t, \tau_1) &= x(t + \tau_1)x^*(t - \tau_1) = |A|^2 e^{j2a_1\tau_1 + 2a_3\tau_1^3} e^{j4a_2\tau_1 t} e^{ja_36t^2\tau_1}, \\ R_2(t, \tau_1, \tau_2) &= R(t + \tau_2, \tau_1)R^*(t - \tau_2, \tau_1) = |A|^4 e^{j8a_2\tau_1\tau_2} e^{ja_324\tau_1\tau_2 t}. \end{aligned}$$

The multi-lag higher order ambiguity function is the FT (over  $t$ ) of  $R_2(t, \tau_1, \tau_2)$ , i.e.,

$$X_2(\Omega, \tau_1, \tau_2) = \text{FT}_t\{R_2(t, \tau_1, \tau_2)\} = 2\pi |A|^4 e^{j8a_2\tau_1\tau_2} \delta(\Omega - 24a_3\tau_1\tau_2).$$

Obviously, from the position of the FT maximum, at

$$\Omega_0 = 24a_3\tau_1\tau_2 \quad (3.150)$$

follows

$$\hat{\alpha}_3 = \frac{\Omega_0}{24\tau_1\tau_2} = a_3.$$

The estimated coefficient can be used to unwrap the original signal to

$$\begin{aligned} x_1(t) &= x(t) e^{-j\hat{\alpha}_3 t^3} = A e^{j(a_0 + a_1 t + a_2 t^2 + a_3 t^3)} e^{-ja_3 t^3}, \\ &= A e^{j(a_0 + a_1 t + a_2 t^2)}. \end{aligned}$$

Now the multi-lag higher order instantaneous moments of the signal  $x_1(t)$  are calculated

$$\begin{aligned} x_1(t) &= A e^{j(a_0 + a_1 t + a_2 t^2)}, \\ R(t, \tau_1) &= x_1(t + \tau_1)x_1^*(t - \tau_1) = |A|^2 e^{j2a_1\tau_1} e^{ja_24t\tau_1}. \end{aligned}$$

The FT is

$$X(\Omega, \tau_1) = \text{FT}_t\{R(t, \tau_1)\} = 2\pi |A|^2 e^{j2a_1\tau_1} \delta(\Omega - 4a_2\tau_1)$$

with  $\Omega_0 = 4a_2\tau_1$  and  $\hat{\alpha}_2 = \Omega_0/(4\tau_1) = a_2$ .

The estimated coefficient is used to unwrap  $x_1(t)$ , as

$$x_{11}(t) = x_1(t) e^{-j\hat{\alpha}_2 t^2} = A e^{j(a_0 + a_1 t)}.$$

Now the FT of  $x_{11}(t)$ ,  $X_1(\Omega) = 2\pi A e^{ja_0} \delta(\Omega - a_1)$ , produces  $a_1$  and  $a_0$ .

When  $x(t)$  is a multi-component polynomial phase signal, i.e.,

$$x(t) = \sum_{k=1}^K A_k \exp \left( j \sum_{p=0}^P \alpha_{k,p} t^p \right), \quad (3.151)$$

where  $\alpha_{k,p}$  are the coefficients of the  $k$ th component, the  $P$ th order multi-lag higher instantaneous moments will contain  $K$  sinusoids that correspond to the auto-terms. Each auto-term has the frequency proportional to the corresponding highest order phase coefficient. In addition to the auto-terms, the multi-lag higher instantaneous moments will contain a large number of cross-terms which are, in general,  $P$ th order polynomial phase signals. When the highest order phase coefficients of some components coincide, the corresponding cross-terms are complex sinusoids, implying that some of the peaks in the multi-lag higher order ambiguity function correspond to the cross-terms. The maxima based estimation of phase coefficients is ambiguous, since a peak corresponding to a cross-term can be detected as a maximum, leading to a false estimation.

The effect of cross-terms can be considerably attenuated by using the product higher order ambiguity function (PHAF). The PHAF is based on the fact that, unlike the cross-terms, the auto-terms are at frequencies proportional to the product of time lags used for the calculation of the multi-lag higher order ambiguity function.

### 3.03.5 Processing of sparse signals in time-frequency

A signal is sparse in a certain transform domain if it contains a small number of nonzero coefficients when compared to the original discrete signal length. In the case of sparse signals, signal reconstruction can be performed by using a fewer number of randomly chosen signal samples. This kind of signal acquisition is referred to as compressive sensing. Compressive sensing is based on powerful mathematical algorithms for error minimization [47–49]. The compressive sensing concepts are able to reconstruct the original signal by using a small set of signal samples. Sparsity is one of the main requirements that should be satisfied, in order to efficiently apply the compressive sensing. If the signal is not sparse, then the compressive sensing cannot recover signal successfully. Sparsity in time-frequency analysis is closely related to the signal concentration measures, especially the concentration measures being equal to the area of the time-frequency region with non-zero time-frequency distribution values [50]. Thus, before explaining the basic principles of processing of sparse signals in time-frequency, concentration measures will be shortly reviewed.

#### 3.03.5.1 Concentration measures

Concentration measures in time-frequency analysis were introduced in the sense of the optimization of time-frequency presentations. Concentration measures are used for measurement of the representations quality. Intuitively we can assume that better concentration in time-frequency domain means that the signal energy is focused within smaller time-frequency region. Concentration measure can provide a quantitative criterion for evaluation of various representations performance. It can be used for adaptive and automatic parameter selection in time-frequency analysis, without supervision of a user.

In most cases, some quantities from statistics and information theory were the inspiration for defining concentration measures of time-frequency representations. The basic idea for measuring time-frequency representation concentration can be explained on a simplified example motivated by the probability theory. Consider a set of  $N$  non-negative numbers  $\{p_1, p_2, \dots, p_N\}$ , such that

$$p_1 + p_2 + \dots + p_N = 1. \quad (3.152)$$

Form a simple test function

$$\mathcal{M}(p_1, p_2, \dots, p_N) = p_1^2 + p_2^2 + \dots + p_N^2.$$

It is easy to conclude that  $\mathcal{M}(p_1, p_2, \dots, p_N)$ , under the constraint  $p_1 + p_2 + \dots + p_N = 1$ , has the minimal value for  $p_1 = p_2 = \dots = p_N = 1/N$ , i.e., for maximally spread values of  $p_1, p_2, \dots, p_N$ . The highest value of  $\mathcal{M}(p_1, p_2, \dots, p_N)$ , under the same constraint, is achieved when only one  $p_i$  is different from zero,  $p_i = \delta(i - i_0)$ , where  $i_0$  is an arbitrary integer  $1 \leq i_0 \leq N$ . This case corresponds to the maximally concentrated values of  $p_1, p_2, \dots, p_N$ , at a single  $p_{i_0} = 1$ . Therefore, the function  $\mathcal{M}(p_1, p_2, \dots, p_N)$  can be used as a measure of concentration of the set of numbers  $p_1, p_2, \dots, p_N$ , under the unity sum constraint. In general, constraint (3.152) can be included in the function itself by using the form

$$\mathcal{M}(p_1, p_2, \dots, p_N) = \frac{p_1^2 + p_2^2 + \dots + p_N^2}{(p_1 + p_2 + \dots + p_N)^2}.$$

For non-negative  $p_1, p_2, \dots, p_N$  this function has the minimum for  $p_1 = p_2 = \dots = p_N$ , and reaches its maximal value when only one  $p_i$  is different from zero.

In time-frequency analysis this idea has been used in order to measure the time-frequency representations concentration. Several forms of the concentration measure, based on this fundamental idea, are introduced. Applying the previous reasoning to the spectrogram we may write a function for measuring the concentration of the  $\text{SPEC}(n, k)$  and corresponding  $\text{STFT}(n, k)$  as:

$$\mathcal{M}[\text{SPEC}(n, k)] = \frac{\sum_n \sum_k |\text{SPEC}(n, k)|^2}{(\sum_n \sum_k |\text{SPEC}(n, k)|)^2} = \frac{\sum_n \sum_k |\text{STFT}(n, k)|^4}{(\sum_n \sum_k |\text{STFT}(n, k)|^2)^2}. \quad (3.153)$$

This form is just the fourth power of the ratio of the fourth and second order norms of  $\text{STFT}(n, k)$ . High values of  $\mathcal{M}$  indicate that the representation  $\text{STFT}(n, k)$  is highly concentrated, and vice versa. In general, it has been shown (Jones, Parks, Baraniuk, Flandrin, Williams, et al.) that any other ratio of norms  $L_p$  and  $L_q$ ,  $p > q > 1$ , can also be used for measuring the concentration of  $\text{STFT}(n, k)$ .

When there are two or more components (or regions in time-frequency plane of a single component) of approximately equal energies (importance), whose concentrations are very different, the norm based measures will favor the distribution with a “peaky” component, due to raising of distribution values to a high power. It means that if one component (region) is “extremely highly” concentrated, and all the others are “very poorly” concentrated, then the measure will not look for a trade-off, when all components are “well” concentrated. In order to deal with this kind of problems, common in time-frequency analysis, a concentration measure could be applied to smaller, local time-frequency regions with a localization weighting function which determines the region where the concentration is measured.

Another direction to measure time-frequency representation concentration (analyzed by Stanković [50]) comes from a classical definition of the time-limited signal duration, rather than measuring signal peakedness. If a signal  $x(n)$  is time-limited to the interval  $n \in [n_1, n_2 - 1]$ , i.e.,  $x(n) \neq 0$  only for  $n \in [n_1, n_2 - 1]$ , then the duration of  $x(n)$  is  $d = n_2 - n_1$ . It can be written as

$$d = \lim_{p \rightarrow \infty} \sum_n |x(n)|^{1/p} = \|x(n)\|_0, \quad (3.154)$$

where  $\|x(n)\|_0$  denotes the norm zero of signal. The same definition applied to a two-dimensional function  $|P(n, k)|^2 \neq 0$  only for  $(n, k) \in D_x$ , gives

$$N_D = \lim_{p \rightarrow \infty} \sum_n \sum_k |P(n, k)|^{1/p}, \quad (3.155)$$

where  $N_D$  is the number of points within  $D_x$ . In reality, there is no a sharp edge between  $|P(n, k)|^2 \neq 0$  and  $|P(n, k)|^2 = 0$ , so the value of (3.155) could, for very large  $p$ , be sensitive to small values of  $|P(n, k)|^2$ . The robustness may be achieved by using lower order forms, with  $p \geq 1$  in contrast to (3.153) where, in this notation,  $p = 1/2$ .

Therefore, the spectrogram concentration can be measured with the function of the form

$$\mu[\text{SPEC}(n, k)] = \sum_n \sum_k |\text{STFT}(n, k)|^{2/p} \quad (3.156)$$

with  $p > 1$ . Here, lower value of the concentration measure  $\mu$  indicates better concentrated distribution. For example, with  $p = 2$ , it is of the norm one form

$$\mu[\text{SPEC}(n, k)] = \sum_n \sum_k |\text{STFT}(n, k)| = \|\text{STFT}(n, k)\|_1.$$

In the case that variations of amplitude may be expected, an energy normalized version of measure  $\mu[\text{SPEC}(n, k)] = (\sum_n \sum_k |\text{STFT}(n, k)|)^2 / \sum_n \sum_k |\text{STFT}(n, k)|^2$  should be used. Concentration measures were efficiently uses to optimize time-frequency representations parameters.

In the probability theory all results are derived for the probability values  $p_i$ , assuming that  $\sum_i p_i = 1$  and  $p_i \geq 0$ . The same assumptions are made in classical signal analysis for the signal power. Since a general time-frequency representation commonly does not satisfy non-negativity condition, the concentration measures should be carefully used, in that cases.

### 3.03.5.2 Sparse signals

In signal processing representations a signal is transformed from one domain into another. In many cases it happens that a signal that covers whole considered interval in one domain (dense in that domain) is located within much smaller regions in the other domain (sparse in this domain). For example, a discrete time complex sinusoidal signal with  $N$  samples in discrete time domain, is just one sample in the DFT domain (if the frequency is on a grid position). Similarly,  $M$  complex sinusoids covering  $N$  points in discrete time domain, are represented by  $M$  values in frequency domain. This simple illustration leads

to the conclusion that, for a complex signal containing  $M$  complex sinusoids, we do not need  $N$  samples in time domain, to reconstruct  $K$  samples in frequency domain. Then, it would be enough to have  $M < K < N$  arbitrary and independent samples in time domain. Of course, the Fourier domain is just one of possible domains to transform a signal. A signal may not be sparse in the time domain nor in the Fourier domain, but could be, for example, sparse in polynomial Fourier domain or in the fractional Fourier domain (a linear FM signal transforms into a one value in frequency and rate domain) or in the STFT domain. Then, we will also be able to reconstruct the original signal if some samples are missing.

The samples could be missed due to their physical or measurements unavailability. In applications it could happen that some, arbitrary positioned, samples of the signal are so heavily corrupted by disturbances, that it is better to omit them in the analysis. Under some conditions, the processing could be performed with the remaining samples, almost as in the case if missing samples were available. Of course, some a priori information about the nature of the analyzed signal, its sparsity in a known domain, should be used. Compressive sensing is a field dealing with this problem and provides a solution that differs from the classical signal theory approach. Sparsity is one of the main requirements that should be satisfied, in order to efficiently apply the compressive sensing.

**Example 22.** Four samples of a simple complex valued signal in discrete time domain are considered,  $x(0) = 0.866 + j0.5$ ,  $x(1) = -0.5 + j0.866$ ,  $x(2)$  is missing and  $x(3) = 0.5 - j0.866$ . The third sample is missing (not available or heavily corrupted). We know that the signal is sparse in the STFT domain, as well as that its real and imaginary values are somewhere between  $-1$  and  $1$ . Find the value of the third sample that will produce the best concentrated STFT of this signal, for a given instant  $n = 0$ .

For the signal  $x(n)$ , with the missing sample being replaced by  $\alpha + j\beta$ , the STFT is of the form

$$\begin{aligned} \text{STFT}(0, k) &= \sum_{n=0}^3 x(0+n)e^{-jmk\pi/2} \\ &= x(0) + x(1)e^{-jk\pi/2} + (\alpha + j\beta)e^{-jk\pi} + x(3)e^{-j3k\pi/2}, \\ \text{STFT}(0, 0) &= 0.866 + j0.5 + \alpha + j\beta, \\ \text{STFT}(0, 1) &= 2.598 + j1.5 - (\alpha + j\beta), \\ \text{STFT}(0, 2) &= 0.866 + j0.5 + \alpha + j\beta, \\ \text{STFT}(0, 3) &= -0.866 - j0.5 - (\alpha + j\beta). \end{aligned}$$

Using the measure of concentration of the resulting STFT in the form

$$M(\alpha, \beta) = \sum_{k=0}^3 |\text{STFT}(k)| \quad (3.157)$$

and varying parameters  $\alpha$  and  $\beta$  from  $-1$  to  $1$ , with step  $0.001$ , we get the global minimum of  $M(\alpha, \beta)$  at

$$M(-0.866, -0.500) = 4.$$

It means that the missing sample, producing the best concentrated STFT, is

$$x(2) = -0.866 - j0.5.$$

Then,  $\text{STFT}(0, 0) = 0$ ,  $\text{STFT}(0, 1) = 0$ ,  $\text{STFT}(2) = 3.4641 + j2$  and  $\text{STFT}(0, 3) = 0$ .

Now reformulate this problem into matrix form. Denote the full DFT transformation matrix with

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{-j2\pi/N} & e^{-j4\pi/N} & e^{-j6\pi/N} \\ 1 & e^{-j4\pi/N} & e^{-j8\pi/N} & e^{-j12\pi/N} \\ 1 & e^{-j6\pi/N} & e^{-j12\pi/N} & e^{-j18\pi/N} \end{bmatrix},$$

$$\text{STFT}(0) = \mathbf{W}\mathbf{x}(0),$$

$$\mathbf{x}(0) = \mathbf{W}^{-1}\text{STFT}(0),$$

where  $\mathbf{x}(0)$  is a vector column with the signal values and  $\text{STFT}(0)$  is a vector column with the STFT values, for  $n = 0$ . In the case of missing second signal sample, for the STFT coefficients calculation, we get three equations.

*instant 0:*

$$x(0) = \frac{1}{4} (\text{STFT}(0, 0) + \text{STFT}(0, 1) + \text{STFT}(0, 2) + \text{STFT}(0, 3)),$$

*instant 1:*

$$x(1) = \frac{1}{4} (\text{STFT}(0, 0) + \text{STFT}(0, 1)e^{j2\pi/N} + \text{STFT}(0, 2)e^{j4\pi/N} + \text{STFT}(0, 3)e^{j6\pi/N}),$$

*instant 2:*

**Missing value and equation.**

*instant 3:*

$$x(3) = \frac{1}{4} (\text{STFT}(0, 0) + \text{STFT}(0, 1)e^{j8\pi/N} + \text{STFT}(0, 2)e^{j12\pi/N} + \text{STFT}(0, 3)e^{j18\pi/N}).$$

The transformation matrix

$$\mathbf{W}^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{j2\pi/N} & e^{j4\pi/N} & e^{j6\pi/N} \\ 1 & e^{j4\pi/N} & e^{j8\pi/N} & e^{j12\pi/N} \\ 1 & e^{j6\pi/N} & e^{j12\pi/N} & e^{j18\pi/N} \end{bmatrix}$$

is now reduced, by omitting the third row, to

$$\mathbf{A} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{j2\pi/N} & e^{j4\pi/N} & e^{j6\pi/N} \\ 1 & e^{j6\pi/N} & e^{j12\pi/N} & e^{j18\pi/N} \end{bmatrix}$$

with

$$\mathbf{y} = \mathbf{A} \text{STFT}(0). \quad (3.158)$$

Thus, we have three equations with three signal samples  $x(n)$  in vector

$$\mathbf{y} = [x(0), x(1), x(3)]^T$$

and four unknown STFT values  $\text{STFT}(0, k)$  in vector

$$\text{STFT}(0) = [\text{STFT}(0, 0) \ \text{STFT}(0, 1) \ \text{STFT}(0, 2) \ \text{STFT}(0, 3)]^T.$$

By varying the value for  $x(2)$  we have solved undetermined problem, as a minimization problem,

$$\min \|\text{STFT}(0)\| \text{ subject to } \mathbf{y} = \mathbf{A} \text{STFT}(0), \quad (3.159)$$

where for  $\|\text{STFT}(0)\|$ , the absolute value concentration measure is used, presented and discussed in [Chapter 2](#),

$$\|\text{STFT}(0)\| = \sum_{k=0}^{N-1} |\text{STFT}(0, k)|. \quad (3.160)$$

This measure is known as  $l_1$  norm, in notation  $\|\text{STFT}(0)\|_{l_1}$ . Of course, other concentration measures, described in the previous section, are possible, as well. The measure  $N_D = \lim_{p \rightarrow \infty} \sum_n \sum_k |P(n, k)|^{1/p}$  corresponds to  $l_0$  norm.

It is important to note the minimization solution with the  $l_2$  norm, would be trivial, in this case. For this norm, we would attempt to minimize

$$\|\text{STFT}(0)\|_{l_2} = \sum_{k=0}^{N-1} |\text{STFT}(0, k)|^2.$$

According to the Parseval's theorem

$$\|\text{STFT}(0)\|_{l_2} = N \sum_{n=0}^{N-1} |x(n)|^2. \quad (3.161)$$

Since, any value than  $x(n) = 0$  for the non-available (missing) signal samples, would increase  $\|\text{STFT}(0)\|_2$ , then the solution for the non-available samples, with respect to the  $l_2$  norm, is trivial. This was the reason why this norm was not used as a concentration measure.

Based on the previous example we may easily write a general sparse signals' processing formulation.

Let the original signal be  $x(n)$  with  $n = 0, 1, \dots, N - 1$ . Suppose that an arbitrary number of  $N - K$  signal values are missing. Then, the available  $K$  signal values are denoted by vector  $\mathbf{y}$ . If we denote by  $\mathbf{A}$  the inverse transformation matrix  $\mathbf{W}^{-1}$  with omitted rows, corresponding to missing signal values, then for the DFT coefficients we have to solve  $K$  equations

$$\mathbf{y} = \mathbf{A} \text{STFT}(n)$$

with  $N$  unknowns. Thus, we have to find the best concentrated  $\text{STFT}(n)$ , solving the minimization problem

$$\min \|\text{STFT}(n)\| \text{ subject to } \mathbf{y} = \mathbf{A} \text{STFT}(n), \quad (3.162)$$

$$\text{where } \|\text{STFT}(n)\| = \sum_{k=0}^{N-1} |\text{STFT}(n, k)|.$$

The condition that we will obtain a satisfactory result is that the considered signal was sparse in DFT domain.

Consider a signal with unavailable samples, knowing that it is well concentrated and sparse in the Wigner distribution domain. If a small number of samples is missing (eliminated), then we can still calculate the Wigner distribution with missing samples as parameters and find the parameter values (missing samples) that produce the best concentration. For example, it would follow as the ones producing minimum of

$$\mu[\text{WD}(n, k)] = \sum_{k=1}^N \sum_{n=1}^N |\text{WD}(n, k)|^{1/2},$$

since the Wigner distribution is energy distribution (the same form we used for an energetic version of the FT  $|\text{STFT}(n)| = \sum_{k=0}^{N-1} |\text{STFT}(n, k)|$ ). More often a measure with power 1 is used rather than  $1/2$ , corresponding to the  $l_1$  norm. If the exponent is 0 it will be  $l_0$  norm.

Another approach to the Wigner distribution based sparse signals' processing is in the Fourier domain formulation of the Wigner distribution proposed by Flandrin and Borgnat. It is a two-dimensional FT of the ambiguity function

$$\text{AF}(p, l) = \sum_{k=1}^N \sum_{n=1}^N \text{WD}(n, k) e^{-j2\pi(np-kl)/N}. \quad (3.163)$$

This relation was used for efficient definition of the reduced interference distribution. The ambiguity function was multiplied by a low-pass kernel function. Here, we will use a different approach. A large part of the ambiguity domain values will be not multiplied by zero kernel function, but they will be just omitted and considered as unavailable. Only a small region around the ambiguity origin, where we know that the auto-terms are located, is used. Then, we try to reconstruct other values (now missing, since we removed them from analysis). Then, the linear programming formulation is based on

$$\begin{aligned} \min & \left\| \sum_{k=1}^N \sum_{n=1}^N |\text{WD}(n, k)| \right\| \text{ subject to } \\ & \sum_{k=1}^N \sum_{n=1}^N \text{WD}(n, k) e^{-j2\pi(np-kl)/N} = \frac{1}{N} \text{AF}(p, l), \text{ for } K \text{ selected points in } (p, l). \end{aligned} \quad (3.164)$$

Now, we have a complete formulation of the problem within the sparse signal processing framework. We have reduced the formulation of this problem to the linear programming problem with norm one. It can be then solved by using appropriate minimization algorithms.

Possible variation of this approach is when using highly concentrated distributions with reduced cross-terms, like for example the *S*-method. Then, the ambiguity function of this distribution  $\text{AF}_{\text{SM}}(p, l)$  is used in the appropriate minimization problem over  $\text{SM}(n, k)$ . In this case, we make two different efforts to the same direction, to obtain a highly concentrated representation without cross-terms. The marginal properties will be satisfied if within the  $K$  selected points in  $(p, l)$  all the values of  $\text{AF}(p, l)$  along  $p = 0$  or  $l = 0$  are included.

### 3.03.5.3 Compressive sensing and the $L$ -statistics in time-frequency

The compressive sensing (CS) processing of sparse signals, in combination with the  $L$ -statistics, has recently been used in time-frequency analysis to separate a set of time-varying signals from an unknown sparse signal in Fourier domain, by Stanković et al. [51]. A case when these two sets of components overlap in a significant part of the time-frequency plane is considered. Different components of sparse signal intersect non-stationary components at different, time-varying frequency intervals. By removing overlapping points or intervals, a signal with large number of missing measurements in time-frequency plane is formed, corresponding to a CS signal with time and frequency varying CS matrix. The CS observations are taken in the time-frequency domain, rather than in time domain. This case can be encountered in radar signal processing, where in many applications there are micro-Doppler effects that can obscure rigid body points, rendering the radar image ineffective. Similar situation may, for example, appear in communications, when narrowband signals are disturbed by a frequency hopping jammer that is of shorter duration than the considered time-interval. Any other non-stationary jammer, with high values and a large number of crossing points, lead to the same time-frequency varying CS formulation. Since it has been assumed that components overlap in a significant part of time-frequency plane, linear signal transforms are used, in order to avoid dealing with emphatic cross-terms, spread over the entire time-frequency plane. Since, the final aim is the signal reconstruction, by using linear time-frequency representations the components phases will be preserved.

The theory presented here may be considered as a variant of the CS approach in time-frequency domain, being applied to the STFT as a time-frequency representation. The standard CS approach in time domain can be viewed as a special case. In the standard CS definition, some points in time domain are not available. In the time-frequency domain it would mean that these values are not available for some time intervals and all frequencies, corresponding to these intervals. Thus, the standard CS approach in time domain is just a special case of the case with missing arbitrary positioned measurements in the time-frequency domain.

Consider a discrete-time signal  $x(n)$  of the length  $N$  and its discrete Fourier transform (DFT)  $X(k)$ . The STFT, with a rectangular window of the width  $M$ , in a matrix form, can be written as:

$$\text{STFT}_M(n) = \mathbf{W}_M \mathbf{x}(n), \quad (3.165)$$

where  $\text{STFT}_M(n)$  and  $\mathbf{x}(n)$  are vectors:

$$\begin{aligned} \text{STFT}_M(n) &= [\text{STFT}(n, 0), \dots, \text{STFT}(n, M - 1)]^T, \\ \mathbf{x}(n) &= [x(n), x(n + 1), \dots, x(n + M - 1)]^T, \end{aligned} \quad (3.166)$$

and  $\mathbf{W}_M$  is the  $M \times M$  DFT matrix with coefficients  $W_M(m, k) = \exp(-j2\pi km/M)$ . Considering non-overlapping cases, the next STFT will be calculated at instant  $n + M$ , as follows  $\text{STFT}_M(n + M) = \mathbf{W}_M \mathbf{x}(n + M)$ . Combining all STFT vectors in a single equation, we obtain:

$$\text{STFT} = \mathbf{W}_{M,N} \mathbf{x}. \quad (3.167)$$

The resulting STFT matrix is

$$\text{STFT} = \begin{bmatrix} \text{STFT}_M(0) \\ \text{STFT}_M(M) \\ \vdots \\ \text{STFT}_M(N-M) \end{bmatrix}$$

and the coefficients  $N \times N$  matrix is formed as:

$$\mathbf{W}_{M,N} = \begin{bmatrix} \mathbf{W}_M & \mathbf{0}_M & \cdots & \mathbf{0}_M \\ \mathbf{0}_M & \mathbf{W}_M & \cdots & \mathbf{0}_M \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_M & \mathbf{0}_M & \cdots & \mathbf{W}_M \end{bmatrix}, \quad (3.168)$$

where  $\mathbf{0}_M$  is a  $M \times M$  matrix with all 0 elements. The vector  $[\mathbf{x}(0), \mathbf{x}(M), \dots, \mathbf{x}(N-M)]^T$  is the signal vector  $\mathbf{x}$ , since:

$$\begin{aligned} \mathbf{x} &= [\mathbf{x}(0), \mathbf{x}(M), \dots, \mathbf{x}(N-M)]^T \\ &= [x(0), x(1), \dots, x(N-1)]^T. \end{aligned} \quad (3.169)$$

Expressing the above vector in the Fourier domain,

$$\mathbf{x} = \mathbf{W}_N^{-1} \mathbf{X}, \quad (3.170)$$

where  $\mathbf{W}_N^{-1}$  denotes the inverse DFT matrix of the dimension  $N \times N$  and  $\mathbf{X}$  is the DFT vector, we have:

$$\begin{bmatrix} \text{STFT}_M(0) \\ \text{STFT}_M(M) \\ \vdots \\ \text{STFT}_M(N-M) \end{bmatrix} = \mathbf{W}_{M,N} \mathbf{W}_N^{-1} \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}. \quad (3.171)$$

Accordingly, the relation between the STFT and DFT values can be written as follows:

$$\text{STFT} = \mathbf{A} \mathbf{X}. \quad (3.172)$$

Matrix  $\mathbf{A} = \mathbf{W}_{M,N} \mathbf{W}_N^{-1}$  maps the global frequency information in  $\mathbf{X}$  into local frequency information in  $\text{STFT}$ .

As previously explained, we will not use all the STFT points, since a large number of them will be considered as corrupted and thus will be omitted. Consequently, they will be considered as unavailable. This corresponds to the CS approach in the time-frequency domain, where only some of the STFT values (measurements) are available.

The separation of time-frequency points, that can be declared as the CS points (intervals), belonging to the sparse stationary signal, will be done by using the  $L$ -statistics. For each frequency  $k$ , a vector of STFT in time is formed

$$\mathbf{S}_k(n) = [\text{STFT}(n, k), n = 0, M, \dots, N-M].$$

After sorting the elements of  $\mathbf{S}_k(n)$ , for a given frequency  $k$ , we obtain a new ordered set of elements

$$\Psi_k(n) \in \mathbf{S}_k(n)$$

such that

$$|\Psi_k(0)| \leq |\Psi_k(1)| \leq \cdots \leq |\Psi_k(N-M)|.$$

In the  $L$ -statistics form, we omit  $N_Q$  of the highest values of  $\Psi_k(n)$  for each  $k$ . Note that, in some cases, the overlapping components of the same order of amplitude may decrease the intersection value. This happens when a disturbing component of the same value crosses the desired signal with the opposite phase. These cases may also efficiently be treated within the  $L$ -statistics framework, by omitting some of the lowest  $L$ -statistics values, in addition to  $N_Q$  highest values. The omitted values of the STFT are heavily corrupted. Thus, they are declared as useless or unavailable in the CS framework. The rest of the STFT values is considered as CS in the time-frequency plane.

Denote now the vector of available STFT values by  $\mathbf{STFT}_{\text{CS}}$ . The corresponding CS matrix  $\mathbf{A}_{\text{CS}}$  is formed by omitting the rows corresponding to the omitted STFT values. We want to reconstruct the original sparse stationary signal, since it produces the best concentrated FT  $X(k)$ . Therefore, the corresponding minimization problem can be defined as follows:

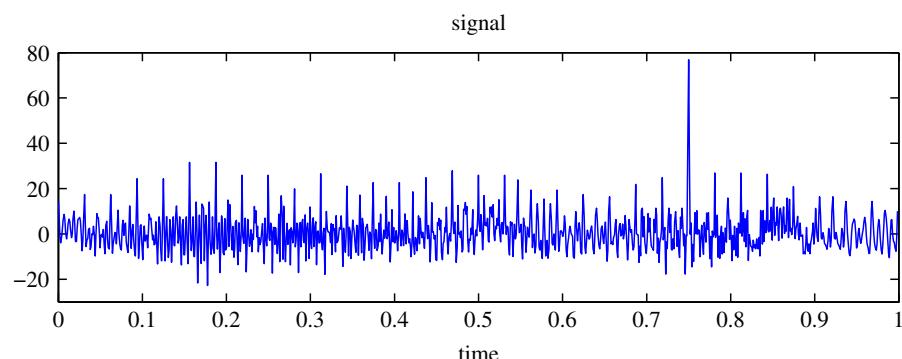
$$\begin{aligned} \min \|\mathbf{X}\| &= \min \sum_{k=0}^{N-1} |X(k)| \\ \text{subject to } \mathbf{STFT}_{\text{CS}} &= \mathbf{A}_{\text{CS}} \mathbf{X}. \end{aligned} \quad (3.173)$$

Thus, based on the  $\mathbf{STFT}_{\text{CS}}$  values, we are going to reconstruct the missing values such to provide minimal  $\sum_{k=0}^{N-1} |X(k)|$ . This is a well known CS formulation of the problem that can be solved by using linear programming tools.

**Example 23.** Consider a signal that consists four complex sinusoids

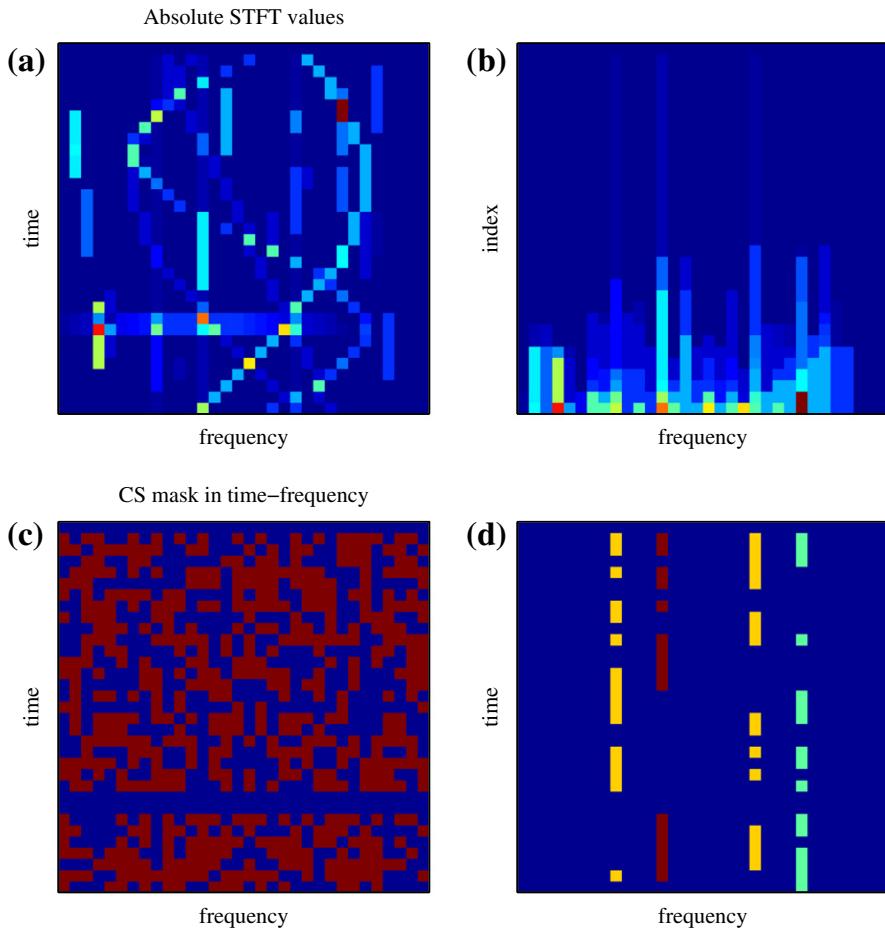
$$x(n) = e^{j256\pi n/N} + 1.5e^{-j2568\pi n/N+j\pi/8} + 0.7e^{j512\pi n/N+j\pi/4} + e^{-j512\pi n/N-j\pi/3}$$

with non-stationary disturbance in form of several short duration sinusoidal signals (some of them are at the same frequencies as the stationary sinusoids) and four sinusoidally modulated signals. The signal in time domain is shown in Figure 3.39.



**FIGURE 3.39**

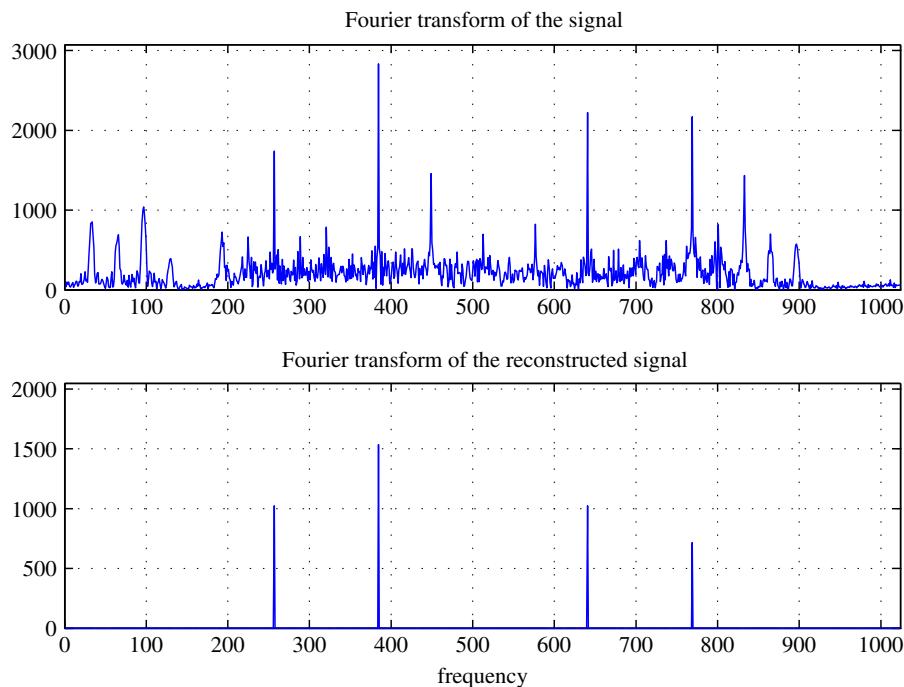
Signal composed of a sparse part and a non-stationary part.

**FIGURE 3.40**

(a) The STFT of the composite signal. (b) Sorted values of the absolute STFT. (c) The compressive sensing mask corresponding to the  $L$ -statistics based STFT values. (d) The STFT values that remain after applying the compressive sensing mask on the absolute values of the STFT.

Its STFT is calculated for  $N = 1024$  and  $M = 32$ . The STFT of the signal is presented in Figure 3.40 (left). After the  $L$ -statistics is performed and 50% of the largest values are removed, along with 10% of the smallest values, the CS form of the STFT, with only 40% of the original values is obtained, Figure 3.40 (middle). The STFT values that remained after the  $L$ -statistics based removal are shown in Figure 3.40 (right).

The reconstruction is performed, based on the STFT values from Figure 3.40 (right). The reconstructed signal's FT is equal to the original FT, preserving amplitude and phase. Its amplitude is shown in Figure 3.41 (bottom), along with the original FT of the signal Figure 3.41 (top).

**FIGURE 3.41**

Reconstructed Fourier transform by using the compressive sensing method of the STFT presented in the previous figure corresponding to the sparse part of the composite signal (bottom) and the Fourier transform of the original signal (top).

## 3.03.6 Examples of time-frequency analysis applications

In this section we present just a few of many possible applications of time-frequency analysis [6, 12, 30, 52–62]. We will explain the basic principles of the application area and its formulation within the time-frequency framework. It will be emphasized what benefits may be achieved by using the time-frequency representations, over the classical tools. Since many of the presented time-frequency tools may be then applied on such a formulated problem, it is left to the reader to apply other time-frequency tools, with their advantages and drawbacks.

### 3.03.6.1 Time-frequency radar signal processing

When radar transmits an electromagnetic signal to a target, the signal reflects from it and returns to radar. The reflected signal, as compared to the transmitted signal, is delayed, changed in amplitude, and possibly shifted in frequency. These parameters of the received signal contain information about the target's characteristics. For example, delay is related to the target's distance from the radar, while

the target's velocity is related to the shift in frequency. Inverse synthetic aperture radar (ISAR) is a method for obtaining high resolution image of a target based on the change in viewing angle of the target with respect to the fixed radar. Rotation, as a result of this angle change, introduces addition velocity proportional to the rotation speed and the distance from the rotation center. Component of the velocity in direction of the radar-target line is proportional to the normal distance of the point from the center of rotation. Thus, in the case of ISAR, the distance and the velocity locate the target point in the range-cross range domain. Since the range and cross range information are contained in two dimensional sinusoids with corresponding frequencies, common technique used for the ISAR signal analysis is the two dimensional FT. The application of the FT application on the ISAR signal of a point target results in a highly concentrated function at a point whose position corresponds to the range and cross range values. Similar situation appears in synthetic aperture radar (SAR) signal processing.

The basic property in spectral analysis is that higher concentration of the FT is achieved by using longer signal sequences. However, within longer time intervals the target point velocity changes its direction, meaning that even constant rotation speed results in the velocity projection changes. These changes cause corresponding frequency changes, that spread FT, blurring information about the cross range. The rotation itself can be non-constant, increasing the radar image distortion. In addition to these disturbances in the radar image, target motion can also be three dimensional, changing the velocity projection along the target-radar line in a very complex way. Standard techniques for these kind of problems are in movement compensation by using time-frequency analysis base tools.

Another problem in radar imaging where the time-frequency analysis is used is the micro-Doppler effect processing. Namely, very fast moving parts of a radar target can cause micro-Doppler effect that can cover rigid parts of a target or degrade the ISAR image. Separation of patterns caused by rigid parts of the target from the patterns caused by fast moving parts is an interesting area of time-frequency analysis application.

Micro-Doppler effect appears in the ISAR radar imaging when the target has one or more very fast moving parts. This effect can decrease the readability of radar images. However, the micro-Doppler effect, at the same time, carries information about the features of moving parts (type, velocity, size, etc.).

Procedures for separation of signals coming from the rigid (slow moving) body and fast moving parts are based on the time-frequency (more accurately space/spatial-frequency) analysis of the return signal within the CIT.

Based on the order statistics of time-frequency representations, it is possible to make a decision if the components belong to the rigid body or to the rotating target parts. Another approach is based on the Radon transform of the time-frequency representation of signal returned from rotating parts which produce sinusoidal FM signals. A time-frequency representation of the sinusoidal FM signal can be concentrated in a single point by applying the inverse Radon transform to the time-frequency representation. In this way, patterns of the rotating parts can be easily extracted from the signal mixture.

Another application of time-frequency analysis in radars is related to the spacial localization of sources by passive sensor array. Direction-of-arrival (DOA) is the key parameter in this problem. Most of the DOA estimators are based on the estimates of the data covariance matrix. In the case when phase of the input is not linear, then the time-frequency distributions may be used to analyze the data snapshots across the array. This kind of distributions is known as spatial time-frequency distributions. They are able to spread the noise energy over the entire time-frequency plane, while localizing the energy of the input signals, in the cases of non-linear phases. Better localization of signals permits the division of

the time-frequency domain into smaller regions, each containing fewer signals than the input signal on the array. It relaxes the condition on the array aperture and simplifies optimization procedures.

### 3.03.6.2 Biomedical signal analysis

Biomedical signals usually contain interferences of different nature. The separation of undesirable content from the considered signal is easier in joint-time-frequency plane. In some cases the detection of frequency content changes and irregularities carry the most important information about the underlying process. Thus, analysis of highly non-stationary signals in biomedicine could benefit from using time-frequency tools. Such examples are EEG analysis in epilepsy patients, ECG analysis of cardiac abnormalities, identification of genetic abnormalities in brain tumors, early detection of multiple sclerosis lesions, study of pediatric disorders, or analysis of respiration sounds in asthmatic patients.

### 3.03.6.3 Seismic signal analysis

Earthquakes analysis is very important since they cause many catastrophes all over the world. One of the application in the analysis of seismic signals is in their reliable detection. Due to the high probability of false earthquake alarms when the seismometers detect non-seismic signals, the time-frequency analysis can be used for seismic signal separation and detection. It is based on the instantaneous frequency analysis. Analyzing the energy of seismic signal it is possible to decide and send an earthquake alarm if the energy above a specific level.

Seismic signals are also used in the analysis of propagation media, with application, for example, in gas and oil detection. Seismic wave is a periodic wave which can transmit energy through the media without causing their permanent deformation. The p-waves and s-waves are the waves used for the time-frequency analysis. Seismic reflection patterns can provide information of subsurface imaging by using the time-frequency tools (seismic stratigraphy). Analyzing the received reflected signal and its energy at the surface, it is possible to observe the characteristics of subsurfaces where the signal propagated. The subsurface imaging is widely used in gas and oil detection. A quantitative analysis in seismic reflection patterns, in order to get more information of geology by seismic reflection pattern, is possible by using the time-frequency analysis.

### 3.03.6.4 Car engine signal analysis

Structure-borne sound signals and more seldom pressure signals are used for efficient combustion control of spark-ignition engines. This control can increase efficiency, reduce pollution and noise, and protect against knocks. A knock is an abnormal combustion that causes a rapid rise of the temperature and pressure. Its detection is an important problem since a frequent knock occurrence can destroy the engine or significantly degrade its performances. By measuring the pressure at a suitable point inside the cylinder we can observe the combustions. However pressure sensors are expensive, difficult to mount and not robust enough. These are the reasons why their application is mainly limited to test beds. Sound signals can be considered as time-varying filtered versions of the pressure and can be used for the observation of diagnostic parameters, as resonance frequencies that are function of temperature, and resonance energies that indicate knock. The application of acceleration sensors on the surface of the

engine is easy and economical, but sound signals are superimposed with mechanical noise that can significantly influence the analysis.

Such sound and pressure signals are highly non-stationary, therefore classical signal analysis tools in time domain or in frequency domains are no efficient here. Even the short-time Fourier transform and its energetic version spectrogram, as extensions of the Fourier analysis to the non-stationary problems, cannot be used due to high non-stationary effects in the car engine signals. These signals require higher resolution joint time-frequency analysis. It has been shown that pressure signals and sound signals can be considered as frequency modulated multi-component signals with random amplitudes and phases of the components when low frequency parts are neglected. Due to cyclostationarity and the property that the components of these signals are mutually not correlated, it has been found that the Wigner spectrum can be used as an efficient time-frequency tool for their analysis. The problem of cross-terms in the Wigner distribution was resolved by averaging over pressure or sound signals of different combustion cycles of the engine under similar working conditions. Since the components are not correlated the cross-terms disappear and theoretically, using an infinite number of combustions, the mean of Wigner distributions converges to the Wigner-Ville spectrum containing the auto-terms only. The main disadvantage of this approach is that the elimination of cross-terms requires a large number of combustions meaning a long observation time and can mask the effects in a single combustion or a decision based on the analysis can be too late for an action. Reduced interference time-frequency distributions are efficiently used for a single realization based analysis and knock detection.

### 3.03.6.5 Velocities of moving objects in video sequence

In some video surveillance applications it is important to detect and analyze object motion in a video sequence. A common approach for this problem is based on frame projections and the SLIDE (subspace-based line detection) algorithm. A key step of the SLIDE algorithm is to map a line into a complex sinusoid. By using frame projections on coordinate axes the velocity of an object is mapped into an image, containing lines. In order to transform line parameters to the complex sinusoid parameters, the SLIDE algorithm with constant  $\mu$ -propagation is applied. Very accurate results are obtained when objects have uniform velocities. However, for non-uniform velocities the SLIDE algorithm produces a frequency modulated signal. This is the reason why the instantaneous frequency estimators, based on the time-frequency distributions, are introduced in this area and efficiently used.

### 3.03.6.6 Time-variant filtering

Speech signals are highly non-stationary, with a wide dynamic range of multiple frequency components in the short-time spectra. Time-frequency distributions have been introduced for the analysis frequency components as a function of time of non-stationary signals. The most common time-frequency representation, the spectrogram, is characterized by a trade-off between time and frequency resolutions. Consequently, the development of other quadratic time-frequency distributions for representation and processing of non-stationary signals is an interesting challenge. For example, a possible application is the Wiener optimum filtering of speech signals corrupted by noise. A more accurate estimation of the signal spectrum yields higher suppression of noise components.

Because of the non-stationary nature of speech signals, statistically optimum filtering requires time-variant filtering methods. Filtering in the time-frequency domain could be advantageous compared to

separate filtering in the time or frequency domain. Since there exists no unique definition of time-frequency spectra, many approaches for time-variant filtering have been proposed. Zadeh suggested to use the Rihaczek distribution. However, this time-frequency spectrum is complex valued and badly concentrated in time. Therefore, filtering in the time-frequency domain has been redefined using the Wigner distribution, using the Weyl correspondence. The time-variant transfer function has been defined as the Weyl symbol mapping of the impulse response into the time-frequency plane. The Wigner spectrum is used in order to average out the cross-terms. For single-realization cases, reduced interference distributions are used.

### 3.03.6.7 Interference rejection in spread spectrum systems

Spread spectrum is a transmission coding technique used in digital telecommunication systems, where pseudo-noise, or pseudo-random code, independent of the information data, is employed as a modulation waveform. The pseudo-random code significantly expands the bandwidth of original signal. The spread signal has a lower power density, but the same total power. At the receiver side signal is “despreaded” using the synchronized replica of the pseudo-noise code. Spread spectrum technology has been recognized as a good alternative to both frequency division multiple access (FDMA) and time division multiple access (TDMA) for the cellular systems. The most common spread spectrum systems are of the direct sequence or frequency hopping type. Direct sequence spread spectrum systems employ a high-speed code sequence to introduce rapid phase transitions into the carrier containing the data. The result of modulating the carrier is a signal centered at the carrier frequency, but with main lobe bandwidth significantly wider than the original bandwidth. The frequency hopping spread spectrum achieves the band spreading by using the pseudo-noise sequence to pseudo-randomly hop the carrier frequency.

While the influence of low power interfering signals is significantly reduced by despreading process at the receiver, in case of very high power interferences preprocessing is required. This is a common case, when the interference stations are much closer to the receiver than the signal transmitting station.

Different methods have been proposed for rejection or mitigation of interferences of this kind, in order to improve robustness of spread spectrum systems and more reliable receiving and decoding of the useful signal. Amin proposed an open-loop adaptive filtering method for the case of narrowband jammer. Wang and Amin applied multiple-zero FIR filters whose notch is in synchronization with the jammer instantaneous frequency, in order to remove the jammer power at every time sample. Barbarossa and Scaglione have proposed a method based on a generalized Wigner-Hough transform. They characterized linear and sinusoidal jammers by the appropriate parametric models. Suleesathira and Chaparro used a method for interference mitigation based on the evolutionary and Hough transform. The fractional Fourier transform for the case of linear chirp signals can improve the presentation and an overall bit error performance of the receiver when the angular parameter of the transform matches chirp rate of the interferences. A non-parametric approach for the jammer excision by using local polynomial Fourier transform, which is linear with respect to the signal is also used. The optimal order of the local polynomial Fourier transform was determined and calculated for each considered instant with appropriate optimization of the selected order transform parameters. The jammer was represented in the domain of its best concentration. Time-varying filtering is then implemented in that domain. In this way the local polynomial Fourier transform based method can be used for a general, including non-linear FM type of interferences.

### 3.03.6.8 Watermarking in the space/spatial-frequency domain

Digital watermarking has been rapidly developing in the last few years due to the widespread use of multimedia contents. It can provide an image copyright protection, i.e., identification of its owner and authorized distributor. The most commonly used approaches for watermarking are either by embedding in the spatial domain, or in the transformation (frequency) domain. Regardless the watermarking approach, it is desirable to ensure that watermarking satisfies the following important properties: (a) it is perceptually invisible; (b) watermark must be robust to the various image processing algorithms, such as common geometric distortion (rotation, translation, cropping), resampling, filtering, and compression; (c) detection of the watermark by copyright owner should be possible without the original image.

A watermarking scheme in the space/spatial-frequency domain is also proposed and used. Two dimensional chirp signals are used as watermarks, due to their geometrical symmetry and robustness to stationary filtering. An ideal space/spatial-frequency representation, for this type of signals, is achieved by using the two-dimensional Wigner distribution. In order to additionally emphasize the watermark, with respect to the Wigner distribution of the original image, the Wigner distribution projections (two dimensional Radon Wigner distribution) are used. In this way the maximum watermark projection will be dominant, as compared to the projections of the Wigner distribution, of the original image. It results in an efficient watermark detection procedure.

---

## References and Further Reading

- [1] F. Auger, F. Hlawatsch, Time-Frequency Analysis, Wiley-ISTE, November 2008.
- [2] L. Cohen, Time-Frequency Analysis, Prentice-Hall, 1995.
- [3] D. Gabor, Theory of communications, J. Inst. Electr. Eng. 93 (1946) 423–457.
- [4] F. Hlawatch, G.F. Boudreault-Bartels, Line and quadratic time-frequency signal representations, IEEE Signal Process. Mag. (1992) 21–67.
- [5] O. Rioul, M. Vetterli, Wavelets and signal processing, IEEE Signal Process. Mag. (1991) 14–38.
- [6] L. Stankovic, M. Dakovic, T. Thayaparan, Time-Frequency Signal Analysis with Applications, Artech House, Boston, April 2013.
- [7] L.E. Atlas, Y. Zhao, R.J. Marks II, The use of cone shape kernels for generalized time-frequency representations of nonstationary signals, IEEE Trans. Acoust. Speech Signal Process. 38 (1990) 1084–1091.
- [8] M.J. Bastiaans, T. Alieva, L. Stanković, On rotated time-frequency kernels, IEEE Signal Proc. Lett. 9 (11) (2002) 378–381.
- [9] T.A.C.M. Claasen, W.F.G. Mecklenbrauker, The Wigner distribution—a tool for time frequency signal analysis, Parts I, II and III, Phillips J. Res. 35 (3/4/5/6) (1980) 217–250, 276–300, 372–389.
- [10] P. Flandrin, Temps-Fréquence, Paris, Hermès, 1993.
- [11] W. Mecklenbrauker, The Wigner Distribution—Theory and Applications in Signal Processing, Elsevier Science, Amsterdam, 1992.
- [12] W. Martin, P. Flandrin, Wigner-Will spectral analysis of nonstationary processes, IEEE Trans. ASSP 33 (6) (1985) 1461–1470.
- [13] W.F.G. Mecklenbrauker, F. Hlawatsch (Eds.), The Wigner Distributions—Theory and Applications in Signal Processing, Elsevier, 1997.
- [14] S. Qian, Introduction to Time-Frequency and Wavelet Transforms, Prentice Hall, 2001.
- [15] A.W. Rihaczek, Signal energy distribution in time and frequency, IEEE Trans. Info. Theory 14 (1968) 369–374.
- [16] J. Ville, Theorie et applications de la notion de signal analytique, Cables es Transmission 2 (1) (1946) 61–74.

- [17] P.E. Wigner, On the quantum correction for thermodynamic equilibrium, *Phys. Rev.* 40 (1932) 246–254.
- [18] Y.M. Zhu, F. Peyrin, R. Goutte, Transformation de Wigner-Ville: description d'un nouvel outil de traitement du signal et des images, *Ann. Telecommun.* 42 (3–4) (1987) 105–117.
- [19] B. Boashash, B. Ristić, Polynomial time-frequency distributions and time-varying higher order spectra: applications to analysis of multi-component FM signals and to treatment of multiplicative noise, *Signal Process.* 67 (1) (1998) 1–23.
- [20] J.R. Fonollosa, C.L. Nikias, Wigner higher order moment spectra: definitions, properties, computation and application to transient signal analysis, *IEEE Trans. Signal Process.* 41 (1993) 245–266.
- [21] B. Porat, B. Friedlander, Asymptotic analysis of the high-order ambiguity function for parameter estimation of the polynomial-phase signal, *IEEE Trans. Info. Theory* 42 (1996) 995–1001.
- [22] B. Ristic, B. Boashash, Relationship between the polynomial and higher order Wigner-Ville distribution, *IEEE Signal Process. Lett.* 2 (12) (1995) 227–229.
- [23] L. Stanković, A method for time-frequency analysis, *IEEE Trans. Signal Process.* 42 (1) (1994) 225–229.
- [24] B. Boashash, Estimating and interpreting the instantaneous frequency of a signal Part 1, *IEEE Proc.* 80 (4) (1992) 519–538.
- [25] A.J.E.M. Janssen, On the locus and spread of pseudo density functions in the time-frequency plane, *Philips J. Res.* 37 (1982) 79–110.
- [26] V. Katkovnik, A new form of the Fourier transform for time-frequency estimation, *Signal Process.* 47 (2) (1995) 187–200.
- [27] I. Daubechies, *Ten Lecture on Wavelets*, SIAM, Philadelphia, Pennsylvania, 1992.
- [28] M. Vetterli, J. Kovacević, *Wavelets and Subband Coding*, Prentice Hall, 1994.
- [29] P. Goncalves, R.G. Baraniuk, Pseudo affine Wigner distributions: definition and kernel formulation, *IEEE Trans. Signal Process.* 46 (6) (1998) 1505–1517.
- [30] A. Papandreou-Suppappa, *Applications in Time-Frequency Signal Processing*, CRC Press, 2002.
- [31] O. Rioul, P. Flandrin, Time-scale energy distributions: a general class extending wavelet transforms, *IEEE Trans. Signal Process.* 37 (7) (1992) 1746–1757.
- [32] H.M. Ozatkas, N. Erkaya, M.A. Kutay, Effect of fractional Fourier transformation on time-frequency distributions belonging to the Cohen class, *IEEE Signal Process. Lett.* 3 (2) (1996) 40–41.
- [33] X.G. Xia, On bandlimited signals with fractional Fourier transform, *IEEE Signal. Process. Lett.* 3 (3) (1996) 72–74.
- [34] L.B. Almeida, Product and convolution theorems for the fractional Fourier transform, *IEEE Signal Process. Lett.* 4 (1) (1997) 15–17.
- [35] L. Stanković, An analysis of some time-frequency and time-scale distributions, *Ann. Telecommun.* 49 (9–10) 1 (1994) 505–517.
- [36] M.G. Amin, Minimum variance time-frequency distribution kernels for signals in additive noise, *IEEE Trans. Signal Process.* 44 (9) (1996) 2352–2356.
- [37] L. Cohen, Time-Frequency Analysis, *Signal Process. Mag.* (1999) 22–28.
- [38] R.G. Baraniuk, D.L. Jones, Signal-dependent time-frequency analysis using radially-Gaussian kernel, *IEEE Trans. Signal Process.* 41 (3) (1993) 263–284.
- [39] D.L. Jones, R.G. Baraniuk, An adaptive optimal-kernel time-frequency representation, *IEEE Trans. Signal Process.* 43 (10) (1995) 2361–2372.
- [40] M.G. Amin, Spectral decomposition of time-frequency distribution kernels, *IEEE Trans. Signal Process.* 42 (5) (1994) 1156–1165.
- [41] L.L. Scharf, B. Friedlander, Toeplitz and Hankel kernels for estimating time-varying spectra of discrete-time random processes, *IEEE Trans. Signal Process.* 49 (1) (2001) 179–189.
- [42] F. Auger, P. Flandrin, Improving the readability of time-frequency and time-scale representations by the reassignment method, *IEEE Trans. Signal Process.* 43 (5) (1995) 1068–1089.

- [43] B. Boashash, P. O’Shea, Polynomial Wigner-Ville distributions and their relationship to time-varying higher order spectra, *IEEE Trans. Signal Process.* 42 (1) (1994) 216–220.
- [44] S. Barbarossa, Analysis of multi-component LFM signals by a combined Wigner-Hough transform, *IEEE Trans. Signal Process.* 43 (6) (1995) 1511–1515.
- [45] B. Friedlander, J.M. Francos, Estimation of amplitude and phase parameters of multi-component signals, *IEEE Trans. Signal Process.* 43 (4) (1995) 917–927.
- [46] S. Peleg, B. Porat, Estimation and classification of polynomial phase signals, *IEEE Trans. Info. Theory* (3) (1991) 422–430.
- [47] R. Baraniuk, Compressive sensing, *IEEE Signal Process. Mag.* 24 (4) (2007) 118–121.
- [48] P. Flandrin, P. Borgnat, Time-frequency energy distributions meet compressed sensing, *IEEE Trans. Signal Process.* 58 (6) (2010) 2974–2982.
- [49] S. Stankovic, I. Orovic, E. Sejdic, *Multimedia Signals and Systems*, Springer, 2012.
- [50] L. Stankovic, A measure of some time-frequency distributions concentration, *Signal Process.* 81 (3) (2001) 621–631.
- [51] L. Stanković, I. Orović, S. Stanković, M. Amin, Compressive sensing based separation of non-stationary and stationary signals Overlapping in time-frequency, *IEEE Trans. Signal Process.* 61 (2013) in print.
- [52] G.F. Boudreaux-Bartels, T.W. Parks, Time-varying filtering and signal estimation using Wigner distribution synthesis techniques, *IEEE Trans. Acoust. Speech and Signal Process.*, 34 (3) (1986) 442–451.
- [53] V.C. Chen, *The Micro-Doppler Effect in Radar*, Artech House, 2011.
- [54] I. Djurović, L. Stanković, J.F. Bohme, Robust  $L$ -estimation based forms of signal transforms and time-frequency representations, *IEEE Trans. Signal Process.* 51 (7) (2003) 1753–1761.
- [55] V. Katkovnik, L. Stanković, Instantaneous frequency estimation using the Wigner distribution with varying and data-driven window length, *IEEE Trans. Signal Process.* 46 (9) (1998) 2315–2326.
- [56] C. Richard, Time-frequency-based detection using discrete-time discrete-frequency Wigner distributions, *IEEE Trans. Signal Process.* 50 (9) (2002) 2170–2176.
- [57] E. Sejdic, I. Djurovic, J. Jiang, L. Stankovic, *Time-Frequency Based Feature Extraction and Classification*, VDM Verlag, 2009.
- [58] L. Stankovic, Performance analysis of the adaptive algorithm for bias-to-variance trade-off, *IEEE Trans. Signal Process.* 52 (5) (2004) 1228–1234.
- [59] L. Stankovic, T. Thayaparan, M. Dakovic, Signal decomposition by using the s-method with application to the analysis of HF radar signals in sea-clutter, *IEEE Trans. Signal Process.* 54 (11) (2006) 4332–4342.
- [60] L. Stanković, Analysis of noise in time-frequency distributions, *IEEE Signal Process. Lett.* 9 (9) (2002) 286–289.
- [61] S. Stanković, I. Djurović, I. Pitas, Watermarking in the space/spatial-frequency domain using two-dimensional Radon-Wigner distribution, *IEEE Trans. Image Process.* 10 (4) (2001) 650–658.
- [62] L. Stanković, I. Orović, S. Stanković, M. Amin, Robust time-frequency analysis based on the l-estimation and compressive sensing, *IEEE Signal Process. Lett.* 20 5 (2013) 499–502.

# Bayesian Computational Methods in Signal Processing

# 4

Simon Godsill

*Signal Processing and Communications Laboratory, Department of Engineering, University of Cambridge, UK*

---

## 3.04.1 Introduction

Over recent decades Bayesian computational techniques have risen from relative obscurity to becoming one of the principal means of solving complex statistical signal processing problems. Methodology is now mature and sophisticated. This article aims to provide a brief and basic introduction for those starting on the topic, with the aim of inspiring further research work in this vibrant area.

The paper is structured as follows. Section 3.04.2 covers Bayesian parameter estimation, with a worked case study in Bayesian linear Gaussian models. This section also introduces model uncertainty from a Bayesian perspective. Section 3.04.3 introduces some of the computational tools available for batch inference, including the expectation-maximization (EM) algorithm and the Markov chain Monte Carlo (MCMC) algorithm. Section 3.04.4 moves on to sequential inference problems, covering linear state-space models, Kalman filters, nonlinear state-space models, and particle filtering.

---

## 3.04.2 Parameter estimation

In parameter estimation we suppose that a random process  $\{X_n\}$  depends in some well-defined stochastic manner upon an unobserved parameter vector  $\theta$ . If we observe  $N$  data points from the random process, we can form a vector  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ . The parameter estimation problem is to deduce the value of  $\theta$  from the observations  $\mathbf{x}$ . In general it will not be possible to deduce the parameters exactly from a finite number of data points since the process is random, but various schemes will be described which achieve different levels of performance depending upon the amount of data available and the amount of prior information available regarding  $\theta$ .

We now define a general parametric model, the linear Gaussian model, which will be referred to frequently in this and subsequent sections. This model will be used as a motivating example in much of the work in this section, with details of the calculations required to perform the various aspects of inference in classical and Bayesian settings. This model is a fundamental building block for many more complex and sophisticated non-linear or non-Gaussian models in signal processing, and so these fundamentals will be referred to numerous times later in the text. However, although we use these models as a motivating example throughout the early sections, it should be borne in mind that there are indeed many classes of model in practical application that have none, or very little, linear Gaussian structure, and for these results specialized techniques are required for inference; none of the analytic results here

will apply in these cases, but the fundamental structure of the Bayesian inference methodology does indeed carry through unchanged, and inference can still be successfully performed, albeit with more effort and typically greater computational burden. The Monte Carlo methods described later, and in particular the particle filtering and Markov chain Monte Carlo (MCMC) approaches, are well suited to inference in the hardest models with no apparent linear or Gaussian structure.

Another class of model that is amenable to certain simple analytic computations beyond the linear Gaussian class is the discrete Markov chain model and its counterpart the hidden Markov model (HMM). This model, not covered in this tutorial, can readily be combined with linear Gaussian structures to yield sophisticated switching models that can be elegantly inferred within the fully Bayesian framework.

### 3.04.2.1 The linear Gaussian model

In the general model it is assumed that the data  $\mathbf{x}$  are generated as a function of the parameters  $\boldsymbol{\theta}$  with a random modeling error term  $e_n$ :

$$x_n = g_n(\boldsymbol{\theta}, e_n),$$

where  $g_n(\cdot)$  is a deterministic and possibly non-linear function of the parameter  $\boldsymbol{\theta}$  and the random error, or “disturbance”  $e_n$ . Now we will consider the important special case where the function is linear and the error is additive, so we may write

$$x_n = \mathbf{g}_n^T \boldsymbol{\theta} + e_n,$$

where  $\mathbf{g}_n$  is a  $P$ -dimensional column vector, and the expression may be written for the whole vector  $\mathbf{x}$  as

$$\mathbf{x} = \mathbf{G}\boldsymbol{\theta} + \mathbf{e}, \quad (4.1)$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_1^T \\ \mathbf{g}_2^T \\ \vdots \\ \mathbf{g}_N^T \end{bmatrix} = [\mathbf{h}_1 \ \mathbf{h}_2 \ \dots \ \mathbf{h}_P].$$

The columns  $\mathbf{h}_i$  of  $\mathbf{G}$  form a fixed basis vector representation of the data, for example sinusoids of different frequencies in a signal which is known to be made up of pure frequency components in noise. A variant of this model will be seen later in the section, the autoregressive (AR) model, in which previous data points are used to predict the current data point  $x_n$ . The error sequence  $\mathbf{e}$  will usually (but not necessarily) be assumed drawn from an independent, identically distributed (i.i.d.) noise distribution, that is,

$$p(\mathbf{e}) = p_e(e_1) \cdot p_e(e_2) \cdots p_e(e_N),$$

where  $p_e(\cdot)$  denotes some noise distribution which is identical for all  $n$ . Note that some powerful models can be constructed using heterogeneous noise sources, in which the distribution  $p(e_n)$  can vary with  $n$ . This might be for the purpose of modeling a time-varying noise term, or for modeling a heavy-tailed noise distribution through the introduction of additional latent variables at each time point.<sup>1</sup>  $\{e_n\}$  can be

---

<sup>1</sup> Whenever the context makes it unambiguous we will adopt from now on a notation  $p(\cdot)$  to denote both probability density functions (PDFs) and probability mass functions (PMFs) for random variables and vectors.

viewed as a modeling error, innovation or observation noise, depending upon the type of model. When  $p_e(\cdot)$  is the normal distribution and  $g_n$  is a linear function, we have the linear Gaussian model.

There are many examples of the use of such a linear Gaussian modeling framework. Two very simple and standard cases are the sinusoidal model and the autoregressive (AR) model.

**Example.** (Sinusoidal model) If we write a single sinusoid as the sum of sine and cosine components we have:

$$x_n = a \cos(\omega n) + b \sin(\omega n).$$

Thus we can form a second order ( $P = 2$ ) linear model from this if we take:

$$\mathbf{G} = [\mathbf{c}(\omega) \quad \mathbf{s}(\omega)], \quad \boldsymbol{\theta} = \begin{bmatrix} a \\ b \end{bmatrix},$$

where

$$\mathbf{c}(\omega) = [\cos(\omega), \quad \cos(2\omega), \quad \cos(N\omega)]^T$$

and

$$\mathbf{s}(\omega) = [\sin(\omega), \quad \sin(2\omega), \quad \sin(N\omega)]^T.$$

And similarly, if the model is composed of  $J$  sinusoids at different frequencies  $\omega_j$  we have

$$x_n = \sum_j a_j \cos(\omega_j n) + b_j \sin(\omega_j n)$$

and the linear model expression is

$$\mathbf{G} = [\mathbf{c}(\omega_1) \quad \mathbf{s}(\omega_1) \quad \mathbf{c}(\omega_2) \quad \mathbf{s}(\omega_2) \quad \cdots \quad \mathbf{c}(\omega_J) \quad \mathbf{s}(\omega_J)], \quad \boldsymbol{\theta} = \begin{bmatrix} a_1 \\ b_1 \\ a_2 \\ b_2 \\ \vdots \\ a_J \\ b_J \end{bmatrix}.$$

**Example.** (Autoregressive (AR) model) The AR model is a standard time series model based on an all-pole filtered version of the noise residual:

$$x_n = \sum_{i=1}^P a_i x_{n-i} + e_n. \quad (4.2)$$

The coefficients  $\{a_i; i = 1 \dots P\}$  are the filter coefficients of the all-pole filter, henceforth referred to as the AR parameters, and  $P$ , the number of coefficients, is the order of the AR process. The AR model formulation is closely related to the linear prediction framework used in many fields of signal processing (see e.g., [1,2]). The AR modeling equation of (4.2) is now rewritten for the block of  $N$  data samples as

$$\mathbf{x} = \mathbf{G}\mathbf{a} + \mathbf{e}, \quad (4.3)$$

where  $\mathbf{e}$  is the vector of  $(N - P)$  error values and the  $((N - P) \times P)$  matrix  $\mathbf{G}$  is given by

$$\mathbf{G} = \begin{bmatrix} x_P & x_{P-1} & \cdots & x_2 & x_1 \\ x_{P+1} & x_P & \cdots & x_3 & x_2 \\ \vdots & \ddots & & & \vdots \\ x_{N-2} & x_{N-3} & \cdots & x_{N-P} & x_{N-P-1} \\ x_{N-1} & x_{N-2} & \cdots & x_{N-P+1} & x_{N-P} \end{bmatrix}. \quad (4.4)$$

### 3.04.2.2 Maximum likelihood (ML) estimation

Prior to consideration of Bayesian approaches, we first present the maximum likelihood (ML) estimator, which treats the parameters as unknown constants about which we incorporate no prior information. The observed data  $\mathbf{x}$  is, however, considered random and we can often then obtain the PDF for  $\mathbf{x}$  when the value of  $\boldsymbol{\theta}$  is known. This PDF is termed the *likelihood*  $L(\mathbf{x}; \boldsymbol{\theta})$ , which is defined as

$$L(\mathbf{x}; \boldsymbol{\theta}) = p(\mathbf{x}|\boldsymbol{\theta}). \quad (4.5)$$

The likelihood is of course implicitly conditioned upon all of our modeling assumptions  $\mathcal{M}$ , which could more properly be expressed as  $p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{M})$ .

The ML estimate for  $\boldsymbol{\theta}$  is then that value of  $\boldsymbol{\theta}$  which maximizes the likelihood for given observations  $\mathbf{x}$ :

$$\boldsymbol{\theta}^{\text{ML}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\{p(\mathbf{x}|\boldsymbol{\theta})\}, \quad (4.6)$$

Maximum likelihood (ML) estimator.

The rationale behind this is that the ML solution corresponds to the parameter vector which would have generated the observed data  $\mathbf{x}$  with highest probability. The maximization task required for ML estimation can be achieved using standard differential calculus for well-behaved and differentiable likelihood functions, and it is often convenient analytically to maximize the log-likelihood function  $l(\mathbf{x}; \boldsymbol{\theta}) = \log(L(\mathbf{x}; \boldsymbol{\theta}))$  rather than  $L(\mathbf{x}; \boldsymbol{\theta})$  itself. Since log is a monotonically increasing function the two solutions are identical.

In data analysis and signal processing applications the likelihood function is arrived at through knowledge of the stochastic model for the data. For example, in the case of the linear Gaussian model (4.1) the likelihood can be obtained easily if we know the form of  $p(\mathbf{e})$ , the joint PDF for the components of the error vector. The likelihood  $p(\mathbf{x}|\boldsymbol{\theta})$  is then found from a transformation of variables  $\mathbf{e} \rightarrow \mathbf{x}$  where  $\mathbf{e} = \mathbf{x} - \mathbf{G}\boldsymbol{\theta}$ . The Jacobian for this transformation is unity, so the likelihood is:

$$L(\mathbf{x}; \boldsymbol{\theta}) = p(\mathbf{x}|\boldsymbol{\theta}) = p_{\mathbf{e}}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta}). \quad (4.7)$$

The ML solution may of course be posed for any error distribution, including highly non-Gaussian and non-independent cases. Here we give the most straightforward case, for the linear model where the elements of the error vector  $\mathbf{e}$  are assumed to be i.i.d. and Gaussian and zero mean. If the variance of the Gaussian is  $\sigma_e^2$  we then have:

$$p_{\mathbf{e}}(\mathbf{e}) = \frac{1}{(2\pi\sigma_e^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_e^2} \mathbf{e}^T \mathbf{e}\right).$$

The likelihood is then

$$L(\mathbf{x}; \boldsymbol{\theta}) = p(\mathbf{x}|\boldsymbol{\theta}) = p_{\mathbf{e}}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta}) = \frac{1}{(2\pi\sigma_e^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_e^2}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})\right), \quad (4.8)$$

which leads to the following log-likelihood expression:

$$\begin{aligned} l(\mathbf{x}; \boldsymbol{\theta}) &= -(N/2) \log(2\pi\sigma_e^2) - \frac{1}{2\sigma_e^2}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T(\mathbf{x} - \mathbf{G}\boldsymbol{\theta}) \\ &= -(N/2) \log(2\pi\sigma_e^2) - \frac{1}{2\sigma_e^2} \sum_{n=1}^N (x_n - \mathbf{g}_n^T \boldsymbol{\theta})^2. \end{aligned}$$

Maximization of this function w.r.t.  $\boldsymbol{\theta}$  is equivalent to minimizing the sum-squared of the error sequence  $E = \sum_{n=1}^N (x_n - \mathbf{g}_n^T \boldsymbol{\theta})^2$ . This is exactly the criterion which is applied in the familiar *least squares* (LS) estimation method. The ML estimator is obtained by taking derivatives w.r.t.  $\boldsymbol{\theta}$  and equating to zero:

$$\boldsymbol{\theta}^{\text{ML}} = (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T \mathbf{x}, \quad (4.9)$$

Maximum likelihood for the linear Gaussian model,

which is, as expected, the familiar linear least squares estimate for model parameters calculated from finite length data observations. Thus we see that the ML estimator under the i.i.d. Gaussian error assumption is exactly equivalent to the well known least squares (LS) parameter estimator.

### 3.04.2.3 Bayesian inference

The ML methods treat parameters as unknown constants. If we are prepared to treat parameters as random variables it is possible to assign prior PDFs to the parameters. These PDFs should ideally express some prior knowledge about the relative probability of different parameter values *before the data are observed*. Of course if nothing is known *a priori* about the parameters then the prior distributions should in some sense express no initial preference for one set of parameters over any other. Note that in many cases a prior density is chosen to express some highly qualitative prior knowledge about the parameters. In such cases the prior chosen will be more a reflection of a *degree of belief* concerning parameter values than any true modeling of an underlying random process which might have generated those parameters. This willingness to assign priors which reflect subjective information is a powerful feature and also one of the most fundamental differences between the Bayesian and “classical” inferential procedures. For various expositions of the Bayesian methodology and philosophy see, for example [3–6]. The precise form of probability distributions assigned *a priori* to the parameters requires careful consideration since misleading results can be obtained from erroneous priors, but in principle at least we can apply the Bayesian approach to any problem where statistical uncertainty is present.

Bayes’ theorem is now stated as applied to estimation of random parameters  $\boldsymbol{\theta}$  from a random vector  $\mathbf{x}$  of observations, known as the posterior or *a posteriori* probability for the parameter:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}, \quad (4.10)$$

Posterior probability.

Note that all of the distributions in this expression are implicitly conditioned upon all prior modeling assumptions, as was the likelihood function earlier. The distribution  $p(\mathbf{x}|\boldsymbol{\theta})$  is the likelihood as used for ML estimation, while  $p(\boldsymbol{\theta})$  is the prior or *a priori* distribution for the parameters. This term is one of the critical differences between Bayesian and “classical” techniques. It expresses in an objective fashion the probability of various model parameters values *before* the data  $\mathbf{x}$  has been observed. As we have already observed, the prior density may be an expression of highly subjective information about parameter values. This transformation from the subjective domain to an objective form for the prior can clearly be of great significance and should be considered carefully when setting up an inference problem.

The term  $p(\boldsymbol{\theta}|\mathbf{x})$ , the posterior or *a posteriori* distribution, expresses the probability of  $\boldsymbol{\theta}$  given the observed data  $\mathbf{x}$ . This is now a true measure of how “probable” a particular value of  $\boldsymbol{\theta}$  is, given the observations  $\mathbf{x}$ .  $p(\boldsymbol{\theta}|\mathbf{x})$  is in a more intuitive form for parameter estimation than the likelihood, which expresses how probable the *observations* are given the *parameters*. The generation of the posterior distribution from the prior distribution when data  $\mathbf{x}$  is observed can be thought of as a refinement to any previous (“prior”) knowledge about the parameters. Before  $\mathbf{x}$  is observed  $p(\boldsymbol{\theta})$  expresses any information previously obtained concerning  $\boldsymbol{\theta}$ . Any new information concerning the parameters contained in  $\mathbf{x}$  is then incorporated to give the posterior distribution. Clearly if we start off with little or no information about  $\boldsymbol{\theta}$  then the posterior distribution is likely to obtain information obtained almost solely from  $\mathbf{x}$ . Conversely, if  $p(\boldsymbol{\theta})$  expresses a significant amount of information about  $\boldsymbol{\theta}$  then  $\mathbf{x}$  will contribute less new information to the posterior distribution.

The denominator  $p(\mathbf{x})$ , referred to as the marginal likelihood, or the “evidence” in machine learning circles, is a fundamentally useful quantity in model selection problems (see later), and is constant for any given observation  $\mathbf{x}$ ; thus it may be ignored if we are only interested in the relative posterior probabilities of different parameters. As a result of this, Bayes’ theorem is often stated in the form:

$$p(\boldsymbol{\theta}|\mathbf{x}) \propto p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}), \quad (4.11)$$

Posterior probability (proportionality).

$p(\mathbf{x})$  may be calculated in principle by integration:

$$p(\mathbf{x}) = \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta} \quad (4.12)$$

and this effectively serves as the normalizing constant for the posterior density (in this and subsequent results the integration would be replaced by a summation in the case of a discrete random vector  $\boldsymbol{\theta}$ ).

It is worth reiterating at this stage that we are here implicitly conditioning in this framework on many pieces of additional prior information beyond just the prior parameters of the model  $\boldsymbol{\theta}$ . For example, we are assuming a precise form for the data generation process in the model; if the linear Gaussian model is assumed, then the whole data generation process *must* follow the probability law of that model, otherwise we cannot guarantee the quality of our answers; the same argument applies to ML estimation, although in the Bayesian setting the distributional form of the Bayesian prior must be assumed in addition. Some texts therefore adopt a notation  $p(\boldsymbol{\theta}|\mathbf{x}, \mathcal{M})$  for Bayesian posterior distributions, where  $\mathcal{M}$  denotes all of the additional modeling and distributional assumptions that are being made. We will omit this convention for the sake of notational simplicity. However, it will be partially reintroduced

when we augment a model with other terms such as unknown hyperparameters (such as  $\sigma_e$  in the linear Gaussian model) or a model index  $\mathcal{M}_i$ , when we are considering several competing models (and associated prior distributions) for explaining the data. In these cases, for example, the notation will be naturally extended as required, e.g.,  $p(\boldsymbol{\theta}|\mathbf{x}, \sigma_e)$  or  $p(\boldsymbol{\theta}|\mathbf{x}, \mathcal{M}_i)$  in the two cases just described.

### 3.04.2.3.1 Posterior inference and Bayesian cost functions

The posterior distribution gives the probability for any chosen  $\boldsymbol{\theta}$  given observed data  $\mathbf{x}$ , and as such optimally combines our prior information about  $\boldsymbol{\theta}$  and any additional information gained about  $\boldsymbol{\theta}$  from observing  $\mathbf{x}$ . We may in principle manipulate the posterior density to infer any required statistic of  $\boldsymbol{\theta}$  conditional upon  $\mathbf{x}$ . This is a significant advantage over ML and least squares methods which strictly give us only a single estimate of  $\boldsymbol{\theta}$ , known as a “point estimate.” However, by producing a posterior PDF with values defined for all  $\boldsymbol{\theta}$  the Bayesian approach gives a fully interpretable probability distribution. In principle this is as much as one could ever need to know about the inference problem. In signal processing problems, however, we usually require a single point estimate for  $\boldsymbol{\theta}$ , and a suitable way to choose this is via a “cost function”  $C(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta})$  which expresses objectively a measure of the cost associated with a particular parameter estimate  $\hat{\boldsymbol{\theta}}$  when the true parameter is  $\boldsymbol{\theta}$  (see e.g., [3, 6, 7]). The form of cost function will depend on the requirements of a particular problem. A cost of 0 indicates that the estimate is perfect for our requirements (this does not necessarily imply that  $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$ , though it usually will) while positive values indicate poorer estimates. The *risk* associated with a particular estimator is then defined as the expected posterior cost associated with that estimate:

$$R(\hat{\boldsymbol{\theta}}) = E[C(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta})] = \int_{\mathbf{x}} \int_{\boldsymbol{\theta}} C(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{x}) p(\mathbf{x}) d\boldsymbol{\theta} d\mathbf{x}. \quad (4.13)$$

We require the estimation scheme which chooses  $\hat{\boldsymbol{\theta}}$  in order to minimize the risk. The minimum risk is known as the “Bayes risk.” For non-negative cost functions it is sufficient to minimize only the inner integral

$$I(\hat{\boldsymbol{\theta}}) = \int_{\boldsymbol{\theta}} C(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{x}) d\boldsymbol{\theta} \quad (4.14)$$

for all  $\hat{\boldsymbol{\theta}}$ . Typical cost functions are the quadratic cost function  $|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}|^2$  and the uniform cost function, defined for arbitrarily small  $\varepsilon$  as

$$C(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) = \begin{cases} 1, & |\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}| > \varepsilon, \\ 0, & \text{otherwise.} \end{cases} \quad (4.15)$$

The quadratic cost function leads to the minimum mean-squared error (MMSE) estimator and as such is reasonable for many examples of parameter estimation, where we require an estimate representative of the whole posterior density. To see the form of the resulting estimator, consider differentiation

under the integral:

$$\begin{aligned}\frac{dI(\hat{\theta})}{d\hat{\theta}} &= \frac{d}{d\hat{\theta}} \int_{\theta} C(\hat{\theta}, \theta) p(\theta|x)d\theta \\ &= \int_{\theta} \frac{d}{d\hat{\theta}} |\hat{\theta} - \theta|^2 p(\theta|x)d\theta \\ &= \int_{\theta} 2(\hat{\theta} - \theta) p(\theta|x)d\theta \\ &= 2\hat{\theta} - 2 \int_{\theta} \theta p(\theta|x)d\theta.\end{aligned}$$

And hence at the optimum estimator we have that the MMSE estimate equals the *mean* of the posterior distribution:

$$\hat{\theta} = \int_{\theta} \theta p(\theta|x)d\theta. \quad (4.16)$$

Where the posterior distribution is symmetrical about its mean the posterior mean can in fact be shown to be the minimum risk solution for any cost function which is a convex function of  $|\hat{\theta} - \theta|$  [8]. Thus the posterior mean estimator is very often used as the “standard” Bayesian estimate of a parameter. It always needs to be remembered however that there is a whole posterior distribution of parameters available here and that at the very least posterior uncertainty should be summarized through the covariance of the posterior distribution, or posterior standard deviations for the individual elements of the parameter vector. It should be noted as well that in cases where the posterior distribution is multi-modal, that is there is more than one maximum to  $p(\theta|x)$ , the posterior mean can give very misleading results.

The uniform cost function on the other hand is useful for the “all or nothing” scenario where we wish to attain the correct parameter estimate at all costs and any other estimate is of no use. Therrien [2] cites the example of a pilot landing a plane on an aircraft carrier. If he does not estimate within some small finite error he misses the ship, in which case the landing is a disaster. The uniform cost function for  $\varepsilon \rightarrow 0$  leads to the maximum *a posteriori* (MAP) estimate, the value of  $\hat{\theta}$  which maximizes the posterior distribution:

$$\theta^{\text{MAP}} = \underset{\theta}{\operatorname{argmax}} \{p(\theta|x)\}, \quad (4.17)$$

Maximum a posteriori (MAP) estimator.

Note that for Gaussian posterior distributions the MMSE and MAP solutions coincide, as indeed they do for any distribution symmetric about its mean with its maximum at the mean.

We now work through the MAP estimation scheme under the linear Gaussian model (4.1). Suppose that the prior on parameter vector  $\theta$  is the multivariate Gaussian (4.71):

$$p(\theta) = N(\mathbf{m}_{\theta}, \mathbf{C}_{\theta}) = \frac{1}{(2\pi)^{P/2} |\mathbf{C}_{\theta}|^{1/2}} \exp\left(-\frac{1}{2} (\theta - \mathbf{m}_{\theta})^T \mathbf{C}_{\theta}^{-1} (\theta - \mathbf{m}_{\theta})\right), \quad (4.18)$$

where  $\mathbf{m}_{\theta}$  is the prior parameter mean vector,  $\mathbf{C}_{\theta}$  is the parameter covariance matrix and  $P$  is the number of parameters in  $\theta$ . If the likelihood  $p(\mathbf{x}|\theta)$  takes the same form as before (4.8), the posterior distribution

is as follows:

$$\begin{aligned} p(\boldsymbol{\theta}|\mathbf{x}) \propto & \frac{1}{(2\pi)^{P/2} |\mathbf{C}_{\boldsymbol{\theta}}|^{1/2}} \frac{1}{(2\pi\sigma_e^2)^{N/2}} \\ & \times \exp \left( -\frac{1}{2\sigma_e^2} (\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T (\mathbf{x} - \mathbf{G}\boldsymbol{\theta}) \right. \\ & \left. - \frac{1}{2} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}})^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}}) \right) \end{aligned} \quad (4.19)$$

and the MAP estimate  $\boldsymbol{\theta}^{\text{MAP}}$  is obtained by differentiation of the log-posterior and finding its unique maximizer as:

$$\boldsymbol{\theta}^{\text{MAP}} = \left( \mathbf{G}^T \mathbf{G} + \sigma_e^2 \mathbf{C}_{\boldsymbol{\theta}}^{-1} \right)^{-1} \left( \mathbf{G}^T \mathbf{x} + \sigma_e^2 \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{m}_{\boldsymbol{\theta}} \right), \quad (4.20)$$

MAP estimator—linear Gaussian model.

In this expression we can clearly see the “regularizing” effect of the prior density on the ML estimate of (4.9). As the prior becomes more “diffuse,” i.e., the diagonal elements of  $\mathbf{C}_{\boldsymbol{\theta}}$  increase both in magnitude and relative to the off-diagonal elements, we impose “less” prior information on the estimate. In the limit the prior tends to a uniform (“flat”) prior with all  $\boldsymbol{\theta}$  equally probable. In this limit  $\mathbf{C}_{\boldsymbol{\theta}}^{-1} = 0$  and the estimate is identical to the ML estimate (4.9). This useful relationship demonstrates that the ML estimate may be interpreted as the MAP estimate with a uniform prior assigned to  $\boldsymbol{\theta}$ . The MAP estimate will also tend towards the ML estimate when the likelihood is strongly “peaked” around its maximum compared with the prior. Once again the prior will then have little influence on the shape of the posterior density. It is in fact well known [5] that as the sample size  $N$  tends to infinity the Bayes solution tends to the ML solution. This of course says nothing about small sample parameter estimates where the effect of the prior may be very significant.

The choice of a multivariate Gaussian prior may well be motivated by physical considerations about the problem, or it may be motivated by subjective prior knowledge about the value of  $\boldsymbol{\theta}$  (before the data  $\mathbf{x}$  are seen!) in terms of a rough value  $\mathbf{m}_{\boldsymbol{\theta}}$  and a confidence in that value through the covariance matrix  $\mathbf{C}_{\boldsymbol{\theta}}$  (a “subjective” prior). In fact the choice of Gaussian also has the very special property that it makes the Bayesian calculations straightforward and available in closed form. Such a prior is known as a “conjugate” prior [3].

### 3.04.2.3.2 Posterior distribution for parameters in the linear Gaussian model

Reconsidering the form of (4.19), we can obtain a lot more information than simply the MAP estimate of (4.20). A fully Bayesian analysis of the problem will study the whole posterior distribution for the unknowns, computing measures of uncertainty, confidence intervals, and so on. The required distribution can be obtained by rearranging the exponent of (4.19) using the result of (4.74),

$$\begin{aligned} & \frac{1}{\sigma_e^2} (\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T (\mathbf{x} - \mathbf{G}\boldsymbol{\theta}) + (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}})^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}}) \\ &= \frac{1}{\sigma_e^2} \left( \left( \boldsymbol{\theta} - \boldsymbol{\theta}^{\text{MAP}} \right)^T \Phi \left( \boldsymbol{\theta} - \boldsymbol{\theta}^{\text{MAP}} \right) + \mathbf{x}^T \mathbf{x} + \sigma_e^2 \mathbf{m}_{\boldsymbol{\theta}}^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{m}_{\boldsymbol{\theta}} - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}} \right) \end{aligned} \quad (4.21)$$

with terms defined as

$$\boldsymbol{\theta}^{\text{MAP}} = \boldsymbol{\Phi}^{-1} \boldsymbol{\Theta}, \quad (4.22)$$

$$\boldsymbol{\Phi} = \mathbf{G}^T \mathbf{G} + \sigma_e^2 \mathbf{C}_{\boldsymbol{\theta}}^{-1}, \quad (4.23)$$

$$\boldsymbol{\Theta} = \mathbf{G}^T \mathbf{x} + \sigma_e^2 \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{m}_{\boldsymbol{\theta}}. \quad (4.24)$$

Now we can observe that the first term in (4.21),  $\frac{1}{\sigma_e^2} ((\boldsymbol{\theta} - \boldsymbol{\theta}^{\text{MAP}})^T \boldsymbol{\Phi} (\boldsymbol{\theta} - \boldsymbol{\theta}^{\text{MAP}}))$ , is in exactly the correct form for the exponent of a multivariate Gaussian, see (4.71), with mean vector and covariance matrix as follows:

$$\mathbf{m}_{\boldsymbol{\theta}}^{\text{post}} = \boldsymbol{\theta}^{\text{MAP}} \quad \text{and} \quad \mathbf{C}_{\boldsymbol{\theta}}^{\text{post}} = \sigma_e^2 \boldsymbol{\Phi}^{-1}.$$

Since the remaining terms in (4.21) do not depend on  $\boldsymbol{\theta}$ , and we know that the multivariate density function must be proper (i.e., integrate to 1), we can conclude that the posterior distribution is itself a multivariate Gaussian,

$$p(\boldsymbol{\theta} | \mathbf{x}) = N \left( \boldsymbol{\theta}^{\text{MAP}}, \sigma_e^2 \boldsymbol{\Phi}^{-1} \right). \quad (4.25)$$

This result will be fundamental for the construction of inference methods in later sections, in cases where the model is at least partially linear/Gaussian, or can be approximated as linear and Gaussian.

### 3.04.2.3.3 Marginal likelihood

In problems where model choice or classification are of importance, and in inference algorithms which marginalize (“Rao-Blackwellize”) a Gaussian parameter (see later sections for details of these approaches) it will be required to compute the marginal likelihood for the data, i.e.,

$$p(\mathbf{x}) = \int_{\boldsymbol{\theta}} p(\mathbf{x} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}. \quad (4.26)$$

Now, filling in the required density functions for the linear Gaussian case, we have:

$$\begin{aligned} p(\mathbf{x}) &= \int_{\boldsymbol{\theta}} \frac{1}{(2\pi)^{P/2} |\mathbf{C}_{\boldsymbol{\theta}}|^{1/2}} \frac{1}{(2\pi\sigma_e^2)^{N/2}} \\ &\quad \times \exp \left( -\frac{1}{2\sigma_e^2} (\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T (\mathbf{x} - \mathbf{G}\boldsymbol{\theta}) \right. \\ &\quad \left. - \frac{1}{2} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}})^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}}) \right) d\boldsymbol{\theta}. \end{aligned} \quad (4.27)$$

This multivariate Gaussian integral can be performed after some rearrangement using result (4.76) to give:

$$\begin{aligned} p(\mathbf{x}) &= \frac{1}{(2\pi)^{P/2} |\mathbf{C}_{\boldsymbol{\theta}}|^{1/2} |\boldsymbol{\Phi}|^{1/2} (2\pi\sigma_e^2)^{(N-P)/2}} \\ &\quad \times \exp \left( -\frac{1}{2\sigma_e^2} (\mathbf{x}^T \mathbf{x} + \sigma_e^2 \mathbf{m}_{\boldsymbol{\theta}}^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{m}_{\boldsymbol{\theta}} - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}}) \right) \end{aligned} \quad (4.28)$$

with terms defined exactly as for the posterior distribution calculation above in (4.22)–(4.24).

### 3.04.2.3.4 Hyperparameters and marginalization of unwanted parameters

In many cases we can formulate a likelihood function for a particular problem which depends on more unknown parameters than are actually wanted for estimation. These will often be “scale” parameters such as unknown noise or excitation variances but may also be unobserved (“missing”) data values or unwanted system parameters. A simple example of such a parameter is  $\sigma_e$  in the linear Gaussian model above. In this case we can directly express the likelihood function exactly as before, but now explicitly conditioning on the unknown noise standard deviation:

$$\log p(\mathbf{x}|\boldsymbol{\theta}, \sigma_e) = -(N/2) \log (2\pi\sigma_e^2) - \frac{1}{2\sigma_e^2} \sum_{n=1}^N (x_n - \mathbf{g}_n^T \boldsymbol{\theta})^2.$$

A full ML procedure requires that the likelihood be maximized w.r.t. all of these parameters and the unwanted values are then simply discarded to give the required estimate, a “concentrated likelihood” estimate.<sup>2</sup>

However, this may not in general be an appropriate procedure for obtaining only the required parameters—a cost function which depends only upon a certain subset of parameters leads to an estimator which only depends upon the *marginal* probability for those parameters. The Bayesian approach allows for the interpretation of these unwanted or “nuisance” parameters as random variables, for which as usual we can specify prior densities. If the (possibly multivariate) hyperparameters are  $\phi$  ( $\phi$  would be taken to equal  $\sigma_e$  in the above simple example) then the full model can be specified through a joint prior on the parameters/hyperparameters and the joint likelihood:

$$\begin{aligned} p(\boldsymbol{\theta}, \phi, \mathbf{x}) &= p(\mathbf{x}|\boldsymbol{\theta}, \phi)p(\boldsymbol{\theta}, \phi) \\ &= p(\mathbf{x}|\boldsymbol{\theta}, \phi)p(\boldsymbol{\theta}|\phi)p(\phi), \end{aligned}$$

where the joint prior has been factored using probability chain rule in a way that is often convenient for specification and calculation.

The marginalisation identity can be used to eliminate these parameters from the posterior distribution, and from this we are able to obtain a posterior distribution in terms of only the desired parameters. Consider an unwanted parameter  $\phi$  which is present in the modeling assumptions. The unwanted parameter is now eliminated from the posterior expression by marginalisation:

$$\begin{aligned} p(\boldsymbol{\theta}|\mathbf{x}) &= \int_{\phi} p(\boldsymbol{\theta}, \phi|\mathbf{x})d\phi \\ &\propto \int_{\phi} p(\mathbf{x}|\boldsymbol{\theta}, \phi)p(\boldsymbol{\theta}, \phi)d\phi. \end{aligned} \tag{4.29}$$

### 3.04.2.3.5 Hyperparameters for the linear Gaussian model

As indicated above, it will often be important to estimate or marginalize the unknown hyperparameters of the linear Gaussian model. There is quite a lot that can be done in the special case of the linear

---

<sup>2</sup>Although in time series likelihood-based analysis it is common to treat missing and other unobserved data in a Bayesian fashion, see [9, 10] and references therein.

Gaussian model using analytic results. Studying the form of the likelihood function in (4.8), it can be observed that the form of the likelihood is quite similar to that of an inverted-gamma distribution, when considered as a function of  $\sigma_e^2$ , see Section A.4. This gives a hint as to a possible prior structure that might be workable in this model. If we take the prior itself for  $\sigma_e^2$  to be inverted-gamma (IG) then it is fairly straightforward to rearrange the conditional posterior distribution for  $\sigma_e^2$  into a tractable form. Since we are now treating  $\sigma_e$  as an unknown, conditioning of all terms on this unknown is now made explicit:

$$\begin{aligned} p(\sigma_e^2 | \boldsymbol{\theta}, \mathbf{x}) &= \frac{p(\mathbf{x} | \boldsymbol{\theta}, \sigma_e^2) p(\boldsymbol{\theta} | \sigma_e^2) p(\sigma_e^2)}{p(\boldsymbol{\theta}, \mathbf{x})} \\ &\propto p(\mathbf{x} | \boldsymbol{\theta}, \sigma_e^2) p(\boldsymbol{\theta} | \sigma_e^2) p(\sigma_e^2) \end{aligned} \quad (4.30)$$

and initially we take  $\boldsymbol{\theta}$  to be independent of  $\sigma_e^2$  so that  $p(\boldsymbol{\theta} | \sigma_e^2) = p(\boldsymbol{\theta})$  and

$$p(\sigma_e^2 | \boldsymbol{\theta}, \mathbf{x}) \propto p(\mathbf{x} | \boldsymbol{\theta}, \sigma_e^2) p(\sigma_e^2).$$

Taking the IG prior  $p(\sigma_e^2) = \text{IG}(\alpha_e, \beta_e)$ , the conditional probability becomes

$$\begin{aligned} p(\sigma_e^2 | \boldsymbol{\theta}, \mathbf{x}) &\propto \frac{1}{(2\pi\sigma_e^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_e^2}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})\right) \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \sigma_e^{-2(\alpha_e+1)} \exp\left(-\beta_e/\sigma_e^2\right) \\ &= \text{IG}\left(\alpha_e + N/2, \beta_e + \frac{1}{2}(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T(\mathbf{x} - \mathbf{G}\boldsymbol{\theta})\right), \end{aligned} \quad (4.31)$$

where the equality on the last line follows because we know that the posterior conditional distribution is a proper (normalized) density function, following similar reasoning to the Gaussian posterior distribution leading to (4.25). The IG prior is once again the conjugate prior for the scale parameter  $\sigma_e^2$  in the model: the posterior distribution is in the same form as the prior distribution, with parameters updated to include the effect of the data (through the likelihood function). This conditional posterior distribution can then be employed later for construction of efficient Gibbs sampling, expectation-maximization, variational Bayes and particle filtering solutions to models involving linear Gaussian components.

### 3.04.2.3.6 Normal-inverted-gamma prior

If a little more structure is included in the prior  $p(\boldsymbol{\theta}, \sigma_e^2)$  then still more can be done analytically. We retain the Gaussian and IG forms for the individual components, but introduce prior dependence between the parameters. Specifically, let

$$\mathbf{C}_{\boldsymbol{\theta}}^{-1} = \mathbf{M}_{\boldsymbol{\theta}} / \sigma_e^2,$$

where  $\mathbf{M}_{\boldsymbol{\theta}}$  is a positive definite matrix, considered fixed and known. Here then we are saying that the prior covariance of  $\boldsymbol{\theta}$  is known up to a scale factor of  $\sigma_e^2$ . The joint prior is then:

$$p(\boldsymbol{\theta}, \sigma_e^2) = p(\boldsymbol{\theta} | \sigma_e^2) p(\sigma_e^2) = N(\mathbf{m}_{\boldsymbol{\theta}}, \sigma_e^2 \mathbf{M}_{\boldsymbol{\theta}}^{-1}) \text{IG}(\sigma_e^2 | \alpha_e, \beta_e),$$

which is of normal-inverted-gamma form, see (4.83).

Inserting the joint prior into the expression for joint posterior distribution in a similar way to the simple Gaussian prior model for  $\boldsymbol{\theta}$  alone,

$$p(\boldsymbol{\theta}, \sigma_e^2 | \mathbf{x}) \propto p(\mathbf{x} | \boldsymbol{\theta}, \sigma_e^2) p(\boldsymbol{\theta} | \sigma_e^2) p(\sigma_e^2),$$

we obtain as an extension of (4.19),

$$\begin{aligned} p(\boldsymbol{\theta}, \sigma_e^2 | \mathbf{x}) &\propto \frac{1}{(2\pi)^{P/2} |\mathbf{C}_{\boldsymbol{\theta}}|^{1/2}} \frac{1}{(2\pi \sigma_e^2)^{N/2}} \\ &\times \exp\left(-\frac{1}{2\sigma_e^2} (\mathbf{x} - \mathbf{G}\boldsymbol{\theta})^T (\mathbf{x} - \mathbf{G}\boldsymbol{\theta})\right. \\ &\quad \left.- \frac{1}{2} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}})^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} (\boldsymbol{\theta} - \mathbf{m}_{\boldsymbol{\theta}})\right) \\ &\times \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \sigma_e^{-2(\alpha_e+1)} \exp\left(-\beta_e/\sigma_e^2\right). \end{aligned} \quad (4.32)$$

This joint distribution must factor, by the probability chain rule, into:

$$p(\boldsymbol{\theta}, \sigma_e^2 | \mathbf{x}) = p(\boldsymbol{\theta} | \sigma_e^2, \mathbf{x}) p(\sigma_e^2 | \mathbf{x}).$$

The second term in this can be obtained directly from previous results since

$$p(\sigma_e^2 | \mathbf{x}) \propto p(\mathbf{x} | \sigma_e^2) p(\sigma_e^2)$$

but we already have the first term  $p(\mathbf{x} | \sigma_e^2)$  in the marginal likelihood calculation of (4.28), now though making the dependence on  $\sigma_e^2$  explicit and substituting the form  $\mathbf{C}_{\boldsymbol{\theta}}^{-1} = \mathbf{M}_{\boldsymbol{\theta}} / \sigma_e^2$ , so

$$\begin{aligned} p(\sigma_e^2 | \mathbf{x}) &\propto \frac{|\mathbf{M}_{\boldsymbol{\theta}}|^{1/2}}{(2\pi)^{P/2} |\Phi|^{1/2} (2\pi \sigma_e^2)^{(N)/2}} \\ &\times \exp\left(-\frac{1}{2\sigma_e^2} \left(\mathbf{x}^T \mathbf{x} + \mathbf{m}_{\boldsymbol{\theta}}^T \mathbf{M}_{\boldsymbol{\theta}} \mathbf{m}_{\boldsymbol{\theta}} - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}}\right)\right) \\ &\times \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \sigma_e^{-2(\alpha_e+1)} \exp\left(-\beta_e/\sigma_e^2\right) \\ &= \text{IG}\left(\alpha_e + N/2, \beta_e + \frac{(\mathbf{x}^T \mathbf{x} + \mathbf{m}_{\boldsymbol{\theta}}^T \mathbf{M}_{\boldsymbol{\theta}} \mathbf{m}_{\boldsymbol{\theta}} - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}})}{2}\right) \end{aligned}$$

with appropriately simplified terms taken from (4.22)–(4.24) and substituting the form  $\mathbf{C}_{\boldsymbol{\theta}}^{-1} = \mathbf{M}_{\boldsymbol{\theta}} / \sigma_e^2$ :

$$\begin{aligned} \boldsymbol{\theta}^{\text{MAP}} &= \boldsymbol{\Phi}^{-1} \boldsymbol{\Theta}, \\ \boldsymbol{\Phi} &= \mathbf{G}^T \mathbf{G} + \mathbf{M}_{\boldsymbol{\theta}}, \\ \boldsymbol{\Theta} &= \mathbf{G}^T \mathbf{x} + \mathbf{M}_{\boldsymbol{\theta}} \mathbf{m}_{\boldsymbol{\theta}}. \end{aligned}$$

It can then be seen by inspection that all of the  $\boldsymbol{\theta}$  terms in may be grouped together in (4.32) as for the Gaussian model to give the remaining required conditional,

$$p(\boldsymbol{\theta} | \mathbf{x}, \sigma_e^2) = N(\boldsymbol{\theta}^{\text{MAP}}, \sigma_e^2 \boldsymbol{\Phi}^{-1}). \quad (4.33)$$

Multiplying these two conditionals back together we obtain:

$$\begin{aligned} p(\boldsymbol{\theta}, \sigma_e^2 | \mathbf{x}) &= p(\boldsymbol{\theta} | \sigma_e^2, \mathbf{x}) p(\sigma_e^2 | \mathbf{x}) \\ &= N(\boldsymbol{\theta} | \boldsymbol{\theta}^{\text{MAP}}, \sigma_e^2 \boldsymbol{\Phi}^{-1}) \text{IG}\left(\sigma_e^2 \mid \alpha_e + N/2, \beta_e + \frac{(\mathbf{x}^T \mathbf{x} + \mathbf{m}_\theta^T \mathbf{M}_\theta \mathbf{m}_\theta - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}})}{2}\right), \end{aligned}$$

which is in normal-inverted-gamma form—once again, the normal-inverted-gamma prior is *conjugate* to the linear Gaussian model with unknown linear parameters and noise variance.

The marginal likelihood can also be computed for this model, which will be required for Rao-Blackwellized inference schemes and for model choice problems,

$$p(\mathbf{x}) = \int_{\boldsymbol{\theta}} \int_{\sigma_e^2} p(\mathbf{x}, \boldsymbol{\theta}, \sigma_e^2) d\boldsymbol{\theta} d\sigma_e^2 = \int_{\sigma_e^2} p(\mathbf{x} | \sigma_e^2) p(\sigma_e^2) d\sigma_e^2, \quad (4.34)$$

where the first term in this integrand has already been obtained from (4.28), once again with appropriately simplified terms taken from (4.22)–(4.24) and substituting the form  $\mathbf{C}_{\boldsymbol{\theta}}^{-1} = \mathbf{M}_\theta / \sigma_e^2$ :

$$\begin{aligned} p(\mathbf{x} | \sigma_e^2) &= \frac{|\mathbf{M}_\theta|^{1/2}}{|\boldsymbol{\Phi}|^{1/2} (2\pi\sigma_e^2)^{N/2}} \\ &\times \exp\left(-\frac{1}{2\sigma_e^2} (\mathbf{x}^T \mathbf{x} + \mathbf{m}_\theta^T \mathbf{M}_\theta \mathbf{m}_\theta - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}})\right). \end{aligned}$$

Now, substituting this back into (4.34), the integral can be completed using the integral result from (4.78),

$$\begin{aligned} p(\mathbf{x}) &= \int_{\sigma_e^2} p(\mathbf{x} | \sigma_e^2) p(\sigma_e^2) d\sigma_e^2 \\ &= \int \frac{|\mathbf{M}_\theta|^{1/2}}{|\boldsymbol{\Phi}|^{1/2} (2\pi\sigma_e^2)^{(N)/2}} \\ &\times \exp\left(-\frac{1}{2\sigma_e^2} (\mathbf{x}^T \mathbf{x} + \mathbf{m}_\theta^T \mathbf{M}_\theta \mathbf{m}_\theta - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}})\right) \\ &\times \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \sigma_e^{-2(\alpha_e+1)} \exp(-\beta_e/\sigma_e^2) d\sigma_e^2 \\ &= \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \frac{|\mathbf{M}_\theta|^{1/2} \Gamma(\alpha_e + N/2)}{|\boldsymbol{\Phi}|^{1/2} (2\pi)^{N/2}} \left(\beta_e + \frac{1}{2} (\mathbf{x}^T \mathbf{x} + \mathbf{m}_\theta^T \mathbf{M}_\theta \mathbf{m}_\theta - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}})\right)^{-(\alpha_e+N/2)}. \end{aligned}$$

Despite the advantage of its analytic structure, an obvious challenge of this prior structure is how to specify the matrix  $\mathbf{M}_\theta$ . It can be chosen as some constant  $\kappa$  times the identity matrix in the absence

of further information, in which case  $\kappa$  can be interpreted as a “parameter-to-noise-ratio” term in the model. This could still be hard to specify in practice without unduly biasing results. One commonly used version of the general structure is the G-prior, which has an interpretation as a signal-to-noise ratio specification.

### 3.04.2.3.7 G-prior

A commonly used form of the above normal-inverted-gamma prior structure is the so-called G-prior, proposed by Zellner [11]. In this, the matrix  $\mathbf{M}_\theta$  is set as follows:

$$\mathbf{M}_\theta = g \mathbf{G}^T \mathbf{G},$$

where now  $g$  is a fixed tuning parameter of the model and has the intuitively appealing interpretation as a prior “signal-to-noise ratio” between the “signal” term  $\mathbf{G}\theta$  and the “noise” term  $\mathbf{e}$ .

The algebra for this prior follows through exactly as for the general case, but with further simplifications. For example, the marginal likelihood becomes

$$p(\mathbf{x}) = \frac{\beta_e^{\alpha_e}}{\Gamma(\alpha_e)} \frac{g^{P/2} \Gamma(\alpha_e + N/2)}{(1+g)^{P/2} (2\pi)^{N/2}} \left( \beta_e + \frac{1}{2} \left( \mathbf{x}^T \mathbf{x} + \mathbf{m}_\theta^T \mathbf{M}_\theta \mathbf{m}_\theta - \boldsymbol{\Theta}^T \boldsymbol{\theta}^{\text{MAP}} \right) \right)^{-(\alpha_e + N/2)}$$

with

$$\begin{aligned} \boldsymbol{\theta}^{\text{MAP}} &= \boldsymbol{\Phi}^{-1} \boldsymbol{\Theta}, \\ \boldsymbol{\Phi} &= \mathbf{G}^T \mathbf{G} (1+g), \\ \boldsymbol{\Theta} &= \mathbf{G}^T \mathbf{x} + g \mathbf{G}^T \mathbf{G} \mathbf{m}_\theta. \end{aligned}$$

The G-prior has been found to have useful properties for use in model choice problems.

### 3.04.2.3.8 Priors on covariance matrices

So far in the Bayesian linear Gaussian model no attempt has been made to model correlated structures, either in the noise  $\mathbf{x}$  or the parameters  $\theta$ , other than through the specification of the covariance matrix of the parameters,  $\mathbf{C}_\theta$ , which thus far has fixed value or well defined structure (the normal-inverse-gamma model). It is perfectly possible however to assign priors to full covariance matrices and integrate out the matrix as a hyperparameter, or derive its posterior distribution, as required of the particular application. In the linear Gaussian model this principle can be applied to the covariance matrix of the noise term (so far considered to be proportional to the identity matrix, corresponding to independent noise), and/or to the prior covariance matrix of the parameters  $\theta$ . Take this latter case as an example. We will wish to find an appropriate prior over positive definite matrices that has some tractable properties. In fact, remarkably, there is a conjugate prior for this case which gives full tractability in the linear Gaussian model. This is the inverse Wishart distribution, see Section A.7.

The prior for  $\mathbf{C}_\theta$  would then be:

$$p(\mathbf{C}_\theta) = \text{IW}(M_\theta, \alpha_\theta).$$

The full conditional distribution under the linear Gaussian model is obtained in a similar way to the IG prior applied to  $\sigma_e^2$  in (4.31):

$$\begin{aligned} p(\mathbf{C}_\theta | \boldsymbol{\theta}, \mathbf{x}, \sigma_e^2) &\propto p(\mathbf{x} | \boldsymbol{\theta}, \mathbf{C}_\theta, \sigma_e^2) p(\boldsymbol{\theta} | \mathbf{C}_\theta, \sigma_e^2) p(\mathbf{C}_\theta | \sigma_e^2) p(\sigma_e^2) \\ &\propto p(\boldsymbol{\theta} | \mathbf{C}_\theta) p(\mathbf{C}_\theta), \end{aligned} \quad (4.35)$$

where the simplification in the final line is owing to the fact that  $p(\mathbf{x} | \boldsymbol{\theta}, \mathbf{C}_\theta, \sigma_e^2) = p(\mathbf{x} | \boldsymbol{\theta}, \sigma_e^2)$  is the likelihood function, and does not depend upon  $\mathbf{C}_\theta$ , and the fact that the prior on  $\boldsymbol{\theta}$  does not depend on  $\sigma_e^2$ , i.e.,  $p(\mathbf{C}_\theta | \sigma_e^2) = p(\mathbf{C}_\theta)$ .

Now, substituting in the Gaussian and IW prior terms here from (4.18) and (4.85) we have

$$\begin{aligned} p(\mathbf{C}_\theta | \boldsymbol{\theta}, \mathbf{x}, \sigma_e^2) &\propto N(\boldsymbol{\theta} | \mathbf{m}_\theta, \mathbf{C}_\theta) \text{IW}(\mathbf{C}_\theta | \mathbf{M}_\theta, \alpha_\theta) \\ &\propto \frac{1}{(2\pi)^{P/2} |\mathbf{C}_\theta|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \mathbf{m}_\theta)^T \mathbf{C}_\theta^{-1} (\boldsymbol{\theta} - \mathbf{m}_\theta)\right) \\ &\quad \times \frac{\pi^{-P(P-1)/4}}{\prod_{i=1}^P \Gamma(0.5(2\alpha_\theta + 1 - i))} |\mathbf{M}_\theta|^{\alpha_\theta} |\mathbf{C}_\theta|^{-(\alpha_\theta + (P+1)/2)} \exp\left(-\text{tr}(\mathbf{M}_\theta \mathbf{C}_\theta^{-1})\right) \\ &= \text{IW}\left(\alpha_\theta + 1/2, \mathbf{M}_\theta + \frac{1}{2}(\boldsymbol{\theta} - \mathbf{m}_\theta)(\boldsymbol{\theta} - \mathbf{m}_\theta)^T\right) \end{aligned} \quad (4.36)$$

and we can see once again that the inverse Wishart prior is conjugate to the multivariate normal with unknown covariance matrix. Here the result  $\mathbf{a}^T \mathbf{b} = \text{tr}(\mathbf{b} \mathbf{a}^T)$  for column vectors  $\mathbf{a}$  and  $\mathbf{b}$  has been used to group together the exponent terms in the expression into a single trace expression.

### 3.04.2.4 Model uncertainty and Bayesian decision theory

In many problems of practical interest there are also issues of model choice and model uncertainty involved. For example, how can one choose the number of basis functions  $P$  in the linear model (4.1)? This could amount for example to a choice of how many sinusoids are necessary to model a given signal, or how many coefficients are required in an autoregressive model formulation. These questions should be answered automatically from the data and can as before include any known prior information about the models. There are a number of frequentist approaches to model choice, typically involving computation of the maximum likelihood parameter vector in each possible model order and then penalizing the likelihood function to prevent over-fitting by inappropriately high model orders. Such standard techniques include the AIC, MDL, and BIC procedures [12], which are typically based on asymptotic and information-theoretic considerations. In the fully Bayesian approach we aim to determine directly the posterior probability for each candidate model. An implicit assumption of the approach is that the true model which generated the data lies within the set of possible candidates; this is clearly an idealization for most real-world problems.

As for Bayesian parameter estimation, we consider the unobserved variable (in this case the model  $\mathcal{M}_i$ ) as being generated by some random process whose prior probabilities are known. These prior probabilities are assigned to each of the possible model states using a probability mass function (PMF)

$p(\mathcal{M}_i)$ , which expresses the prior probability of occurrence of different states given all information available except the data  $\mathbf{x}$ . The required form of Bayes' theorem for this discrete estimation problem is then

$$p(\mathcal{M}_i|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M}_i)p(\mathcal{M}_i)}{p(\mathbf{x})}. \quad (4.37)$$

$p(\mathbf{x})$  is constant for any given  $\mathbf{x}$  and will serve to normalize the posterior probabilities over all  $i$  in the same way that the marginal likelihood, or “evidence,” normalized the posterior parameter distribution (4.10). In the same way that Bayes rule gave a posterior distribution for parameters  $\theta$ , this expression gives the posterior probability for a particular model given the observed data  $\mathbf{x}$ . It would seem reasonable to choose the model  $\mathcal{M}_i$  corresponding to maximum posterior probability as our estimate for the true state (we will refer to this state estimate as the MAP estimate), and this can be shown to have the desirable property of minimum classification error rate  $P_E$  (see e.g., [7]), that is, it has minimum probability of choosing the wrong model. Note that determination of the MAP model estimate will usually involve an exhaustive search of  $p(\mathcal{M}_i|\mathbf{x})$  for all feasible  $i$ .

These ideas are formalized by consideration of a “loss function”  $\lambda(\alpha_i|\mathcal{M}_j)$  which defines the penalty incurred by taking action  $\alpha_i$  when the true state is  $\mathcal{M}_j$ . Action  $\alpha_i$  will usually refer to the action of choosing model  $\mathcal{M}_i$  as the estimate.

The expected risk associated with action  $\alpha_i$  (known as the conditional risk) is then expressed as

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{N_{\mathcal{M}}} \lambda(\alpha_i|\mathcal{M}_j)p(\mathcal{M}_j|\mathbf{x}). \quad (4.38)$$

where  $N_{\mathcal{M}}$  is the total number of models under consideration. It can be shown that it is sufficient to minimize this conditional risk in order to achieve the optimal decision rule for a given problem and loss function.

Consider a loss function which is zero when  $i = j$  and unity otherwise. This “symmetric” loss function can be viewed as the equivalent of the uniform cost function used for parameter estimation (4.15). The conditional risk is then given by:

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1, (j \neq i)}^{N_s} p(\mathcal{M}_j|\mathbf{x}) \quad (4.39)$$

$$= 1 - p(\mathcal{M}_i|\mathbf{x}). \quad (4.40)$$

The second line here is simply the conditional probability that action  $\alpha_i$  is *incorrect*, and hence minimization of the conditional risk is equivalent to minimization of the probability of classification error,  $P_E$ . It is clear from this expression that selection of the MAP state is the optimal decision rule for the symmetric loss function.

#### 3.04.2.4.1 Calculation of the marginal likelihood, $p(\mathbf{x}|\mathcal{M}_i)$

The term  $p(\mathbf{x}|\mathcal{M}_i)$  is equivalent to the marginal likelihood term  $p(\mathbf{x})$  which was encountered in the parameter estimation section, since  $p(\mathbf{x})$  was implicitly conditioned on a particular model structure or state in that scheme.

If one uses a uniform model prior  $p(\mathcal{M}_i) = \frac{1}{N_{\mathcal{M}}}$ , then, according to Eq. (4.37), it is only necessary to compare values of  $p(\mathbf{x}|\mathcal{M}_i)$  for model selection since the remaining terms are constant for all models.  $p(\mathbf{x}|\mathcal{M}_i)$  can then be viewed literally as the relative “evidence” for a particular model, and two candidate models can be compared through their Bayes Factor:

$$\text{BF}_{ij} = \frac{p(\mathbf{x}|\mathcal{M}_i)}{p(\mathbf{x}|\mathcal{M}_j)}.$$

Typically each model or state  $\mathcal{M}_i$  will be expressed in a parametric form whose parameters  $\boldsymbol{\theta}_i$  are unknown. As for the parameter estimation case it will usually be possible to obtain the state conditional parameter likelihood  $p(\mathbf{x}|\boldsymbol{\theta}_i, \mathcal{M}_i)$ . Given a model-dependent prior distribution for  $\boldsymbol{\theta}_i$  the marginal likelihood may be obtained by integration to eliminate  $\boldsymbol{\theta}_i$  from the joint probability  $p(\mathbf{x}, \boldsymbol{\theta}_i|\mathcal{M}_i)$ . The marginal likelihood is then obtained using result (4.29) as

$$p(\mathbf{x}|\mathcal{M}_i) = \int_{\boldsymbol{\theta}_i} p(\mathbf{x}|\boldsymbol{\theta}_i, \mathcal{M}_i) p(\boldsymbol{\theta}_i|\mathcal{M}_i) d\boldsymbol{\theta}_i. \quad (4.41)$$

If the linear Gaussian model of (4.1) is extended to the multi-model scenario we obtain:

$$\mathbf{x} = \mathbf{G}_i \boldsymbol{\theta}_i + \mathbf{e}_i, \quad (4.42)$$

where  $\mathbf{G}_i$  refers to the state-dependent basis matrix and  $\mathbf{e}_i$  is the corresponding error sequence. For this model the state dependent parameter likelihood  $p(\mathbf{x}|\boldsymbol{\theta}_i, \mathcal{M}_i)$  is (see (4.8)):

$$p(\mathbf{x}|\boldsymbol{\theta}_i, \mathcal{M}_i) = \frac{1}{(2\pi\sigma_{e_i}^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_{e_i}^2} (\mathbf{x} - \mathbf{G}_i \boldsymbol{\theta}_i)^T (\mathbf{x} - \mathbf{G}_i \boldsymbol{\theta}_i)\right). \quad (4.43)$$

If we for example take the same Gaussian form for the state conditional parameter prior  $p(\boldsymbol{\theta}_i|\mathcal{M}_i)$  (with  $P_i$  parameters) as we used for  $p(\boldsymbol{\theta})$  in (4.18) the marginal likelihood is then given with minor modification as:

$$\begin{aligned} p(\mathbf{x}|\mathcal{M}_i) &= \frac{1}{(2\pi)^{P_i/2} |\mathbf{C}_{\boldsymbol{\theta}_i}|^{1/2} |\boldsymbol{\Phi}|^{1/2} (2\pi\sigma_{e_i}^2)^{(N-P_i)/2}} \\ &\times \exp\left(-\frac{1}{2\sigma_{e_i}^2} \left( \mathbf{x}^T \mathbf{x} + \sigma_{e_i}^2 \mathbf{m}_{\boldsymbol{\theta}_i}^T \mathbf{C}_{\boldsymbol{\theta}_i}^{-1} \mathbf{m}_{\boldsymbol{\theta}_i} - \boldsymbol{\Theta}^T \boldsymbol{\theta}_i^{\text{MAP}} \right) \right) \end{aligned} \quad (4.44)$$

with terms defined as

$$\boldsymbol{\theta}_i^{\text{MAP}} = \boldsymbol{\Phi}^{-1} \boldsymbol{\Theta}, \quad (4.45)$$

$$\boldsymbol{\Phi} = \mathbf{G}_i^T \mathbf{G}_i + \sigma_{e_i}^2 \mathbf{C}_{\boldsymbol{\theta}_i}^{-1}, \quad (4.46)$$

$$\boldsymbol{\Theta} = \mathbf{G}_i^T \mathbf{x} + \sigma_{e_i}^2 \mathbf{C}_{\boldsymbol{\theta}_i}^{-1} \mathbf{m}_{\boldsymbol{\theta}_i}. \quad (4.47)$$

Notice that  $\boldsymbol{\theta}_i^{\text{MAP}}$  is simply the model-dependent version of the MAP parameter estimate given by (4.20). We can also of course look at the more sophisticated models using normal-inverted-gamma priors or G-priors, as before, and then the marginal likelihood expressions are again as given for the parameter estimation case.

### 3.04.2.5 Structures for model uncertainty

Model uncertainty may often be expressed in a highly structured way: in particular the models may be *nested* or *subset* structures. In the nested structure for the linear model, the basis matrix for model  $\mathcal{M}_{i+1}$  is obtained by adding an additional basis vector to the  $\mathbf{G}_i$ :

$$\mathbf{G}_{i+1} = [\mathbf{G}_i \mathbf{h}_{i+1}]$$

and hence the model of higher order inherits part of its structure from the lower order models. In subset, or “variable selection” models, a (potentially very large) pool of basis vectors  $\mathbf{h}_i$ ,  $i = 1, \dots, P_{\max}$  is available for construction of the model, and each candidate model is composed by taking a specific subset of the available vectors. A particular model  $\mathcal{M}_i$  is then specified by a set of  $P_i$  distinct integer indices  $\{i_1, i_2, \dots, i_{P_i}\}$  and the  $\mathbf{G}_i$  matrix is constructed as

$$\mathbf{G}_i = [\mathbf{h}_{i_1} \mathbf{h}_{i_2} \cdots \mathbf{h}_{i_{P_i}}].$$

In such a case there are  $2^{P_{\max}}$  possible subset models, which could be a huge number. Very often it will be infeasible to explore the posterior probabilities of all possible subsets, and hence sub-optimal search strategies are adopted, such as the MCMC algorithms of [13].

### 3.04.2.6 Bayesian model averaging

In many scenarios where there is model uncertainty, model choice is not the primary aim of the inference. Take for example the case where a state vector  $\mathbf{x}$  is common to all models, but each model has different parameters  $\theta_i$ , and the observed data are  $\mathbf{y}$ . Then the correct Bayesian procedure for inference about  $\mathbf{x}$  is to perform marginalisation over all possible models, or perform Bayesian model averaging (BMA):

$$p(\mathbf{x}|\mathbf{y}) = \sum_i p(\mathcal{M}_i|\mathbf{y}) p(\mathbf{x}|\mathbf{y}, \mathcal{M}_i).$$

Note though that  $p(\mathcal{M}_i|\mathbf{y})$  and  $p(\mathbf{x}|\mathbf{y}, \mathcal{M}_i)$  may not be analytically computable since they both require marginalizations (over  $\theta_i$  and/or  $\mathbf{x}$ ) and in these cases numerical strategies such as Markov chain Monte Carlo (MCMC) would be required.

---

## 3.04.3 Computational methods

In previous sections we have considered the basic frameworks for Bayesian inference, illustrated through the linear Gaussian model. This model makes many of the required calculations straightforward and analytically computable. However, for most real-world problems there will nearly always be intractable elements in the models, and for these numerical Bayesian methods are required. Take for example the sinusoidal model presented earlier. If we make the frequencies  $\{\omega_j\}$  unknown then the model is highly non-linear in these parameters. Similarly with the autoregressive model, if the signal is now observed in noise  $v_n$  as

$$y_n = x_n + v_n$$

then the posterior distribution for the parameters is no longer obtainable in closed form.

There is a wide range of computational tools now available for solving complex Bayesian inference problems, ranging from simple Laplace approximations to posterior densities, through variational Bayes methods to highly sophisticated Monte Carlo schemes. We will only attempt to give a flavor of some of the techniques out there, starting with one of the simplest and most effective: the EM algorithm.

### 3.04.3.1 Expectation-maximization (EM) for MAP estimation

The expectation-maximization (EM) algorithm [14] is an iterative procedure for finding modes of a posterior distribution or likelihood function, particularly in the context of “missing data.” EM has been used quite extensively in the signal processing literature for maximum likelihood parameter estimation, see e.g., [15–18]. The notation used here is essentially similar to that of Tanner [19, pp. 38–57].

The problem is formulated in terms of observed data  $\mathbf{y}$ , parameters  $\boldsymbol{\theta}$  and unobserved (“latent” or “missing”) data  $\mathbf{x}$ . A prototypical example of such a set-up is the AR model in noise:

$$\begin{aligned} x_n &= \sum_{i=1}^P a_i x_{n-i} + e_n, \\ y_n &= x_n + v_n \end{aligned} \tag{4.48}$$

in which the parameters to learn will be  $\boldsymbol{\theta} = [a_1, \dots, a_P]^T$ .

EM is useful in certain cases where it is straightforward to manipulate the conditional posterior distributions  $p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{y})$  and  $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$ , but perhaps not straightforward to deal with the marginal distributions  $p(\boldsymbol{\theta}|\mathbf{y})$  and  $p(\mathbf{x}|\mathbf{y})$ . The objective of EM in the Bayesian case is to obtain the MAP estimate for parameters:

$$\boldsymbol{\theta}^{\text{MAP}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \{p(\boldsymbol{\theta}|\mathbf{y})\} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \int_{\mathbf{x}} p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y}) d\mathbf{x} \right\}.$$

The basic Bayesian EM algorithm can be summarized as:

**1. Expectation step:**

Given the current estimate  $\boldsymbol{\theta}^i$ , calculate:

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^i) &= \int_{\mathbf{x}} \log(p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{x})) p(\mathbf{x}|\boldsymbol{\theta}^i, \mathbf{y}) d\mathbf{x} \\ &= E[\log(p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{x}))|\boldsymbol{\theta}^i, \mathbf{y}]. \end{aligned} \tag{4.49}$$

**2. Maximization step:**

$$\boldsymbol{\theta}^{i+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \{Q(\boldsymbol{\theta}, \boldsymbol{\theta}^i)\}. \tag{4.50}$$

These two steps are iterated until convergence is achieved. The algorithm is guaranteed to converge to a stationary point of  $p(\boldsymbol{\theta}|\mathbf{y})$ , although we must beware of convergence to local maxima when the posterior distribution is multimodal. The starting point  $\boldsymbol{\theta}^0$  determines which posterior mode is reached and can be critical in difficult applications.

EM-based methods can be thought of as a special case of variational Bayes methods [20,21]. In these, typically favoured by the machine learning community, a factored approximation to the joint posterior density is constructed, and each factor is iteratively updated using formulae somewhat similar to EM.

### 3.04.3.2 Markov chain Monte Carlo (MCMC)

At the more computationally expensive end of the spectrum we can consider Markov chain Monte Carlo (MCMC) simulation methods [22, 23]. The object of these methods is to draw *samples* from some target distribution  $\pi(\omega)$  which may be too complex for direct estimation procedures. The MCMC approach sets up an irreducible, aperiodic Markov chain whose stationary distribution is the target distribution of interest,  $\pi(\omega)$ . The Markov chain is then simulated from some arbitrary starting point and convergence in distribution to  $\pi(\omega)$  is then guaranteed under mild conditions as the number of state transitions (iterations) approaches infinity [24]. Once convergence is achieved, subsequent samples from the chain form a (dependent) set of samples from the target distribution, from which Monte Carlo estimates of desired quantities related to the distribution may be calculated.

For the statistical models considered in this chapter, the target distribution will be the *joint* posterior distribution for all unknowns,  $p(\mathbf{x}, \boldsymbol{\theta} | \mathbf{y})$ , from which samples of the unknowns  $\mathbf{x}$  and  $\boldsymbol{\theta}$  will be drawn conditional upon the observed data  $\mathbf{y}$ . Since the joint distribution can be factorised as  $p(\mathbf{x}, \boldsymbol{\theta} | \mathbf{y}) = p(\boldsymbol{\theta} | \mathbf{x}, \mathbf{y})p(\mathbf{x} | \mathbf{y})$  it is clear that the samples in  $\mathbf{x}$  which are extracted from the joint distribution are equivalent to samples from the *marginal* posterior  $p(\mathbf{x} | \mathbf{y})$ . The sampling method thus implicitly performs the (generally) analytically intractable marginalisation integral w.r.t.  $\boldsymbol{\theta}$ .

The Gibbs Sampler [25, 26] is perhaps the most simple and popular form of MCMC currently in use for the exploration of posterior distributions. This method, which can be derived as a special case of the more general Metropolis-Hastings method [22], requires the full specification of conditional posterior distributions for each unknown parameter or variable. Suppose that the reconstructed data and unknown parameters are split into (possibly multivariate) subsets  $\mathbf{x} = \{x_0, x_1, \dots, x_{N-1}\}$  and  $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_q\}$ . Arbitrary starting values  $\mathbf{x}^0$  and  $\boldsymbol{\theta}^0$  are assigned to the unknowns. A single iteration of the Gibbs Sampler then comprises sampling each variable from its conditional posterior with all remaining variables fixed to their current sampled value. The  $(i + 1)$ th iteration of the sampler may be summarized as:

$$\begin{aligned} \theta_1^{i+1} &\sim p\left(\theta_1 | \theta_2^i, \dots, \theta_q^i, x_0^i, x_1^i, \dots, x_{N-1}^i, \mathbf{y}\right), \\ \theta_2^{i+1} &\sim p\left(\theta_2 | \theta_1^{i+1}, \dots, \theta_q^i, x_0^i, x_1^i, \dots, x_{N-1}^i, \mathbf{y}\right), \\ &\vdots \\ \theta_q^{i+1} &\sim p\left(\theta_q | \theta_1^{i+1}, \theta_2^{i+1}, \dots, x_0^i, x_1^i, \dots, x_{N-1}^i, \mathbf{y}\right), \\ x_0^{i+1} &\sim p\left(x_0 | \theta_1^{i+1}, \theta_2^{i+1}, \dots, \theta_q^{i+1}, x_1^i, \dots, x_{N-1}^i, \mathbf{y}\right), \\ x_1^{i+1} &\sim p\left(x_1 | \theta_1^{i+1}, \theta_2^{i+1}, \dots, \theta_q^{i+1}, x_0^{i+1}, x_2^i, \dots, x_{N-1}^i, \mathbf{y}\right), \\ &\vdots \\ x_{N-1}^{i+1} &\sim p\left(x_{N-1} | \theta_1^{i+1}, \theta_2^{i+1}, \dots, \theta_q^{i+1}, x_0^{i+1}, x_1^{i+1}, \dots, x_{N-2}^{i+1}, \mathbf{y}\right), \end{aligned}$$

where the notation “ $\sim$ ” denotes that the variable to the left is drawn as a random independent sample from the distribution to the right.

The utility of the Gibbs Sampler arises as a result of the fact that the conditional distributions, under appropriate choice of parameter and data subsets ( $\theta_i$  and  $x_j$ ), will be more straightforward to sample than the full posterior. Multivariate parameter and data subsets can be expected to lead to faster convergence in terms of number of iterations (see e.g., [27, 28]), but there may be a trade-off in the extra computational complexity involved per iteration. Convergence properties are a difficult and important issue and concrete results are fairly rare. Numerous (but mostly *ad hoc*) convergence diagnostics have been devised for more general scenarios and a review may be found in [29], for example.

Once the sampler has converged to the desired posterior distribution, inference can easily be made from the resulting samples. One useful means of analysis is to form histograms or kernel density estimates of any parameters of interest. These converge in the limit to the true marginal posterior distribution for those parameters and can be used to estimate MAP values and Bayesian confidence intervals, for example. Alternatively a Monte Carlo estimate can be made for the expected value of any desired posterior functional  $f(\cdot)$  as a finite summation:

$$E[f(\mathbf{x})|\mathbf{y}] \approx \frac{\sum_{i=N_0+1}^{N_{\max}} f(\mathbf{x}^i)}{N_{\max} - N_0}, \quad (4.51)$$

where  $N_0$  is the convergence (“burn in”) time and  $N_{\max}$  is the total number of iterations. The MMSE estimator for example, is simply the posterior mean, estimated by setting  $f(\mathbf{x}) = \mathbf{x}$  in (4.51).

MCMC methods are computer-intensive and will only be applicable when off-line processing is acceptable and the problem is sufficiently complex to warrant their sophistication. However, they are currently unparalleled in ability to solve the most challenging of modeling problems. A more extensive survey can be found in the E-reference article by A.T. Cemgil. Further reading material includes MCMC for model uncertainty [30, 31].

### 3.04.4 State-space models and sequential inference

In this section we describe inference methods that run sequentially, or on-line, using the state-space formulation. These are vital in many applications, where either the data are too large to process in one batch, or results need to be produced in real-time as new data points arrive. In sequential inference we have seen some of the most exciting advances in methodology over the last decade.

#### 3.04.4.1 Linear Gaussian state-space models

Most linear time series models, including the AR models discussed earlier, can be expressed in the *state space* form:

$$y_n = Z\alpha_n + v_n \quad (\text{observation equation}), \quad (4.52)$$

$$\alpha_{n+1} = T\alpha_n + He_n \quad (\text{state update equation}). \quad (4.53)$$

In the top line, the *observation equation*, the observed data  $y_n$  is expressed in terms of an unobserved state  $\alpha_n$  and a noise term  $v_n$ .  $v_n$  is uncorrelated (i.e.,  $E[v_n v_m^T] = 0$  for  $n \neq m$ ) and zero mean, with covariance  $C_v$ . In the second line, the *state update equation*, the state  $\alpha_n$  is updated to its new value  $\alpha_n$

at time  $n + 1$  and a second noise term  $e_n$ .  $e_n$  is uncorrelated (i.e.,  $E[e_n e_m^T] = 0$  for  $n \neq m$ ) and zero mean, with covariance  $C_e$ , and is also uncorrelated with  $v_n$ . Note that in general the state  $\alpha_n$ , observation  $y_n$  and noise terms  $e_n/v_n$  can be column vectors and the constants  $Z$ ,  $T$ , and  $H$  are then matrices of the implied dimensionality. Also note that all of these constants can be made time index dependent without altering the form of the results given below.

Take, for example, an AR model  $\{x_t\}$  observed in noise  $\{v_n\}$ , so that the equations in standard form are:

$$\begin{aligned} y_n &= x_n + v_n, \\ x_n &= \sum_{i=1}^P a_i x_{n-i} + e_n. \end{aligned}$$

One way to express this in state-space form is as follows:

$$\begin{aligned} \alpha_n &= [x_n \ x_{n-1} \ x_{n-2} \ \dots \ x_{n-P+1}]^T, \\ T &= \begin{bmatrix} a_1 & a_2 & \dots & a_P \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 1 \end{bmatrix}, \\ H &= [1 \ 0 \ \dots \ 0]^T, \\ Z &= H^T, \\ C_e &= \sigma_e^2, \\ C_v &= \sigma_v^2. \end{aligned}$$

The state-space form is useful since some elegant results exist for the general form which can be applied in many different special cases. In particular, we will use it for sequential estimation of the state  $\alpha_n$ . In probabilistic terms this will involve updating the posterior probability for  $\alpha_n$  with the input of a new observation  $y_{n+1}$ :

$$p(\alpha_{n+1}|y_{1:n+1}) = g(p(\alpha_n|y_{1:n}), y_{n+1}),$$

where  $y_{1:n} = [y_1, \dots, y_n]^T$  and  $g(\cdot)$  denotes the sequential updating function.

Suppose that the noise sources  $e_n$  and  $v_n$  are Gaussian. Assume also that an initial state probability or prior  $p(\alpha_0)$  exists and is Gaussian  $N(m_0, C_0)$ . Then the posterior distributions are all Gaussian themselves and the posterior distribution for  $\alpha_n$  is fully represented by its *sufficient statistics*: its mean  $a_n = E[\alpha_n|y_0, \dots, y_n]$  and covariance matrix  $P_n = E[(\alpha_n - a_n)(\alpha_n - a_n)^T | y_0, \dots, y_n]$ . The *Kalman filter* [9,32,33] performs the update efficiently, as follows:

1. Initialise:  $a_0 = m_0$ ,  $P_0 = C_0$
2. Repeat for  $n = 1$  to  $N$ :

**a. Prediction:**

$$\begin{aligned} a_{n|n-1} &= T a_{n-1}, \\ P_{n|n-1} &= T P_{n-1} T^T + H C_e H^T. \end{aligned}$$

**b. Correction:**

$$\begin{aligned} a_n &= a_{n|n-1} + K_n (y_n - Z a_{n|n-1}), \\ P_n &= (I - K_n Z) P_{n|n-1}, \end{aligned}$$

where

$$K_n = P_{n|n-1} Z^T (Z P_{n|n-1} Z^T + C_v)^{-1} \quad (\text{Kalman Gain}).$$

Here  $a_{n|n-1}$  is the predictive mean  $E[\alpha_n | \mathbf{y}_{n-1}]$ ,  $P_{n|n-1}$  the predictive covariance  $E[(\alpha_n - a_{n|n-1})(\alpha_n - a_{n|n-1})^T | \mathbf{y}_{n-1}]$  and  $I$  denotes the (appropriately sized) identity matrix.

This is the probabilistic interpretation of the Kalman filter, assuming Gaussian distributions for noise sources and initial state, and it is this form that we will require in the next section on sequential Monte Carlo. In that section it will be more fully developed in the probabilistic version. The more standard interpretation is as the best linear estimator for the state [9, 32, 33].

### 3.04.4.2 The prediction error decomposition

One remarkable property of the Kalman filter is the *prediction error decomposition* which allows exact sequential evaluation of the *likelihood* function for the observations. If we suppose that the model depends upon some hyperparameters  $\theta$ , then the Kalman filter updates sequentially, for a particular value of  $\theta$ , the density  $p(\alpha_n | \mathbf{y}_{1:n}, \theta)$ . We define the likelihood for  $\theta$  in this context to be:

$$p(\mathbf{y}_{1:n} | \theta) = \int p(\alpha_n, \mathbf{y}_{1:n} | \theta) d\alpha_n$$

from which the ML or MAP estimator for  $\theta$  can be obtained by optimization. The prediction error decomposition [9, pp. 125–126] calculates this term from the outputs of the Kalman filter:

$$\log(p(\mathbf{y}_{1:n} | \theta)) = -\frac{Mn}{2} \log(2\pi) - \frac{1}{2} \sum_{t=1}^n \log |F_t| - \frac{1}{2} \sum_{t=1}^n w_t^T F_t^{-1} w_t, \quad (4.54)$$

where

$$\begin{aligned} F_t &= Z P_{t|t-1} Z^T + C_v, \\ w_t &= y_t - Z a_{t|t-1} \end{aligned}$$

and where  $M$  is the dimension of the observation vector  $y_n$ .

### 3.04.4.3 Sequential Monte Carlo (SMC)

It is now many years since the pioneering contribution of Gordon et al. [34], which is commonly regarded as the first instance of modern sequential Monte Carlo (SMC) approaches. Initially focussed on applications to tracking and vision, these techniques are now very widespread and have had a significant impact in virtually all areas of signal and image processing concerned with Bayesian dynamical models. This section serves as a brief introduction to the methods for the practitioner.

Consider the following generic nonlinear extension of the linear state-space model:

- *System model:*

$$x_t = a(x_{t-1}, u_t) \Leftrightarrow \underbrace{f(x_t | x_{t-1})}_{\text{Transition Density}} . \quad (4.55)$$

- *Measurement model:*

$$y_t = b(x_t, v_t) \Leftrightarrow \underbrace{g(y_t | x_t)}_{\text{Observation Density}} . \quad (4.56)$$

By these equations we mean that the hidden states  $x_t$  and data  $y_t$  are assumed to be generated by nonlinear functions  $a(\cdot)$  and  $b(\cdot)$ , respectively, of the state and noise disturbances  $u_t$  and  $v_t$ . The precise form of the functions and the assumed probability distributions of the state  $u_t$  and the observation  $v_t$  noises imply via a change of variables the transition probability density function  $f(x_t | x_{t-1})$  and the observation probability density function  $g(y_t | x_t)$ . It is assumed that  $x_t$  is Markovian, i.e., its conditional probability density given the past states  $x_{0:t-1} \stackrel{\text{def}}{=} (x_0, \dots, x_{t-1})$  depends only on  $x_{t-1}$  through the transition density  $f(x_t | x_{t-1})$ , and that the conditional probability density of  $y_t$  given the states  $x_{0:t}$  and the past observations  $y_{0:t-1}$  depends only upon  $x_t$  through the conditional likelihood  $g(y_t | x_t)$ . We further assume that the initial state  $x_0$  is distributed according to a density function  $p(x_0)$ . Such nonlinear dynamic systems arise frequently from many areas in science and engineering such as target tracking, computer vision, terrain referenced navigation, finance, pollution monitoring, communications, audio engineering, to list but a few.

A simple and standard example is now given to illustrate the model class.

**Example 1.** (Nonlinear time series model) This model has been used quite extensively in the filtering literature [34–36]. The required equations are as follows:

$$\begin{aligned} x_t &= \frac{x_{t-1}}{2} + 25 \frac{x_{t-1}}{1 + x_{t-1}^2} + 8 \cos(1.2t) + u_t, \\ y_t &= \frac{x_t^2}{20} + v_t, \end{aligned}$$

where  $u_t \sim N(0, \sigma_u^2)$  and  $v_t \sim N(0, \sigma_v^2)$  and here  $\sigma_u^2 = 10$  and  $\sigma_v^2 = 1$  are considered fixed and known;  $N(\mu, \sigma^2)$  once again denotes the normal distribution with mean  $\mu$  and variance  $\sigma^2$ . The representation

in terms of densities  $f(x_t|x_{t-1})$  and  $g(y_t|x_t)$  is given by:

$$f(x_t|x_{t-1}) = N\left(x_t \left| \frac{x_{t-1}}{2} + 25 \frac{x_{t-1}}{1+x_{t-1}^2} + 8 \cos(1.2t), \sigma_u^2\right.\right),$$

$$g(y_t|x_t) = N\left(y_t \left| \frac{x_t^2}{20}, \sigma_v^2\right.\right).$$

The form of these densities was straightforward to obtain in this case. For more complex cases a Jacobian term might be required when either  $x_t$  or  $y_t$  is a nonlinear function of  $u_t$  or  $v_t$ , respectively.

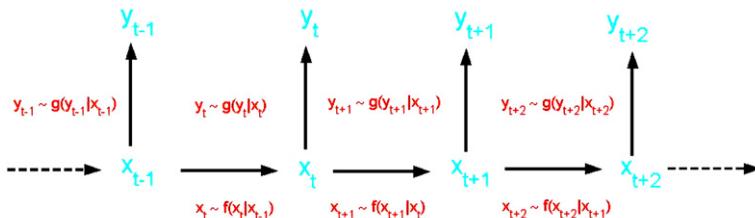
States may easily be simulated from a dynamical model of this sort owing to the Markovian assumptions on  $x_t$  and  $y_t$ , which imply that the joint probability density of states and observations, denoted  $p(x_{0:T}, y_{0:T})$ , may be factorized as

$$p(x_{0:T}, y_{0:T}) = f(x_0) \prod_{t=1}^T f(x_t|x_{t-1}) g(y_t|x_t).$$

A graphical representation of the dependencies between different states and observations is shown in Figure 4.1.

The ability to simulate random states and to evaluate the transition and observation densities (at least up to an unknown normalizing constant) will be principal components of the particle filtering algorithms described shortly.

Bayesian inference for the general nonlinear dynamic system above involves computing the *posterior distribution* of a collection of state variables  $x_{s:s'} \stackrel{\text{def}}{=} (x_s, \dots, x_{s'})$  conditioned on a batch of observations,  $y_{0:t} = (y_0, \dots, y_t)$ , which we denote  $p(x_{s:s'}|y_{0:t})$ . Specific problems include *filtering*, for  $s = s' = t$ , *fixed lag smoothing*, when  $s = s' = t - L$  and fixed interval smoothing, if  $s = 0$  and  $s' = t$ . Despite the apparent simplicity of the above problem, the posterior distribution can be computed in closed form only in very specific cases, principally, the linear Gaussian model (where the functions  $a()$  and  $b()$  are linear and  $u_t$  and  $v_t$  are Gaussian) and the discrete hidden Markov model (where  $x_t$  takes its values in a finite alphabet). In the vast majority of cases, nonlinearity or non-Gaussianity render an analytic solution intractable [32, 37–39].



**FIGURE 4.1**

Graphical model illustrating the Markovian dependencies between states and observations.

There are many classical (non-Monte Carlo-based) methods for nonlinear dynamic systems, including the extended Kalman filter (EKF) and its variants [40], Gaussian sum filters [41], unscented and quadrature Kalman filters [42–44]. However, limitations in the generality of these approaches to the most nonlinear/non-Gaussian systems has stimulated interest in more general techniques. Among these, Monte Carlo methods, in which the posterior distribution is represented by a collection of random points, are central. Early approaches include sequential importance sampling in the pioneering works of Handschin and Mayne [45] and Handschin [46]. The incorporation of resampling techniques was key though to the practical success of such approaches, as given in [34]. Without resampling, as the number of time points increases, the importance weights tend to degenerate, a phenomenon known as *sample impoverishment* or *weight degeneracy*. Since then, there have been several independent variants of similar filtering ideas, including the Condensation filter [47], Monte Carlo filter [35], Sequential imputations [48], and the Particle filter [49]. So far, sequential Monte Carlo (SMC) methods have been successfully applied in many different fields including computer vision, signal processing, tracking, control, econometrics, finance, robotics, and statistics; see [37, 39, 50, 51] and the references therein for a good review coverage.

#### 3.04.4.3.1 Predictor-corrector formulation

We now present the basic update for the state-space model. Starting with the initial, or “prior,” distribution  $p(x_0)$ , the posterior density  $p(x_{0:t}|y_{0:t})$  can be obtained using the following *prediction-correction* recursion [52]:

- *Prediction:*

$$p(x_{0:t}|y_{0:t-1}) = p(x_{0:t-1}|y_{0:t-1})f(x_t|x_{t-1}), \quad (4.57)$$

- *Correction:*

$$p(x_{0:t}|y_{0:t}) = \frac{g(y_t|x_t)p(x_{0:t}|y_{0:t-1})}{p(y_t|y_{0:t-1})}, \quad (4.58)$$

where  $p(y_t|y_{0:t-1})$  is a constant for any given data realization. It will not be necessary to compute this term in standard implementations of SMC methods.

#### 3.04.4.4 Particle filtering and auxiliary sampling

We now present a general form for of the particle filter, using the auxiliary filtering ideas of Pitt and Shephard [53], and which includes most common variants as special cases. We first represent the posterior smoothing density approximately as a weighted sum of  $\delta$  functions

$$p(x_{0:t}) \approx \sum_{i=1}^N \omega_t^{(i)} \delta_{x_{0:t}^{(i)}}(x_{0:t}), \quad (4.59)$$

where the notation  $\delta_{x_{0:t}^{(i)}}$  denotes the Dirac mass at point  $x_{0:t}^{(i)}$ . Under suitable technical assumptions, this is a consistent approximation, i.e., for any function  $h$  on the path space

$$\sum_{i=1}^N \omega_t^{(i)} h\left(x_{0:t}^{(i)}\right),$$

converges to  $E[h(x_{0:t})]$  as the number  $N$  of particles increases to infinity, see [37, 54–58] for technical considerations.

The formulation given here is equivalent to that given by Pitt and Shephard [53], although we avoid the explicit inclusion of an auxiliary indexing variable by considering a proposal over the entire path of the process up to time  $t$ . The starting assumption is that the joint posterior at  $t - 1$  is well approximated as

$$p(x_{0:t-1}|y_{0:t-1}) \approx \sum_{i=1}^N \omega_{t-1}^{(i)} \delta_{x_{0:t}^{(i)}}(x_{0:t}).$$

Based on this assumption the joint posterior distribution at time  $t$  is approximated by substitution into the prediction-correction equations:

$$p(x_{0:t}|y_{0:t}) \approx \frac{1}{Z} \sum_{i=1}^N \omega_{t-1}^{(i)} \delta_{x_{0:t-1}^{(i)}}(x_{0:t-1}) g(y_t|x_t) f\left(x_t|x_{t-1}^{(i)}\right), \quad (4.60)$$

where the normalization factor  $Z$  is given by

$$Z = \sum_{j=1}^N \omega_{t-1}^{(j)} \int f\left(x|x_{t-1}^{(j)}\right) g(y_t|x) dx.$$

Now we consider a general joint proposal for the entire path of the new particles  $x_{0:t}^{(i)}$ , that is,

$$\begin{aligned} q_t(x_{0:t}) &= q_{0:t-1}(x_{0:t-1}|y_{0:t}) q_t(x_t|x_{t-1}, y_t) \\ &= \left( \sum_{i=1}^N v_{t-1}^{(i)} \delta_{x_{0:t-1}^{(i)}}(x_{0:t-1}) \right) \\ &\quad \times \left( q_t(x_t|x_{t-1}^{(i)}, y_t) \right), \end{aligned}$$

where  $\sum_{i=1}^N v_{t-1}^{(i)} = 1$  and  $v_{t-1}^{(i)} > 0$ . Notice that the proposal splits into two parts: a marginal proposal  $q_{0:t-1}$  for the past path of the process  $x_{0:t-1}$  and a conditional proposal  $q_t$  for the new state  $x_t$ . Note that the first component is constructed to depend explicitly on data up to time  $t$  in order to allow adaptation of the proposal in the light of the new data point  $y_t$  (and indeed it may depend on future data points as well if some look-ahead and latency is allowable). The first part of the proposal is a discrete distribution centered upon the old particle paths  $\{x_{0:t-1}^{(i)}\}$ , but now with probability mass for each component in the proposal distribution designed to be  $\{v_{t-1}^{(i)}\}$ . The weighting function  $v_{t-1}^{(i)}$  can be data dependent, the rationale being that we should preselect particles that are a good fit to the new data point  $y_t$ . For example, Pitt and Shephard [53] suggest taking a point value  $\mu^{(i)}$  of the state, say the mean or mode of  $f(x_t|x_{t-1}^{(i)})$ , and computing the weighting function as the likelihood evaluated at this point, i.e.,  $v_{t-1}^{(i)} = g(y_t|\mu_t^{(i)})$ ; or if the particles from  $t - 1$  are weighted, one would choose  $v_{t-1}^{(i)} = \omega_{t-1}^{(i)} g(y_t|\mu_t^{(i)})$ .

Using this proposal mechanism it is then possible to define an importance ratio between the approximate posterior in (4.60) and the full path proposal  $q$ , given by

$$\tilde{\omega}_t^{(i)} = \frac{\omega_{t-1}^{(i)}}{v_{t-1}^{(i)}} \times \frac{g\left(y_t \mid x_t^{(i)}\right) f\left(x_t^{(i)} \mid x_{t-1}^{(i)}\right)}{q_t\left(x_t^{(i)} \mid x_{t-1}^{(i)}, y_t\right)}.$$

The choice of the proposal terms  $v_{t-1}$  and  $q_t$  then determines the behavior of the particle filter. With  $v_{t-1}$  constant and  $q_t$  set to equal the transition density, we achieve the standard bootstrap filter of [34]. With  $v_{t-1}$  constant and  $q_t$  chosen to be a data-dependent proposal for the new state, we have the particle filter of [59].

More general schemes that allow some exploration of future data points by so-called pilot sampling to generate the weighting function have been proposed in, for example [60], while further discussion of the framework can be found in [61]. A summary of the auxiliary particle filter is given in Algorithm 1. We assume here that the selection (resampling) step occurs at each point, although it may be omitted exactly as in the standard particle filter, in which case no weight correction is applied. These techniques are developed and extensively reviewed in [37, 39, 44, 59, 62, 63].

A diagrammatic representation of the bootstrap filter in operation is given in Figure 4.2, in which the resampling (selection) step is seen to concentrate particles (asterisks) into the two high probability modes of the density function.

#### 3.04.4.4.1 Marginalized particle filters

In many practical scenarios, especially those found in the tracking domain, the models are not entirely nonlinear and non-Gaussian. By this we mean that some subset of the state vector is linear and Gaussian, *conditional upon* the other states. In these cases one may use standard linear Gaussian optimal filtering for the linear part, and particle filtering for the nonlinear part. This may be thought of as an optimal Gaussian mixture approximation to the filtering distribution. See [59, 64, 65] for detailed descriptions of this approach to the problem, which is referred to either as the *Rao-Blackwellized* particle filter, or *Mixture Kalman* filter. Recent work [66, 67] has studied in detail the possible classes of model that may be handled by the marginalized filter, and computational complexity issues. The formulation is as follows.<sup>3</sup> First, the state is partitioned into two components,  $x_t^L$  and  $x_t^N$ , referring respectively to the linear (“L”) and nonlinear (“N”) components. The linear part of the model is expressed in the form of a linear Gaussian state-space model as follows, with state-space matrices that may depend upon the nonlinear state  $x_t^N$ :

---

<sup>3</sup>Karlsson et al. [67] and Schön et al. [66] present a more general class of models to which the marginalized filter may be applied, but we present a more basic framework for the sake of simplicity here.

**Algorithm 1.** Auxiliary Particle Filter

---

**for**  $i = 1, \dots, N$  **do** ▷Initialisation

 Sample  $\tilde{x}_0^{(i)} \sim q_0(x_0|y_0)$ .

Assign initial importance weights

$$\tilde{\omega}_0^{(i)} = \frac{g(y_0|\tilde{x}_0^{(i)})\pi_0(\tilde{x}_0^{(i)})}{q_0(\tilde{x}_0^{(i)}|y_0)}.$$

**end for**
**for**  $t = 1, \dots, T$  **do**

 Select  $N$  particle indices  $j_i \in \{1, \dots, N\}$  according to weights

$$\{v_{t-1}^{(i)}\}_{i=1}^N.$$

**for**  $i = 1, \dots, N$  **do**

 Set  $x_{t-1}^{(i)} = \tilde{x}_{t-1}^{(j_i)}$ .

Set first stage weights:

$$u_{t-1}^{(i)} = \frac{\omega_{t-1}^{(j_i)}}{v_{t-1}^{(j_i)}}.$$

**end for**
**for**  $i = 1, \dots, N$  **do**

Propagate:

$$\tilde{x}_t^{(i)} \sim q_t(\tilde{x}_t^{(i)} | x_{t-1}^{(i)}, y_t).$$

Compute weight:

$$\tilde{\omega}_t^{(i)} = u_{t-1}^{(i)} \frac{g(y_t | \tilde{x}_t^{(i)}) f(\tilde{x}_t^{(i)} | x_{t-1}^{(i)})}{q_t(\tilde{x}_t^{(i)} | x_{t-1}^{(i)}, y_t)}.$$

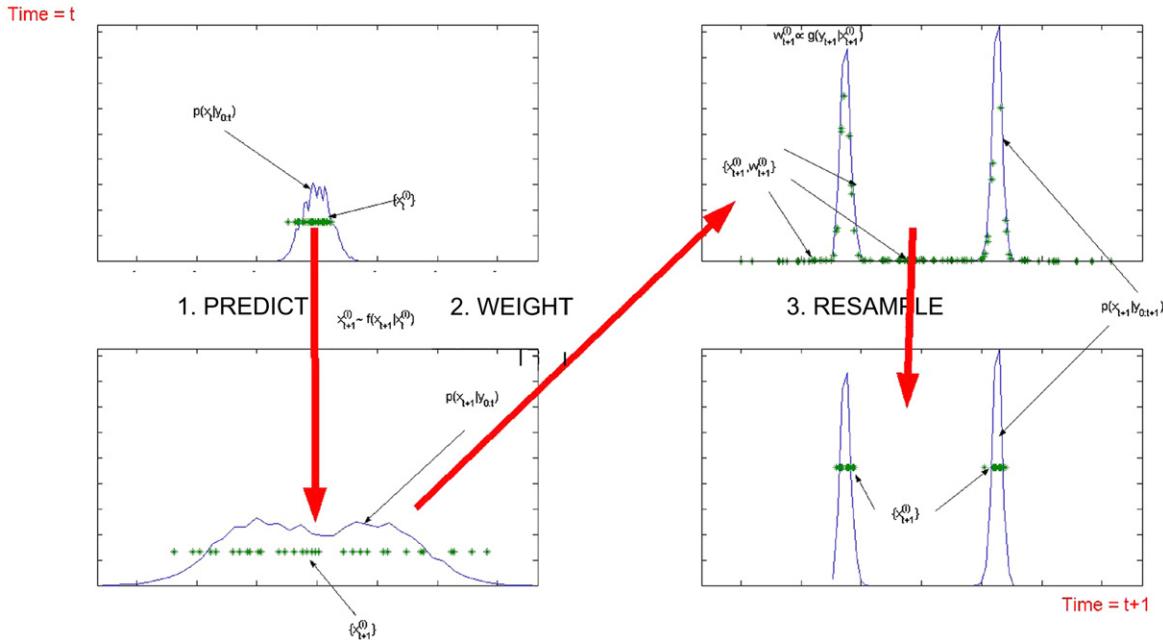
**end for**

Normalize weights:

$$\omega_t^{(i)} = \tilde{\omega}_t^{(i)} \left/ \sum_{j=1}^N \tilde{\omega}_t^{(j)} \right., \quad i = 1, \dots, N.$$

---

**end for**

**FIGURE 4.2**

The bootstrap filter in operation from time  $t$  to  $t + 1$ , nonlinear time series Example 1. Asterisks show the positions of (a small selection of) the particles at each stage. The solid line shows a kernel density estimate of the distributions represented at each stage. Ten thousand particles were used in total. Notice that resampling concentrates particles into the region of high probability.

$$x_t^L = A \left( x_t^N \right) x_{t-1}^L + u_t^L, \quad (4.61)$$

$$y_t = B \left( x_t^N \right) x_t^L + v_t^L. \quad (4.62)$$

Here  $u_t^L$  and  $v_t^L$  are independent, zero-mean, Gaussian disturbances with covariances  $C_u$  and  $C_v$ , respectively, and  $A(\cdot)$  and  $B(\cdot)$  are matrices of compatible dimensions that may depend upon the nonlinear state  $x_t^N$ . At  $t = 0$ , the linear part of the model is initialised with  $x_0^L \sim N(\mu_0(x_0^N), P_0(x_0^N))$ .

Now the nonlinear part of the state obeys a general dynamical model (which is not necessarily Markovian):

$$x_t^N \sim f \left( x_t^N \mid x_{0:t-1}^N \right), \quad x_0^N \sim f \left( x_0^N \right). \quad (4.63)$$

In such a case, conditioning on the nonlinear part of the state  $x_{0:t}^N$  and the observations  $y_{0:t}$ , the linear part of the state is jointly Gaussian and the means and covariances of this Gaussian representation may be obtained by using the classical Kalman filtering recursions [68]. The basic idea is then to *marginalize* the linear part of the state vector to obtain the posterior distribution of the nonlinear part of the state:

$$p \left( x_{0:t}^N \mid y_{0:t} \right) = \int p \left( x_{0:t}^L, x_{0:t}^N \mid y_{0:t} \right) dx_{0:t}^L.$$

Particle filtering is then run on the nonlinear state sequence only, with target distribution  $p(x_{0:t}^N \mid y_{0:t})$ . The resulting algorithm is almost exactly as before, requiring only a slight modification to the basic particle filter to allow for the fact that the marginalized system is no longer Markovian, since

$$p(y_t \mid y_{0:t-1}, x_{0:t}^N) \neq p(y_t \mid x_t^N).$$

Moreover, the dynamical model for the nonlinear part of the state may itself be non-Markovian, see Eq. (4.63).

Thus, instead of the usual updating rule we have:

- *Prediction:*

$$p \left( x_{0:t}^N \mid y_{0:t-1} \right) = p \left( x_{0:t-1}^N \mid y_{0:t-1} \right) f \left( x_t^N \mid x_{0:t-1}^N \right). \quad (4.64)$$

- *Correction:*

$$p \left( x_{0:t}^N \mid y_{0:t} \right) = \frac{p \left( y_t \mid y_{0:t-1}, x_{0:t}^N \right) p \left( x_{0:t}^N \mid y_{0:t-1} \right)}{p(y_t \mid y_{0:t-1})}, \quad (4.65)$$

where as before  $p(y_t \mid y_{0:t-1})$  is the predictive distribution of  $y_t$  given the past observations  $y_{0:t-1}$ , which is a fixed normalizing constant (independent of the state sequence  $x_{0:t}^N$ ).

Note that if  $\{(x_{0:t}^{N,(i)}, \omega_t^{(i)})\}_{i=1,\dots,N}$  denote the particles evolving in the state-space of the nonlinear variables according to the above equations, and their associated importance weights, estimation of the

linear part of the state may be done using a *Rao-Blackwellized* estimation scheme [69]: the posterior density for the linear part is obtained as a random Gaussian mixture approximation given by

$$p\left(x_t^L \mid y_{0:t}\right) \approx \sum_{i=1}^N \omega_t^{(i)} p\left(x_t^L \mid x_{0:t}^{N,(i)}, y_{0:t}\right), \quad (4.66)$$

where the conditional densities  $p(x_t^L \mid x_{0:t}^{N,(i)}, y_{0:t})$  are Gaussian and computed again using Kalman filtering recursions. Equation (4.66) replaces the standard point-mass approximation (4.59) arising in the generic particle filter. The Rao-Blackwellized estimate is usually better in terms of Monte Carlo error than the corresponding scheme that performs standard particle filtering jointly in both nonlinear and linear states. The computational trade-off is more complex, however, since the marginalized filter can be significantly more time-consuming than the standard filter *per particle*. These trade-offs have been extensively studied by Schön et al. [66] and in many cases the performance/computation trade-off comes out in favor of the marginalized filter.

To give further detail to the approach, we first summarize the Kalman filter itself in this probabilistic setting [52], then we place the whole scheme back in the particle filtering context. As a starting point, assume the distribution  $p(x_{t-1}^L \mid y_{0:t-1}, x_{0:t-1}^N)$  has been obtained. This is a Gaussian, denoted by

$$p\left(x_{t-1}^L \mid y_{0:t-1}, x_{0:t-1}^N\right) = N\left(x_{t-1}^L \mid \mu_{t-1|0:t-1}, C_{t-1|0:t-1}\right),$$

where the mean and covariance terms are dependent upon both  $y_{0:t-1}$  and  $x_{0:t-1}^N$ . Now, (4.61) shows how to update this distribution one step, since  $x_t^L$  is just a summation of two transformed independent Gaussian random vectors,  $A(x_t^N)x_{t-1}^L$  and  $u_t^L$ , which itself must be a Gaussian. Under the standard rules for summation of independent Gaussian random vectors, we obtain the predictive distribution for  $x_t^L$ , conditioned upon  $y_{0:t-1}$  and  $x_{0:t}^N$ , as follows:

$$p\left(x_t^L \mid y_{0:t-1}, x_{0:t}^N\right) = N\left(x_t^L \mid \mu_{t|0:t-1}, C_{t|0:t-1}\right), \quad (4.67)$$

where

$$\begin{aligned} \mu_{t|0:t-1} &= A\left(x_t^N\right) \mu_{t-1|0:t-1}, \\ C_{t|0:t-1} &= A\left(x_t^N\right) C_{t-1|0:t-1} A\left(x_t^N\right)^T + C_u. \end{aligned}$$

As a second step in the update, the new data point  $y_t$  is incorporated through Bayes' theorem:

$$\begin{aligned} p\left(x_t^L \mid y_{0:t}, x_{0:t}^N\right) &= \frac{p\left(x_t^L \mid y_{0:t-1}, x_{0:t}^N\right) \times p\left(y_t \mid x_t^L, x_t^N\right)}{p\left(y_t \mid y_{0:t-1}, x_{0:t}^N\right)} \\ &\propto N\left(x_t^L \mid \mu_{t|0:t-1}, C_{t|0:t-1}\right) \times N\left(y_t \mid B\left(x_t^N\right) x_t^L, C_v\right) \\ &= N\left(x_t^L \mid \mu_{t|0:t}, C_{t|0:t}\right), \end{aligned} \quad (4.68)$$

where  $\mu_{t|0:t}$  and  $C_{t|0:t}$  are obtained by standard rearrangement formulae as

$$\begin{aligned}\mu_{t|0:t} &= \mu_{t|0:t-1} + K_t \left( y_t - B \left( x_t^N \right) \mu_{t|0:t-1} \right), \\ C_{t|0:t} &= \left( I - K_t B \left( x_t^N \right) \right) C_{t|0:t-1}, \\ K_t &= C_{t|0:t-1} B^T \left( x_t^N \right) \left( B \left( x_t^N \right) C_{t|0:t-1} B^T \left( x_t^N \right) + C_v \right)^{-1},\end{aligned}$$

and where the term  $K_t$  is known as the *Kalman Gain*. In order to complete the analysis for particle filter use, one further term is required,  $p(y_t | y_{0:t-1}, x_{0:t}^N)$ . This is obtained by the so-called *prediction error decomposition*, which is easily obtained from (4.67), since  $y_t$  is obtained by summing a transformed version of  $x_t^L$ , i.e.,  $B(x_t^N)x_t^L$ , with an independent zero-mean Gaussian noise term  $v_t^L$  having covariance  $C_v$ , leading to:

$$p \left( y_t | y_{0:t-1}, x_{0:t}^N \right) = N \left( y_t | \mu_{y_t}, C_{y_t} \right), \quad (4.69)$$

where

$$\begin{aligned}\mu_{y_t} &= B \left( x_t^N \right) \mu_{t|0:t-1}, \\ C_{y_t} &= B \left( x_t^N \right) C_{t|0:t-1} B^T \left( x_t^N \right) + C_v.\end{aligned}$$

In order to construct the marginalized particle filter, notice that for any realization of the nonlinear state sequence  $x_{0:t}^N$  and data sequence  $y_{0:t}$ , one may calculate the value of  $p(y_t | y_{0:t-1}, x_{0:t}^N)$  in (4.69) through sequential application of the formulae (4.67) and (4.68). The marginalized particle filter then requires computation and storage of the term  $p(y_t | y_{0:t-1}, x_{0:t}^{N,(i)})$  in (4.69), for each particle realization  $x_{0:t}^{N,(i)}$ . In the marginalized particle filter the particles are stored as the nonlinear part of the state  $x_t^N$ , the associated sufficient statistics for each particle, i.e.,  $\mu_{t|0:t}$  and  $C_{t|0:t}$ , and the weight for each particle. We do not give the entire modified algorithm. The only significant change is to the weighting step, which becomes

$$\tilde{\omega}_t^{(i)} = \omega_{t-1}^{(i)} \frac{p \left( y_t | y_{0:t-1}, \tilde{x}_t^{N,(i)} \right) f \left( \tilde{x}_t^{N,(i)} | x_{0:t-1}^{N,(i)} \right)}{q_t \left( \tilde{x}_t^{N,(i)} | x_{0:t-1}^{N,(i)}, y_{0:t} \right)}.$$

As an important aside, we note that the marginalized filter may also be used to good effect when the linear states are unknown but “static” over time, i.e.,  $f(dx_t^L | x_{t-1}^L) = \delta_{x_{t-1}^L}(dx_t^L)$  with some Gaussian initial distribution or prior  $x_0^L \sim N(\mu_0(x_0^N), P_0(x_0^N))$ , as before. Then the marginalized filter runs exactly as before but we are now able to marginalize, or infer the value of, a static parameter  $\theta = x_t^L$ . Early versions of such filters are found in the sequential imputations work of [48], for example.

We have focused here on the linear Gaussian case of the marginalized filter. However, another important class of models is the discrete state-space hidden Markov model, in which the states are discrete values and switching may occur between one time and the next according to a Markov transition matrix. As for the linear Gaussian case, the discrete state values may be marginalized to form a marginalized particle filter, using the HMM forward algorithm [70] instead of the Kalman filter [59]. For simulations and examples within both frameworks, see [37, 59].

As mentioned before, several generalisations are possible to the basic model. The most basic of these allow dependence of the matrices  $A()$ ,  $B()$ ,  $C_u$ , and  $C_v$  to depend on time, and any or all elements of the nonlinear state sequence  $x_{0:t}^N$ . None of these changes require any modification to the algorithm formulation. Another useful case allows a deterministic function of the nonlinear states to be present in the observation and dynamical equations. These two features combined lead to the following form:

$$\begin{aligned} x_t^L &= A_t \left( x_{0:t}^N \right) x_{t-1}^L + c \left( x_{0:t}^N \right) + u_t^L, \\ y_t &= B_t \left( x_{0:t}^N \right) x_t^L + d \left( x_{0:t}^N \right) + v_t^L, \end{aligned}$$

and again the form of the algorithm is unchanged; see [9] for a good coverage of the most general form of Kalman filters required in these cases.

One other important case involves nonlinear observations that are not a function of the linear state. Then the linear observation Eq. (4.62) can be generalized to  $y_t \sim g(y_t | x_t^N)$ , which is a general observation density. This form is quite useful in tracking examples, where observation functions are often nonlinear (range and bearings, for example, or range-only), but dynamics can be considered as linear to a good approximation [64, 66, 67]. If in addition the nonlinear state can be expressed in linear Gaussian state-space form with respect to the linear state, i.e.:

$$\begin{aligned} x_t^N &= B \left( x_t^N \right) x_t^L + c \left( x_{t-1}^N \right) + v_t^L, \\ x_t^L &= A \left( x_t^N \right) x_{t-1}^L + u_t^L, \end{aligned}$$

then once again the Kalman filter can be run to marginalize the linear state variable. In this case the weight expression becomes:

$$\tilde{\omega}_t^{(i)} = \omega_{t-1}^{(i)} \frac{g \left( y_t \mid \tilde{x}_t^{N,(i)} \right) p \left( \tilde{x}_t^{N,(i)} \mid x_{0:t-1}^{N,(i)} \right)}{q_t \left( \tilde{x}_t^{N,(i)} \mid x_{0:t-1}^{N,(i)}, y_{0:t} \right)},$$

where now the term  $p(\tilde{x}_t^{N,(i)} | x_{0:t-1}^{N,(i)})$  is computed using the Kalman filter. In some cases the linear state transition matrices and observation matrices  $A()$  and  $B()$  for this Kalman filter are independent of the nonlinear state and the observations; then this form of marginalized particle filter may be computed very efficiently, since the covariance matrices are identical for all particles and thus need only be computed once at each time step.

#### 3.04.4.4.2 Further material

In this section on particle filtering we have given a general introduction. For further material please see [50, 71, 72]. For smoothing with particle filters, see for example [35, 58, 59, 73–81], for parameter estimation with particle filters [34, 37, 63, 77, 82–93], and for recent combinations with MCMC [94].

---

### 3.04.5 Conclusion

This article has introduced the basic principles of parameter inference and state estimation within the Bayesian framework. We have considered a linear Gaussian parametric model by way of illustration. We have also given an introductory coverage to some of the principal computational tools available for the practitioner, including MCMC and the particle filter.

---

## A Probability densities and integrals

### A.1 Univariate Gaussian

The univariate Gaussian, or normal, density function with mean  $\mu$  and variance  $\sigma^2$  is defined for a real-valued random variable as:

$$N(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}, \quad (4.70)$$

Univariate normal density.

### A.2 Multivariate Gaussian

The multivariate Gaussian probability density function (PDF) for a column vector  $\mathbf{x}$  with  $N$  real-valued components is expressed in terms of the mean vector  $\mathbf{m}_x$  and the covariance matrix  $\mathbf{C}_x = E[(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T]$  as:

$$N_N(\mathbf{x}|\mathbf{m}, \mathbf{C}_x) = \frac{1}{(2\pi)^{N/2} |\mathbf{C}_x|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \mathbf{m}_x)^T \mathbf{C}_x^{-1} (\mathbf{x} - \mathbf{m}_x)\right), \quad (4.71)$$

Multivariate Gaussian density.

An integral which is used on many occasions throughout the text is of the general form:

$$I = \int_{\mathbf{y}} \exp\left(-\frac{1}{2} (a + \mathbf{b}^T \mathbf{y} + \mathbf{y}^T \mathbf{C} \mathbf{y})\right) d\mathbf{y}, \quad (4.72)$$

where  $d\mathbf{y}$  is interpreted as the infinitesimal volume element:

$$d\mathbf{y} = \prod_{i=1}^N dy_i$$

and the integral is over the real line in all dimensions, i.e., the single integration sign should be interpreted as:

$$\int_{\mathbf{y}} \equiv \int_{y_1=-\infty}^{\infty} \cdots \int_{y_N=-\infty}^{\infty} .$$

For non-singular symmetric  $\mathbf{C}$  it is possible to form a “perfect square” for the exponent and hence simplify the integral. Take negative of twice the exponent:

$$a + \mathbf{b}^T \mathbf{y} + \mathbf{y}^T \mathbf{C} \mathbf{y} \quad (4.73)$$

and try to express it in the form:

$$(\mathbf{y} - \mathbf{m}_y)^T \mathbf{C} (\mathbf{y} - \mathbf{m}_y) + k$$

for some constants  $k$  and  $\mathbf{C}$ , to be determined. Multiplying out this expression leads to the required form:

$$\mathbf{y}^T \mathbf{C} \mathbf{y} + \mathbf{m}_y^T \mathbf{C} \mathbf{m}_y - 2\mathbf{m}_y^T \mathbf{C} \mathbf{y} + k.$$

Here we have used the fact that  $\mathbf{m}_y^T \mathbf{C} \mathbf{y} = \mathbf{y}^T \mathbf{C} \mathbf{m}_y$ , since  $\mathbf{C}$  is assumed symmetric.

Hence, equating the constant, linear and quadratic terms in  $\mathbf{y}$ , we arrive at the result:

$$a + \mathbf{b}^T \mathbf{y} + \mathbf{y}^T \mathbf{C} \mathbf{y} = (\mathbf{y} - \mathbf{m}_y)^T \mathbf{C} (\mathbf{y} - \mathbf{m}_y) + k, \quad (4.74)$$

where

$$\mathbf{m}_y = -\frac{\mathbf{C}^{-1} \mathbf{b}}{2}$$

and

$$k = \left( a - \frac{\mathbf{b}^T \mathbf{C}^{-1} \mathbf{b}}{4} \right).$$

Thus the integral  $I$  can be re-expressed as:

$$\begin{aligned} I &= \int_{\mathbf{y}} \exp \left( -\frac{1}{2} \left( (\mathbf{y} - \mathbf{m}_y)^T \mathbf{C} (\mathbf{y} - \mathbf{m}_y) \right) \right) \\ &\quad \times \exp \left( -\frac{1}{2} \left( a - \frac{\mathbf{b}^T \mathbf{C}^{-1} \mathbf{b}}{4} \right) \right) d\mathbf{y}, \end{aligned} \quad (4.75)$$

where, again

$$\mathbf{m}_y = -\frac{\mathbf{C}^{-1} \mathbf{b}}{2}.$$

Comparison with the multivariate PDF of (4.71) which has unity volume leads directly to the result:

$$\begin{aligned} &\int_{\mathbf{y}} \exp \left( -\frac{1}{2} \left( a + \mathbf{b}^T \mathbf{y} + \mathbf{y}^T \mathbf{C} \mathbf{y} \right) \right) d\mathbf{y} \\ &= \frac{(2\pi)^{N/2}}{|\mathbf{C}|^{1/2}} \exp \left( -\frac{1}{2} \left( a - \frac{\mathbf{b}^T \mathbf{C}^{-1} \mathbf{b}}{4} \right) \right), \end{aligned} \quad (4.76)$$

Multivariate Gaussian integral.

This result can be also be obtained directly by a transformation which diagonalizes  $\mathbf{C}$  and this approach then verifies the normalization constant given for the PDF of (4.71).

### A.3 Gamma density

Another distribution which will be of use is the two parameter gamma density  $G(\alpha, \beta)$ , defined for  $\alpha > 0, \beta > 0$  as

$$G(y|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} \exp(-\beta y) \quad (0 < y < \infty), \quad (4.77)$$

Gamma density.

$\Gamma()$  is the Gamma function (see e.g., [95]), defined for positive arguments. This distribution with its associated normalization enables us to perform marginalisation of scale parameters with Gaussian likelihoods and a wide range of parameter priors (including uniform, Jeffreys, Gamma, and Inverse Gamma (see [96])) priors which all require the following result:

$$\int_{y=0}^{\infty} y^{\alpha-1} \exp(-\beta y) dy = \Gamma(\alpha)/\beta^\alpha, \quad (4.78)$$

Gamma integral.

Furthermore the mean, mode and variance of such a distribution are obtained as:

$$\mu = E[Y] = \alpha/\beta, \quad (4.79)$$

$$m = \underset{y}{\operatorname{argmax}}(p(y)) = (\alpha - 1)/\beta, \quad (4.80)$$

$$\sigma^2 = E[(Y - \mu)^2] = \alpha/\beta^2. \quad (4.81)$$

### A.4 Inverted-gamma distribution

A closely related distribution is the inverted-gamma distribution,  $IG(\alpha, \beta)$  which describes the distribution of the variable  $1/Y$ , where  $Y$  is distributed as  $G(\alpha, \beta)$ :

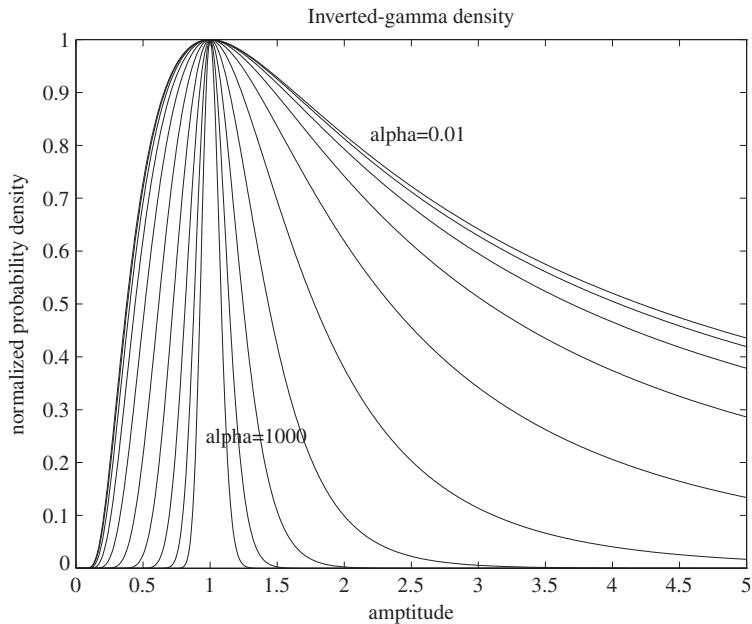
$$IG(y|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{-(\alpha+1)} \exp(-\beta/y) \quad (0 < y < \infty), \quad (4.82)$$

Inverted-gamma density.

The IG distribution has a unique maximum at  $\beta/(\alpha + 1)$ , mean value  $\beta/(\alpha - 1)$  (for  $\alpha > 1$ ) and variance  $\beta^2/((\alpha - 1)^2(\alpha - 2))$  (for  $\alpha > 2$ ).

It is straightforward to see that the improper Jeffreys prior  $p(x) = 1/x$  is obtained in the limit as  $\alpha \rightarrow 0$  and  $\beta \rightarrow 0$ .

The family of IG distributions is plotted in Figure A.1 as  $\alpha$  varies over the range 0.01–1000 and with maximum value fixed at unity. The variety of distributions available indicates that it is possible to incorporate either very vague or more specific prior information about variances by choice of the mode and degrees of freedom of the distribution. With high values of  $\alpha$  the prior is very tightly clustered around its mean value, indicating a high degree of prior belief in a small range of values, while for smaller  $\alpha$  the prior can be made very diffuse, tending in the limit to the uninformative Jeffreys prior. Values of  $\alpha$  and  $\beta$  might be chosen on the basis of mean and variance information about the unknown parameter or from estimated percentile positions on the axis.



**FIGURE A.1**

Inverted-gamma family with mode = 1,  $\alpha = 0.01 \dots 1000$ .

## A.5 Normal-inverted-gamma distribution

$$p(\boldsymbol{\theta}, \sigma^2) = p(\boldsymbol{\theta}|\sigma^2)p(\sigma^2) = N(\mathbf{m}_{\boldsymbol{\theta}}, \mathbf{M}_{\boldsymbol{\theta}}^{-1}/\sigma^2). \quad (4.83)$$

## A.6 Wishart distribution

If  $\boldsymbol{\Lambda}$  is a  $P \times P$  symmetric positive definite matrix, the Wishart distribution is defined as

$$Wi(\boldsymbol{\Lambda}|\mathbf{M}, \alpha) = \frac{\pi^{-P(P-1)/4}}{\prod_{i=1}^P \Gamma(0.5(2\alpha + 1 - i))} |\mathbf{M}|^\alpha |\boldsymbol{\Lambda}|^{\alpha-(P+1)/2} \exp(-\text{tr}(\mathbf{M}\boldsymbol{\Lambda})), \quad (4.84)$$

where  $\alpha > (P - 1)/2$ . The mean of the distribution is

$$E[\boldsymbol{\Lambda}] = \alpha \mathbf{M}^{-1}$$

and the mode (maximum) is

$$(\alpha - (P + 1)/2) \mathbf{M}^{-1}.$$

### A.7 Inverse Wishart distribution

The inverse Wishart distribution is the distribution of  $\mathbf{C} = \boldsymbol{\Lambda}^{-1}$ , where  $\boldsymbol{\Lambda}$  has the Wishart distribution as described above,

$$\text{IW}_i(\mathbf{C}|\mathbf{M}, \alpha) = \frac{\pi^{-P(P-1)/4}}{\prod_{i=1}^P \Gamma(0.5(2\alpha + 1 - i))} |\mathbf{M}|^\alpha |\mathbf{C}|^{-(\alpha + (P+1)/2)} \exp(-\text{tr}(\mathbf{MC}^{-1})). \quad (4.85)$$

The mean of the distribution is

$$E[\mathbf{C}] = \frac{\mathbf{M}}{\alpha - (P + 1)/2}$$

and the mode (maximum) is

$$\frac{\mathbf{M}}{\alpha + (P + 1)/2}.$$

## References

- [1] J. Makhoul, Linear prediction: a tutorial review, Proc. IEEE 63 (4) (1975) 561–580.
- [2] C.W. Therrien, Discrete Random Signals and Statistical Signal Processing, Prentice-Hall, 1992.
- [3] J.M. Bernardo, A.F.M. Smith, Bayesian Theory, John Wiley & Sons, 1994.
- [4] G.E.P. Box, G.C. Tiao, Bayesian Inference in Statistical Analysis, Addison-Wesley, 1973.
- [5] H. Jeffreys, Theory of Probability, Oxford University Press, 1939.
- [6] C.P. Robert, The Bayesian Choice, second ed., Springer, New York, 2001.
- [7] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, 1973.
- [8] H. VanTrees, Decision, Estimation and Modulation Theory, Part 1, Wiley and Sons, 1968.
- [9] A.C. Harvey, Forecasting Structural Time Series Models and the Kalman Filter, Cambridge University Press, 1989.
- [10] G. Kitagawa, W. Gersch, Smoothness Priors Analysis of Time Series, Lecture Notes in Statistics, vol. 116, Springer-Verlag, New York, 1996.
- [11] A. Zellner, On assessing prior distributions and Bayesian regression analysis with g-prior distribution, in: Bayesian Inference and Decision Techniques: Essays in Honour of Bruno de Finetti, Elsevier, 1986, pp. 233–243.
- [12] H. Akaike, A new look at the statistical model identification, IEEE Trans. Automat Control 19 (6) (1974) 716–723.
- [13] D. Madigan, J. York, Bayesian graphical models for discrete data, Int. Stat. Rev. 63 (1995) 215–232.
- [14] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Roy. Stat. Soc. Ser. B 39 (1) (1977) 1–38.
- [15] M. Feder, A.V. Oppenheim, E. Weinstein, Maximum likelihood noise cancellation using the EM algorithm, IEEE Trans. Acoust. Speech Signal Process. 37 (2) (1989).
- [16] T.K. Moon, The expectation-maximization algorithm, IEEE Signal Process. Mag. (1996) 47–60.
- [17] J.J.K. Ó Ruanaidh, W.J. Fitzgerald, Numerical Bayesian Methods Applied to Signal Processing, Springer-Verlag, 1996.
- [18] E. Weinstein, A.V. Oppenheim, M. Feder, J.R. Buck, Iterative and sequential algorithms for multisensor signal enhancement, IEEE Trans. Signal Process. 42 (4) (1994).

- [19] M.A. Tanner, Tools for Statistical Inference, second ed., Springer-Verlag, 1993.
- [20] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
- [21] David J.C. MacKay, Information Theory, Inference, and Learning Algorithms, CUP, 2003.
- [22] W.K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* 57 (1970) 97–109.
- [23] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculations by fast computing machines, *J. Chem. Phys.* 21 (1953) 1087–1091.
- [24] L. Tierney, Markov chains for exploring posterior distributions (with discussion), *Ann. Stat.* 22 (1994) 1701–1762.
- [25] A.E. Gelfand, A.F.M. Smith, Sampling-based approaches to calculating marginal densities, *J. Am. Stat. Assoc.* 85 (1990) 398–409.
- [26] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, *IEEE Trans. Pattern Anal. Mach. Intell.* 6 (1984) 721–741.
- [27] J. Liu, W.H. Wong, A. Kong, Covariance structure of the Gibbs sampler with applications to the comparison of estimators and augmentation schemes, *Biometrika* 81 (1994) 27–40.
- [28] G.O. Roberts, S.K. Sahu, Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler, *J. Roy. Stat. Soc. Ser. B* 59 (1997) 291–317.
- [29] M.K. Cowles, B.P. Carlin, Markov chain Monte Carlo convergence diagnostics—a comparative review, *J. Am. Stat. Assoc.* 91 (434) (1996) 883–904.
- [30] S.J. Godsill, On the relationship between MCMC methods for model uncertainty, *J. Comput. Graph. Stat.* 10 (2001) 230–248.
- [31] P.J. Green, Reversible jump Markov chain Monte Carlo computation and Bayesian model determination, *Biometrika* 82 (1995) 711–732.
- [32] B.D.O. Anderson, J.B. Moore, Optimal Filtering, Prentice-Hall, 1979.
- [33] R.E. Kalman, A new approach to linear filtering and prediction problems, *ASME J. Basic Eng.* 82 (1960) 35–45.
- [34] N. Gordon, D. Salmond, A.F. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE Proc. F, Radar Signal Process.* 140 (1993) 107–113.
- [35] G. Kitagawa, Monte-Carlo filter and smoother for non-Gaussian nonlinear state space models, *J. Comput. Graph. Stat.* 1 (1996) 1–25.
- [36] M. West, Mixture models, Monte Carlo, Bayesian updating and dynamic models, *Comput. Sci. Stat.* 24 (1993) 325–333.
- [37] O. Cappé, E. Moulines, T. Rydén, Inference in Hidden Markov Models, Springer, 2005.
- [38] T. Kailath, A. Sayed, B. Hassibi, Linear Estimation, Prentice-Hall, 2000.
- [39] B. Ristic, M. Arulampalam, A. Gordon, Beyond Kalman Filters: Particle Filters for Target Tracking, Artech House, 2004.
- [40] A.H. Jazwinski, Stochastic Processes and Filtering Theory, Academic Press, New York, 1970.
- [41] D.L. Alspach, H.W. Sorenson, Nonlinear Bayesian estimation using Gaussian sum approximation, *IEEE Trans. Automat. Control* 17 (4) (1972) 439–448.
- [42] K. Ito, K. Xiong, Gaussian filters for nonlinear filtering problems, *IEEE Trans. Automat. Control* 45 (2000) 910–927.
- [43] S.J. Julier, J.K. Uhlmann, A new extension of the Kalman filter to nonlinear systems, in: Aerosense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, 1997.
- [44] R. Van der Merwe, A. Doucet, N. De Freitas, E. Wan, The unscented particle filter, in: T.K. Leen, T.G. Dietterich, V. Tresp (Eds.), Advanced in Neural Information Processing System, vol. 13, MIT Press, 2000.

- [45] J. Handschin, D. Mayne, Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering, *Int. J. Control.* 9 (1969) 547–559.
- [46] J. Handschin, Monte Carlo techniques for prediction and filtering of non-linear stochastic processes, *Automatica* 6 (1970) 555–563.
- [47] A. Blake, M. Isard, *Active Contours*, Springer, 1998.
- [48] J. Liu, R. Chen, Blind deconvolution via sequential imputations, *J. Roy. Stat. Soc. Ser. B* 430 (1995) 567–576.
- [49] P. Del Moral, Nonlinear filtering: interacting particle solution, *Markov Process. Rel. Fields* 2 (1996) 555–579.
- [50] A. Doucet, N. De Freitas, N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, Springer, New York, 2001.
- [51] J.S. Liu, *Monte Carlo Strategies in Scientific Computing*, Springer, New York, 2001.
- [52] Y.C. Ho, R.C.K. Lee, A Bayesian approach to problems in stochastic estimation and control, *IEEE Trans. Automat. Control* 9 (4) (1964) 333–339.
- [53] M.K. Pitt, N. Shephard, Filtering via simulation: auxiliary particle filters, *J. Am. Stat. Assoc.* 94 (446) (1999) 590–599.
- [54] N. Chopin, Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference, *Ann. Stat.* 32 (6) (2004) 2385–2411.
- [55] D. Crisan, A. Doucet, A survey of convergence results on particle filtering methods for practitioners, *IEEE Trans. Signal Process.* 50 (3) (2002) 736–746.
- [56] P. Del Moral, Measure-valued processes and interacting particle systems, application to nonlinear filtering problems, *Ann. Appl. Prob.* 8 (1998) 69–95.
- [57] P. Del Moral, *Feynman-Kac Formulae, Genealogical and Interacting Particle Systems with Applications*, Springer, 2004.
- [58] H.R. Künsch, Recursive Monte-Carlo filters: algorithms and theoretical analysis, *Ann. Stat.* 33 (5) (2005) 1983–2021.
- [59] A. Doucet, S. Godsill, C. Andrieu, On sequential Monte-Carlo sampling methods for Bayesian filtering, *Stat. Comput.* 10 (2000) 197–208.
- [60] J.L. Zhang, J.S. Liu, A new sequential importance sampling method and its application to the two-dimensional hydrophobic-hydrophilic model, *J. Chem. Phys.* 117 (7) (2002).
- [61] S. Godsill, T. Clapp, Improvement strategies for Monte Carlo particle filters, in: A. Doucet, N. De Freitas, N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, Springer, 2001.
- [62] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for on line non-linear/non-Gaussian Bayesian tracking, *IEEE Trans. Signal Process.* 50 (2002) 241–254.
- [63] N. Shephard, M. Pitt, Likelihood analysis of non-Gaussian measurement time series, *Biometrika* 84 (3) (1997) 653–667 (Erratum in volume 91, 249–250, 2004).
- [64] C. Andrieu, A. Doucet, Particle filtering for partially observed Gaussian state space models, *J. Roy. Stat. Soc. Ser. B* 64 (4) (2002) 827–836.
- [65] R. Chen, J.S. Liu, Mixture Kalman filter, *J. Roy. Stat. Soc. Ser. B* 62 (3) (2000) 493–508.
- [66] T. Schön, F. Gustafsson, P.-J. Nordlund, Marginalized particle filters for mixed linear/nonlinear state-space models, *IEEE Trans. Signal Process.* 53 (7) (2005) 2279–2289.
- [67] R. Karlsson, T. Schön, F. Gustafsson, Complexity analysis of the marginalized particle filter, *IEEE Trans. Signal Process.* 53 (11) (2005) 4408–4411.
- [68] R.E. Kalman, R. Bucy, New results in linear filtering and prediction theory, *J. Basic Eng. Trans. ASME, Ser. D* 83 (3) (1961) 95–108.
- [69] C.P. Robert, G. Casella, *Monte Carlo Statistical Methods*, second ed., Springer, New York, 2004.
- [70] L.R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, *Proc. IEEE* 77 (2) (1989) 257–285.
- [71] O. Cappé, S.J. Godsill, E. Moulines, An overview of existing methods and recent advances in sequential Monte Carlo, *Proc. IEEE* 95 (5) (2007).

- [72] A. Doucet, A.M. Johansen, A tutorial on particle filtering and smoothing: fifteen years later, in: D. Crisan, B. Rozovsky (Eds.), Oxford Handbook of Nonlinear Filtering, OUP, 2011.
- [73] M. Briers, A. Doucet, S. Maskell, Smoothing algorithms for state-space models, Technical Report TR-CUED-F-INFENG 498, Department of Engineering, University of Cambridge, 2004.
- [74] W. Fong, S. Godsill, A. Doucet, M. West, Monte Carlo smoothing with application to audio signal enhancement, *IEEE Trans. Signal Process.* 50 (2) (2002) 438–449.
- [75] S.J. Godsill, A. Doucet, M. West, Maximum a posteriori sequence estimation using Monte Carlo particle filters, *Ann. Inst. Stat. Math.* 53 (1) (2001) 82–96.
- [76] S.J. Godsill, A. Doucet, M. West, Monte Carlo smoothing for non-linear time series, *J. Am. Stat. Assoc.* 50 (2004) 438–449.
- [77] M. Hürzeler, H.R. Künsch, Monte Carlo approximations for general state-space models, *J. Comput. Graph. Stat.* 7 (1998) 175–193.
- [78] M. Isard, A. Blake, CONDENSATION—conditional density propagation for visual tracking, *Int. J. Comput. Vis.* 29 (1) (1998) 5–28.
- [79] M. Klaas, M. Briers, N. De Freitas, A. Doucet, S. Maskell, D. Lang, Fast particle smoothing: if I had a million particles, in: 23rd International Conference on Machine Learning (ICML), Pittsburgh, Pennsylvania, June 25–29, 2006.
- [80] H.R. Künsch, State space and hidden Markov models, in: O.E Barndorff-Nielsen, D.R. Cox, C. Klueppelberg (Eds.), Complex Stochastic Systems, CRC Publisher, Boca Raton, 2001, pp. 109–173.
- [81] S. Sarkka, P. Bunch, S. Godsill, A backward-simulation based rao-blackwellized particle smoother for conditionally linear gaussian models, in: 16th IFAC Symposium on System Identification, 2012.
- [82] R. Shumway, D. Stoffer, An approach to time series smoothing and forecasting using the EM algorithm, *J. Time Ser. Anal.* 3 (4) (1982) 253–264.
- [83] C. Andrieu, A. Doucet, S.S. Singh, V.B. Tadic, Particle methods for change detection, system identification, and control, *IEEE Proc.* 92 (3) (2004) 423–438.
- [84] F. Campillo, F. Le Gland, MLE for partially observed diffusions: direct maximization vs. the EM algorithm, *Stochast. Process. Appl.* 33 (1989) 245–274.
- [85] N. Chopin, A sequential particle filter method for static models, *Biometrika* 89 (2002) 539–552.
- [86] P. Del Moral, A. Doucet, A. Jasra, Sequential monte carlo samplers, *J. Roy. Stat. Soc. Ser. B* 68 (3) (2006) 411.
- [87] P. Fearnhead, Markov chain Monte Carlo, sufficient statistics and particle filter, *J. Comput. Graph. Stat.* 11 (4) (2002) 848–862.
- [88] Walter R. Gilks, Carlo Berzuini, Following a moving target—Monte Carlo inference for dynamic Bayesian models, *J. Roy. Stat. Soc. Ser. B* 63 (1) (2001) 127–146.
- [89] G. Kitagawa, A self-organizing state-space model, *J. Am. Stat. Assoc.* 93 (443) (1998) 1203–1215.
- [90] J. Liu, M. West, Combined parameter and state estimation in simulation-based filtering, in: N. De Freitas, A. Doucet, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer, 2001.
- [91] R.M. Neal, Annealed importance sampling, *Stat. Comput.* 11 (2) (2001) 125–139.
- [92] Jimmy Olsson, Tobias Rydén, Asymptotic properties of the bootstrap particle filter maximum likelihood estimator for state space models, Technical Report LUTFMS-5052-2005, Lund University, 2005.
- [93] M. Segal, E. Weinstein, A new method for evaluating the log-likelihood gradient, the Hessian, and the Fisher information matrix for linear dynamic systems, *IEEE Trans. Inform. Theory* 35 (1989) 682–687.
- [94] C. Andrieu, A. Doucet, R. Holenstein, Particle Markov chain monte carlo methods, *J. Roy. Stat. Soc. Ser. B (Stat. Methodol.)* 72 (3) (2010) 269–342.
- [95] A.C. Bajpai, L.R. Mustoe, D. Walker, Advanced Engineering Mathematics, Wiley, 1977.
- [96] Manouchehr Kheradmandnia, Aspects of Bayesian Threshold Autoregressive Modelling, PhD Thesis, University of Kent, 1991.

# Distributed Signal Detection<sup>1</sup>

# 5

Pramod K. Varshney\* and Engin Masazade†

\*Syracuse University, Department of Electrical Engineering and Computer Science,  
4-206 Center for Science and Technology, Syracuse, NY, USA

†Department of Electrical and Electronics Engineering, Yeditepe University, Istanbul, Turkey

## 3.05.1 Introduction

There are many practical situations in which one is faced with a decision-making problem. For example, in a radar detection context multiple radars work together to make a decision regarding the presence or absence of a target based on the radar returns [1]. In a digital communication system, one of the possible waveforms is transmitted over a channel and based on the received noisy observations, one needs to determine the symbol that was transmitted [2,3]. In a biomedical application [4,5], based on a smear of human tissue, one needs to determine if it is cancerous or not. In a pattern recognition problem [6,7], the type of the aircraft being observed needs to be determined based on some aircraft features. In cognitive radio networks [8,9], detection of spectrum holes and opportunistic use of under-utilized frequency bands without causing harmful interference to legacy networks is an essential functionality. In a wireless sensor network (WSN), detection of an event of interest is an important task of the network before other attributes of the event are estimated (see [10,11] for different application scenarios). In all of the above applications, the common underlying problem is to make a decision among several possible choices. This is carried out based on available noisy measurements. The branch of statistics dealing with these types of problems is known as statistical decision theory or hypothesis testing. In the context of radar and communication theory, it is known as detection theory [12–15]. In distributed signal detection, multiple detectors (sensors) work collaboratively to distinguish between two or more hypotheses, e.g., the absence or presence of a signal of interest. Deployment of multiple sensors for signal detection improves system survivability, results in improved detection performance or in a shorter decision time to attain a prespecified performance level.

In classical multi-sensor detection, local sensors transmit their raw observations to a central processor where optimal detection is carried out based on conventional statistical techniques. However, centralized processing based on raw observations is neither efficient nor necessary in many practical applications. It may consume excessive energy and bandwidth in communications, may impose a heavy computation burden at the central processor. In distributed processing [12,16,17], on the other hand, local sensors can carry out preliminary processing of data and only communicate with each other

<sup>1</sup>This work was supported by US Air Force Office of Scientific Research (AFOSR) under Grant FA9550-10-1-0263 and Army Research Office (ARO) Grant W911NF-09-1-0244.

and/or the fusion center with the most informative information relevant to the global objective. With advances in sensing technology and wireless communications, wireless sensors can be deployed in situ to monitor phenomena of interest with increased precision and coverage. This has given rise to many detection problems involving wireless sensor networks. Such detection ability of a wireless sensor network is crucial for various applications. As an example, in a surveillance scenario, the presence or absence of a target is usually determined before attributes, such as its position or velocity, are estimated. Since wireless sensors are assumed to be tiny battery powered devices with limited on board signal processing capabilities, energy limitation is one of the major differences between a WSN and other wireless networks such as wireless local area networks. Therefore, prolonging the lifetime of a WSN is important for both commercial and tactical applications. For instance, in order to maximize battery lifetime and reduce communication bandwidth, it is essential for each sensor to locally compress its observed data and transmit quantized measurements so that only low rate inter-sensor or sensor to fusion center communication is required. The advantages of distributed processing are obvious, i.e., reduced communication bandwidth requirement and energy consumption, increased reliability and robustness. Moreover, there may be channel impairments such as fading and path loss in the network environment that can considerably degrade the quality of wireless links among sensors. Such challenges should be taken into account while designing the communication and local signal processing algorithms.

This survey paper is organized as follows. In Section 3.05.2, we focus on the distributed detection problem with independent sensor observations. Under the conditional independence assumption, the optimal design of decision rules at the local sensors and at the fusion center is discussed for both Bayesian and Neyman-Pearson formulations. We subsequently talk about non-parametric, computationally and energy-efficient distributed detection approaches. Distributed detection over fading channels is also addressed in this section. The problem of distributed detection in the presence of dependent observations is considered in Section 3.05.3. Finally, the summary of the paper and some challenging issues for distributed detection are presented in Section 3.05.4.

### 3.05.2 Distributed detection with independent observations

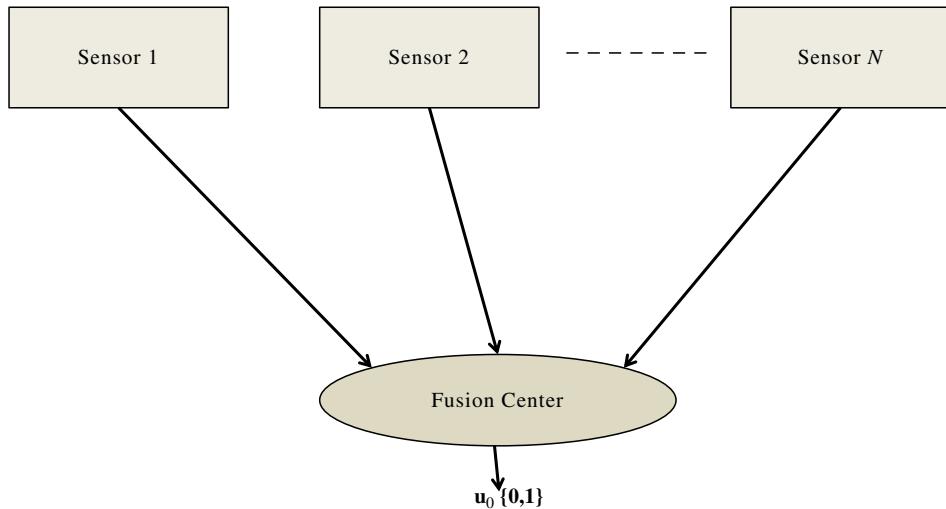
In this paper, we focus on a simple binary hypothesis testing problem in which the observations at each of the  $N$  sensors,  $y_i, i \in \{1, 2, \dots, N\}$ , correspond to either of the two hypotheses,

$$\begin{aligned} H_0 &\sim p_0(\boldsymbol{\theta}), \\ H_1 &\sim p_1(\boldsymbol{\theta}), \end{aligned} \tag{5.1}$$

where  $p_0(\boldsymbol{\theta})$  and  $p_1(\boldsymbol{\theta})$  are the probability density functions (pdfs) under  $H_0$  and  $H_1$  respectively. More specifically, if the problem is to detect the absence or presence of the signal of interest, hypothesis  $H_1$  represents the presence of a signal and  $H_0$  represents the absence of a signal. Then the distributed detection problem has the form,

$$y_i = \begin{cases} n_i, & \text{under } H_0, \\ \mathbf{c}^T \boldsymbol{\theta} + n_i, & \text{under } H_1, \end{cases} \tag{5.2}$$

where  $\boldsymbol{\theta}$  represent the parameter vector that is characterizing the hypothesis  $H_1$ , if the measurement model is linear  $\mathbf{c}$  is an appropriate scaling vector which might have the same size as  $\boldsymbol{\theta}$ . If the measurement

**FIGURE 5.1**

Parallel configuration.

model is non-linear, the received sensor measurements have the form,

$$y_i = \begin{cases} n_i, & \text{under } H_0, \\ f(\theta) + n_i, & \text{under } H_1, \end{cases} \quad (5.3)$$

where  $f(\cdot)$  can be modeled according to the isotropic signal emission model [18], or disk model [19].  $n_i$  represents the noise samples. In this paper, we focus on the binary hypothesis testing problem. More detailed treatment for the multiple hypothesis testing (classification) problem can be found in the literature [6, 7, 20–23].

Parallel configuration, as shown in Figure 5.1, is the most common topological structure that has been studied quite extensively in the literature (see [11, 24] and references therein). The sensors do not communicate with each other and there is no feedback from the fusion center to any sensor. Sensors either transmit their measurements  $y_i$  directly to the fusion center or send a quantized version of their local measurements defined by the mapping rule  $u_i = \gamma_i(y_i)$ . Based on the received information  $\mathbf{u} = [u_1, \dots, u_N]$ , the fusion center arrives at a global decision  $u_0 = \gamma_0(\mathbf{u})$  that favors either  $H_1$  (decides  $u_0 = 1$ ) or  $H_0$  (decides  $u_0 = 0$ ). The goal is to obtain the optimal set of decision rules  $\Gamma = (\gamma_0, \gamma_1, \dots, \gamma_N)$  according to the objective function under consideration which can be formulated according to Bayesian formulation or Neyman-Pearson formulation. For general network structures, the optimal solution to the distributed detection problem, i.e., the optimal decision rules  $(\gamma_1, \dots, \gamma_N)$ , is NP-complete, i.e., in general cannot be solved in polynomial time [25–27]. Nonetheless, under certain assumptions and specific network topologies, the optimum solution becomes tractable.

### 3.05.2.1 Conditional independence assumption

The conditional independence assumption implies that the joint density of the observations obeys

$$p(y_1, \dots, y_N | H_j) = \prod_{i=1}^N p(y_i | H_j), \quad \text{for } j = 0, 1. \quad (5.4)$$

Consider a scenario in which the observations at the sensors are conditionally independent as well as identically distributed. The symmetry in the problem suggests that the decision rules at the sensors should be identical. But counterexamples have been found in which nonidentical decision rules are optimal [27–30]. In the following subsections, the decision rules at local sensors and the fusion center are designed according to Bayesian and Neyman-Pearson formulations under the parallel configuration.

#### 3.05.2.1.1 Bayesian formulation

In the Bayesian formulation, certain costs are assigned to different courses of actions under the knowledge of prior probabilities, and the objective is to minimize the Bayesian risk of the overall system operation which can be expressed as

$$\min_{\Gamma=(y_0, y_1, \dots, y_N)} R, \quad (5.5)$$

where the Bayes risk  $R$  is defined as

$$\begin{aligned} R &\triangleq \sum_{i=0}^1 \sum_{j=0}^1 C_{ij} P(u_0 = i | H_j) P_j \\ &= C + C_F \sum_{\mathbf{u}} P(u_0 = 1 | \mathbf{u}) P(\mathbf{u} | H_0) - C_D \sum_{\mathbf{u}} P(u_0 = 1 | \mathbf{u}) P(\mathbf{u} | H_1). \end{aligned} \quad (5.6)$$

Here,  $C_{ij}$  is defined to be the cost of declaring  $H_i$  true when  $H_j$  is present to reflect different consequences of all decisions.  $C_F = P_0(C_{10} - C_{00})$ ,  $C_D = (1 - P_0)(C_{01} - C_{11})$  and  $C = C_{01}(1 - P_0) + C_{00}P_0$ .  $\sum_{\mathbf{u}}$  indicates summation over all possible values of  $\mathbf{u}$ . Under the conditionally independence assumption, it can be shown that the sensor decision rules and the fusion rule are likelihood ratio tests (LRTs) given by [12]. The LRT at each sensor has the form,

$$\frac{p(y_i | H_1)}{p(y_i | H_0)} \stackrel{u_i=1}{\gtrless} \frac{\sum_{\mathbf{u}^i} C_F A(\mathbf{u}^i) \prod_{k=1, k \neq i}^N P(u_k | H_0)}{\sum_{\mathbf{u}^i} C_D A(\mathbf{u}^i) \prod_{k=1, k \neq i}^N P(u_k | H_1)} \quad \text{for } i = 1, \dots, N. \quad (5.7)$$

Note that the LRT of each sensor depends on the decision rules of the other sensors. Given that the decision rules of the other sensors remain fixed, the right-hand side of (5.7) becomes constant. Then, the LRT at the fusion center is,

$$\prod_{i=1}^N \frac{P(u_i | H_1)}{P(u_i | H_0)} \stackrel{u_0=1}{\gtrless} \frac{C_F}{C_D}, \quad (5.8)$$

where

$$\begin{aligned}\mathbf{u}^i &= [u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_N], \\ A(\mathbf{u}^i) &= P(u_0 = 1|\mathbf{u}^{i1}) - P(u_0 = 1|\mathbf{u}^{i0}), \\ \mathbf{u}^{ij} &= [u_1, \dots, u_{i-1}, u_i = j, u_{i+1}, \dots, u_N], \quad j = 0, 1.\end{aligned}$$

To find the optimal set of decision rules amounts to simultaneously solving the above  $N$  coupled nonlinear equations. A popular method to design distributed detection systems is to employ a person-by-person optimization (PBPO) technique. This technique consists of optimizing the decision rule of one sensor at a time while keeping the decision rules of the remaining sensors fixed. The overall performance at the fusion center is guaranteed to improve (or, at least, to not worsen) with every iteration of the PBPO algorithm. However, system design equations resulting from this PBPO procedure represent necessary but not, in general, sufficient conditions to determine the globally optimum solution. The Gauss-Seidel cyclic coordinate descent algorithm has been proposed in the literature [31,32] to obtain the PBPO solution satisfying the necessary conditions of optimality in an iterative manner.

### 3.05.2.1.2 Neyman-Pearson formulation

The Neyman-Pearson formulation of the distributed detection problem can be stated as follows: for a prescribed bound on the global probability of false alarm,  $P_f = P(u_0 = 1|H_0)$ , find (optimum) local and global decision rules that maximize the global probability of detection  $P_d = P(u_0 = 1|H_1)$  as,

$$\begin{aligned}\max \quad & P_d, \\ \text{s.t. } & P_f \leq \alpha.\end{aligned}\tag{5.9}$$

Under the conditional independence assumption, the mapping rules at the sensors as well as the decision rule at the fusion center are threshold rules based on the appropriate likelihood ratios [33,34]:

$$\frac{p(y_i|H_1)}{p(y_i|H_0)} \begin{cases} > t_i, & \text{then } u_i = 1, \\ = t_i, & \text{then } u_i = 1 \text{ with probability } \epsilon_i, \\ < t_i, & \text{then } u_i = 0, \end{cases}\tag{5.10}$$

for  $i = 1, \dots, N$ , and

$$\prod_{i=1}^N \frac{P(u_i|H_1)}{P(u_i|H_0)} \begin{cases} > \lambda_0, & \text{decide } H_1 \text{ or set } u_0 = 1, \\ = \lambda_0, & \text{randomly decide } H_1 \text{ with probability } \epsilon, \\ < \lambda_0, & \text{decide } H_0 \text{ or set } u_0 = 0. \end{cases}\tag{5.11}$$

If the likelihood ratio in (5.10) is a continuous random variable,  $\epsilon_i$  can be assumed to be zero. The threshold  $\lambda_0$  in (5.11) as well as the local thresholds  $t_i$  in (5.10) need to be determined so as to maximize  $P_d$  for a given  $P_f = \alpha$ . This can still be quite difficult because the local decision rules and the global fusion rule are coupled to each other [12]. Since (5.11) is known to be a monotone fusion rule, one can solve for the set of optimal local thresholds  $\{t_i, i = 1, \dots, N\}$  for a given monotone fusion rule and compute the corresponding  $P_d$ . One can then successively consider other possible monotone fusion rules and obtain the corresponding detection probabilities. The final optimal solution is the one monotone

fusion rule and the corresponding local decision rules that provide the largest  $P_d$ . An iterative gradient method was proposed in [35] to find the thresholds satisfying the preassigned false alarm probability. Finding the optimal solution in this fashion is possible only for very small values of  $N$ . The complexity increases with  $N$  because (1) the number of monotone rules grows exponentially with  $N$  and (2) finding the optimal  $\{t_i, i = 1, \dots, N\}$  for a given fusion rule is an optimization problem involving an  $N - 1$  dimensional search (it is one dimension less than  $N$  because of the constraint  $P_f = \alpha$ ).

### 3.05.2.1.3 The decision fusion problem

Given the local detectors, the problem is to determine the fusion rule to combine local decisions optimally. Let us first consider the case where local detectors makes only hard decisions, i.e.,  $u_i$  can take only two values 0 or 1 corresponding to the two hypotheses  $H_0$  and  $H_1$  respectively. Let  $P_{fi}$  and  $P_{di}$  denote the probabilities of false alarm and detection of sensor  $i$  respectively, i.e.,  $P_{fi} = P(u_i = 1|H_0)$  and  $P_{di} = P(u_i = 1|H_1)$ . As we know, the optimum fusion rule is given by the likelihood ratio test:

$$\prod_{i=1}^N \frac{P(u_i|H_1)}{P(u_i|H_0)} \stackrel{u_0=1}{\gtrless} \lambda. \quad (5.12)$$

Here,  $\lambda$  is determined by the optimization criterion in use. The left-hand side of (5.12) can be written as

$$\begin{aligned} \prod_{i=1}^N \frac{P(u_i|H_1)}{P(u_i|H_0)} &= \prod_{i=1}^N \left( \frac{P(u_i = 1|H_1)}{P(u_i = 1|H_0)} \right)^{u_i} \left( \frac{P(u_i = 0|H_1)}{P(u_i = 0|H_0)} \right)^{1-u_i} \\ &= \prod_{i=1}^N \left( \frac{P_{di}}{P_{fi}} \right)^{u_i} \left( \frac{1 - P_{di}}{1 - P_{fi}} \right)^{1-u_i}. \end{aligned} \quad (5.13)$$

Taking the logarithm of both sides of (5.12), we have the Chair-Varshney fusion rule [36]

$$\sum_{i=1}^N \left[ u_i \log \frac{P_{di}}{P_{fi}} + (1 - u_i) \log \frac{1 - P_{di}}{1 - P_{fi}} \right] \stackrel{u_0=1}{\gtrless} \log \lambda. \quad (5.14)$$

This rule can also be expressed as

$$\sum_{i=1}^N \left[ \log \frac{P_{di}(1 - P_{fi})}{P_{fi}(1 - P_{di})} \right] u_i \stackrel{u_0=1}{\gtrless} \underbrace{\log \lambda + \sum_{i=1}^N \log \frac{1 - P_{fi}}{1 - P_{di}}}_{\lambda'}. \quad (5.15)$$

Thus, the optimum fusion rule can be implemented by forming a weighted sum of the incoming local decisions and comparing it with a threshold  $\lambda'$ . The weights and the threshold are determined by the local probabilities of detection and false alarm. If the local decisions have the same statistics, i.e.,  $P_{f,1} = \dots = P_{f,N}$  and  $P_{d,1} = \dots = P_{d,N}$ , the Chair-Varshney fusion rule reduces to a  $K$ -out-of- $N$  form or a counting rule, i.e., the global decision  $u_0 = 1$  if  $K$  or more sensor decisions are one. This structure of the fusion rule reduces the computational complexity considerably.

If the local detectors are allowed to make multilevel or soft decisions, the observation space at each local detector is partitioned into  $L$  mutually exclusive regions with  $L > 2$ . If the observation at detector  $i$  lies in the partition  $l$ , we set  $u_i = l, l = 0, \dots, L - 1$ . Define  $\alpha_i^l = P(u_i = l|H_0)$  and  $\beta_i^l = P(u_i = l|H_1)$ . The likelihood ratio in (5.12) can be written as

$$\prod_{i=1}^N \frac{P(u_i|H_1)}{P(u_i|H_0)} = \prod_{l=0}^{L-1} \prod_{S_l} \frac{\beta_i^l}{\alpha_i^l}, \quad (5.16)$$

where  $S_l$  is the set of local decisions  $u_i$  that are equal to  $l$ . Taking the logarithm of both sides, the optimal fusion rule is as follows:

$$\sum_{l=0}^{L-1} \sum_{S_l} \log \frac{\beta_i^l}{\alpha_i^l} \stackrel{u_0=1}{\gtrless} \stackrel{u_0=0}{\gtrless} \log \lambda. \quad (5.17)$$

This fusion rule is a generalization of the fusion rule for the hard decision case [12].

So far, we have assumed that the parameters characterizing a hypothesis,  $\boldsymbol{\theta}$ , are fixed and known and the corresponding detection problem is known as simple hypothesis testing. In many situations, however, these parameters can take unknown values or a range of values. Such hypotheses are called composite hypotheses and the corresponding detection problem is known as composite hypothesis testing. If  $\boldsymbol{\theta}$  is characterized as a random vector with known probability densities under the two hypotheses, the LRT can be extended to composite hypothesis testing in a straightforward manner:

$$\Lambda(\mathbf{y}) = \frac{\int_{\Theta_1} p(\mathbf{y}|\boldsymbol{\theta}_1) p(\boldsymbol{\theta}|H_1) d\boldsymbol{\theta}}{\int_{\Theta_0} p(\mathbf{y}|\boldsymbol{\theta}_0) p(\boldsymbol{\theta}|H_0) d\boldsymbol{\theta}}, \quad (5.18)$$

where  $\boldsymbol{\theta}_j$  is the random variable characterizing hypothesis  $H_j$  and  $\Theta_j$  is the support of  $\boldsymbol{\theta}_j$ . If  $\boldsymbol{\theta}$  is nonrandom, one can use the maximum likelihood estimates of its value under the two hypotheses as the true values in an LRT, resulting in a so called generalized likelihood ratio test (GLRT) [37]:

$$\Lambda_g(\mathbf{y}) = \frac{\max_{\boldsymbol{\theta} \in \Theta_1} p(\mathbf{y}|\boldsymbol{\theta}_1, H_1)}{\max_{\boldsymbol{\theta} \in \Theta_0} p(\mathbf{y}|\boldsymbol{\theta}_0, H_0)} \stackrel{D_1}{\gtrless} \stackrel{D_0}{\gtrless} \eta. \quad (5.19)$$

Note that the optimum Neyman-Pearson or Bayesian detectors involve a likelihood ratio test (LRT) as in (5.12). The complete knowledge of the likelihoods,  $p(\mathbf{u}|H_1)$  and  $p(\mathbf{u}|H_0)$ , may not always be available in a practical application. Also, there are many detection problems where the exact form of the LRT is too complicated to implement. Therefore, simpler and more robust suboptimal detectors are used in numerous applications [38]. For some suboptimal detectors, the detection performance can be improved by adding an independent noise to the observations under certain conditions which is known as stochastic resonance (SR) [39]. Given a suboptimal fixed detector, the work in [40] first discusses the improbability of the detection performance by adding SR noise. If the performance can be improved, then the best noise type is determined in order to maximize  $P_D$  without increasing  $P_F$ . The work in [41] discusses variable detectors.

### 3.05.2.1.4 Asymptotic regime

It has been shown that for sensors whose observations are independent and identically distributed given either hypothesis, identical decision rules are optimal in the asymptotic regime where the number of sensors increases to infinity [27, 42]. In other words, the identical decision rule assumption often results in little or no loss of optimality. Therefore, identical local decision rules are frequently assumed in many situations, which reduces the computational complexity considerably.

For any reasonable collection of decision rules  $\Gamma$ , the probability of error at the fusion center goes to zero exponentially as the number of sensors  $N$  grows unbounded. It is then adequate to compare collections of decision rules based on their exponential rate of convergence to zero,

$$\lim_{N \rightarrow \infty} \frac{\log Pe(\Gamma)}{N}. \quad (5.20)$$

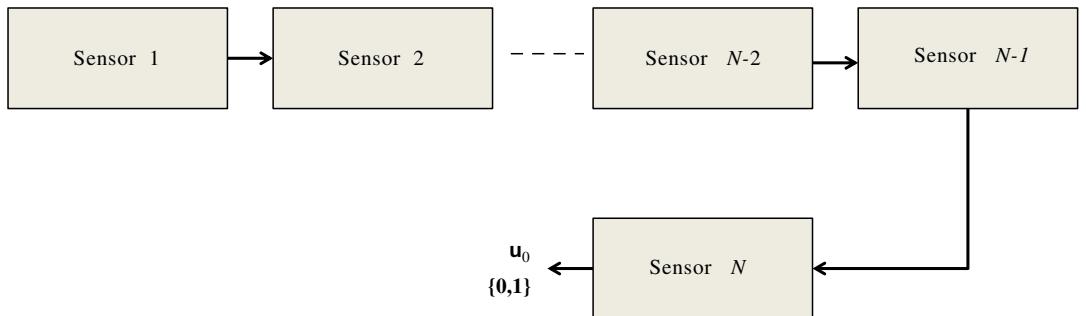
It was shown that, for the binary hypothesis testing problem, using identical local decision rules for all the sensor nodes is asymptotically optimal in terms of the error exponent [42]. In [43], the exact asymptotics of the minimum error probabilities achieved by the optimal parallel fusion network and the system obtained by imposing the identical decision rule constraint was investigated. It was shown analytically that the restriction of identical decision rules leads to little or no loss of performance. Asymptotic regimes applied to distributed detection are convenient because they capture the dominating behaviors of large systems. This leads to valuable insights into the problem structure and its solution.

In [44], by studying in the asymptotic regime, it is shown that sensors transmit binary decisions that are optimal if there exists a binary quantization function  $\gamma_b$  whose Chernoff information exceeds half of the information contained in an unquantized observation. The requirement is fulfilled by many practical applications [45] such as the problem of detecting deterministic signals in Gaussian noise and the problem of detecting fluctuating signals in Gaussian noise using a square-law detector. In these scenarios, the gain offered by having more sensor nodes outperforms the benefits of getting detailed information from each sensor.

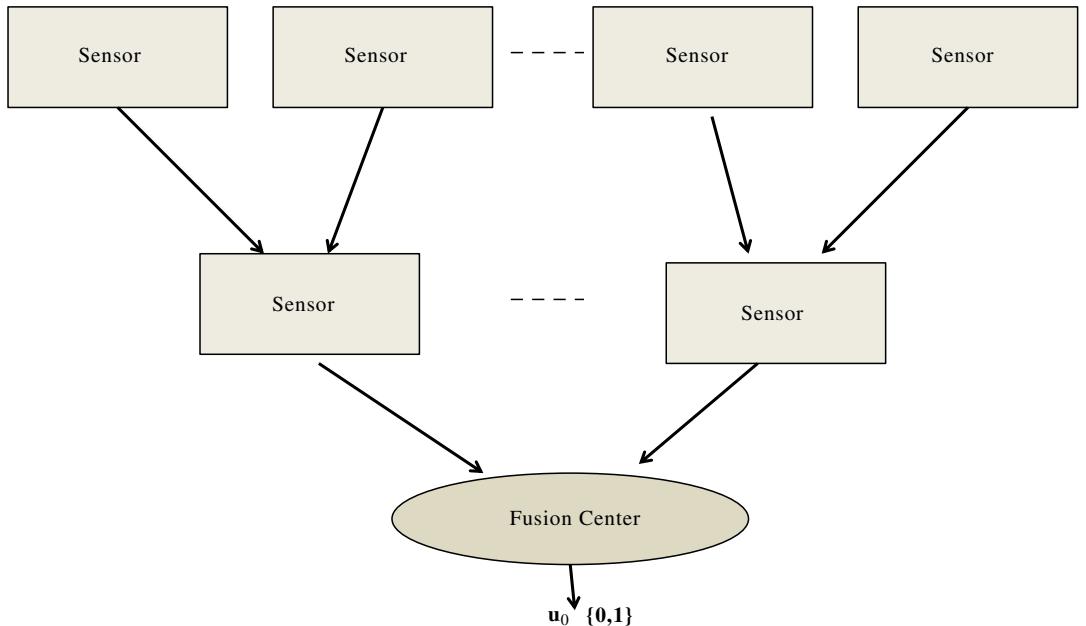
### 3.05.2.2 Other network topologies

Solutions for arbitrary topologies such as serial and tree have been derived and are discussed in [12, 46]. In serial, or tandem, topology (Figure 5.2) [12, 47–49] all of the sensors are connected in series and receive direct observations from the common phenomenon. The decision of the first sensor is based solely on its observation. This decision is transmitted to the second sensor which uses it in conjunction with its direct observation. Then the decision of the second sensor is sent to its successor. This process is repeated at each sensor until the decision of the last sensor is accepted as the final decision of the system.

One can also envisage configurations that are combinations of parallel and serial topologies, e.g., a tree or hierarchical network as shown in Figure 5.3 [50–52]. In the tree topology, the leaf sensors report to their cluster heads and the cluster heads reports to their cluster heads hierarchically until reaching the fusion center. It has been shown that the detection performance of parallel topology is superior than that of serial and tree topologies [12, 53]. Under the asymptotic regime, the work in [53] lists the conditions where parallel and tree topologies achieve the same detection performance. Moreover, one may also incorporate feedback in the system design as in [54, 55]. There are other topologies also where a subset of the sensors are allowed to both transmit their messages to the fusion center and to also broadcast

**FIGURE 5.2**

Serial configuration.

**FIGURE 5.3**

Tree configuration.

them to the remaining sensors [56]. In the asymptotic regime and for the Neyman-Pearson formulation, it has been proved in [56] that sharing of decisions does not improve the optimal error exponent.

In the parallel fusion structure, the channels between sensors and the fusion center have been assumed as mutually independent. In sensor networks with a large number of sensors, this assumption is often

impractical and violated. An alternative is to utilize the multiple access channel (MAC), by taking advantage of the shared nature of the wireless medium. Decentralized detection and estimation have been considered as a MAC channel model in [57–62].

So far, we have discussed fixed-sample-size detection problems where the fusion center arrives at a decision after receiving the entire set of sensor observations. Sequential detectors may choose to stop at any time and make a final decision or continue to take an additional observations [63–66]. Moreover, in consensus-based detection [67–69], which requires no fusion center, sensors first collect sufficient observations over a period of time. Then, they subsequently run the consensus algorithm to fuse their local log likelihood ratios.

### 3.05.2.3 Nonparametric rules in distributed detection

Most of the results discussed so far on distributed detection are based on the assumption that the local sensors' detection performances, namely either the local sensors' signal to noise ratio (SNR) or their probability of detection and false alarm rate, are known to the fusion center. For a wireless sensor network consisting of passive sensors, it might be very difficult to estimate local sensors' performances via experiments because sensors' distances from the signal of interest might be unknown to the fusion center and to the local sensors. Even if the local sensors can somehow estimate their detection performances in real time, it can be still very expensive to transmit them to the fusion center, especially for a WSN with very limited system resources. Hence, the knowledge of the local sensors' performances cannot be taken for granted and a fusion rule that does not require local sensors' performances is highly preferable. Without the knowledge of local sensors' detection performances and their positions, an approach at the fusion center is to treat every sensor equally. An intuitive solution is to use the total number of “1”s as a statistic since the information about which sensor reports a “1” is of little use to the fusion center. In [18, 70, 71], a counting based fusion rule is proposed, which uses the total number of detections (“1”s) transmitted from local sensors as the statistic,

$$\Lambda = \sum_{i=1}^N u_i \stackrel{u_0=1}{\gtrless} T, \quad (5.21)$$

where  $T$  is the threshold at the fusion center, which can be decided by a pre-specified probability of false alarm  $P_F$ . This fusion rule is called the counting rule. It is an attractive solution, since it is quite simple to implement, and achieves very good detection performance in a WSN with randomly and densely deployed low-cost sensor nodes.

When the received signal intensity at a sensor is assumed to be inversely proportional its the distance from the source location, the sensor measurements and hence their decisions become conditionally independent as long as the source location is known exactly. In [72], a generalized likelihood ratio test (GLRT) based decision fusion method that uses quantized data from local sensors has been proposed which jointly detects and localizes a target in a wireless sensor field. The GLRT based fusion method significantly improves detection performance, as compared with the counting rule. Moreover, in a scenario where the sensors which are located within the target's finite radius of influence, receive identical target signal and the rest of the sensors do not receive any target signal, the authors in [19] control the false discovery rate (FDR) to determine the local sensor decision rules that are nonidentical.

FDR based method improves the global detection performance as compared to employing identical decision rules at each sensor.

### 3.05.2.4 Energy efficient distributed detection and multi-objective optimization

In many detection applications, the event of interest occurs infrequently and the null hypothesis is observed for the majority of time. For sensor networks operating on limited energy resources, an energy-efficient transmission technique is sensor censoring where sensors transmit only when the observations indicate that the target event is likely [73–76]. When the event of interest becomes very unlikely, sensor nodes can afford to go to sleep for an extended period of time, thus saving energy. On the other hand, when in a critical situation, sensor nodes must stay awake [77, 78].

In distributed detection in wireless sensor networks, let us consider a scenario where each sensor compares its local observation with a threshold and transmits a binary decision to the fusion center. Also consider, each sensor employs sensor censoring that is, a sensor transmits its decision to the fusion center only if it decides on the presence of the signal. Then the sensor decision thresholds not only determine the probability of error at the fusion center but also determine the total energy consumption of the network. The decision thresholds minimizing the decision probability of error might require excessive amount of energy consumption. Then the designer may wish to trade a slight increase in the probability of error for a solution with less energy consumption. In [79], for a composite binary hypothesis testing problem, where under  $H_1$ , the source location is assumed to be a random variable uniformly distributed in a given region of interest, the detection problem has been solved by formulating a multi-objective optimization problem (MOP) with two conflicting objective functions, minimizing the probability of error at the fusion center  $P_e$  and minimizing the total network energy consumption  $E_T$  as,

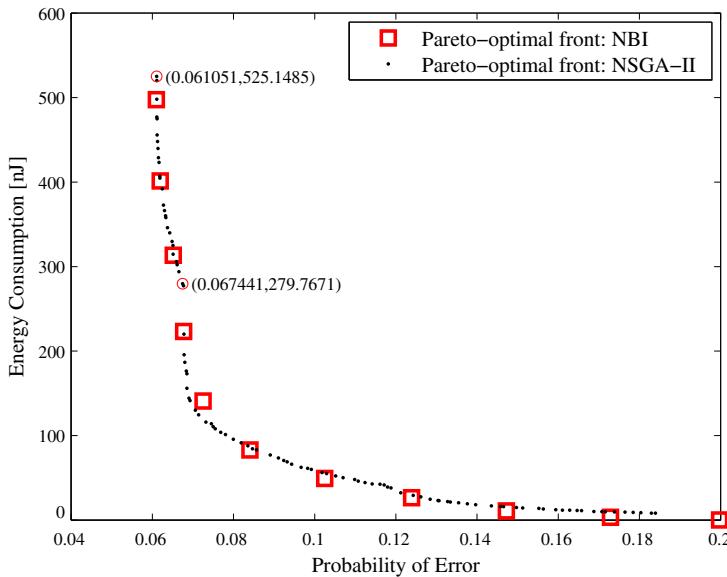
$$\min_{t_1, t_2, \dots, t_N} \{P_e(t_1, t_2, \dots, t_N), E_T(t_1, t_2, \dots, t_N)\} \quad (5.22)$$

$$t_{\min} \leq t_i \leq t_{\max}, \quad i \in \{1, 2, \dots, N\}.$$

The variables of the MOP  $\{t_1, t_2, \dots, t_N\}$  are the thresholds employed at the local sensors. As shown in Figure 5.4, with parallel network topology and  $N = 5$  sensors, the minimum  $P_e$  is achieved as 0.061 with energy consumption 525.56 nJ. By using the Pareto-optimal solutions for the MOP, one can accept the neighboring solution with error probability 0.067, and energy consumption 279.77 nJ. Therefore, a 10% increase in the probability of error, delivers around 88% saving in energy consumption. Moreover, in [80], a MOP has been formulated for the sensor placement problem where the objectives are maximizing the probability of detection and minimizing the total number of sensors.

### 3.05.2.5 Channel aware distributed detection

A distributed detection system can be designed in such a way that the channels between local sensors and the channels between sensors and the fusion center are considered error-free, by adopting a high transmission power and/or employing powerful error correction codes. In a wireless sensor network setting with severe constraints on energy, bandwidth and delay, such mechanisms may become prohibitive. Therefore, channel impairments should be taken into account in the design of distributed detection systems. The distributed detection problem in the presence of non-ideal communication channels has been studied

**FIGURE 5.4**

Pareto-optimal front between two objectives obtained with non-dominated sorting genetic algorithm-II (NSGA-II) [81] and normal boundary intersection (NBI) [82] ( $P(H_1) = 0.2$ ,  $N = 5$ ) [79].

quite extensively recently (see [83] and references therein). Under the Bayesian criterion, the optimality of the LRT for local sensor decisions has been established for a binary hypothesis testing problem in a sensor network with non-ideal channels [84]. For a finite number of sensors, Cheng et al. [85] provides the conditions under which the channel outputs no longer reduce the error probability at the fusion center. Channel aware decision fusion algorithms have been investigated with different degrees of channel state information for both single-hop [86–89] and multi-hop WSNs [90, 91], while channel-optimized local quantizer design methods are provided in [92, 93]. To counter sensor or channel failures, robust binary quantizer design has been proposed in [94]. Channel aware distributed detection has also been studied in the context of cooperative relay networks [95, 96].

### 3.05.3 Distributed detection with dependent observations

In distributed detection, the likelihood ratio tests at the local sensors are optimal if observations are conditionally independent given each hypothesis [27]. In general, it is reasonable to assume conditional independence across sensor nodes if the uncertainty comes mainly from device and ambient noise. However, for arbitrary sensor systems it does not necessarily hold. As an example, when sensors are in close proximity of one another, it is very likely that their observations are strongly correlated. If the observed signal is random in nature or the sensors are subject to common external noise, conditional

independence assumption may also fail. Without the conditional independence assumption, the joint density of the observations, given the hypothesis, cannot be written as the product of the marginal densities, as in (5.4). Then, the optimal tests at the sensors are no longer of the threshold type based solely on the likelihood ratio of the observations. Then, finding the optimal solution to the distributed detection problem becomes intractable [25].

Detection of known and unknown signals in correlated noise has been considered in [97]. For the case of two sensors observing a shift-in-mean of Gaussian data, the authors in [98] develop sufficient conditions for the optimality of each sensor implementing a local likelihood ratio test. In [99], the authors assume local likelihood ratio tests at multiple sensors. Then, the effect of correlated noise has been studied by considering the detection of a known signal in additive Gaussian and Laplacian noise where the authors show that system performance deteriorates when the correlation increases.

In [100], two correlation models are considered. In one, the correlation coefficient between any two sensors decreases geometrically as the sensor separation increases. In the other model, correlation coefficient between any two sensors is a constant. Asymptotic performance with Gaussian noise when the number of sensors goes to infinity is examined. In [101], distributed detection of known signals in correlated non-Gaussian noise is studied, where the noise is restricted to be circularly symmetric. Furthermore, the authors in [102] examine two-sensor distributed detection of known signals in correlated  $t$ -distributed noise. A distributed  $M$ -ary hypothesis testing problem when observations are correlated is examined from a numerical perspective in [103]. The authors in [104] discover that the nature of the local decision rules can be quite complicated for the simplest meaningful problem one can consider, i.e., the two detector case with dependent Gaussian observations.

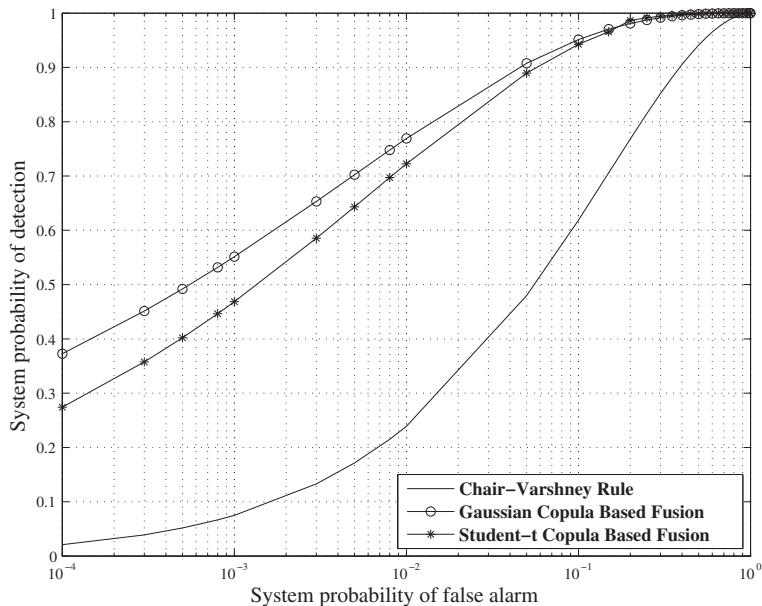
Constraining the local sensor decision rules to be suboptimal binary quantizers for the dependent observations problem, improvement in the global detection performance can still be attained by taking into account the correlation of local decisions while designing the fusion rule. Towards this end, design of fusion rules using correlated decisions has been proposed in [105, 106]. In [105], the authors have developed an optimum fusion rule based on the Neyman-Pearson criterion for correlated decisions assuming that the correlation coefficients between the sensor decisions are known and local sensor thresholds generating the correlated decisions are given. Using a special correlation structure, they studied the performance of the detection system versus the degree of correlation and showed how the performance advantage obtained by using a large number of sensors degrades as the degree of correlation between local decisions increases. In [106], the authors employed the Bahadur-Lazarsfeld series expansion of probability density functions to derive the optimum fusion rule for correlated local decisions.

In many applications, the dependence can get manifested in many different non-linear ways. As a result, more general descriptors of correlation than the Pearson correlation coefficient, which only characterizes linear dependence, may be required [107]. Moreover, the marginal distributions of sensor observations characterizing their univariate statistics may also not be identical. Here, emphasis should be laid on the fact that multivariate density (or mass) functions do not necessarily exist for arbitrary marginal density (or mass) functions. In other words, given arbitrary marginal distributions, their joint distribution function cannot be written in a straight-forward manner.

An interesting approach for the fusion of correlated decisions, that does not necessarily require prior information about the joint statistics of the sensor observations or decisions, is described next. Its novelty lies in the usage of *copula theory* [108]. The application of copula theory is widespread

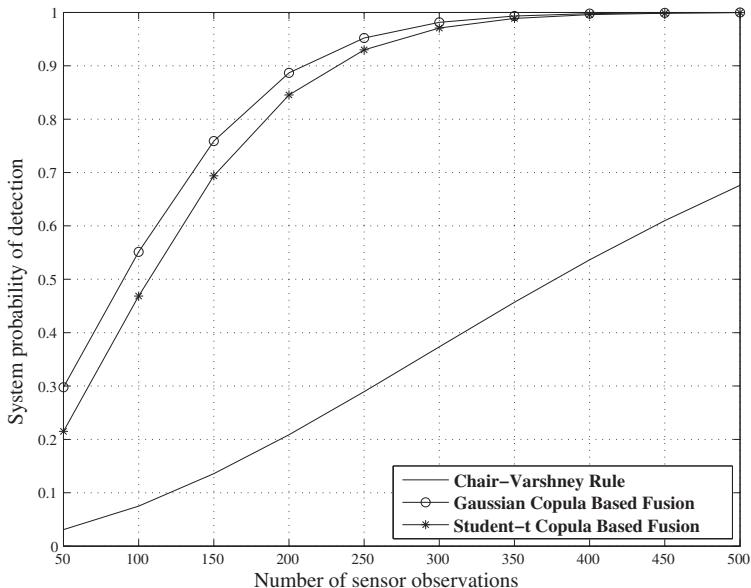
in the fields of econometrics and finance. However, its use for signal processing applications has been quite limited. The authors in [109, 110] employ copula theory for signal detection problems involving correlated observations as well as for heterogenous sensors observing a common scene. For the fusion of correlated decisions, copula theory does not require prior information about the joint statistics of the sensor observations or decisions and constructs the joint statistics based on a copula selection procedure. From Figure 5.5, it is clear that the performance of the copula-based fusion rules is superior to that of the Chair-Varshney fusion rule. Note that the copula function based fusion will fail to perform better than the Chair-Varshney rule if the constructed joint distribution using a particular parametric copula function does not adequately model the underlying joint distribution of the sensor observations. Therefore, training is necessary in order to select the best copula function. The topic of copula function selection for the distributed detection problem is considered in [111].

In Figure 5.6, the performance comparison of the copula based fusion rules and the Chair-Varshney fusion rule with varying number of sensor observations ( $N$ ) is shown with  $P_F = 0.001$ . The correlation parameters within the copula functions and the signal parameters under both hypotheses are maintained the same. Again, the performance of the proposed copula based fusion methods is better than the Chair-Varshney fusion rule, and they require fewer sensor observations than the Chair-Varshney rule to attain the same value of  $P_D$ .



**FIGURE 5.5**

Theoretical ROC curves comparing the Chair-Varshney fusion rule and the copula based fusion rules [111].

**FIGURE 5.6**

Detection performance comparison of the Chair-Varshney test and copula based tests with increase in the number of sensor observations [111].

### 3.05.4 Conclusion

In this chapter, distributed detection and decision fusion for a multi-sensor system have been discussed. In a conventional distributed detection framework, it is assumed that local sensors' performance indices are known and communication channels between the sensors and fusion center are perfect. Under these assumptions, the design for optimal decision fusion rule at the fusion center and the optimal local decision rules at sensors have been discussed. Under the conditional independence assumption, optimal decision rules at the local sensors and at the fusion center reduce to LRTs under Bayesian and Neyman-Pearson formulations. Using the PBPO procedure, the determination of the optimal LRT thresholds at the local sensors is still quite computationally complex, especially for a large scale sensor network. However, as the number of sensors goes to infinity an identical decision rule at all sensors become optimal. Further, the performance indices of local sensors were assumed to be unknown to the detection system. Non-parametric fusion rules have been discussed. Distributed detection system design under energy constraints and distributed detection system design using multi-objective optimization has also been presented.

Distributed detection with correlated observations remains a very difficult problem, since in such cases, the LRT test based solely on local observation is no longer an optimal test at sensors, and the optimal solution is intractable in general. Suboptimal solutions that can still exploit the dependence

information are required. For the dependent observations problem, a framework for fusion of correlated decisions using copula theory has been described. The local sensor decision rules are assumed to be based on simple binary quantization of the sensor observations. The described method is particularly useful when the marginal densities of sensor observations are non-Gaussian (and potentially nonidentical) and when dependence between sensor observations can get manifested in several different non-linear ways.

While many advances have been made in the area of distributed detection and decision fusion, many open and challenging problems remain that need further research. For example, distributed detection in a sensor network where the signal decays following a non-isotropic model, is a very difficult problem. Another challenging problem is the design of the optimal local sensor decision rules when observations from different sensors are dependent conditioned on either hypothesis. Copula theory has provided one approach to treat dependence, further work on copula-based methods and other approaches will be quite valuable.

---

## References

- [1] H.L. Van Trees, *Detection, Estimation, and Modulation Theory, Radar-Sonar Signal Processing and Gaussian Signals in Noise*, Wiley-Interscience, 2003.
- [2] J.G. Proakis, *Digital Communications*, McGraw-Hill, 2005.
- [3] M. Schwartz, W.R. Bennett, S. Stein, *Communication Systems and Techniques*, Wiley-IEEE Press, 1995.
- [4] Jinshan Tang, R.M. Rangayyan, Jun Xu, I. El Naqa, Yongyi Yang, Computer-aided detection and diagnosis of breast cancer with mammography: recent advances, *IEEE Trans. Inform. Technol. Biomed.* 13 (2) (2009) 236–251.
- [5] R. Peng, H. Chen, P.K. Varshney, Noise-enhanced detection of micro-calcifications in digital mammograms, *IEEE J. Sel. Top. Signal Process.* 3 (1) (2009) 62–73.
- [6] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, 1990.
- [7] R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification*, vol. 2, Wiley, New York, 2001.
- [8] S. Haykin, D.J. Thomson, J.H. Reed, Spectrum sensing for cognitive radio, *IEEE Proc.* 97 (5) (2009) 849–877.
- [9] T. Yucek, H. Arslan, A survey of spectrum sensing algorithms for cognitive radio applications, *IEEE Commun. Surv. Tutor.* 11 (1) (2009) 116–130.
- [10] I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, *IEEE Commun. Mag.* 40 (8) (2002) 102–114.
- [11] J.-F. Chamberland, V.V. Veeravalli, Wireless sensors in distributed detection applications, *IEEE Signal Process. Mag.* 24 (3) (2007) 16–25.
- [12] P.K. Varshney, *Distributed Detection and Data Fusion*, Springer, New York, 1997.
- [13] H.L. Van Trees, *Detection, Estimation and Modulation Theory*, vol. 1, Wiley, New York, 1968.
- [14] H.V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, 1988.
- [15] C.W. Helstrom, *Elements of Signal Detection and Estimation*, Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [16] R. Viswanathan, P.K. Varshney, Distributed detection with multiple sensors: Part I—Fundamentals, *Proc. IEEE* 85 (1) (1997) 54–63.
- [17] R.S. Blum, S.A. Kassam, H.V. Poor, Distributed detection with multiple sensors: Part II—Advanced topics, *Proc. IEEE* 85 (1) (1997).
- [18] R. Niu, P.K. Varshney, Performance analysis of distributed detection in a random sensor field, *IEEE Trans. Signal Process.* 56 (1) (2008) 339–349.
- [19] P. Ray, P.K. Varshney, False discovery rate based sensor decision rules for the network-wide distributed detection problem, *IEEE Trans. Aerosp. Electron. Syst.* 47 (3) (2011) 1785–1799.

- [20] J.P. Shaffer, Multiple hypothesis testing, *Annu. Rev. Psychol.* 46 (1) (1995) 561–584.
- [21] X. Zhu, Y. Yuan, C. Rorres, M. Kam, Distributed  $M$ -ary hypothesis testing with binary local decisions, *Inform. Fusion* 5 (3) (2004) 157–167.
- [22] Q. Zhang, P.K. Varshney, Decentralized  $M$ -ary detection via hierarchical binary decision fusion, *Inform. Fusion* 2 (1) (2001) 3–16.
- [23] C.W. Baum, V.V. Veeravalli, A sequential procedure for multihypothesis testing, *IEEE Trans. Inform. Theory* 40 (6) (1994).
- [24] R. Viswanathan, P.K. Varshney, Distributed detection with multiple sensors I—Fundamentals, *Proc. IEEE* 85 (1) (1997) 54–63.
- [25] J. Tsitsiklis, M. Athans, On the complexity of decentralized desicion making and detection problems, *IEEE Trans. Automat. Control* 30 (1985) 440–446.
- [26] N.S.V. Rao, Computational complexity issues in synthesis of simple distributed detection networks, *IEEE Trans. Syst. Man Cybern.* 21 (1991) 1071–1081.
- [27] J.N. Tsitsiklis, Decentralized detection, in: H.V. Poor, J.B Thomas (Eds.), *Advances in Statistical Signal Processing*, JAI Press, Greenwich, CT, 1993.
- [28] J.N. Tsitsiklis, On threshold rules in decentralized detection, in: *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens, Greece, 1986, pp. 232–236.
- [29] P. Willet, D. Warren, Decentralized detection: when are identical sensors identical, in: *Proceedings of the Conference on Information Science and Systems*, 1991, pp. 287–292.
- [30] M. Cherikh, P.B. Kantor, Counterexamples in distributed detection, *IEEE Trans. Inform. Theory* 38 (1992) 162–165.
- [31] Z.B. Tang, K.R. Pattipati, D. Kleinman, An algorithm for determining the detection thresholds in a distributed detection problem, *IEEE Trans. Syst. Man. Cybern.* 21 (1991) 231–237.
- [32] Z.B. Tang, Optimization of Detection Networks, Ph.D. Thesis, University of Connecticut, Storrs, CT, December 1990.
- [33] A.R. Reibman, Performance and Fault-Tolerance of Distributed Detection Networks, Ph.D. Thesis, Duke University, Durham, NC, 1987.
- [34] S.C.A. Thomopoulos, R. Viswanathan, D.K. Bougoulias, Optimal distributed decision fusion, *IEEE Trans. Aerosp. Electron. Syst.* 25 (1989) 761–765.
- [35] C.W. Helstrom, Gradient algorithms for quantization levels in distributed detection systems, *IEEE Trans. Aerosp. Electron. Syst.* 31 (1995) 390–398.
- [36] Z. Chair, P.K. Varshney, Optimal data fusion in multiple sensor detection systems, *IEEE Trans. Aerosp. Electron. Syst.* 22 (1986) 98–101.
- [37] R. Niu, P.K. Varshney, Joint detection and localization in sensor networks based on local decisions, in: *40th Asilomar Conference on Signals, Systems and Computers*, November 2006, pp. 525–529.
- [38] J.B. Thomas, Nonparametric detection, *Proc. IEEE* 58 (5) (1970) 623–631.
- [39] S. Kay, Can detectability be improved by adding noise? *IEEE Signal Proc. Lett.* 7 (1) (2000) 8–10.
- [40] H. Chen, P.K. Varshney, S.M. Kay, J.H. Michels, Theory of the stochastic resonance effect in signal detection: Part I—Fixed detectors, *IEEE Trans. Signal Process.* 55 (7) (2007) 3172–3184.
- [41] H. Chen, P.K. Varshney, Theory of the stochastic resonance effect in signal detection: Part II—Variable detectors, *IEEE Trans. Signal Process.* 56 (10) (2008) 5031–5041.
- [42] J.N. Tsitsiklis, Decentralized detection with a large number of sensors, *Math. Control Signals Syst.* 1 (1988) 167–182.
- [43] P. Chen, A. Papamarcou, New asymptotic results in parallel distributed detection, *IEEE Trans. Inform. Theory* 39 (6) (1993) 1847–1863.
- [44] J. Chamberland, V.V. Veeravalli, Decentralized detection in sensor networks, *IEEE Trans. Signal Process.* 51 (2003) 407–416.

- [45] J.F. Chamberland, V.V. Veeravalli, Asymptotic results for decentralized detection in power constrained wireless sensor networks, *IEEE J. Sel. Areas Commun.* 22 (6) (2004) 1007–1015.
- [46] S. Alhakeem, P.K. Varshney, A unified approach to the design of decentralized detection systems, *IEEE Trans. Aerosp. Electron. Syst.* 31 (1) (1995) 9–20.
- [47] P.F. Swaszek, On the performance of serial networks in distributed detection, *IEEE Trans. Aerosp. Electron. Syst.* 29 (1) (1993) 254–260.
- [48] Z.B. Tang, K.R. Pattipati, D.L. Kleinman, Optimization of detection networks. I. Tandem structures, *IEEE Trans. Syst. Man Cybern.* 21 (5) (1991) 1044–1059.
- [49] W.P. Tay, J.N. Tsitsiklis, M.Z. Win, On the subexponential decay of detection error probabilities in long tandems, *IEEE Trans. Inform. Theory* 54 (10) (2008) 4767–4771.
- [50] Z.-B. Tang, K.R. Pattipati, D.L. Kleinman, Optimization of detection networks. II. Tree structures, *IEEE Trans. Syst. Man Cybern.* 23 (1) (1993) 211–221.
- [51] W.P. Tay, J.N. Tsitsiklis, M.Z. Win, On the impact of node failures and unreliable communications in dense sensor networks, *IEEE Trans. Signal Process.* 56 (6) (2008) 2535–2546.
- [52] W.P. Tay, J.N. Tsitsiklis, M.Z. Win, Bayesian detection in bounded height tree networks, *IEEE Trans. Signal Process.* 57 (10) (2009) 4042–4051.
- [53] W.P. Tay, J.N. Tsitsiklis, M.Z. Win, Data fusion trees for detection: does architecture matter? *IEEE Trans. Inform. Theory* 54 (9) (2008) 4155–4168.
- [54] R. Srinivasan, Distributed detection with decision feedback (radar), *IEE Proc. Radar Signal Process.* 137 (6) (1990) 427–432.
- [55] S. Alhakeem, P.K. Varshney, Decentralized Bayesian detection with feedback, *IEEE Trans. Syst. Man Cybern.* A 26 (4) (1996) 503–513.
- [56] O.P. Kreidl, J.N. Tsitsiklis, S.I. Zoumpoulis, On decentralized detection with partial information sharing among sensors, *IEEE Trans. Signal Process.* 59 (4) (2011) 1759–1765.
- [57] T.M. Duman, M. Salehi, Decentralized detection over multiple-access channels, *IEEE Trans. Aerosp. Electron. Syst.* 34 (1998) 469–476.
- [58] Y. Sung, L. Tong, A. Swami, Asymptotic locally optimal detector for large-scale sensor networks under the poisson regime, *IEEE Trans. Signal Process.* 53 (6) (2005) 2005–2017.
- [59] G. Mergen, L. Tong, Type based estimation over multiple access channels, *IEEE Trans. Signal Process.* 54 (2) (2006) 613–626.
- [60] G. Mergen, V. Naware, L. Tong, Asymptotic detection performance of type-based multiple access over multiaccess fading channels, *IEEE Trans. Signal Process.* 55 (3) (2007) 1081–1092.
- [61] K. Liu, A.M. Sayeed, Type-based decentralized detection in wireless sensor networks, *IEEE Trans. Signal Process.* 55 (5) (2007) 1899–1910.
- [62] K. Liu, H. El Gamal, A.M. Sayeed, Decentralized inference over multiple-access channels, *IEEE Trans. Signal Process.* 55 (7) (2007) 3445–3455.
- [63] Q. Zou, S. Zheng, A.H. Sayed, Cooperative sensing via sequential detection, *IEEE Trans. Signal Process.* 58 (12) (2010) 6266–6283.
- [64] Q. Cheng, P.K. Varshney, K.G. Mehrotra, C.K. Mohan, Bandwidth management in distributed sequential detection, *IEEE Trans. Inform. Theory* 51 (8) (2005) 2954–2961.
- [65] H. Chen, P.K. Varshney, J.H. Michels, Improving sequential detection performance via stochastic resonance, *IEEE Signal Process. Lett.* 15 (2008) 685–688.
- [66] V.V. Veeravalli, Decentralized quickest change detection, *IEEE Trans. Inform. Theory* 47 (4) (2001) 1657–1665.
- [67] D. Bajovic, D. Jakovetic, J. Xavier, B. Sinopoli, J.M.F. Moura, Distributed detection via gaussian running consensus: large deviations asymptotic analysis, *IEEE Trans. Signal Process.* 59 (9) (2011) 4381–4396.

- [68] Z. Li, F.R. Yu, M. Huang, A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios, *IEEE Trans. Veh. Technol.* 59 (1) (2010) 383–393.
- [69] S.S. Stankovic, N. Ilic, M.S. Stankovic, K.H. Johansson, Distributed change detection based on a consensus algorithm, *IEEE Trans. Signal Process.* 59 (12) (2011) 5686–5697.
- [70] R. Niu, P.K. Varshney, Q. Cheng, Distributed detection in a large wireless sensor network, *Int. J. Inform. Fusion* 7 (4) (2006) 380–394.
- [71] R. Niu, P.K. Varshney, Distributed detection and fusion in a large wireless sensor network of random size, *EURASIP J. Wireless Commun. Network.* 5 (4) (2005) 462–472.
- [72] R. Niu, P.K. Varshney, Joint detection and localization in sensor networks based on local decisions, in: Fortieth Asilomar (Ed.), Conference on Signals, Systems and Computers, November 2006, pp. 525–529.
- [73] C. Rago, P.K. Willett, Y. Bar-Shalom, Censoring sensors: a low-communication-rate scheme for distributed detection, *IEEE Trans. Aerosp. Electron. Syst.* 32 (1996) 554–568.
- [74] S. Appadwedula, V.V. Veeravalli, D.L. Jones, Decentralized detection with censoring sensors, *IEEE Trans. Signal Process.* 56 (4) (2008) 1362–1373.
- [75] R. Jiang, Y. Lin, B. Chen, B. Suter, Distributed sensor censoring for detection in sensor networks under communication constraints, in: 39th Asilomar Conference on Signals, Systems and Computers, October 2005, pp. 946–950.
- [76] P. Addesso, S. Marano, V. Matta, Sequential sampling in sensor networks for detection with censoring nodes, *IEEE Trans. Signal Process.* 55 (11) (2007) 5497–5505.
- [77] Q. Cao, T. Abdelzaher, T. He, J. Stankovic, Towards optimal sleep scheduling in sensor networks for rare-event detection, in: Fourth International Symposium on Information Processing in Sensor Networks, IPSN, April 2005, pp. 20–27.
- [78] V.P. Sadaphal, B.N. Jain, Random and periodic sleep schedules for target detection in sensor networks, in: IEEE International Conference on Mobile Adhoc and Sensor Systems, MASS, October 2007, pp. 1–11.
- [79] E. Masazade, R. Rajagopalan, P.K. Varshney, C.K. Mohan, G.K. Sendur, M. Keskinoz, A multiobjective optimization approach to obtain decision thresholds for distributed detection in wireless sensor networks, *IEEE Trans. Syst. Man Cybern. B* 40 (2) (2010) pp. 444–457.
- [80] R. Rajagopalan, P.K. Varshney, C.K. Mohan, K. Mehrotra, Sensor placement for distributed detection of air pollutants: a constrained multi-objective optimization approach, in: Cognitive Systems with Interactive sensors, November 2007.
- [81] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evolut. Comput.* 6 (2) (2002) 182–197.
- [82] I. Das, J. Dennis, Normal-boundary interaction: a new method for generating the pareto surface in nonlinear multicriteria optimization problems, *SIAM J. Optim.* 8 (1998) 631–657.
- [83] B. Chen, L. Tong, P.K. Varshney, Channel aware distributed detection in wireless sensor networks, *IEEE Signal Process. Mag.* 23 (4) (2006) 16–26 (special issue on Distributed Signal Processing for Sensor Networks).
- [84] B. Chen, P.K. Willett, On the optimality of likelihood ratio test for local sensor decision rules in the presence of non-ideal channels, *IEEE Trans. Inform. Theory* 51 (2) (2005) 693–699.
- [85] Q. Cheng, B. Chen, P.K. Varshney, Detection performance limits for distributed sensor networks in the presence of nonideal channels, *IEEE Trans. Wireless Commun.* 5 (11) (2006) 3034–3038.
- [86] B. Chen, R. Jiang, T. Kasetkasem, P.K. Varshney, Channel aware decision fusion for wireless sensor networks, *IEEE Trans. Signal Process.* 52 (2004) 3454–3458.
- [87] R. Niu, B. Chen, P.K. Varshney, Fusion of decisions transmitted over Rayleigh fading channels in wireless sensor networks, *IEEE Trans. Signal Process.* 54 (3) (2006) 1018–1027.

- [88] R. Jiang, B. Chen, Fusion of censored decisions in wireless sensor networks, *IEEE Trans. Wirel. Commun.* 4 (2005) 2668–2673.
- [89] I. Bahcecı, G. Al-Regib, Y. Altunbasak, Parallel distributed detection for wireless sensor networks: performance analysis and design, in: *IEEE Global Telecommunications Conference, GLOBECOM '05*, vol. 4, December 2005, pp. 2420–2424.
- [90] Y. Lin, B. Chen, P.K. Varshney, Decision fusion rules in multi-hop wireless sensor networks, *IEEE Trans. Aerosp. Electron. Syst.* 51 (2005) 475–488.
- [91] I. Bahcecı, G. Al-Regib, Y. Altunbasak, Serial distributed detection for wireless sensor networks, in: *International Symposium on Information Theory, ISIT, 2005*, pp. 830–834.
- [92] B. Liu, B. Chen, Channel optimized quantizers for decentralized detection in wireless sensor networks, *IEEE Trans. Inform. Theory* 52 (2006) 3349–3358.
- [93] B. Liu, B. Chen, Decentralized detection in wireless sensor networks with channel fading statistics, *EURASIP J. Wireless Commun. Network* 2007 (2007) 11–11.
- [94] Y. Lin, B. Chen, B. Suter, Robust binary quantizers for distributed detection, *IEEE Trans. Wireless Commun.* 6 (6) (2007) 2172–2181.
- [95] B. Liu, B. Chen, R.S. Blum, Minimum error probability cooperative relay design, *IEEE Trans. Signal Process.* 55 (2) (2007) 656–664.
- [96] H. Chen, P.K. Varshney, B. Chen, Cooperative relay for decentralized detection, in: *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, Nevada, March 2008, pp. 2293–2296.
- [97] G.S. Lauer, N.R. Sandell Jr., Distributed detection with waveform observations: correlated observation processes, in: *Proceedings of the 1982 American Controls Conference*, vol. 2, 1982, pp. 812–819.
- [98] P. Chen, A. Papamarcou, Likelihood ratio partitions for distributed signal detection in correlated Gaussian noise, in: *Proceedings of the IEEE International Symposium on Information Theory*, October 1996, p. 118.
- [99] V. Aalo, R. Viswanathan, On distributed detection with correlated sensors: two examples, *IEEE Trans. Aerosp. Electron. Syst.* 25 (1989) 414–421.
- [100] V. Aalo, R. Viswanathan, Asymptotic performance of a distributed detection system in correlated Gaussian noise, *IEEE Trans. Signal Process.* 40 (1992) 211–213.
- [101] R. Blum, P. Willett, P. Swaszek, Distributed detection of known signals in nonGaussian noise which is dependent from sensor to sensor, in: *Proceedings of the Conference on Information Science and Systems*, March 1997, pp. 825–830.
- [102] X. Lin, R. Blum, Numerical solutions for optimal distributed detection of known signals in dependent  $t$ -distributed noise: the two-sensor problem, in: *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, November 1998, pp. 613–617.
- [103] Z. Tang, K. Pattipati, D. Kleinman, A distributed  $M$ -ary hypothesis testing problem with correlated observations, *IEEE Trans. Automat. Control* 37 (1992) 1042–1046.
- [104] P.K. Willett, P.F. Swaszek, R.S. Blum, The good, bad, and ugly: distributed detection of a known signal in dependent Gaussian noise, *IEEE Trans. Signal Process.* 48 (2000) 3266–3279.
- [105] E. Drakopoulos, C.-C. Lee, Optimum multisensor fusion of correlated local decisions, *IEEE Trans. Aerosp. Electron. Syst.* 27 (4) (1991) 593–606.
- [106] M. Kam, Q. Zhu, W.S. Gray, Optimal data fusion of correlated local decisions in multiple sensor detection systems, *IEEE Trans. Aerosp. Electron. Syst.* 28 (3) (1992) 916–920.
- [107] D.D. Mari, S. Kotz, *Correlation and Dependence*, Imperial College Press, 2001.
- [108] R.B. Nelsen, *An Introduction to Copulas*, Springer-Verlag, New York, 1999.
- [109] A. Sundaresan, P.K. Varshney, N.S.V. Rao, Copula-based fusion of correlated decisions, *IEEE Trans. Aerosp. Electron. Syst.* 47 (1) (2011) 454–471.

- [110] S.G. Iyengar, P.K. Varshney, T. Damarla, A parametric copula-based framework for hypothesis testing using heterogeneous data, *IEEE Trans. Signal Process.* 59 (5) (2011) 2308–2319.
- [111] A. Sundaresan, Detection and Source Location Estimation of Random Signal Sources Using Sensor Networks, Ph.D. Thesis, Syracuse University, 2010.

# Quickest Change Detection

# 6

Venugopal V. Veeravalli and Taposh Banerjee

ECE Department and Coordinated Science Laboratory, Urbana, IL, USA

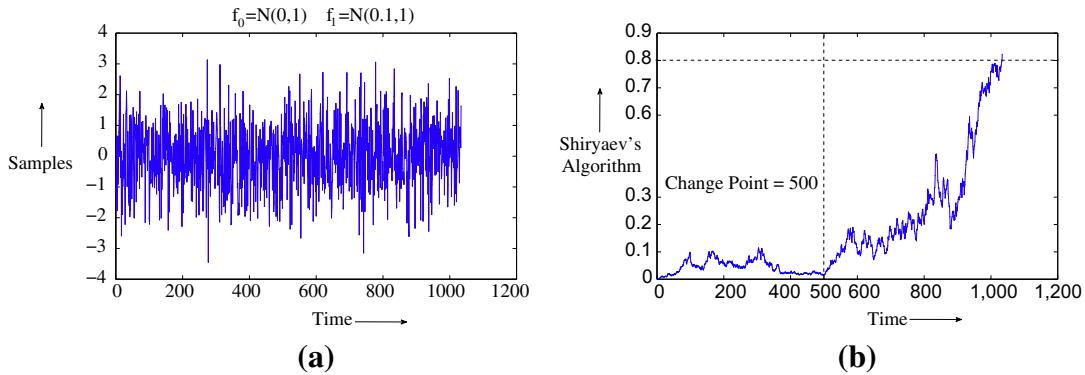
## 3.06.1 Introduction

The problem of quickest change detection comprises three entities: a stochastic process under observation, a change point at which the statistical properties of the process undergo a change, and a decision maker that observes the stochastic process and aims to detect this change in the statistical properties of the process. A *false alarm* event happens when the change is declared by the decision maker before the change actually occurs. The general objective of the theory of quickest change detection is to design algorithms that can be used to detect the change as soon as possible, subject to false alarm constraints.

The quickest change detection problem has a wide range of important applications, including biomedical signal and image processing, quality control engineering, financial markets, link failure detection in communication networks, intrusion detection in computer networks and security systems, chemical or biological warfare agent detection systems (as a protection tool against terrorist attacks), detection of the onset of an epidemic, failure detection in manufacturing systems and large machines, target detection in surveillance systems, econometrics, seismology, navigation, speech segmentation, and the analysis of historical texts. See Section 3.06.7 for a more detailed discussion of the applications and related references.

To motivate the need for quickest change detection algorithms, in Figure 6.1a we plot a sample path of a stochastic sequence whose samples are distributed as  $\mathcal{N}(0, 1)$  before the change, and distributed as  $\mathcal{N}(0.1, 1)$  after the change. For illustration, we choose time slot 500 as the change point. As is evident from the figure, the change cannot be detected through manual inspection. In Figure 6.1b, we plot the evolution of the *Shiryayev statistic* (discussed in detail in Section 3.06.3), computed using the samples of Figure 6.1a. As seen in Figure 6.1b, the value of the Shiryayev statistic stays close to zero before the change point, and grows up to one after the change point. The change is detected by using a threshold of 0.8.

We also see from Figure 6.1b that it takes around 500 samples to detect the change after it occurs. Can we do better than that, at least on an average? Clearly, declaring change before the change point (time slot 500) will result in zero delay, but it will cause a false alarm. The theory of quickest change detection deals with finding algorithms that have provable optimality properties, in the sense of minimizing the average detection delay under a false alarm constraint. We will show later that the Shiryayev algorithm, employed in Figure 6.1b, is optimal for a certain *Bayesian* model.

**FIGURE 6.1**

Detecting a change in the mean of a Gaussian random sequence. (a) Stochastic sequence with samples from  $f_0 \sim \mathcal{N}(0, 1)$  before the change (time slot 500), and with samples from  $f_1 \sim \mathcal{N}(0.1, 1)$  after the change. (b) Evolution of the classical Shiryaev algorithm when applied to the samples given on the left. We see that the change is detected approximately at time slot 1000.

Earliest results on quickest change detection date back to the work of Shewhart [1,2] and Page [3] in the context of statistical process/quality control. Here the *state* of the system is monitored by taking a sequence of measurements, and an alarm has to be raised if the measurements indicate a fault in the process under observation or if the state is *out of control*. Shewhart proposed the use of a control chart to detect a change, in which the measurements taken over time are plotted on a chart and an alarm is raised the first time the measurements fall outside some pre-specified control limits. In the Shewhart control chart procedure, the statistic computed at any given time is a function of only the measurements at that time, and not of the measurements taken in the past. This simplifies the algorithm but may result in a loss in performance (unacceptable delays when detecting small changes). In [3], Page proposed that instead of ignoring the past observations, a weighted sum (moving average chart) or a cumulative sum (CuSum) of the past statistics (likelihood ratios) can be used in the control chart to detect the change more efficiently. It is to be noted that the motivation in the work of Shewhart and Page was to design easily implementable schemes with good performance, rather than to design schemes that could be theoretically proven to be optimal with respect to a suitably chosen performance criterion.

Initial theoretical formulations of the quickest change detection problem were for an observation model in which, conditioned on the change point, the observations are independent and identically distributed (i.i.d.) with some known distribution before the change point, and i.i.d. with some other known distribution after the change point. This observation model will be referred to as the *i.i.d. case* or *i.i.d. model* in this article.

The i.i.d. model was studied by Shiryaev [4,5], under the further assumption that the change point is a random variable with a known geometric distribution. Shiryaev obtained an optimal algorithm that minimizes the *average detection delay* over all stopping times that meet a given constraint on the *probability of false alarm*. We refer to Shiryaev's formulation as the *Bayesian* formulation; details are provided in Section 3.06.3.

When the change point is modeled as non-random but unknown, the probability of false alarm is not well defined and therefore false alarms are quantified through the *mean time to false alarm* when the system is operating under the pre-change state, or through its reciprocal, which is called the *false alarm rate*. Furthermore, it is generally not possible to obtain an algorithm that is uniformly efficient over all possible values of the change point, and therefore a *minimax* approach is required. The first minimax theory is due to Lorden [6] in which he proposed a measure of detection delay obtained by taking the supremum (over all possible change points) of a worst-case delay over all possible realizations of the observations, conditioned on the change point. Lorden showed that the CuSum algorithm of [3] is asymptotically optimal according to his minimax criterion for delay, as the mean time to false alarm goes to infinity (false alarm rate goes to zero). This result was improved upon by Moustakides [7] who showed that the CuSum algorithm is exactly optimal under Lorden's criterion. An alternative proof of the optimality of the CuSum procedure is provided in [8]. See Section 3.06.4 for details.

Pollak [9] suggested modifying Lorden's minimax criterion by replacing the double maximization of Lorden by a single maximization over all possible change points of the detection delay conditioned on the change point. He showed that an algorithm called the Shiryaev-Roberts algorithm, one that is obtained by taking a limit on Shiryaev's Bayesian solution as the geometric parameter of the change point goes to zero, is asymptotically optimal as the false alarm rate goes to zero. It was later shown in [10] that even the CuSum algorithm is asymptotically optimum under the Pollak's criterion, as the false alarm rate goes to zero. Recently a family of algorithms based on the Shiryaev-Roberts statistic was shown to have strong optimality properties as the false alarm rate goes to zero. See [11] and Section 3.06.4 for details.

For the case where the pre- and post-change observations are not independent conditioned on the change point, the quickest change detection problem was studied in the minimax setting by Lai [10] and in the Bayesian setting by Tartakovsky and Veeravalli [12]. In both of these works, an asymptotic lower bound on the delay is obtained for any stopping rule that meets a given false alarm constraint (on false alarm rate in [10] and on the probability of false alarm in [12]), and an algorithm is proposed that meets the lower bound on the detection delay asymptotically. Details are given in Sections 3.06.3.2 and 3.06.4.3.

To summarize, in Sections 3.06.3 and 3.06.4, we discuss the Bayesian and Minimax versions of the quickest change detection problem, where the change has to be detected in a single sequence of random variables, and where the pre- and post-change distributions are given. In Section 3.06.6, we discuss variants and generalizations of the classical quickest change detection problem, for which significant progress has been made. We consider the cases where the pre- or post-change distributions are not completely specified (Section 3.06.6.1), where there is an additional constraint on the cost of observations used in the detection process (Section 3.06.6.2), and where the change has to be detected using multiple geographically distributed sensor nodes (Section 3.06.6.3). In Section 3.06.7 we provide a brief overview of the applications of quickest change detection. We conclude in Section 3.06.8 with a discussion of other possible extensions and future research directions.

For a more detailed treatment of some of the topics discussed in this chapter, we refer the reader to the books by Poor and Hadjiliadis [13] and Chow et al. [14], and the upcoming book by Tartakovsky et al. [15]. We will restrict our attention in this chapter to detecting changes in discrete-time stochastic systems; the continuous time setting is discussed in [13].

In Table 6.1, a glossary of important symbols used in this chapter is provided.

**Table 6.1** Glossary

Symbol	Definition/interpretation
$o(1)$	$x = o(1)$ as $c \rightarrow c_0$ , if $\forall \epsilon > 0, \exists \delta > 0$ s.t., $ x  \leq \epsilon$ if $ c - c_0  < \delta$
$O(1)$	$x = O(1)$ as $c \rightarrow c_0$ , if $\exists \epsilon > 0, \delta > 0$ s.t., $ x  \leq \epsilon$ if $ c - c_0  < \delta$
$g(c) \sim h(c)$ as $c \rightarrow c_0$	$\lim_{c \rightarrow c_0} \frac{g(c)}{h(c)} = 1$ or $g(c) = h(c)(1 + o(1))$ as $c \rightarrow c_0$
$\{X_k\}$	Observation sequence
Stopping time $\tau$ on $\{X_k\}$	$\mathbb{I}_{\{\tau=n\}} = 0$ or 1 depends only on the values of $X_1, \dots, X_n$
Change point $\Gamma, \gamma$	Time index at which distribution of observations changes from $f_0$ to $f_1$
$\mathbb{P}_n (\mathbb{E}_n)$	Probability measure (expectation) when the change occurs at time $n$
$\mathbb{P}_\infty (\mathbb{E}_\infty)$	Probability measure (expectation) when the change does not occur
ess sup $X$	Essential supremum of $X$ , i.e., smallest $K$ such that $\mathbb{P}(X \leq K) = 1$
$D(f_1 \  f_0)$	K-L Divergence between $f_1$ and $f_0$ , defined as $\mathbb{E}_1 \left( \log \frac{f_1(X)}{f_0(X)} \right)$
$(x)^+$	$\max\{x, 0\}$
ADD( $\tau$ )	$\text{ADD}(\tau) = \sum_{n=0}^{\infty} \mathbb{P}(\Gamma = n) \mathbb{E}_n [(\tau - n)^+]$
PFA( $\tau$ )	$\text{PFA}(\tau) = \mathbb{P}(\tau < \Gamma) = \sum_{n=0}^{\infty} \mathbb{P}(\Gamma = n) \mathbb{P}_n(\tau < n)$
FAR( $\tau$ )	$\text{FAR}(\tau) = \frac{1}{\mathbb{E}_\infty[\tau]}$
WADD( $\tau$ )	$\text{WADD}(\tau) = \sup_{n \geq 1} \text{ess sup } \mathbb{E}_n [(\tau - n)^+   X_1, \dots, X_{n-1}]$
CADD( $\tau$ )	$\text{CADD}(\tau) = \sup_{n \geq 1} \mathbb{E}_n [\tau - n   \tau \geq n]$

## 3.06.2 Mathematical preliminaries

A typical observation process will be denoted by sequence  $\{X_n, n = 1, 2, \dots\}$ . Before we describe the quickest change detection problem, we present some useful definitions and results that summarize the required mathematical background. For a detailed treatment of the topics discussed below we recommend [14, 16–18].

### 3.06.2.1 Martingales

**Definition 1.** The random sequence  $\{X_n, n = 1, 2, \dots\}$  is called a *martingale* if  $\mathbb{E}[X_n]$  is finite for all  $n$ , and for any  $k_1 < k_2 < \dots < k_n < k_{n+1}$ ,

$$\mathbb{E}[X_{k_n+1} | X_{k_1}, \dots, X_{k_n}] = X_{k_n}. \quad (6.1)$$

If the “=” in (6.1) is replaced by “ $\leq$ ,” then the sequence  $\{X_n\}$  is called a *supermartingale*, and if the “=” is replaced by “ $\geq$ ,” the sequence is called a *submartingale*. A martingale is both a supermartingale and a submartingale.

Some important and useful results regarding martingales are as follows:

**Theorem 1 (Kolmogorov's Inequality [14]).** *Let  $\{X_n, n = 1, 2, \dots\}$  be a submartingale. Then*

$$\mathbb{P}\left(\max_{1 \leq k \leq n} X_k \geq \gamma\right) \leq \frac{\mathbb{E}[X_n^+]}{\gamma}, \quad \forall \gamma > 0,$$

where  $X_n^+ = \max\{0, X_n\}$ .

Kolmogorov's inequality can be considered to be a generalization of Markov's inequality, which is given by

$$\mathbb{P}(X \geq \gamma) \leq \frac{\mathbb{E}[X^+]}{\gamma}, \quad \forall \gamma > 0. \quad (6.2)$$

As we will see in the following sections, quickest change detection procedures often involve comparing a stochastic sequence to a threshold to make decisions. Martingale inequalities often play a crucial role in the design of the threshold so that the procedure meets a false alarm constraint. We now state one of the most useful results regarding martingales.

**Theorem 2 (Martingale Convergence Theorem [16]).** *Let  $\{X_n, n = 1, 2, \dots\}$  be a martingale (or submartingale or supermartingale), such that  $\sup_n \mathbb{E}[|X_n|] < \infty$ . Then, with probability one, the limit  $X_\infty = \lim_{k \rightarrow \infty} X_n$  exists and is finite.*

### 3.06.2.2 Stopping times

**Definition 2.** A *stopping time* with respect to the random sequence  $\{X_n, n = 1, 2, \dots\}$  is a random variable  $\tau$  with the property that for each  $n$ , the event  $\{\tau = n\} \in \sigma(X_1, \dots, X_n)$ , where  $\sigma(X_1, \dots, X_n)$  denotes the sigma-algebra generated by  $(X_1, \dots, X_n)$ . Equivalently, the random variable  $\mathbb{I}_{\{\tau=n\}}$ , which is the indicator of the event  $\{\tau = n\}$ , is a function of only  $X_1, \dots, X_n$ .

Sometimes the definition of a stopping time  $\tau$  also requires that  $\tau$  be finite almost surely, i.e., that  $\mathbb{P}(\tau < \infty) = 1$ .

Stopping times are essential to sequential decision making procedures such as quickest change detection procedures, since the times at which decisions are made are stopping times with respect to the observation sequence. There are two main results concerning stopping times that are of interest.

**Theorem 3 (Doob's Optional Stopping Theorem [14]).** *Let  $\{X_n, n = 1, 2, \dots\}$  be a martingale, and let  $\tau$  be a stopping time with respect to  $\{X_n, n = 1, 2, \dots\}$ . If the following conditions hold:*

1.  $\mathbb{P}(\tau < \infty) = 1$ ,
2.  $\mathbb{E}[|X_\tau|] < \infty$ ,
3.  $\mathbb{E}[X_n \mathbb{I}_{\{\tau > n\}}] \rightarrow 0$  as  $n \rightarrow \infty$ ,

then

$$\mathbb{E}[X_\tau] = \mathbb{E}[X_1].$$

Similarly, if the above conditions hold, and if  $\{X_n, n = 1, 2, \dots\}$  is a submartingale, then

$$\mathbb{E}[X_\tau] \geq \mathbb{E}[X_1],$$

and if  $\{X_n, n = 1, 2, \dots\}$  is a supermartingale, then

$$\mathbb{E}[X_\tau] \leq \mathbb{E}[X_1].$$

**Theorem 4 (Wald's Identity [17]).** Let  $\{X_n, n = 1, 2, \dots\}$  be a sequence of independent and identically distributed (i.i.d.) random variables, and let  $\tau$  be a stopping time with respect to  $\{X_n, n = 1, 2, \dots\}$ . Furthermore, define the sum at time  $n$  as

$$S_n = \sum_{k=1}^n X_k.$$

Then, if  $\mathbb{E}[|X_1|] < \infty$  and  $\mathbb{E}[\tau] < \infty$ ,

$$\mathbb{E}[S_\tau] = \mathbb{E}[X_1]\mathbb{E}[\tau].$$

Like martingale inequalities, the optional stopping theorem is useful in the false alarm analysis of quickest change detection procedures. Both the optional stopping theorem and Wald's identity also play a key role in the delay analysis of quickest change detection procedures.

### 3.06.2.3 Renewal and nonlinear renewal theory

As we will see in subsequent sections, quickest change detection procedures often involve comparing a stochastic sequence to a threshold to make decisions. Often the stochastic sequence used in decision-making can be expressed as a sum of a random walk and possibly a *slowly changing* perturbation. To obtain accurate estimates of the performance of the detection procedure, one needs to obtain an accurate estimate of the distribution of the overshoot of the stochastic sequence when it crosses the decision threshold. Under suitable assumptions, and when the decision threshold is large enough, the overshoot distribution of the stochastic sequence can be approximated by the overshoot distribution of the random walk. It is then of interest to have asymptotic estimates of the overshoot distribution, when a random walk crosses a large boundary.

Consider a sequence of i.i.d. random variables  $\{Y_n\}$  (with  $Y$  denoting a generic random variable in the sequence) and let

$$S_n = \sum_{k=1}^n Y_k$$

and

$$\tau = \inf\{n \geq 1 : S_n > b\}.$$

The quantity of interest is the distribution of the overshoot  $S_\tau - b$ . If  $\{Y_n\}$  are i.i.d. *positive* random variables with cumulative distribution function (c.d.f.)  $F(y)$ , then  $\{Y_n\}$  can be viewed as inter-arrival times of buses at a stop. The overshoot is then the time to next bus when an observer is waiting for a bus at time  $b$ . The distribution of the overshoot, and hence also of the time to next bus, as  $b \rightarrow \infty$  is a well known result in renewal theory.

**Theorem 5 [17].** If  $\{Y_n\}$  are nonarithmetic<sup>1</sup> random variables, and  $\mathbb{P}(Y > 0) = 1$ , then

$$\lim_{b \rightarrow \infty} \mathbb{P}(S_\tau - b > y) = (\mathbb{E}[Y])^{-1} \int_y^\infty \mathbb{P}\{Y > x\} dx.$$

Further, if  $\mathbb{E}[Y^2] < \infty$ , then

$$\lim_{b \rightarrow \infty} \mathbb{E}(S_\tau - b) = \frac{\mathbb{E}[Y^2]}{2\mathbb{E}[Y]}.$$

When the  $\{Y_n\}$  are i.i.d. but not necessarily non-negative, and  $\mathbb{E}[Y] > 0$ , then the following concept of *ladder variables* can be used. Let

$$\tau_+ = \inf\{n \geq 1 : S_n > 0\}.$$

Note that if  $\tau_+ < \infty$ , then  $S_{\tau_+}$  is a positive random variable. Also, if

$$\tau = \inf\{n \geq 1 : S_n > b\} < \infty$$

then the distribution of  $S_\tau - b$  is the same as the overshoot distribution for the sum of a sequence of i.i.d. positive random variables (each with distribution equal to that of  $S_{\tau_+}$ ) crossing the boundary  $b$ . Therefore, by applying Theorem 5, we have the following result.

**Theorem 6 [17].** If  $\{Y_n\}$  are nonarithmetic, then

$$\lim_{b \rightarrow \infty} \mathbb{P}(S_\tau - b > y) = (\mathbb{E}[S_{\tau_+}])^{-1} \int_y^\infty \mathbb{P}(S_{\tau_+} > x) dx.$$

Further, if  $\mathbb{E}[Y^2] < \infty$ , then

$$\lim_{b \rightarrow \infty} \mathbb{E}(S_\tau - b) = \frac{\mathbb{E}[S_{\tau_+}^2]}{2\mathbb{E}[S_{\tau_+}]}.$$

Techniques for computing the required quantities involving the distribution of the ladder height  $S_{\tau_+}$  in Theorem 6 can be found in [17].

As mentioned earlier, often the stochastic sequence considered in quickest change detection problem can be written as a sum of a random walk and a sequence of small perturbations. Let

$$Z_n = \sum_{k=1}^n Y_k + \eta_n$$

and

$$\tau = \inf\{n \geq 1 : Z_n > b\}.$$

Then,

$$Z_\tau = \sum_{k=1}^\tau Y_k + \eta_\tau.$$

---

<sup>1</sup>A random variable is arithmetic if all of its probability mass is on a lattice. Otherwise it is said to be non-arithmetic.

Therefore, assuming that  $\mathbb{E}[\tau] < \infty$ , Wald's Identity (see Theorem 4) implies that

$$\mathbb{E}[Z_\tau] = \mathbb{E}\left[\sum_{k=1}^{\tau} Y_k\right] + \mathbb{E}[\eta_\tau], \quad (6.3)$$

$$= \mathbb{E}[\tau]\mathbb{E}[Y] + \mathbb{E}[\eta_\tau]. \quad (6.4)$$

Thus,

$$\begin{aligned}\mathbb{E}[\tau] &= \frac{\mathbb{E}[Z_\tau] - \mathbb{E}[\eta_\tau]}{\mathbb{E}[Y]} \\ &= \frac{b + \mathbb{E}[Z_\tau - b] - \mathbb{E}[\eta_\tau]}{\mathbb{E}[Y]}.\end{aligned}$$

If  $\mathbb{E}[\eta_\tau]$  and  $\mathbb{E}[Z_\tau - b]$  are finite then it is easy to see that

$$\mathbb{E}[\tau] \sim \frac{b}{\mathbb{E}[Y]} \text{ as } b \rightarrow \infty,$$

where  $\sim$  is as defined in Table 6.1.

But if we can characterize the overshoot distribution of  $\{Z_n\}$  when it crosses a large threshold then we can obtain better approximations for  $\mathbb{E}[\tau]$ . Nonlinear renewal theory allows us to obtain distribution of the overshoot when  $\{\eta_n\}$  satisfies some properties.

**Definition 3.**  $\{\eta_n\}$  is a *slowly changing* sequence if

$$n^{-1} \max\{|\eta_1|, \dots, |\eta_n|\} \xrightarrow[i.p.]{n \rightarrow \infty} 0, \quad (6.5)$$

and for every  $\epsilon > 0$ , there exists  $n^*$  and  $\delta > 0$  such that for all  $n \geq n^*$

$$\mathbb{P}\left(\max_{1 \leq k \leq n\delta} |\eta_{n+k} - \eta_n| > \epsilon\right) < \epsilon. \quad (6.6)$$

If indeed  $\{\eta_n\}$  is a slowly changing sequence, then the distribution of  $Z_\tau - b$ , as  $b \rightarrow \infty$ , is equal to the asymptotic distribution of the overshoot when the random walk  $S_n = \sum_{k=1}^n Y_k$  crosses a large positive boundary, as stated in the following result.

**Theorem 7 [17].** *If  $\{Y_n\}$  are nonarithmetic and  $\{\eta_n\}$  is a slowly changing sequence then*

$$\lim_{b \rightarrow \infty} \mathbb{P}(Z_\tau - b \leq x) = \lim_{b \rightarrow \infty} \mathbb{P}(S_\tau - b \leq x).$$

*Further, if  $\text{Var}(Y) < \infty$  and certain additional conditions ((9.22)–(9.27) in [17]) are satisfied, then*

$$\mathbb{E}[\tau] = \frac{b + \zeta - \mathbb{E}[\eta]}{\mathbb{E}[Y]} + o(1) \text{ as } b \rightarrow \infty,$$

*where  $\zeta = \frac{\mathbb{E}[S_{\tau+}^2]}{2\mathbb{E}[S_{\tau+}]}$ , and  $\eta$  is the limit of  $\{\eta_n\}$  in distribution.*

### 3.06.3 Bayesian quickest change detection

As mentioned earlier we will primarily focus on the case where the observation process  $\{X_n\}$  is a discrete time stochastic process, with  $X_n$  taking real values, whose distribution changes at some unknown change point. In the Bayesian setting it is assumed that the change point is a random variable  $\Gamma$  taking values on the non-negative integers, with  $\pi_n = \mathbb{P}\{\Gamma = n\}$ . Let  $\mathbb{P}_n$  (correspondingly  $\mathbb{E}_n$ ) be the probability measure (correspondingly expectation) when the change occurs at time  $\tau = n$ . Then,  $\mathbb{P}_\infty$  and  $\mathbb{E}_\infty$  stand for the probability measure and expectation when  $\tau = \infty$ , i.e., the change does not occur. At each time step a decision is made based on all the information available as to whether to stop and declare a change or to continue taking observations. Thus the time at which the change is declared is a stopping time  $\tau$  on the sequence  $\{X_n\}$  (see Section 3.06.2.2). Define the average detection delay (ADD) and the probability of false alarm (PFA), as

$$\text{ADD}(\tau) = \mathbb{E}[(\tau - \Gamma)^+] = \sum_{n=0}^{\infty} \pi_n \mathbb{E}_n[(\tau - n)^+], \quad (6.7)$$

$$\text{PFA}(\tau) = \mathbb{P}(\tau < \Gamma) = \sum_{n=0}^{\infty} \pi_n \mathbb{P}_n(\tau < n). \quad (6.8)$$

Then, the Bayesian quickest change detection problem is to minimize ADD subject to a constraint on PFA. Define the class of stopping times that satisfy a constraint  $\alpha$  on PFA:

$$\mathcal{C}_\alpha = \{\tau : \text{PFA}(\tau) \leq \alpha\}. \quad (6.9)$$

Then the Bayesian quickest change detection problem as formulated by Shiryaev is as follows.

*Shiryaev's Problem:* For a given  $\alpha$ , find a stopping time  $\tau \in \mathcal{C}_\alpha$  to minimize  $\text{ADD}(\tau)$ . (6.10)

Under an i.i.d. model for the observations, and a geometric model for the change point  $\Gamma$ , (6.10) can be solved exactly by relating it to a stochastic control problem [4, 5]. We discuss this i.i.d. model in detail in Section 3.06.3.1. When the model is not i.i.d., it is difficult to find algorithms that are exactly optimal. However, asymptotically optimal solutions, as  $\alpha \rightarrow 0$ , are available in a very general non-i.i.d. setting [12], as discussed in Section 3.06.3.2.

#### 3.06.3.1 The Bayesian i.i.d. setting

Here it is assumed that conditioned on the change point  $\Gamma$ , the random variables  $\{X_n\}$  are i.i.d. with probability density function (p.d.f.)  $f_0$  before the change point, and i.i.d. with p.d.f.  $f_1$  after the change point. The change point  $\Gamma$  is modeled as *geometric* with parameter  $\rho$ , i.e., for  $0 < \rho < 1$

$$\pi_n = \mathbb{P}\{\Gamma = n\} = \rho(1 - \rho)^{n-1} \mathbb{I}_{\{n \geq 1\}}, \quad \pi_0 = 0 \quad (6.11)$$

where  $\mathbb{I}$  is the indicator function. The goal is to choose a stopping time  $\tau$  on the observation sequence  $\{X_n\}$  to solve (6.10).

A solution to (6.10) is provided in Theorem 8 below. Let  $X_1^n = (X_1, \dots, X_n)$  denote the observations up to time  $n$ . Also let

$$p_n = \mathbb{P}(\Gamma \leq n | X_1^n) \quad (6.12)$$

be the *a posteriori* probability at time  $n$  that the change has taken place given the observation up to time  $n$ . Using Bayes' rule,  $p_n$  can be shown to satisfy the recursion

$$p_{n+1} = \Phi(X_{n+1}, p_n), \quad (6.13)$$

where

$$\Phi(X_{n+1}, p_n) = \frac{\tilde{p}_n L(X_{n+1})}{\tilde{p}_n L(X_{n+1}) + (1 - \tilde{p}_n)}, \quad (6.14)$$

$\tilde{p}_n = p_n + (1 - p_n)\rho$ ,  $L(X_{n+1}) = f_1(X_{n+1})/f_0(X_{n+1})$  is the likelihood ratio, and  $p_0 = 0$ .

**Definition 4 (Kullback-Leibler (K-L) Divergence).** The K-L divergence between two p.d.f.'s  $f_1$  and  $f_0$  is defined as

$$D(f_1 \| f_0) = \int f_1(x) \log \frac{f_1(x)}{f_0(x)} dx.$$

Note that  $D(f_1 \| f_0) \geq 0$  with equality iff  $f_1 = f_0$  almost surely. We will assume that

$$0 < D(f_1 \| f_0) < \infty.$$

**Theorem 8 [4,5].** *The optimal solution to Bayesian optimization problem of (6.10) is the Shiryaev algorithm/test, which is described by the stopping time:*

$$\tau_S = \inf \{n \geq 1 : p_n \geq A_\alpha\} \quad (6.15)$$

if  $A_\alpha \in (0, 1)$  can be chosen such that

$$\text{PFA}(\tau_S) = \alpha. \quad (6.16)$$

**Proof.** Towards solving (6.10), we consider a Lagrangian relaxation of this problem that can be solved using dynamic programming.

$$J^* = \min_{\tau} (\mathbb{E}[(\tau - \Gamma)^+] + \lambda_f \mathbb{P}(\tau < \Gamma)), \quad (6.17)$$

where  $\lambda_f$  is the Lagrange multiplier,  $\lambda_f \geq 0$ . It is shown in [4,5] that under the assumption (6.16), there exists a  $\lambda_f$  such that the solution to (6.17) is also the solution to (6.10).

Let  $\Theta_n$  denote the state of the system at time  $n$ . After the stopping time  $\tau$  it is assumed that the system enters a terminal state  $\mathcal{T}$  and stays there. For  $n < \tau$ , we have  $\Theta_n = 0$  for  $n < \Gamma$ , and  $\Theta_n = 1$  otherwise. Then we can write

$$\text{ADD}(\tau) = \mathbb{E} \left[ \sum_{n=0}^{\tau-1} \mathbb{I}_{\{\Theta_n=1\}} \right] \quad \text{and} \quad \text{PFA}(\tau) = \mathbb{E} [\mathbb{I}_{\{\Theta_\tau=0\}}].$$

Furthermore, let  $D_n$  denote the stopping decision variable at time  $n$ , i.e.,  $D_n = 0$  if  $k < \tau$  and  $D_n = 1$  otherwise. Then the optimization problem in (6.17) can be written as a minimization of an additive cost over time:

$$J^* = \min_{\tau} \mathbb{E} \left[ \sum_{n=0}^{\tau} g_n(\Theta_n, D_n) \right]$$

with

$$g_n(\theta, d) = \mathbb{I}_{\{\theta \neq T\}} \left[ \mathbb{I}_{\{\theta=1\}} \mathbb{I}_{\{d=0\}} + \lambda_f \mathbb{I}_{\{\theta=0\}} \mathbb{I}_{\{d=1\}} \right].$$

Using standard arguments [19] it can be seen that this optimization problem can be solved using infinite horizon dynamic programming with sufficient statistic (belief state) given by:

$$\mathbb{P}(\Theta_n = 1 | X_1^n) = \mathbb{P}(\Gamma \leq n | X_1^n) = p_n,$$

which is the *a posteriori probability* of (6.12).

The optimal policy for the problem given in (6.17) can be obtained from the solution to the Bellman equation:

$$J(p_n) = \min_{d_n} \lambda_f (1 - p_n) \mathbb{I}_{\{d_n=1\}} + \mathbb{I}_{\{d_n=0\}} [p_n + A_J(p_n)], \quad (6.18)$$

where

$$A_J(p_n) = \mathbb{E}[J(\Phi(X_{n+1}, p_n))].$$

It can be shown by using an induction argument that both  $J$  and  $A_J$  are non-negative concave functions on the interval  $[0, 1]$ , and that  $J(1) = A_J(1) = 0$ . Then, it is easy to show that the optimal solution for the problem in (6.17) has the following structure:

$$\tau_S = \inf \{k \geq 1 : p_n \geq A\}.$$

■

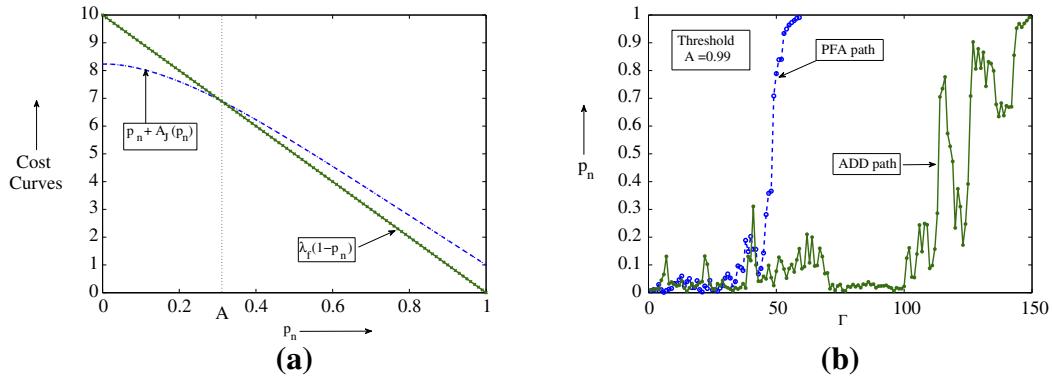
See Figure 6.2a for a plot of  $\lambda_f(1 - p_n)$  and  $p_n + A_J(p_n)$  as a function of  $p_n$ . Figure 6.2b shows a typical evolution of the optimal Shiryaev algorithm.

We now discuss some alternative descriptions of the Shiryaev algorithm. Let

$$\Lambda_n = \frac{p_n}{(1 - p_n)}$$

and

$$R_{n,\rho} = \frac{p_n}{(1 - p_n)\rho}.$$

**FIGURE 6.2**

Cost curves and typical evolution of the Shiryaev algorithm. (a) A plot of the cost curves for  $\lambda_f = 10$ ,  $\rho = 0.01$ ,  $f_0 \sim \mathcal{N}(0,1)$ ,  $f_1 \sim \mathcal{N}(0.75,1)$ . (b) Typical evolution of the Shiryaev algorithm. Threshold  $A = 0.99$  and change point  $\Gamma = 100$ .

We note that  $\Lambda_n$  is the likelihood ratio of the hypotheses “ $H_1 : \Gamma \leq n$ ” and “ $H_0 : \Gamma > n$ ” averaged over the change point:

$$\begin{aligned}\Lambda_n &= \frac{p_n}{(1-p_n)} \\ &= \frac{\mathbb{P}(\Gamma \leq n | X_1^n)}{\mathbb{P}(\Gamma > n | X_1^n)} \\ &= \frac{\sum_{k=1}^n (1-\rho)^{k-1} \rho \prod_{i=1}^{k-1} f_0(X_i) \prod_{i=k}^n f_1(X_i)}{(1-\rho)^n \prod_{i=1}^n f_0(X_i)} \\ &= \frac{1}{(1-\rho)^n} \sum_{k=1}^n (1-\rho)^{k-1} \rho \prod_{i=k}^n L(X_i),\end{aligned}\tag{6.19}$$

where as defined before  $L(X_i) = \frac{f_1(X_i)}{f_0(X_i)}$ . Also,  $R_{n,\rho}$  is a scaled version of  $\Lambda_n$ :

$$R_{n,\rho} = \frac{1}{(1-\rho)^n} \sum_{k=1}^n (1-\rho)^{k-1} \prod_{i=k}^n L(X_i).\tag{6.20}$$

Like  $p_n$ ,  $R_{n,\rho}$  can also be computed using a recursion:

$$R_{n+1,\rho} = \frac{1 + R_{n,\rho}}{1 - \rho} L(X_{n+1}), \quad R_{0,\rho} = 0.$$

It is easy to see that  $\Lambda_n$  and  $R_{n,\rho}$  have one-to-one mappings with the Shiryaev statistic  $p_n$ .

**Algorithm 1 (Shiryaev Algorithm).** The following three stopping times are equivalent and define the Shiryaev stopping time

$$\tau_S = \inf \{n \geq 1 : p_n \geq A\}, \quad (6.21)$$

$$\tau_S = \inf \{n \geq 1 : \Lambda_n \geq a\}, \quad (6.22)$$

$$\tau_S = \inf \left\{ n \geq 1 : R_{n,\rho} \geq \frac{a}{\rho} \right\} \quad (6.23)$$

with  $a = \frac{A}{1-A}$ .

We will later see that defining the Shiryaev algorithm using the statistic  $\Lambda_n$  (6.19) will be useful in Section 3.06.3.2, where we discuss the Bayesian quickest change detection problem in a non-i.i.d. setting. Also, defining the Shiryaev algorithm using the statistic  $R_{n,\rho}$  (6.20) will be useful in Section 3.06.4 where we discuss quickest change detection in a minimax setting.

### 3.06.3.2 General asymptotic Bayesian theory

As mentioned earlier, when the observations are not i.i.d. conditioned on the change point  $\Gamma$ , then finding an exact solution to the problem (6.10) is difficult. Fortunately, a Bayesian *asymptotic* theory can be developed for quite general pre- and post-change distributions [12]. In this section we discuss the results from [12] and provide a glimpse of the proofs.

We first state the observation model studied in [12]. When the process evolves in the pre-change regime, the conditional density of  $X_n$  given  $X_1^{n-1}$  is  $f_{0,n}(X_n | X_1^{n-1})$ . After the change happens, the conditional density of  $X_n$  given  $X_1^{n-1}$  is given by  $f_{1,n}(X_n | X_1^{n-1})$ .

As in the i.i.d. case, we can define the *a posteriori* probability of change having taken place before time  $n$ , given the observation up to time  $n$ , i.e.,

$$p_n = \mathbb{P}(\Gamma \leq n | X_1^n) \quad (6.24)$$

with the understanding that the recursion (6.13) is no longer valid, except for the i.i.d. model.

We note that in the non-i.i.d. case also  $\Lambda_n = \frac{p_n}{1-p_n}$  is the likelihood ratio of the hypotheses “ $H_1 : \Gamma \leq n$ ” and “ $H_0 : \Gamma > n$ .” If  $\pi_n = \mathbb{P}\{\Gamma = n\}$ , then following (6.19),  $\Lambda_n$  can be written for a general change point distribution  $\{\pi_n\}$  as

$$\begin{aligned} \Lambda_n &= \frac{p_n}{(1-p_n)} \\ &= \frac{\mathbb{P}(\Gamma \leq n | X_1^n)}{\mathbb{P}(\Gamma > n | X_1^n)} \\ &= \frac{\sum_{k=1}^n \pi_n \prod_{i=1}^{k-1} f_{0,i}(X_i | X_1^{i-1}) \prod_{i=k}^n f_{1,i}(X_i | X_1^{i-1})}{\mathbb{P}(\Gamma > n) \prod_{i=1}^n f_{0,i}(X_i | X_1^{i-1})} \\ &= \frac{1}{\mathbb{P}(\Gamma > n)} \sum_{k=1}^n \pi_n \prod_{i=k}^n \exp(Y_i), \end{aligned}$$

where

$$Y_i = \log \frac{f_{1,i}(X_i | X_1^{i-1})}{f_{0,i}(X_i | X_1^{i-1})}.$$

If  $\Gamma$  is geometrically distributed with parameter  $\rho$ , the above expression reduces to

$$\Lambda_n = \frac{1}{(1-\rho)^n} \sum_{k=1}^n (1-\rho)^{k-1} \rho \prod_{i=k}^n \exp(Y_i).$$

In fact,  $\Lambda_n$  can even be computed recursively in this case:

$$\Lambda_{n+1} = \frac{1 + \Lambda_n}{1 - \rho} \exp(Y_{n+1}) \quad (6.25)$$

with  $\Lambda_0 = 0$ .

In [12], it is shown that if there exists  $q$  such that

$$\frac{1}{t} \sum_{i=n}^{n+t} Y_i \rightarrow q \quad \text{a.s. } \mathbb{P}_n \quad \text{when } t \rightarrow \infty \forall n \quad (6.26)$$

( $q = D(f_1 \| f_0)$  for the i.i.d. model), and some additional conditions on the rates of convergence are satisfied, then the Shiryaev algorithm (6.21) is asymptotically optimal for the Bayesian optimization problem of (6.10) as  $\alpha \rightarrow 0$ . In fact,  $\tau_S$  minimizes all moments of the detection delay as well as the moments of the delay, conditioned on the change point. The asymptotic optimality proof is based on first finding a lower bound on the asymptotic moment of the delay of all the detection procedures in the class  $\mathcal{C}_\alpha$ , as  $\alpha \rightarrow 0$ , and then showing that the Shiryaev stopping time (6.21) achieves that lower bound asymptotically.

To state the theorem, we need the following definitions. Let  $q$  be the limit as specified in (6.26), and let  $0 < \epsilon < 1$ . Then define

$$T_\epsilon^{(n)} = \sup \left\{ t \geq 1 : \left| \frac{1}{t} \sum_{i=n}^{n+t} Y_i - q \right| > \epsilon \right\}.$$

Thus,  $T_\epsilon^{(n)}$  is the last time that the log likelihood sum  $\sum_{i=n}^{n+t} Y_i$  falls outside an interval of length  $\epsilon$  around  $q$ . In general, existence of the limit  $q$  in (6.26) only guarantees  $\mathbb{P}_n(T_\epsilon^{(n)} < \infty) = 1$ , and not the finiteness of the moments of  $T_\epsilon^{(n)}$ . Such conditions are needed for existence of moments of detection delay of  $\tau_S$ . In particular, for some  $r \geq 1$ , we need:

$$\mathbb{E}_n[T_\epsilon^{(n)}]^r < \infty \text{ for all } \epsilon > 0 \text{ and } n \geq 1, \quad (6.27)$$

and

$$\sum_{n=1}^{\infty} \pi_n \mathbb{E}_n[T_\epsilon^{(n)}]^r < \infty \text{ for all } \epsilon > 0. \quad (6.28)$$

Now, define

$$d = - \lim_{n \rightarrow \infty} \frac{\log \mathbb{P}(\Gamma > n)}{n}.$$

The parameter  $d$  captures the tail parameter of the distribution of  $\Gamma$ . If  $\Gamma$  is “heavy tailed” then  $d = 0$ , and if  $\Gamma$  has an “exponential tail” then  $d > 0$ . For example, for the geometric prior with parameter  $\rho$ ,  $d = |\log(1 - \rho)|$ .

**Theorem 9 [12].** *If the likelihood ratios are such that (6.26) is satisfied then*

1. *If  $a = a_\alpha = \frac{1-\alpha}{\alpha}$ , then  $\tau_S$  as defined in (6.21) belongs to the set  $\mathcal{C}_\alpha$ .*
2. *For all  $n \geq 1$ , the  $m$ th moment of the conditional delay, conditioned on  $\Gamma = n$ , satisfies:*

$$\inf_{\tau \in \mathcal{C}_\alpha} \mathbb{E}_n[(\tau - n)^+]^m \geq \left( \frac{|\log \alpha|}{q + d} \right)^m (1 + o(1)) \text{ as } \alpha \rightarrow 0. \quad (6.29)$$

3. *For all  $n \geq 1$ , if (6.27) is satisfied then for all  $m \leq r$ ,*

$$\begin{aligned} \mathbb{E}_n[(\tau_S - n)^+]^m &\leq \left( \frac{\log a}{q + d} \right)^m (1 + o(1)) \text{ as } a \rightarrow \infty \\ &= \left( \frac{|\log \alpha|}{q + d} \right)^m (1 + o(1)) \text{ as } \alpha \rightarrow 0 \text{ if } a = a_\alpha = \frac{1 - \alpha}{\alpha}. \end{aligned} \quad (6.30)$$

4. *The  $m$ th (unconditional) moment of the delay satisfies*

$$\inf_{\tau \in \mathcal{C}_\alpha} \mathbb{E}[(\tau - \Gamma)^+]^m \geq \left( \frac{|\log \alpha|}{q + d} \right)^m (1 + o(1)) \text{ as } \alpha \rightarrow 0. \quad (6.31)$$

5. *If (6.27) and (6.28) are satisfied, then for all  $m \leq r$*

$$\begin{aligned} \mathbb{E}[(\tau_S - \Gamma)^+]^m &\leq \left( \frac{\log a}{q + d} \right)^m (1 + o(1)) \text{ as } a \rightarrow \infty \\ &= \left( \frac{|\log \alpha|}{q + d} \right)^m (1 + o(1)) \text{ as } \alpha \rightarrow 0 \text{ if } a = a_\alpha = \frac{1 - \alpha}{\alpha}. \end{aligned} \quad (6.32)$$

**Proof.** We provide sketches of the proofs for part (1), (2), and (3). The proofs of (4) and (5) follow by averaging the results in (2) and (3) over the prior on the change point.

1. Note that

$$\text{PFA}(\tau_S) = \mathbb{E}[1 - p_{\tau_S}] \leq 1 - A_\alpha.$$

Thus,  $A_\alpha = 1 - \alpha$  would ensure  $\text{PFA}(\tau_S) \leq \alpha$ . Since,  $a_\alpha = \frac{A_\alpha}{1 - A_\alpha}$ , we have the result.

2. Let  $L_\alpha$  be a positive number. By Chebyshev inequality,

$$\mathbb{P}_n((\tau - n)^m > L_\alpha^m) \leq \frac{\mathbb{E}_n[(\tau - n)^+]^m}{L_\alpha^m}.$$

This gives a lower bound on the detection delay

$$\begin{aligned}\mathbb{E}_n[(\tau - n)^+]^m &\geq L_\alpha^m \mathbb{P}_n((\tau - n)^m > L_\alpha^m) \\ &= L_\alpha^m \mathbb{P}_n(\tau - n > L_\alpha).\end{aligned}$$

Minimizing over the family  $C_\alpha$ , we get

$$\inf_{\tau \in C_\alpha} \mathbb{E}_n[(\tau - n)^+]^m \geq L_\alpha^m \left[ \inf_{\tau \in C_\alpha} \mathbb{P}_n(\tau - n > L_\alpha) \right].$$

Thus, if

$$\inf_{\tau \in C_\alpha} \mathbb{P}_n(\tau - n > L_\alpha) \rightarrow 1 \text{ as } \alpha \rightarrow 0 \quad (6.33)$$

then  $L_\alpha^m$  is a lower bound for the detection delay of the family  $C_\alpha$ . It is shown in [12] that if (6.26) is satisfied then (6.33) is true for  $L_\alpha = (1 - \epsilon) \frac{|\log \alpha|}{q + d}$  for all  $\epsilon > 0$ .

3. We only summarize the technique used to obtain the upper bound. Let  $\{S_n\}$  be any stochastic process such that

$$\frac{S_n}{n} \rightarrow q \text{ as } n \rightarrow \infty.$$

Let

$$\eta = \inf\{n \geq 1 : S_n > b\}$$

and for  $\epsilon > 0$

$$T_\epsilon = \sup \left\{ n \geq 1 : \left| \frac{S_n}{n} - b \right| > \epsilon \right\}.$$

First note that  $S_{\eta-1} < b < S_\eta$ . Also, on the set  $\{k \geq T_\epsilon\}$ ,  $|S_n/n - q| < \epsilon$  for all  $n \geq k$ . The event  $\{|S_n/n - q| < \epsilon\}$  implies  $n \leq \frac{S_n}{q - \epsilon}$ . Using these observations we have

$$\begin{aligned}\eta - 1 &= (\eta - 1)\mathbb{I}_{\{\eta-1 > T_\epsilon\}} + (\eta - 1)\mathbb{I}_{\{\eta-1 \leq T_\epsilon\}} \\ &\leq (\eta - 1)\mathbb{I}_{\{\eta-1 > T_\epsilon\}} + T_\epsilon \\ &\leq \frac{b}{q - \epsilon} + T_\epsilon.\end{aligned}$$

If  $\mathbb{E}[T_\epsilon] < \infty$ , and because  $\epsilon$  was chosen arbitrarily, we have

$$\mathbb{E}[\eta] \leq \frac{b}{q} (1 + o(1)) \text{ as } b \rightarrow \infty.$$

This also motivates the need for conditions on finiteness of higher order moments of  $T_\epsilon$  to obtain upper bound on the moments of the detection delay. ■

From the above theorem, the following corollary easily follows.

**Corollary 1 [12].** *If the likelihood ratios are such that (6.26–6.28) are satisfied for some  $r \geq 1$ , then for the Shiryaev stopping time  $\tau_S$  defined in (6.21)*

$$\inf_{\tau \in C_\alpha} \mathbb{E}[(\tau - \Gamma)^+]^m \sim \mathbb{E}[(\tau_S - \Gamma)^+]^m \sim \left( \frac{|\log \alpha|}{q + d} \right)^m \text{ as } \alpha \rightarrow 0, \text{ for all } m \leq r. \quad (6.34)$$

A similar result can be concluded for the conditional moments as well.

### 3.06.3.3 Performance analysis for i.i.d. model with geometric prior

We now present the second order asymptotic analysis of the Shiryaev algorithm for the i.i.d. model, provided in [12] using the tools from nonlinear renewal theory introduced in Section 3.06.2.3.

When the observations  $\{X_n\}$  are i.i.d. conditioned on the change point, condition (6.26) is satisfied and

$$\frac{1}{t} \sum_{i=n}^{k+t} Y_i \rightarrow D(f_1 \| f_0) \quad \text{a.s. } \mathbb{P}_n \quad \text{when } t \rightarrow \infty \forall n,$$

where  $D(f_1 \| f_0)$  is the K-L divergence between the densities  $f_1$  and  $f_0$  (see Definition 4). From Thereom 9, it follows that for  $a_\alpha = \frac{1-\alpha}{\alpha}$ ,

$$\text{PFA}(\tau_S) \leq \alpha.$$

Also, it is shown in [12] that if

$$0 < D(f_1 \| f_0) < \infty \quad \text{and} \quad 0 < D(f_0 \| f_1) < \infty,$$

then conditions (6.27) and (6.28) are satisfied, and hence from Corollary 1,

$$\inf_{\tau \in C_\alpha} \mathbb{E}[(\tau - \Gamma)^+]^m \sim \mathbb{E}[(\tau_S - \Gamma)^+]^m \sim \left( \frac{|\log \alpha|}{D(f_1 \| f_0) + |\log(1-\rho)|} \right)^m \quad \text{as } \alpha \rightarrow 0. \quad (6.35)$$

Note that the above statement provides the asymptotic delay performance of the Shiryaev algorithm. However, the bound for PFA can be quite loose and the first order expression for the ADD (6.35) may not provide good estimate for ADD if the PFA is not very small. In that case it is useful to have a second order estimate based on nonlinear renewal theory, as obtained in [12].

First note that (6.25) for the i.i.d. model will reduce to

$$\Lambda_{n+1} = \frac{1 + \Delta_n}{1 - \rho} L(X_{n+1}). \quad (6.36)$$

with  $\Lambda_0 = 0$ . Now, let  $Z_n = \log \Lambda_n$ , and recall that  $Y_k = \log \frac{f_1(X_k)}{f_0(X_k)}$ . Then, it can be shown that

$$\begin{aligned} Z_n &= \sum_{k=1}^n [Y_k + |\log(1-\rho)|] + \log(e^{Z_0} + \rho) + \sum_{k=1}^{n-1} \log(1 + e^{-Z_k} \rho) \\ &= \sum_{k=1}^n Y_k + n|\log(1-\rho)| + \eta_n. \end{aligned} \quad (6.37)$$

Therefore the Shiryaev algorithm can be equivalently written as

$$\tau_S = \inf \{n \geq 1 : Z_n \geq b\}.$$

We now show how the asymptotic overshoot distribution plays a key role in second order asymptotic analysis of PFA and ADD. Since,  $p_{\tau_S} \geq A$  implies that  $Z_{\tau_S} \geq \log \frac{A}{1-A} = \log a = b$ , we have,

$$\frac{1}{1 + e^{-Z_{\tau_S}}} \geq \frac{a}{1 + a}.$$

$$\begin{aligned}
\text{PFA}(\tau_S) &= \mathbb{E}[1 - p_{\tau_S}] = \mathbb{E}\left[\frac{1}{1 + e^{Z_{\tau_S}}}\right] \leq \mathbb{E}\left[e^{-Z_{\tau_S}}\right] \\
&= \mathbb{E}\left[\frac{1}{e^{Z_{\tau_S}}} \frac{1}{1 + e^{-Z_{\tau_S}}}\right] \geq \mathbb{E}\left[\frac{1}{e^{Z_{\tau_S}}} \frac{a}{1 + a}\right] \\
&= \mathbb{E}\left[e^{-Z_{\tau_S}}\right](1 + o(1)) \text{ as } a \rightarrow \infty.
\end{aligned}$$

Thus,

$$\begin{aligned}
\text{PFA}(\tau_S) &= \mathbb{E}[e^{-Z_{\tau_S}}](1 + o(1)) = e^{-b}\mathbb{E}[e^{-(Z_{\tau_S}-b)}](1 + o(1)) \text{ as } b \rightarrow \infty \\
&= e^{-b}\mathbb{E}[e^{-(Z_{\tau_S}-b)}|\tau \geq \Gamma](1 + o(1)) \text{ as } b \rightarrow \infty,
\end{aligned}$$

and we see that PFA is a function of the overshoot when  $Z_n$  crosses  $a$  from below.

Similarly,

$$Z_{\tau_S} = \sum_{k=1}^{\tau_S} Y_k + \tau_S |\log(1-\rho)| + \eta_{\tau_S}.$$

Following the developments in Section 3.06.2.3, if the sequence  $Y_n$  satisfies some additional conditions, then we can write<sup>2</sup>:

$$\begin{aligned}
\mathbb{E}_1[\tau] &= \frac{\mathbb{E}_1[Z_\tau] - \mathbb{E}_1[\eta_\tau]}{|\log(1-\rho)| + D(f_1\|f_0)} \\
&= \frac{b + \mathbb{E}_1[Z_\tau - b] - \mathbb{E}_1[\eta_\tau]}{|\log(1-\rho)| + D(f_1\|f_0)}.
\end{aligned}$$

It is shown in [12] that  $\eta_n$  is a slowly changing sequence, and hence the distribution of  $Z_{\tau_S} - b$ , as  $b \rightarrow \infty$ , is equal to the asymptotic distribution of the overshoot when the random walk  $\sum_{k=1}^n Y_k + n|\log(1-\rho)|$  crosses a large positive boundary. We define the following quantities: the asymptotic overshoot distribution of a random walk

$$R(x) = \lim_{b \rightarrow \infty} \mathbb{P}\left(\sum_{k=1}^{\tau_S} Y_k + \tau_S |\log(1-\rho)| - b \leq x\right) \quad (6.38)$$

its mean

$$\kappa = \int_0^\infty x dR(x) \quad (6.39)$$

and its Laplace transform at 1

$$\zeta = \int_0^\infty e^{-x} dR(x). \quad (6.40)$$

Also, the sequence  $\{Y_n\}$  satisfies some additional conditions and hence the following results are true.

---

<sup>2</sup>As explained in [12], this analysis is facilitated if we restrict to the worst-case detection delay, which is obtained by conditioning on the event that the change happens at time 1.

**Table 6.2**  $f_0 \sim \mathcal{N}(0,1)$ ,  $f_1 \sim \mathcal{N}(1,1)$ ,  $\rho = 0.01$ 

<b><math>b</math></b>	<b>PFA</b>		<b>ADD</b>	
	<b>Simulations</b>	<b>Analysis Theorem 10</b>	<b>Simulations</b>	<b>Analysis Theorem 10</b>
1.386	$1.22 \times 10^{-1}$	$1.39 \times 10^{-1}$	6.93	10.31
2.197	$5.85 \times 10^{-2}$	$6.19 \times 10^{-2}$	8.87	11.9
4.595	$5.61 \times 10^{-3}$	$5.63 \times 10^{-3}$	13.9	16.6
6.906	$5.59 \times 10^{-4}$	$5.58 \times 10^{-4}$	18.59	21.13
11.512	$5.6 \times 10^{-6}$	$5.58 \times 10^{-6}$	27.64	30.16

**Theorem 10 [12].** If  $\{Y_n\}_n$  are nonarithmetic then  $\eta_n$  is a slowly changing sequence. Then by Theorem 7

$$\lim_{b \rightarrow \infty} \mathbb{P}(Z_{\tau_S} - b \leq x) = R(x).$$

This implies

$$\text{PFA}(\tau_S) \sim \zeta e^{-b} \quad \text{as } b \rightarrow \infty.$$

If in addition  $\mathbb{E}[Y^2] < \infty$  then

$$\mathbb{E}_1[\tau_S] = \frac{b + \kappa - \mathbb{E}_1[\eta]}{|\log(1-\rho)| + D(f_1 \| f_0)} + o(1) \text{ as } b \rightarrow \infty,$$

where  $\eta$  is the a.s. limit of the sequence  $\{\eta_n\}$ .

In Table 6.2, we compare the asymptotic expressions for PFA and ADD given in Theorem 10 with simulations. As can be seen in the table, the asymptotic expressions get more accurate as PFA approaches 0.

In Figure 6.3 we plot the ADD as a function of  $\log(\text{PFA})$  for Gaussian observations. For a PFA constraint of  $\alpha$  that is small,  $b \approx |\log(\alpha)|$ , and

$$\text{ADD} \approx \mathbb{E}_1[\tau_S] \approx \frac{|\log(\alpha)|}{|\log(1-\rho)| + D(f_1 \| f_0)}$$

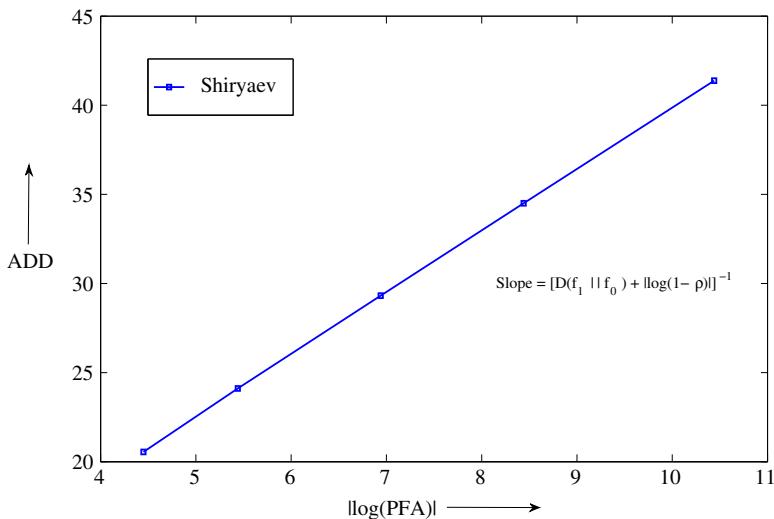
giving a slope of  $\frac{1}{|\log(1-\rho)| + D(f_1 \| f_0)}$  to the trade-off curve.

When  $|\log(1-\rho)| \ll D(f_1 \| f_0)$ , the observations contain more information about the change than the prior, and the tradeoff slope is roughly  $\frac{1}{D(f_1 \| f_0)}$ . On the other hand, when  $|\log(1-\rho)| \gg D(f_1 \| f_0)$ , the prior contains more information about the change than the observations, and the tradeoff slope is roughly  $\frac{1}{|\log(1-\rho)|}$ . The latter asymptotic slope is achieved by the stopping time that is based only on the prior:

$$\tau = \inf\{n \geq 1 : \mathbb{P}(\Gamma > n) \leq \alpha\}.$$

This is also easy to see from (6.14). With  $D(f_1 \| f_0)$  small,  $L(X) \approx 1$ , and the recursion for  $p_k$  reduces to

$$p_{n+1} = p_n + (1-p_n)\rho, \quad p_0 = 0.$$

**FIGURE 6.3**

ADD-PFA trade-off curve for the Shiryaev algorithm:  $\rho = 0.01$ ,  $f_0 = \mathcal{N}(0,1)$ ,  $f_1 = \mathcal{N}(0.75,1)$ .

Expanding we get  $p_n = \rho \sum_{k=0}^{n-1} (1 - \rho)^k$ . The desired expression for the mean delay is obtained from the equation  $p_\tau = 1 - \alpha$ .

### 3.06.4 Minimax quickest change detection

When the distribution of the change point is not known, we may model the change point as a deterministic but unknown positive integer  $\gamma$ . A number of heuristic algorithms have been developed in this setting. The earliest work is due to Shewhart [1,2], in which the log likelihood based on the current observation is compared with a threshold to make a decision about the change. The motivation for such a technique is based on the following fact: if  $X$  represents the generic random variable for the i.i.d. model with  $f_0$  and  $f_1$  as the pre- and post-change p.d.fs, then

$$\mathbb{E}_\infty [\log L(X)] = -D(f_0 \| f_1) < 0 \quad \text{and} \quad \mathbb{E}_1 [\log L(X)] = D(f_1 \| f_0) > 0, \quad (6.41)$$

where as defined earlier  $L(x) = f_1(x)/f_0(x)$ , and  $\mathbb{E}_\infty$  and  $\mathbb{E}_1$  correspond to expectations when  $\gamma = \infty$  and  $\gamma = 1$ , respectively. Thus, after  $\gamma$ , the log likelihood of the observation  $X$  is more likely to be above a given threshold. Shewhart's method is widely employed in practice due to its simplicity; however, significant performance gain can be achieved by making use of past observations to make the decision about the change. Page [3] proposed such an algorithm that uses past observations, which he called the CuSum algorithm. The motivation for the CuSum algorithm is also based on (6.41). By the law of large numbers,  $\sum_{i=\gamma}^n \log L(X_i)$  grows to  $\infty$  as  $n \rightarrow \infty$ . Thus, if  $S_n = \sum_{i=1}^n \log L(X_i)$  is the accumulated log likelihood sum, then before  $\gamma$ ,  $S_n$  has a *negative drift* and evolves towards  $-\infty$ . After  $\gamma$ ,  $S_n$  has a

*positive drift* and climbs towards  $\infty$ . Therefore, intuition suggests the following algorithm should detect this change in drift:

$$\tau_C = \inf \left\{ n \geq 1 : \left( S_n - \min_{1 \leq k \leq n} S_k \right) \geq b \right\}, \quad (6.42)$$

where  $b > 0$ . Note that

$$S_n - \min_{1 \leq k \leq n} S_k = \max_{0 \leq k \leq n} \sum_{i=k+1}^n \log L(X_i) = \max_{1 \leq k \leq n+1} \sum_{i=k}^n \log L(X_i).$$

Thus,  $\tau_C$  can be equivalently defined as follows.

**Algorithm 2 (CuSum algorithm).**

$$\tau_C = \inf \{n \geq 1 : W_n \geq b\}, \quad (6.43)$$

where

$$W_n = \max_{1 \leq k \leq n+1} \sum_{i=k}^n \log L(X_i). \quad (6.44)$$

The statistic  $W_n$  has the convenient recursion:

$$W_{n+1} = (W_n + \log L(X_{n+1}))^+, \quad W_0 = 0. \quad (6.45)$$

It is this cumulative sum recursion that led Page to call  $\tau_C$  the CuSum algorithm.

The summation on the right hand side (RHS) of (6.44) is assumed to take the value 0 when  $k = n+1$ . It turns out that one can get an algorithm that is equivalent to the above CuSum algorithm by removing the term  $k = n+1$  in the maximization on the RHS of (6.44), to get the statistic:

$$C_n = \max_{1 \leq k \leq n} \sum_{i=k}^n \log L(X_i). \quad (6.46)$$

The statistic  $C_n$  also has a convenient recursion:

$$C_{n+1} = (C_n)^+ + \log L(X_{n+1}), \quad C_0 = 0. \quad (6.47)$$

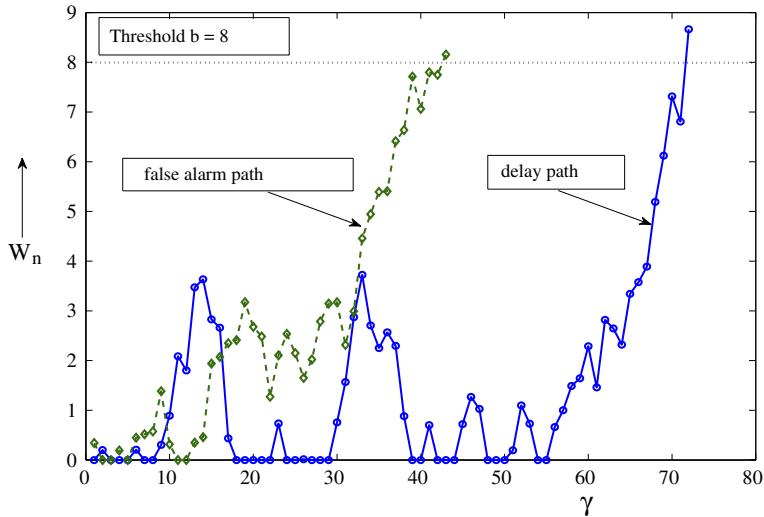
Note that unlike the statistic  $W_n$ , the statistic  $C_n$  can be negative. Nevertheless it is easy to see that both  $W_n$  and  $C_n$  will cross a positive threshold  $b$  at the same time (sample path wise) and hence the CuSum algorithm can be equivalently defined in terms of  $C_n$  as:

$$\tau_C = \inf \{n \geq 1 : C_n \geq b\}. \quad (6.48)$$

An alternative way to derive the CuSum algorithm is through the maximum likelihood approach i.e., to compare the likelihood of  $\{\Gamma \leq n\}$  against  $\{\Gamma > n\}$ . Formally,

$$\tau_C = \inf \left\{ n \geq 1 : \frac{\max_{1 \leq k \leq n} \prod_{i=1}^{k-1} f_0(X_i) \prod_{i=k}^n f_1(X_i)}{\prod_{i=1}^n f_0(X_i)} \geq B \right\}. \quad (6.49)$$

Cancelling terms and taking log in (6.49) gives us (6.48) with  $b = \log B$ .

**FIGURE 6.4**

Typical Evolution of the CuSum algorithm. Threshold  $b = 8$  and change point  $\gamma = 60$ .

See Figure 6.4 for a typical evolution of the CuSum algorithm.

Although, the CuSum algorithm was developed heuristically by Page [3], it was later shown in [6–8, 10], that it has very strong optimality properties. In this section, we will study the CuSum and related algorithms from a fundamental viewpoint, and discuss how each of these algorithms is provably optimal with respect to a meaningful and useful optimization criterion.

Without a prior on the change point, a reasonable measure of false alarms is the mean time to false alarm, or its reciprocal, which is the false alarm rate (FAR):

$$\text{FAR}(\tau) = \frac{1}{\mathbb{E}_\infty[\tau]}. \quad (6.50)$$

Finding a uniformly powerful test that minimizes the delay over all possible values of  $\gamma$  subject to a FAR constraint is generally not possible. Therefore it is more appropriate to study the quickest change detection problem in a minimax setting in this case. There are two important minimax problem formulations, one due to Lorden [6] and the other due to Pollak [9].

In Lorden's formulation, the objective is to minimize the supremum of the average delay conditioned on the worst possible realizations, subject to a constraint on the false alarm rate. In particular, if we define<sup>3</sup>

$$\text{WADD}(\tau) = \sup_{n \geq 1} \text{ess sup } \mathbb{E}_n [(\tau - n)^+ | X_1, \dots, X_{n-1}]. \quad (6.51)$$

<sup>3</sup>Lorden defined WADD with  $(\tau - n + 1)^+$  inside the expectation, i.e., he assumed a delay penalty of 1 if the algorithm stops at the change point. We drop this additional penalty in our definition in order to be consistent with the other delay definitions in this chapter.

and denote the set of stopping times that meet a constraint  $\alpha$  on the FAR by

$$\mathcal{D}_\alpha = \{\tau : \text{FAR}(\tau) \leq \alpha\}. \quad (6.52)$$

We have the following problem formulation due to Lorden.

*Lorden's Problem:* For a given  $\alpha$ , find a stopping time  $\tau \in \mathcal{D}_\alpha$  to minimize  $\text{WADD}(\tau)$ . (6.53)

For the i.i.d. setting, Lorden showed that the CuSum algorithm (6.43) is asymptotically optimal for Lorden's formulation (6.53) as  $\alpha \rightarrow 0$ . It was later shown in [7,8] that the CuSum algorithm is actually exactly optimal for (6.53). Although the CuSum algorithm enjoys such a strong optimality property under Lorden's formulation, it can be argued that WADD is a somewhat pessimistic measure of delay. A less pessimistic way to measure the delay was suggested by Pollak [9]:

$$\text{CADD}(\tau) = \sup_{n \geq 1} \mathbb{E}_n[\tau - n | \tau \geq n]. \quad (6.54)$$

for all stopping times  $\tau$  for which the expectation is well-defined.

**Lemma 1.**

$$\text{WADD}(\tau) \geq \text{CADD}(\tau).$$

**Proof.** Due to the fact that  $\tau$  is a stopping time on  $\{X_n\}$ ,

$$\{\tau \geq n\} = \{\tau \leq n-1\}^c \in \sigma(X_1, X_2, \dots, X_{n-1}).$$

Therefore, for each  $n$

$$\text{ess sup } \mathbb{E}_n[(\tau - n)^+ | X_1, \dots, X_{n-1}] \geq \mathbb{E}_n[(\tau - n)^+ | \tau \geq n] = \mathbb{E}_n[\tau - n | \tau \geq n]$$

and the lemma follows. ■

We now state Pollak's formulation of the problem that uses CADD as the measure of delay.

*Pollak's Problem:* For a given  $\alpha$ , find a stopping time  $\tau \in \mathcal{D}_\alpha$  to minimize  $\text{CADD}(\tau)$ . (6.55)

Pollak's formulation has been studied in the i.i.d. setting in [9,20], where it is shown that algorithms based on the Shiryaev-Roberts statistic (to be defined later) are within a constant of the best possible performance over the class  $\mathcal{D}_\alpha$ , as  $\alpha \rightarrow 0$ .

Lai [10] studied both (6.53) and (6.55) in a non-i.i.d. setting and developed a general minimax asymptotic theory for these problems. In particular, Lai obtained a lower bound on  $\text{CADD}(\tau)$ , and hence also on the  $\text{WADD}(\tau)$ , for every stopping time in the class  $\mathcal{D}_\alpha$ , and showed that an extension of the CuSum algorithm for the non-i.i.d. setting achieves this lower bound asymptotically as  $\alpha \rightarrow 0$ .

In Section 3.06.4.1 we introduce a number of alternatives to the CuSum algorithm for minimax quickest change detection in the i.i.d. setting that are based on the Bayesian Shiryaev algorithm. We then discuss the optimality properties of these algorithms in Section 3.06.4.2. While we do not discuss the exact optimality of the CuSum algorithm from [7] or [8], we briefly discuss the asymptotic optimality result from [6]. We also note that the asymptotic optimality of the CuSum algorithm for both (6.53) and (6.55) follows from the results in the non-i.i.d. setting of [10], which are summarized in Section 3.06.4.3.

### 3.06.4.1 Minimax algorithms based on the Shiryaev algorithm

Recall that the Shiryaev algorithm is given by (see (6.20) and (6.23)):

$$\tau_S = \inf \{n \geq 1 : R_{n,\rho} \geq a\},$$

where

$$R_{n,\rho} = \frac{1}{(1-\rho)^n} \sum_{k=1}^n (1-\rho)^{k-1} \prod_{i=k}^n L(X_i).$$

Also recall that  $R_{n,\rho}$  has the recursion:

$$R_{n+1,\rho} = \frac{1 + R_{n,\rho}}{1 - \rho} L(X_{n+1}), \quad R_{0,\rho} = 0.$$

Setting  $\rho = 0$  in the expression for  $R_{n,\rho}$  we get the Shiryaev-Roberts (SR) statistic [21]:

$$R_n = \sum_{k=1}^n \prod_{i=k}^n L(X_i) \tag{6.56}$$

with the recursion:

$$R_{n+1} = (1 + R_n)L(X_{n+1}), \quad R_0 = 0. \tag{6.57}$$

**Algorithm 3 (Shiryaev-Roberts (SR) Algorithm).**

$$\tau_{SR} = \inf \{n \geq 1 : R_n \geq B, \quad R_0 = 0\}. \tag{6.58}$$

It is shown in [9] that the SR algorithm is the limit of a sequence of Bayes tests, and in that limit it is asymptotically Bayes efficient. Also, it is shown in [20] that the SR algorithm is *second order* asymptotically optimal for (6.55), as  $\alpha \rightarrow 0$ , i.e., its delay is within a constant of the best possible delay over the class  $\mathcal{D}_\alpha$ . Further, in [22], the SR algorithm is shown to be exactly optimal with respect to a number of other interesting criteria.

It is also shown in [9] that a modified version of the SR algorithm, called the Shiryaev-Roberts-Pollak (SRP) algorithm, is *third order* asymptotically optimal for (6.55), i.e., its delay is within a constant of the best possible delay over the class  $\mathcal{D}_\alpha$ , and the constant goes to zero as  $\alpha \rightarrow 0$ . To introduce the SRP algorithm, let  $Q^B$  be the quasi-stationary distribution of the SR statistic  $R_n$  above:

$$Q^B(x) = \lim_{n \rightarrow \infty} \mathbb{P}_0(R_n \leq x | \tau_{SR} > n).$$

The new recursion, called the Shiryaev-Roberts-Pollak (SRP) recursion, is given by,

$$R_{n+1}^B = (1 + R_n^B)L(X_{n+1})$$

with  $R_0^B$  distributed according to  $Q^B$ .

**Algorithm 4 (Shiryaev-Roberts-Pollak (SRP) Algorithm).**

$$\tau_{SRP} = \inf \left\{ n \geq 1 : R_n^B \geq B \right\}. \tag{6.59}$$

Although the SRP algorithm is strongly asymptotically optimal for Pollak's formulation of (6.55), in practice, it is difficult to compute the quasi-stationary distribution  $Q^B$ . A numerical framework for computing  $Q^B$  efficiently is provided in [23]. Interestingly, the following modification of the SR algorithm with a *specifically designed* starting point  $R_0 = r \geq 0$  is found to outperform the SRP procedure uniformly over all possible values of the change point [20]. This modification, referred to as the SR- $r$  procedure, has the recursion:

$$R_{n+1}^r = (1 + R_n^r)L(X_{n+1}), \quad R_0 = r.$$

**Algorithm 5 (Shiryayev-Roberts- $r$  (SR- $r$ ) Algorithm).**

$$\tau_{\text{SR}-r} = \inf \{n \geq 1 : R_n^r \geq B\}. \quad (6.60)$$

It is shown in [20] that the SR- $r$  algorithm is also third order asymptotically optimal for (6.55), i.e., its delay is within a constant of the best possible delay over the class  $\mathcal{D}_\alpha$ , and the constant goes to zero as  $\alpha \rightarrow 0$ .

Note that for an arbitrary stopping time, computing the CADD metric (6.54) involves taking supremum over all possible change times, and computing the WADD metric (6.51) involves another supremum over all possible past realizations of observations. While we can analyze the performance of the proposed algorithms through bounds and asymptotic approximations, as we will see in Section 3.06.4.2, it is not obvious how one might evaluate CADD and WADD for a given algorithm in computer simulations. This is in contrast with the Bayesian setting, where ADD (see (6.7)) can easily be evaluated in simulations by averaging over realizations of change point random variable  $\Gamma$ .

Fortunately, for the CuSum algorithm (6.43) and for the Shiryaev-Roberts algorithm (6.58), both CADD and WADD are easy to evaluate in simulations due to the following lemma.

**Lemma 2.**

$$\text{CADD}(\tau_C) = \text{WADD}(\tau_C) = \mathbb{E}_1 [(\tau_C - 1)], \quad (6.61)$$

$$\text{CADD}(\tau_{\text{SR}}) = \text{WADD}(\tau_{\text{SR}}) = \mathbb{E}_1 [(\tau_{\text{SR}} - 1)]. \quad (6.62)$$

**Proof.** The CuSum statistic  $W_n$  (see (6.44)) has initial value 0 and remains non-negative for all  $n$ . If the change were to happen at some time  $n > 1$ , then the pre-change statistic  $W_{n-1}$  is greater than or equal 0, which equals the pre-change statistic if the change happens at  $n = 1$ . Therefore, the delay for the CuSum statistic to cross a positive threshold  $b$  is largest when the change happens at  $n = 1$ , irrespective of the realizations of the observations,  $X_1, X_2, \dots, X_{n-1}$ . Therefore

$$\text{WADD}(\tau_C) = \sup_{n \geq 1} \text{ess sup } \mathbb{E}_n [(\tau_C - n)^+ | X_1, \dots, X_{n-1}] = \mathbb{E}_1 [(\tau_C - 1)^+] = \mathbb{E}_1 [(\tau_C - 1)]$$

and

$$\text{CADD}(\tau_C) = \sup_{n \geq 1} \mathbb{E}_n [\tau - n | \tau_C \geq n] = \mathbb{E}_1 [(\tau_C - 1) | \tau_C \geq 1] = \mathbb{E}_1 [(\tau_C - 1)].$$

This proves (6.61). A similar argument can be used to establish (6.62).

Note that the above proof crucially depended on the fact that both the CuSum algorithm and the Shiryaev-Roberts algorithm start with the initial value of 0. Thus it is not difficult to see that Lemma 2 does not hold for the SR- $r$  algorithm, unless of course  $r = 0$ . Lemma 2 holds partially for the SRP algorithm since the initial distribution  $Q^B$  makes the statistic  $R_n^B$  stationary in  $n$ . As a result  $\mathbb{E}_n[\tau_{\text{SRP}} - n | \tau_{\text{SRP}} \geq n]$  is the same for every  $n$ . However, as mentioned previously,  $Q^B$  is difficult to compute in practice, and this makes the evaluation of CADD and WADD in simulations somewhat challenging.

### 3.06.4.2 Optimality properties of the minimax algorithms

In this section we first show that the algorithms based on the Shiryaev-Roberts statistics, SR, SRP, and SR- $r$  are asymptotically optimal for Pollak's formulation of (6.55). We need an important theorem that is proved in [22].

**Theorem 11 [22].** *If the threshold in the SR algorithm (6.58) can be selected to meet the constraint  $\alpha$  on FAR with equality, then*

$$\tau_{\text{SR}} = \arg \min_{\tau \in \mathcal{D}_\alpha} \frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau - n)^+]}{\mathbb{E}_\infty[\tau]}.$$

**Proof.** We give a sketch of the proof here. Note that

$$\begin{aligned} \sum_{n=1}^{\infty} \mathbb{E}_n[(\tau - n)^+] &= \sum_{n=1}^{\infty} \mathbb{E}_n[\tau - n | \tau \geq n] \mathbb{P}_\infty(\tau \geq n) \\ &\leq \text{CADD}(\tau) \sum_{n=1}^{\infty} \mathbb{P}_\infty(\tau \geq n) \\ &= \text{CADD}(\tau) \mathbb{E}_\infty[\tau], \end{aligned}$$

and hence is finite for all stopping times for which CADD and FAR are finite. The first part of the proof is to show that

$$\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau_{\text{SR}} - n)^+] = \min_{\tau \in \mathcal{D}_\alpha} \sum_{n=1}^{\infty} \mathbb{E}_n[(\tau - n)^+].$$

This follows from the following result of [9]. If

$$J_{\lambda, \rho}(\tau) = \min_{\tau} (\mathbb{E}[(\tau - \Gamma)^+] + \lambda \mathbb{P}(\tau < \Gamma))$$

with  $\Gamma$  having the geometric distribution of (6.11) with parameter  $\rho$ , and

$$\tau_{\lambda, \rho} = \arg \min_{\tau} J_{\lambda, \rho}(\tau). \quad (6.63)$$

Then for a given  $\tau_{\text{SR}}$  (with a given threshold), there exists a sequence  $\{(\lambda_i, \rho_i)\}$  and with  $\lambda_i \rightarrow \lambda^*$  and  $\rho_i \rightarrow 0$  such that  $\tau_{\lambda_i, \rho_i}$  converge to  $\tau_{\text{SR}}$ , as  $i \rightarrow \infty$ . Thus, the SR algorithm is the limit of a sequence of Bayes tests. Moreover,

$$\limsup_{\rho \rightarrow 0, \lambda \rightarrow \lambda^*} \frac{1 - J_{\lambda, \rho}(\tau_{\lambda, \rho})}{1 - J_{\lambda, \rho}(\tau_{\text{SR}})} = 1.$$

By (6.63), for any stopping time  $\tau$ , it holds that

$$\frac{1 - J_{\lambda,\rho}(\tau)}{1 - J_{\lambda,\rho}(\tau_{SR})} \leq \frac{1 - J_{\lambda,\rho}(\tau_{SR})}{1 - J_{\lambda,\rho}(\tau_{SR})}.$$

Now by taking the limit  $\rho \rightarrow 0, \lambda \rightarrow \lambda^*$  on both sides, using the fact that for any stopping time  $\tau$  [22]

$$\frac{1 - J_{\lambda,\rho}(\tau)}{\rho} \rightarrow \lambda^* \mathbb{E}_{\infty}[\tau] - \sum_{n=1}^{\infty} \mathbb{E}[(\tau - n)^+] \text{ as } \rho \rightarrow 0$$

and using the hypothesis of the theorem that the FAR constraint can be met with equality by using  $\tau_{SR}$ , we have the desired result.

The next step in the proof is to show that it is enough to consider stopping times in the class  $\mathcal{D}_\alpha$  that meet the constraint of  $\alpha$  with quality. The result then follows easily from the fact that  $\tau_{SR}$  is optimal with respect to the numerator in  $\frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau - n)^+]}{\mathbb{E}_{\infty}[\tau]}$ . ■

We now state the optimality proof for the procedures SR, SR- $r$  and SRP. We only provide an outline of the proof to illustrate the fundamental ideas behind the result.

**Theorem 12 [20].** If  $\mathbb{E}_1 \left[ \log \frac{f_1(X_n)}{f_0(X_n)} \right]^2 < \infty$ , and  $\log \frac{f_1(X_n)}{f_0(X_n)}$  is nonarithmetic then

1.

$$\inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) \geq \frac{|\log \alpha|}{D(f_1 \| f_0)} + \hat{\kappa} + o(1) \text{ as } \alpha \rightarrow 0, \quad (6.64)$$

where  $\hat{\kappa}$  is a constant that can be characterized using renewal theory [18].

2. For the choice of threshold  $B = \frac{1}{\alpha}$ ,  $\text{FAR}(\tau_{SR}) \leq \alpha$ , and

$$\begin{aligned} \text{CADD}(\tau_{SR}) &= \frac{|\log \alpha|}{D(f_1 \| f_0)} + \hat{\zeta} + o(1) \text{ as } \alpha \rightarrow 0 \\ &= \inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) + O(1) \text{ as } \alpha \rightarrow 0, \end{aligned} \quad (6.65)$$

where  $\hat{\zeta}$  is again a constant that can be characterized using renewal theory [18].

3. There exists a choice for the threshold  $B = B_\alpha$  such that  $\text{FAR}(\tau_{SRP}) \leq \alpha(1 + o(1))$  and

$$\begin{aligned} \text{CADD}(\tau_{SRP}) &= \frac{|\log \alpha|}{D(f_1 \| f_0)} + \hat{\kappa} + o(1) \text{ as } \alpha \rightarrow 0 \\ &= \inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) + o(1) \text{ as } \alpha \rightarrow 0. \end{aligned} \quad (6.66)$$

4. There exists a choice for the threshold  $B = B_\alpha$  such that  $\text{FAR}(\tau_{SRP}) \leq \alpha(1 + o(1))$  and a choice for the initial point  $r$  such that

$$\begin{aligned} \text{CADD}(\tau_{SR-r}) &= \frac{|\log \alpha|}{D(f_1 \| f_0)} + \hat{\kappa} + o(1) \text{ as } \alpha \rightarrow 0 \\ &= \inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) + o(1) \text{ as } \alpha \rightarrow 0. \end{aligned} \quad (6.67)$$

**Proof.** To prove that the above mentioned choice of thresholds meets the FAR constraint, we note that  $\{R_n^r - n - r\}$  is a martingale. Specifically,  $\{R_n - n\}$  is a martingale and the conditions of Theorem 3 are satisfied [24]. Hence,

$$\mathbb{E}_\infty[R_{\tau_{SR}} - \tau_{SR}] = 0.$$

Since,  $\mathbb{E}_\infty[R_{\tau_{SR}}] \geq B$ , for  $B = \frac{1}{\alpha}$  we have

$$\text{FAR}(\tau_{SR}) = \frac{1}{\mathbb{E}_\infty[R_{\tau_{SR}}]} \leq \frac{1}{B} = \alpha.$$

For a description of how to set the thresholds for  $\tau_{SR-r}$  and  $\tau_{SRP}$ , we refer the reader to [20].

The proofs of the delay expressions for all the algorithms have a common theme. The first part of these proofs is based on Theorem 11. We first show that  $\frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau-n)^+]}{\mathbb{E}_\infty[\tau]}$  is a lower bound to CADD( $\tau$ ).

$$\begin{aligned} \text{CADD}(\tau) &= \sup_n \mathbb{E}_n[\tau - n | \tau \geq n] = \frac{\sum_{j=1}^{\infty} \sup_n \mathbb{E}_n[\tau - n | \tau \geq n] \mathbb{P}_\infty(\tau \geq j)}{\mathbb{E}_\infty[\tau]} \\ &\geq \frac{\sum_{j=1}^{\infty} \mathbb{E}_j[\tau - j | \tau \geq j] \mathbb{P}_\infty(\tau \geq j)}{\mathbb{E}_\infty[\tau]} \\ &= \frac{\sum_{j=1}^{\infty} \mathbb{E}_j[(\tau - j)^+]}{\mathbb{E}_\infty[\tau]}. \end{aligned}$$

From Theorem 11, since  $\tau_{SR}$  is optimal with respect to minimizing  $\frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau-n)^+]}{\mathbb{E}_\infty[\tau]}$ , we have

$$\inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) \geq \frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau_{SR} - n)^+]}{\mathbb{E}_\infty[\tau_{SR}]}.$$

The next step is to use nonlinear renewal theory (see Section 3.06.2.3) to obtain a second order approximation for the right hand side of the above equation, as we did for the Shiryaev procedure in Section 3.06.3.3:

$$\frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau_{SR} - n)^+]}{\mathbb{E}_\infty[\tau_{SR}]} = \frac{|\log \alpha|}{D(f_1 \| f_0)} + \hat{\kappa} + o(1) \quad \text{as } \alpha \rightarrow 0.$$

The final step is to show that the CADD for the SR- $r$  and SRP procedures are within  $o(1)$ , and the CADD for SR procedure is within  $O(1)$  of this lower bound (6.64). This is done by obtaining second order approximations using nonlinear renewal theory for the CADD of SR, SRP, SR- $r$  procedures, and get (6.65–6.67), respectively.

It is shown in [22] that  $\frac{\sum_{n=1}^{\infty} \mathbb{E}_n[(\tau-n)^+]}{\mathbb{E}_\infty[\tau]}$  is also equivalent to the average delay when the change happens at a “far horizon”:  $\gamma \rightarrow \infty$ . Thus, the SR algorithm is also optimal with respect to that criterion.

The following corollary follows easily from the above two theorems. Recall that in the minimax setting, an algorithm is called third order asymptotically optimum if its delay is within an  $o(1)$  term of the best possible, as the FAR goes to zero. An algorithm is called second order asymptotically optimum if its delay is within an  $O(1)$  term of the best possible, as the FAR goes to zero. And an algorithm is called first order asymptotically optimal if the ratio of its delay with the best possible goes to 1, as the FAR goes to zero.

**Corollary 2.** Under the conditions of Theorem 11, the SR- $r$  (6.60) and the SRP (6.59) algorithms are third order asymptotically optimum, and the SR algorithm is second order asymptotically optimum for the Pollak's formulation (6.55). Furthermore, using the choice of thresholds specified in Theorem 11 to meet the FAR constraint of  $\alpha$ , the asymptotic performance of all three algorithms are equal up to first order:

$$\text{CADD}(\tau_{SR}) \sim \text{CADD}(\tau_{SRP}) \sim \text{CADD}(\tau_{SR-r}) \sim \frac{|\log \alpha|}{D(f_1 \| f_0)}.$$

Furthermore, by Lemma 2, we also have:

$$\text{WADD}(\tau_{SR}) \sim \frac{|\log \alpha|}{D(f_1 \| f_0)}.$$

In [6] the asymptotic optimality of the CuSum algorithm (6.43) as  $\alpha \rightarrow 0$  is established for Lorden's formulation of (6.53). First, ergodic theory is used to show that choosing the threshold  $b = |\log \alpha|$  ensures  $\text{FAR}(\tau_C) \leq \alpha$ . For the above choice of threshold  $B = |\log \alpha|$ , it is shown that

$$\text{WADD}(\tau_C) \leq \frac{|\log \alpha|}{D(f_1 \| f_0)}(1 + o(1)) \quad \text{as } \alpha \rightarrow 0.$$

Then the exact optimality of the SPRT [25] is used to find a lower bound on the WADD of the class  $\mathcal{D}_\alpha$ :

$$\inf_{\tau \in \mathcal{D}_\alpha} \text{WADD}(\tau) \geq \frac{|\log \alpha|}{D(f_1 \| f_0)}(1 + o(1)) \quad \text{as } \alpha \rightarrow 0.$$

These arguments establish the first order asymptotic optimality of the CuSum algorithm for Lorden's formulation. Furthermore, as we will see later in Theorem 13, Lai [10] showed that:

$$\inf_{\tau \in \mathcal{D}_\alpha} \text{WADD}(\tau) \geq \inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) \geq \frac{|\log \alpha|}{I}(1 + o(1)).$$

Thus by Lemma 2, the first order asymptotic optimality of the CuSum algorithm extends to Pollak's formulation (6.55), and we have the following result.

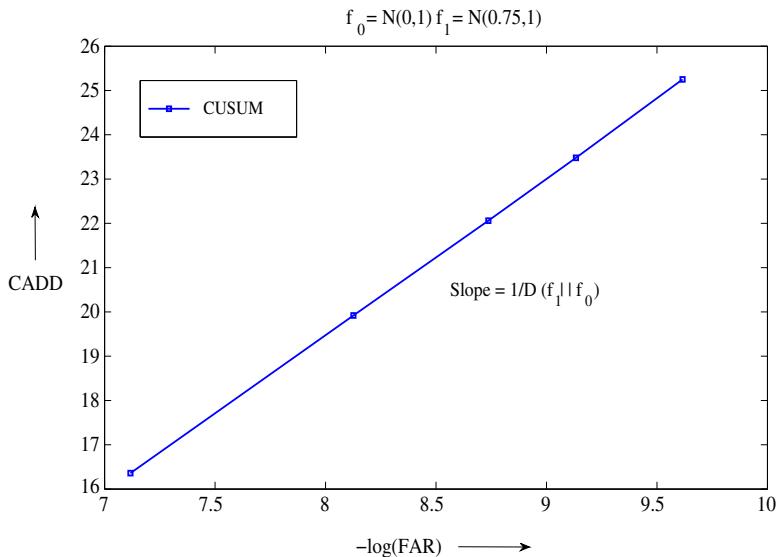
**Corollary 3.** The CuSum algorithm (6.43) with threshold  $b = |\log \alpha|$  is first order asymptotically optimum for both Lorden's and Pollak's formulations. Furthermore,

$$\text{CADD}(\tau_C) = \text{WADD}(\tau_C) \sim \frac{|\log \alpha|}{D(f_1 \| f_0)}.$$

In Figure 6.5, we plot the trade-off curve for the CuSum algorithm, i.e., plot CADD as a function of  $-\log \text{FAR}$ . Note that the curve has a slope of  $1/D(f_1 \| f_0)$ .

### 3.06.4.3 General asymptotic minimax theory

In [10], the non-i.i.d. setting earlier discussed in Section 3.06.3.2 is considered, and asymptotic lower bounds on the WADD and CADD for stopping times in  $\mathcal{D}_\alpha$  are obtained under quite general conditions.

**FIGURE 6.5**

ADD-PFA trade-off curve for the CuSum algorithm:  $f_0 = \mathcal{N}(0,1)$ ,  $f_1 = \mathcal{N}(0.75,1)$ .

It is then shown that the an extension of the CuSum algorithm (6.43) to this non-i.i.d. setting achieves this lower bound asymptotically as  $\alpha \rightarrow 0$ .

Recall the non-i.i.d. model from Section 3.06.3.2. When the process evolves in the pre-change regime, the conditional density of  $X_n$  given  $X_1^{n-1}$  is  $f_{0,n}(X_n|X_1^{n-1})$ . After the change happens, the conditional density of  $X_n$  given  $X_1^{n-1}$  is given by  $f_{1,n}(X_n|X_1^{n-1})$ . Also

$$Y_i = \log \frac{f_{1,i}(X_i|X_1^{i-1})}{f_{0,i}(X_i|X_1^{i-1})}.$$

The CuSum algorithm can be generalized to the non-i.i.d. setting as follows.

**Algorithm 6 (Generalized CuSum algorithm).** Let

$$C_n = \max_{1 \leq k \leq n} \sum_{i=k}^n Y_i.$$

Then the stopping time for the generalized CuSum is

$$\tau_G = \inf \{n \geq 1 : C_n \geq b\}. \quad (6.68)$$

Then the following result is proved in [10].

**Theorem 13.** If there exists a positive constant  $I$  such that the  $\{Y_i\}$ s satisfy the following condition

$$\lim_{t \rightarrow \infty} \sup_{n \geq 1} \text{ess sup } \mathbb{P}_n \left( \max_{m \leq t} \sum_{i=n}^{n+m} Y_i \geq I(1 + \delta)n \middle| X_1, \dots, X_{n-1} \right) = 0 \quad \forall \delta > 0 \quad (6.69)$$

then, as  $\alpha \rightarrow 0$

$$\inf_{\tau \in \mathcal{D}_\alpha} \text{WADD}(\tau) \geq \inf_{\tau \in \mathcal{D}_\alpha} \text{CADD}(\tau) \geq \frac{|\log \alpha|}{I} (1 + o(1)). \quad (6.70)$$

Further

$$\mathbb{E}_\infty[\tau_G] \geq e^b,$$

and under the additional condition of

$$\lim_{t \rightarrow \infty} \sup_{m \geq n} \text{ess sup } \mathbb{P}_n \left( t^{-1} \sum_{i=m}^{m+t} Y_i \geq I - \delta \middle| X_1, \dots, X_{m-1} \right) = 0 \quad \forall \delta > 0, \quad (6.71)$$

we have

$$\text{CADD}(\tau_G) \leq \text{WADD}(\tau_G) \leq \frac{b}{I} (1 + o(1)) \text{ as } b \rightarrow \infty.$$

Thus, if we set  $b = |\log \alpha|$ , then

$$\text{FAR}(\tau_G) = \frac{1}{\mathbb{E}_\infty[\tau_G]} \leq \alpha$$

and

$$\text{WADD}(\tau_G) \leq \frac{|\log \alpha|}{I} (1 + o(1)),$$

which is asymptotically equal to the lower bound in (6.70) up to first order. Thus  $\tau_G$  is first-order asymptotically optimum within the class  $\mathcal{D}_\alpha$  of tests that meet the FAR constraint of  $\alpha$ .

**Proof.** We only provide a sketch of the proof for the lower bound since it also based on the idea of using Chebyshev's inequality. The fundamental idea of the proof is to use Chebyshev's inequality to get a lower bound on any arbitrary stopping time  $\tau$  from  $\mathcal{D}_\alpha$ , such that the lower bound is not a function of  $\tau$ . The lower bound obtained is then a lower bound on the CADD for the entire family  $\mathcal{D}_\alpha$ .

Let  $L_\alpha$  and  $V_\alpha$  be positive constants. To show that

$$\sup_{n \geq 1} \mathbb{E}_n[\tau - n | \tau \geq n] \geq L_\alpha(1 + o(1)) \text{ as } \alpha \rightarrow 0$$

it is enough to show that there exists  $n$  such that

$$\mathbb{E}_n[\tau - n | \tau \geq n] \geq L_\alpha(1 + o(1)) \text{ as } \alpha \rightarrow 0.$$

This  $n$  is obtained from the following condition. Let  $m$  be any positive integer. Then if  $\tau \in \mathcal{D}_\alpha$ , there exists  $n$  such that

$$\mathbb{P}_\infty(\tau \geq n) > 0 \quad \text{and} \quad \mathbb{P}_\infty(\tau < n + m | \tau \geq k) \leq m\alpha. \quad (6.72)$$

We use the  $n$  that satisfies the condition (6.72). Then, by Chebyshev's inequality

$$\mathbb{P}_n(\tau - n \geq L_\alpha | \tau \geq n) \leq (L_\alpha)^{-1} \mathbb{E}_n[\tau - n | \tau \geq n].$$

We can then write

$$\mathbb{E}_n[\tau - n | \tau \geq n] \geq L_\alpha \mathbb{P}_n(\tau - n \geq L_\alpha | \tau \geq n).$$

Now it has to be shown that  $\mathbb{P}_n(\tau - n \geq L_\alpha | \tau \geq n) \rightarrow 1$  uniformly over the family  $\mathcal{D}_\alpha$ . To show this, we condition on  $V_\alpha$ .

$$\begin{aligned} \mathbb{P}_n(\tau - n < L_\alpha | \tau \geq n) &= \mathbb{P}_n\left(\tau - n < L_\alpha; \sum_{i=n}^{\tau} Y_i < V_\alpha | \tau \geq n\right) \\ &\quad + \mathbb{P}_n\left(\tau - n < L_\alpha; \sum_{i=n}^{\tau} Y_i \geq V_\alpha | \tau \geq n\right). \end{aligned}$$

The trick now is to use the hypothesis of the theorem and choose proper values of  $V_\alpha$  and  $L_\alpha$  such that the two terms on the right hand side of the above equations are bounded by terms that go to zero and are not a function of the stopping time  $\tau$ . We can write

$$\mathbb{P}_n\left(\tau - n < L_\alpha; \sum_{i=n}^{\tau} Y_i \geq V_\alpha | \tau \geq n\right) \leq \text{ess sup } \mathbb{P}_n\left(\max_{t \leq L_\alpha} \sum_{i=n}^{n+t} Y_i \geq V_\alpha | X_1, \dots, X_{n-1}\right).$$

In [10], it is shown that if we choose

$$L_\alpha = (1 - \delta) \frac{|\log \alpha|}{I}$$

and

$$V_\alpha = (1 - \delta^2) |\log \alpha|$$

then the condition (6.69) ensures that the above probability goes to zero. The other term goes to zero by using a change of measure argument and using condition (6.72):

$$\mathbb{P}_n\left(\tau - n < L_\alpha; \sum_{i=n}^{\tau} Y_i < V_\alpha | \tau \geq n\right) \leq e^{V_\alpha} \mathbb{P}_\infty(\tau < n + L_\alpha | \tau \geq n). \quad \blacksquare$$

### 3.06.5 Relationship between the models

We have discussed the Bayesian formulation of the quickest change detection problem in Section 3.06.3 and the minimax formulations of the problem in Section 3.06.4. The choice between the Bayesian and the minimax formulations is obvious, and is governed by the knowledge of the distribution of  $\Gamma$ . However, it is not obvious which of the two minimax formulations should be chosen for a given application. As noted

earlier, the formulations of Lorden and Pollak are equivalent for low FAR constraints, but differences arise for moderate values of FAR constraints. Recent work by Moustakides [26] has connected these three formulations and found possible interpretations for each of them. We summarize the result below.

Consider a model in which the change point is dependent on the stochastic process. That is, the probability that change happens at time  $n$  depends on  $\{X_1, \dots, X_n\}$ . Let

$$\pi_n = \mathbb{P}(\Gamma = n | X_1, \dots, X_n).$$

The distribution of  $\Gamma$  thus belongs to a family of distributions. Assume that while we are trying to find a suitable stopping time  $\tau$  to minimize delay, an adversary is searching for a process  $\{\pi_n\}$  such that the delay for any stopping time is maximized. That is the adversary is trying to solve

$$J(\tau) = \sup_{\{\pi_n\}} \mathbb{E}[\tau - \Gamma | \tau \geq \Gamma].$$

It is shown in [26] that if the adversary has access to the observation sequence, and uses this information to choose  $\pi_n$ , then  $J(\tau)$  becomes the delay expression in Lorden's formulation (6.53) for a given  $\tau$ , i.e.,

$$J(\tau) = \sup_{n \geq 1} \text{ess sup } \mathbb{E}_n[(\tau - n)^+ | X_1, \dots, X_{n-1}].$$

However, if we assume that the adversary does not have access to the observations, but only has access to the test performance, then  $J(\tau)$  is equal to the delay in Pollak's formulation (6.55), i.e.,

$$J(\tau) = \sup_{n \geq 1} \mathbb{E}_n[\tau - n | \tau \geq n].$$

Finally, the delay for the Shiryaev formulation (6.10) corresponds to the case when  $\pi_n$  is restricted to only one possibility, the distribution of  $\Gamma$ .

### 3.06.6 Variants and generalizations of the quickest change detection problem

In the previous sections we reviewed the state-of-the-art for quickest change detection in a single sequence of random variables, under the assumption of complete knowledge of the pre- and post-change distributions. In this section we review three important variants and extensions of the classical quickest change detection problem, where significant progress has been made. We discuss other variants of the change detection problem as part of our future research section.

#### 3.06.6.1 Quickest change detection with unknown pre- or post-change distributions

In the previous sections we discussed quickest change detection problem when both the pre- and post-change distributions are completely specified. Although this assumption is a bit restrictive, it helped in obtaining recursive algorithms with strong optimality properties in the i.i.d. setting, and allowed the

development of asymptotic optimality theory in a very general non-i.i.d. setting. In this section, we provide a review of the existing results for the case where this assumption on the knowledge of the pre- and post-change distributions is relaxed.

Often it is reasonable to assume that the pre-change distribution is known. This is the case when changes occur rarely and/or the decision maker has opportunities to estimate the pre-change distribution parameters before the change occurs. When the post-change distribution is unknown but the pre-change distribution is completely specified, the post-change uncertainty may be modeled by assuming that the post-change distribution belongs to a parametric family  $\{\mathbb{P}_\theta\}$ . Approaches for designing algorithms in this setting include the generalized likelihood ratio (GLR) based approach or the mixture based approach. In the GLR based approach, at any time step, all the past observations are used to first obtain a maximum likelihood estimate of the post-change parameter  $\theta$ , and then the post-change distribution corresponding to this estimate of  $\theta$  is used to compute the CuSum, Shiryaev-Roberts or related statistics. In the mixture based approach, a prior is assumed on the space of parameters, and the statistics (e.g., CuSum or Shiryaev-Roberts), computed as a function of the post change parameter, are then integrated over the assumed prior.

In the i.i.d. setting this problem is studied in [6] for the case when the post-change p.d.f. belongs to a single parameter exponential family  $\{f_\theta\}$ , and the following generalized likelihood ratio (GLR) based algorithm is proposed:

$$\tau_G = \inf \left\{ n \geq 1 : \max_{1 \leq k \leq n} \sup_{|\theta| \geq \epsilon_\alpha} \left[ \sum_{i=k}^n \log \frac{f_\theta(X_i)}{f_0(X_i)} \right] \geq c_\alpha \right\}. \quad (6.73)$$

If  $\epsilon_\alpha = \frac{1}{|\log \alpha|}$ , and  $c_\alpha$  is of the order of  $|\log \alpha|$ , then it is shown in [6] that as the FAR constraint  $\alpha \rightarrow 0$ ,

$$\text{FAR}(\tau_G) \leq \alpha (1 + o(1)),$$

and for each post-change parameter  $\theta \neq 0$ ,

$$\text{WADD}^\theta(\tau_G) \sim \frac{|\log \alpha|}{D(f_\theta, f_0)}.$$

Here, the notation  $\text{WADD}^\theta[\cdot]$  is used to denote the WADD when the post-change observations have p.d.f  $f_\theta$ . Note that  $\frac{|\log \alpha|}{D(f_\theta, f_0)}$  is the best one can do asymptotically for a given FAR constraint of  $\alpha$ , even when the exact post-change distribution is known. Thus, this GLR scheme is uniformly asymptotically optimal with respect to the Lorden's criterion (6.53).

In [27], the post-change distribution is assumed to belong to an exponential family of p.d.f.s as above, but the GLR based test is replaced by a mixture based test. Specifically, a prior  $G(\cdot)$  is assumed on the range of  $\theta$  and the following test is proposed:

$$\tau_M = \inf \left\{ n \geq 1 : \max_{1 \leq k \leq n} \left[ \int \prod_{i=k}^n \frac{f_\theta(X_i)}{f_0(X_i)} dG(\theta) \right] \geq \frac{1}{\alpha} \right\}. \quad (6.74)$$

For the above choice of threshold, it is shown that

$$\text{FAR}(\tau_M) \leq \alpha,$$

and for each post-change parameter  $\theta \neq 0$ , if  $\theta$  is in a set over which  $G(\cdot)$  has a positive density, as the FAR constraint  $\alpha \rightarrow 0$ ,

$$\text{WADD}^\theta[\tau_M] \sim \frac{|\log \alpha|}{D(f_\theta, f_0)}.$$

Thus, even this mixture based test is uniformly asymptotically optimal with respect to the Lorden's criterion.

Although the GLR and mixture based tests discussed above are efficient, they do not have an equivalent recursive implementation, as the CuSum test or the Shiryaev-Roberts tests have when the post-change distribution is known. As a result, to implement the GLR or mixture based tests, we need to use the entire past information  $(X_1, \dots, X_n)$ , which grows with  $n$ . In [10], asymptotically optimal sliding-window based GLR and mixtures tests are proposed, that only used a finite number of past observations. The window size has to be chosen carefully and is a function of the FAR. Moreover, the results here are valid even for the non-i.i.d. setting discussed earlier in Section 3.06.4.3. Recall the non-i.i.d. model from Section 3.06.3.2, with the prechange conditional p.d.f. given by  $f_{0,n}(X_n | X_1^{n-1})$ , and the post-change conditional p.d.f. given by  $f_{1,n}(X_n | X_1^{n-1})$ . Then the generalized CuSum (6.68) GLR and mixture based algorithms are given, respectively, by:

$$\hat{\tau}_G = \inf \left\{ n \geq 1 : \max_{n-m_\alpha \leq k \leq n-m'_\alpha} \sup_\theta \left[ \sum_{i=k}^n \log \frac{f_{\theta,i}(X_i | X_1^{i-1})}{f_{0,i}(X_i | X_1^{i-1})} \right] \geq c_\alpha \right\} \quad (6.75)$$

and

$$\hat{\tau}_M = \inf \left\{ n \geq 1 : \max_{n-m_\alpha \leq k \leq n} \left[ \int \prod_{i=k}^n \frac{f_{\theta,i}(X_i | X_1^{i-1})}{f_{0,i}(X_i | X_1^{i-1})} dG(\theta) \right] \geq e^{c_\alpha} \right\}. \quad (6.76)$$

It is shown that for a suitable choice of  $c_\alpha$ ,  $m_\alpha$  and  $m'_\alpha$ , under some conditions on the likelihood ratios and on the distribution  $G$ , both of these tests satisfy the FAR constraint of  $\alpha$ . In particular,

$$\text{FAR}(\hat{\tau}_M) \leq \alpha$$

and as  $\alpha \rightarrow 0$ ,

$$\text{FAR}(\hat{\tau}_G) \leq \alpha(1 + o(1)).$$

Moreover, under some conditions on the post-change parameter  $\theta$ ,

$$\text{WADD}^\theta[\hat{\tau}_G] \sim \text{WADD}^\theta[\hat{\tau}_M] \sim \frac{|\log \alpha|}{D(f_\theta, f_0)}.$$

Thus, under the conditions stated, the above window-limited tests are also asymptotically optimal. We remark that, although finite window based tests are useful for the i.i.d. setting here, we still need to store the entire history of observations in the non-i.i.d. setting to compute the conditional densities, unless the dependence is finite order Markov. See [10, 28] for details.

To detect a change in mean of a sequence of Gaussian observations in an i.i.d. setting, i.e., when  $f_0 \sim \mathcal{N}(0, 1)$  and  $f_1 \sim \mathcal{N}(\mu, 1)$ ,  $\mu \neq 0$ , the GLR rule discussed above (6.75) (with  $m_\alpha = n$  and  $m'_\alpha = 1$ ) reduces to

$$\tau_v = \inf \left\{ n \geq 1 : \max_{0 \leq k < n} \left[ \sum_{i=k}^n \frac{|S_n - S_i|}{\sqrt{(n-k)}} \right] \geq b \right\}. \quad (6.77)$$

This test is studied in [29] and performance of the test, i.e., expressions for  $\text{FAR}(\tau_v)$  and  $\mathbb{E}_1[\tau_v]$  are obtained. In [28], it is shown that a window-limited modification of the above scheme (6.77) can also be designed.

When both pre- and post-change distributions are not known, again GLR based or mixture based tests have been studied and asymptotic properties characterized. In [30], this problem has been studied in the i.i.d setting (Bayesian and non-Bayesian), when both pre- and post-change distributions belong to an exponential family. For the Bayesian setting, the change point is assumed to be geometric and there are priors (again from an exponential family) on both the pre- and post-change parameters. GLR and mixture based tests are proposed that have asymptotic optimality properties. For a survey of other algorithms when both the pre- and post-change distributions are not known, see [28].

In [31], rather than developing procedures that are uniformly asymptotically optimal for each possible post-change distribution, robust tests are characterized when the pre- and post-change distributions belong to some uncertainty classes. The objective is to find a stopping time that satisfies a given false alarm constraint (probability of false alarm or the FAR depending on the formulation) for each possible pre-change distribution, and minimizes the detection delay (Shiryayev, Lorden or the Pollak version) supremized over each possible pair of pre- and post-change distributions. It is shown that under some conditions on the uncertainty classes, the *least favorable distribution* from each class can be obtained, and the robust test is the classical test designed according to the least favorable distribution pair. It is shown in [31] that although robust test is not uniformly asymptotically optimal, it can be significantly better than the GLR based test of [6] for some parameter ranges and for moderate values of FAR. The robust solution also has the added advantage that it can be implemented in a simple recursive manner, while the GLR test does not admit a recursive solution in general, and may require the solution to a complex nonconvex optimization problem at every time instant.

### 3.06.6.2 Data-efficient quickest change detection

In the classical problem of quickest change detection that we discussed in Section 3.06.3, a change in the distribution of a sequence of random variables has to be detected as soon as possible, subject to a constraint on the probability of false alarm. Based on the past information, the decision maker has to decide whether to stop and declare change or to continue acquiring more information. In many engineering applications of quickest change detection there is a cost associated with acquiring information or taking observations, e.g., cost associated with taking measurements, or cost of batteries in sensor networks, etc. (see [32] for a detailed motivation and related references). In [32], Shiryaev's formulation (Section 3.06.3) is extended by including an additional constraint on the cost of observations used in the detection process. The observation cost is captured through the average number of observations used before the change point, with the understanding that the cost of observations after the change point is included in the delay metric.

In order to minimize the average number of observations used before  $\Gamma$ , at each time instant, a decision is made on whether to use the observation in the next time step, based on all the available information. Let  $S_n \in \{0, 1\}$ , with  $S_n = 1$  if it is been decided to take the observation at time  $n$ , i.e.,  $X_n$  is available for decision making, and  $S_n = 0$  otherwise. Thus,  $S_n$  is an on-off (binary) control input based on the information available up to time  $k - 1$ , i.e.,

$$S_n = \mu_{k-1}(I_{k-1}), \quad k = 1, 2, \dots$$

with  $\mu$  denoting the control law and  $I$  defined as:

$$I_n = [S_1, \dots, S_n, X_1^{(S_1)}, \dots, X_n^{(S_n)}].$$

Here,  $X_i^{(S_i)}$  represents  $X_i$  if  $S_i = 1$ , otherwise  $X_i$  is absent from the information vector  $I_n$ .

Let  $\psi = \{\tau, \mu_0, \dots, \mu_{\tau-1}\}$  represent a policy for data-efficient quickest change detection, where  $\tau$  is a stopping time on the information sequence  $\{I_n\}$ . The objective in [32] is to solve the following optimization problem:

$$\begin{aligned} & \underset{\psi}{\text{minimize}} \quad \text{ADD}(\psi) = E[(\tau - \Gamma)^+] \quad (6.78) \\ & \text{subject to} \quad \text{PFA}(\psi) = P(\tau < \Gamma) \leq \alpha, \\ & \text{and} \quad \text{ANO}(\psi) = E \left[ \sum_{k=1}^{\min(\tau, \Gamma-1)} S_n \right] \leq \beta. \end{aligned}$$

Here, ADD, PFA and ANO stand for average detection delay, probability of false alarm and average number of observations used, respectively, and  $\alpha$  and  $\beta$  are given constraints.

Define,

$$p_n = \mathbb{P}(\Gamma \leq k | I_n).$$

Then, the two-threshold algorithm from [32] is:

**Algorithm 7 (DE-Shiryayev:  $\psi(A, B)$ ).** Start with  $p_0 = 0$  and use the following control, with  $B < A$ , for  $k \geq 0$ :

$$S_{k+1} = \mu_n(p_n) = \begin{cases} 0 & \text{if } p_n < B, \\ 1 & \text{if } p_n \geq B, \end{cases} \quad (6.79)$$

$$\tau = \inf \{k \geq 1 : p_n > A\}.$$

The probability  $p_n$  is updated using the following recursions:

$$p_{k+1} = \begin{cases} \tilde{p}_n = p_n + (1 - p_n)\rho & \text{if } S_{k+1} = 0, \\ \frac{\tilde{p}_n L(X_{k+1})}{\tilde{p}_n L(X_{k+1}) + (1 - \tilde{p}_n)} & \text{if } S_{k+1} = 1 \end{cases}$$

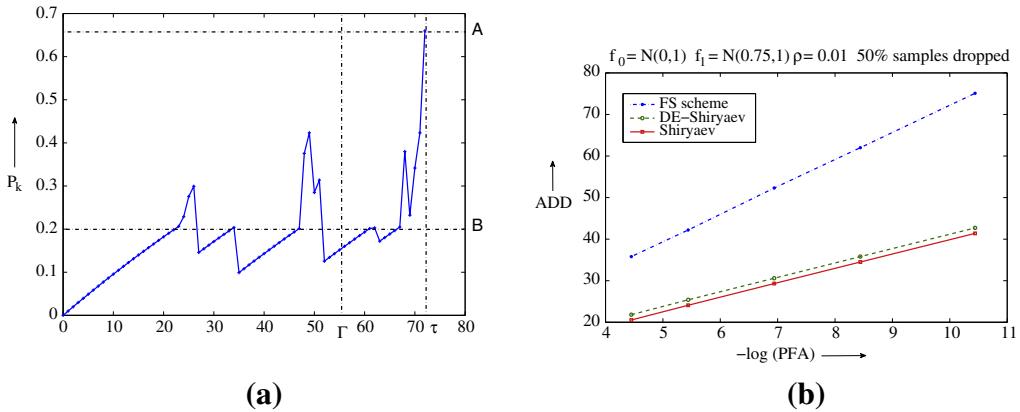
with  $L(X_{k+1}) = f_1(X_{k+1})/f_0(X_{k+1})$ .

With  $B = 0$  the DE-Shiryayev algorithm reduces to the Shiryaev algorithm. When the DE-Shiryayev algorithm is employed, the a posteriori probability  $p_n$  typically evolves as depicted in Figure 6.6a. It is shown in [32] that for a fixed  $\beta$ , as  $\alpha \rightarrow 0$ :

$$\text{ADD}(\psi(A, B)) \sim \frac{|\log(\alpha)|}{D(f_1, f_0) + |\log(1 - \rho)|} \text{ as } \alpha \rightarrow 0 \quad (6.80)$$

and

$$\text{PFA}(\psi(A, B)) \sim \alpha \left( \int_0^\infty e^{-x} dR(x) \right) \text{ as } \alpha \rightarrow 0. \quad (6.81)$$

**FIGURE 6.6**

Evolution and performance of the DE-Shiryaev algorithm. (a) Evolution of  $p_k$  for  $f_0 \sim N(0,1)$ ,  $f_1 \sim \mathcal{N}(0.75,1)$ , and  $\rho = 0.01$ , with thresholds  $A = 0.65$  and  $B = 0.2$ . (b) Trade-off curves comparing performance of the DE-Shiryaev algorithm with the Fractional Sampling scheme when  $B$  is chosen to achieve ANO equal to 50% of mean time to change. Also  $f_0 \sim N(0,1)$ ,  $f_1 \sim N(0.75,1)$ , and  $\rho = 0.01$ .

Here,  $R(x)$  is the asymptotic overshoot distribution of the random walk  $\sum_{k=1}^n [\log L(X_k) + |\log(1-\rho)|]$ , when it crosses a large positive boundary. It is shown in [12] that these are also the performance expressions for the Shiryaev algorithm. Thus, the PFA and ADD of the DE-Shiryaev algorithm approach that of the Shiryaev algorithm as  $\alpha \rightarrow 0$ , i.e., the DE-Shiryaev algorithm is asymptotically optimal.

The DE-Shiryaev algorithm is also shown to have good delay-observation cost trade-off curves: for moderate values of probability of false alarm, for Gaussian observations, the delay of the algorithm is within 10% of the Shiryaev delay even when the observation cost is reduced by more than 50%. Furthermore, the DE-Shiryaev algorithm is substantially better than the standard approach of *fractional sampling* scheme, where the Shiryaev algorithm is used and where the observations to be skipped are determined a priori in order to meet the observation constraint (see Figure 6.6b).

In most practical applications, prior information about the distribution of the change point is not available. As a result, the Bayesian solution is not directly applicable. For the classical quickest change detection problem, an algorithm for the non-Bayesian setting was obtained by taking the geometric parameter of the prior on the change point to zero (see Section 3.06.4.1). Such a technique cannot be used in the data-efficient setting. This is because when an observation is skipped in the DE-Shiryaev algorithm in [32], the *a posteriori* probability is updated using the geometric prior. In the absence of prior information about the distribution of the change point, it is by no means obvious what the right substitute for the prior is. A useful way to capture the cost of observations in a minimax setting is also needed.

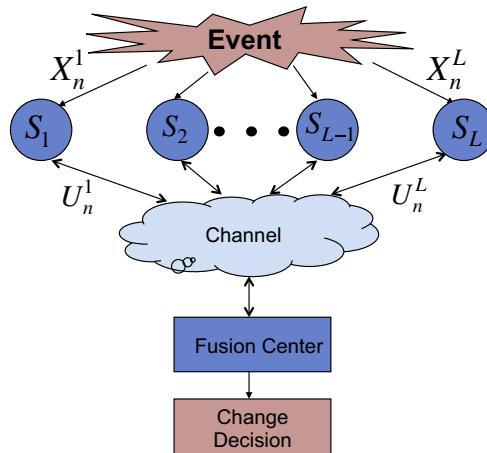
In [33], the minimax formulation of [9] is used to propose a minimax formulation for data-efficient quickest change detection. We observe that in the two-threshold algorithm  $\psi(A, B)$ , when the change occurs at a far horizon, it is the fraction of observations taken before change that is controlled. This insight

is used to formulate a duty cycle based metric to capture the cost of taking observations before the change point. Also, we note that the duration for which observations are not taken in the algorithm in [32], is also a function of the undershoot of the *a posteriori* probability when it goes below the threshold  $B$ . Given the fact that  $\frac{p_n}{1-p_n}$  for the DE-Shiryayev algorithm, has the interpretation of the likelihood ratio of the hypotheses “ $H_1 : \Gamma \leq n$ ” and “ $H_0 : \Gamma > n$ ,” the undershoots essentially carry the information on the likelihood ratio. It is shown in [33] that this insight can be used to design a good test in the non-Bayesian setting. This algorithm is called the DE-CuSum algorithm and it is shown that it inherits good properties of the DE-Shiryayev algorithm. The DE-CuSum algorithm is also asymptotically optimal in a sense similar to (6.80) and (6.81), has good trade-off curves, and performs significantly better than the standard approach of fractional sampling.

### 3.06.6.3 Distributed sensor systems

In the previous sections, we provided a summary of existing work on quickest change detection and classification in the single sensor (equivalently, centralized multisensor) setting. For the problem of detecting biological and chemical agents, the setting that is more relevant is one where there is a set of distributed sensors collecting the data relevant for detection, as shown in Figure 6.7. Based on the observations that the sensors receive, they send messages (which could be local decisions, but not necessarily) to a fusion center where a final decision about the hypothesis or change is made.

Since the information available for detection is distributed across the sensors in the network, these detection problems fall under the umbrella of distributed (or decentralized) detection, except in the impractical setting where all the information available at the sensors is immediately available without any errors at the fusion center. Such decentralized decision making problems are extremely difficult.



**FIGURE 6.7**

Change detection using distributed sensors.

Without certain conditional independence assumptions across sensors, the problem of finding the optimal solutions, even in the simplest case of static binary hypothesis testing, is computationally intractable [34–39]. Decentralized dynamic decision making problems, of which the quickest change detection problem is a special case, are even more challenging due to the fact that information pattern in these problems is *non-classical*, i.e., the different decision makers have access to different pieces of information [40].

The problem of decentralized quickest change detection in distributed sensor systems was introduced in [41], and is described as follows. Consider the distributed multisensor system with  $L$  sensors, communicating with a fusion center shown in Figure 6.7. At time  $n$ , an observation  $X_n^\ell$  is made at sensor  $\mathcal{S}_\ell$ . The changes in the statistics of the sequences  $\{X_n^\ell\}$  are governed by the event. Based on the information available at time  $n$ , a message  $U_n^\ell$  is sent from sensor  $\mathcal{S}_\ell$  to the fusion center. The fusion center may possibly feedback some control signals to the sensors based on all the messages it has received so far. For example, the fusion center might inform the sensors how to update their local decision thresholds. The final decision about the change is made at the fusion center.

There are two main approaches to generating the messages at the sensors. In the first approach, the sensors can be thought of simply quantizing their observations, i.e.,  $U_n^\ell$  is simply a quantized version of  $X_n^\ell$ . The goal then is to choose these quantizers over time and across the sensors, along with a decision rule at the fusion center, to provide the best tradeoff between detection delay and false alarms. In the second approach, the sensors perform local change detection, using all of their observations, and the fusion center combines these decisions to make the final decision about the change.

The simplest observation model for the decentralized setting is one where the sensors are affected by the change at the same time, and the observations are i.i.d. in time at each sensor and independent across sensors in both the pre-change and post-change modes. This model was introduced in [41], and studied in a Bayesian setting with a geometric prior on the change point for the case of quantization at the sensors. It was shown that, unlike the centralized problem for which the Shiryaev test is optimal (see Section 3.06.3), in the decentralized setting the optimization problem is intractable in even for this simple observation model. Some progress can be made if we allow for feedback from the fusion center [41]. Useful results can be obtained in the asymptotic setting where the probability of false (Bayesian formulation) or false alarm rate (minimax formulation) go to zero. These results can be summarized as follows (see, e.g., [12,42,43] for more details):

- It is asymptotically optimum for the sensors to use *stationary monotone likelihood ratio quantizers*, i.e., the sensors use the same quantization functions at all times, and the quantization regions are obtained by dividing the likelihood ratio of the observations into intervals and assigning the quantization levels to them in increasing order.
- The optimum quantization thresholds at the sensors are chosen to maximize the K-L divergence between the post-change and pre-change distributions at the sensors.
- For fixed stationary quantizers at the sensors, the fusion center is faced with a centralized quickest change detection problem. Therefore, depending on the optimization criterion (Bayes or minimax), asymptotically optimum change detection procedures can be designed using the techniques described in Sections 3.06.3 and 3.06.4
- The tradeoff between delay and false alarms is governed by the K-L divergence of the quantized observations at the output of the sensors, and hence the first order asymptotic performance with quantization is at best equal to that without quantization.

For the case where the sensors make local decisions about the change, it is reasonable to assume that the local detection procedures use (asymptotically) optimum centralized (single sensor) statistics. For example, in the Bayesian setting, the Shiryaev statistic described in Algorithm 1 can be used, and in the minimax setting one of the statistics described in Section 3.06.4.1 can be used depending on the specific minimax criterion used. The more interesting aspect of the decision-making here is the fusion rule used to combine the individual sensor decisions. There are three main basic options that can be considered [43]:

- $\tau_{\min}$ : the fusion center stops and declares the change as soon as one of the sensors' statistics crosses its decision threshold.
- $\tau_{\max}$ : the sensors stop taking observations once their local statistics cross their thresholds, and the fusion center stops and declares the change after all sensors have stopped.
- $\tau_{\text{all}}$ : the sensors continue taking observations even if their local statistics cross their thresholds, and the fusion center stops and declares the change after all the sensor statistics are above their local thresholds simultaneously.

It was first shown by [44] that the  $\tau_{\text{all}}$  procedure using CuSum statistics at the sensors is globally first order asymptotically optimal under Lorden's criterion (6.53) if the sensor thresholds are chose appropriately. That is, the first order performance is the same as that of the centralized procedure that has access to all the sensor observations. A more detailed analysis of minimax setting was carried out in [45], in which procedures based on using CuSum and Shiryaev-Roberts statistics at the sensors were studied under Pollak's criterion (6.55). It was again shown that  $\tau_{\text{all}}$  is globally first order asymptotically optimal, and that  $\tau_{\max}$  and  $\tau_{\min}$  are not.

For the Bayesian case, if the sensors use Shiryaev statistics, then both  $\tau_{\max}$  and  $\tau_{\text{all}}$  can be shown to be globally first order asymptotically optimal, with an appropriate choice of sensor thresholds [43, 46]. The procedure  $\tau_{\min}$  does not share this asymptotic optimality property.

Interestingly, while tests based on local decision making at the sensors can be shown to have the same first order performance as that of the centralized test, simulations reveal that these asymptotics "kick in" at unreasonably low values of false alarm probability (rate). In particular, even schemes based on *binary* quantization at the sensors can perform better than the globally asymptotically optimum local decision based tests at moderate values of false alarm probability (rate) [43]. These results point to the need for further research on designing procedures that perform local detection at the sensors that provide good performance at moderate levels of false alarms.

#### 3.06.6.4 Variants of quickest change detection problem for distributed sensor systems

In Section 3.06.6.3, it is assumed for the decentralized quickest change detection problem, that the change affects all the sensors in the system simultaneously. In many practical systems it is reasonable to assume that the change will be seen by only a subset of the sensors. This problem can be modeled as quickest change detection with unknown post-change distribution, with a finite number of possibilities. A GLRT based approached can of course be used, in which multiple CuSum tests are run in parallel, corresponding to each possible post-change hypotheses. But this can be quite expensive from an implementation view point. In [47], a CuSum based algorithm is proposed in which, at each sensor a CuSum test is employed,

the CuSum statistic is transmitted from the sensors to the fusion center, and at the fusion center, the CuSum statistics from all the sensors are added and compared with a threshold. This test has much lower computational complexity as compared to the GLR based test, and is shown to be asymptotically as good as the centralized CuSum test, as the FAR goes to zero. Although this test is asymptotically optimal, the noise from the sensors not affected by change can degrade the performance for moderate values of false alarm rate. In [48], this work of [47], is extended to the case where information is transmitted from the sensors only when the CuSum statistic at each sensor is above a certain threshold. It is shown that this has the surprising effect of suppressing the unwanted noise and improving the performance. In [49], it is proposed that a *soft-thresholding* function should be used to suppress these noise terms, and a GLRT based algorithm is proposed to detect presence of a stationary intruder (with unknown position) in a sensor network with Gaussian observations. A similar formulation is described in [50].

The Bayesian decentralized quickest change detection problem under an additional constraint on the cost of observations used is studied in [51]. The cost of observations is captured through the average number of observations used until the stopping time and it is shown that a threshold test similar to the Shiryaev algorithm is optimal. Recently, this problem has been studied in a minimax setting in [52] and asymptotically minimax algorithms have been proposed. Also, see [53, 54] for other interesting energy-efficient algorithms for quickest change detection in sensor networks.

---

### 3.06.7 Applications of quickest change detection

As mentioned in the introduction, the problem of quickest change detection has a variety of applications. A complete list of references to applications of quickest change detection can be quite overwhelming and therefore we only provide representative references from some of the areas. For a detailed list of references to application in areas such as climate modeling, econometrics, environment and public health, finance, image analysis, navigation, remote sensing, etc., see [13, 55].

1. *Statistical process control (SPC)*: As discussed in the introduction, algorithms are required that can detect a sudden fault arising in an industrial process or a production process. In recent years algorithms for SPC with sampling rate and sampling size control have also been developed to minimize the cost associated with sampling [56, 57]. See [58, 59] and the references therein for some recent contributions.
2. *Sensor networks*: As discussed in [32], quickest change detection algorithms can be employed in sensor networks for infrastructure monitoring [60], or for habitat monitoring [61]. Note that in these applications, the sensors are deployed for a long time, and the change may occur rarely. Therefore, data-efficient quickest change detection algorithms are needed (see Section 3.06.6.2).
3. *Computer network security*: Algorithms for quickest change detection have been applied in the detection of abnormal behavior in computer networks due to security breaches [62–64].
4. *Cognitive radio*: Algorithms based on the CuSum algorithm or other quickest change detection algorithms can be developed for cooperative spectrum sensing in cognitive radio networks to detect activity of a primary user. See [53, 65–67].
5. *Neuroscience*: The evolution of the Shiryaev algorithm is found to be similar to the dynamics of the *Leaky Integrate-and-Fire* model for neurons [68].

6. *Social networks:* It is suggested in [69, 70] that the algorithms from the change detection literature can be employed to detect the onset of the outbreak of a disease, or the effect of a bioterrorist attack, by monitoring drug sales at retail stores.

### 3.06.8 Conclusions and future directions

In this article we reviewed the state-of-the-art in the theory of quickest change detection. We saw that while exactly or nearly optimal algorithms are available only for the i.i.d. model and for the detection of a change in a single sequence of observations, asymptotically optimal algorithms can be obtained in a much broader setting. We discussed the uniform asymptotic optimality of GLR and mixture based tests, when the post-change distribution is not known. We discussed algorithms for data-efficient quickest change detection, and showed that they are also asymptotically equivalent to their classical counterparts. For the decentralized quickest change detection model, we discussed various algorithms that are asymptotically optimal. We also reviewed the asymptotic optimality theory in the Bayesian as well as in the minimax setting for a general non-i.i.d. model, and showed that extensions of the Shiryaev algorithm and the CuSum algorithm to the non-i.i.d. setting are asymptotically optimal. Nevertheless, the list of topics discussed in this article is far from exhaustive.

Below we enumerate possible future directions in which the quickest change detection problem can be explored. We also provide references to some recent articles in which some research on these topics has been initiated.

1. *Transient change detection:* It is assumed throughout this article that the change is persistent, i.e., once the change occurs, the system stays in the post-change state forever. In many applications it might be more appropriate to model change as *transient*, i.e., the system only stays in the post-change state for a finite duration and then returns to the pre-change state; see e.g., [71–73]. In this setting, in addition to false alarm and delay metrics, it may be of interest to consider metrics related to the detection of the change while the system is still in the change state.
2. *Change propagation:* In applications with multiple sensors, unlike the model assumed in Section 3.06.6.3, it may happen that the change does not affect all the sensors simultaneously [74]. The change may *propagate* from one sensor to the next, with the statistics of the propagation process being known before hand [75].
3. *Multiple change points in networks:* In some monitoring application there may be multiple change points that affect different sensors in a network, and the goal is to exploit the relationship between the change points and the sensors affected to detect the changes [76].
4. *Quickest change detection with social learning:* In the classical quickest change detection problem, to compute the statistic at each time step, the decision maker has access to the entire past observations. An interesting variant of the problem is quickest change detection with social learning, when the time index is replaced by an agent index, i.e., when the statistic is updated over agents and not over time, and the agents do not have access to the entire past history of observations but only to some processed version (e.g., binary decisions) from the previous agent; see [77–79].
5. *Change detection with simultaneous classification:* In many applications, the post-change distribution is not uniquely specified and may come from one of multiple hypotheses  $H_1, \dots, H_M$ ,

in which case along with detecting the change it of interest to identify which hypothesis is true. See, e.g., [80, 81].

6. *Synchronization issues:* If a quickest change detection algorithm is implemented in a sensor network where sensors communicate with the fusion center using a MAC protocol, the fusion center might receive asynchronous information from the sensors due to networking delays. It is of interest to develop algorithms that can detect changes while handling MAC layer issues; see, e.g., [50].

## Acknowledgments

The authors would like to thank Prof. Abdelhak Zoubir for useful suggestions, and Mr. Michael Fauss and Mr. Shang Kee Ting for their detailed reviews of an early draft of this work. The authors would also like to acknowledge the support of the National Science Foundation under Grants CCF 0830169 and CCF 1111342, through the University of Illinois at Urbana-Champaign, and the U.S. Defense Threat Reduction Agency through subcontract 147755 at the University of Illinois from prime award HDTRA1-10-1-0086.

*Relevant Theory:* Signal Processing Theory, and Machine Learning, Statistical Signal Processing

See Vol. 1, Chapter 11 Parametric Estimation

See Vol. 1, Chapter 12 Adaptive Filters

See Vol. 1, Chapter 18 Introduction to Probabilistic Graphical Models

See this Volume, Chapter 5 Distributed Signal Detection

## References

- [1] W.A. Shewhart, The application of statistics as an aid in maintaining quality of a manufactured product, *J. Am. Stat. Assoc.* 20 (1925) 546–548.
- [2] W.A. Shewhart, *Economic Control of Quality of Manufactured Product*, American Society for Quality Control, 1931.
- [3] E.S. Page, Continuous inspection schemes, *Biometrika* 41 (1954) 100–115.
- [4] A.N. Shiryaev, On optimum methods in quickest detection problems, *Theory Probab. Appl.* 8 (1963) 22–46.
- [5] A.N. Shirayev, *Optimal Stopping Rules*, Springer-Verlag, New York, 1978.
- [6] G. Lorden, Procedures for reacting to a change in distribution, *Ann. Math. Stat.* 42 (1971) 1897–1908.
- [7] G.V. Moustakides, Optimal stopping times for detecting changes in distributions, *Ann. Stat.* 14 (1986) 1379–1387.
- [8] Y. Ritov, Decision theoretic optimality of the CUSUM procedure, *Ann. Stat.* 18 (1990) 1464–1469.
- [9] M. Pollak, Optimal detection of a change in distribution, *Ann. Stat.* 13 (1985) 206–227.
- [10] T.L. Lai, Information bounds and quick detection of parameter changes in stochastic systems, *IEEE Trans. Inf. Theory* 44 (1998) 2917–2929.
- [11] A. Polunchenko, A.G. Tartakovsky, State-of-the-art in sequential change-point detection, *Methodol. Comput. Appl. Probab.* (2011) 1–36.
- [12] A.G. Tartakovsky, V.V. Veeravalli, General asymptotic Bayesian theory of quickest change detection, *SIAM Theory Probab. Appl.* 49 (2005) 458–497.
- [13] H.V. Poor, O. Hadjiliadis, *Quickest Detection*, Cambridge University Press, 2009.
- [14] Y.S. Chow, H. Robbins, D. Siegmund, *Great Expectations: The Theory of Optimal Stopping*, Houghton Mifflin, 1971.

- [15] A.G. Tartakovsky, I.V. Nikiforov, M. Basseville, Sequential Analysis: Hypothesis Testing and Change-Point Detection, Statistics, CRC Press, 2013.
- [16] D. Williams, Probability with Martingales, Cambridge Mathematical Textbooks, Cambridge University Press, 1991.
- [17] D. Siegmund, Sequential Analysis: Tests and Confidence Intervals, Springer Series in Statistics, Springer-Verlag, 1985.
- [18] M. Woodroofe, Nonlinear Renewal Theory in Sequential Analysis, CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, 1982.
- [19] D. Bertsekas, Dynamic Programming and Optimal Control, vols. I and II, Athena Scientific, Belmont, Massachusetts, 1995.
- [20] A.G. Tartakovsky, M. Pollak, A. Polunchenko, Third-order asymptotic optimality of the generalized Shiryaev-Roberts changepoint detection procedures, May, 2010, arXiv e-prints.
- [21] S.W. Roberts, A comparison of some control chart procedures, *Technometrics* 8 (1966) 411–430.
- [22] M. Pollak, A.G. Tartakovsky, Optimality properties of the Shiryaev-Roberts procedure, *Stat. Sinica* 19 (2009) 1729–1739.
- [23] G.V. Moustakides, A.S. Polunchenko, A.G. Tartakovsky, A numerical approach to performance analysis of quickest change-point detection procedures, *Stat. Sinica* 21 (2011) 571–596.
- [24] M. Pollak, Average run lengths of an optimal method of detecting a change in distribution, *Ann. Stat.* 15 (1987) 749–779.
- [25] A. Wald, J. Wolfowitz, Optimum character of the sequential probability ratio test, *Ann. Math. Stat.* 19 (3) (1948) 326–339.
- [26] G.V. Moustakides, Sequential change detection revisited, *Ann. Stat.* 36 (2008) 787–807.
- [27] M. Pollak, D. Siegmund, Approximations to the expected sample size of certain sequential tests, *Ann. Stat.* 3 (1975) 1267–1282.
- [28] T.L. Lai, Sequential changepoint detection in quality control and dynamical systems, *J. Roy. Stat. Soc. Suppl.* 57 (4) (1995) 613–658.
- [29] D. Siegmund, E.S. Venkatraman, Using the generalized likelihood ratio statistic for sequential detection of a change-point, *Ann. Stat.* 23 (1995) 255–271.
- [30] T.L. Lai, H. Xing, Sequential change-point detection when the pre- and post-change parameters are unknown, *Sequent. Anal.* 29 (2010) 162–175.
- [31] J. Unnikrishnan, V.V. Veeravalli, S.P. Meyn, Minimax robust quickest change detection, *IEEE Trans. Inf. Theory* 57 (2011) 1604–1614.
- [32] T. Banerjee, V.V. Veeravalli, Data-efficient quickest change detection with on-off observation control, *Sequent. Anal.* 31 (2012) 40–77.
- [33] T. Banerjee, V.V. Veeravalli, Data-efficient minimax quickest change detection, in: IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP), March 2012, pp. 3937–3940.
- [34] J.N. Tsitsiklis, Decentralized detection, in: H.V. Poor, J.B. Thomas (Eds.), Advances in Statistical Signal Processing, vol. 2, JAI Press, Greenwich, CT, 1993.
- [35] P.K. Varshney, Distributed Detection and Data Fusion, Springer-Verlag, New York, 1996.
- [36] P. Willett, P.F. Swaszek, R.S. Blum, The good, bad and ugly: distributed detection of a known signal in dependent Gaussian noise, *IEEE Trans. Signal Process.* 48 (2000) 3266–3279.
- [37] J.F. Chamberland, V.V. Veeravalli, Decentralized detection in sensor networks, *IEEE Trans. Signal Process.* 51 (2003) 407–416.
- [38] J.-F. Chamberland, V.V. Veeravalli, Wireless sensors in distributed detection applications, *IEEE Signal Process. Mag.* 24 (2007) 16–25 (special issue on Resource-Constrained Signal Processing, Communications, and Networking).

- [39] V.V. Veeravalli, P.K. Varshney, Distributed inference in wireless sensor networks, *Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci.* 370 (2012) 100–117.
- [40] Y. Ho, Team decision theory and information structures, *Proc. IEEE* 68 (1980) 644–654.
- [41] V.V. Veeravalli, Decentralized quickest change detection, *IEEE Trans. Inf. Theory* 47 (2001) 1657–1665.
- [42] A.G. Tartakovsky, V.V. Veeravalli, Change-point detection in multichannel and distributed systems, in: N. Mukhopadhyay, S. Datta, S. Chattopadhyay (Eds.), *Applied Sequential Methodologies: Real-World Examples with Data Analysis, Statistics: A Series of Textbooks and Monographs*, vol. 173, Marcel Dekker, Inc., New York, USA, 2004, pp. 339–370.
- [43] A.G. Tartakovsky, V.V. Veeravalli, Asymptotically optimal quickest change detection in distributed sensor systems, *Sequent. Anal.* 27 (2008) 441–475.
- [44] Y. Mei, Information bounds and quickest change detection in decentralized decision systems, *IEEE Trans. Inf. Theory* 51 (2005) 2669–2681.
- [45] A.G. Tartakovsky, H. Kim, Performance of certain decentralized distributed change detection procedures, in: *IEEE International Conference on Information Fusion*, Florence, Italy, July 2006, pp. 1–8.
- [46] A.G. Tartakovsky, V.V. Veeravalli, Quickest change detection in distributed sensor systems, in: *IEEE International Conference on Information Fusion*, Cairns, Australia, July 2003, pp. 756–763.
- [47] Y. Mei, Efficient scalable schemes for monitoring a large number of data streams, *Biometrika* 97 (2010) 419–433.
- [48] Y. Mei, Quickest detection in censoring sensor networks, in: *IEEE International Symposium on Information Theory (ISIT)*, August 2011, pp. 2148–2152.
- [49] Y. Xie, D. Siegmund, Multi-sensor change-point detection, in: *Joint Statistical Meetings*, August 2011.
- [50] K. Premkumar, A. Kumar, J. Kuri, Distributed detection and localization of events in large ad hoc wireless sensor networks, in: *Allerton Conference on Communication, Control, and Computing*, October 2009, pp. 178–185.
- [51] K. Premkumar, A. Kumar, Optimal sleep-wake scheduling for quickest intrusion detection using wireless sensor networks, in: *IEEE Conference on Computer Communications (INFOCOM)*, April 2008, pp. 1400–1408.
- [52] T. Banerjee, V.V. Veeravalli, Energy-efficient quickest change detection in sensor networks, in: *IEEE Statistical Signal Processing Workshop*, August 2012.
- [53] T. Banerjee, V. Sharma, V. Kavitha, A.K. JayaPrakasam, Generalized analysis of a distributed energy efficient algorithm for change detection, *IEEE Trans. Wireless Commun.* 10 (2011) 91–101.
- [54] L. Zacharias, R. Sundaresan, Decentralized sequential change detection using physical layer fusion, *IEEE Trans. Wireless Commun.* 7 (2008) 4999–5008.
- [55] M. Basseville, I.V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, Prentice Hall, Englewood Cliffs, NJ, 1993.
- [56] G. Tagaras, A survey of recent developments in the design of adaptive control charts, *J. Qual. Technol.* 30 (1998) 212–231.
- [57] Z.G. Stoumbos, M.R. Reynolds, T.P. Ryan, W.H. Woodall, The state of statistical process control as we proceed into the 21st century, *J. Am. Stat. Assoc.* 95 (2000) 992–998.
- [58] Z.G. Stoumbos, M.R. Reynolds, Economic statistical design of adaptive control schemes for monitoring the mean and variance: an application to analyzers, *Nonlinear Anal. Real World Appl.* 6 (2005) 817–844.
- [59] V. Makis, Multivariate bayesian control chart, *Oper. Res.* 56 (2008) 487–496.
- [60] J.A. Rice, K. Mechitov, S. Sim, T. Nagayama, S. Jang, R. Kim, B.F. Spencer, G. Agha, Y. Fujino, Flexible smart sensor framework for autonomous structural health monitoring, *Smart Struct. Syst.* 6 (5–6) (2010) 423–438.
- [61] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, J. Anderson, Wireless sensor networks for habitat monitoring, in: *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, WSNA '02, New York, NY, USA, ACM, September 2002, pp. 88–97.

- [62] M. Thottan, C. Ji, Anomaly detection in IP networks, *IEEE Trans. Signal Process.* 51 (2003) 2191–2204.
- [63] A.G. Tartakovsky, B.L. Rozovskii, R.B. Blazek, H. Kim, A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods, *IEEE Trans. Signal Process.* 54 (2006) 3372–3382.
- [64] A.A. Cardenas, S. Radosavac, J.S. Baras, Evaluation of detection algorithms for MAC layer misbehavior: theory and experiments, *IEEE/ACM Trans. Network.* 17 (2009) 605–617.
- [65] L. Lai, Y. Fan, H.V. Poor, Quickest detection in cognitive radio: A sequential change detection framework, in: *IEEE GLOBECOM*, December 2008, pp. 1–5.
- [66] A.K. Jayaprakasam, V. Sharma, Cooperative robust sequential detection algorithms for spectrum sensing in cognitive radio, in: *International Conference on Ultramodern Telecommunications (ICUMT)*, October 2009, pp. 1–8.
- [67] A.K. Jayaprakasam, V. Sharma, Sequential detection based cooperative spectrum sensing algorithms in cognitive radio, in: *First UK-India International Workshop on Cognitive Wireless Systems (UKIWCWS)*, December 2009, pp. 1–6.
- [68] A.J. Yu, Optimal change-detection and spiking neurons, in: B. Schölkopf, J. Platt, T. Hoffman (Eds.), *Advances in Neural Information Processing Systems*, vol. 19, MIT Press, Cambridge, MA, 2007, pp. 1545–1552.
- [69] M. Frisen, Optimal sequential surveillance for finance, public health, and other areas, *Sequent. Anal.* 28 (2009) 310–337.
- [70] S.E. Fienberg, G. Shmueli, Statistical issues and challenges associated with rapid detection of bio-terrorist attacks, *Stat. Med.* 24 (2005) 513–529.
- [71] C. Han, P.K. Willett, D.A. Abraham, Some methods to evaluate the performance of Page's test as used to detect transient signals, *IEEE Trans. Signal Process.* 47 (1999) 2112–2127.
- [72] Z. Wang, P.K. Willett, A performance study of some transient detectors, *IEEE Trans. Signal Process.* 48 (2000) 2682–2685.
- [73] K. Premkumar, A. Kumar, V.V. Veeravalli, Bayesian quickest transient change detection, in: *International Workshop on Applied Probability (IWAP)*, July 2010.
- [74] O. Hadjiliadis, H. Zhang, H.V. Poor, One shot schemes for decentralized quickest change detection, *IEEE Trans. Inf. Theory* 55 (2009) 3346–3359.
- [75] V. Raghavan, V.V. Veeravalli, Quickest change detection of a Markov process across a sensor array, *IEEE Trans. Inf. Theory* 56 (2010) 1961–1981.
- [76] X. Nguyen, A. Amini, R. Rajagopal, Message-passing sequential detection of multiple change points in networks, in: *IEEE International Symposium on Information Theory (ISIT)*, July 2012.
- [77] V. Krishnamurthy, Quickest time change detection with social learning, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, March 2012, pp. 5257–5260.
- [78] V. Krishnamurthy, Bayesian sequential detection with phase-distributed change time and nonlinear penalty; a POMDP lattice programming approach, *IEEE Trans. Inf. Theory* 57 (2011) 7096–7124.
- [79] V. Krishnamurthy, Quickest detection with social learning: interaction of local and global decision makers, March 2012, arXiv e-prints.
- [80] I.V. Nikiforov, A lower bound for the detection/isolation delay in a class of sequential tests, *IEEE Trans. Inf. Theory* 49 (2003) 3037–3046.
- [81] A.G. Tartakovsky, Multidecision quickest change-point detection: previous achievements and open problems, *Sequent. Anal.* 27 (2008) 201–231.

# Geolocation—Maps, Measurements, Models, and Methods

7

Fredrik Gustafsson

*Division of Automatic Control, Department of Electrical Engineering, Linköping University, Sweden*

## 3.07.1 Introduction

Geolocation is commonly used to describe the position of an object as a geographical location. It is also used to denote the process of how to compute a geolocation. It is closely related to the broader concepts of position and positioning/localization/navigation, but with a focus on a particular geographical context.

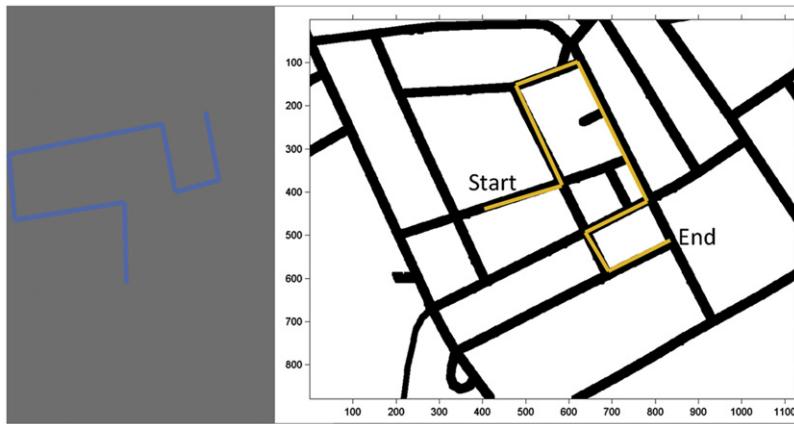
Target tracking is an area which has much in common with geolocation. Also here the position is delivered relative to a sensor network. The main difference is that the position is computed in the network. We exclude such *network-centric* applications, and focus on *user-centric* solutions where the network does not need to cooperate with the user.

The application areas where the phrase geolocation is used in literature, include the position of the following objects:

- Sensors, such as radars in a sensor network. Sensor geolocation is a pre-requisite in all target tracking and surveillance systems.
- Radio receivers and transmitters, such as cell phones in a cellular network. Geolocation of cell phones is required in some countries for emergency response. It is also a basis for many location based services, and an important tool for the operators to optimize the network.
- Animals, such as migrating birds. Besides being an important branch of ecology, the study of long-term migration is also important for understanding the spread of infectious disease risk [1].
- Computers in the Internet.

We will here cover the first three cases, but exclude the last item on geolocation of IP hosts, which significantly differs from the other ones, see [2,3]. We will exclude positioning methods that only provide a set of coordinates such as longitude and latitude on a globe. This rules out satellite based localization and terrestrial navigation, the latter based on the position of planets and stars [4]. In summary, our definition of geolocation is as follows:

Positioning—the process of computing a position as coordinates on a flat surface or a sphere.  
*Geolocation—position or positioning in a geographical context.*

**FIGURE 7.1**

Dynamic fingerprint in road networks, where a sufficiently long trajectory provides a unique fingerprint in the road map.

The geographical context is provided by what is usually referred to as a *geographical information system* (GIS) in general terms. We will simply use the term *map*. The sensors typically measure range, angle or orientation to *landmarks* in the map.

The geographical (static or dynamic) fingerprint is a key concept introduced here to distinguish different applications. The term fingerprint of course comes from human identification, and in this context the human retina also provides a unique “fingerprint” of each human. On the contrary, one sample of speech is not sufficient for identifying the speaker, but a sequence of speech samples provides a “dynamic fingerprint.” We extend this example to geographical fingerprints for geolocation. One or a sequence of ranges/bearings to given landmarks in the map forms a static or dynamic fingerprint. Some examples:

- Geolocation of road-bound vehicles using the driven path as a dynamic fingerprint, which is fitted to a road map. Figure 7.1 illustrates the principle.
- Geolocation of airborne platforms using a measured terrain profile as a dynamic fingerpring, which is fitted to a terrain elevation map.
- Geolocation of underwater vessels using a measured seafloor profile similar to the case above.
- Geolocation of surface vessels using the shore profile from a scanning radar as a dynamic fingerprint, which is compared to a sea chart.
- Local variations on the earth magnetic field provides a map to which magnetometer measurements can be mapped in a fingerprint manner [5].
- A lightlogger [6,7] mounted on an animal can detect sunset and sunrise. Each detection corresponds to a one-dimensional manifold of position on earth, and two such detections from sunset and sunrise have a unique intersection (except twice a year). This information can be merged with maps of possible resting areas for the bird (excluding water for instance), to form a geolocation estimate.
- Water animals can be gelocalized by logging the water temperature and pressure [7].

- At each point on earth, radio waves from a number of stationary radio transmitters (cellular networks, television, etc.) can be detected. The *received signal strength* (RSS) is one radio measurement [8] that can be used to form a fingerprint in the following two conceptually different ways:
  - The vector of RSS measurements from available transmitters forms a static fingerprint which is more or less unique for each point, provided that such an RSS map is available. The advantage is that the method is completely independent of the deployment of transmitters, and the drawback is that it requires a separate mapping procedure in advance. The location service in Google Maps is a good example to show the flexibility and strength of this approach.
  - If the position and emitted power of each transmitter are known, then generic path loss models can be used to compute the fingerprint, without the need for a map. This is a common principle for how the cellular phone operators implement their localization algorithms.

RSS based geolocation is particularly challenging for indoor applications [9–11].

Note that a geolocation is in many cases of higher relevance to the user than the exact position given in latitude and longitude. Take for instance road navigation. A road map is seldom correct, so the actual true position might be 10–30 m off the road network. Thus, the true position would be quite confusing information for the user.

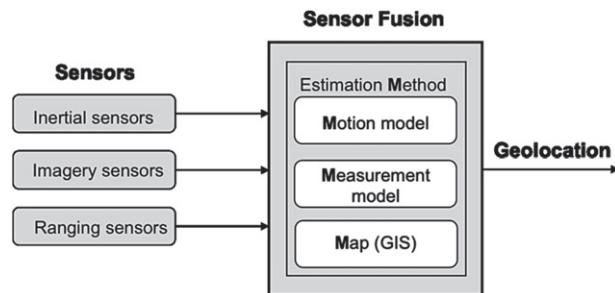
The discussion and examples above can be summarized as follows:

A sequence of measurements, related to landmarks in a map, along a trajectory forms a *dynamic fingerprint*.

### 3.07.2 Theory—overview

Figure 7.2 shows the main blocks in a geolocation system. The concepts and main function of each block in Figure 7.2 can be summarized as follows:

- Sensors:
  - The inertial sensors indicate how the object is moving, for instance by measuring the speed and course changes.
  - The imagery sensors can be used to recognize landmarks, and they provide primarily the angle to the landmark, but also orientation and distance if the landmark is sufficiently well known. A camera is a simple example.
  - The ranging sensors provide range to landmarks. A radar is the typical example, but also time of flight and received signal strength (RSS) of radio signals are commonly used.
  - There are sensors that provide both range and angle, such as stereo cameras. An antenna array of radio receivers also provides both range (from time of flight) and angle.
- The motion model is used to predict how the object is moving over time. For instance, odometric sensor data can be integrated to a trajectory. This procedure is known as *dead-reckoning* or *path integration*.

**FIGURE 7.2**

Framework of geolocation. The important and distinguishing features are the kind of sensors, the sensor model providing a geographic fingerprint, the motion model describing how the object moves in time, the map and the state estimation algorithm.

- The map contains the landmarks that the sensors can detect.
- The sensor model connects the range and bearing measurements to landmarks in the map. The sensor model is a function of the position of the object.
- The state estimation method computes an estimate of the state vector, given all models and information above. The state includes at least the position of the object, and possibly also speed and course.

Geographical information systems (GIS) represent our world with:

- Landmarks stored as point objects.
- Lines and polylines to represent topography, rivers, roads, railways, etc.
- Areas stored as polygons to represent land use, lakes, islands, city boundaries, etc.

All of these GIS data may be used to define the map to be used for geolocation. The examples in Section 3.07.5 use land and water depth topography, land use and coast line.

The following sections will describe each block in more detail, with some concrete examples. The description is primarily verbal, but in parallel some detailed mathematical formulas will be provided, to show what kind of computations that are involved and perhaps stimulate interested readers to make their own implementations. The following section sets up the mathematical estimation framework, and outlines the most important methods. The reader not interesting in this more theoretical part can skip this section, or only read the more verbal description of the theory.

### 3.07.3 Estimation methods

#### 3.07.3.1 Mathematical framework

The mathematical framework can be summarized in a state space model with state  $x_k$ , position dependent measurement  $y_k$ , input signal  $u_k$ , process noise  $w_k$ , and measurement noise  $e_k$ :

$$x_{k+1} = f(x_k, u_k, w_k), \quad (7.1a)$$

$$y_k = h(x_k) + e_k. \quad (7.1b)$$

The state includes at least position ( $X_k, Y_k$ ) and possibly also heading (or course)  $\psi_k$  and speed  $v_k$ . In our geolocation framework, (7.1a) corresponds to the motion model and (7.1b) the measurement model. In summary, the measurement model (7.1b) describes how the measurement  $y_k$  relates to the map  $h(x_k)$  and sensor errors  $e_k$ , while the motion model (7.1a) describes how the next state  $x_{k+1}$  depends on the previous state  $x_k$  and the measured system inputs  $u_k$  and unmeasured system inputs  $w_k$ . State estimation is the problem of estimating  $x_k$  from the measurements  $y_k$ . A dynamic fingerprint shows how a sequence of  $L$  measurements  $y_{k-L+1:k} = (y_{k-L+1}^T, \dots, y_k^T)^T$  fits the map for the trajectory  $x_{k-L+1:k} = (x_{k-L+1}^T, \dots, x_k^T)^T$ .

The following subsections will substantiate the mathematical framework.

### 3.07.3.2 Nonlinear filtering

The problem of estimating the state, given a dynamical model for the time evolution of the state and a measurement model showing how the measurements relate to the state, is called filtering. When one or more models are nonlinear, the problem is called *nonlinear filtering*. The theory has developed during the past 50 years, and today there are many powerful methods available. The output of a nonlinear filter is a conditional distribution of the state, and the different classes of algorithms can be classified according to how they parametrize this distribution, see Figure 7.3.

- Kalman filter variants: the state is represented with a Gaussian distribution.
- Kalman filter banks based on multiple model approaches: the state is represented with a mixture of Gaussian distributions, where each Gaussian mode has an associated weight.
- Point mass and particle filters: the state is represented with a set of grid points or samples with an associated weight.
- Marginalized, or Rao-Blackwellized, particle filters: the state is represented with a number of trajectories over the geolocation area, each one has an associated weight *and* Gaussian distribution for the other state variables than position.

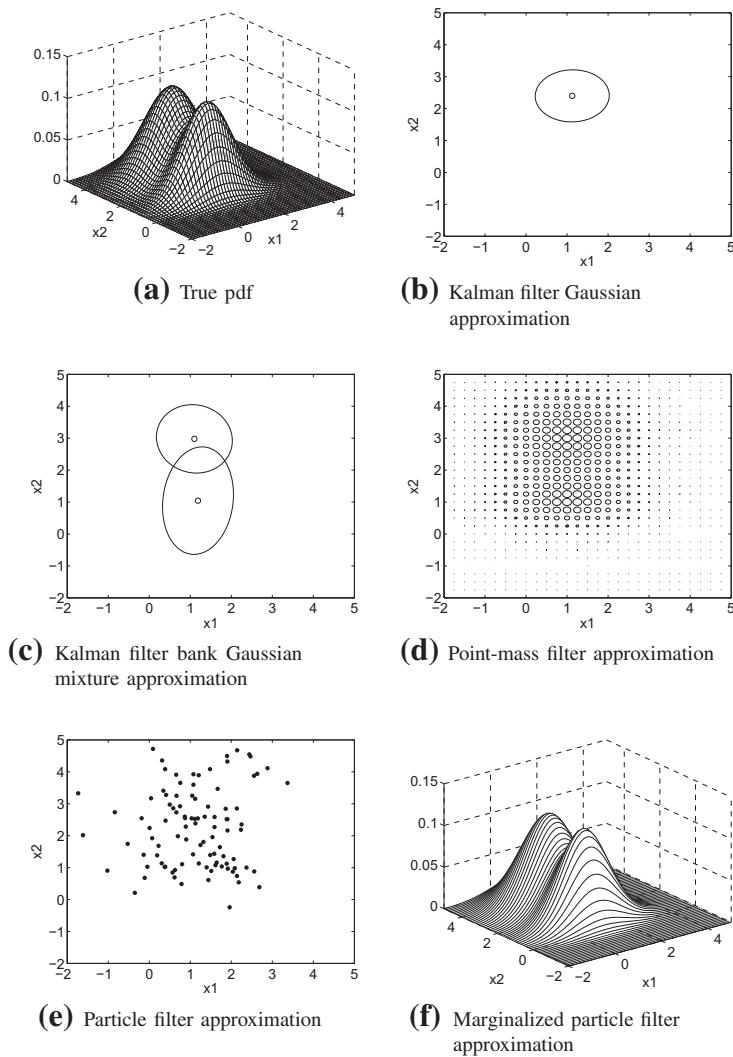
We will here focus on the last one, since it focuses on estimating trajectories, and for that reason it is the approach that best fits the dynamic fingerprinting concept.

### 3.07.3.3 Nonlinear filter theory

#### 3.07.3.3.1 Bayes optimal filter

The Bayesian approach to nonlinear filtering represents the information in the observations in a *posterior distribution*  $p(x_k|y_{1:k})$ . This can be seen as a conditional probability density function (PDF) of the current state  $x_k$  given all previous observations  $y_{1:k}$ . The Bayesian optimal filter propagates the posterior distribution via a time update and a measurement update, respectively,

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})}, \quad (7.2)$$

**FIGURE 7.3**

True probability density function and different approximate representations.

$$p(x_{k+1}|y_{1:k}) = \int_{\mathcal{R}^n} p(x_{k+1}|x_k) p(x_k|y_{1:k}) dx_k. \quad (7.3)$$

The derivation of this recursion is quite straightforward and based on Bayes law and the marginalization theorem, see Chapter 6 in [12] for instance.

### 3.07.3.3.2 Mean and covariance

The mean and covariance are defined as

$$\bar{x}_{k|k} = \int x_k p(x_k | y_{1:k}) dx_k, \quad (7.4)$$

$$\bar{P}_{k|k} = \int (x_k - \bar{x}_{k|k}) (x_k - \bar{x}_{k|k})^T p(x_k | y_{1:k}) dx_k. \quad (7.5)$$

The double index notation in  $\bar{x}_{k|k}$  should be interpreted as the estimate of the stochastic variable  $x_k$  given observations up to time  $k$ . The interpretation for  $\bar{P}_{k|k}$  is the same. With this convention,  $\bar{x}_{k|k-1}$  is the expected value of the state prediction of  $x_k$  and  $\bar{x}_{k|N}$  is the expected value of the state  $x_k$  given observations  $y_{1:N}$ , which is the smoothing solution for  $N > k$  and general prediction solution for  $N < k$ .

An estimator provides approximations of the mean and covariance. These will be denoted  $\hat{x}_{k|k}$  and  $P_{k|k}$ , respectively.

Covariance has a natural interpretation as providing a confidence area for geolocation. This area is represented with an ellipsoid centered around the mean where there is a, say, 99% probability to find the true position (note that the true position is considered to be a random variable in the Bayesian approach).

### 3.07.3.3.3 Covariance bound

In order to provide a confidence ellipsoid to a geolocation problem, we need first to get data, then compute the posterior PDF with a specific method (which might involve approximations) and from this compute the mean and covariance. An alternative is provided by the Cramer-Rao Lower Bound (CRLB). This gives a lower bound on the covariance matrix

$$P_{k|k} \geq P_{k|k}^{\text{CRLB}}. \quad (7.6)$$

This bound applies to all methods that give an unbiased state estimate. There are two versions of this bound with a bit different interpretations and computational complexity:

- The parametric CRLB  $P_{k|k}^{\text{parCRLB}}$ , where the true state sequence  $x_{1:k}^o$  is assumed to be known. The bound is then a function of this sequence  $P_{k|k}^{\text{parCRLB}}(x_{1:k}^o)$ . This bound is simple to compute, and the formulas are closely related to the covariance update in the (extended) Kalman filter, with a measurement update

$$H_k = \left. \frac{dh(x)}{dx} \right|_{x=x_k^o}, \quad (7.7a)$$

$$P_{k|k} = P_{k|k-1} - P_{k|k-1} H_k^T \left( H_k P_{k|k-1} H_k^T + R_k \right)^{-1} H_k P_{k|k-1}, \quad (7.7b)$$

and time update

$$F_k = \left. \frac{df(x, u, w)}{dx} \right|_{x=x_k^o, u=u_k, w=0}, \quad (7.7c)$$

$$G_{w,k} = \left. \frac{df(x, u, w)}{dw} \right|_{x=x_k^o, u=u_k, w=0}, \quad (7.7d)$$

$$P_{k+1|k} = F_k P_{k|k} F_k^T + G_{w,k} Q_k G_{w,k}^T. \quad (7.7e)$$

- The posterior CRLB  $P_{k|k}^{\text{postCRLB}}$ , where the true state sequence is assumed to be random according to the Bayesian paradigm (this version is sometimes referred to as the Bayesian CRLB for this reason). The relation between the bounds can be written

$$P_{k|k}^{\text{postCRLB}} = \int P_{k|k}^{\text{parCRLB}}(x_{1:k}) p(x_{1:k}) dx_{1:k}, \quad (7.8)$$

where  $p(x_{1:k})$  is the prior of the state sequence. This high-dimensional integral makes this bound harder to compute, however, there are recursive algorithms, see [13, 14].

### 3.07.3.3.4 The Kalman filter

The posterior (7.2) in the Bayes optimal filter is in general not possible to compute or even represent in a computer. Instead, a common solution is to update the mean and covariance rather than the full PDF. This coincides with the optimal Bayes filter only for a linear Gaussian model

$$x_{k+1} = F_k x_k + G_{u,k} u_k + G_{w,k} w_k, \quad (7.9a)$$

$$y_k = H_k x_k + D_k u_k + e_k, \quad (7.9b)$$

in which case the Kalman filter provides the mean and covariance in a Gaussian posterior  $p(x_k | y_{1:k}) = \mathcal{N}(\hat{x}_{k|k}, P_{k|k})$ . The measurement and time update, respectively, are given by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + P_{k|k-1} H_k^T \left( H_k P_{k|k-1} H_k^T + R_k \right)^{-1} (y_k - H_k \hat{x}_{k|k-1} - D_k u_k), \quad (7.10a)$$

$$P_{k|k} = P_{k|k-1} - P_{k|k-1} H_k^T \left( H_k P_{k|k-1} H_k^T + R_k \right)^{-1} H_k P_{k|k-1}, \quad (7.10b)$$

$$\hat{x}_{k+1|k} = F_k \hat{x}_{k|k} + G_{u,k} u_k, \quad (7.10c)$$

$$P_{k+1|k} = F_k P_{k|k} F_k^T + G_{w,k} Q_k G_{w,k}^T. \quad (7.10d)$$

This can be seen as a special case of Algorithm 1. The derivation, variations and properties are described in any text book on Kalman filtering, for instance Chapter 7 in [12].

### 3.07.3.4 The extended Kalman filter

The Kalman filter recursions can be used for nonlinear filtering problems by applying a first order Taylor expansion of the model,

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, w_k) \\ &\approx f(\hat{x}_{k|k}, u_k, 0) + f'_x(\hat{x}_{k|k}, u_k, 0)(x - \hat{x}_{k|k}) + f'_w(\hat{x}_{k|k}, u_k, 0)w_k, \\ y_k &= h(\hat{x}_{k|k-1}) + h'_x(\hat{x}_{k|k-1})(x - \hat{x}_{k|k-1}) + e_k. \end{aligned}$$

By re-arranging these equations, it can be seen as a linear model (7.10) with some additional terms in the right-hand side that do not depend on the state  $x_k$ . Thus, the Kalman filter is easily modified for this approximate model. The resulting approximation of the Bayes optimal filter is called the *extended Kalman filter*.

The extended Kalman filter consists of the following main steps:

- Define an initial Gaussian distribution  $\mathcal{N}(\hat{x}_{1|0}, P_{1|0})$  for the state  $x_1$  before the first observation  $y_1$  is provided.
- Iterate in  $k = 1, 2, \dots$ :
  1. Measurement update: use a Taylor approximation of (7.1b) and the analytical solution provided by the Kalman filter to get  $\mathcal{N}(\hat{x}_{k|k}, P_{k|k})$  from  $\mathcal{N}(\hat{x}_{k|k-1}, P_{k|k-1})$ .
  2. Time update: use a Taylor approximation of (7.1a) and a simple formula from multivariate statistics to get  $\mathcal{N}(\hat{x}_{k+1|k}, P_{k+1|k})$  from  $\mathcal{N}(\hat{x}_{k|k}, P_{k|k})$ .

See Algorithm 1 for the details.

More details are given in for instance Chapter 8 in [12].

### 3.07.3.5 The unscented Kalman filter

The EKF takes care of the first two terms in a Taylor expansion of the model. The unscented Kalman filter (UKF) makes an attempt to also include the information in the quadratic term. Consider the following nonlinear mapping  $z = g(x)$  of a Gaussian vector  $x$ ,

$$\begin{aligned} x &\in \mathcal{N}(\mu_x, P_x), \\ z = g(x) &\approx \mathcal{N}(\mu_z, P_z). \end{aligned}$$

---

#### Algorithm 1. The Extended Kalman Filter

---

*Given:* Motion model (7.1a), measurement model (7.1b), the noise covariances  $Q = \text{Cov}(w)$  and  $R = \text{Cov}(e)$ , respectively, and the prior mean  $x_0$  and covariance  $P_0$ .

*Design parameter:* None. *Initialization:* Let  $\hat{x}_{1|0} = x_0$  and  $P_{1|0} = P_0$

*Iteration:* For  $k = 1, 2, \dots$

**1. Measurement update:**

$$H_k = \left. \frac{dh(x)}{dx} \right|_{x=\hat{x}_{k|k-1}}, \quad (7.11a)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + P_{k|k-1} H_k^T \left( H_k P_{k|k-1} H_k^T + R_k \right)^{-1} (y_k - h(\hat{x}_{k|k-1})), \quad (7.11b)$$

$$P_{k|k} = P_{k|k-1} - P_{k|k-1} H_k^T \left( H_k P_{k|k-1} H_k^T + R_k \right)^{-1} H_k P_{k|k-1}. \quad (7.11c)$$

**2. Time update:**

$$F_k = \left. \frac{df(x, u, w)}{dx} \right|_{x=\hat{x}_{k|k}, u=u_k, w=0}, \quad (7.11e)$$

$$G_{w,k} = \left. \frac{df(x, u, w)}{dw} \right|_{x=\hat{x}_{k|k}, u=u_k, w=0}, \quad (7.11f)$$


---

$$\hat{x}_{k+1|k} = f(\hat{x}_{k|k}, u_k, 0), \quad (7.11e)$$

$$P_{k+1|k} = F_k P_{k|k} F_k^T + G_{w,k} Q_k G_{w,k}^T. \quad (7.11f)$$

The unscented transform generates samples of  $x^{(i)}$ , called *sigma points*, systematically on the surface of an ellipsoid centered around its mean, maps these points to  $z^{(i)} = g(x^{(i)})$ , and then computes the first two moments of the samples  $z^{(i)}$ . The procedure is quite similar to a Monte Carlo simulations, but there are two important differences: (1) the samples are deterministically chosen and (2) the weights in the mean and covariance estimators are not simply  $1/N$ , where  $N$  is the number of samples.

The following equations summarize the unscented transform. Let

$$x^{(0)} = \mu_x, \quad x^{(\pm i)} = \mu_x \pm \sqrt{n_x + \lambda} \sigma_i u_i, \quad (7.12a)$$

$$\omega^{(0)} = \frac{\lambda}{n_x + \lambda}, \quad \omega^{(\pm i)} = \frac{1}{2(n_x + \lambda)}, \quad (7.12b)$$

where  $i = 1, \dots, n_x$ . Let  $z^{(i)} = g(x^{(i)})$ , and apply

$$\mu_z = \sum_{i=-n_x}^{n_x} \omega^{(i)} z^{(i)}, \quad (7.13a)$$

$$P_z = \sum_{i=-n_x}^{n_x} \omega^{(i)} (z^{(i)} - \mu_z) (z^{(i)} - \mu_z)^T \quad (7.13b)$$

$$+ (1 - \alpha^2 + \beta) (z^{(0)} - \mu_z) (z^{(0)} - \mu_z)^T. \quad (7.13c)$$

The design parameters of the UT are summarized below:

- $\lambda$  is defined by  $\lambda = \alpha^2(n_x + \kappa) - n_x$ .
- $\alpha$  controls the spread of the sigma points and is suggested to be approximately  $10^{-3}$ .
- $\beta$  compensates for the distribution, and should be chosen to  $\beta = 2$  for Gaussian distributions.
- $\kappa$  is usually chosen to zero.

**Table 7.1** Different Versions of the UT in (7.12) and (7.13) Using the Definition  $\lambda = \alpha^2(n_x + \kappa) - n_x$

Parameter	UT1	UT2	CT
$\alpha$	$\sqrt{3/n_x}$	$10^{-3}$	1
$\beta$	$3/n_x - 1$	2	0
$\kappa$	0	0	0
$\lambda$	$3 - n_x$	$10^{-6}n_x - n_x$	0
$\sqrt{n_x + \lambda}$	$\sqrt{3}$	$10^{-3}\sqrt{n_x}$	$\sqrt{n_x}$
$\omega^{(0)}$	$1 - n_x/3$	$-10^6$	0

Note that  $n_x + \lambda = \alpha^2 n_x$  when  $\kappa = 0$ , and that for  $n_x + \lambda \rightarrow 0^+$  the central weight  $\omega^{(0)} \rightarrow -\infty$ . Furthermore,  $\sum_i \omega^{(i)} = 1$ . We will consider the two versions of UT in Table 7.1, corresponding to the original one in [15] and an improved one in [16]. Most interestingly, [17] describes a completely different approach yielding the same transformation with a different tuning. This approach starts from the integral defining the first two moments, and applies the cubature integration rule, hence the name cubature transform used here. It appears that this tuning gives a good result in many practically interesting cases, see Chapter 8 in [12] or [18].

As an illustration, the following mapping has a well-known distribution

$$z = g(x) = x^T x, \quad x \in \mathcal{N}(0, I_n) \Rightarrow z \in \chi_n^2. \quad (7.14)$$

This distribution has mean  $n$  and variance  $2n$ . For the Taylor expansion, we get  $z \sim \mathcal{N}(0, 0)$ . This is of course not a useful approximation, but it is still what an EKF uses implicitly for quadratic functions. Now, the unscented transform gives  $z \sim \mathcal{N}(n, (3-n)n)$ ,  $z \sim \mathcal{N}(n, 2n^2)$  and  $z \sim \mathcal{N}(n, n)$ , respectively, for the three tunings in Table 7.1 (UT1, UT2, CT). This leads to useful approximations in filtering, except for the original UT1 for  $n > 3$  (when the variance becomes negative).

Having defined the UT, the UKF can now be summarized as follows.

The unscented Kalman filter consists of the following main steps:

- Define an initial Gaussian distribution  $\mathcal{N}(\hat{x}_{1|0}, P_{1|0})$  for the state  $x_1$  before the first observation  $y_1$  is provided.
- Iterate in  $k = 1, 2, \dots$ :
  1. Measurement update: apply the unscented transform to (7.1b) and use an analytical result to get  $\mathcal{N}(\hat{x}_{k|k}, P_{k|k})$  from  $\mathcal{N}(\hat{x}_{k|k-1}, P_{k|k-1})$ .
  2. Time update: apply the unscented transform to (7.1a) to immediately get  $\mathcal{N}(\hat{x}_{k+1|k}, P_{k+1|k})$  from  $\mathcal{N}(\hat{x}_{k|k}, P_{k|k})$ .

See Algorithm 2 for the details.

### 3.07.3.6 The particle filter

The particle filter (PF) works with a set of random trajectories. Each trajectory is formed recursively by iteratively simulating the model with some randomness, and then updating the likelihood of each trajectory based on the observed fingerprint:

The (marginalized) particle filter, (M)PF, for geolocation consists of the following main steps:

- Define a set of random positions (hypotheses, particles).
- For each particle, define a random velocity vector.
- Iterate:
  1. Compute the likelihood of the measurement fingerprint using the map.

2. Modify the weight (probability) of each particle.
3. Resample the set of particles.
4. Predict next position of the particles.
5. Marginalization step: update the velocity mean and covariance by a conditional Kalman filter.

See Algorithm 3 for the details.

### Algorithm 2. Unscented Kalman Filter

*Given:* Motion model (7.1a), measurement model (7.1b), the noise covariances  $Q = \text{Cov}(w)$  and  $R = \text{Cov}(e)$ , respectively, and the prior mean  $x_0$  and covariance  $P_0$ .

*Design parameter:*  $\alpha, \beta, \kappa$  in the UT, see Table 7.1.

*Initialization:* Let  $\hat{x}_{1|0} = x_0$  and  $P_{1|0} = P_0$

*Iteration:* For  $k = 1, 2, \dots$

**1. Measurement update:** Let

$$\bar{x} = \begin{pmatrix} x_k \\ e_k \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} \hat{x}_{k|k-1} \\ 0 \end{pmatrix}, \begin{pmatrix} P_{k|k-1} & 0 \\ 0 & R_k \end{pmatrix} \right), \quad (7.15a)$$

$$z = \begin{pmatrix} x_k \\ y_k \end{pmatrix} = \begin{pmatrix} x_k \\ h(x_k, u_k, e_k) \end{pmatrix}. \quad (7.15b)$$

UT in (7.12) and (7.13) gives

$$z \sim \mathcal{N} \left( \begin{pmatrix} \hat{x}_{k|k-1} \\ \hat{y}_{k|k-1} \end{pmatrix}, \begin{pmatrix} P_{k|k-1}^{xx} & P_{k|k-1}^{xy} \\ P_{k|k-1}^{yx} & P_{k|k-1}^{yy} \end{pmatrix} \right). \quad (7.15c)$$

The measurement update is then

$$K_k = P_{k|k-1}^{xy} \left( P_{k|k-1}^{yy} \right)^{-1}, \quad (7.15d)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}), \quad (7.15e)$$

$$P_{k|k}^{xx} = P_{k|k-1}^{xx} - K_k P_{k|k-1}^{yy} K_k^T. \quad (7.15f)$$

**2. Time update:** Let

$$\bar{x} = \begin{pmatrix} x_k \\ v_k \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} \hat{x}_{k|k} \\ 0 \end{pmatrix}, \begin{pmatrix} P_{k|k} & 0 \\ 0 & Q_k \end{pmatrix} \right), \quad (7.15g)$$

$$z = x_{k+1} = f(x_k, u_k, v_k). \quad (7.15h)$$

UT in (7.12) and (7.13) gives

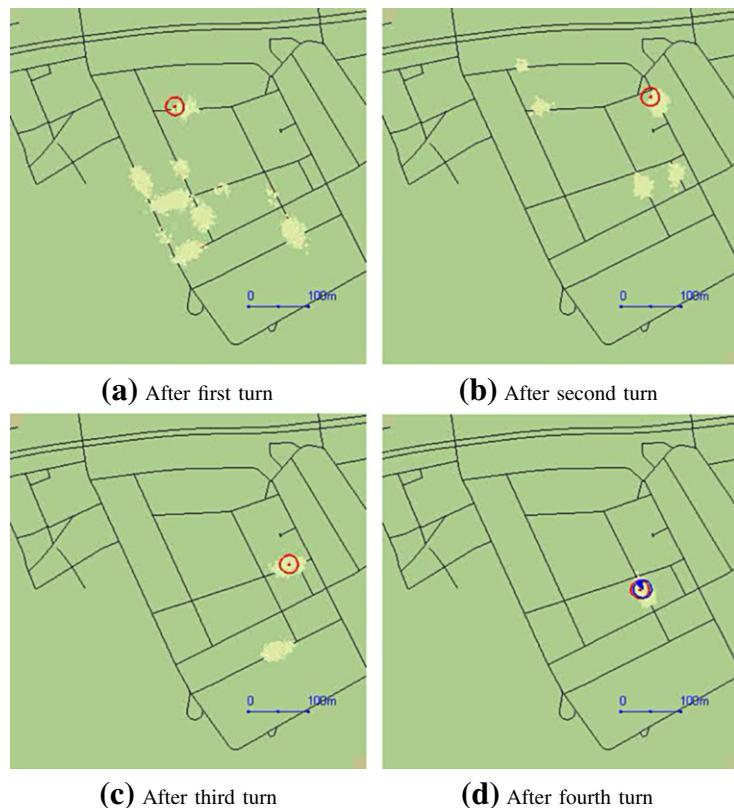
$$z \sim \mathcal{N} (\hat{x}_{k+1|k}, P_{k+1|k}). \quad (7.15i)$$

### 3.07.3.6.1 Particle filter illustration

Figure 7.4 illustrates how the dynamic fingerprint depicted in Figure 7.1 is fitted to a road-map recursively in time. Initially, the particles are spread uniformly over a part of the road network corresponding to prior knowledge. The initial particle cloud can be seen in Figure 7.4b. Figure 7.4 shows how the particle cloud improves after each turn, to eventually become one single cloud, where a marker indicates that the position can be unambiguously estimated and used as input to the navigation system.

### 3.07.3.6.2 Particle filter details

Nonlinear filtering is the branch of statistical signal processing concerned with recursively estimating the state  $x_k$  in (7.1) based on the measurements up to time  $k$ ,  $y_{1:k} \triangleq \{y_1, \dots, y_k\}$  from sample 1 to  $k$ . The



**FIGURE 7.4**

Illustration of how the particle representation of the position posterior distribution (course state is now shown) improves as the fingerprint becomes more informative. After four turns, the fingerprint is unique (unimodal posterior distribution), and a marker for geolocation is shown. The circle denotes GPS position, which is only used for evaluation purposes. Compare with Figure 7.1.

most general problem it solves is to compute the Bayesian conditional posterior density  $p(x_{1:k}|y_{1:k})$ . The posterior distribution is in the particle filter approximated with

$$p(x_{1:k}|y_{1:k}) \approx \sum_{i=1}^N \omega_{k|k}^{(i)} \delta\left(x_{1:k}^{(i)} - x_{1:k}\right). \quad (7.16)$$

This is a weighted sum of Dirac distributions. A Dirac distribution is characterized with the identity  $\int g(x)\delta(x)dx = g(0)$  for all smooth functions  $g(x)$ . This implies for instance that the mean of the trajectory is approximated with

$$\hat{x}_{1:k|k} \approx \sum_{i=1}^N \omega_{k|k}^{(i)} x_{1:k}^{(i)}. \quad (7.17)$$

Here  $\omega_{k|m}^{(i)}$  denotes the likelihood of trajectory  $x_{1:k}^{(i)}$ , given the observations  $y_{1:m}$ . The likelihood of the dynamic fingerprint can be expressed as a conditional probability density function  $p(y_{1:k}|x_{1:k})$ . Given a set of trajectories  $\{x_{1:k}^{(i)}\}_{i=1}^N$  with prior probabilities  $\{\omega_{k|0}^{(i)}\}_{i=1}^N$ , the posterior probabilities become

$$\omega_{k|k}^{(i)} \propto \omega_{k|0}^{(i)} p\left(y_{1:k}|x_{1:k}^{(i)}\right). \quad (7.18)$$

This is the main principle for how the fingerprint is used to objectively compare the set of state trajectories. The particle filter does this in a recursive manner. Its simplest form is given in Algorithm 3.

Algorithm 3 gives the basic bootstrap or SIR particle filter, that works well when the sensor observations are not very accurate (of course, a relative notion). For more accurate observations, other so called proposal distributions should be used for the prediction step, and the weight update is modified accordingly, see [19] for the details. The last marginalization step will be illustrated in Section 3.07.4, when concrete motion models are given.

### 3.07.4 Motion models

The motion model  $x_{k+1} = f(x_k, u_k, w_k)$  in (7.1b) provides a description of how the object moves from time  $t_k$  to time  $t_{k+1}$  when the next measurement is available. One interpretation is that the motion model describes how to interpolate between the position estimates computed from the measurements. It also relates position to other state variables, such as course and speed, and makes it possible to estimate these as well. The different classes of motion models are summarized below:

The trajectory may be obtained from one of the following motion model principles.

1. Integrating a kinematic model with white noise inputs.
2. Inertial navigation using accelerometers and gyroscopes as inputs to a model of a rigid body dead-reckoning model.
3. Odometry using wheel speeds as inputs to an odometric model.

- 4.** Simulation using control signals (engine speed and steering angle) as inputs to a dynamic model.

In all cases, the dead-reckoned trajectory is subject to drift, and the dynamic fingerpring can stabilize the drift relative to the map.

---

**Algorithm 3.** The (Marginalized) Particle Filter

---

*Given:* Motion model (7.1a), measurement model (7.1b), the noise distributions  $p_w(w)$  and  $p_e(e)$ , respectively, and the prior  $p_{x_0}(x)$ .

*Design parameter:* The number of particles  $N$ .

*Initialization:* Generate  $x_1^{(i)} \sim p_{x_0}$ ,  $i = 1, \dots, N$ , from prior knowledge and let  $\omega_{1|0}^{(i)} = 1/N$ .

*Iteration:* For  $k = 1, 2, \dots$

- 1. Measurement update:** For  $i = 1, 2, \dots, N$ ,

$$\omega_{k|k}^{(i)} = \frac{1}{c_k} \omega_{k|k-1}^{(i)} p_e(y_k - h(x_k^{(i)})), \quad (7.19a)$$

where the normalization weight is given by

$$c_k = \sum_{i=1}^N \omega_{k|k-1}^{(i)} p_e(y_k - h(x_k^{(i)})). \quad (7.19b)$$

- 2. Estimation:** The state can be estimated by the conditional mean (MAP is another example)

$$\hat{x}_{1:k} \approx \sum_{i=1}^N \omega_{k|k}^{(i)} x_{1:k}^{(i)}.$$

- 3. Resampling:** Take  $N$  random samples with replacement from the set  $\{x_{1:k}^{(i)}\}_{i=1}^N$  where the probability to take sample  $i$  is  $\omega_{k|k}^{(i)}$  and let  $\omega_{k|k}^{(i)} = 1/N$ .

- 4. Time update:** Generate predictions by simulating trajectories

$$w_k^{(i)} \sim p_w, \quad (7.19c)$$

$$x_{k+1}^{(i)} = f(x_k^{(i)}, u_k, w_k^{(i)}), \quad (7.19d)$$

and append this to the trajectory  $x_{1:k+1}^{(i)} = \{x_{k+1}^{(i)}, x_{1:k}^{(i)}\}$ .

- 5. Marginalization:** Update states that are not part of the fingerpring and that appear linearly in the motion model using a Kalman filter (see (7.24) and (7.28) for two examples).
- 

We here describe a couple of specific and simple two-dimensional motion models that are typical for geolocation applications. Three-dimensional extensions do exist, but are much more complex, so we prefer to focus on horizontal position in two dimensions. See [20] or Chapters 12 and 13 in [12] for a survey on motion models.

### 3.07.4.1 Dead-reckoning model

A very instructive and also quite useful motion model is based on a state vector consisting of position ( $X, Y$ ) and course (yaw angle)  $\psi$ . This assumes that there are measurements of yaw rate (derivative of course)  $\dot{\psi}$  and speed  $\vartheta$ , in which case the principle of *dead-reckoning* can be applied. In ecology, dead-reckoning is more commonly called *path integration*.

The dead-reckoning model can be formulated in continuous time using the following equations:

$$\mathbf{x}(t) = \begin{pmatrix} X(t) \\ Y(t) \\ \psi(t) \end{pmatrix}, \quad \dot{\mathbf{x}}(t) = \begin{pmatrix} \vartheta(t) \cos(\psi(t)) \\ \vartheta(t) \sin(\psi(t)) \\ \dot{\psi}(t) \end{pmatrix}. \quad (7.20)$$

A discrete time model expressed at the sampling instants  $kT$  for the nonlinear dynamics is given by

$$\begin{aligned} X_{k+1} &= X_k + \frac{2\vartheta_k}{\dot{\psi}_k} \sin\left(\frac{\dot{\psi}_k T}{2}\right) \cos\left(\psi_k + \frac{\dot{\psi}_k T}{2}\right) \\ &\approx X_k + \vartheta_k T \cos(\psi_k), \end{aligned} \quad (7.21a)$$

$$\begin{aligned} Y_{k+1} &= Y_k + \frac{2\vartheta_k}{\dot{\psi}_k} \sin\left(\frac{\dot{\psi}_k T}{2}\right) \sin\left(\psi_k + \frac{\dot{\psi}_k T}{2}\right) \\ &\approx Y_k + \vartheta_k T \sin(\psi_k), \end{aligned} \quad (7.21b)$$

$$\psi_{k+1} = \psi_k + T\dot{\psi}_k. \quad (7.21c)$$

If the yaw rate  $\dot{\psi}_k$  is small compared to the sample interval  $T$ , then the model can be simplified. Further, assume that the speed  $\vartheta_k^m$  and angular velocity  $\dot{\psi}_k^m$  are measured with some error  $w_k = (w_k^\vartheta, w_k^\psi)$ , with variance  $Q_k^\vartheta$  and  $Q_k^\psi$ , respectively. This gives the following dynamic model with process noise  $w_k$ :

$$X_{k+1} = X_k + \vartheta_k^m T \cos(\psi_k) + T \cos(\psi_k) w_k^\vartheta, \quad (7.22a)$$

$$Y_{k+1} = Y_k + \vartheta_k^m T \sin(\psi_k) + T \sin(\psi_k) w_k^\vartheta, \quad (7.22b)$$

$$\psi_{k+1} = \psi_k + T\dot{\psi}_k^m + T w_k^\psi. \quad (7.22c)$$

This model has the following structure:

$$x_{k+1} = f(x_k, u_k) + g(x_k, u_k)w_k, \quad u_k = \begin{pmatrix} \vartheta_k^m \\ \dot{\psi}_k^m \end{pmatrix}, \quad w_k = \begin{pmatrix} w_k^\vartheta \\ w_k^\psi \end{pmatrix} \quad (7.22d)$$

that fits the particle filter perfectly. Note that the speed and the angular velocity measurements are modeled as inputs, rather than measurements. This is in accordance to many navigation systems, where inertial measurements are dead-reckoned in similar ways. The main advantage is that the state dimension is kept as small as possible, which is important for the particle filter performance and efficiency. This basic model can be used in a few different cases described next.

#### 3.07.4.1.1 Odometric models

The wheel speeds  $\omega^{\text{left}}$  and  $\omega^{\text{right}}$  of two wheels with radius  $r$  on one axle of length  $L$  can be transformed to speed and yaw rate,

$$\vartheta_k^m = \frac{r}{2} (\omega_k^{\text{left}} + \omega_k^{\text{right}}), \quad (7.23a)$$

$$\dot{\psi}_k^m = \frac{r}{L} (\omega_k^{\text{left}} - \omega_i^{\text{right}}). \quad (7.23b)$$

This fits the dead-reckoning model, and this special case is commonly referred to as odometry. The state vector might need to be augmented with parameters for deviations from nominal wheel radius. For the road navigation example, the time update in Algorithm 3 can be given explicitly as in Algorithm 4

---

**Algorithm 4.** PF time update for odometry

---

In step 4 of Algorithm 3, do

1. Generate  $N$  samples of noise and add these to the wheel speed measurements  $\omega_k^{\text{left}}$  and  $\omega_i^{\text{right}}$  to get  $N$  samples of wheel speeds.
  2. Compute the speed and yaw rate from (7.23) for each sample.
  3. Propagate the set of particles according to (7.22).
- 

#### 3.07.4.1.2 Inertial models

A coarse gyro provides  $\dot{\psi}_k^m$  and a longitudinal accelerometer give the acceleration  $\dot{v}_k^m$ . If the state vector is extended with speed  $v_k$  to  $x_k = (X_k, Y_k, \psi_k, v_k)$ , and the state space model (7.22a–c) is extended with  $v_{k+1} = v_k + w_k^v$ , then we are back in the structure of (7.22d).

#### 3.07.4.1.3 Dynamical models

The steering and accelerator inputs to a car, or the rudder and engine commands to a vessel, can be statically mapped to speed  $\vartheta_k^m$  and yaw rate  $\dot{\psi}_k^m$ , and the dead-reckoning model can be applied directly. However, here the dynamics can be included to improve the motion model.

#### 3.07.4.1.4 Marginalization of speed

In inertial navigation, the state consists of four elements. Also in odometry and dynamical models, the speed is often used as a state. This fourth state usually increases the number of required particles in the PF substantially, and marginalization is recommended. The marginalization step 4 in Algorithm 3 can be used to eliminate the state  $v_k$  from the PF. The marginalized particle filter is in general quite complex to implement [21], but for special cases like this the formulas become quite concrete, as shown in Algorithm 5.

---

**Algorithm 5.** PF marginalization for dead-reckoning

---

In step 5 of Algorithm 3, do

1. Let the longitudinal acceleration  $a_k$  be an unknown continuous time white noise input with intensity  $Q^a$ .
2. Use  $\vartheta_k^m \sim \mathcal{N}(\hat{\vartheta}_{k|k}, P_{k|k}^a)$  as the speed in the motion model (7.22a,b).

3. Reformulate (7.22a,b) as a measurement of speed. Step 5 in Algorithm 3 consists of a time update

$$\hat{\vartheta}_{k+1|k}^{(i)} = \hat{\vartheta}_{k|k}^{(i)} + Ta_k, \quad (7.24a)$$

$$P_{k+1|k}^{a(i)} = P_{k|k}^{a(i)} + Q^a, \quad (7.24b)$$

and an artificial measurement update based on the simulated new value of position in step 4,

$$\varepsilon_{k+1}^{(i)} = \begin{pmatrix} X_{k+1}^{(i)} - X_k^{(i)} \\ T \cos(\psi_k^{(i)}) \\ Y_{k+1}^{(i)} - Y_k^{(i)} \\ T \sin(\psi_k^{(i)}) \end{pmatrix} - \hat{\vartheta}_{k+1|k}^{(i)}, \quad (7.24c)$$

$$H = (1, 1)^T, \quad (7.24d)$$

$$S_{k+1} = HP_{k+1|k}^{a(i)}H^T + T^2Q^{\vartheta}1_{2 \times 2}, \quad (7.24e)$$

$$K_{k+1} = P_{k+1|k}^a H^T S_{k+1}^{-1}, \quad (7.24f)$$

$$\hat{\vartheta}_{k+1|k+1}^{(i)} = \hat{\vartheta}_{k+1|k}^{(i)} + K_{k+1}\varepsilon_{k+1}^{(i)}, \quad (7.24g)$$

$$P_{k+1|k+1}^a = P_{k+1|k}^a - K_{k+1}S_{k+1}K_{k+1}^T. \quad (7.24h)$$


---

We have only discussed two-dimensional models here, but the principles are easily extended to three dimensions. The state vector then includes the height  $Z_k$ , the roll  $\phi_k$  and the pitch  $\theta_k$  angles. Thus, the state vector becomes about twice as large. In most practical application, the height is either trivial (surface vessels, cars) or an almost separate estimation problem (underwater vessels, aircraft), where pressure sensors provide accurate information to estimate height.

### 3.07.4.2 Kinematic model

The simplest possible motion model, yet one of the most common ones in target tracking applications where no inertial measurements are available, is given by a two-dimensional version of Newton's force law:

$$x(t) = \begin{pmatrix} X(t) \\ Y(t) \\ \dot{X}(t) \\ \dot{Y}(t) \end{pmatrix}, \quad \dot{x}(t) = \begin{pmatrix} \dot{X}(t) \\ \dot{Y}(t) \\ w^X(t) \\ w^Y(t) \end{pmatrix}. \quad (7.25a)$$

The corresponding discrete time model is given by

$$x_{k+1} = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x_k + \begin{pmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{pmatrix} \begin{pmatrix} w_k^X \\ w_k^Y \end{pmatrix}. \quad (7.25b)$$

The time update in Algorithm 3 is here explicitly formulized in Algorithm 6.

**Algorithm 6.** PF time update for kinematic models

In step 4 of Algorithm 3, do

1. Simulate  $N$  noise vectors  $w_k^{(i)}$ .
2. Propagate the set of particles according to (7.25b).

**3.07.4.2.1 Marginalization of speed**

Suppose the sensor model depends on the position only, which is typical in geolocation applications,

$$y_k = h(X_k, Y_k) + e_k. \quad (7.26)$$

Since the motion model is linear in the state and noise, the marginalized PF applies, so the velocity component can be handled in a numerically very efficient way.

Let  $p_k = (X_k, Y_k)^T$  and  $v_k = (\dot{X}_k, \dot{Y}_k)^T$ . Then, (7.25b) and (7.26) can be rewritten as

$$p_{k+1} = p_k + T v_k + \frac{T^2}{2} w_k, \quad (7.27a)$$

$$y_k = h(p_k) + e_k, \quad (7.27b)$$

$$v_{k+1} = v_k + T w_k, \quad (7.27c)$$

$$p_{k+1} - p_k = T v_k + \frac{T^2}{2} w_k. \quad (7.27d)$$

We here use the particle filter for (7.27a,b) and the Kalman filter for (7.27c,d). Note that (7.27a) and (7.27d) are the same, but interpreted in two different ways. The time update in the particle filter becomes

$$v_k^{(i)} = \mathcal{N}\left(\hat{v}_{k|k-1}^{(i)}, P_{k|k-1}\right), \quad (7.28a)$$

$$w_k^{(i)} = \mathcal{N}(0, Q_k), \quad (7.28b)$$

$$p_{k+1}^{(i)} = p_k^{(i)} + T v_k^{(i)} + \frac{T^2}{2} w_k^{(i)}, \quad (7.28c)$$

where we treat the velocity as a noise term. Conversely, we use the position as a measurement in the Kalman filter. For this particular structure, the general result given in Theorem 2.1 in [21] simplifies a lot, and we get a combined update

$$\hat{v}_{k+1|k}^{(i)} = \frac{p_{k+1}^{(i)} - p_k^{(i)}}{T}, \quad (7.28d)$$

$$P_{k+1|k} = P_{k|k-1} - P_{k|k-1} \left( P_{k|k-1} + \frac{T^2}{4} Q_k \right)^{-1} P_{k|k-1}. \quad (7.28e)$$

Note that each particle has an individual velocity estimate  $\hat{v}_{k|k-1}^{(i)}$  but a common covariance  $P_{k|k-1}$ . Furthermore, if  $Q_k > 0$ , the covariance matrix converges quite quickly to zero,  $P_{k|k-1} \rightarrow 0$ , and the Kalman filter is in fact not needed for the particle filter. The prediction step (7.28a,b) in the PF consists only of sampling from  $w_k^{(i)} = \mathcal{N}(0, Q)$  in (7.28a). The Kalman and particle filters are thus decoupled.

### 3.07.5 Maps and applications

This part describes a number of applications in more detail. A summary of the applications and their features is provided in Table 7.2. The common theme of these applications is that they have been applied to real data and real maps, all using the (marginalized) particle filter. A detailed mathematical description of the particle filter and marginalized particle filter for some of these applications is provided in [19].

To give some objective performance measure of the particle filter, the root mean square error (RMSE) of position (loosely speaking the standard deviation of the position error in meters) from the particle filter is in most cases below compared to a lower bound provided by the Cramer-Rao theory for nonlinear filters [14, 22]. The lower bound is an information bound, that depends on the assumption in the model. Only asymptotically in information, the lower bound can be obtained. Since the same models are used in both the filter and bound, it can be used to judge the performance of the estimation method, here the particle filter. The particle filter performance can in theory never beat the lower bound. However, in practice it can, the typical case being when the actual measurements are more accurate than what is assumed in the model.

#### 3.07.5.1 Road-bound vehicles

This application illustrates how the odometric model (7.22) with the wheel speed transformed measurements (7.23) is combined with a road map. First, the road map is discussed.

Figure 7.5 illustrates how a standard map can be converted to a likelihood function for the position. Positions on roads get the highest likelihood, and the likelihood quickly decays as the distance to the closest road increases. A small offset can be added to the likelihood function to allow for off-road driving, for instance on un-mapped roads and parking areas.

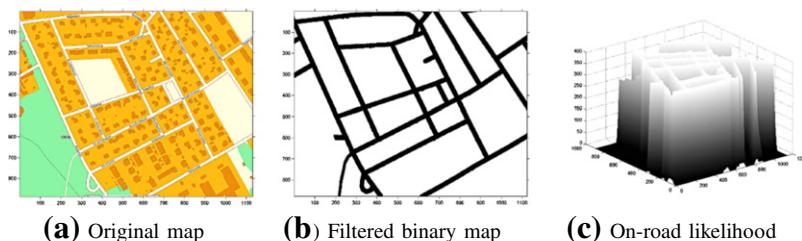
The weights in the particle filter in Algorithm 3 are multiplied with the likelihood function in the measurement update (7.19a) as

$$\omega_{k|k}^{(i)} = \omega_{k|k-1}^{(i)} l\left(X_k^{(i)}, Y_k^{(i)}\right), \quad (7.29)$$

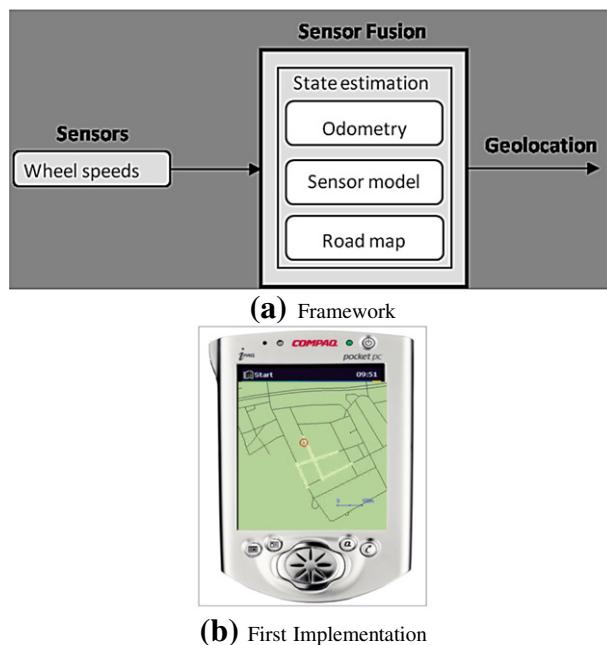
where one example of the likelihood function  $l\left(X_k^{(i)}, Y_k^{(i)}\right)$  is shown graphically in Figure 7.5.

**Table 7.2** Overview of the Illustrative Applications. The (marginalized) Particle Filter is Used in All of Them

Application	Map	Measurements	Fingerprinting	Motion Model
Road-bound vehicles	Road	Wheel speeds	Dynamic	Odometry
Airborne fast vehicles	Terrain elevation	Staring radar	Dynamic	INS
Airborne slow vehicles	Aerial image	Camera	Static	INS
Underwater vessels	Depth	Sonar	Dynamic	Dynamic
Surface vessels	Sea chart	Scanning radar	Static	INS or dynamic
Cell phones	RSS	Radio receiver	Static	Kinematic

**FIGURE 7.5**

(a) Original map. (b) The road color is masked out, and local maxima over a  $4 \times 4$  region is computed to remove text information. (c) The resulting map is low-pass filtered to allow for small deviations from the road boarders, which yields a smooth likelihood function for on-road vehicles.

**FIGURE 7.6**

(a) Framework of geolocation of road-bound vehicles. (b) A first implementation from 2001, running 15,000 particles in 2 Hz in parallel with voice-based route guidance.

The likelihood function  $l(X, Y)$  in (7.29), shown graphically in Figure 7.5c, is here used as a fingerprint. This summarizes the whole algorithm. This algorithm together with a complete navigation system including voice guidance was implemented in a student project on the platform shown in Figure 7.6 in 2001 [23]. Here, 15,000 particles were used in 2 Hz filter speed in this real-time GPS-free car navigation

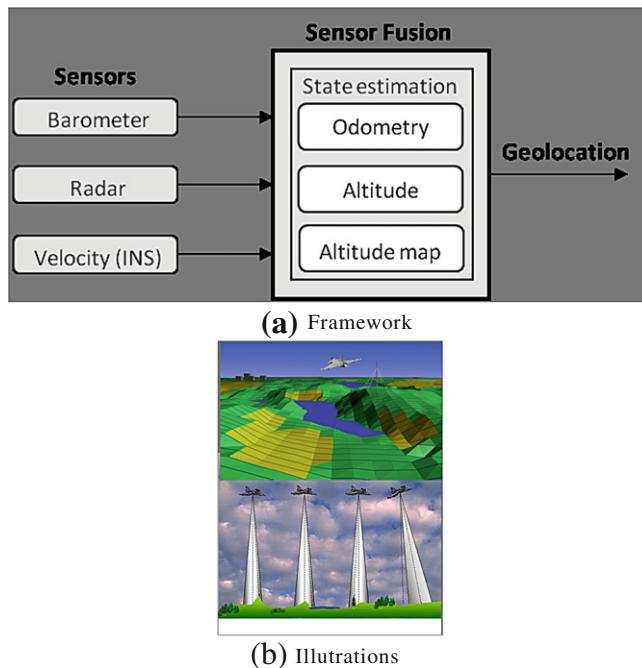
system. The conclusion from this project is that the computational complexity of the particle filter should not be overemphasized in practice.

### 3.07.5.2 Airborne fast vehicles

Figure 7.7 illustrates the main concepts of terrain navigation [24,25]. The flying platform can compute the terrain altitude variations, and this dynamic fingerprint is matched to a terrain elevation map,

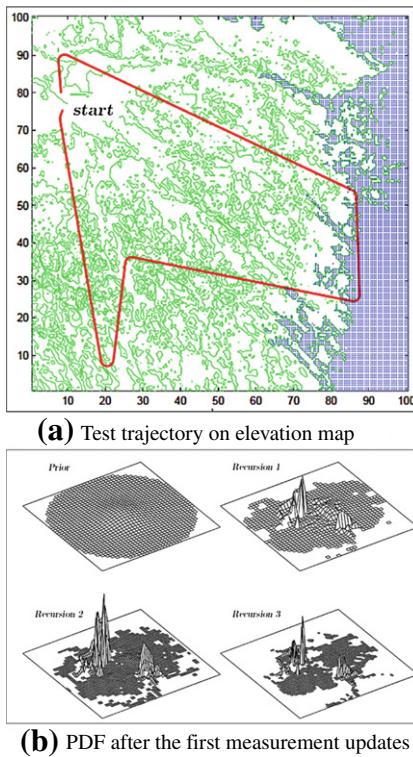
$$\omega_{k|k}^{(i)} = \omega_{k|k-1}^{(i)} p_e \left( y_k - h \left( X_k^{(i)}, Y_k^{(i)} \right) \right). \quad (7.30)$$

A separate vegetation classification map can be used to change the measurement noise distribution  $p_e(e)$ , where the idea is that the measurement from the radar is more reliable over an open field compared to a forest, for instance. In [24], it was shown that a Gaussian mixture is a good model for the radar altitude measurement error.



**FIGURE 7.7**

(a) Framework of geolocation of airborne vehicles. The principle is that the down-looking radar provides distance to ground, while the barometer connected to the INS gives a reliable altitude estimate. The difference gives the altitude on ground. (b) The measured ground altitude is compared to a terrain elevation map, where the noise distribution models possible errors due to the radar lobe width and reflectors such as tree tops.

**FIGURE 7.8**

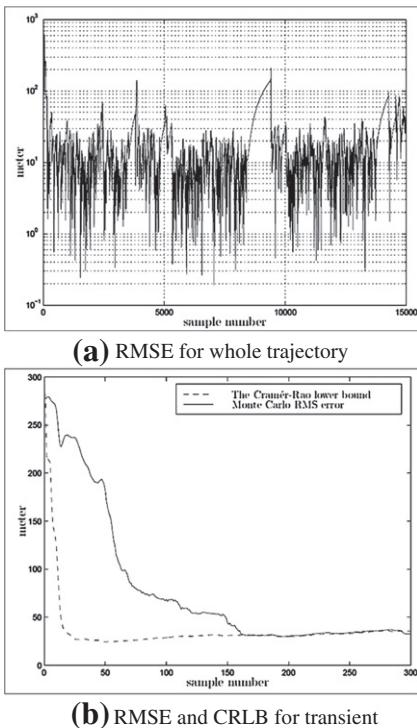
Flight trajectory overlayed the terrain elevation map (a). The position distribution after the first few iterations of the particle filter (b). Figures from [24].

Figure 7.8a shows a flight trajectory overlayed on the topographic map. Figure 7.8b shows how the particle filter approximates the position distribution after the first few measurements. It is quite clear that the distribution is peaky with a lot of local maxima corresponding to positions with a good fit to the fingerprint. This is the strength of the particle filter compared to the classical Kalman filter, that can only handle one peak, or filterbanks, where the number of peaks must be specified beforehand.

Figure 7.9a shows the RMS performance over time for the trajectory in Figure 7.8. The typical error is 10 m. One can notice that the RMSE grows when the aircraft is over the sea, since there is no information in the measurements.

Figure 7.9b shows the convergence of the particle filter, compared to the Cramer-Rao lower bound. Interestingly, the performance reaches the bound after 160 samples.

Finally, Figure 7.10 illustrates the information in the terrain elevation map. Figure 7.10a shows the map itself, and Figure 7.10b the RMSE bound as a function of position. One can here clearly identify the most informative areas at the highland, and the least informative over sea and lakes.

**FIGURE 7.9**

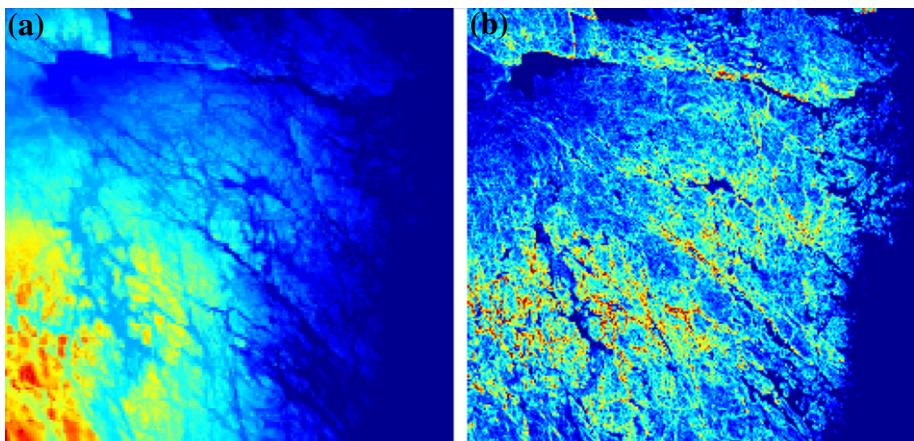
(a) RMSE as a function of sample number for the whole trajectory. (b) Zoom for the first initial phase. The RMSE is in both plots compared to the Cramer-Rao lower bound. Figures from [24].

### 3.07.5.3 Airborne slow vehicles

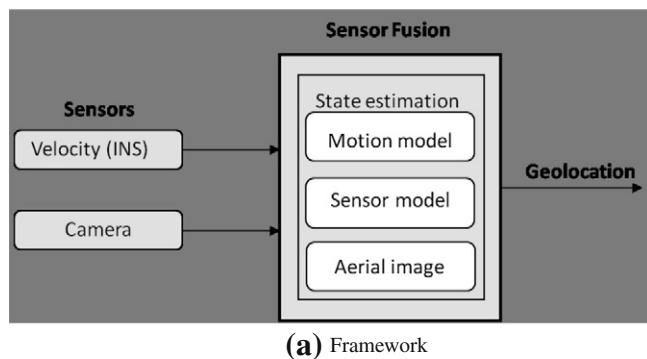
The principle in the previous section assumes that the flying platform moves quickly compared to the terrain variations. For slow platforms such as *unmanned aerial vehicles* (UAV's), this principle does not work. We here summarize another approach based on camera images of the ground.

Figure 7.11a shows the framework of the airborne geolocation system. A down-looking camera provides an aerial image that can in principle be compared to aerial photos in a database. However, this does not work in practice, due to large variations in light conditions and shadows, and also the seasonal variations. There is also a problem with a large dimensional search space. Even if the altitude of the platform is known, and the platform is assumed to be horizontal, there are still three degrees of freedom (two-dimensional translation and rotation). The solution described in [26] is based on the following steps:

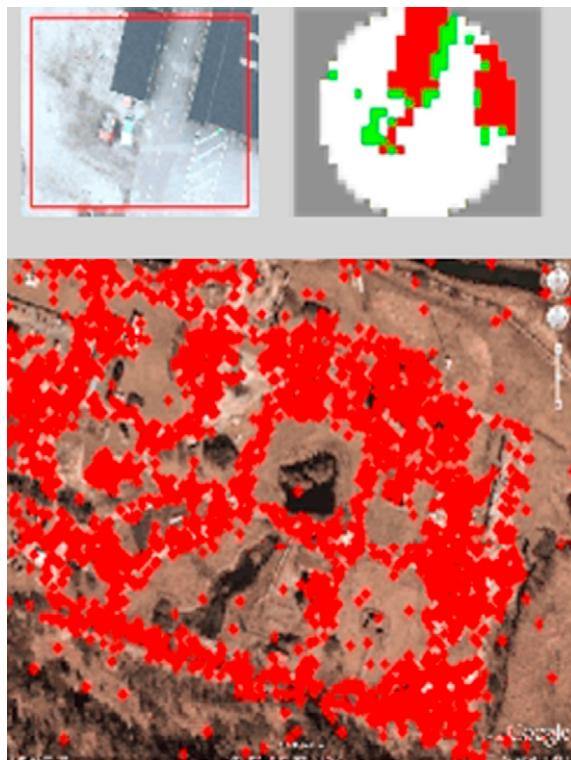
- Make a circular cut of the image.
- Segment the image and classify each segment.
- Use a histogram of the different segments as the fingerprint.

**FIGURE 7.10**

(a) Terrain elevation map. (b) Information content in the map, expressed as an expected value of the Cramer-Rao lower bound for each position. Figures from [24].

**(a)** Framework**(b)** AUV (true, geolocated, and dead-reckoned)**FIGURE 7.11**

(a) Framework of geolocation of airborne vehicles. (b) Virtual reality snapshot of the AUV experiment.

**FIGURE 7.12**

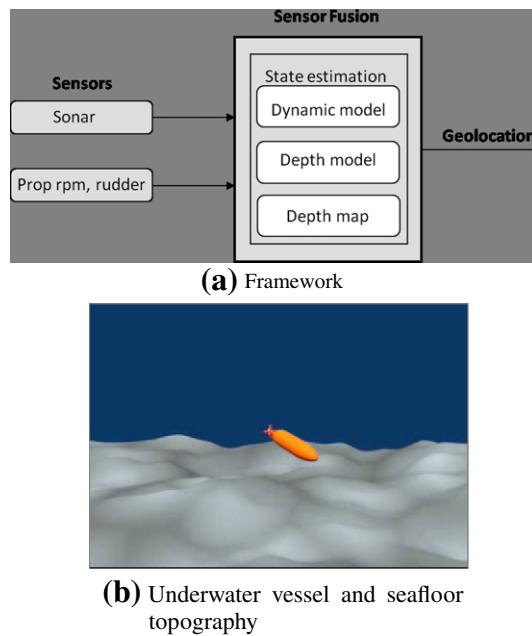
Overview of the airborne geolocation system. The camera image in the upper left corner is segmented and classified to the image in the upper right corner. The histogram of the classes is used as the fingerprint. The lower plot shows the particle cloud after a few iterations.

Figure 7.12 summarizes some of the main steps in the particle filter. Due to the circular area of interest, the method is rotation invariant. A database of fingerprints for each position can easily be pre-computed from aerial photos. The resulting matching process is in this way computationally very attractive.

### 3.07.5.4 Underwater vessels

Figure 7.13 illustrates the framework for underwater navigation [27]. A sonar measures the distance to the seafloor. The control inputs (propeller speed and rudder angle) can be simulated in a dynamic model between the sampling instants in the prediction step in the particle filter. Alternatively, an inertial measurement unit can be used for the prediction step, possibly in combination with a Doppler velocity log.

Figure 7.14 shows an underwater map as described in [28]. Figure 7.14a shows the systematic trajectory used for mapping the seafloor using a surface vessel with sonar and GPS, together with a

**FIGURE 7.13**

(a) Framework of geolocation of underwater vessels. (b) The principle is that a down-looking sonar provides distance to seafloor, while a pressure sensor (or up-looking sonar) gives the depth of the vessel. The difference gives the depth of the seafloor, which can be compared to a terrain elevation map.

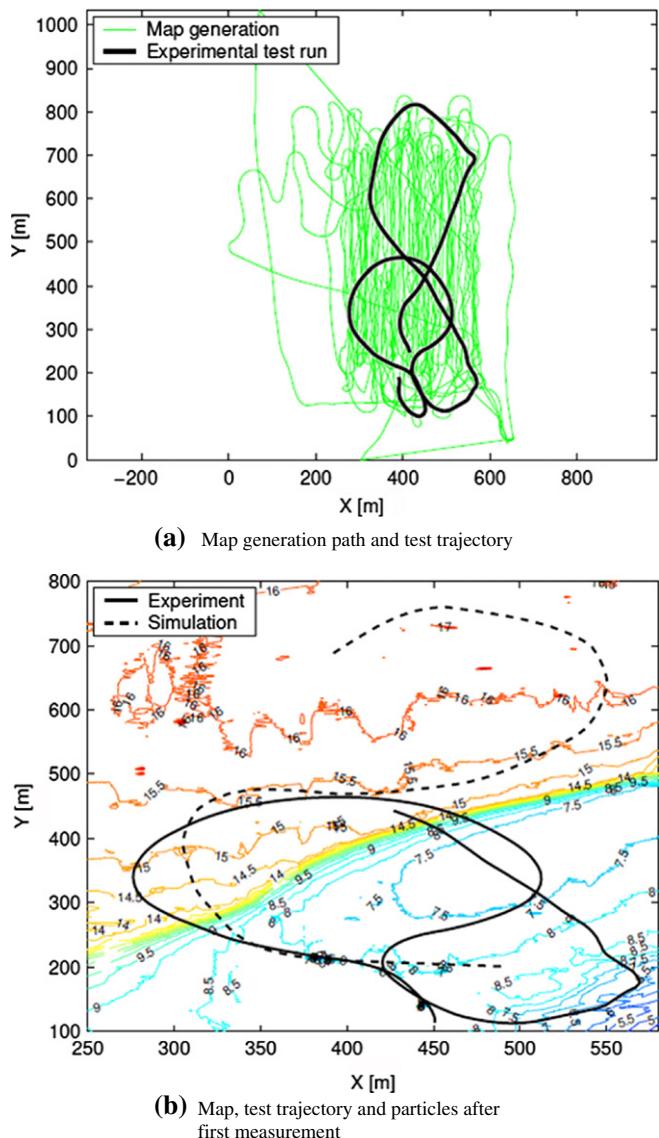
benchmark trajectory for evaluation purposes. Figure 7.14b shows the resulting seafloor map, with the ground truth trajectory, and the one obtained by just using the dynamical model (a simulation).

Figure 7.15 shows the RMSE in the two position coordinates, compared to the Cramer-Rao lower bound. The performance is within a few meter error, and actually better than the bound, indicating that the measurements are more accurate than what is modeled.

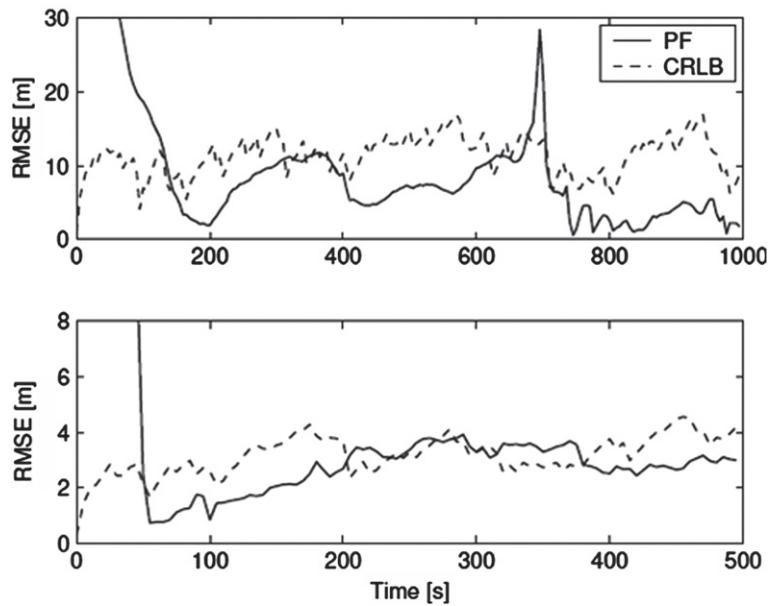
### 3.07.5.5 Surface vessels

Figure 7.16a shows the framework for surface navigation using radar map matching principle [27]. Figure 7.16b shows a zoom of the sea chart in the upper left corner. The detections from a scanning radar are shown in polar coordinates in the top right corner. This is a fingerprint, which can be fitted in the area of the sea chart corresponding to our prior knowledge of position. This is a three-dimensional search (two position coordinates and one rotation). The particle filter solves this optimization for each radar scan, using a motion model to predict the motion and rotation of the radar platform.

The performance of the filter is shown for one test trajectory in Figure 7.17a. The precision is most of the time in the order of 5 m, which is often more accurate than the coastline in a sea chart. Thus, this geolocation is even more useful for navigation than GPS, and in contrast to GPS it is not possible to

**FIGURE 7.14**

A surface vessel with GPS was used to map the area of interest in advance according to the systematic trajectory in (a), which also shows the test trajectory. The test trajectory is shown on the final map in (b).

**FIGURE 7.15**

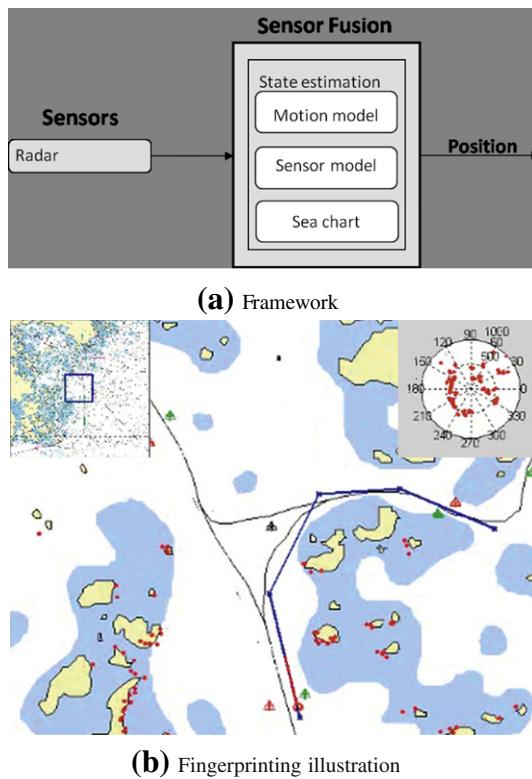
RMSE for the  $X$  and  $Y$  coordinates, respectively, for the whole trajectory in Figure 7.14.

jam. Between 50 and 60 s, the ship is moving out closer to open sea, thus decreasing the information in the radar scan. The performance is here somewhat worse than the lower bound.

Figure 7.17b shows the lower bound on RMSE as a function of position. In this case, a number of random straight line trajectories are used to average the bound.

### 3.07.5.6 Cellular phones

Figure 7.18 shows the framework of geolocation based on received signal strength (RSS) and the transmitter's identification number (ID). In this section, we survey results from a Wimax deployment in Brussels [29,30], but the same principle applies to all other cellular radio systems. At each time, a vector with  $n$  observed RSS values is obtained. The map provides a vector of  $m$  RSS values at each position (some grid is assumed here). A first problem is that  $n$  and  $m$  do not necessarily have to be the same, and the  $n$  ID's do not even need to be a subset of the  $m$  ID's in the map. One common solution is to only consider the largest set of common RSS elements. The advantage is that we can use the usual vector norms in the comparison of the measured fingerprint with the map. The disadvantage is that a missing ID gives a kind of negative information that is lost. That is, we can exclude areas based on a missing value from a certain site. A theoretically justified approach is still missing.

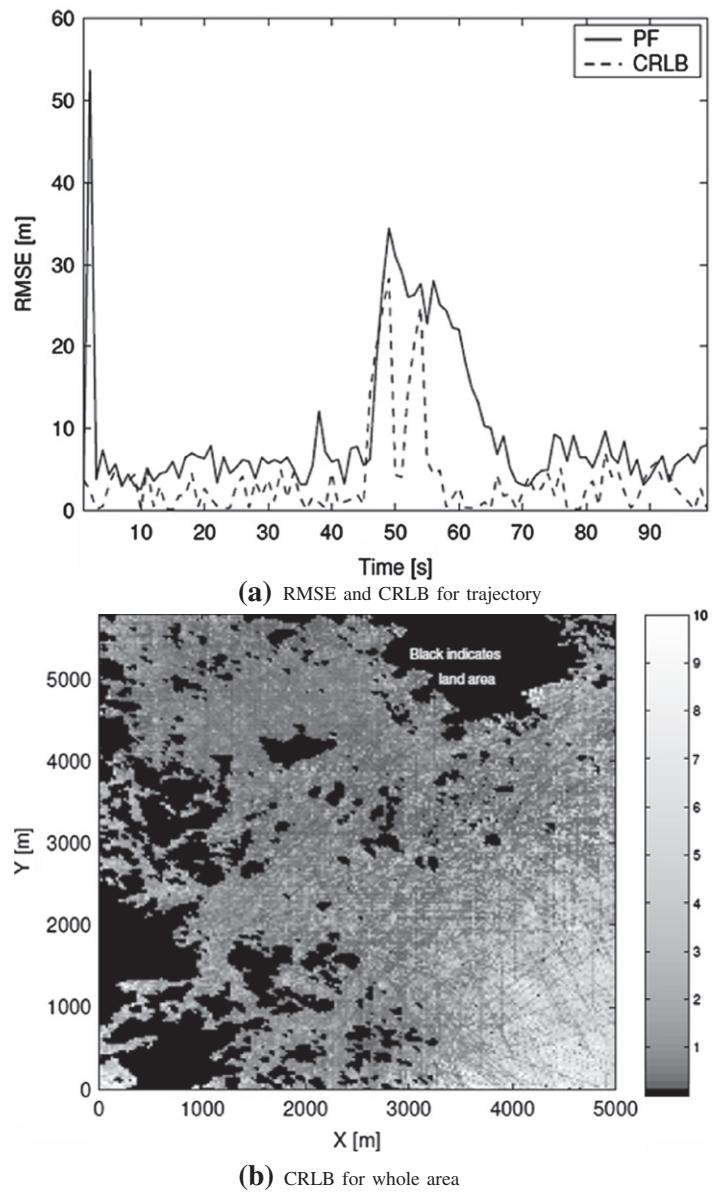
**FIGURE 7.16**

(a) Framework of geolocation of surface vessels. (b) The principle is that the scanning radar provides a 360° view of the distance to shore, which can be compared to sea chart.

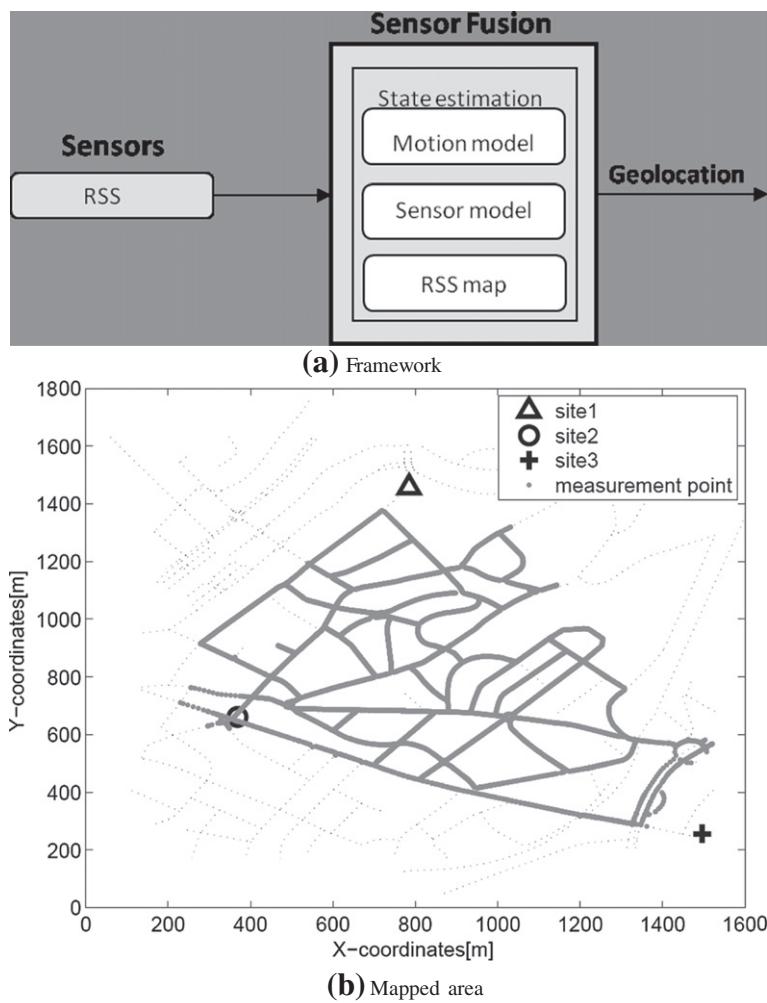
### 3.07.5.6.1 Continuous RSS measurements

Figure 7.19 shows the RSS maps from three sites over the test area [29]. There are certain combinations of knowledge in the geolocation algorithm:

- One can use the static fingerprint and estimate the position independently at each time, or use a motion model and the particle filter to fit a dynamic fingerprint. We call these approaches *static* and *dynamic*, respectively.
- One can assume an arbitrary position, or that the position is constrained to the road network. We label this *off-road* or *on-road*, respectively.
- The RSS vector from the map can either be an average computed from field tests, or based on generic path loss models, such as the Okamura-Hata model. The latter gives a quite coarse fingerprint map compared to the first one. We call these approaches *OH* and *fingerprinting*, respectively.

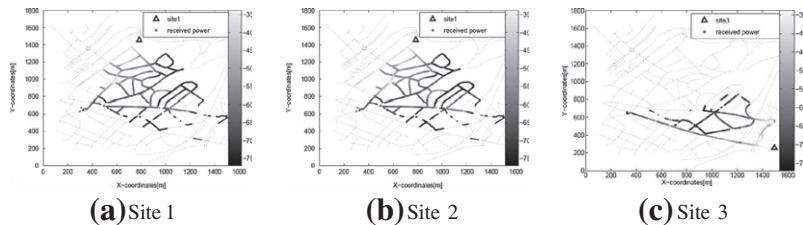
**FIGURE 7.17**

(a) RMSE for test trajectory as a function of time, in comparison to the CRLB. (b) The CRLB as a function of position, averaged over a set of straight line trajectories ending up at this spot. Figures from [27].

**FIGURE 7.18**

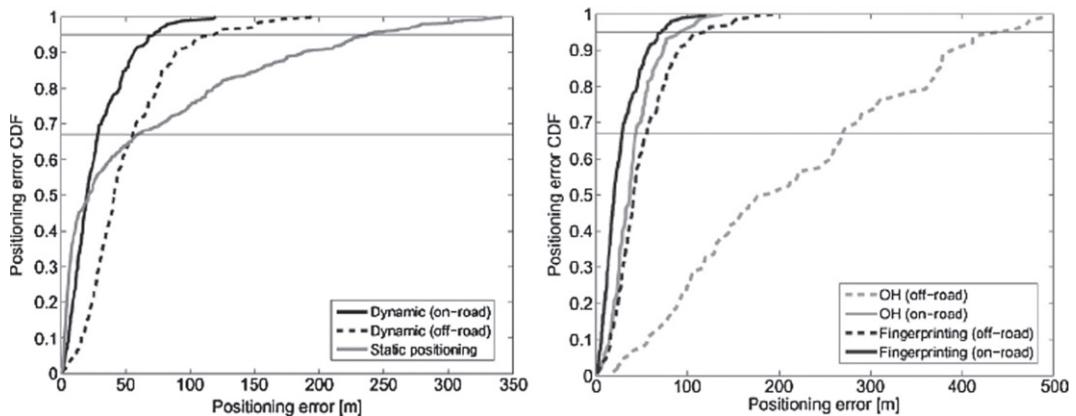
Framework of geolocation of cellular phones (a). The RSS map covers the highlighted road network from three sites (b) (from [29]).

This gives in principle eight combinations, of which six ones are compared in Figure 7.20. The solid line is the same. The plots show the position error cumulative distribution function (CDF), rather than the averaged RMSE plots used otherwise. The reason is that the US FCC legislations for geolocation as required in the emergency response system. The rules are specified for the two marked levels (horizontal lines). For mobile-centric solutions, the error should be less than 50 m in 67% of all cases and less than 150 m in 95% of all cases.



## FIGURE 7.19

RSS fingerprint consists of the submaps from the available sites in the map. Figures from [29].



**FIGURE 7.20**

RSS fingerprint consists of the submaps from the available sites in the map. Figures from [29].

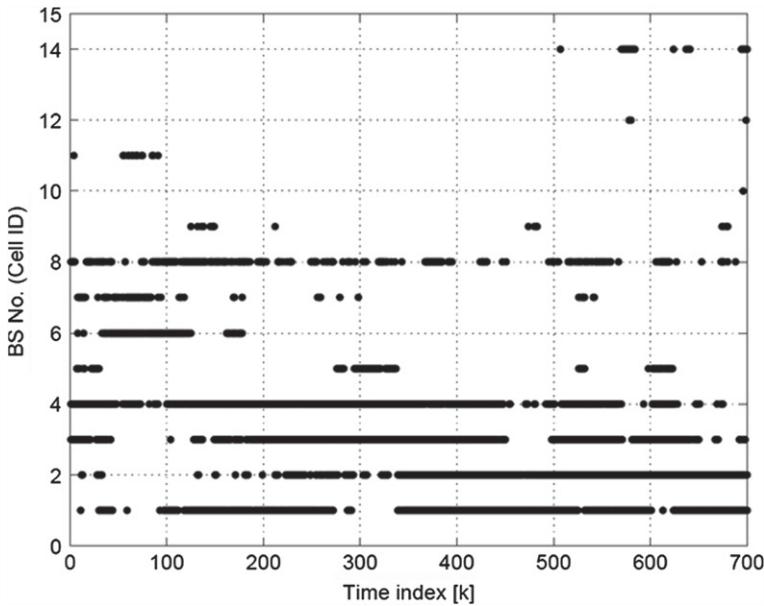
Figure 7.20a shows that filtering improves the tail compared to snapshot estimation in the case of OH model for RSS. That is, filtering avoids heavy tails in the position error. The on-road constraint contributes significantly to the performance. With filtering, the FCC requirements are satisfied.

Figure 7.20b compares four different particle filters. It is here concluded that either the road constraint or the RSS fingerprint map should be used to get a significant improvement over a solution based on the OH model and arbitrary position. In that case, the FCC requirements are satisfied with margin.

### **3.07.5.6.2 Binary RSS measurements**

A drawback with the continuous RSS map in the previous section is memory requirement. RSS is stored as integers in [0,127] for a dense grid on the road network. An alternative described in [30], is to store a binary value in larger grid areas. Figure 7.21 shows a binary dynamic fingerprint for a test drive.

Figure 7.22a illustrates one grid area in the map. Each site serves two or three cells. The arrows shows the probability to get an RSS measurement from each cell averaged over the whole grid area. Figure 7.22b shows the cumulative density function when the algorithm is evaluated on a number of tests.

**FIGURE 7.21**

Example of dynamic fingerprint for binary RSS measurements. Figure from [30].

### 3.07.5.7 Small migrating animals

Figure 7.23 shows how the sunset or sunrise at each time defines a manifold on earth [4]. A sensor consisting of a light-logger and clock can detect these two events. This principle is applied in animal geolocation [31] using lightweight sensor units (0.5 g). The theory of geolocation by light levels is described in [6] for elephant seals, where sensitivity and geometrical relations are discussed in detail. The accuracy of the geolocation is evaluated on different sharks by comparing to a satellite navigation system, and the error is shown to be in the order of  $1^\circ$  in longitude and latitude. The first filter approach to this problem is presented in [32], where a particle filter is applied.

The measurement model in the form of (7.1b) for the two events of sunset and sunrise, respectively, is

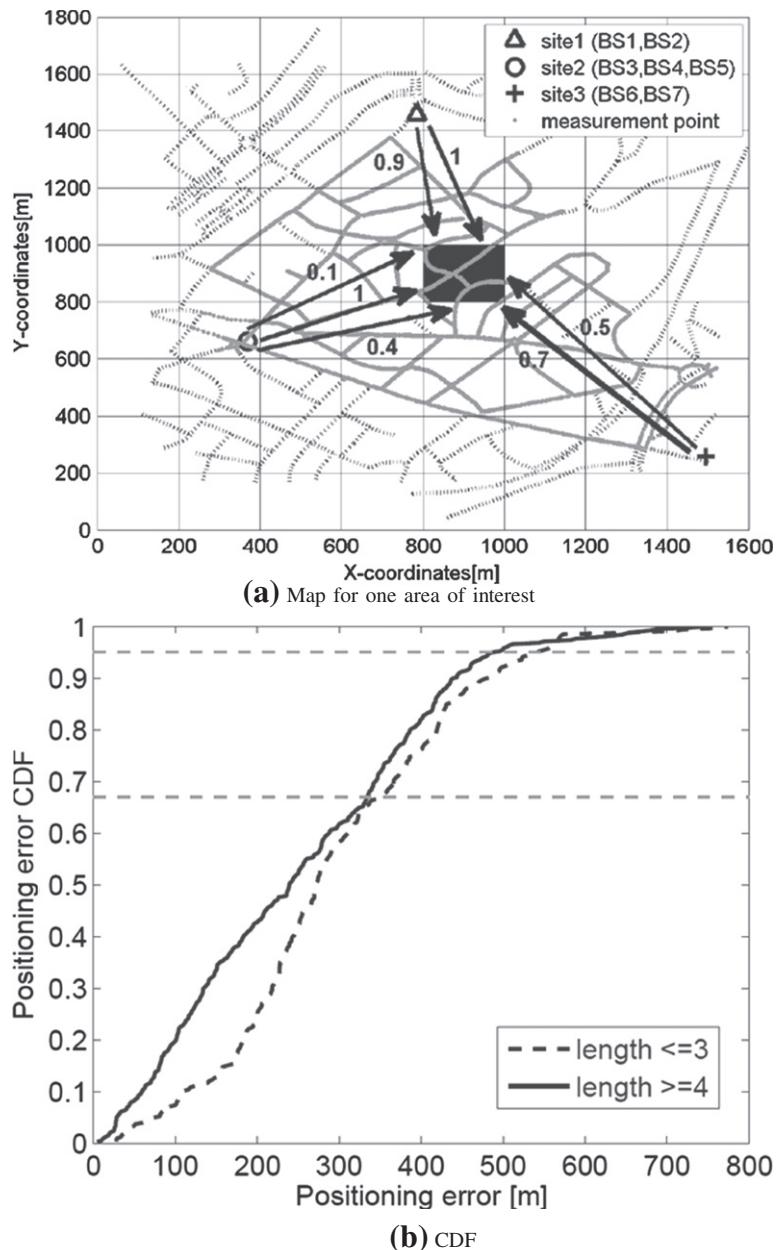
$$y^{\text{sunset}}(t_k) = h^{\text{sunset}}(t_k, X_k, Y_k) + e_k, \quad (7.31\text{a})$$

$$y^{\text{sunrise}}(t_{k+1}) = h^{\text{sunrise}}(t_{k+1}, X_{k+1}, Y_{k+1}) + e_{k+1}. \quad (7.31\text{b})$$

In this case, the measurement update (7.19a) in Algorithm 3 is

$$\begin{aligned} \omega_{k|k}^{(i)} &= \omega_{k|k-1}^{(i)} p_e^{\text{sunset}} \left( y^{\text{sunset}}(t_k) - h^{\text{sunset}}(t_k, X_k, Y_k) \right) \\ \omega_{k+1|k+1}^{(i)} &= \omega_{k+1|k}^{(i)} p_e^{\text{sunrise}} \left( y^{\text{sunrise}}(t_{k+1}) - h^{\text{sunrise}}(t_{k+1}, X_{k+1}, Y_{k+1}) \right). \end{aligned} \quad (7.32)$$

Here,  $p_e^{\text{sunset}}$  and  $p_e^{\text{sunrise}}$  denote the probability density functions for the light events.

**FIGURE 7.22**

(a) Illustration of binary RSS map averaged over an area. (b) Result of binary fingerprinting. Compare with the measurement sequence in Figure 7.21. Figures from [30].

**FIGURE 7.23**

Binary day-light model for a particular time  $t$ . The shape of the dark area depends on the time of the year, and the horizontal position of the dark area depends on the time of the day. Source: Wikipedia.

If the animal is known to be at rest during the night, the two positions  $(X_k, Y_k) = (X_l, Y_l)$  can be assumed the same, and the two unknowns can be solved from the two measurements uniquely, except for the two days of equinox when the sun is in the same plane as the equator, and thus the two manifolds in Figure 7.23 are vertical lines. Still, the PF gives useful information though the uncertainty in latitude increases, see Figure 7.24.

### 3.07.6 Mapping in practice

Many of the maps described here are publically or commercially available. Highly accurate topographic maps on dense grids are now available from satellite radar or laser measurements. Road maps and sea charts over whole continents are also available, though still with some absolute errors. This error will likely be corrected in the future based on aerial imagery and satellite measurements.

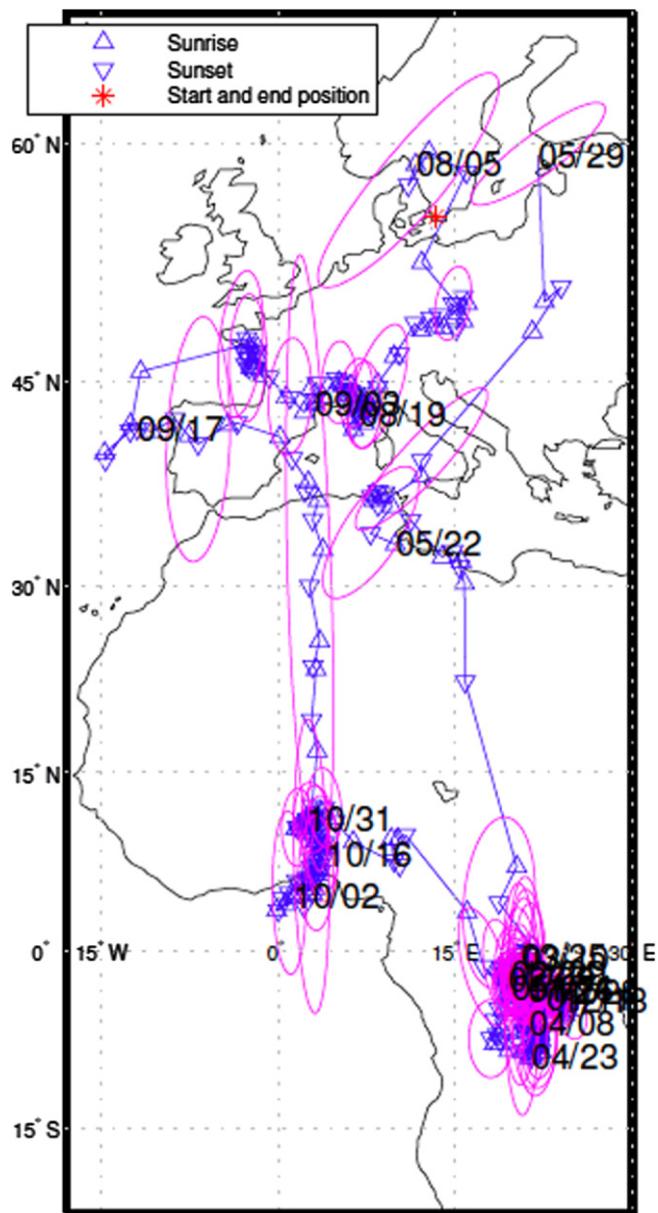
Maps of seafloor and magnetic field variations are still quite sparsely gridded, and not useful for geolocation. In particular the magnetic field, is time-varying, so regular updates of the map are required.

Cellular network maps of RSS or base station locations are also rapidly changing. Changes in the environment also affect the RSS values, which is particular true in indoor scenarios.

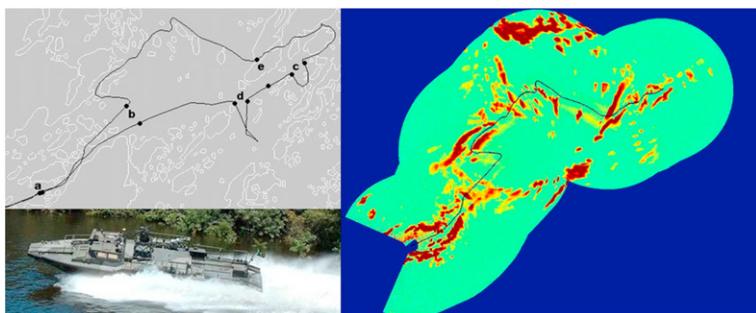
Thus, there is a need for efficient mapping algorithms that avoid manual tests to the largest possible extent. There are two main principles:

- Creating and adapting the map at the same time as performing geolocation. This approach is known as *simultaneous localization and mapping* (SLAM) in literature [33,34].
- Opportunistic reports from objects with a reference system.

Google Maps is an interesting project that illustrates two different principles for localization when GPS is not available. If a known WLAN is detected, whose ID is found in a database, the geolocation is based on the coverage of WLAN. The database is created by the company during street view campaigns. Second, if no WLAN is detected, the strongest RSS from the cellular network is used. Again, the ID is

**FIGURE 7.24**

Estimated path of a Swift from 298 sunsets and sunrises, respectively. The ellipsoids illustrate a confidence interval at selected days. Figure from [32].

**FIGURE 7.25**

Simultaneous geolocation of surface ships and generation of a sea chart. Figures from [35].

looked up in a map. This time, the map is created by users with GPS available. The map contains for each ID a circle covering a large fraction of the GPS reports.

The SLAM approach is best illustrated with an example, here taken from [35]. The surface navigation application based on radar scan matching to a sea chart in Section 3.07.5.5 is revisited, assuming that there is no sea chart available. This might not be a relevant example in practice, but the idea is easily extended to seafloor mapping, earth magnetic field mapping, indoor RSS mapping, etc.

Figure 7.25 shows the test trajectory (upper left), the boat (lower left) and the radar scans overlaid on each other to form a sea chart like map (right). Here, also the estimated first half of the test trajectory is plotted. No other information than the radar scans has been used, which shows that clever algorithms can actually solve the complex mapping task autonomously.

### 3.07.7 Conclusion

Geolocation is the art of innovative combination of properties of nature and sensor technology. We have provided a set of applications illustrating how different sensors and maps (geographical information systems) can be used to compute geolocation. The particle filter has been used as a general method for geolocation. Fingerprinting is a general concept for fitting measurements (along a trajectory) to a map.

### Acknowledgment

This work has been supported by the Swedish Research Council (VR) under the Linnaeus Center CADICS, the VR project grant *Extended Target Tracking* and the SSF project *Cooperative Localization*.

*Relevant Theory:* Statistical Signal Processing and Machine Learning

See Vol. 1, Chapter 19 A Tutorial Introduction to Monte Carlo Methods, Markov Chain Monte Carlo and Particle Filtering

See this Volume, Chapter 1 Introduction to Statistical Signal Processing

See this Volume, Chapter 4 Bayesian Computational Methods in Signal Processing

---

## References

- [1] S. Altizer, R. Bartel, B.A. Han, Animal migration and infectious disease risk, *Science* 331 (2011) 296–302.
- [2] J.A. Muir, P.C. Van oorschot, Internet geolocation: evasion and counterevasion, *ACM Comput. Surv. (CSUR)* 42 (1) (2009).
- [3] E. Katz-Bassett, J.P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, Y. Chawathe, Towards IP geolocation using delay and topology measurements, in: *Proceedings of the Sixth ACM SIGCOMM Conference on Internet Measurement*, 2006.
- [4] J.H. Meeus, *Astronomical Algorithms*, Willmann-Bell, Inc., 1991.
- [5] C. Tyren, Magnetic terrain navigation, in: *Fifth International Symposium on Unmanned Untethered Submersible Technology*, 1987.
- [6] Roger D. Hill, Theory of geolocation by light levels, in: Burney J. Le Boeuf, Richard M. Laws (Eds.), *Elephant Seals: Population Ecology, Behavior, and Physiology*, University of California Press, 1994.
- [7] S.L.H. Teo, A. Boustany, S. Blackwell, A. Walli, K.C. Weng, B.A. Block, Validation of geolocation estimates based on light level and sea surface temperature from electronic tags, *Mar. Ecol. Prog. Ser.* 283 (2004) 81–88.
- [8] F. Gustafsson, F. Gunnarsson, Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements, *IEEE Signal Process. Mag.* 22 (2005) 41–53.
- [9] L. Hui, H. Darabi, P. Banerjee, L. Jing, Survey of wireless indoor positioning techniques and systems, *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* 37 (6) (2007) 1067–1080.
- [10] K. Pahlavan, L. Xinrong, J.P. Makela, Indoor geolocation science and technology, *IEEE Commun. Mag.* 40 (2) (2002) 112–118.
- [11] Jouni Rantakokko, Joakim Rydell, Peter Stromback, Peter Handel, Jonas Callmer, David Törnqvist, Fredrik Gustafsson, Magnus Jobs, Matthias Gruden, Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization, *IEEE Wirel. Commun.* 18 (2) (2011) 10–18.
- [12] F. Gustafsson, *Statistical Sensor Fusion*, Studentlitteratur, second ed., 2010.
- [13] N. Bergman, *Recursive Bayesian Estimation: Navigation and Tracking Applications*, Dissertation no. 579, Linköping University, Sweden, 1999.
- [14] P. Tichavsky, C.H. Muravchik, A. Nehorai, Posterior Cramér-Rao bounds for discrete-time nonlinear filtering, *IEEE Trans. Signal Process.* 46 (5) (1998) 1386–1396.
- [15] S.J. Julier, J.K. Uhlmann, Hugh F. Durrant-Whyte, A new approach for filtering nonlinear systems, in: *IEEE American Control Conference*, 1995, pp. 1628–1632.
- [16] E.A. Wan, R. van der Merwe, The unscented Kalman filter for nonlinear estimation, in: *Proceedings of IEEE Symposium (AS-SPCC)*, pp. 153–158.
- [17] I. Arasaratnam, S. Haykin, R.J. Elliot, Cubature Kalman filter, *IEEE Trans. Autom. Control* 54 (2009) 1254–1269.
- [18] Fredrik Gustafsson, Gustaf Hendeby, Some relations between extended and unscented Kalman filters, *IEEE Trans. Signal Process.* 60 (2) (2012) 545–555 (funding agencies—Swedish research council (VR)).
- [19] F. Gustafsson, Particle filter theory and practice with positioning applications, *IEEE Trans. Aerosp. Electron. Mag.* 7 (2010) 53–82.
- [20] X.R. Li, V.P. Jilkov, Survey of maneuvering target tracking. Part I: dynamic models, *IEEE Trans. Aerosp. Electron. Syst.* 39 (4) (2003) 1333–1364.
- [21] T.B. Schön, F. Gustafsson, P.J. Nordlund, Marginalized particle filters for nonlinear state-space models, *IEEE Trans. Signal Process.* 53 (2005) 2279–2289.
- [22] N. Bergman, Posterior Cramér-Rao bounds for sequential estimation, in: A. Doucet, N. de Freitas, N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, 2001.
- [23] U. Forssell, P. Hall, S. Ahlgvist, F. Gustafsson, Novel map-aided positioning system, in: *Proceedings of FISITA*, No. F02-1131, Helsinki, 2002.

- [24] N. Bergman, L. Ljung, F. Gustafsson, Terrain navigation using Bayesian statistics, *IEEE Control Syst. Mag.* 19 (3) (1999) 33–40.
- [25] P.-J. Nordlund, F. Gustafsson, Marginalized particle filter for accurate and reliable terrain-aided navigation, *IEEE Trans. Aerosp. Electron. Syst.* (2009).
- [26] P. Skoglar, U. Orguner, D. Törnqvist, F. Gustafsson, in: *IEEE Aerospace Conference*, 2010.
- [27] R. Karlsson, F. Gustafsson, Bayesian surface and underwater navigation, *IEEE Trans. Signal Process.* 54 (11) (2006) 4204–4213.
- [28] T. Karlsson, Terrain aided underwater navigation using Bayesian statistics, Master Thesis LiTH-ISY-EX-3292, Dept. Elec. Eng., Linköping University, S-581 83 Linköping, Sweden, 2002.
- [29] M. Bshara, U. Orguner, F. Gustafsson, L. Van Biesen, Fingerprinting localization in wireless networks based on received signal strength measurements: a case study on wimax networks, 2010. <[www.control.isy.liu.se/fredrik/reports/09tvtmussa.pdf](http://www.control.isy.liu.se/fredrik/reports/09tvtmussa.pdf)>.
- [30] M. Bshara, U. Orguner, F. Gustafsson, L. VanBiesen, Robust tracking in cellular networks using HMM filters and cell-ID measurements, 60 (3) (2011) 1016–1024.
- [31] T. Alerstam, A. Hedenström, S. Åkesson, Long-distance migration: evolution and determinants, *Oikos* 103 (2) (2003) 247–260.
- [32] Niklas Wahlström, Fredrik Gustafsson, Susanne Åkesson, A voyage to Africa by Mr. Swift, in: Proceedings of 15th International Conference on Information Fusion, Singapore, July 2012.
- [33] T. Bailey, H. Durrant-Whyte, Simultaneous localization and mapping (SLAM): Part II, *IEEE Robot. Autom. Mag.* 13 (3) (2006) 108–117.
- [34] H. Durrant-Whyte, T. Bailey, Simultaneous localization and mapping (SLAM): Part I, *IEEE Robot. Autom. Mag.* 13 (2) (2006) 99–110.
- [35] J. Callmer, D. Törnqvist, H. Svensson, P. Carlbom, F. Gustafsson, Radar SLAM using visual features, *EURASIP J. Adv. Signal Process.* (2011).

# Performance Analysis and Bounds

# 8

**Brian M. Sadler and Terrence J. Moore**

*Army Research Laboratory, Adelphi, MD, USA*

## 3.08.1 Introduction

In this chapter we consider performance analysis of estimators, as well as bounds on estimation performance. We introduce key ideas and avenues for analysis, referring to the literature for detailed examples. We seek to provide a description of the analytical procedure, as well as to provide some insight, intuition, and guidelines on applicability and results.

Bounds provide fundamental limits on estimation given some assumptions on the probability laws and a model for the parameters of interest. Bounds are intended to be independent of the particular choice of estimator, although any nontrivial bound requires some assumptions, e.g., a bound may hold for some class of estimators, or may be applicable only for unbiased estimators, and so on.

On the other hand, given a specific estimation algorithm, we would like to analytically characterize its performance as a function of the relevant system parameters, such as available data length, signal-to-noise ratio (SNR), etc. Together, fundamental bounds and analysis of algorithms go hand in hand to provide a complete picture.

We begin with the specific, examining the classic theory of parameterized probability models, and the associated Cramér-Rao bounds and their relationship with maximum likelihood estimation. These ideas are fundamental to statistical signal processing, and understanding them provides a significant foundation for general understanding. We then consider mean square error analysis more generally, as well as perturbation methods for algorithm small-error analysis that is especially useful for algorithms that rely on subspace decompositions. We also describe the more recent theory of the constrained Cramér-Rao bound, and its relationship with constrained maximum-likelihood estimation.

We examine the case of a parameterized signal with both additive and multiplicative noise in detail, including Gaussian and non-Gaussian cases. The multiplicative random process introduces signal variations as commonly arise with propagation through a randomly fluctuating medium. General forms for the CRB for estimating signal parameters in multiplicative and additive noise are available that encompass many cases of interest.

We next consider asymptotic analysis, as facilitated by the law of large numbers and the central limit theorem. In particular, we consider two fundamental and broadly applied cases, the asymptotics of Fourier coefficients, and asymptotics of nonlinear least squares estimators. Under general conditions, both result in tractable expressions for the estimator distribution as it converges to a Gaussian.

Finally, we look at Monte Carlo methods for evaluating expressions involving random variables, such as numerical evaluation of the Cramér-Rao bound when obtaining an analytical expression is challenging. We close with a discussion of confidence intervals that provide statistical evidence of estimator quality, often based on asymptotic arguments, and can be computed from the given data.

### 3.08.2 Parametric statistical models

Let  $p(\mathbf{x}; \boldsymbol{\theta})$  denote an  $N$ -dimensional probability density function (pdf) that depends on the parameters in the vector  $\boldsymbol{\theta} \in \mathcal{R}^m$ . The vector  $\mathbf{x}$  is a collection of random variables. For example, suppose  $p(\mathbf{x}; \boldsymbol{\theta})$  describes a normal distribution with independent and identically distributed (iid) elements each with variance  $\sigma_x^2$ . Then the parameters describing the pdf are  $\boldsymbol{\theta} = [\boldsymbol{\mu}_x^T, \sigma_x]^T$ , where  $E[\mathbf{x}] = \boldsymbol{\mu}_x$  is the mean vector. With the iid assumption, the covariance is completely determined by the scalar variance  $\sigma_x^2$ ; see Section 3.08.2.1.4. Here,  $\mathbf{x} = [x(0), x(1), \dots, x(N - 1)]^T$  are random variables from a normal distribution with the specified mean and variance. In the 1-D case, such as a scalar time series  $x(n)$ , the random variables may be stacked into the vector  $\mathbf{x} = [x(0), x(1), \dots, x(N - 1)]^T$ , and  $p(\mathbf{x}; \boldsymbol{\theta})$  is the joint pdf of  $N$  consecutive observations of  $x(n)$ .

When we have samples from the random process then we can think of the vector  $\mathbf{x}$  as containing a realization of the random process governed by the pdf; now the contents of  $\mathbf{x}$  are also often called the *observations*. This can be confusing because we have not altered the notation, but have altered our interpretation of  $p(\mathbf{x}; \boldsymbol{\theta})$  to be a function of  $\boldsymbol{\theta}$ , with a given set of observations  $\mathbf{x}$  regarded as fixed and known constants. In this interpretation, the pdf is called the likelihood function.

The functional description  $p(\mathbf{x}; \boldsymbol{\theta})$  is remarkably general, incorporating knowledge of the underlying model via the functional dependence on the parameters in  $\boldsymbol{\theta}$ , and expressing randomness (i.e., lack of specific knowledge about  $\mathbf{x}$ ) through the distribution. Much of the art of statistical signal processing is defining the model for a specific problem, expressed in  $p(\mathbf{x}; \boldsymbol{\theta})$ , that can sufficiently capture the essential nature of the observations, lead to tractable and useful signal processing algorithms, and enable performance analysis. We seek the smallest dimensionality in  $\boldsymbol{\theta}$  such that the underlying model remains sufficiently detailed to capture the behavior of the observations, while not over-parameterizing in a way that adds too much complexity or unneeded model variation. It is important to keep in mind that a model is a useful abstraction. Over-specifying the model may be as bad as underspecification. While a large number of parameters may be appealing to better fit some observed data, the model can easily lose generality and become too cumbersome to manipulate and estimate its parameters. On the other hand, if the model is underspecified then it may be too abstract to capture essential behavior. All of this motivates exploration of families of models and levels of abstraction. A key component of performance analysis in this context is to explore the pros and cons of models in their descriptive power.

In statistical signal processing we are often interested in a signal plus noise model. Expressed in 1-D, we have

$$x(n) = s(n; \boldsymbol{\theta}) + v(n), \quad n = 0, 1, \dots, N - 1, \quad (8.1)$$

where  $s(n; \boldsymbol{\theta})$  is a deterministic parameterized signal model, and  $v(n)$  is an additive random process modeling noise and/or interference. The most basic assumptions typically begin with assuming  $v(n)$  are samples from a stationary independent Gaussian process, i.e., additive white Gaussian noise (AWGN).

AWGN is fundamental for several reasons; it models additive thermal noise arising in sensors and their associated electronics, it arises through the combination of many small effects as described through the central limit theorem, it is often the worst case as compared with additive iid non-Gaussian noise (see Section 3.08.7), and Gaussian processes are tractable and fully specified by their first and second-order statistics. (We can generalize (8.1) so that the signal  $s(n; \theta)$  is also a random process, in which case it is also common to assume the noise is statistically independent of the signal.) Note that for  $E[v(n)] = 0$ , then information about  $\theta$  in the observations  $x(n)$  is contained in the time-varying mean  $s(n; \theta)$ . If instead  $v(n) = v(n; \theta)$  is a function of the parameters in  $\theta$ , then there is information about  $\theta$  in the covariance of  $x(n)$ . A fundamental extension to (8.1) incorporates propagation of  $s(n; \theta)$  through a linear channel, given by  $x(n) = h(n) * s(n; \theta) + v(n)$  where  $*$  denotes convolution. The observation model (8.1) is the tip of the iceberg for a vast array of models that incorporate aspects such as man-made signals or naturally occurring signals, interference, and the physics of sensors and electronics. Understanding of performance analysis for (8.1) is fundamental to these many cases.

Given observations  $\mathbf{x}$ , let

$$\mathbf{t}(\mathbf{x}) = \hat{\boldsymbol{\theta}} \quad (8.2)$$

be an estimator of  $\theta$ . Performance analysis primarily focuses on two objectives. First, we wish to find bounds on the possible performance of  $\mathbf{t}(\mathbf{x})$ , without specifying  $\mathbf{t}(\mathbf{x})$ . To do this we typically need some further assumptions or restrictions on the possible form or behavior of  $\mathbf{t}(\mathbf{x})$ . For example, we might consider only those estimators that are unbiased so that  $E[\hat{\boldsymbol{\theta}}] = \boldsymbol{\theta}$ . Or, we might restrict our attention to the class of  $\mathbf{t}(\mathbf{x})$  that are linear functions of the observations, so  $\hat{\boldsymbol{\theta}} = H\mathbf{x}$  for some matrix  $H$ . The second objective is to analyze the performance of a specific estimator. We are given a specific estimation algorithm  $t_0(\mathbf{x})$  and we wish to assess its performance. These two objectives are bound together, and their combination provides a clear picture; comparing algorithm performance with bounds will reveal the optimality or sub-optimality of the algorithm, and helps guide the development of good algorithms.

The most commonly applied algorithm and performance framework for parametric models is maximum likelihood estimation (MLE) and the Cramér-Rao bound (CRB). These are directly related, as described in the following.

In the context of performance analysis, it can be useful to evaluate the MLE and compare it to the CRB, even if the MLE has undesirably high complexity. This gives us a benchmark to compare other algorithms against. For example, we might simplify our assumptions, or derive an approximation to the MLE, resulting in an algorithm with lower complexity. We can then explore the complexity-performance tradeoff in a meaningful way.

### 3.08.2.1 Cramér-Rao bound on parameter estimation

The Cramér-Rao bound (CRB) is the most widely applied technique for bounding our ability to estimate  $\theta$  given observations from  $p(\mathbf{x}; \theta)$ . The basic definition is as follows. Suppose we have an observation  $\mathbf{x}$  in  $\mathbb{X} \subset \mathcal{R}^n$  from the pdf  $p(\mathbf{x}; \theta)$  where  $\theta$  is a vector of deterministic parameters in an open set  $\Theta \subset \mathcal{R}^m$ . The *Fisher information matrix* (FIM) for this model is given by

$$\mathbf{I}(\theta) \triangleq E_{\theta} \left\{ \mathbf{s}(\mathbf{x}; \theta) \mathbf{s}^T(\mathbf{x}; \theta) \right\}, \quad (8.3)$$

where  $s(\mathbf{x}; \boldsymbol{\theta})$  is the *Fisher score* defined by

$$s(\mathbf{x}; \boldsymbol{\theta}) \triangleq \frac{\partial \log p(\mathbf{x}; \boldsymbol{\theta}')}{\partial \boldsymbol{\theta}'} \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} \quad (8.4)$$

and the expectation is evaluated at  $\boldsymbol{\theta}$ , i.e.,  $E_{\boldsymbol{\theta}}(\cdot) = \int_{\mathbb{X}} (\cdot) p(\mathbf{x}; \boldsymbol{\theta}) d\mathbf{x}$ . We assume certain *regularity conditions*, namely that the pdf is differentiable with respect to  $\boldsymbol{\theta}$  and satisfies

$$\frac{\partial}{\partial \boldsymbol{\theta}'^T} E_{\boldsymbol{\theta}'}(\mathbf{h}(\mathbf{x})) \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} = E_{\boldsymbol{\theta}}(\mathbf{h}(\mathbf{x}) s^T(\mathbf{x}; \boldsymbol{\theta})) \quad (8.5)$$

for both  $\mathbf{h}(\mathbf{x}) \equiv 1$  and  $\mathbf{h}(\mathbf{x}) \equiv \mathbf{t}(\mathbf{x})$  where  $\mathbf{t}(\mathbf{x})$  is an unbiased estimator of  $\boldsymbol{\theta}$  [1]. Intuitively, existence of the bound relies on the smoothness of the pdf in the parameters, and that the allowable range of the parameters does not have hard boundaries where the derivatives may fail to exist. Regularity and the assumption that  $\boldsymbol{\theta}$  is in an open set guarantee this.

Evaluation of  $\mathbf{I}(\boldsymbol{\theta})$  in (8.3) uses the product of first derivatives of  $\log p(\mathbf{x}; \boldsymbol{\theta})$ . Note the condition in (8.5) when  $\mathbf{h}(\mathbf{x}) = 1$  permits us to substitute

$$-E_{\boldsymbol{\theta}} \left\{ \frac{\partial}{\partial \boldsymbol{\theta}'^T} s(\mathbf{x}; \boldsymbol{\theta}') \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} \right\} \quad (8.6)$$

in (8.3), yielding

$$\mathbf{I}(\boldsymbol{\theta}) = -E_{\boldsymbol{\theta}} \left\{ \frac{\partial^2}{\partial \boldsymbol{\theta}' \partial \boldsymbol{\theta}'^T} \log p(\mathbf{x}; \boldsymbol{\theta}') \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}} \right\}. \quad (8.7)$$

Equation (8.7) provides an alternative expression for the FIM in terms of second-order derivatives of  $\log p(\mathbf{x}; \boldsymbol{\theta})$ , that in some cases may be easier to evaluate.

The regularity condition in (8.5) is assured when the Jacobian and Hessian of the density function  $p(\mathbf{x}; \boldsymbol{\theta})$  are absolutely integrable with respect to both  $\mathbf{x}$  and  $\boldsymbol{\theta}$  [2], and this essentially permits switching the order of integration and differentiation. This is the most commonly assumed regularity condition, although other scenarios can ensure regularity is satisfied. Under these assumptions, we have the following Cramér-Rao bound theorem (sometimes referred to as the *information inequality*) [1–4], that was independently developed by Cramér [5] and Rao [6].

Given appropriate regularity conditions on  $p(\mathbf{x}; \boldsymbol{\theta})$  as above, the variance of any unbiased estimator  $\mathbf{t}(\mathbf{x})$  of  $\boldsymbol{\theta}$  satisfies the inequality

$$\text{Var}(\mathbf{t}(\mathbf{x})) \geq \mathbf{I}^{-1}(\boldsymbol{\theta}), \quad (8.8)$$

if the inverse exists, and the variance is exactly  $\mathbf{I}^{-1}(\boldsymbol{\theta})$  if and only if

$$\mathbf{I}(\boldsymbol{\theta}) (\mathbf{t}(\mathbf{x}) - \boldsymbol{\theta}) = s(\mathbf{x}; \boldsymbol{\theta}) \quad (8.9)$$

(in the mean-square sense). We refer to the inverse of the Fisher information matrix as the Cramér-Rao bound and denote it as

$$\text{CRB}(\boldsymbol{\theta}) \triangleq \mathbf{I}^{-1}(\boldsymbol{\theta}), \quad (8.10)$$

where the CRB on the  $i$ th element of  $\boldsymbol{\theta}$  is given by the  $i$ th element of the diagonal of  $\mathbf{I}^{-1}(\boldsymbol{\theta})$ .

### 3.08.2.1.1 CRB on transformations of the parameters

The performance of estimation of a function of the parameters, e.g., the transformation  $\alpha = k(\theta)$ , is often of more interest than the performance of direct estimation of the parameters. Note that  $\alpha$  and  $\theta$  need not have the same dimension. If the Jacobian of the transformation function is  $K(\theta) = \left. \frac{\partial k(\theta')}{\partial \theta'^T} \right|_{\theta'=\theta}$ , then the CRB on the performance of an unbiased estimator  $t_\alpha(x)$  of  $\alpha$  is given in the following [1,4]. The variance of any unbiased estimator  $t_\alpha(x)$  of  $\alpha = k(\theta)$  satisfies the inequality

$$\text{Var}(t_\alpha(x)) \geq \text{CRB}(\alpha) \triangleq K(\theta) \text{CRB}(\theta) K^T(\theta) \quad (8.11)$$

with equality if and only if  $t_\alpha(x) - \alpha = K(\theta) I^{-1}(\theta) s(x; \theta)$  (in the mean-square sense). Equation (8.11) relates the CRB on  $\theta$  to the CRB on  $\alpha$  by incorporating the Jacobian of the function  $k$  that transforms  $\theta$  to  $\alpha$ . Implicit in (8.11) is that  $\alpha$  is differentiable with respect to  $\theta$  and (8.5) must also be satisfied for  $h(x) \equiv t_\alpha(x)$ .

### 3.08.2.1.2 The bias-informed CRB

The CRB theory above applies only to unbiased estimates. But suppose we have a particular biased estimator of  $\theta$ , given by  $t_b(x)$ , with bias given by  $b(\theta) = E_\theta[t_b(x) - \theta]$ . We can use (8.11) to find a bound on  $\alpha = \theta + b(\theta)$ . Because  $t_b(x)$  is an unbiased estimator of the function  $\alpha = k(\theta) = \theta + b(\theta)$ , it follows that

$$\text{Var}(t_b(x)) \geq \text{CRB}(\theta + b(\theta)), \quad (8.12)$$

where

$$\text{CRB}(\theta + b(\theta)) = (I_m + B(\theta)) \text{CRB}(\theta) (I_m + B^T(\theta)) \quad (8.13)$$

with  $B(\theta) = \left. \frac{\partial b(\theta')}{\partial \theta'^T} \right|_{\theta'=\theta}$ . We refer to (8.12) as the *bias-informed CRB*.

The bias-informed CRB only applies to the class of estimators with the specified bias gradient  $B(\theta)$ , and does not provide a general bound for an arbitrary biased estimator. Because of this the bias-informed CRB has limited use and has not found broad application. An approach that seeks to overcome this to some extent is the *uniform CRB*, that broadens the applicability of the bias-informed CRB to the class of estimators whose bias function is nearly constant over a region [7,8]. When a closed form is not available the bias-gradient can be computed numerically, at least for problems that have relatively few parameters [7].

### 3.08.2.1.3 A general CRB expression

The above results on parameter transformation and estimator bias are special cases of a combined general CRB expression, given by

$$\text{Var}(t(x)) \geq \text{CRB}(\alpha) = H(\theta) I^{-1}(\theta) H^T(\theta), \quad (8.14)$$

where  $t(x)$  is an estimator of  $\alpha = k(\theta)$  with bias  $b(\theta)$ ,  $I^{-1}(\theta)$  is the CRB matrix for estimating  $\theta$ , and

$$H(\theta) = K(\theta) + B(\theta) \quad (8.15)$$

is the sum of the Jacobian of the parameter transformation and the bias gradient. The previous expressions then follow as special cases. For example, the CRB on estimation of  $\alpha$  for unbiased estimators is given by

(8.11) with  $\mathbf{H}(\boldsymbol{\theta}) = \mathbf{K}(\boldsymbol{\theta})$  since  $\mathbf{b}(\boldsymbol{\theta}) = \mathbf{0}$ . Or, the CRB on estimation of  $\boldsymbol{\theta}$  with a biased estimator  $\mathbf{t}(\mathbf{x})$  whose bias is  $\mathbf{b}(\boldsymbol{\theta})$ , is given by (8.12) with  $\mathbf{H}(\boldsymbol{\theta}) = \mathbf{I} + \mathbf{B}(\boldsymbol{\theta})$  since there is no parameter transformation (i.e.,  $\mathbf{k}(\boldsymbol{\theta}) = \boldsymbol{\theta}$  is the identity transformation). When introducing the CRB, some texts begin with (8.10), whereas others begin with (8.14) or one of the variations.

### 3.08.2.1.4 Normal distributions

When the density function  $p(\mathbf{x}; \boldsymbol{\theta})$  is Gaussian, denoted  $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{\Sigma}(\boldsymbol{\theta}))$ , then the Cramér-Rao bound is completely characterized by the dependence on the parameter  $\boldsymbol{\theta}$  of the mean  $\boldsymbol{\mu}(\boldsymbol{\theta})$  and covariance  $\boldsymbol{\Sigma}(\boldsymbol{\theta})$ . Assuming differentiability with respect to  $\boldsymbol{\theta}$ , then the CRB can be evaluated in terms of otherwise arbitrary functions for the mean  $\boldsymbol{\mu}(\boldsymbol{\theta})$  and covariance  $\boldsymbol{\Sigma}(\boldsymbol{\theta})$ , and the CRB is given by the Slepian-Bangs formula [9, 10],

$$[\mathbf{I}(\boldsymbol{\theta})]_{ij} = \frac{\partial \boldsymbol{\mu}^T(\boldsymbol{\theta}')}{\partial \theta'_i} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}') \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta}')}{\partial \theta'_j} + \frac{1}{2} \text{tr} \left( \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}') \frac{\partial \boldsymbol{\Sigma}(\boldsymbol{\theta}')}{\partial \theta'_i} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}') \frac{\partial \boldsymbol{\Sigma}(\boldsymbol{\theta}')}{\partial \theta'_j} \right) \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}}. \quad (8.16)$$

There are many examples of application of (8.16), e.g., see Kay [4]. When the signal plus Gaussian noise model of (8.1) holds, then covariance  $\boldsymbol{\Sigma}$  is not a function of  $\boldsymbol{\theta}$ , and the second term in (8.16) vanishes.

### 3.08.2.1.5 Properties of the CRB

We discuss some properties and interpretation of the CRB.

- *CRB existence and regularity conditions:* An example where the CRB does not exist due to violation of the regularity conditions occurs when estimating the parameters of a uniform distribution. Specifically, suppose we have observations  $x(n)$  from a uniform distribution on  $[0, \theta]$ , and we seek the CRB on estimation of  $\theta$ . Now, the regularity conditions do not hold and the CRB cannot be applied. This is a classic text book example, see Kay [4, Problem 3.1].
- *The CRB is a local bound:* As we know from basic calculus, the second derivative corresponds to the curvature or concavity of the function at a point on the curve. Thus the CRB, evaluated at a particular value of the parameter  $\boldsymbol{\theta}$ , can be interpreted as the expected value of the curvature of the likelihood around the parameter. This is an important interpretation since the curvature measure is local, and consequently the CRB is often referred to as a local bound. We remark on this point again when considering the asymptotic behavior of the MLE in the next section. Because it is a local bound the CRB at a parameter value  $\boldsymbol{\theta}_0$  gives us a tight lower bound on the variance of any unbiased estimator of  $\boldsymbol{\theta}_0$  only when the estimation errors are small. Depending on the particular scenario, this might correspond to obtaining a large number of observations, or a high signal-to-noise ratio (SNR). This also means that the CRB can be a loose lower bound for the opposite cases, when the number of observations is low, or at low SNR.
- *Effect of adding parameters:* If an additional parameter is brought into the model, so that the dimension of  $\boldsymbol{\theta}$  grows by one, then the bound on the previously existing parameters will remain the same or rise. It will remain the same if the new parameter is decoupled from the previous parameters in the model. Or, if the parameters are coupled, the bound can only increase. Intuitively, this is because the introduction of a new unknown into the model can only increase uncertainty, implying more

difficulty in obtaining accurate parameter estimates, and hence a larger bound. The proof relies on linear algebra and is a standard textbook problem, e.g., see Kay [4, Problem 3.11].

- *Not necessarily a tight bound:* Generally there is no guarantee that an estimator exists that can achieve the CRB. More typically, under certain conditions the MLE can be shown to asymptotically achieve the CRB, but there is no guarantee this will occur with finite data; see the discussion below on the MLE in Section 3.08.3. Other bounds exist that may be tighter than the CRB in the non-asymptotic region, such as bounding the mean square error; see Section 3.08.4.
- *Identifiability:* Given the parametric model, we would like the distribution function to be distinguishable by the choice of the parameters. In other words, we want a given pdf  $p(\mathbf{x}; \boldsymbol{\theta})$  to be unique on some non-zero measurable set for different choices of  $\boldsymbol{\theta}$ . If not, e.g., supposing  $\boldsymbol{\theta}_1 \neq \boldsymbol{\theta}_2$  yet  $p(\mathbf{x}; \boldsymbol{\theta}_1) = p(\mathbf{x}; \boldsymbol{\theta}_2)$  for almost every  $\mathbf{x} \in \mathbb{X}$ , then we cannot uniquely infer anything about the parameters even with an infinite number of observations. This is the problem of *identifiability*. While there are various definitions and contexts for identifiability, here we can note that a fundamental requirement for the existence of the CRB is that the Fisher information matrix be full rank so that its inverse exists. Generally nonsingularity in the FIM evaluated at the true value of the parameter implies identifiability locally about the parameter [11], reflecting the fact that the CRB is a local bound governed by the local smooth behavior of the pdf. For example, for the Gaussian pdf case, Hochwald and Nehorai showed there exists a strong link between identifiability and nonsingularity in the FIM [12]. Consider scalar cases such as  $\mathbf{x} = a \cdot \mathbf{b} + \mathbf{v}$ , or  $\mathbf{x} = a + b + \mathbf{v}$ , where  $\mathbf{v}$  is additive Gaussian noise and we wish to find the CRB on estimation of  $a$  and  $b$ . For both examples we find that the FIM is not full rank, hence the CRB does not exist. Studying the models for the observation  $\mathbf{x}$ , it is apparent that we cannot uniquely identify the parameter(s). This situation can be resolved if there is additional information in the form of equality constraints on the parameters; see Section 3.08.6.
- *Biased estimators:* It can happen that the MLE is a biased estimator, even with infinite data. An interesting example is the estimation of the covariance of a multivariate normal distribution. Given  $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{\Sigma}(\boldsymbol{\theta}))$ , the maximum likelihood estimator for  $\boldsymbol{\Sigma}(\boldsymbol{\theta})$  is the sample covariance using observations of  $\mathbf{x}$ . However, it has been shown that this estimate is biased, and so the CRB does not bound the MLE in this case, e.g., see Chen et al. [13]. For more on biased estimators see Sections 3.08.2.1.2 and 3.08.4.
- *Estimation versus classification bounds:* The CRB bounds continuous estimation of a continuous parameter. In this case perfect (zero-error) estimation is not possible. Suppose instead the parameter is drawn from a finite set, e.g., we draw from a finite alphabet for communicating symbols. Now, the parameter is discrete valued, and perfect estimation is possible. In such a case the CRB goes to zero; it is uninformative. The problem is now one of classification, rather than estimation, and a classification bound is needed for performance analysis, such as bit error rate bounds in digital communications [14, 15].

### 3.08.3 Maximum likelihood estimation and the CRB

The CRB provides a lower bound on the variance of any unbiased estimator, subject to the regularity conditions on  $p(\mathbf{x}; \boldsymbol{\theta})$ . Next we consider the maximum-likelihood estimator, which has the remarkable

property that under some basic assumptions the estimate will asymptotically achieve the CRB, and therefore achieves asymptotic optimality.

The maximum likelihood approach to estimation of  $\theta$  chooses as an estimator  $t_{\text{ML}}(\mathbf{x}) = \hat{\theta}_{\text{ML}}$  that, if true, would have the highest probability (the maximum likelihood) of resulting in the given observations  $\mathbf{x}$ . Thus, the MLE results from the optimization problem

$$t_{\text{ML}}(\mathbf{x}) = \hat{\theta}_{\text{ML}} = \arg \max_{\theta} \log p(\mathbf{x}; \theta), \quad (8.17)$$

where for convenience  $\log p(\mathbf{x}; \theta)$  may be equivalently maximized since  $\log(\cdot)$  is a monotonic transformation. Previously we thought of  $\theta$  as fixed constants in the pdf  $p(\mathbf{x}; \theta)$ . Now, in (8.17), we are given observations  $\mathbf{x}$  and regard  $p(\mathbf{x}; \theta)$  as a function of  $\theta$ . In this case  $p(\mathbf{x}; \theta)$  is referred to as the *likelihood function*.

Using basic optimization principles, a solution of (8.17) follows by solving the equations resulting from differentiating with respect to the parameters and setting these equations equal to zero. The derivatives of the log-likelihood are the Fisher score previously defined in (8.4), so the first-order conditions for finding the MLE are solutions  $\hat{\theta}_{\text{ML}}$  of

$$s(\mathbf{x}; \theta') = \mathbf{0}. \quad (8.18)$$

Applying (8.18) requires smoothness in the parameters so that the derivatives exist. The parameters lie in a (possibly infinite) set, denoted  $\theta \in \Theta$ . Then, assuming the derivatives exist and provided  $\Theta$  is an open set,  $\hat{\theta}_{\text{ML}}$  will satisfy (8.18). Intuitively, the solution to (8.18) relies on the smoothness of the function in the parameters, and the parameters themselves do not have hard boundaries where the derivatives may fail to exist. This is similar to the assumptions needed for the existence of the CRB on  $\theta$ ; see the discussion in Section 3.08.2.1.

The principle of maximum likelihood provides a systematic approach to estimation given the parametric model  $p(\mathbf{x}; \theta)$ . If (8.18) cannot be solved in closed form, we can fall back on optimization algorithms applied directly to (8.17). Many sophisticated tools and algorithms have been developed for this over the past several decades. Very often the maximization is non-convex, although our smoothness conditions guarantee local convexity in some neighborhood around the true value for  $\theta$ , so algorithms often consist of finding a good initial value and then seeking the local maximum.

If the parameters take on values from a discrete set then the assumptions on the solution of (8.18) are violated. However, the MLE may still exist although estimation will generally rely directly on (8.17). Note that in this case we have a classification rather than an estimation problem: choose  $\theta$  from a discrete set of possible values that provides the maximum in (8.17). Consequently, the CRB is not applicable and performance analysis will rely on classification bounds; see the comment in Section 3.08.2.1.5.

### 3.08.3.1 Asymptotic normality and consistency of the MLE

As we noted, the MLE has a fundamental link with the CRB, as described next. Let the samples  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$  be iid observations from the pdf  $p(\mathbf{x}; \theta)$ . Denote  $\mathbf{y}_n = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$  to be the collection of these samples. Then, the likelihood is given by  $p(\mathbf{y}_n; \theta) = \prod_{i=1}^n p(\mathbf{x}_i; \theta)$ , and we denote the maximum likelihood estimate from these samples as  $t_{\text{ML}}(\mathbf{y}_n) = \hat{\theta}_{\text{ML}}(\mathbf{y}_n)$ .

Assuming appropriate regularity conditions on the pdf  $p(\mathbf{x}; \boldsymbol{\theta})$  hold, such as described above, then the MLE of the parameter  $\boldsymbol{\theta}$  is asymptotically distributed according to

$$\sqrt{n} (\hat{\boldsymbol{\theta}}(\mathbf{y}_n) - \boldsymbol{\theta}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathbf{I}^{-1}(\boldsymbol{\theta})), \quad (8.19)$$

where  $\mathbf{I}(\boldsymbol{\theta})$  is derived from the pdf  $p(\mathbf{x}; \boldsymbol{\theta})$ , i.e., it is the Fisher information of a single observation or sample, and  $\xrightarrow{d}$  denotes convergence in distribution.

This remarkable property states that the MLE is asymptotically normally distributed as observation size  $n$  goes to infinity, is unbiased so that  $\hat{\boldsymbol{\theta}}_{\text{ML}}(\mathbf{y}_n)$  converges to  $\boldsymbol{\theta}$  in the mean, and has covariance given by the CRB. Thus, the MLE is asymptotically optimal because the CRB is a lower bound on the variance of any unbiased estimator. Intuitively, the asymptotic normal distribution of the estimate occurs due to a central limit theorem effect. Note that only general regularity conditions (e.g., smoothness) are required on  $p(\cdot)$  for this to (eventually) be true. We discuss the central limit theorem more generally in Section 3.08.8.

Equation (8.19) shows that the MLE is consistent. A *consistent estimator* is one that converges in probability to  $\boldsymbol{\theta}$ , meaning that the distribution of  $\hat{\boldsymbol{\theta}}_{\text{ML}}(\mathbf{y}_n)$  becomes more tightly concentrated around  $\boldsymbol{\theta}$  as  $n$  grows. In the specific case of the MLE, as  $n$  grows the distribution of the estimator converges to a Gaussian distribution with mean given by the true parameter and covariance given by the CRB, and this distribution is collapsing around the true value as we obtain more observations. If the mean of  $\hat{\boldsymbol{\theta}}_{\text{ML}}$  were not equal to the true value of  $\boldsymbol{\theta}$ , then the estimator would be converging to the wrong answer. And, since the variance of the MLE converges to the CRB, we are assured that the MLE is asymptotically optimal, implying that as the data size becomes large no other estimator will have a distribution that collapses faster around the true value.

### 3.08.4 Mean-square error bound

As we have noted the CRB is only valid for unbiased estimators, whereas we may have an estimator that is biased. If the bias is known and we can compute its gradient then we can use the bias-informed CRB, but as we noted this is rarely the case. More generally we may be interested in analyzing the bias-variance tradeoff, for example with finite data an estimator may be biased even if it is asymptotically unbiased and approaches the CRB for large data size. This naturally leads to the mean-square error (MSE) of the estimator, given by

$$\text{MSE}(\mathbf{t}(\mathbf{x})) = E_{\boldsymbol{\theta}} \left[ (\mathbf{t}(\mathbf{x}) - \boldsymbol{\theta}) (\mathbf{t}(\mathbf{x}) - \boldsymbol{\theta})^T \right]. \quad (8.20)$$

The variance and the MSE of the estimator  $\mathbf{t}(\mathbf{x})$  are easily related through

$$\text{MSE}(\mathbf{t}(\mathbf{x})) = \text{Var}(\mathbf{t}(\mathbf{x})) + \mathbf{b}(\boldsymbol{\theta}) \mathbf{b}^T(\boldsymbol{\theta}), \quad (8.21)$$

and they are equivalent in the unbiased case, so that if  $\mathbf{t}(\mathbf{x})$  is unbiased then the CRB bounds both the variance and the MSE. If the estimator remains biased even as the observation size grows, then the MSE may become dominated by the bias error asymptotically.

Study of the MSE can be valuable when relatively fewer observations are available. In this regime an estimator may trade off bias and variance, with the goal of minimizing the MSE. It may be tempting to say that a particular biased estimator is “better” than the CRB when  $\text{MSE}(\mathbf{t}(\mathbf{x})) < \text{Var}(\mathbf{t}(\mathbf{x}))$ , but

the CRB only bounds unbiased estimators; it is better to refer to Eqs. (8.20) or (8.21) that quantify the relationship. Ultimately, as the number of observations grows, then it is desired to have the estimate become unbiased and the variance of the estimator diminish, otherwise the estimator will asymptotically converge to the wrong answer.

The MSE and bias-variance tradeoff is also valuable for choosing a particular model, e.g., see Spall [16, Chapter 12].

In Section 3.08.2.1.5, when discussing biased estimators, we noted that the MLE for the covariance of a multivariate normal distribution is just the sample covariance, and that in this case the MLE is a biased estimator. In addition, the sample covariance is not a minimum MSE estimator for this problem, in the sense of minimizing the expected Frobenius norm squared of the error matrix. In fact, a lower MSE can be obtained by the James-Stein estimator, that uses a shrinkage-regularization estimation technique [13, 17]. This example illustrates that the maximum likelihood estimator is not necessarily guaranteed to achieve the minimum MSE among biased or unbiased estimators.

## 3.08.5 Perturbation methods for algorithm analysis

Perturbation methods provide an approach to small-error analysis of algorithms. This is a close relative of using a Taylor expansion and dropping higher order terms. Consequently, perturbation analysis is accurate for larger data size, and so can be thought of as an asymptotic analysis, and it will generally predict asymptotic performance. The idea has been broadly applied, especially for cases when the estimation algorithm is a highly nonlinear function of the data and/or the model parameters. An important application area occurs when algorithms incorporate matrix decompositions, such as subspace methods in array processing. However, we emphasize that the ideas are general and can be readily applied in many scenarios.

### 3.08.5.1 Perturbations and statistical analysis

The general idea is as follows. A statistical perturbation error analysis consists of two basic steps. The first is a deterministic step, adding small perturbations into the model at the desired place. The perturbations can be applied to the original data such as occurs with additive noise. But, the perturbations can also be applied at other stages, such as perturbing the estimate itself. The perturbed model is then manipulated to obtain an expression for the estimation error as a function of the perturbations, where (typically) only the first order perturbation terms are kept, and terms that involve higher order functions of the perturbation are dropped under the assumption that they are negligible. The second step is statistical. We now assume the perturbations are random variables drawn from a stationary process with finite variance (typically zero-mean), and proceed to evaluate the statistics of the estimation error expression. This typically involves the mean and variance of the first-order perturbed estimation error expression.

Let  $\hat{\theta}$  be an estimator of parameters  $\theta$ , which we write as  $\hat{\theta} = \theta + \Delta\theta$ , where  $\Delta\theta$  is a small perturbation. Therefore, we have the simple estimation error expression

$$\Delta\theta = \hat{\theta} - \theta. \quad (8.22)$$

To quantify the error in the estimator, we wish to find quantities such as the mean and variance of the estimation error expression. Thus we treat  $\Delta\theta$  as samples from a stationary random process, commonly

assumed to be zero-mean independent and identically distributed; it is not necessary to specify the pdf of the perturbation. The important point is that when deriving (8.22) for the specific estimator of interest we make use of the small error assumption and keep only first-order terms in  $\Delta\theta$ . Derivation of (8.22) is the primary hurdle to this analytical approach, so that we can evaluate  $E_{\Delta\theta}[\hat{\theta} - \theta]$  and  $\text{Var}_{\Delta\theta}(\hat{\theta} - \theta)$ . If  $E_{\Delta\theta}[\hat{\theta} - \theta] = 0$  then the algorithm is unbiased for small error (this may mean the algorithm is asymptotically unbiased).

The perturbation method is appealing when an algorithm is biased. If  $E_{\Delta\theta}[\hat{\theta} - \theta] \neq 0$  then we can use it with  $\text{Var}_{\Delta\theta}(\hat{\theta} - \theta)$  to find the MSE in Eq. (8.21). This enables study of the bias-variance tradeoff, for cases where the CRB does not apply due to the estimator bias. For example, the perturbation method has been applied to geolocation based on biased range measurements, such that the estimation of the source location is inherently biased [18].

While the accuracy of first order perturbation analysis is generally limited to the high SNR and/or large data regime, the perturbation method can be broadened by considering second-order terms, as developed by Xu for subspace methods [19]. The resulting error expressions now contain more terms, but the resulting analyses are more accurate over a larger range of SNR and data size (roughly speaking, the second-order analyses are accurate for moderate to large SNR or data size).

### 3.08.5.2 Matrix perturbations

As we noted an important application of these ideas is for algorithms that involve matrix decompositions. These algorithms are typically not very amenable to error analysis because the estimation error expression becomes highly nonlinear in the parameters of interest. Instead, we can apply a perturbation to the matrix and then study the effect of the perturbation after the matrix decomposition.

Suppose we have an  $m \times n$  matrix  $A = A(\theta)$ , that is input to an algorithm for estimating some parameters  $\theta$ . Often an algorithm includes an eigendecomposition of  $A$  (for  $m = n$ ), or more generally with a singular value decomposition (SVD) of  $A$  (for  $m \neq n$ ). To carry out the first step in the perturbation based analysis, we would like to express the matrix decomposition as a function of a perturbation on  $A$ .

Let  $\bar{A} = A + \Delta A$  be a perturbed version of  $A$ , where  $\Delta A$  is small. Regarding  $\Delta A$  as deterministic, Stewart developed expressions for many important cases, carrying the perturbations along. These include the eigenvalues of  $A$ , the singular values of  $A$ , the subspaces of  $A$ , as well as the pseudo-inverse of  $A$ , projections, and linear least squares algorithms [20–22]. These expressions generally rely on keeping only first-order terms in  $\Delta A$ , and dropping higher-order terms under the assumption that  $\Delta A$  is small, where here small is meant in the sense of some appropriate matrix norm [22].

If an estimation algorithm is a function of  $A$  and involves subspace decomposition say, then we can use the expressions from Stewart in our approach. Given the perturbed subspace, we then find the estimation error expression, that is now a function of  $\Delta A$ , again keeping only the first order terms. The analysis is then completed by finding the statistical properties of the error, such as mean and variance.

Many array processing methods rely on subspace decomposition and eigenvalue expressions [23], and consequently the perturbation ideas have been broadly applied to develop small error analysis [24]. Perturbations can be applied to the data, or to the estimated spatial covariance matrix, for example [19]. These applications typically use complex values, and the perturbation ideas readily go over from the real-valued to the complex-valued case.

### 3.08.5.3 Obtaining the CRB through perturbation analysis of the MLE

An important special case is the application of perturbation analysis to the MLE. Now, under the conditions for which the MLE is consistent, i.e., is asymptotically unbiased and attains the CRB, then the perturbation approach can yield the CRB by finding the variance of the estimation error expression as above. This follows because we are carrying out a small error (asymptotic) analysis for an algorithm that is asymptotically guaranteed to achieve a variance that is equal to the CRB. For examples, see [18, 25].

---

## 3.08.6 Constrained Cramér-Rao bound and constrained MLE

In this section we show how the classic parametric statistical modeling framework from Sections 3.08.2 and 3.08.3, including the CRB and MLE, can be extended by incorporating additional information in the form equality constraints. We make the same assumptions as in Section 3.08.2, given observations  $\mathbf{x} \in \mathbb{X} \subset \mathcal{R}^n$  from a probability density function  $p(\mathbf{x}; \boldsymbol{\theta})$  where  $\boldsymbol{\theta} \in \Theta \subset \mathcal{R}^m$  is a vector of unknown deterministic parameters. Suppose now that these parameters satisfy *k* *consistent* and *nonredundant* continuously differentiable parametric equality constraints. These constraints can be expressed as a vector function  $\mathbf{f}(\boldsymbol{\theta}) = \mathbf{0}$  for some  $\mathbf{f} : \Theta \rightarrow \mathcal{R}^k$ . Equivalently, the constraints can also be stated  $\boldsymbol{\theta} \in \Theta_f$ , i.e., the equality constraints act to restrict the parameters to a subset of the original parameter space. The constraints being consistent means that the set  $\Theta_f$  is nonempty, and the constraints being nonredundant means that the Jacobian  $\mathbf{F}(\boldsymbol{\theta}') = \frac{f(\boldsymbol{\theta}')}{\partial \boldsymbol{\theta}'^T}$  has rank *k* whenever  $\mathbf{f}(\boldsymbol{\theta}') = \mathbf{0}$ .

The presence of the constraints provides further specific additional information about the model  $p(\mathbf{x}; \boldsymbol{\theta})$ , expressed functionally as  $\mathbf{f}(\boldsymbol{\theta}') = \mathbf{0}$ . Equivalently, in addition to obeying  $p(\mathbf{x}; \boldsymbol{\theta})$  the parameters are also restricted to live in  $\Theta_f$ . Intuitively, we can expect that more accurate estimation should be possible, and that we can incorporate the constraints into both performance bounds and estimators. This is the case, and we detail the extension of the CRB and MLE to incorporate the constraints in the following.

### 3.08.6.1 Constrained CRB

The following relates the Fisher information  $\mathbf{I}(\boldsymbol{\theta})$  for the original unconstrained model  $p(\mathbf{x}; \boldsymbol{\theta})$ , to a new bound on  $\boldsymbol{\theta}$  that incorporates the constraints [26, Theorem 1]. Let  $\mathbf{t}(\mathbf{x})$  be an unbiased estimator of  $\boldsymbol{\theta}$ , where in addition equality constraints are imposed as above so that  $\boldsymbol{\theta} \in \Theta_f$ , or equivalently  $\mathbf{f}(\boldsymbol{\theta}) = \mathbf{0}$ . Then

$$\text{Var}(\mathbf{t}(\mathbf{x})) \geq \text{CCRB}(\boldsymbol{\theta}) \triangleq \mathbf{U}(\boldsymbol{\theta}) \left( \mathbf{U}^T(\boldsymbol{\theta}) \mathbf{I}(\boldsymbol{\theta}) \mathbf{U}(\boldsymbol{\theta}) \right)^{-1} \mathbf{U}^T(\boldsymbol{\theta}), \quad (8.23)$$

where  $\mathbf{U}(\boldsymbol{\theta})$  is a matrix whose column vectors form an orthonormal basis for the null space of the Jacobian  $\mathbf{F}(\boldsymbol{\theta})$ , i.e.,

$$\mathbf{F}(\boldsymbol{\theta}) \mathbf{U}(\boldsymbol{\theta}) = \mathbf{0}, \quad \mathbf{U}^T(\boldsymbol{\theta}) \mathbf{U}(\boldsymbol{\theta}) = \mathbf{I}_{(m-k) \times (m-k)}. \quad (8.24)$$

In (8.24),  $\mathbf{I}_{(m-k) \times (m-k)}$  is an  $(m - k) \times (m - k)$  identity matrix (not to be confused with the Fisher information matrix  $\mathbf{I}(\boldsymbol{\theta})$ ). Equation (8.23) defines the *constrained Cramér-Rao bound* (CCRB) on  $\boldsymbol{\theta}$ . Equality in the bound is achieved if and only if  $\mathbf{t}(\mathbf{x}) - \boldsymbol{\theta} = \text{CCRB}(\boldsymbol{\theta}) \mathbf{s}(\mathbf{x}; \boldsymbol{\theta})$  (in the mean-square sense), where  $\mathbf{s}(\mathbf{x}; \boldsymbol{\theta})$  is the Fisher score defined in (8.4). This is similar to the condition for achieving equality with the CRB noted earlier.

The CCRB can be extended to bound transformations of parameters, just as for the CRB in Section 3.08.2.1.1. Consider a continuously differentiable function of the parameters  $\mathbf{k} : \Theta_f \rightarrow \mathbb{R}^q$  and denote its Jacobian as  $\mathbf{K}(\boldsymbol{\theta}) = \frac{\partial \mathbf{k}(\boldsymbol{\theta}')}{\partial \boldsymbol{\theta}^T} \Big|_{\boldsymbol{\theta}'=\boldsymbol{\theta}}$ . Given constraints  $\mathbf{f}(\boldsymbol{\theta}) = \mathbf{0}$ , then the variance of any unbiased estimator  $\mathbf{t}(\mathbf{x})$  of  $\boldsymbol{\alpha} = \mathbf{k}(\boldsymbol{\theta})$  satisfies the inequality

$$\text{Var}(\mathbf{t}(\mathbf{x})) \geq \text{CCRB}(\boldsymbol{\alpha}) \triangleq \mathbf{K}(\boldsymbol{\theta}) \text{CCRB}(\boldsymbol{\theta}) \mathbf{K}^T(\boldsymbol{\theta}). \quad (8.25)$$

This extends the CCRB on  $\boldsymbol{\theta}$  to the parameters  $\boldsymbol{\alpha} = \mathbf{k}(\boldsymbol{\theta})$ , and has a form similar to (8.11).

### 3.08.6.2 Comments and properties of the CCRB

- *Existence and regularity conditions:* It is important to recognize that the CCRB relies on the Fisher information  $\mathbf{I}(\boldsymbol{\theta})$  expressed through the probability density  $p(\mathbf{x}; \boldsymbol{\theta})$ . We have not altered the construction of the Fisher score or FIM in (8.4) and (8.3). The constraints are regarded as additional side information, resulting in the CCRB.

Note that the CCRB does *not* require  $\mathbf{I}(\boldsymbol{\theta})$  be nonsingular, but does require that  $\mathbf{U}^T(\boldsymbol{\theta})\mathbf{I}(\boldsymbol{\theta})\mathbf{U}(\boldsymbol{\theta})$  be nonsingular. Thus the addition of constraints can lead to a meaningful bound on estimation even when the original model is not identifiable. For example, constraints can lead to bounds on blind estimation problems [26,27].

- *Constrained estimators versus constrained parameters:* Stoica and Ng actually developed a bound for an unbiased, constrained estimator, as opposed to a bound for an unbiased estimator of constrained parameters [26]. This is an important distinction since, in general, a parameter and its unbiased estimator will not simultaneously satisfy a constraint. The conditions for when this does occur, i.e., when  $\mathbf{f}(E_{\boldsymbol{\theta}}\mathbf{t}(\mathbf{x})) = E_{\boldsymbol{\theta}}\mathbf{f}(\mathbf{t}(\mathbf{x}))$ , are the equality conditions for Jensen's inequality. Many of the citations of Stoica and Ng mistakenly apply the bound to the case of constrained parameters. Fortunately, the result under either assumption is the same CCRB expression, even if the interpretation is different [28].
- *Reparameterizing the model:* An equally valid approach to bounding estimation performance is to reparameterize the pdf in a way that incorporates the constraints, resulting in a distribution  $p(\mathbf{x}; \boldsymbol{\theta}')$  with new parameter vector  $\boldsymbol{\theta}'$ . Then the Fisher information  $\mathbf{I}(\boldsymbol{\theta}')$  can be obtained for the reparameterized model, and the CRB results in Section 3.08.2 can be applied to bound estimation of  $\boldsymbol{\theta}'$  or functions of  $\boldsymbol{\theta}'$ . Obviously  $\mathbf{I}(\boldsymbol{\theta}') \neq \mathbf{I}(\boldsymbol{\theta})$ , and it often happens that the dimension of  $\boldsymbol{\theta}'$  is less than the dimension of  $\boldsymbol{\theta}$  because the constraints amount to a manifold reduction by requiring  $\boldsymbol{\Theta} \in \Theta_f$  [29]. However, reparameterizing the model into  $p(\mathbf{x}; \boldsymbol{\theta}')$  can often be analytically impractical or numerically complex. The CCRB offers an alternative way to incorporate parametric equality constraints into the CRB and, for example, enables comparison of a variety of constraints in a straightforward way.
- *Inequality constraints:* The CCRB directly incorporates equality constraints. In some problems, the parameters are specified to obey inequality constraints. In such a case, the conventional CRB applies so long as the specified value of  $\boldsymbol{\theta}$  obeys the inequality, and so is a feasible value.
- *Evaluating the CCRB:* Evaluating the CCRB requires evaluating  $\mathbf{F}(\boldsymbol{\theta})$ , which is typically straightforward analytically, and then finding the nullspace basis  $\mathbf{U}(\boldsymbol{\theta})$  (and note that  $\mathbf{U}(\boldsymbol{\theta})$  is not unique). The matrix  $\mathbf{U}(\boldsymbol{\theta})$  may be found analytically, or it can be readily computed numerically, e.g., using standard software packages.

### 3.08.6.3 Constrained MLE

Just as the CRB is intimately related with maximum-likelihood estimation, the CCRB is related with constrained maximum-likelihood estimation (CMLE). The CMLE  $\hat{\theta}_{\text{CML}}$  solves the constrained optimization problem given by

$$\begin{aligned} & \arg \max_{\theta'} \log p(\mathbf{x}; \theta') \\ \text{s.t. } & \mathbf{f}(\theta') = \mathbf{0}. \end{aligned} \quad (8.26)$$

We can also show that given appropriate smoothness conditions,  $\hat{\theta}_{\text{CML}}$  converges in distribution to  $\mathcal{N}(\theta, \text{CCRB}(\theta))$ . Thus, asymptotically, the CMLE becomes normally distributed with mean equal to the true parameter and covariance given by the CCRB. This is analogous to the classic results for the MLE in (8.19). For derivations and details, including CMLE cast as a constrained scoring algorithm, see Moore et al. [30].

## 3.08.7 Multiplicative and non-Gaussian noise

In this section we consider two broad generalizations of the signal plus additive noise model in (8.1), multiplicative noise on the signal, and allowing the noise distributions to be non-Gaussian. For clarity we focus on the one-dimensional time series case, and show general CRB results. This model is broadly applicable and widely studied in many disciplines, for both man-made and naturally occurring signals.

### 3.08.7.1 Multiplicative and additive noise model

The signal and noise observation model in one dimension is given by

$$y(n) = m(n)s(n; \theta) + w(n), \quad n = 0, \dots, N - 1, \quad (8.27)$$

where  $s(n; \theta)$  is a signal parameterized by  $\theta$ , and  $m(n)$  and  $w(n)$  are stationary random processes. We denote  $E[m(n)] = \mu_m$  and  $\text{Var}\{m(n)\} = \sigma_m^2$ , and similarly for  $w(n)$  we have  $\mu_w$  and  $\sigma_w^2$ . With  $m(n) \equiv 1$  we revert to (8.1).

Including  $m(n)$  in the model introduces random amplitude and phase fluctuations into the signal, and so is often referred to as *multiplicative noise* [31]. Some important applications include propagation in a randomly fluctuating medium such as aeroacoustic signals in a turbulent atmosphere [32], underwater acoustics [33], radar Doppler spreading [34], and radio communications fading due to the superposition of multipath arrivals (such as Rayleigh-Ricean distributions) [15]. The model can be generalized to  $y(n) = h(n) * s(n; \theta) + w(n)$ , where  $h(n)$  is the impulse response of a linear filter and  $*$  represents convolution, and  $h(n)$  can be modeled as having random elements.

The rate of variation in  $m(n)$  can be slow or fast relative to the signal bandwidth and the observation time. A constant but random complex amplitude is an important special case of time-varying multiplicative noise, e.g., so-called flat fading, such that each block realization of  $y(n)$ ,  $n = 0, 1, \dots, N - 1$ , has a scalar random coefficient  $m(n) \equiv m$ . At the other extreme, fast variation in  $m(n)$  can play havoc in applications, e.g., fast fading in communications, that can yield dramatic signal distortion and thus can significantly degrade estimation of  $\theta$ .

### 3.08.7.2 Gaussian case

A Gaussian model for  $m(n)$  is reasonable for many applications, such as those mentioned above. Propagation distortion is often the result of many small effects, culminating in a Gaussian distribution. It is typically assumed that  $m(n)$  and  $w(n)$  are independent, arising from separate physical mechanisms, such as fluctuation caused by propagation combined with additive sensor noise.

Suppose  $m(n)$  and  $w(n)$  are iid Gaussian. Now  $y(n)$  is Gaussian and non-stationary, with mean and variance given by

$$E[y(n)] = \mu_y(n) = \mu_m s(n; \boldsymbol{\theta}) + \mu_w \quad (8.28)$$

and

$$\sigma_y^2(n) = \sigma_m^2 s^2(n; \boldsymbol{\theta}) + \sigma_w^2. \quad (8.29)$$

With  $E[m(n)] = 0$  then  $y(n)$  becomes stationary in the mean, but  $\mu_m = 0$  is generally undesirable, and from the standpoint of detection and estimation of  $s(n; \boldsymbol{\theta})$  large  $|\mu_m|$  is much preferred. It is often assumed that  $\mu_w = 0$ , which is reasonably safe because in practice  $\mu_w$  can be estimated and subtracted from  $y(n)$ . However, it should not be assumed that the multiplicative noise has mean  $\mu_m = 0$  if in fact it does not.

For real-valued  $y(n)$  several cases of the FIM and CRB are derived by Swami in [35]. In the iid Gaussian case, with  $\mu_w = 0$ , then the parameter vector of size  $p + 4$  is given by

$$\boldsymbol{\theta} = [\boldsymbol{\theta}_s, \sigma_m^2, \sigma_w^2, \mu_m], \quad (8.30)$$

where  $\boldsymbol{\theta}_s$  are the  $p + 1$  signal parameters from  $s(n; \boldsymbol{\theta}_s)$ . Now, the elements of the Fisher information matrix  $\mathbf{I}_{ij}$ ,  $i, j = 0, 1, \dots, p + 3$  are given by [35, Eqs. (4) and (5)]. Several foundational results are available in [35], including a polynomial signal model for  $s(n; \boldsymbol{\theta})$ , as well as the iid non-Gaussian noise case (see Section 3.08.7.3).

Specific signal models and estimators that have been addressed for the multiplicative and additive noise model include non-linear least squares estimation of sinusoids [36, 37], the CRB for change point detection of a step-like signal [38], and Doppler shift [34, 39]. Estimation of a single harmonic in multiplicative and additive noise arises in communications, radar, and other applications and has been studied extensively (e.g., see references in the above citations).

### 3.08.7.3 Non-Gaussian case

Next we consider the case when  $m(n)$  and  $w(n)$  in (8.27) may be non-Gaussian. While AWGN may always be present to some degree there are times when the central limit theorem is not applicable, such as a structured additive interference that is significantly non-Gaussian, and in such a case we might rely on a non-Gaussian pdf model for  $w(n)$ . For example, some electromagnetic interference environments are well modeled with a non-Gaussian pdf that has tails significantly heavier than Gaussian [40]. The study of the heavy-tailed non-Gaussian case is also strongly motivated to model the presence of outliers, leading to robust detection and estimation algorithms [41].

Let  $p_m(m)$  and  $p_w(w)$  denote the respective pdf's of  $m(n)$  and  $w(n)$  in (8.27). Closed forms for  $p_y(y)$  are typically not available, and it is difficult to obtain general results for the case when both  $m(n)$  and  $w(n)$  are present and at least one of them is non-Gaussian. However, following Swami [35], we can at least treat them separately as follows.

Consider (8.27) with  $m(n)$  a random constant so that  $E[m(n)] = \mu_m$ , and  $\text{Var}\{m(n)\} = \sigma_m^2 = 0$ . Let  $p_w(w)$  be a symmetric, not necessarily Gaussian pdf, and assume that  $w(n)$  is iid. Now, the FIM for  $\boldsymbol{\theta}$  has elements given by [35, Eq. (21)]

$$I_{ij} = \gamma_{w0} \frac{\mu_m^2}{\sigma_w^2} \sum_{n=0}^{N-1} \frac{\partial s(n; \boldsymbol{\theta})}{\partial \theta_i} \frac{\partial s(n; \boldsymbol{\theta})}{\partial \theta_j}, \quad i, j = 0, 1, \dots, p, \quad (8.31)$$

where  $\dim(\boldsymbol{\theta}) = p + 1$ ,  $\theta_i$  is the  $i$ th element of  $\boldsymbol{\theta}$ , and we define [35, Eq. (20)]

$$\gamma_{xk} = \frac{1}{\sigma_x^{k-2}} \int \left[ \frac{dp_x(x)}{dx} \right]^2 \frac{(x - \mu_x)^k}{p_x(x)} dx, \quad (8.32)$$

where  $x$  indicates the random variable with pdf  $p_x(x)$ , and  $k$  is integer with  $k = 0$  in (8.31). Thus the FIM  $\mathbf{I}$  depends on  $p_w(w)$  through its variance  $\sigma_w^2$  and the value of  $\gamma_{w0}$ . In the Gaussian case  $\gamma_{w0} = 1$ , and (8.31) is well known, e.g., see Kay [42, Chapter 3]. For any symmetric  $p_x(x)$  it follows that  $\gamma_{x0} \geq 1$ , so that non-Gaussian  $p_w(w)$  increases the Fisher information (i.e., lowers the CRB), and therefore AWGN is the worst case for estimating  $\boldsymbol{\theta}$ . For example, the CRB for estimation of the parameters of an autoregressive process is minimized when the process is Gaussian [43]. On the other hand, it is often the case in practice with non-Gaussian noise that the exact form of the pdf is not known, so it may be necessary to estimate both the signal parameters and the parameters of a non-Gaussian pdf model.

The impact of non-Gaussian  $p_m(m)$  can be studied by setting  $\sigma_w^2 = 0$ . This purely multiplicative noise case has Fisher information given by [35, Eq. (25)]

$$I_{ij} = \left[ \gamma_{m2} + \frac{2\mu_m \gamma_{m1}}{\sigma_m} + \frac{\mu_m^2 \gamma_{m0}}{\sigma_m^2} - 1 \right] \sum_{n=0}^{N-1} \frac{\partial s(n; \boldsymbol{\theta})}{\partial \theta_i} \frac{\partial s(n; \boldsymbol{\theta})}{\partial \theta_j} \frac{1}{s^2(n; \boldsymbol{\theta})}. \quad (8.33)$$

The FIM now depends on  $p_m(m)$  entirely through the first bracketed term in (8.33). This term is a function of the ratio  $\mu_m/\sigma_m$ , and  $\gamma_{mk}$ ,  $k = 0, 1, 2$ . Note that information is lost when  $\mu_m = 0$ , and that large  $\mu_m/\sigma_m$  is desired to increase the FIM. Intuitively, when  $\mu_m \gg \sigma_m$ , then the observed random fluctuation in  $y(n)$  is reduced. When  $m(n)$  is Gaussian, then  $\gamma_{m1} = 0$ ,  $\gamma_{m2} = 3$ .

The preceding was limited to iid processes. The more general non-iid case is typically not very analytically tractable, because with dependence in  $m(n)$  or  $w(n)$  the joint pdf on all the observations is required, unlike the Gaussian case where second-order statistics are sufficient to characterize dependence. An exception is linear non-Gaussian processes, obtained as the output of a linear filter driven by an iid non-Gaussian random process. Here it may be of interest to identify the filter parameters [43], and if the filter is invertible then the problem can be returned to the iid case by a linear operation (whitening) [44, 45]. Further discussion of non-iid issues in the non-Gaussian case is given by Poor and Thomas [46].

In our discussion around (8.27) we have largely focused on the signal parameters, but when we consider the non-Gaussian case we can also draw on the rich selection of parameterized non-Gaussian distributions. Well known families include Gaussian mixtures (GM) [47–50], stable distributions [51], and others [52].

The stable distributions include the normal distribution as a special limiting case, but otherwise have infinite variance. For almost all values of the stable distribution parameters simple closed form

expressions are not available, with the notable exception of the Cauchy pdf. Consequently specialized tools and estimators have been developed for the case when  $w(n)$  is drawn from the stable distribution family, such as fractional lower order statistics, in order to accommodate the infinite variance of the pdf. For the case of linear (or iid) stable processes, normalized versions of conventional covariance and cumulant estimates are convergent and enable the use of many estimation algorithms originally developed for AWGN, e.g., covariance matrix based subspace algorithms [53, 54].

CRBs are generally available for the cases mentioned here with additive iid non-Gaussian noise, using the expressions in [35]. Several cases are described in [50, 55], generalizing the AWGN cases to non-Gaussian noise.

### 3.08.8 Asymptotic analysis and the central limit theorem

The CRB analysis above appeals to asymptotics, in the sense that the bound is generally tight only asymptotically, and application of the CRB requires a fully specified parametric probability model. However, we can often employ a more general asymptotic analysis without a parametric model, and without completely specifying the pdf, by resorting to the central limit theorem (CLT) and related ideas. This approach is generally useful for analysis of algorithms. And, if there is an underlying parametric model, then we can compare algorithm performance analysis with parameter estimation bounds such as the CRB. However, the asymptotic tools are general and apply even when there is no underlying parametric model. We consider the application of the CLT to two broad classes of algorithms in Section 3.08.9.

Given an estimation algorithm, a full statistical analysis requires the pdf of the estimate, as a function of the observations. If we know this pdf we can find meaningful measures of performance such as the mean, variance, and MSE, and compare these to benchmark bounds. We would like to do this as a function of the number of observations (data size), as well as other parameters such as the SNR. However, even if we have full statistical knowledge of the observations, it may be intractable to find the pdf of the estimates due to the often complicated and nonlinear estimation function.

An alternative approach is to appeal to asymptotics as the observation size grows [56]. The basic tools are two probability limit laws, (i) the law of large numbers (LLN) and (ii) the central limit theorem. The LLN reveals when a sample average of a sequence of  $n$  random variables converges to the expected mean as  $n$  goes to infinity, and the CLT tells us about the distribution of the sample average (or more generally about the distribution of a normalized sum of random variables).

The basic CLT theorem states that the sample average asymptotically converges in distribution to normal,

$$\frac{1}{n} \sum_{i=1}^n x_i \xrightarrow{d} \mathcal{N}(\mu, \sigma^2/n), \quad (8.34)$$

where  $x_i$  are iid,  $E[x_i] = \mu$ , and  $\text{Var}\{x_i\} = \sigma^2 < \infty$ . There are many extensions, such as the vector case, when the  $x_i$  are non-identically distributed, and when the  $x_i$  are not independent. A generalized version of the CLT also exists for the non-Gaussian case with convergence to a stable distribution [57].

Intuitively, when summing random variables, we recognize that the sum of independent random variables has pdf given by the convolution of the individual pdfs, and the repeated convolution results in smoothing to a Gaussian distribution. For example, suppose we average  $n$  random variables that are iid from a uniform distribution on  $[0, 1]$ . In this case  $n = 10$  is sufficient to provide a very good

fit to a normal distribution (up to the finite region of support of the ten-fold convolution) [58], so the asymptotics may be effective even for relatively small  $n$  in some cases.

A physical intuition is also broadly applicable, that a combination of many small effects yields Gaussian behavior on the macro-scale, such as thermal noise in electronic devices resulting from electron motion.

### 3.08.8.1 Comments on application of the CLT

- *Bounding asymptotic deviation from normality:* The deviation of the asymptotic distribution from a normal pdf can be bounded. Perhaps the best known result is the Berry-Esséen theorem [56], that bounds the maximum deviation from Gaussian. The bound is proportional to  $\gamma = E[|x_i - \mu|^3]$ , the third central moment.  $\gamma$  is a measure of skewness of a pdf, i.e., it measures asymmetry in the pdf. If  $\gamma < 0$  then the pdf is skewed left (e.g., has a heavier tail on left), and vice-versa. Generally then, asymmetry in the pdf of  $x$  will slow the asymptotic convergence to Gaussian.
- *Deviation from normality is generally worse in the tails of the pdf:* The approximation to a normal process is generally more accurate when closer to the peak of the normal distribution, and may require a very large  $n$  for the fit to be accurate in the tails of the distribution. The fit between the distribution of the sample mean and its normal approximation worsens, at a given  $n$ , as we go out in the tails. This is intuitively evident in the example mentioned above, summing uniformly distributed random variables, because the convolution of multiple uniform pdfs will always have a finite support beyond which the true pdf is zero and so will always have error with respect to a normal distribution with its infinite support. Consequently, some care is needed when inferring from the tail behavior of the CLT Gaussian approximation. This is especially true in considering extreme tail behavior based on the form of the normal approximation, such as deriving tests of significance, confidence intervals (error bars), and rare events.
- *Applying the CLT to arbitrary functions:* The various statements of the CLT usually stem from a (perhaps weighted) linear combination of random variables. However, in practice we are interested in applying the asymptotic theorem with empirical observations from an arbitrary function of the observations of the random variables, such as an estimator  $t(x)$ . The CLT ideas are generally useful in this context when the estimator function stabilizes as  $n$  grows, i.e., given a sequence of random observations the estimator asymptotically converges to some value. If for large enough  $n$  then  $t(x)$  is reasonably Gaussian we can find its mean and variance, and the resulting normal distribution is a useful description of the pdf of the estimator. Note that, if we have a parametric model, then if  $\hat{\theta} = t(x)$  is asymptotically normally distributed, is unbiased, and the variance approaches the CRB, then this estimator has asymptotic optimality. However, we stress that the CLT ideas are very general and don't require an underlying parametric model.
- *Tests for Gaussianity:* Various statistical tests exist to determine if a random variable is well described by a Gaussian pdf, and these can and should be applied to verify the CLT effect in a particular case. For example, tests can be based on sample moments such as comparing the computed skewness and kurtosis with those expected for a normal distribution, as well as comparison through plots such as the Q-Q and P-P plots. Issues with such tests include obtaining sufficient data to accurately characterize the tail behavior, and that the test will provide some level of statistical evidence but not certainty.

- *Proving convergence to normality:* Given an algorithm, we can rely on simulation and tests for Gaussianity to determine the validity of the asymptotic approximation. In general, it can be relatively difficult to prove convergence of the distribution of the output of a particular algorithm or estimator to normality. One approach is to show that the skewness and kurtosis converge to their expected values under a normal assumption (e.g., the skewness will approach zero because the normal distribution is symmetric). An example of this form of analysis is given by Brillinger, demonstrating that sample estimates of higher order statistics become normally distributed [59].

### 3.08.9 Asymptotic analysis and parametric models

In the previous section we noted that asymptotic analysis of nonparametric algorithms is often facilitated by the law of large numbers and results associated with the central limit theorem. These ideas apply equally well to parameter estimators, even if we have the full parametric model specified in  $p(\mathbf{x}; \boldsymbol{\theta})$ . While we may have the CRB or other bound that is applicable to a broad class of estimators, we would still like to carry out performance analysis for specific estimators, that may be the MLE or not.

In this section we consider asymptotic analysis for two fundamentally important and widely applicable cases, Fourier based algorithms, and weighted least squares (LS) estimators. In both cases we can appeal to the CLT to show asymptotic normality, and then the algorithm performance can be characterized by the mean and covariance.

#### 3.08.9.1 Fourier transform

The Fourier transform is a fundamental linear decomposition that can be exploited to facilitate performance analysis, because the transformation yields Fourier coefficients that tend to a complex-valued Gaussian distribution. The Fourier transform approach to analysis is particularly useful for wideband problems, such as wideband array processing [60], and bit error rate (BER) analysis of OFDM systems in a fading environment [61]. This represents an extremely small fraction of the literature using these ideas, and the basic concept can be readily extended to other signal decompositions.

We illustrate the idea in the following. Suppose we have observations from a random process  $y(t; \boldsymbol{\theta})$ , that depend on the deterministic parameters  $\boldsymbol{\theta}$ . We observe  $y(t; \boldsymbol{\theta})$  over  $M$  intervals each of length  $T$  seconds. We may treat both the discrete-time and the continuous-time cases similarly. In discrete-time, samples  $y(t; \boldsymbol{\theta})$  are placed in a vector and the DFT is applied. In continuous-time, we apply the short-time Fourier transform to obtain

$$x_m(\omega_k; \boldsymbol{\theta}) = \frac{1}{T} \int_{-T/2}^{T/2} y(t; \boldsymbol{\theta}) e^{-j\omega_k t} dt, \quad (8.35)$$

over  $m = 1, \dots, M$  intervals, for  $k = 1, \dots, L$  distinct frequencies, and it is understood the integral is over successive time intervals of length  $T$ . We will drop the explicit mention of  $\boldsymbol{\theta}$ , keeping in mind the Fourier coefficients are functions of the parameters in the underlying pdf model that generates  $y(t; \boldsymbol{\theta})$ . Asymptotically in  $T$ , the Fourier coefficients  $x_m(\omega_k)$  become normally distributed via the central limit theorem. In addition, in many cases, the  $x_m(\omega_k)$  become uncorrelated at different frequencies. This generally occurs when  $T$  is large compared to the time lag  $\tau_0$  such that  $y(t)$  and  $y(t + \tau_0)$  are weakly or completely uncorrelated, i.e.,  $E[y(t)y(t + \tau_0)] \approx 0$ . For example, for a bandlimited random process with bandwidth  $B$  Hz, then the decorrelation time is roughly  $\tau_0 \approx 1/B$  seconds.

Under the above assumptions, we place the  $LM$  complex-valued Fourier coefficients into  $L$  vectors

$$\mathbf{x}^k = [x_1(\omega_k), \dots, x_M(\omega_k)]^T, \quad k = 1, \dots, L, \quad (8.36)$$

and let  $\boldsymbol{\mu}^k$  and  $C_k$  be the mean and covariance matrix of  $\mathbf{x}^k$ , respectively. Note that  $C_k$  describes the correlation at the frequency  $\omega_k$  across time intervals; we are assuming that the Fourier coefficients are uncorrelated across frequencies. Asymptotically the vectors  $\mathbf{x}^k$  are independent with a normal distribution, so the log-likelihood function can be written as

$$\mathcal{L}(\mathbf{x}; \boldsymbol{\theta}) = -\sum_{k=1}^L (\mathbf{x}^k - \boldsymbol{\mu}^k)^H C_k^{-1} (\mathbf{x}^k - \boldsymbol{\mu}^k) + \log \det(\pi C_k), \quad (8.37)$$

where  $H$  denotes conjugate-transpose (the Hermitian operator), and  $\det(\cdot)$  is the determinant. Recalling that  $\boldsymbol{\mu}^k$  and  $C_k$  are functions of the parameters  $\boldsymbol{\theta}$ , we can now use  $\mathcal{L}(\mathbf{x}; \boldsymbol{\theta})$  to find the MLE and CRB for estimation of  $\boldsymbol{\theta}$ . Note, however, that the MLE and CRB we are referring to are asymptotic approximations and were not derived directly from the original model.

If the original random process  $y(t; \boldsymbol{\theta})$  is Gaussian then the Fourier coefficients are also normally distributed because the Fourier transform is a linear operation. In this case, the only assumption leading to (8.37) is  $T \gg \tau_0$ , to ensure that the coefficients are uncorrelated at distinct frequencies. This does not require application of the CLT. More generally, if  $y(t; \boldsymbol{\theta})$  is non-Gaussian, then we rely on the CLT to ensure the Fourier coefficients are approaching a normal distribution. This depends on the observation interval length  $T$  (or equivalently the number of Nyquist samples when applying the DFT). For example, it is not difficult to accept that a DFT whose length is on the order of  $10^3$  or greater will produce a strong CLT effect.

As we noted in Section 3.08.8, it is important to validate the Gaussian assumption, such as using tests for Gaussianity as a function of the Fourier window time  $T$ . In addition, the assumption that the Fourier coefficients are uncorrelated should also be verified analytically or, e.g., using sample cross correlations. Note also that the Gaussian model can be readily modified to include frequency cross-correlation.

### 3.08.9.2 Least squares estimation

As a second example of the application of the CLT in parametric models, we consider weighted least squares nonlinear estimation, an extremely general tool. We will specifically assume the parameterized signal plus noise model in Eq. (8.1), where the noise is a stationary process with finite variance (not necessarily Gaussian). The least squares approach seeks to minimize the sum of squared errors criterion

$$J_N(\boldsymbol{\theta}) = \sum_{n=1}^N (x(n) - s(n; \boldsymbol{\theta}))^2, \quad (8.38)$$

so that

$$\hat{\boldsymbol{\theta}}_{\text{LS}} = \arg \min_{\boldsymbol{\theta}} J_N(\boldsymbol{\theta}). \quad (8.39)$$

Unless  $s(n; \boldsymbol{\theta})$  is linear in  $\boldsymbol{\theta}$ , then solving (8.39) is a nonlinear optimization problem. If the additive noise is Gaussian then (8.39) corresponds to maximum likelihood estimation, and we can appeal to

classic CRB results, e.g., see [4]. If the noise pdf is unknown, then we cannot readily compute a CRB for estimation of  $\theta$ .

The following is based on Wu [62]. Consider the parameterized signal plus additive noise model in (8.1). We assume that the additive noise is stationary with finite variance, but do not necessarily assume the noise pdf is known, nor do we assume the signal model is linear in the parameters  $\theta$ . Putting the noisy observation into (8.38), the LS criterion seeks to fit the signal model to the noisy observations using the squared-error penalty for deviation from the model. This approach is very general and reasonable. Solution of (8.38) will require optimization, and with a nonlinear signal model there will generally be local optima, i.e., the cost function will not generally be convex. Thus the optimization algorithm may require search or a good initial estimate. For many examples of this scenario see Kay [4, Chapter 8]. Generally LS (or weighted LS) does not have any particular guaranteed finite sample optimality, but we can appeal to asymptotics.

Wu has shown the following asymptotic (as  $N$  grows) properties of LS for the case of a parametric signal in additive noise given by (8.1), resulting in (8.38). The LS estimator is consistent, i.e., it converges in probability to the true value of  $\theta$ , meaning that the LS estimate has a probability distribution that is collapsing about the true value of  $\theta$ . This assumes continuity in the pdf, and requires the LS error to grow with  $N$  except when the LS criterion is evaluated at the correct parameter. Intuitively, this last requirement implies that the signal does not die out. For example, suppose the signal undergoes an exponential decay. Then, the asymptotic properties are not guaranteed, but this is not surprising since the signal is dying off making additional observations of decreasing value. The signal must have sufficient energy and duration relative to the noise power for the asymptotic results to be meaningful. (Alternatively, we might have multiple observations of a finite length signal.)

The asymptotic results still hold when the noise is non-identically distributed (independent, but non-identical), so long as the variance is bounded. The same ideas hold even if the parameter  $\theta$  is taken from a finite set, implying we are solving a classification problem rather than a continuous parameter estimation problem; see Section 3.08.2.1.5. Also, the average of the errors squared (the average sum of residuals squared) converges to the noise variance, so it is straightforward to asymptotically estimate the noise power.

Finally, note that the LS estimate is asymptotically normally distributed [62, Thrm. 5]. The variance of the normal distribution provides a large sample confidence level on the estimate of  $\theta$ , which provides a probability statement about our confidence in a particular realization of the estimate, and this can be computed based on the observations. It is important to recognize that a confidence level provides statistical evidence, and should not be confused with a bound such as the CRB; see Section 3.08.11.

As with the Fourier case discussed previously, the validity of the asymptotic assumption should be carefully checked for a particular problem. For example, if the additive noise distribution strongly deviates from normal, or if there is significant dependence in the noise, then the convergence rate to the asymptotic regime may be slow.

### 3.08.10 Monte Carlo methods

A Monte Carlo method (MCM) is a computational algorithm that utilizes random sampling in some way during the computation, such as computing an expected value, where the algorithm uses realizations of

some random process. This is a broad area of applications, and we consider some basic ideas as applied to performance analysis.

### 3.08.10.1 Using Monte Carlo to approximate an expectation

Consider numerically computing (or approximating) an expected value  $E_{\theta}(\cdot) = \int_{\mathbb{X}} (\cdot)p(x; \theta)dx$ . As a simple example, suppose we wish to compute  $E[x]$ , where  $x$  is a scalar random variable. We can generate many realizations of  $x$  and compute the sample average, which will converge to the expectation in most cases of interest based on the law of large numbers. Similar thinking applies to  $E[f(x)]$ , for some function  $f(x)$ ; generate realizations of  $x$ , and then compute the sample average of  $f(x)$ . Because the expectation involves random variables, we can exploit random sampling in a straightforward averaging procedure. A particular motivation for MCM is when we are unable to carry out the expectation analytically. For example, the function  $f(x)$  may be highly nonlinear such that no closed form is known for the expectation integral.

Thus we see that applying MCM is a fundamentally different sample-based approach to approximating the integral, as opposed to classic numerical methods such as using the trapezoidal rule. The integral may have infinite limits, and be of high dimension, cases that often confound classic numerical methods. Consequently, it may be more efficient and more accurate to employ MCM.

To reduce complexity, and to avoid excessive sampling in regions of small probability, additional sophistication can be employed in the sampling strategy; an example is importance sampling.

### 3.08.10.2 Computing the CRB via Monte Carlo

The MCM can be applied to numerically computing the CRB at a specified value of the parameter  $\theta$ . Realizations of  $x$  are generated according to  $p(x; \theta)$ . These are then plugged into the Fisher score in (8.4), followed by averaging to estimate the FIM given by (8.3). Note that the averaging is over  $x$  for a fixed value of  $\theta$ .

An alternative is to use the realizations of  $x$  in (8.7). Note that in (8.7) the expectation is over the Hessian (the matrix of second derivatives). Thus, in this case, each realization of  $x$  results in a realization of the Hessian, and the Hessian matrices are averaged to approximate the FIM in (8.7).

Using either (8.3) or (8.7) assumes that the score or Hessian are known in closed form. If these derivatives are not known, then a Monte Carlo method may first be applied to estimate the derivatives for a given  $x$  and  $\theta$ . Then, these are averaged over the realizations  $x$ . This procedure is described in detail by Spall [16,63, chapter 12].

---

### 3.08.11 Confidence intervals

As described in Sections 3.08.8 and 3.08.9, in many cases we can exploit asymptotics to obtain an approximate (typically Gaussian) distribution for an estimate. For example, under the normal assumption, then an interval can be defined around the mean based on the variance  $\sigma$  of say,  $\pm\sigma$ . While the mean of the distribution is our estimate, the variance provides a measure of confidence or reliability in the estimate, because it provides a statistical picture of the range over which the estimate might reasonably deviate given a new set of observations.

Note that the distribution parameters, and hence the confidence interval, are derived from the observations. So, the estimate and the confidence interval will change given a new set of observations. This should not be confused with a bound such as the CRB. The CRB is based on complete statistical knowledge of the underlying probability model  $p(\mathbf{x}; \boldsymbol{\theta})$ .

Confidence intervals have been extensively applied in regression analysis, especially for additive Gaussian noise, but also more generally in the asymptotic sense with additive noise modeled as samples from an arbitrary stationary process. For example, see the comprehensive treatment by Draper and Smith [64].

An interesting alternative to estimating confidence levels is to use the bootstrap method; see Efron [65]. This approach does not rely on asymptotics, instead using resampling of the data. The approach is relatively simple to implement using the given observations, and does not require extensive knowledge of the underlying probability model, although it can be computationally demanding due to extensive resampling. An overview of the bootstrap in statistical signal processing is given by Zoubir [66].

### 3.08.12 Conclusion

We have introduced many key concepts in performance analysis for statistical signal processing. But of course, given the decades of developments in statistics and signal processing, there are many topics left untouched. These include treatment of random parameters, for example see the edited paper collection, Van Trees and Bell [67]. These are often referred to as Bayesian bounds. Alternatives to the CRB include the Ziv-Zakai bound [68,69], and the Chapman-Robbins bound [70]. These may be more challenging to derive analytically, but typically result in a tighter bound in the non-asymptotic region.

*Relevant Theory:* Signal Processing Theory, Machine Learning, and Statistical Signal Processing

See [Vol. 1, Chapter 11](#) Parametric Estimation

See [Vol. 1, Chapter 19](#) A Tutorial Introduction to Monte Carlo Methods, Markov Chain Monte Carlo and Particle Filtering

See [Vol. 1, Chapter 25](#) Model Selection

See this Volume, [Chapter 2](#) Model Order Selection

See this Volume, [Chapter 4](#) Bayesian Computational Methods in Signal Processing

## References

- [1] Jun Shao, Mathematical Statistics, Springer-Verlag, New York, NY, 2003.
- [2] Harry L. Van Trees, Detection, Estimation, and Modulation Theory, Part I, Wiley, 1968.
- [3] George Casella, Roger L. Berger, Statistical Inference, second ed., Duxbury Press, Boca-Raton, FL, 2002.
- [4] S.M. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory, Prentice Hall, Englewood Cliffs, NJ, 1993.
- [5] H. Cramér, Mathematical Methods of Statistics, Princeton University Press, Princeton, NJ, 1946.
- [6] Calyampudi Radakrishna Rao, Information and the accuracy attainable in the estimation of statistical parameters, Bull. Calcutta Math. Soc. 37 (1945) 81–89.

- [7] A.O. Hero, J.A. Fessler, M. Usman, Exploring estimator bias-variance tradeoffs using the uniform CR bound, *IEEE Trans. Signal Process.* 44 (8) (1996) 2026–2042.
- [8] A.O. Hero, J.A. Fessler, M. Usman, Bias-resolution-variance tradeoffs for single pixel estimation tasks using the uniform Cramér-Rao bound, in: Proc. of IEEE Nuclear Science Symposium, vol. 2, pp. 296–298.
- [9] W.J. Bangs, Array processing with generalized beamformers. PhD Thesis, Yale University, New Haven, CT, 1971.
- [10] D. Slepian, Estimation of signal parameters in the presence of noise, *Trans. IRE Prof. Group Inform. Theory PG IT-3* (1954) 68–69.
- [11] Thomas J. Rothenberg, Identification in parametric models, *Econometrica* 39 (3) (1971) 577–591 (May).
- [12] B. Hochwald, A. Nehorai, On identifiability and information-regularity in parameterized normal distributions, *Circ. Syst. Signal Process.* 16 (1) (1997) 83–89.
- [13] Y. Chen, A. Wiesel, Y.C. Eldar, A.O. Hero, Shrinkage algorithms for MMSE covariance estimation, *IEEE Trans. Signal Process.* 58 (10) (2010) 5016–5029.
- [14] J.G. Proakis, M. Salehi, *Digital Communications*, fifth ed., McGraw-Hill, 2007.
- [15] M.K. Simon, M.-S. Alouini, *Digital Communications Over Fading Channels*, second ed., Wiley, 2005.
- [16] J.C. Spall, *Introduction to Stochastic Search and Optimization*, Wiley, 2003.
- [17] Y. Chen, A. Wiesel, A.O. Hero, Robust shrinkage estimation of high dimensional covariance matrices, *IEEE Trans. Signal Process.* 59 (9) (2011) 4097–4107.
- [18] N. Liu, Z. Xu, B.M. Sadler, Geolocation performance with biased range measurements, *IEEE Trans. Signal Process.* 60 (5) (2012) 2315–2329.
- [19] Z. Xu, Perturbation analysis for subspace decomposition with applications in subspace-based algorithms, *IEEE Trans. Signal Process.* 50 (11) (2002) 2820–2830.
- [20] G.W. Stewart, Error and perturbation bounds for subspaces associated with certain eigenvalue problems, *SIAM Rev.* 15 (4) (1973) 727–764.
- [21] G.W. Stewart, On the perturbation of psuedo-inverses, projections and linear least squares problems, *SIAM Rev.* 19 (4) (1977) 634–662.
- [22] G.W. Stewart, J. Sun, *Matrix Perturbation Theory*, Academic Press, 1990.
- [23] Harry L. Van Trees, *Optimum Array Processing*, Wiley, 2002.
- [24] F. Li, H. Liu, R.J. Vaccaro, Performance analysis for DOA estimation algorithms: unification, simplification, and observations, *IEEE Trans. Aerosp. Electron. Syst.* 29 (4) (1993) 1170–1184.
- [25] A.J. Weiss, J. Picard, Maximum-likelihood position estimation of network nodes using range measurements, *IET Signal Process.* 2 (4) (2008) 394–404.
- [26] Petre Stoica, Boon Chong Ng, On the Cramér-Rao bound under parametric constraints, *IEEE Signal Process. Lett.* 5 (7) (1998) 177–179.
- [27] Terrence J. Moore, Brian M. Sadler, Sufficient conditions for regularity and strict identifiability in MIMO systems, *IEEE Trans. Signal Process.* 52 (9) (2004) 2650–2655.
- [28] Zvika Ben-Haim, Yonina Eldar, On the constrained Cramér-Rao bound with a singular Fisher information matrix, *IEEE Signal Process. Lett.* 16 (6) (2009) 453–456.
- [29] Terrence J. Moore, Richard J. Kozick, Brian M. Sadler, The constrained Cramér-Rao bound from the perspective of fitting a model, *IEEE Signal Process. Lett.* 14 (8) (2007) 564–567.
- [30] Terrence J. Moore, Brian M. Sadler, Richard J. Kozick, Maximum-likelihood estimation, the Cramér-Rao bound, and the method of scoring with parameter constraints, *IEEE Trans. Signal Process.* 56 (3) (2008) 895–908.
- [31] P.K. Rajasekaran, N. Satyanarayana, M.D. Srinath, Optimum linear estimation of stochastic signals in the presence of multiplicative noise, *IEEE Trans. Aerosp. Electron. Syst.* 7 (3) (1971) 462–468.
- [32] R.J. Kozick, B.M. Sadler, D.K. Wilson, *Signal Processing and Propagation for Aeroacoustic Networks in Distributed Sensor Networks*, CRC Press, 2004.

- [33] Petre Stoica, Olivier Besson , Alex. B. Gershman, Direction-of-arrival estimation of an amplitude-distorted wavefront, *IEEE Trans. Signal Process.* 49 (2) (2001) 269–276.
- [34] M. Ghogho, A. Swami, T.S. Durrani, Frequency estimation in the presence of Doppler spread: performance analysis, *IEEE Trans. Signal Process.* 49 (4) (2001) 777–789.
- [35] A. Swami, Cramer-Rao bounds for deterministic signals in additive and multiplicative noise, *Signal Process.* 53 (1996) 231–244.
- [36] G.B. Giannakis, G. Zhou, Harmonics in Gaussian multiplicative and additive noise: Cramer-Rao bounds, *IEEE Trans. Signal Process.* 43 (5) (1995) 1217–1231.
- [37] M. Ghogho, A. Swami, A.K. Nandi, Non-linear least squares estimation for harmonics in multiplicative and additive noise, *Signal Process.* 78 (1999) 43–60.
- [38] J.Y. Tourneret, A. Ferrari, A. Swami, Cramer-Rao lower bounds for change points in additive and multiplicative noise, *Signal Process.* 84 (2004) 1071–1088.
- [39] P. Ciblat, M. Ghogho, P. Forster, P. Larzabal, Harmonic retrieval in the presence of non-circular Gaussian multiplicative noise: performance bounds, *Signal Process.* 85 (2005) 737–749.
- [40] D. Middleton, A.D. Spaulding, Elements of weak signal detection in non-gaussian noise environments, in: *Advances in Statistical Signal Processing*, vol. 2, JAI Press, 1993.
- [41] P.J. Huber, E.M. Ronchetti, *Robust Statistics*, second ed., Wiley, 2009.
- [42] S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, 1993.
- [43] D. Sengupta, S. Kay, Efficient estimation of parameters for a non-Gaussian autoregressive process, *IEEE Trans. Acoust. Speech Signal Process.* 37 (6) (1989) 785–794.
- [44] B.M. Sadler, G.B. Giannakis, K.-S. Lii, Estimation and detection in non-Gaussian noise using higher order statistics, *IEEE Trans. Signal Process.* 42 (10) (1994) 2729–2741.
- [45] B.M. Sadler, Detection in correlated impulsive noise using fourth-order cumulants, *IEEE Trans. Signal Process.* 44 (11) (1996) 2793–2800.
- [46] H.V. Poor, J.B. Thomas, Signal detection in dependent non-gaussian noise, in: *Advances in Statistical Signal Processing*, vol. 2, JAI Press, 1993.
- [47] R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification*, second ed., Wiley, 2001.
- [48] R.J. Kozick, R.S. Blum, B.M. Sadler, Signal processing in non-Gaussian noise using mixture distributions and the EM algorithm, in: Proc. 31st Asilomar Conference on Signals, Systems, and Computers, vol. 1, 1997, pp. 438–442.
- [49] R.S. Blum, R.J. Kozick, B.M. Sadler, An adaptive spatial diversity receiver for non-gaussian interference and noise, *IEEE Trans. Signal Process.* 47 (8) (1999) 2100–2111.
- [50] R.J. Kozick, B.M. Sadler, Maximum-likelihood array processing in non-Gaussian noise with Gaussian mixtures, *IEEE Trans. Signal Process.* 48 (12) (2000) 3520–3535.
- [51] G. Samorodnitsky, M.S. Taqqu, *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*, Chapman and Hall, 1994.
- [52] S.A. Kassam, J.B. Thomas, *Signal Detection in Non-Gaussian Noise*, Springer, 1987.
- [53] R.A. Davis, S.I. Resnick, Limit theory for the sample covariances and correlation functions of moving averages, *Ann. Stat.* 14 (1986) 533–558.
- [54] A. Swami, B.M. Sadler, On some detection and estimation problems in heavy-tailed noise, *Signal Process.* 82 (2002) 1829–1846.
- [55] A. Swami, B.M. Sadler, Parameter estimation for linear alpha-stable processes, *IEEE Signal Process. Lett.* 5 (2) (1998) 48–50.
- [56] Robert J. Serfling, *Approximation Theorems of Mathematical Statistics*, Wiley, 1980.
- [57] W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. II, Wiley, 1971.
- [58] G. Marsaglia, Ratios of normal variables and ratios of sums of uniform variables, *J. Am. Statist. Assoc.* 60 (1965) 193–204.

- [59] D.R. Brillinger, M. Rosenblatt, Spectral analysis of time series, in: B. Harris (Ed.), Asymptotic Theory of Estimates of k-th Order Spectra, Wiley, 2002, pp. 153–188 (Chapter 7).
- [60] P.M. Schultheiss, H. Messer, Optimal and sub-optimal broad-band source location estimation, IEEE Trans. Signal Process. 41 (9) (1993) 2752–2763.
- [61] W.P. Siriwongpairat, K.J.R. Liu, Ultra-Wideband Communications Systems, IEEE Press, Wiley, 2008.
- [62] Chien-Fu Wu, Asymptotic theory of nonlinear least squares estimation, Ann. Stat. 9 (3) (1981) 501–513.
- [63] J.C. Spall, On Monte Carlo methods for estimating the Fisher information matrix in difficult problems, in: 43rd Annual Conference on Information Sciences and Systems, 2009, pp. 741–746.
- [64] N.R. Draper, H. Smith, Applied Regression Analysis, third ed., Wiley, 1998.
- [65] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman and Hall, 1993.
- [66] A.M. Zoubir, D.R. Iskander, Bootstrap methods and applications, IEEE Signal Process. Mag. 24 (4) (2007) 10–19.
- [67] H.L. Van Trees, K.L. Bell, Bayesian Bounds for Parameter Estimation and Nonlinear/Filtering Tracking, IEEE Press, Wiley, 2007.
- [68] J. Ziv, M. Zakai, Some lower bounds on signal parameter estimation, IEEE Trans. Inform. Theory (1969) 386–391.
- [69] D. Chazan, M. Zakai, J. Ziv, Improved lower bounds on signal parameter estimation, IEEE Trans. Inform. Theory 21 (1) (1975) 90–93.
- [70] D.G. Chapman, H. Robbins, Minimum variance estimation without regularity assumptions, Ann. Math. Stat. 22 (6) (1951) 581–586.

# Diffusion Adaptation Over Networks\*

# 9

**Ali H. Sayed**

*Electrical Engineering Department, University of California at Los Angeles, USA*

Adaptive networks are well-suited to perform decentralized information processing and optimization tasks and to model various types of self-organized and complex behavior encountered in nature. Adaptive networks consist of a collection of agents with processing and learning abilities. The agents are linked together through a connection topology, and they cooperate with each other through local interactions to solve distributed optimization, estimation, and inference problems in real-time. The continuous diffusion of information across the network enables agents to adapt their performance in relation to streaming data and network conditions; it also results in improved adaptation and learning performance relative to non-cooperating agents. This chapter provides an overview of diffusion strategies for adaptation and learning over networks. The chapter is divided into several sections and includes appendices with supporting material intended to make the presentation rather self-contained for the benefit of the reader.

---

### 3.09.1 Motivation

Consider a collection of  $N$  agents interested in estimating the same parameter vector,  $w^o$ , of size  $M \times 1$ . The vector is the minimizer of some global cost function, denoted by  $J^{\text{glob}}(w)$ , which the agents seek to optimize, say,

$$w^o = \underset{w}{\operatorname{argmin}} J^{\text{glob}}(w) . \quad (9.1)$$

We are interested in situations where the individual agents have access to partial information about the global cost function. In this case, cooperation among the agents becomes beneficial. For example, by cooperating with their neighbors, and by having these neighbors cooperate with their neighbors, procedures can be devised that would enable all agents in the network to converge towards the global optimum  $w^o$  through local interactions. The objective of decentralized processing is to allow spatially distributed agents to achieve a global objective by relying solely on local information and on in-network processing. Through a continuous process of cooperation and information sharing with neighbors, agents in a network can be made to approach the global performance level despite the localized nature of their interactions.

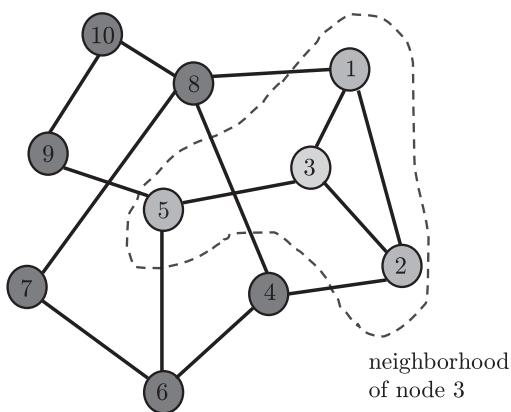
---

\*The work was supported in part by NSF grants EECS-060126, EECS-0725441, CCF-0942936, and CCF-1011918.

### 3.09.1.1 Networks and neighborhoods

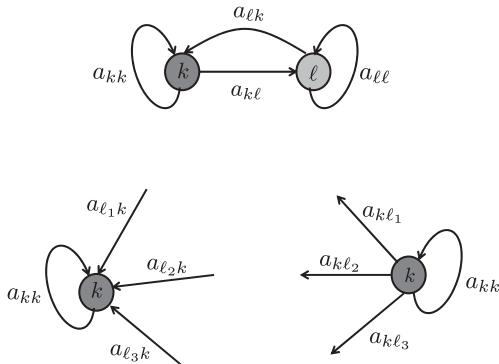
In this chapter we focus mainly on *connected* networks, although many of the results hold even if the network graph is separated into disjoint subgraphs. In a connected network, if we pick any two arbitrary nodes, then there will exist at least one path connecting them: the nodes may be connected directly by an edge if they are neighbors, or they may be connected by a path that passes through other intermediate nodes. Figure 9.1 provides a graphical representation of a connected network with  $N = 10$  nodes. Nodes that are able to share information with each other are connected by edges. The sharing of information over these edges can be unidirectional or bi-directional. The neighborhood of any particular node is defined as the set of nodes that are connected to it by edges; we include in this set the node itself. The figure illustrates the neighborhood of node 3, which consists of the following subset of nodes:  $\mathcal{N}_3 = \{1, 2, 3, 5\}$ . For each node, the size of its neighborhood defines its degree. For example, node 3 in the figure has degree  $|\mathcal{N}_3| = 4$ , while node 8 has degree  $|\mathcal{N}_8| = 5$ . Nodes that are well connected have higher degrees. Note that we are denoting the neighborhood of an arbitrary node  $k$  by  $\mathcal{N}_k$  and its size by  $|\mathcal{N}_k|$ . We shall also use the notation  $n_k$  to refer to  $|\mathcal{N}_k|$ .

The neighborhood of any node  $k$  therefore consists of all nodes with which node  $k$  can exchange information. We assume a symmetric situation in relation to neighbors so that if node  $k$  is a neighbor of node  $\ell$ , then node  $\ell$  is also a neighbor of node  $k$ . This does not necessarily mean that the flow of information between these two nodes is symmetrical. For instance, in future sections, we shall assign pairs of nonnegative weights to each edge connecting two neighboring nodes—see Figure 9.2. In particular, we will assign the coefficient  $a_{\ell k}$  to denote the weight used by node  $k$  to scale the data it receives from node  $\ell$ ; this scaling can be interpreted as a measure of trustworthiness or reliability that node  $k$  assigns to its interaction with node  $\ell$ . Note that we are using two subscripts,  $\ell k$ , with the



## **FIGURE 9.1**

A network consists of a collection of cooperating nodes. Nodes that are linked by edges can share information. The neighborhood of any particular node consists of all nodes that are connected to it by edges (including the node itself). The figure illustrates the neighborhood of node 3, which consists of nodes {1,2,3,5}. Accordingly, node 3 has degree 4, which is the size of its neighborhood.

**FIGURE 9.2**

In the top part, and for emphasis purposes, we are representing the edge between nodes  $k$  and  $\ell$  by two separate directed links: one moving from  $k$  to  $\ell$  and the other moving from  $\ell$  to  $k$ . Two nonnegative weights are used to scale the sharing of information over these directed links. The scalar  $a_{k\ell}$  denotes the weight used to scale data sent from node  $k$  to  $\ell$ , while  $a_{\ell k}$  denotes the weight used to scale data sent from node  $\ell$  to  $k$ . The weights  $\{a_{k\ell}, a_{\ell k}\}$  can be different, and one or both of them can be zero, so that the exchange of information over the edge connecting any two neighboring nodes need not be symmetric. The bottom part of the figure illustrates directed links arriving to node  $k$  from its neighbors  $\{\ell_1, \ell_2, \ell_3, \dots\}$  (left) and leaving from node  $k$  towards these same neighbors (right).

first subscript denoting the source node (where information originates from) and the second subscript denoting the sink node (where information moves to) so that:

$$a_{\ell k} \equiv a_{\ell \rightarrow k} \text{ (information flowing from node } \ell \text{ to node } k\text{)} . \quad (9.2)$$

In this way, the alternative coefficient  $a_{k\ell}$  will denote the weight used to scale the data sent from node  $k$  to  $\ell$ :

$$a_{k\ell} \equiv a_{k \rightarrow \ell} \text{ (information flowing from node } k \text{ to node } \ell\text{)} . \quad (9.3)$$

The weights  $\{a_{k\ell}, a_{\ell k}\}$  can be different, and one or both of them can be zero, so that the exchange of information over the edge connecting the neighboring nodes  $(k, \ell)$  need not be symmetric. When one of the weights is zero, say,  $a_{k\ell} = 0$ , then this situation means that even though nodes  $(k, \ell)$  are neighbors, node  $\ell$  is either not receiving data from node  $k$  or the data emanating from node  $k$  is being annihilated before reaching node  $\ell$ . Likewise, when  $a_{kk} > 0$ , then node  $k$  scales its own data, whereas  $a_{kk} = 0$  corresponds to the situation when node  $k$  does not use its own data. Usually, in graphical representations like those in Figure 9.1, edges are drawn between neighboring nodes that can share information. And, it is understood that the actual sharing of information is controlled by the values of the scaling weights that are assigned to the edge; these values can turn off communication in one or both directions and they can also scale one direction more heavily than the reverse direction, and so forth.

### 3.09.1.2 Cooperation among agents

Now, depending on the application under consideration, the solution vector  $w^o$  from (9.1) may admit different interpretations. For example, the entries of  $w^o$  may represent the location coordinates of a nutrition source that the agents are trying to find, or the location of an accident involving a dangerous chemical leak. The nodes may also be interested in locating a predator and tracking its movements over time. In these localization applications, the vector  $w^o$  is usually two or three-dimensional. In other applications, the entries of  $w^o$  may represent the parameters of some model that the network wishes to learn, such as identifying the model parameters of a biological process or the occupied frequency bands in a shared communications medium. There are also situations where different agents in the network may be interested in estimating different entries of  $w^o$ , or even different parameter vectors  $w^o$  altogether, say,  $\{w_k^o\}$  for node  $k$ , albeit with some relation among the different vectors so that cooperation among the nodes can still be rewarding. In this chapter, however, we focus exclusively on the special (yet frequent and important) case where all agents are interested in estimating the *same* parameter vector  $w^o$ .

Since the agents have a common objective, it is natural to expect cooperation among them to be beneficial in general. One important question is therefore how to develop cooperation strategies that can lead to better performance than when each agent attempts to solve the optimization problem individually. Another important question is how to develop strategies that endow networks with the ability to adapt and learn in real-time in response to changes in the statistical properties of the data. This chapter provides an overview of results in the area of *diffusion adaptation* with illustrative examples. Diffusion strategies are powerful methods that enable adaptive learning and cooperation over networks. There have been other useful works in the literature on the use of alternative *consensus strategies* to develop distributed optimization solutions over networks. Nevertheless, we explain in Appendix E why diffusion strategies outperform consensus strategies in terms of their mean-square-error stability and performance. For this reason, we focus in the body of the chapter on presenting the theoretical foundations for diffusion strategies and their performance.

### 3.09.1.3 Notation

In our treatment, we need to distinguish between random variables and deterministic quantities. For this reason, we use **boldface** letters to represent random variables and normal font to represent deterministic (non-random) quantities. For example, the boldface letter  $\mathbf{d}$  denotes a random quantity, while the normal font letter  $d$  denotes an observation or realization for it. We also need to distinguish between matrices and vectors. For this purpose, we use CAPITAL letters to refer to matrices and small letters to refer to both vectors and scalars; Greek letters always refer to scalars. For example, we write  $R$  to denote a covariance matrix and  $w$  to denote a vector of parameters. We also write  $\sigma_v^2$  to refer to the variance of a random variable. To distinguish between a vector  $d$  (small letter) and a scalar  $d$  (also a small letter), we use parentheses to index scalar quantities and subscripts to index vector quantities. Thus, we write  $d(i)$  to refer to the value of a scalar quantity  $d$  at time  $i$ , and  $d_i$  to refer to the value of a vector quantity  $d$  at time  $i$ . Thus,  $d(i)$  denotes a scalar while  $d_i$  denotes a vector. All vectors in our presentation are column vectors, with the exception of the regression vector (denoted by the letter  $u$ ), which will be taken to be a row vector for convenience of presentation. The symbol  $T$  denotes transposition, and the symbol  $*$  denotes complex conjugation for scalars and complex-conjugate transposition for matrices. The notation  $\text{col}\{a, b\}$  denotes a column vector with entries  $a$  and  $b$  stacked on top of each other,

**Table 9.1** Summary of Notation Conventions Used in the Chapter

$d$	Boldface notation denotes random variables
$d$	Normal font denotes realizations of random variables
$A$	Capital letters denote matrices
$a$	Small letters denote vectors or scalars
$\alpha$	Greek letters denote scalars
$d(i)$	Small letters with parenthesis denote scalars
$d_i$	Small letters with subscripts denote vectors
$T$	Matrix transposition
$*$	Complex conjugation for scalars and complex-conjugate transposition for matrices
$\text{col}\{a, b\}$	Column vector with entries $a$ and $b$
$\text{diag}\{a, b\}$	Diagonal matrix with entries $a$ and $b$
$\text{vec}(A)$	Vectorizes matrix $A$ and stacks its columns on top of each other
$\ x\ $	Euclidean norm of its vector argument
$\ x\ _{\Sigma}^2$	Weighted square value $x^* \Sigma x$
$\ x\ _{b,\infty}$	Block maximum norm of a block vector—see Appendix D
$\ A\ _{b,\infty}$	Block maximum norm of a matrix—see Appendix D
$\ A\ $	2-induced norm of matrix $A$ (its largest singular value)
$\rho(A)$	Spectral radius of its matrix argument
$I_M$	Identity matrix of size $M \times M$ ; sometimes, we drop the subscript $M$

and the notation  $\text{diag}\{a, b\}$  denotes a diagonal matrix with entries  $a$  and  $b$ . Likewise, the notation  $\text{vec}(A)$  vectorizes its matrix argument and stacks the columns of  $A$  on top of each other. The notation  $\|x\|$  denotes the Euclidean norm of its vector argument, while  $\|x\|_{b,\infty}$  denotes the block maximum norm of a block vector (defined in Appendix D). Similarly, the notation  $\|x\|_{\Sigma}^2$  denotes the weighted square value,  $x^* \Sigma x$ . Moreover,  $\|A\|_{b,\infty}$  denotes the block maximum norm of a matrix (also defined in Appendix D), and  $\rho(A)$  denotes the spectral radius of the matrix (i.e., the largest absolute magnitude among its eigenvalues). Finally,  $I_M$  denotes the identity matrix of size  $M \times M$ ; sometimes, for simplicity of notation, we drop the subscript  $M$  from  $I_M$  when the size of the identity matrix is obvious from the context. Table 9.1 provides a summary of the symbols used in the chapter for ease of reference.

### 3.09.2 Mean-square-error estimation

Readers interested in the development of the distributed optimization strategies and their adaptive versions can move directly to Section 3.09.3. The purpose of the current section is to motivate the virtues of distributed in-network processing, and to provide illustrative examples in the context of mean-square-error estimation. Advanced readers may skip this section.

We start our development by associating with each agent  $k$  an individual cost (or utility) function,  $J_k(w)$ . Although the algorithms presented in this chapter apply to more general situations, we nevertheless assume in our presentation that the cost functions  $J_k(w)$  are strictly convex so that each one of them

has a unique minimizer. We further assume that, for all costs  $J_k(w)$ , the minimum occurs at the same value  $w^o$ . Obviously, the choice of  $J_k(w)$  is limitless and is largely dependent on the application. It is sufficient for our purposes to illustrate the main concepts underlying diffusion adaptation by focusing on the case of mean-square-error (MSE) or quadratic cost functions. In the sequel, we provide several examples to illustrate how such quadratic cost functions arise in applications and how cooperative processing over networks can be beneficial. At the same time, we note that most of the arguments in this chapter can be extended beyond MSE optimization to more general cost functions and to situations where the minimizers of the individual costs  $J_k(w)$  need not agree with each other—as already shown in [1–3]; see also Section 3.09.10.4 for a brief summary.

In non-cooperative solutions, each agent would operate individually on its own cost function  $J_k(w)$  and optimize it to determine  $w^o$ , without any interaction with the other nodes. However, the analysis and derivations in future sections will reveal that nodes can benefit from cooperation between them in terms of better performance (such as converging faster to  $w^o$  or tracking a changing  $w^o$  more effectively)—see, e.g., Theorems 9.6.3–9.6.5 and Section 3.09.7.3.

### 3.09.2.1 Application: autoregressive modeling

Our first example relates to identifying the parameters of an auto-regressive (AR) model from noisy data. Thus, consider a situation where agents are spread over some geographical region and each agent  $k$  is observing realizations  $\{d_k(i)\}$  of an AR zero-mean random process  $\{\mathbf{d}_k(i)\}$ , which satisfies a model of the form:

$$\mathbf{d}_k(i) = \sum_{m=1}^M \beta_m \mathbf{d}_k(i-m) + \mathbf{v}_k(i). \quad (9.4)$$

The scalars  $\{\beta_m\}$  represent the model parameters that the agents wish to identify, and  $\mathbf{v}_k(i)$  represents an additive zero-mean white noise process with power:

$$\sigma_{v,k}^2 \triangleq \mathbb{E}|\mathbf{v}_k(i)|^2. \quad (9.5)$$

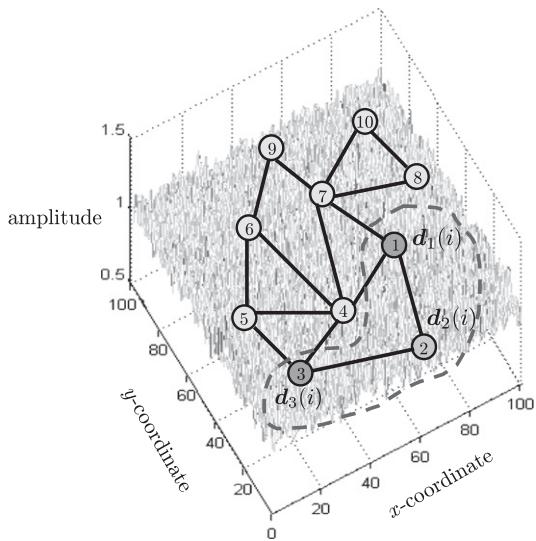
It is customary to assume that the noise process is temporally white *and* spatially independent so that noise terms across different nodes are independent of each other and, at the same node, successive noise samples are also independent of each other with a time-independent variance  $\sigma_{v,k}^2$ :

$$\begin{cases} \mathbb{E}\mathbf{v}_k(i)\mathbf{v}_k^*(j) = 0, & \text{for all } i \neq j \text{ (temporal whiteness),} \\ \mathbb{E}\mathbf{v}_k(i)\mathbf{v}_m^*(j) = 0, & \text{for all } i, j \text{ whenever } k \neq m \text{ (spatial whiteness).} \end{cases} \quad (9.6)$$

The noise process  $\mathbf{v}_k(i)$  is further assumed to be independent of past signals  $\{\mathbf{d}_\ell(i-m), m \geq 1\}$  across all nodes  $\ell$ . Observe that we are allowing the noise power profile,  $\sigma_{v,k}^2$ , to vary with  $k$ . In this way, the quality of the measurements is allowed to vary across the network with some nodes collecting noisier data than other nodes. Figure 9.3 illustrates an example of a network consisting of  $N = 10$  nodes spread over a region in space. The figure shows the neighborhood of node 2, which consists of nodes  $\{1, 2, 3\}$ .

#### 3.09.2.1.1 Linear model

To illustrate the difference between cooperative and non-cooperative estimation strategies, let us first explain that the data can be represented in terms of a linear model. To do so, we collect the model

**FIGURE 9.3**

A collection of nodes, spread over a geographic region, observes realizations of an AR random process and cooperates to estimate the underlying model parameters  $\{\beta_m\}$  in the presence of measurement noise. The noise power profile can vary over space.

parameters  $\{\beta_m\}$  into an  $M \times 1$  column vector  $w^o$ :

$$w^o \triangleq \text{col}\{\beta_1, \beta_2, \dots, \beta_M\} \quad (9.7)$$

and the past data into a  $1 \times M$  row regression vector  $\mathbf{u}_{k,i}$ :

$$\mathbf{u}_{k,i} \triangleq [\mathbf{d}_k(i-1) \ \mathbf{d}_k(i-2) \ \cdots \ \mathbf{d}_k(i-M)]. \quad (9.8)$$

Then, we can rewrite the measurement equation (9.4) at each node  $k$  in the equivalent *linear model* form:

$$\boxed{\mathbf{d}_k(i) = \mathbf{u}_{k,i} w^o + \mathbf{v}_k(i)}. \quad (9.9)$$

Linear relations of the form (9.9) are common in applications and arise in many other contexts (as further illustrated by the next three examples in this section).

We assume the stochastic processes  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  in (9.9) have zero means and are jointly wide-sense stationary. We denote their second-order moments by:

$$\sigma_{d,k}^2 \triangleq \mathbb{E}|\mathbf{d}_k(i)|^2 \quad (\text{scalar}), \quad (9.10)$$

$$R_{u,k} \triangleq \mathbb{E}\mathbf{u}_{k,i}^* \mathbf{u}_{k,i} \quad (M \times M), \quad (9.11)$$

$$r_{du,k} \triangleq \mathbb{E}\mathbf{d}_k(i)\mathbf{u}_{k,i}^* \quad (M \times 1). \quad (9.12)$$

As was the case with the noise power profile, we are allowing the moments  $\{\sigma_{d,k}^2, R_{u,k}, r_{du,k}\}$  to depend on the node index  $k$  so that these moments can vary with the spatial dimension as well. The covariance

matrix  $R_{u,k}$  is, by definition, always nonnegative definite. However, for convenience of presentation, we shall assume that  $R_{u,k}$  is actually positive-definite (and, hence, invertible):

$$R_{u,k} > 0. \quad (9.13)$$

### 3.09.2.1.2 Non-cooperative mean-square-error solution

One immediate result that follows from the linear model (9.9) is that the unknown parameter vector  $w^o$  can be recovered *exactly* by each individual node from knowledge of the local moments  $\{r_{du,k}, R_{u,k}\}$  alone. To see this, note that if we multiply both sides of (9.9) by  $\mathbf{u}_{k,i}^*$  and take expectations we obtain

$$\underbrace{\mathbb{E}\mathbf{u}_{k,i}^*\mathbf{d}_k(i)}_{r_{du,k}} = \underbrace{(\mathbb{E}\mathbf{u}_{k,i}^*\mathbf{u}_{k,i})w^o}_{R_{u,k}} + \underbrace{\mathbb{E}\mathbf{u}_{k,i}^*\mathbf{v}_k(i)}_{=0}, \quad (9.14)$$

so that

$$r_{du,k} = R_{u,k}w^o \iff w^o = R_{u,k}^{-1}r_{du,k}. \quad (9.15)$$

It is seen from (9.15) that  $w^o$  is the solution to a linear system of equations and that this solution can be computed by every node directly from its moments  $\{R_{u,k}, r_{du,k}\}$ . It is useful to re-interpret construction (9.15) as the solution to a minimum mean-square-error (MMSE) estimation problem [4,5]. Indeed, it can be verified that the quantity  $R_{u,k}^{-1}r_{du,k}$  that appears in (9.15) is the unique solution to the following MMSE problem:

$$\min_w \mathbb{E}|\mathbf{d}_k(i) - \mathbf{u}_{k,i}w|^2. \quad (9.16)$$

To verify this claim, we denote the cost function that appears in (9.16) by

$$J_k(w) \triangleq \mathbb{E}|\mathbf{d}_k(i) - \mathbf{u}_{k,i}w|^2 \quad (9.17)$$

and expand it to find that

$$J_k(w) = \sigma_{d,k}^2 - w^*r_{du,k} - r_{du,k}^*w + w^*R_{u,k}w. \quad (9.18)$$

The cost function  $J_k(w)$  is quadratic in  $w$  and it has a unique minimizer since  $R_{u,k} > 0$ . Differentiating  $J_k(w)$  with respect to  $w$  we find its gradient vector:

$$\nabla_w J(w) = (R_{u,k}w - r_{du,k})^*. \quad (9.19)$$

It is seen that this gradient vector is annihilated at the same value  $w = w^o$  given by (9.15). Therefore, we can equivalently state that if each node  $k$  solves the MMSE problem (9.16), then the solution vector coincides with the desired parameter vector  $w^o$ . This observation explains why it is often justified to consider mean-square-error cost functions when dealing with estimation problems that involve data that satisfy linear models similar to (9.9).

Besides  $w^o$ , the solution of the MMSE problem (9.16) also conveys information about the noise level in the data. Note that by substituting  $w^o$  from (9.15) into expression (9.16) for  $J_k(w)$ , the resulting

minimum mean-square-error value that is attained by node  $k$  is found to be:

$$\begin{aligned} \text{MSE}_k &\triangleq J_k(w^o) \\ &= \mathbb{E}|\mathbf{d}_k(i) - \mathbf{u}_{k,i}w^o|^2 \\ &\stackrel{(9.9)}{=} \mathbb{E}|\mathbf{v}_k(i)|^2 \\ &= \sigma_{v,k}^2 \\ &\triangleq J_{k,\min}. \end{aligned} \quad (9.20)$$

We shall use the notation  $J_k(w^o)$  and  $J_{k,\min}$  interchangeably to denote the minimum cost value of  $J_k(w)$ . The above result states that, when each agent  $k$  recovers  $w^o$  from knowledge of its moments  $\{R_{u,k}, r_{du,k}\}$  using expression (9.15), then the agent attains an MSE performance level that is equal to the noise power at its location,  $\sigma_{v,k}^2$ . An alternative useful expression for the minimum cost can be obtained by substituting expression (9.15) for  $w^o$  into (9.18) and simplifying the expression to find that

$$\text{MSE}_k = \sigma_{d,k}^2 - r_{du,k}^* R_{u,k}^{-1} r_{du,k}. \quad (9.21)$$

This second expression is in terms of the data moments  $\{\sigma_{d,k}^2, r_{du,k}, R_{u,k}\}$ .

### 3.09.2.1.3 Non-cooperative adaptive solution

The optimal MMSE implementation (9.15) for determining  $w^o$  requires knowledge of the statistical information  $\{r_{du,k}, R_{u,k}\}$ . This information is usually not available beforehand. Instead, the agents are more likely to have access to successive time-indexed observations  $\{d_k(i), u_{k,i}\}$  of the random processes  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  for  $i \geq 0$ . In this case, it becomes necessary to devise a scheme that would allow each node to use its measurements to approximate  $w^o$ . It turns out that an adaptive solution is possible. In this alternative implementation, each node  $k$  feeds its observations  $\{d_k(i), u_{k,i}\}$  into an adaptive filter and evaluates successive estimates for  $w^o$ . As time passes by, the estimates would get closer to  $w^o$ .

The adaptive solution operates as follows. Let  $w_{k,i}$  denote an estimate for  $w^o$  that is computed by node  $k$  at time  $i$  based on all the observations  $\{d_k(j), u_{k,j}, j \leq i\}$  it has collected up to that time instant. There are many adaptive algorithms that can be used to compute  $w_{k,i}$ ; some filters are more accurate than others (usually, at the cost of additional complexity) [4–7]. It is sufficient for our purposes to consider one simple (yet effective) filter structure, while noting that most of the discussion in this chapter can be extended to other structures. One of the simplest choices for an adaptive structure is the least-mean-squares (LMS) filter, where the data are processed by each node  $k$  as follows:

$$e_k(i) = d_k(i) - u_{k,i}w_{k,i-1}, \quad (9.22)$$

$$w_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^* e_k(i), \quad i \geq 0. \quad (9.23)$$

Starting from some initial condition, say,  $w_{k,-1} = 0$ , the filter iterates over  $i \geq 0$ . At each time instant,  $i$ , the filter uses the local data  $\{d_k(i), u_{k,i}\}$  at node  $k$  to compute a local estimation error,  $e_k(i)$ , via (9.22). The error is then used to update the existing estimate from  $w_{k,i-1}$  to  $w_{k,i}$  via (9.23). The factor  $\mu_k$  that appears in (9.23) is a constant positive step-size parameter; usually chosen to be sufficiently small to ensure mean-square stability and convergence, as discussed further ahead in the chapter. The step-size

parameter can be selected to vary with time as well; one popular choice is to replace  $\mu_k$  in (9.23) with the following construction:

$$\mu_k(i) \triangleq \frac{\mu_k}{\epsilon + \|u_{k,i}\|^2}, \quad (9.24)$$

where  $\epsilon > 0$  is a small positive parameter and  $\mu_k > 0$ . The resulting filter implementation is known as normalized LMS [5] since the step-size is normalized by the squared norm of the regression vector. Other choices include step-size sequences  $\{\mu(i) \geq 0\}$  that satisfy both conditions:

$$\sum_{i=0}^{\infty} \mu(i) = \infty, \quad \sum_{i=0}^{\infty} \mu^2(i) < \infty. \quad (9.25)$$

Such sequences converge slowly towards zero. One example is the choice  $\mu_k(i) = \frac{\mu_k}{i+1}$ . However, by virtue of the fact that such step-sizes die out as  $i \rightarrow \infty$ , then these choices end up turning off adaptation. As such, step-size sequences satisfying (9.25) are not generally suitable for applications that require continuous learning, especially under non-stationary environments. For this reason, in this chapter, we shall focus exclusively on the constant step-size case (9.23) in order to ensure continuous adaptation and learning.

Equations (9.22) and (9.23) are written in terms of the observed quantities  $\{d_k(i), u_{k,i}\}$ ; these are deterministic values since they correspond to observations of the random processes  $\{d_k(i), u_{k,i}\}$ . Often, when we are interested in highlighting the random nature of the quantities involved in the adaptation step, especially when we study the mean-square performance of adaptive filters, it becomes more useful to rewrite the recursions using the **boldface** notation to highlight the fact that the quantities that appear in (9.22) and (9.23) are actually realizations of random variables. Thus, we also write:

$$\mathbf{e}_k(i) = \mathbf{d}_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i-1}, \quad (9.26)$$

$$\mathbf{w}_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \mathbf{u}_{k,i}^* \mathbf{e}_k(i), \quad i \geq 0, \quad (9.27)$$

where  $\{\mathbf{e}_k(i), \mathbf{w}_{k,i}\}$  will be random variables as well.

The performance of adaptive implementations of this kind are well-understood for both cases of stationary  $w^o$  and changing  $w^o$  [4–7]. For example, in the stationary case, if the adaptive implementation (9.26) and (9.27) were to succeed in having its estimator  $\mathbf{w}_{k,i}$  tend to  $w^o$  with probability one as  $i \rightarrow \infty$ , then we would expect the error signal  $\mathbf{e}_k(i)$  in (9.26) to tend towards the noise signal  $\mathbf{v}_k(i)$  (by virtue of the linear model (9.9)). This means that, under this ideal scenario, the variance of the error signal  $\mathbf{e}_k(i)$  would be expected to tend towards the noise variance,  $\sigma_{v,k}^2$ , as  $i \rightarrow \infty$ . Recall from (9.20) that the noise variance is the least cost that the MSE solution can attain. Therefore, such limiting behavior by the adaptive filter would be desirable. However, it is well-known that there is always some loss in mean-square-error performance when adaptation is employed due to the effect of gradient noise, which is caused by the algorithm's reliance on observations (or realizations)  $\{d_k(i), u_{k,i}\}$  rather than on the actual moments  $\{r_{du,k}, R_{u,k}\}$ . In particular, it is known that for LMS filters, the variance of  $\mathbf{e}_k(i)$  in steady-state will be larger than  $\sigma_{v,k}^2$  by a small amount, and the size of the offset is proportional to the step-size parameter  $\mu_k$  (so that smaller step-sizes lead to better mean-square-error (MSE) performance albeit at the expense of slower convergence). It is easy to see why the variance of  $\mathbf{e}_k(i)$  will be larger

than  $\sigma_{v,k}^2$  from the definition of the error signal in (9.26). Introduce the weight-error vector

$$\tilde{\mathbf{w}}_{k,i} \triangleq \mathbf{w}^o - \mathbf{w}_{k,i} \quad (9.28)$$

and the so-called *a priori* error signal

$$\mathbf{e}_{a,k}(i) \triangleq \mathbf{u}_{k,i} \tilde{\mathbf{w}}_{k,i-1}. \quad (9.29)$$

This second error measures the difference between the uncorrupted term  $\mathbf{u}_{k,i} \mathbf{w}^o$  and its estimator prior to adaptation,  $\mathbf{u}_{k,i} \mathbf{w}_{k,i-1}$ . It then follows from the data model (9.9) and from the defining expression (9.26) for  $\mathbf{e}_k(i)$  that

$$\begin{aligned} \mathbf{e}_k(i) &= \mathbf{d}_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i-1} \\ &= \mathbf{u}_{k,i} \mathbf{w}^o + \mathbf{v}_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i-1} \\ &= \mathbf{e}_{a,k}(i) + \mathbf{v}_k(i). \end{aligned} \quad (9.30)$$

Since the noise component,  $\mathbf{v}_k(i)$ , is assumed to be zero-mean and independent of all other random variables, we conclude that

$$\mathbb{E}|\mathbf{e}_k(i)|^2 = \mathbb{E}|\mathbf{e}_{a,k}(i)|^2 + \sigma_{v,k}^2. \quad (9.31)$$

This relation holds for any time instant  $i$ ; it shows that the variance of the output error,  $\mathbf{e}_k(i)$ , is larger than  $\sigma_{v,k}^2$  by an amount that is equal to the variance of the *a priori* error,  $\mathbf{e}_{a,k}(i)$ . We define the filter mean-square-error (MSE) and excess-mean-square-error (EMSE) as the following steady-state measures:

$$\text{MSE}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}|\mathbf{e}_k(i)|^2, \quad (9.32)$$

$$\text{EMSE}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}|\mathbf{e}_{a,k}(i)|^2, \quad (9.33)$$

so that, for the adaptive implementation (compare with (9.20)):

$$\text{MSE}_k = \text{EMSE}_k + \sigma_{v,k}^2. \quad (9.34)$$

Therefore, the EMSE term quantifies the size of the offset in the MSE performance of the adaptive filter. We also define the filter mean-square-deviation (MSD) as the steady-state measure:

$$\text{MSD}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|^2, \quad (9.35)$$

which measures how far  $\mathbf{w}_{k,i}$  is from  $\mathbf{w}^o$  in the mean-square-error sense in steady-state. It is known that the MSD and EMSE of LMS filters of the form (9.26) and (9.27) can be approximated for sufficiently small-step sizes by the following expressions [4–7]:

$$\text{EMSE}_k \approx \mu_k \sigma_{v,k}^2 \text{Tr}(\mathbf{R}_{u,k})/2, \quad (9.36)$$

$$\text{MSD}_k \approx \mu_k \sigma_{v,k}^2 M/2. \quad (9.37)$$

It is seen that the smaller the step-size parameter is, the better the performance of the adaptive solution.

### 3.09.2.1.4 Cooperative adaptation through diffusion

Observe from (9.36) and (9.37) that even if all nodes employ the same step-size,  $\mu_k = \mu$ , and even if the regression data are spatially uniform so that  $R_{u,k} = R_u$  for all  $k$ , the mean-square-error performance across the nodes still varies in accordance with the variation of the noise power profile,  $\sigma_{v,k}^2$ , across the network. Nodes with larger noise power will perform worse than nodes with smaller noise power. However, since all nodes are observing data arising from the *same* underlying model  $w^o$ , it is natural to expect cooperation among the nodes to be beneficial. By cooperation we mean that neighboring nodes can share information (such as measurements or estimates) with each other as permitted by the network topology. Starting in the next section, we will motivate and describe algorithms that enable nodes to carry out adaptation and learning in a cooperative manner to enhance performance.

Specifically, we are going to see that one way to achieve cooperation is by developing adaptive algorithms that enable the nodes to optimize the following global cost function in a distributed manner:

$$\min_w \sum_{k=1}^N \mathbb{E}|\mathbf{d}_k(i) - \mathbf{u}_{k,i} w|^2, \quad (9.38)$$

where the global cost is the aggregate objective:

$$J^{\text{glob}}(w) \triangleq \sum_{k=1}^N \mathbb{E}|\mathbf{d}_k(i) - \mathbf{u}_{k,i} w|^2 = \sum_{k=1}^N J_k(w). \quad (9.39)$$

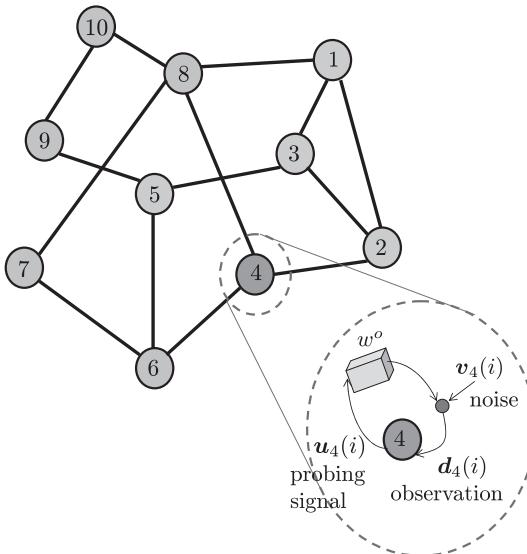
Comparing (9.38) with (9.16), we see that we are now adding the individual costs,  $J_k(w)$ , from across all nodes. Note that since the desired  $w^o$  satisfies (9.15) at every node  $k$ , then it also satisfies

$$\left( \sum_{k=1}^M R_{u,k} \right) w^o = \sum_{k=1}^N r_{du,k}. \quad (9.40)$$

But it can be verified that the optimal solution to (9.38) is given by the same  $w^o$  that satisfies (9.40). Therefore, solving the global optimization problem (9.38) also leads to the desired  $w^o$ . In future sections, we will show how cooperative and distributed adaptive schemes for solving (9.38), such as (9.153) or (9.154) further ahead, lead to improved performance in estimating  $w^o$  (in terms of smaller mean-square-deviation and faster convergence rate) than the non-cooperative mode (9.26) and (9.27), where each agent runs its own individual adaptive filter—see, e.g., Theorems 9.6.3–9.6.5 and Section 3.09.7.3.

### 3.09.2.2 Application: tapped-delay-line models

Our second example to motivate MSE cost functions,  $J_k(w)$ , and linear models relates to identifying the parameters of a moving-average (MA) model from noisy data. MA models are also known as finite-impulse-response (FIR) or tapped-delay-line models. Thus, consider a situation where agents are interested in estimating the parameters of an FIR model, such as the taps of a communications channel or the parameters of some (approximate) model of interest in finance or biology. Assume the agents are able to independently probe the unknown model and observe its response to excitations in the presence

**FIGURE 9.4**

The network is interested in estimating the parameter vector  $w^o$  that describes an underlying tapped-delay-line model. The agents are assumed to be able to independently probe the unknown system, and observe its response to excitations, under noise, as indicated in the highlighted diagram for node 4.

of additive noise; this situation is illustrated in Figure 9.4, with the probing operation highlighted for one of the nodes (node 4).

The schematics inside the enlarged diagram in Figure 9.4 is meant to convey that each node  $k$  probes the model with an input sequence  $\{\mathbf{u}_k(i)\}$  and measures the resulting response sequence,  $\{\mathbf{d}_k(i)\}$ , in the presence of additive noise. The system dynamics for each agent  $k$  is assumed to be described by a MA model of the form:

$$\mathbf{d}_k(i) = \sum_{m=0}^{M-1} \beta_m \mathbf{u}_k(i-m) + \mathbf{v}_k(i). \quad (9.41)$$

In this model, the term  $\mathbf{v}_k(i)$  again represents an additive zero-mean noise process that is assumed to be temporally white and spatially independent; it is also assumed to be independent of the input process,  $\{\mathbf{u}_\ell(j)\}$ , for all  $i, j$ , and  $\ell$ . The scalars  $\{\beta_m\}$  represent the model parameters that the agents seek to identify. If we again collect the model parameters into an  $M \times 1$  column vector  $w^o$ :

$$w^o \triangleq \text{col}\{\beta_0, \beta_1, \dots, \beta_{M-1}\} \quad (9.42)$$

and the input data into a  $1 \times M$  row regression vector:

$$\mathbf{u}_{k,i} \triangleq [\mathbf{u}_k(i) \ \mathbf{u}_k(i-1) \ \cdots \ \mathbf{u}_k(i-M+1)] \quad (9.43)$$

then, we can again express the measurement equation (9.41) at each node  $k$  in the same linear model as (9.9), namely,

$$\boxed{\mathbf{d}_k(i) = \mathbf{u}_{k,i} w^o + \mathbf{v}_k(i)} . \quad (9.44)$$

As was the case with model (9.9), we can likewise verify that, in view of (9.44), the desired parameter vector  $w^o$  satisfies the same normal equations (9.15), i.e.,

$$r_{du,k} = R_{u,k} w^o \iff w^o = R_{u,k}^{-1} r_{du,k}, \quad (9.45)$$

where the moments  $\{r_{du,k}, R_{u,k}\}$  continue to be defined by expressions (9.11) and (9.12) with  $\mathbf{u}_{k,i}$  now defined by (9.43). Therefore, each node  $k$  can determine  $w^o$  on its own by solving the same MMSE estimation problem (9.16). This solution method requires knowledge of the moments  $\{r_{du,k}, R_{u,k}\}$  and, according to (9.20), each agent  $k$  would then attain an MSE level that is equal to the noise power level at its location.

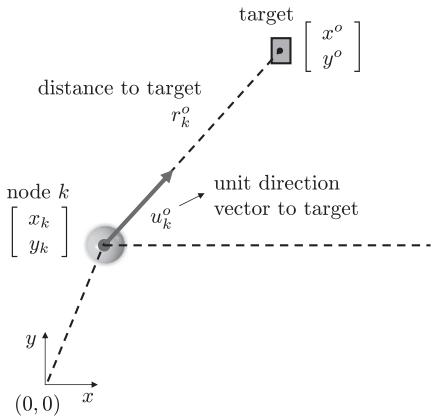
Alternatively, when the statistical information  $\{r_{du,k}, R_{u,k}\}$  is not available, each agent  $k$  can estimate  $w^o$  iteratively by feeding data  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  into the adaptive implementation (9.26) and (9.27). In this way, each agent  $k$  will achieve the same performance level shown earlier in (9.36) and (9.37), with the limiting performance being again dependent on the local noise power level,  $\sigma_{v,k}^2$ . Therefore, nodes with larger noise power will perform worse than nodes with smaller noise power. However, since all nodes are observing data arising from the same underlying model  $w^o$ , it is natural to expect cooperation among the nodes to be beneficial. As we are going to see, starting from the next section, one way to achieve cooperation and improve performance is by developing algorithms that optimize the same global cost function (9.38) in an adaptive and distributed manner, such as algorithms (9.153) and (9.154) further ahead.

### 3.09.2.3 Application: target localization

Our third example relates to the problem of locating a destination of interest (such as the location of a nutrition source or a chemical leak) or locating and tracking an object of interest (such as a predator or a projectile). In several such localization applications, the agents in the network are allowed to move towards the target or away from it, in which case we would end up with a mobile adaptive network [8]. Biological networks behave in this manner such as networks representing fish schools, bird formations, bee swarms, bacteria motility, and diffusing particles [8–12]. The agents may move towards the target (e.g., when it is a nutrition source) or away from the target (e.g., when it is a predator). In other applications, the agents may remain static and are simply interested in locating a target or tracking it (such as tracking a projectile).

To motivate mean-square-error estimation in the context of localization problems, we consider the situation corresponding to a static target and static nodes. Thus, assume that the unknown location of the target in the cartesian plane is represented by the  $2 \times 1$  vector  $w^o = \text{col}\{x^o, y^o\}$ . The agents are spread over the same region of space and are interested in locating the target. The location of every agent  $k$  is denoted by the  $2 \times 1$  vector  $p_k = \text{col}\{x_k, y_k\}$  in terms of its  $x$  and  $y$  coordinates—see Figure 9.5. We assume the agents are aware of their location vectors. The distance between agent  $k$  and the target is denoted by  $r_k^o$  and is equal to:

$$r_k^o = \|w^o - p_k\|. \quad (9.46)$$

**FIGURE 9.5**

The distance from node  $k$  to the target is denoted by  $r_k^o$  and the unit-norm direction vector from the same node to the target is denoted by  $u_k^o$ . Node  $k$  is assumed to have access to noisy measurements of  $\{r_k^o, u_k^o\}$ .

The  $1 \times 2$  unit-norm direction vector pointing from agent  $k$  towards the target is denoted by  $u_k^o$  and is given by:

$$u_k^o = \frac{(w^o - p_k)^T}{\|w^o - p_k\|}. \quad (9.47)$$

Observe from (9.46) and (9.47) that  $r_k^o$  can be expressed in the following useful inner-product form:

$$r_k^o = u_k^o(w^o - p_k). \quad (9.48)$$

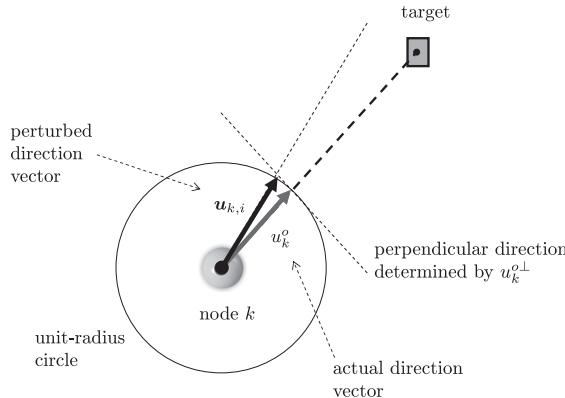
In practice, agents have noisy observations of both their distance and direction vector towards the target. We denote the noisy distance measurement collected by node  $k$  at time  $i$  by:

$$r_k(i) = r_k^o + v_k(i), \quad (9.49)$$

where  $v_k(i)$  denotes noise and is assumed to be zero-mean, and temporally white and spatially independent with variance

$$\sigma_{v,k}^2 \triangleq \mathbb{E}|v_k(i)|^2. \quad (9.50)$$

We also denote the noisy direction vector that is measured by node  $k$  at time  $i$  by  $u_{k,i}$ . This vector is a perturbed version of  $u_k^o$ . We assume that  $u_{k,i}$  continues to start from the location of the node at  $p_k$ , but that its tip is perturbed slightly either to the left or to the right relative to the tip of  $u_k^o$ —see Figure 9.6. The perturbation to the tip of  $u_k^o$  is modeled as being the result of two effects: a small deviation that occurs along the direction that is perpendicular to  $u_k^o$ , and a smaller deviation that occurs along the direction of  $u_k^o$ . Since we are assuming that the tip of  $u_{k,i}$  is only slightly perturbed relative to the tip of  $u_k^o$ , then it is reasonable to expect the amount of perturbation along the parallel direction to be small compared to the amount of perturbation along the perpendicular direction.

**FIGURE 9.6**

The tip of the noisy direction vector is modeled as being approximately perturbed away from the actual direction by two effects: a larger effect caused by a deviation along the direction that is perpendicular to  $u_k^o$ , and a smaller deviation along the direction that is parallel to  $u_k^o$ .

Thus, we write

$$\mathbf{u}_{k,i} = \mathbf{u}_k^o + \boldsymbol{\alpha}_k(i)u_k^{o\perp} + \boldsymbol{\beta}_k(i)u_k^o \quad (1 \times 2), \quad (9.51)$$

where  $u_k^{o\perp}$  denotes a unit-norm row vector that lies in the same plane and whose direction is perpendicular to  $u_k^o$ . The variables  $\boldsymbol{\alpha}_k(i)$  and  $\boldsymbol{\beta}_k(i)$  denote zero-mean independent random noises that are temporally white and spatially independent with variances:

$$\sigma_{\alpha,k}^2 \triangleq \mathbb{E}|\boldsymbol{\alpha}_k(i)|^2, \quad \sigma_{\beta,k}^2 \triangleq \mathbb{E}|\boldsymbol{\beta}_k(i)|^2. \quad (9.52)$$

We assume the contribution of  $\boldsymbol{\beta}_k(i)$  is small compared to the contributions of the other noise sources,  $\boldsymbol{\alpha}_k(i)$  and  $\mathbf{v}_k(i)$ , so that

$$\sigma_{\beta,k}^2 \ll \sigma_{\alpha,k}^2, \quad \sigma_{\beta,k}^2 \ll \sigma_{v,k}^2. \quad (9.53)$$

The random noises  $\{\mathbf{v}_k(i), \boldsymbol{\alpha}_k(i), \boldsymbol{\beta}_k(i)\}$  are further assumed to be independent of each other.

Using (9.48) we find that the noisy measurements  $\{\mathbf{r}_k(i), \mathbf{u}_{k,i}\}$  are related to the unknown  $w^o$  via:

$$\mathbf{r}_k(i) = \mathbf{u}_{k,i}(w^o - p_k) + \mathbf{z}_k(i), \quad (9.54)$$

where the modified noise term  $\mathbf{z}_k(i)$  is defined in terms of the noises in  $\mathbf{r}_k(i)$  and  $\mathbf{u}_{k,i}$  as follows:

$$\begin{aligned} \mathbf{z}_k(i) &\triangleq \mathbf{v}_k(i) - \boldsymbol{\alpha}_k(i)u_k^{o\perp}(w^o - p_k) - \boldsymbol{\beta}_k(i)u_k^o(w^o - p_k) \\ &= \mathbf{v}_k(i) - \boldsymbol{\beta}_k(i)r_k^o \\ &\approx \mathbf{v}_k(i), \end{aligned} \quad (9.55)$$

since, by construction,

$$u_k^{o\perp}(w^o - p_k) = 0 \quad (9.56)$$

and the contribution by  $\beta_k(i)$  is assumed to be sufficiently small. If we now introduce the adjusted signal:

$$\mathbf{d}_k(i) \triangleq \mathbf{r}_k(i) + \mathbf{u}_{k,i} p_k, \quad (9.57)$$

then we arrive again from (9.54) and (9.55) at the following linear model for the available measurement variables  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  in terms of the target location  $w^o$ :

$$\boxed{\mathbf{d}_k(i) \approx \mathbf{u}_{k,i} w^o + \mathbf{v}_k(i).} \quad (9.58)$$

There is one important difference in relation to the earlier linear models (9.9) and (9.44), namely, the variables  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  in (9.58) do not have zero means any longer. It is nevertheless straightforward to determine the first and second-order moments of the variables  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$ . First, note from (9.49), (9.51), and (9.57) that

$$\mathbb{E}\mathbf{u}_{k,i} = \mathbf{u}_k^o, \quad \mathbb{E}\mathbf{d}_k(i) = \mathbf{r}_k^o + \mathbf{u}_k^o p_k. \quad (9.59)$$

Even in this case of non-zero means, and in view of (9.58), the desired parameter vector  $w^o$  can still be shown to satisfy the same normal equations (9.15), i.e.,

$$r_{du,k} = R_{u,k} w^o \iff w^o = R_{u,k}^{-1} r_{du,k}, \quad (9.60)$$

where the moments  $\{r_{du,k}, R_{u,k}\}$  continue to be defined as

$$R_{u,k} \triangleq \mathbb{E}\mathbf{u}_{k,i}^* \mathbf{u}_{k,i}, \quad r_{du,k} \triangleq \mathbb{E}\mathbf{u}_{k,i}^* \mathbf{d}_k(i). \quad (9.61)$$

To verify that (9.60) holds, we simply multiply both sides of (9.58) by  $\mathbf{u}_{k,i}^*$  from the left, compute the expectations of both sides, and use the fact that  $\mathbf{v}_k(i)$  has zero mean and is assumed to be independent of  $\{\mathbf{u}_{\ell,j}\}$  for all times  $j$  and nodes  $\ell$ . However, the difference in relation to the earlier normal equations (9.15) is that the matrix  $R_{u,k}$  is not the actual covariance matrix of  $\mathbf{u}_{k,i}$  any longer. When  $\mathbf{u}_{k,i}$  is not zero mean, its covariance matrix is instead defined as:

$$\begin{aligned} \text{cov}_{u,k} &\triangleq \mathbb{E}(\mathbf{u}_{k,i} - \mathbf{u}_k^o)^* (\mathbf{u}_{k,i} - \mathbf{u}_k^o) \\ &= \mathbb{E}\mathbf{u}_{k,i}^* \mathbf{u}_{k,i} - \mathbf{u}_k^{o*} \mathbf{u}_k^o, \end{aligned} \quad (9.62)$$

so that

$$R_{u,k} = \text{cov}_{u,k} + \mathbf{u}_k^{o*} \mathbf{u}_k^o. \quad (9.63)$$

We conclude from this relation that  $R_{u,k}$  is positive-definite (and, hence, invertible) so that expression (9.60) is justified. This is because the covariance matrix,  $\text{cov}_{u,k}$ , is itself positive-definite. Indeed, some algebra applied to the difference  $\mathbf{u}_{k,i} - \mathbf{u}_k^o$  from (9.51) shows that

$$\text{cov}_{u,k} = \begin{bmatrix} (\mathbf{u}_k^{o\perp})^* & (\mathbf{u}_k^o)^* \end{bmatrix} \begin{bmatrix} \sigma_{\alpha,k}^2 & \\ & \sigma_{\beta,k}^2 \end{bmatrix} \begin{bmatrix} \mathbf{u}_k^{o\perp} \\ \mathbf{u}_k^o \end{bmatrix}, \quad (9.64)$$

where the matrix

$$\begin{bmatrix} \mathbf{u}_k^{o\perp} \\ \mathbf{u}_k^o \end{bmatrix} \quad (9.65)$$

is full rank since the rows  $\{\mathbf{u}_k^o, \mathbf{u}_k^{o\perp}\}$  are linearly independent vectors.

Therefore, each node  $k$  can determine  $w^o$  on its own by solving the same minimum mean-square-error estimation problem (9.16). This solution method requires knowledge of the moments  $\{r_{du,k}, R_{u,k}\}$  and, according to (9.20), each agent  $k$  would then attain an MSE level that is equal to the noise power level,  $\sigma_{v,k}^2$ , at its location.

Alternatively, when the statistical information  $\{r_{du,k}, R_{u,k}\}$  is not available beforehand, each agent  $k$  can estimate  $w^o$  iteratively by feeding data  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  into the adaptive implementation (9.26) and (9.27). In this case, each agent  $k$  will achieve the performance level shown earlier in (9.36) and (9.37), with the limiting performance being again dependent on the local noise power level,  $\sigma_{v,k}^2$ . Therefore, nodes with larger noise power will perform worse than nodes with smaller noise power. However, since all nodes are observing distances and direction vectors towards the same target location  $w^o$ , it is natural to expect cooperation among the nodes to be beneficial. As we are going to see, starting from the next section, one way to achieve cooperation and improve performance is by developing algorithms that solve the same global cost function (9.38) in an adaptive and distributed manner, by using algorithms such as (9.153) and (9.154) further ahead.

### 3.09.2.3.1 Role of adaptation

The localization application helps highlight one of the main advantages of adaptation, namely, the ability of adaptive implementations to learn and track changing statistical conditions. For example, in the context of mobile networks, where nodes can move closer or further away from a target, the location vector for each agent  $k$  becomes time-dependent, say,  $p_{k,i} = \text{col}\{x_k(i), y_k(i)\}$ . In this case, the actual distance and direction vector between agent  $k$  and the target also vary with time and become:

$$r_k^o(i) = \|w^o - p_{k,i}\|, \quad u_{k,i}^o = \frac{(w^o - p_{k,i})^T}{\|w^o - p_{k,i}\|}. \quad (9.66)$$

The noisy distance measurement to the target is then:

$$\mathbf{r}_k(i) = r_k^o(i) + \mathbf{v}_k(i), \quad (9.67)$$

where the variance of  $\mathbf{v}_k(i)$  now depends on time as well:

$$\sigma_{v,k}^2(i) \triangleq \mathbb{E}|\mathbf{v}_k(i)|^2. \quad (9.68)$$

In the context of mobile networks, it is reasonable to assume that the variance of  $\mathbf{v}_k(i)$  varies both with time and with the distance to the target: the closer the node is to the target, the less noisy the measurement of the distance is expected to be. Similar remarks hold for the variances of the noises  $\boldsymbol{\alpha}_k(i)$  and  $\boldsymbol{\beta}_k(i)$  that perturb the measurement of the direction vector, say,

$$\sigma_{\alpha,k}^2(i) \triangleq \mathbb{E}|\boldsymbol{\alpha}_k(i)|^2, \quad \sigma_{\beta,k}^2(i) \triangleq \mathbb{E}|\boldsymbol{\beta}_k(i)|^2, \quad (9.69)$$

where now

$$\mathbf{u}_{k,i} = u_{k,i}^o + \boldsymbol{\alpha}_k(i)u_{k,i}^{o\perp} + \boldsymbol{\beta}_k(i)u_{k,i}^o. \quad (9.70)$$

The same arguments that led to (9.58) can be repeated to lead to the same model, except that now the means of the variables  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  become time-dependent as well:

$$\mathbb{E}\mathbf{u}_{k,i} = u_{k,i}^o, \quad \mathbb{E}\mathbf{d}_k(i) = r_k^o(i) + u_{k,i}^o p_{k,i}. \quad (9.71)$$

Nevertheless, adaptive solutions (whether cooperative or non-cooperative), are able to track such time-variations because these solutions work directly with the observations  $\{d_k(i), u_{k,i}\}$  and the successive observations will reflect the changing statistical profile of the data. In general, adaptive solutions are able to track changes in the underlying signal statistics rather well [4,5], as long as the rate of non-stationarity is slow enough for the filter to be able to follow the changes.

### 3.09.2.4 Application: collaborative spectral sensing

Our fourth and last example to illustrate the role of mean-square-error estimation and cooperation relates to spectrum sensing for cognitive radio applications. Cognitive radio systems involve two types of users: primary users and secondary users. To avoid causing harmful interference to incumbent primary users, unlicensed cognitive radio devices need to detect unused frequency bands even at low signal-to-noise (SNR) conditions [13–16]. One way to carry out spectral sensing is for each secondary user to estimate the aggregated power spectrum that is transmitted by all active primary users, and to locate unused frequency bands within the estimated spectrum. This step can be performed by the secondary users with or without cooperation.

Thus, consider a communications environment consisting of  $Q$  primary users and  $N$  secondary users. Let  $S_q(e^{j\omega})$  denote the power spectrum of the signal transmitted by primary user  $q$ . To facilitate estimation of the spectral profile by the secondary users, we assume that each  $S_q(e^{j\omega})$  can be represented as a linear combination of some basis functions,  $\{f_m(e^{j\omega})\}$ , say,  $B$  of them [17]:

$$S_q(e^{j\omega}) = \sum_{m=1}^B \beta_{qm} f_m(e^{j\omega}), \quad q = 1, 2, \dots, Q. \quad (9.72)$$

In this representation, the scalars  $\{\beta_{qm}\}$  denote the coefficients of the basis expansion for user  $q$ . The variable  $\omega \in [-\pi, \pi]$  denotes the normalized angular frequency measured in radians/sample. The power spectrum is often symmetric about the vertical axis,  $\omega = 0$ , and therefore it is sufficient to focus on the interval  $\omega \in [0, \pi]$ . There are many ways by which the basis functions,  $\{f_m(e^{j\omega})\}$ , can be selected. The following is one possible construction for illustration purposes. We divide the interval  $[0, \pi]$  into  $B$  identical intervals and denote their center frequencies by  $\{\omega_m\}$ . We then place a Gaussian pulse at each location  $\omega_m$  and control its width through the selection of its standard deviation,  $\sigma_m$ , i.e.,

$$f_m(e^{j\omega}) \triangleq e^{-\frac{(\omega-\omega_m)^2}{\sigma_m^2}}. \quad (9.73)$$

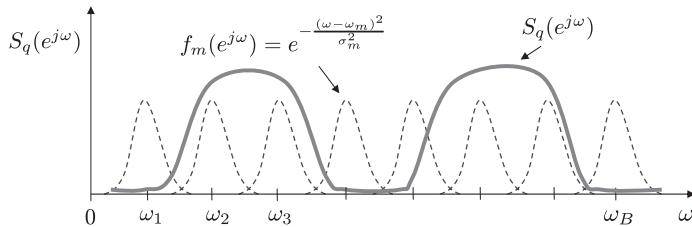
Figure 9.7 illustrates this construction. The parameters  $\{\omega_m, \sigma_m\}$  are selected by the designer and are assumed to be known. For a sufficiently large number,  $B$ , of basis functions, the representation (9.72) can approximate well a large class of power spectra.

We collect the combination coefficients  $\{\beta_{qm}\}$  for primary user  $q$  into a column vector  $w_q$ :

$$w_q \triangleq \text{col}\{\beta_{q1}, \beta_{q2}, \beta_{q3}, \dots, \beta_{qB}\} \quad (B \times 1) \quad (9.74)$$

and collect the basis functions into a row vector:

$$f_\omega \triangleq [f_1(e^{j\omega}) \quad f_2(e^{j\omega}) \quad \dots \quad f_B(e^{j\omega})] \quad (1 \times B). \quad (9.75)$$

**FIGURE 9.7**

The interval  $[0, \pi]$  is divided into  $B$  sub-intervals of equal width; the center frequencies of the sub-intervals are denoted by  $\{\omega_m\}$ . A power spectrum  $S_q(e^{j\omega})$  is approximated as a linear combination of Gaussian basis functions centered on the  $\{\omega_m\}$ .

Then, the power spectrum (9.72) can be expressed in the alternative inner-product form:

$$S_q(e^{j\omega}) = f_\omega w_q. \quad (9.76)$$

Let  $p_{qk}$  denote the path loss coefficient from primary user  $q$  to secondary user  $k$ . When the transmitted spectrum  $S_q(e^{j\omega})$  travels from primary user  $q$  to secondary user  $k$ , the spectrum that is sensed by node  $k$  is  $p_{qk}S_q(e^{j\omega})$ . We assume in this example that the path loss factors  $\{p_{qk}\}$  are known and that they have been determined during a prior training stage involving each of the primary users with each of the secondary users. The training is usually repeated at regular intervals of time to accommodate the fact that the path loss coefficients can vary (albeit slowly) over time. Figure 9.8 depicts a cognitive radio system with 2 primary users and 10 secondary users. One of the secondary users (user 5) is highlighted and the path loss coefficients from the primary users to its location are indicated; similar path loss coefficients can be assigned to all other combinations involving primary and secondary users.

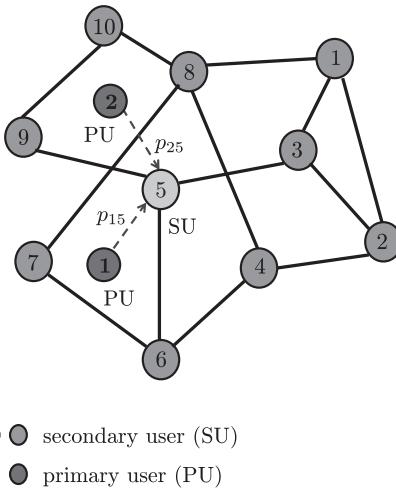
Each user  $k$  senses the *aggregate* effect of the power spectra that are transmitted by all active primary users. Therefore, adding the effect of all primary users, we find that the power spectrum that arrives at secondary user  $k$  is given by:

$$\begin{aligned} S_k(e^{j\omega}) &= \sum_{q=1}^Q p_{qk}S_q(e^{j\omega}) + \sigma_k^2 \\ &= \sum_{q=1}^Q p_{qk}f_\omega w_q + \sigma_k^2 \\ &\triangleq u_{k,\omega}w^o + \sigma_k^2, \end{aligned} \quad (9.77)$$

where  $\sigma_k^2$  denotes the receiver noise power at node  $k$ , and where we introduced the following vector quantities:

$$w^o \triangleq \text{col}\{w_1, w_2, \dots, w_Q\} \quad (BQ \times 1), \quad (9.78)$$

$$u_{k,\omega} \triangleq [p_{1k}f_\omega \quad p_{2k}f_\omega \quad \cdots \quad p_{Qk}f_\omega] \quad (1 \times BQ). \quad (9.79)$$

**FIGURE 9.8**

A network of secondary users in the presence of two primary users. One of the secondary users is highlighted and the path loss coefficients from the primary users to its location are indicated as  $p_{15}$  and  $p_{25}$ .

The vector  $w^o$  is the collection of all combination coefficients for all  $Q$  primary users. The vector  $u_{k,\omega}$  contains the path loss coefficients from all primary users to user  $k$ . Now, at every time instant  $i$ , user  $k$  observes its received power spectrum,  $S_k(e^{j\omega})$ , over a discrete grid of frequencies,  $\{\omega_r\}$ , in the interval  $[0, \pi]$  in the presence of additive measurement noise. We denote these measurements by:

$$\mathbf{d}_{k,r}(i) = \mathbf{u}_{k,\omega_r} w^o + \sigma_k^2 + \mathbf{v}_{k,r}(i), \quad r = 1, 2, \dots, R. \quad (9.80)$$

The term  $\mathbf{v}_{k,r}(i)$  denotes sampling noise and is assumed to have zero mean and variance  $\sigma_{v,k}^2$ ; it is also assumed to be temporally white and spatially independent; and is also independent of all other random variables. Since the row vectors  $\mathbf{u}_{k,\omega}$  in (9.79) are defined in terms of the path loss coefficients  $\{p_{qk}\}$ , and since these coefficients are estimated and subject to noisy distortions, we model the  $\mathbf{u}_{k,\omega_r}$  as zero-mean random variables in (9.80) and use the boldface notation for them.

Observe that in this application, each node  $k$  collects  $R$  measurements at every time instant  $i$  and not only a single measurement, as was the case with the three examples discussed in the previous sections (AR modeling, MA modeling, and localization). The implication of this fact is that we now deal with an estimation problem that involves vector measurements instead of scalar measurements at each node. The solution structure continues to be the same. We collect the  $R$  measurements at node  $k$  at time  $i$  into vectors and introduce the  $R \times 1$  quantities:

$$\mathbf{d}_{k,i} \triangleq \begin{bmatrix} \mathbf{d}_{k,1}(i) - \sigma_k^2 \\ \mathbf{d}_{k,2}(i) - \sigma_k^2 \\ \vdots \\ \mathbf{d}_{k,R}(i) - \sigma_k^2 \end{bmatrix}, \quad \mathbf{v}_{k,i} \triangleq \begin{bmatrix} \mathbf{v}_{k,1}(i) \\ \mathbf{v}_{k,2}(i) \\ \vdots \\ \mathbf{v}_{k,R}(i) \end{bmatrix} \quad (9.81)$$

and the regression matrix:

$$\mathbf{U}_{k,i} \triangleq \begin{bmatrix} \mathbf{u}_{k,\omega_1} \\ \mathbf{u}_{k,\omega_2} \\ \vdots \\ \mathbf{u}_{k,\omega_R} \end{bmatrix} \quad (R \times QB). \quad (9.82)$$

The time subscript in  $\mathbf{U}_{k,i}$  is used to model the fact that the path loss coefficients can change over time due to the possibility of node mobility. With the above notation, expression (9.80) is equivalent to the linear model:

$$\boxed{\mathbf{d}_{k,i} = \mathbf{U}_{k,i}w^o + \mathbf{v}_{k,i}}. \quad (9.83)$$

Compared to the earlier examples (9.9), (9.44), and (9.58), the main difference now is that each agent  $k$  collects a *vector* of measurements,  $\mathbf{d}_{k,i}$ , as opposed to the scalar  $d_k(i)$ , and its regression data are represented by the matrix quantity,  $\mathbf{U}_{k,i}$ , as opposed to the row vector  $\mathbf{u}_{k,i}$ . Nevertheless, the estimation approach will continue to be the same. In cognitive network applications, the secondary users are interested in estimating the aggregate power spectrum of the primary users in order for the secondary users to identify vacant frequency bands that can be used by them. In the context of model (9.83), this amounts to determining the parameter vector  $w^o$  since knowledge of its entries allows each secondary user to reconstruct the aggregate power spectrum defined by:

$$S_A(e^{j\omega}) \triangleq \sum_{q=1}^Q S_q(e^{j\omega}) = (\mathbb{1}_Q^T \otimes f_\omega) w^o, \quad (9.84)$$

where the notation  $\otimes$  denotes the Kronecker product operation, and  $\mathbb{1}_Q$  denotes a  $Q \times 1$  vector whose entries are all equal to one.

As before, we can again verify that, in view of (9.83), the desired parameter vector  $w^o$  satisfies the same normal equations:

$$R_{dU,k} = R_{U,k}w^o \iff w^o = R_{U,k}^{-1}R_{dU,k}, \quad (9.85)$$

where the moments  $\{R_{dU,k}, R_{U,k}\}$  are now defined by

$$R_{dU,k} \triangleq \mathbb{E}\mathbf{U}_{k,i}^* \mathbf{d}_{k,i} \quad (QB \times 1), \quad (9.86)$$

$$R_{U,k} \triangleq \mathbb{E}\mathbf{U}_{k,i}^* \mathbf{U}_{k,i} \quad (QB \times QB). \quad (9.87)$$

Therefore, each secondary user  $k$  can determine  $w^o$  on its own by solving the following minimum mean-square-error estimation problem:

$$\min_w \mathbb{E}\|\mathbf{d}_{k,i} - \mathbf{U}_{k,i}w\|^2. \quad (9.88)$$

This solution method requires knowledge of the moments  $\{R_{dU,k}, R_{U,k}\}$  and, in an argument similar to the one that led to (9.20), it can be verified that each agent  $k$  would attain an MSE performance level that is equal to the noise power level,  $\sigma_{v,k}^2$ , at its location.

Alternatively, when the statistical information  $\{R_{dU,k}, R_{U,k}\}$  is not available, each secondary user  $k$  can estimate  $w^o$  iteratively by feeding data  $\{\mathbf{d}_{k,i}, \mathbf{U}_{k,i}\}$  into an adaptive implementation similar to (9.26) and (9.27), such as the following vector LMS recursion:

$$\mathbf{e}_{k,i} = \mathbf{d}_{k,i} - \mathbf{U}_{k,i} \mathbf{w}_{k,i-1}, \quad (9.89)$$

$$\mathbf{w}_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \mathbf{U}_{k,i}^* \mathbf{e}_{k,i}. \quad (9.90)$$

In this case, each secondary user  $k$  will achieve performance levels similar to (9.36) and (9.37) with  $M$  replaced by  $QB$  and  $R_{u,k}$  replaced by  $R_{U,k}$ . The performance will again be dependent on the local noise level,  $\sigma_{v,k}^2$ . As a result, secondary users with larger noise power will perform worse than secondary users with smaller noise power. However, since all secondary users are observing data arising from the same underlying model  $w^o$ , it is natural to expect cooperation among the users to be beneficial. As we are going to see, starting from the next section, one way to achieve cooperation and improve performance is by developing algorithms that solve the following global cost function in an adaptive and distributed manner:

$$\min_w \sum_{k=1}^N \mathbb{E} \|\mathbf{d}_{k,i} - \mathbf{U}_{k,i} w\|^2. \quad (9.91)$$

### 3.09.3 Distributed optimization via diffusion strategies

The examples in the previous section were meant to illustrate how MSE cost functions and linear models are useful design tools and how they arise frequently in applications. We now return to problem (9.1) and study the distributed optimization of global cost functions such as (9.39), where  $J^{\text{glob}}(w)$  is assumed to consist of the sum of individual components. Specifically, we are now interested in solving optimization problems of the type:

$$\min_w \sum_{k=1}^N J_k(w), \quad (9.92)$$

where each  $J_k(w)$  is assumed to be differentiable and convex over  $w$ . Although the algorithms presented in this chapter apply to more general situations, we shall nevertheless focus on mean-square-error cost functions of the form:

$$J_k(w) \triangleq \mathbb{E} |\mathbf{d}_k(i) - \mathbf{u}_{k,i} w|^2, \quad (9.93)$$

where  $w$  is an  $M \times 1$  column vector, and the random processes  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  are assumed to be jointly wide-sense stationary with zero-mean and second-order moments:

$$\sigma_{d,k}^2 \triangleq \mathbb{E} |\mathbf{d}_k(i)|^2, \quad (9.94)$$

$$R_{u,k} \triangleq \mathbb{E} \mathbf{u}_{k,i}^* \mathbf{u}_{k,i} > 0 \quad (M \times M), \quad (9.95)$$

$$r_{du,k} \triangleq \mathbb{E} \mathbf{d}_k(i) \mathbf{u}_{k,i}^* \quad (M \times 1). \quad (9.96)$$

It is clear that each  $J_k(w)$  is quadratic in  $w$  since, after expansion, we get

$$J_k(w) = \sigma_{d,k}^2 - w^* r_{du,k} - r_{du,k}^* w + w^* R_{u,k} w. \quad (9.97)$$

A completion-of-squares argument shows that  $J_k(w)$  can be expressed as the sum of two squared terms, i.e.,

$$J_k(w) = \left( \sigma_{d,k}^2 - r_{du,k}^* R_{u,k}^{-1} r_{du,k} \right) + (w - w^o)^* R_{u,k} (w - w^o) \quad (9.98)$$

or, more compactly,

$$\boxed{J_k(w) = J_{k,\min} + \|w - w^o\|_{R_{u,k}}^2}, \quad (9.99)$$

where  $w^o$  denotes the minimizer of  $J_k(w)$  and is given by

$$w^o \triangleq R_{u,k}^{-1} r_{du,k} \quad (9.100)$$

and  $J_{k,\min}$  denotes the minimum value of  $J_k(w)$  when evaluated at  $w = w^o$ :

$$\boxed{J_{k,\min} \triangleq \sigma_{d,k}^2 - r_{du,k}^* R_{u,k}^{-1} r_{du,k} = J_k(w^o)}. \quad (9.101)$$

Observe that this value is necessarily nonnegative since it can be viewed as the Schur complement of the following covariance matrix:

$$\mathbb{E} \left( \begin{bmatrix} \mathbf{d}_k^*(i) \\ \mathbf{u}_{k,i}^* \end{bmatrix} \begin{bmatrix} \mathbf{d}_k(i) & \mathbf{u}_{k,i} \end{bmatrix} \right) = \begin{bmatrix} \sigma_{d,k}^2 & r_{du,k}^* \\ r_{du,k} & R_{u,k} \end{bmatrix} \quad (9.102)$$

and covariance matrices are nonnegative-definite.

The choice of the quadratic form (9.93) or (9.97) for  $J_k(w)$  is useful for many applications, as was already illustrated in the previous section for examples involving AR modeling, MA modeling, localization, and spectral sensing. Other choices for  $J_k(w)$  are of course possible and these choices can even be different for different nodes. It is sufficient in this chapter to illustrate the main concepts underlying diffusion adaptation by focusing on the useful case of MSE cost functions of the form (9.97); still, most of the derivations and arguments in the coming sections can be extended beyond MSE optimization to more general cost functions and to situations where the minimizers of the individual costs do not necessarily occur at the same location—as already shown in [1–3]; see also Section 3.09.10.4.

The positive-definiteness of the covariance matrices  $\{R_{u,k}\}$  ensures that each  $J_k(w)$  in (9.97) is strictly convex, as well as  $J^{\text{glob}}(w)$  from (9.39). Moreover, all these cost functions have a unique minimum at the same  $w^o$ , which satisfies the normal equations:

$$R_{u,k} w^o = r_{du,k}, \quad \text{for every } k = 1, 2, \dots, N. \quad (9.103)$$

Therefore, given knowledge of  $\{r_{du,k}, R_{u,k}\}$ , each node can determine  $w^o$  on its own by solving (9.103). One then wonders about the need to seek distributed cooperative and adaptive solutions in this case. There are a couple of reasons:

- a. First, even for MSE cost functions, it is often the case that the required moments  $\{r_{du,k}, R_{u,k}\}$  are not known beforehand. In this case, the optimal  $w^o$  cannot be determined from the solution of the normal equations (9.103). The alternative methods that we shall describe will lead to adaptive techniques that enable each node  $k$  to estimate  $w^o$  directly from data realizations.

- b.** Second, since adaptive strategies rely on instantaneous data, these strategies possess powerful tracking abilities. Even when the moments vary with time due to non-stationary behavior (such as  $w^o$  changing with time), these changes will be reflected in the observed data and will in turn influence the behavior of the adaptive construction. This is one of the key advantages of adaptive strategies: they enable learning and tracking in real-time.
- c.** Third, cooperation among nodes is generally beneficial. When nodes act individually, their performance is limited by the noise power level at their location. In this way, some nodes can perform significantly better than other nodes. On the other hand, when nodes cooperate with their neighbors and share information during the adaptation process, we will see that performance can be improved across the network.

### 3.09.3.1 Relating the global cost to neighborhood costs

Let us therefore consider the optimization of the following global cost function:

$$J^{\text{glob}}(w) = \sum_{k=1}^N J_k(w), \quad (9.104)$$

where  $J_k(w)$  is given by (9.93) or (9.97). Our strategy to optimize  $J^{\text{glob}}(w)$  in a distributed manner is based on two steps [2, 18]. First, using a completion-of-squares argument (or, equivalently, a second-order Taylor series expansion), we approximate the global cost function (9.104) by an alternative local cost that is amenable to distributed optimization. Then, each node will optimize the alternative cost via a steepest-descent method.

To motivate the distributed diffusion-based approach, we start by introducing a set of nonnegative coefficients  $\{c_{k\ell}\}$  that satisfy two conditions:

for  $k = 1, 2, \dots, N$  :

$$c_{k\ell} \geq 0, \quad \sum_{\ell=1}^N c_{k\ell} = 1, \quad \text{and} \quad c_{k\ell} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad (9.105)$$

where  $\mathcal{N}_k$  denotes the neighborhood of node  $k$ . Condition (9.105) means that for every node  $k$ , the sum of the coefficients  $\{c_{k\ell}\}$  that relate it to its neighbors is one. The coefficients  $\{c_{k\ell}\}$  are free parameters that are chosen by the designer; obviously, as shown later in Theorem 9.6.8, their selection will have a bearing on the performance of the resulting algorithms. If we collect the entries  $\{c_{k\ell}\}$  into an  $N \times N$  matrix  $C$ , so that the  $k$ th row of  $C$  is formed of  $\{c_{k\ell}, \ell = 1, 2, \dots, N\}$ , then condition (9.105) translates into saying that each of *row* of  $C$  adds up to one, i.e.,

$$C\mathbb{1} = \mathbb{1}, \quad (9.106)$$

where the notation  $\mathbb{1}$  denotes an  $N \times 1$  column vector with all its entries equal to one:

$$\mathbb{1} \triangleq \text{col}\{1, 1, \dots, 1\}. \quad (9.107)$$

We say that  $C$  is a right stochastic matrix. Using the coefficients  $\{c_{k\ell}\}$  so defined, we associate with each node  $\ell$ , a local cost function of the following form:

$$J_\ell^{\text{loc}}(w) \triangleq \sum_{k \in \mathcal{N}_\ell} c_{k\ell} J_k(w). \quad (9.108)$$

This cost consists of a weighted combination of the individual costs of the neighbors of node  $\ell$  (including  $\ell$  itself)—see Figure 9.9. Since the  $\{c_{k\ell}\}$  are all nonnegative and each  $J_k(w)$  is strictly convex, then  $J_\ell^{\text{loc}}(w)$  is also strictly convex and its minimizer occurs at the same  $w = w^o$ . Using the alternative representation (9.99) for the individual  $J_k(w)$ , we can re-express the local cost  $J_\ell^{\text{loc}}(w)$  as

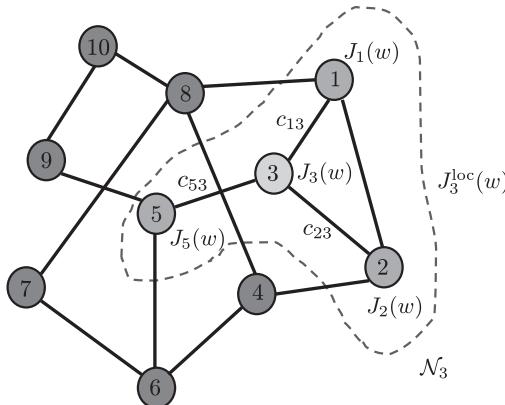
$$J_\ell^{\text{loc}}(w) = \sum_{k \in \mathcal{N}_\ell} c_{k\ell} J_{k,\min} + \sum_{k \in \mathcal{N}_\ell} c_{k\ell} \|w - w^o\|_{R_{u,k}}^2 \quad (9.109)$$

or, equivalently,

$$\boxed{J_\ell^{\text{loc}}(w) = J_{\ell,\min} + \|w - w^o\|_{R_\ell}^2}, \quad (9.110)$$

where  $J_{\ell,\min}$  corresponds to the minimum value of  $J_\ell^{\text{loc}}(w)$  at the minimizer  $w = w^o$ :

$$J_{\ell,\min} \triangleq \sum_{k \in \mathcal{N}_\ell} c_{k\ell} J_{k,\min} \quad (9.111)$$



$$c_{13} + c_{23} + c_{43} + c_{53} = 1$$

$$J_3^{\text{loc}}(w) = c_{13} J_1(w) + c_{23} J_2(w) + c_{43} J_4(w) + c_{53} J_5(w)$$

**FIGURE 9.9**

A network with  $N = 10$  nodes. The nodes in the neighborhood of node 3 are highlighted with their individual cost functions, and with the combination weights  $\{c_{13}, c_{23}, c_{43}, c_{53}\}$  along the connecting edges; there is also a combination weight associated with node 3 and is denoted by  $c_{33}$ . The expression for the local cost function,  $J_3^{\text{loc}}(w)$ , is also shown in the figure.

and  $R_\ell$  is a positive-definite weighting matrix defined by:

$$R_\ell \triangleq \sum_{k \in \mathcal{N}_\ell} c_{k\ell} R_{u,k}. \quad (9.112)$$

That is,  $R_\ell$  is a weighted combination of the covariance matrices in the neighborhood of node  $\ell$ . Equality (9.110) amounts to a (second-order) Taylor series expansion of  $J_\ell^{\text{loc}}(w)$  around  $w = w^o$ . Note that the right-hand side consists of two terms: the minimum cost and a weighted quadratic term in the difference  $(w - w^o)$ .

Now note that we can express  $J^{\text{glob}}(w)$  from (9.104) as follows:

$$\begin{aligned} J^{\text{glob}}(w) &\stackrel{(9.105)}{=} \sum_{k=1}^N \left( \sum_{\ell=1}^N c_{k\ell} \right) J_k(w) \\ &= \sum_{\ell=1}^N \left( \sum_{k=1}^N c_{k\ell} J_k(w) \right) \\ &\stackrel{(9.108)}{=} \sum_{\ell=1}^N J_\ell^{\text{loc}}(w) \\ &= J_k^{\text{loc}}(w) + \sum_{\ell \neq k}^N J_\ell^{\text{loc}}(w). \end{aligned} \quad (9.113)$$

Substituting (9.110) into the second term on the right-hand side of the above expression gives:

$$J^{\text{glob}}(w) = J_k^{\text{loc}}(w) + \sum_{\ell \neq k} \|w - w^o\|_{R_\ell}^2 + \sum_{\ell \neq k} J_{\ell,\min}^{\text{loc}}. \quad (9.114)$$

The last term in the above expression does not depend on  $w$ . Therefore, minimizing  $J^{\text{glob}}(w)$  over  $w$  is equivalent to minimizing the following alternative global cost:

$$J^{\text{glob}'}(w) = J_k^{\text{loc}}(w) + \sum_{\ell \neq k} \|w - w^o\|_{R_\ell}^2.$$

(9.115)

Expression (9.115) relates the optimization of the original global cost function,  $J^{\text{glob}}(w)$  or its equivalent  $J^{\text{glob}'}(w)$ , to the newly-introduced local cost function  $J_k^{\text{loc}}(w)$ . The relation is through the second term on the right-hand side of (9.115), which corresponds to a sum of quadratic factors involving the minimizer  $w^o$ ; this term tells us how the local cost  $J_k^{\text{loc}}(w)$  can be corrected to the global cost  $J^{\text{glob}'}(w)$ . Obviously, the minimizer  $w^o$  that appears in the correction term is not known since the nodes wish to determine its value. Likewise, not all the weighting matrices  $R_\ell$  are available to node  $k$ ; only those matrices that originate from its neighbors can be assumed to be available. Still, expression (9.115) suggests a useful way to replace  $J_k^{\text{loc}}$  by another local cost that is closer to  $J^{\text{glob}'}(w)$ . This alternative cost will be shown to lead to a powerful distributed solution to optimize  $J^{\text{glob}}(w)$  through localized interactions.

Our first step is to limit the summation on the right-hand side of (9.115) to the neighbors of node  $k$  (since every node  $k$  can only have access to information from its neighbors). We thus introduce the modified cost function at node  $k$ :

$$J_k^{\text{glob}'}(w) \triangleq J_k^{\text{loc}}(w) + \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} \|w - w^o\|_{R_\ell}^2. \quad (9.116)$$

The cost functions  $J_k^{\text{loc}}(w)$  and  $J_k^{\text{glob}'}(w)$  are both associated with node  $k$ ; the difference between them is that the expression for the latter is closer to the global cost function (9.115) that we want to optimize.

The weighting matrices  $\{R_\ell\}$  that appear in (9.116) may or may not be available because the second-order moments  $\{R_{u,\ell}\}$  may or may not be known beforehand. If these moments are known, then we can proceed with the analysis by assuming knowledge of the  $\{R_\ell\}$ . However, the more interesting case is when these moments are not known. This is generally the case in practice, especially in the context of adaptive solutions and problems involving non-stationary data. Often, nodes can only observe realizations  $\{u_{\ell,i}\}$  of the regression data  $\{\mathbf{u}_{\ell,i}\}$  arising from distributions whose covariance matrices are the unknown  $\{R_{u,\ell}\}$ . One way to address the difficulty is to replace each of the weighted norms  $\|w - w^o\|_{R_\ell}^2$  in (9.116) by a scaled multiple of the un-weighted norm, say,

$$\|w - w^o\|_{R_\ell}^2 \approx b_{\ell k} \|w - w^o\|^2, \quad (9.117)$$

where  $b_{\ell k}$  is some nonnegative coefficient; we are even allowing its value to change with the node index  $k$ . The above substitution amounts to having each node  $k$  approximate the  $\{R_\ell\}$  from its neighbors by multiples of the identity matrix

$$R_\ell \approx b_{\ell k} I_M. \quad (9.118)$$

Approximation (9.117) is reasonable in view of the fact that all vector norms are equivalent [19–21]; this norm property ensures that we can bound the weighted norm  $\|w - w^o\|_{R_\ell}^2$  by some constants multiplying the un-weighted norm  $\|w - w^o\|^2$ , say, as:

$$r_1 \|w - w^o\|^2 \leq \|w - w^o\|_{R_\ell}^2 \leq r_2 \|w - w^o\|^2 \quad (9.119)$$

for some positive constants  $(r_1, r_2)$ . Using the fact that the  $\{R_\ell\}$  are Hermitian positive-definite matrices, and calling upon the Rayleigh-Ritz characterization of eigenvalues [19,20], we can be more specific and replace the above inequalities by

$$\lambda_{\min}(R_\ell) \cdot \|w - w^o\|^2 \leq \|w - w^o\|_{R_\ell}^2 \leq \lambda_{\max}(R_\ell) \cdot \|w - w^o\|^2. \quad (9.120)$$

We note that approximations similar to (9.118) are common in stochastic approximation theory and they mark the difference between using a Newton's iterative method or a stochastic gradient method [5,22]; the former uses Hessian matrices as approximations for  $R_\ell$  and the latter uses multiples of the identity matrix. Furthermore, as the derivation will reveal, we do not need to worry at this stage about how to select the scalars  $\{b_{\ell k}\}$ ; they will end up being embedded into another set of coefficients  $\{a_{\ell k}\}$  that will be set by the designer or adjusted by the algorithm—see (9.132) further ahead.

Thus, we replace (9.116) by

$$J_k^{\text{glob}''}(w) = J_k^{\text{loc}}(w) + \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} \|w - w^o\|^2. \quad (9.121)$$

The argument so far has suggested how to modify  $J_k^{\text{loc}}(w)$  from (9.108) and replace it by the cost (9.121) that is closer in form to the global cost function (9.115). If we replace  $J_k^{\text{loc}}(w)$  by its definition (9.108), we can rewrite (9.121) as

$$J_k^{\text{glob}''}(w) = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} J_{\ell}(w) + \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} \|w - w^o\|^2.$$

(9.122)

With the exception of the variable  $w^o$ , this approximate cost at node  $k$  relies solely on information that is available to node  $k$  from its neighborhood. We will soon explain how to handle the fact that  $w^o$  is not known beforehand to node  $k$ .

### 3.09.3.2 Steepest-descent iterations

Node  $k$  can apply a steepest-descent iteration to minimize  $J_k^{\text{glob}''}(w)$ . Let  $w_{k,i}$  denote the estimate for the minimizer  $w^o$  that is evaluated by node  $k$  at time  $i$ . Starting from an initial condition  $w_{k,-1}$ , node  $k$  can compute successive estimates iteratively as follows:

$$w_{k,i} = w_{k,i-1} - \mu_k \left[ \nabla_w J_k^{\text{glob}''}(w_{k,i-1}) \right]^*, \quad i \geq 0, \quad (9.123)$$

where  $\mu_k$  is a small positive step-size parameter, and the notation  $\nabla_w J(a)$  denotes the gradient vector of the function  $J(w)$  relative to  $w$  and evaluated at  $w = a$ . The step-size parameter  $\mu_k$  can be selected to vary with time as well. One choice that is common in the optimization literature [5, 22, 23] is to replace  $\mu_k$  in (9.123) by step-size sequences  $\{\mu(i) \geq 0\}$  that satisfy the two conditions (9.25). However, such step-size sequences are not suitable for applications that require continuous learning because they turn off adaptation as  $i \rightarrow \infty$ ; the steepest-descent iteration (9.123) would stop updating since  $\mu_k(i)$  would be tending towards zero. For this reason, we shall focus mainly on the constant step-size case described by (9.123) since we are interested in developing distributed algorithms that will endow networks with continuous adaptation abilities.

Returning to (9.123) and computing the gradient vector of (9.122) we get:

$$w_{k,i} = w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_{\ell}(w_{k,i-1})]^* - \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} (w_{k,i-1} - w^o). \quad (9.124)$$

Using the expression for  $J_{\ell}(w)$  from (9.97) we arrive at

$$w_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} w_{k,i-1}) + \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} (w^o - w_{k,i-1}). \quad (9.125)$$

This iteration indicates that the update from  $w_{k,i-1}$  to  $w_{k,i}$  involves adding two correction terms to  $w_{k,i-1}$ . Among many other forms, we can implement the update in two successive steps by adding one correction term at a time, say, as follows:

$$\psi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} w_{k,i-1}), \quad (9.126)$$

$$w_{k,i} = \psi_{k,i} + \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} (w^o - w_{k,i-1}). \quad (9.127)$$

Step (9.126) updates  $w_{k,i-1}$  to an intermediate value  $\psi_{k,i}$  by using local gradient vectors from the neighborhood of node  $k$ . Step (9.127) further updates  $\psi_{k,i}$  to  $w_{k,i}$ . Two issues stand out from examining (9.127):

- a. First, iteration (9.127) requires knowledge of the minimizer  $w^o$ . Neither node  $k$  nor its neighbors know the value of the minimizer; each of these nodes is actually performing steps similar to (9.126) and (9.127) to estimate the minimizer. However, each node  $\ell$  has a readily available approximation for  $w^o$ , which is its local intermediate estimate  $\psi_{\ell,i}$ . Therefore, we replace  $w^o$  in (9.127) by  $\psi_{\ell,i}$ . This step helps diffuse information throughout the network. This is because each neighbor of node  $k$  determines its estimate  $\psi_{\ell,i}$  by processing information from its own neighbors, which process information from their neighbors, and so forth.
- b. Second, the intermediate value  $\psi_{k,i}$  at node  $k$  is generally a better estimate for  $w^o$  than  $w_{k,i-1}$  since it is obtained by incorporating information from the neighbors through the first step (9.126). Therefore, we further replace  $w_{k,i-1}$  in (9.127) by  $\psi_{k,i}$ . This step is reminiscent of incremental-type approaches to optimization, which have been widely studied in the literature [24–27].

With the substitutions described in items (a) and (b) above, we replace the second step (9.127) by

$$\begin{aligned} w_{k,i} &= \psi_{k,i} + \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} (\psi_{\ell,i} - \psi_{k,i}) \\ &= \left(1 - \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k}\right) \psi_{k,i} + \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k} \psi_{\ell,i}. \end{aligned} \quad (9.128)$$

Introduce the weighting coefficients:

$$a_{kk} \triangleq 1 - \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k}, \quad (9.129)$$

$$a_{\ell k} \triangleq \mu_k b_{\ell k}, \quad \ell \in \mathcal{N}_k \setminus \{k\}, \quad (9.130)$$

$$a_{\ell k} \triangleq 0, \quad \ell \notin \mathcal{N}_k, \quad (9.131)$$

and observe that, for sufficiently small step-sizes  $\mu_k$ , these coefficients are nonnegative and, moreover, they satisfy the conditions:

for  $k = 1, 2, \dots, N$  :

$$a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad \text{and} \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \quad (9.132)$$

Condition (9.132) means that for every node  $k$ , the sum of the coefficients  $\{a_{\ell k}\}$  that relate it to its neighbors is one. Just like the  $\{c_{\ell k}\}$ , from now on, we will treat the coefficients  $\{a_{\ell k}\}$  as free weighting parameters that are chosen by the designer according to (9.132); their selection will also have a bearing on the performance of the resulting algorithms—see Theorem 9.6.8. If we collect the entries  $\{a_{\ell k}\}$  into an  $N \times N$  matrix  $A$ , such that the  $k$ th column of  $A$  consists of  $\{a_{\ell k}, \ell = 1, 2, \dots, N\}$ , then condition (9.132) translates into saying that each *column* of  $A$  adds up to one:

$$\boxed{A^T \mathbf{1} = \mathbf{1}}. \quad (9.133)$$

We say that  $A$  is a left stochastic matrix.

### 3.09.3.3 Adapt-the-Combine (ATC) diffusion strategy

Using the coefficients  $\{a_{\ell k}\}$  so defined, we replace (9.126) and (9.128) by the following recursions for  $i \geq 0$ :

$$\begin{aligned} \psi_{k,i} &= w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} w_{k,i-1}) \\ (ATC \text{ strategy}) \quad w_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{aligned} \quad (9.134)$$

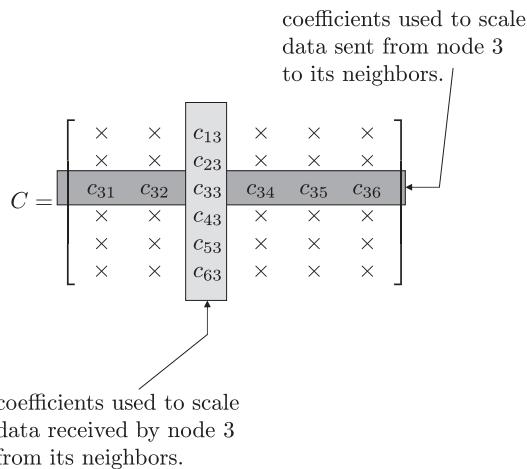
for some nonnegative coefficients  $\{c_{\ell k}, a_{\ell k}\}$  that satisfy conditions (9.106) and (9.133), namely,

$$\boxed{C\mathbf{1} = \mathbf{1}, \quad A^T \mathbf{1} = \mathbf{1}} \quad (9.135)$$

or, equivalently,

$$\begin{aligned} &\text{for } k = 1, 2, \dots, N: \\ c_{\ell k} \geq 0, \quad \sum_{k=1}^N c_{\ell k} &= 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} &= 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad (9.136)$$

To run algorithm (9.134), we only need to select the coefficients  $\{a_{\ell k}, c_{\ell k}\}$  that satisfy (9.135) or (9.136); there is no need to worry about the intermediate coefficients  $\{b_{\ell k}\}$  any longer since they have been blended into the  $\{a_{\ell k}\}$ . The scalars  $\{c_{\ell k}, a_{\ell k}\}$  that appear in (9.134) correspond to weighting coefficients

**FIGURE 9.10**

Interpretation of the columns and rows of combination matrices. The pair of entries  $\{c_{k\ell}, c_{\ell k}\}$  correspond to weighting coefficients used over the edge connecting nodes  $k$  and  $\ell$ . When nodes  $(k, \ell)$  are not neighbors, then these weights are zero.

over the edge linking node  $k$  to its neighbors  $\ell \in \mathcal{N}_k$ . Note that two sets of coefficients are used to scale the data that are being received by node  $k$ : one set of coefficients,  $\{c_{\ell k}\}$ , is used in the first step of (9.134) to scale the moment data  $\{r_{du,\ell}, R_{u,\ell}\}$ , and a second set of coefficients,  $\{a_{\ell k}\}$ , is used in the second step of (9.134) to scale the estimates  $\{\psi_{\ell,i}\}$ . Figure 9.10 explains what the entries on the columns and rows of the combination matrices  $\{A, C\}$  stand for using an example with  $N = 6$  and the matrix  $C$  for illustration. When the combination matrix is right-stochastic (as is the case with  $C$ ), each of its rows would add up to one. On the other hand, when the matrix is left-stochastic (as is the case with  $A$ ), each of its columns would add up to one.

At every time instant  $i$ , the ATC strategy (9.134) performs two steps. The first step is an *information exchange* step where node  $k$  receives from its neighbors their moments  $\{R_{u,\ell}, r_{du,\ell}\}$ . Node  $k$  combines this information and uses it to update its existing estimate  $w_{k,i-1}$  to an intermediate value  $\psi_{k,i}$ . All other nodes in the network are performing a similar step and updating their existing estimates  $\{w_{\ell,i-1}\}$  into intermediate estimates  $\{\psi_{\ell,i}\}$  by using information from their neighbors. The second step in (9.134) is an *aggregation* or *consultation* step where node  $k$  combines the intermediate estimates of its neighbors to obtain its updated estimate  $w_{k,i}$ . Again, all other nodes in the network are simultaneously performing a similar step. The reason for the name Adapt-then-Combine (ATC) strategy is that the first step in (9.134) will be shown to lead to an adaptive step, while the second step in (9.134) corresponds to a combination step. Hence, strategy (9.134) involves adaptation followed by combination or ATC for short. The reason for the qualification “diffusion” is that the combination step in (9.134) allows information to diffuse through the network in real time. This is because each of the estimates  $\psi_{\ell,i}$  is influenced by data beyond the immediate neighborhood of node  $k$ .

In the special case when  $C = I$ , so that no information exchange is performed but only the aggregation step, the ATC strategy (9.134) reduces to:

$$\boxed{\begin{array}{l} \text{(ATC strategy without} \\ \text{information exchange)} \end{array} \quad \begin{array}{l} \psi_{k,i} = w_{k,i-1} + \mu_k(r_{du,k} - R_{u,k}w_{k,i-1}) \\ w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array}}, \quad (9.137)$$

where the first step relies solely on the information  $\{R_{u,k}, r_{du,k}\}$  that is available locally at node  $k$ .

Observe in passing that the term that appears in the information exchange step of (9.134) is related to the gradient vectors of the local costs  $\{J_\ell(w)\}$  evaluated at  $w_{k,i-1}$ , i.e., it holds that

$$r_{du,\ell} - R_{u,\ell}w_{k,i-1} = -[\nabla_w J_\ell(w_{k,i-1})]^*, \quad (9.138)$$

so that the ATC strategy (9.134) can also be written in the following equivalent form:

$$\boxed{\begin{array}{l} \text{(ATC strategy)} \end{array} \quad \begin{array}{l} \psi_{k,i} = w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(w_{k,i-1})]^* \\ w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{array}}. \quad (9.139)$$

The significance of this general form is that it is applicable to optimization problems involving more general local costs  $J_\ell(w)$  that are not necessarily quadratic in  $w$ , as detailed in [1–3]—see also Section 3.09.10.4. The top part of Figure 9.11 illustrates the two steps involved in the ATC procedure for a situation where node  $k$  has three other neighbors labeled  $\{1, 2, \ell\}$ . In the first step, node  $k$  evaluates the gradient vectors of its neighbors at  $w_{k,i-1}$ , and subsequently aggregates the estimates  $\{\psi_{1,i}, \psi_{2,i}, \psi_{\ell,i}\}$  from its neighbors. The dotted arrows represent flow of information towards node  $k$  from its neighbors. The solid arrows represent flow of information from node  $k$  to its neighbors. The CTA diffusion strategy is discussed next.

### 3.09.3.4 Combine-then-Adapt (CTA) diffusion strategy

Similarly, if we return to (9.125) and add the second correction term first, then (9.126) and (9.127) are replaced by:

$$\psi_{k,i-1} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} b_{\ell k}(w^o - w_{k,i-1}), \quad (9.140)$$

$$w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k}(r_{du,\ell} - R_{u,\ell}w_{k,i-1}). \quad (9.141)$$

Following similar reasoning to what we did before in the ATC case, we replace  $w^o$  in step (9.140) by  $w_{\ell,i-1}$  and replace  $w_{k,i-1}$  in (9.141) by  $\psi_{k,i-1}$ . We then introduce the same coefficients  $\{a_{\ell k}\}$  and arrive

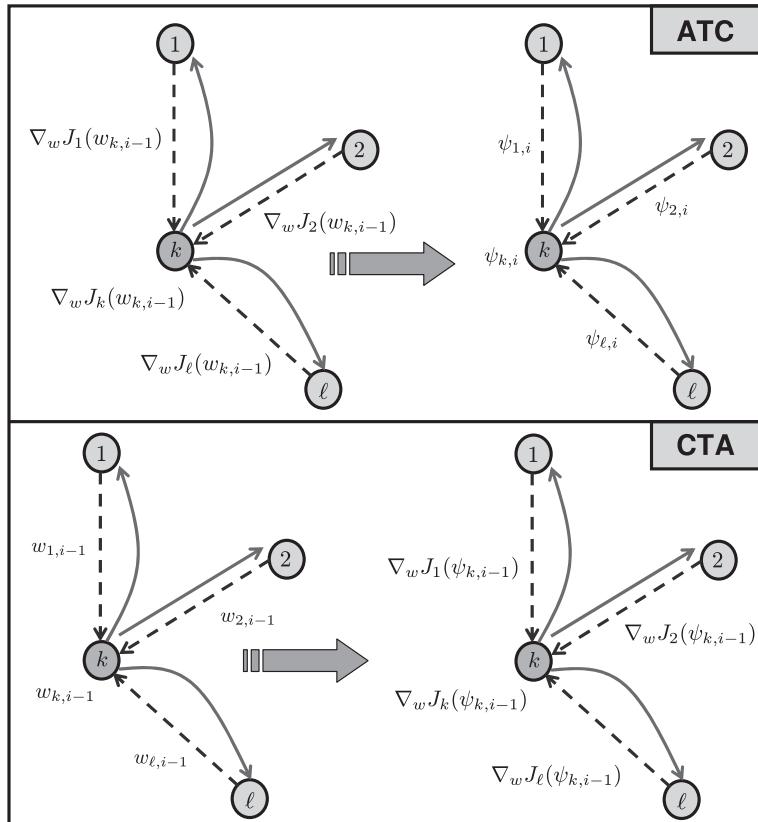

**FIGURE 9.11**

Illustration of the ATC and CTA strategies for a node  $k$  with three other neighbors  $\{1, 2, \ell\}$ . The updates involve two steps: information exchange followed by aggregation in ATC and aggregation followed by information exchange in CTA. The dotted arrows represent the data received from the neighbors of node  $k$ , and the solid arrows represent the data sent from node  $k$  to its neighbors.

at the following Combine-then-Adapt (CTA) strategy:

$$\begin{aligned}
 & \text{(CTA strategy)} \quad \psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\
 & \quad w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} \psi_{k,i-1}) ,
 \end{aligned} \tag{9.142}$$

where the nonnegative coefficients  $\{c_{\ell k}, a_{\ell k}\}$  satisfy the same conditions (9.106) and (9.133), namely,

$$C\mathbb{1} = \mathbb{1}, \quad A^T \mathbb{1} = \mathbb{1} \tag{9.143}$$

or, equivalently,

$$\begin{aligned} c_{\ell k} \geq 0, \quad & \sum_{k=1}^N c_{\ell k} = 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ a_{\ell k} \geq 0, \quad & \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad \text{for } k = 1, 2, \dots, N: \quad (9.144)$$

At every time instant  $i$ , the CTA strategy (9.142) also consists of two steps. The first step is an aggregation step where node  $k$  combines the existing estimates of its neighbors to obtain the intermediate estimate  $\psi_{k,i-1}$ . All other nodes in the network are simultaneously performing a similar step and aggregating the estimates of their neighbors. The second step in (9.142) is an information exchange step where node  $k$  receives from its neighbors their moments  $\{R_{du,\ell}, r_{du,\ell}\}$  and uses this information to update its intermediate estimate to  $w_{k,i}$ . Again, all other nodes in the network are simultaneously performing a similar information exchange step. The reason for the name Combine-then-Adapt (CTA) strategy is that the first step in (9.142) involves a combination step, while the second step will be shown to lead to an adaptive step. Hence, strategy (9.142) involves combination followed by adaptation or CTA for short. The reason for the qualification “diffusion” is that the combination step of (9.142) allows information to diffuse through the network in real time.

In the special case when  $C = I$ , so that no information exchange is performed but only the aggregation step, the CTA strategy (9.142) reduces to:

$$\boxed{\begin{aligned} (\text{CTA strategy without} \\ \text{information exchange}) \quad & \psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ & w_{k,i} = \psi_{k,i-1} + \mu_k (r_{du,k} - R_{u,k} \psi_{k,i-1}) \end{aligned}}, \quad (9.145)$$

where the second step relies solely on the information  $\{R_{u,k}, r_{du,k}\}$  that is available locally at node  $k$ . Again, the CTA strategy (9.142) can be rewritten in terms of the gradient vectors of the local costs  $\{J_\ell(w)\}$  as follows:

$$\boxed{\begin{aligned} (\text{CTA strategy}) \quad & \psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ & w_{k,i} = \psi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(\psi_{k,i-1})]^* \end{aligned}}. \quad (9.146)$$

The bottom part of Figure 9.11 illustrates the two steps involved in the CTA procedure for a situation where node  $k$  has three other neighbors labeled  $\{1, 2, \ell\}$ . In the first step, node  $k$  aggregates the estimates  $\{w_{1,i-1}, w_{2,i-1}, w_{\ell,i-1}\}$  from its neighbors, and subsequently performs information exchange by evaluating the gradient vectors of its neighbors at  $\psi_{k,i-1}$ .

### 3.09.3.5 Useful properties of diffusion strategies

Note that the structure of the ATC and CTA diffusion strategies (9.134) and (9.142) are fundamentally the same: the difference between the implementations lies in which variable we choose to correspond

**Table 9.2** Summary of Steepest-Descent Diffusion Strategies for the Distributed Optimization of Problems of the form (9.92), and their Specialization to the Case of Mean-Square-Error (MSE) Individual Cost Functions Given by (9.93)

Algorithm	Recursions	Equation
<b>ATC strategy</b> (general case)	$\psi_{k,i} = w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(w_{k,i-1})]^*$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$	(9.139)
<b>ATC strategy</b> (MSE costs)	$\psi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} w_{k,i-1})$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$	(9.134)
<b>CTA strategy</b> (general case)	$\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(\psi_{k,i-1})]^*$	(9.146)
<b>CTA strategy</b> (MSE costs)	$\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (r_{du,\ell} - R_{u,\ell} \psi_{k,i-1})$	(9.142)

to the updated weight estimate  $w_{k,i}$ . In the ATC case, we choose the result of the *combination* step to be  $w_{k,i}$ , whereas in the CTA case we choose the result of the *adaptation* step to be  $w_{k,i}$ .

For ease of reference, Table 9.2 lists the steepest-descent diffusion algorithms derived in the previous sections. The derivation of the ATC and CTA strategies (9.134) and (9.142) followed the approach proposed in [18,28]. CTA estimation schemes were first proposed in the works [29–33], and later extended in [18,28,34,35]. The earlier versions of CTA in [29–31] used the choice  $C = I$ . This form of the algorithm with  $C = I$ , and with the additional constraint that the step-sizes  $\mu_k$  should be time-dependent and decay towards zero as time progresses, was later applied by [36,37] to solve distributed optimization problems that require all nodes to reach consensus or agreement. Likewise, special cases of the ATC estimation scheme (9.134), involving an information exchange step followed by an aggregation step, first appeared in the work [38] on diffusion least-squares schemes and subsequently in the works [18,34,35,39–41] on distributed mean-square-error and state-space estimation methods. A special case of the ATC strategy (9.134) corresponding to the choice  $C = I$  with decaying step-sizes was adopted in [42] to ensure convergence towards a consensus state. Diffusion strategies of the form (9.134) and (9.142) (or, equivalently, (9.139) and (9.146)) are general in several respects:

1. These strategies do not only diffuse the local weight estimates, but they can also diffuse the local gradient vectors. In other words, two sets of combination coefficients  $\{a_{\ell k}, c_{\ell k}\}$  can be used.
2. In the derivation that led to the diffusion strategies, the combination matrices  $C$  and  $A$  are only required to be right-stochastic (for  $C$ ) and left-stochastic (for  $A$ ). In comparison, it is common in consensus-type strategies to require the corresponding combination matrix  $A$  to be doubly stochastic (i.e., its rows and columns should add up to one)—see, e.g., Appendix E and [36, 43–45].
3. As the analysis in Section 3.09.6 will reveal, ATC and CTA strategies do *not* force nodes to converge to an agreement about the desired parameter vector  $w^o$ , as is common in consensus-type strategies (see Appendix E and [36, 46–52]). Forcing nodes to reach agreement on  $w^o$  ends up limiting the adaptation and learning abilities of these nodes, as well as their ability to react to information in real-time. Nodes in diffusion networks enjoy more flexibility in the learning process, which allows their individual estimates,  $\{w_{k,i}\}$ , to tend to values that lie within a reasonable mean-square-deviation (MSD) level from the optimal solution,  $w^o$ . Multi-agent systems in nature behave in this manner; they do not require exact agreement among their agents (see, e.g., [8–10]).
4. The step-size parameters  $\{\mu_k\}$  are not required to depend on the time index  $i$  and are not required to vanish as  $i \rightarrow \infty$  (as is common in many works on distributed optimization, e.g., [22, 23, 36, 53]). Instead, the step-sizes can assume constant values, which is a critical property to endow networks with continuous adaptation and learning abilities. An important contribution in the study of diffusion strategies is to show that distributed optimization is still possible even for constant step-sizes, in addition to the ability to perform adaptation, learning, and tracking. Sections 3.09.5 and 3.09.6 highlight the convergence properties of the diffusion strategies—see also [1–3] for results pertaining to more general cost functions.
5. Even the combination weights  $\{a_{\ell k}, c_{\ell k}\}$  can be adapted, as we shall discuss later in Section 3.09.8.3. In this way, diffusion strategies allow multiple layers of adaptation: the nodes perform adaptive processing, the combination weights can be adapted, and even the topology can be adapted especially for mobile networks [8].

### 3.09.4 Adaptive diffusion strategies

The distributed ATC and CTA steepest-descent strategies (9.134) and (9.142) for determining the  $w^o$  that solves (9.92) and (9.93) require knowledge of the statistical information  $\{R_{u,k}, r_{du,k}\}$ . These moments are needed in order to be able to evaluate the gradient vectors that appear in (9.134) and (9.142), namely, the terms:

$$-[\nabla_w J_\ell(w_{k,i-1})]^* = (r_{du,\ell} - R_{u,\ell} w_{k,i-1}), \quad (9.147)$$

$$-[\nabla_w J_\ell(\psi_{k,i-1})]^* = (r_{du,\ell} - R_{u,\ell} \psi_{k,i-1}), \quad (9.148)$$

for all  $\ell \in \mathcal{N}_k$ . However, the moments  $\{R_{u,\ell}, r_{du,\ell}\}$  are often not available beforehand, which means that the true gradient vectors are generally not available. Instead, the agents have access to observations  $\{d_k(i), u_{k,i}\}$  of the random processes  $\{d_k(i), u_{k,i}\}$ . There are many ways by which the true gradient vectors can be approximated by using these observations. Recall that, by definition,

$$R_{u,\ell} \triangleq \mathbb{E} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i}, \quad r_{du,\ell} \triangleq \mathbb{E} \mathbf{d}_\ell(i) \mathbf{u}_{\ell,i}^*. \quad (9.149)$$

One common stochastic approximation method is to drop the expectation operator from the definitions of  $\{R_{u,\ell}, r_{du,\ell}\}$  and to use the following instantaneous approximations instead [4–7]:

$$R_{u,\ell} \approx u_{\ell,i}^* u_{\ell,i}, \quad r_{du,\ell} \approx d_\ell(i) u_{\ell,i}^*. \quad (9.150)$$

In this case, the approximate gradient vectors become:

$$(r_{du,\ell} - R_{u,\ell} w_{k,i-1}) \approx u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}], \quad (9.151)$$

$$(r_{du,\ell} - R_{u,\ell} \psi_{k,i-1}) \approx u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \psi_{k,i-1}]. \quad (9.152)$$

Substituting into the ATC and CTA steepest-descent strategies (9.134) and (9.142), we arrive at the following adaptive implementations of the diffusion strategies for  $i \geq 0$ :

$$\begin{aligned} \psi_{k,i} &= w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}] \\ (\text{adaptive ATC strategy}) \quad w_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{aligned} \quad (9.153)$$

and

$$\begin{aligned} \psi_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ (\text{adaptive CTA strategy}) \quad w_{k,i} &= \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \psi_{k,i-1}] \end{aligned} \quad (9.154)$$

where the coefficients  $\{a_{\ell k}, c_{\ell k}\}$  are chosen to satisfy:

$$\begin{aligned} &\text{for } k = 1, 2, \dots, N : \\ c_{\ell k} \geq 0, \quad \sum_{k=1}^N c_{\ell k} &= 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} &= 1, \quad a_{\ell k} = 0 \text{ if } k \notin \mathcal{N}_k. \end{aligned} \quad (9.155)$$

The adaptive implementations usually start from the initial conditions  $w_{\ell,-1} = 0$  for all  $\ell$ , or from some other convenient initial values. Clearly, in view of the approximations (9.151) and (9.152), the successive iterates  $\{w_{k,i}, \psi_{k,i}, \psi_{k,i-1}\}$  that are generated by the above adaptive implementations are different from the iterates that result from the steepest-descent implementations (9.134) and (9.142). Nevertheless, we shall continue to use the same notation for these variables for ease of reference. One key advantage of the adaptive implementations (9.153) and (9.154) is that they enable the agents to react to changes in the underlying statistical information  $\{r_{du,\ell}, R_{u,\ell}\}$  and to changes in  $w^o$ . This is because these changes end up being reflected in the data realizations  $\{d_k(i), u_{k,i}\}$ . Therefore, adaptive implementations have an innate tracking and learning ability that is of paramount significance in practice.

We say that the stochastic gradient approximations (9.151) and (9.152) introduce gradient noise into each step of the recursive updates (9.153) and (9.154). This is because the updates (9.153) and (9.154) can be interpreted as corresponding to the following forms:

$$\begin{aligned} \psi_{k,i} &= w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\widehat{\nabla_w J_\ell}(w_{k,i-1})]^* \\ \text{(adaptive ATC strategy)} \quad w_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{aligned} \quad (9.156)$$

and

$$\begin{aligned} \psi_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ \text{(adaptive CTA strategy)} \quad w_{k,i} &= \psi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\widehat{\nabla_w J_\ell}(\psi_{k,i-1})]^* \end{aligned} , \quad (9.157)$$

where the true gradient vectors,  $\{\nabla_w J_\ell(\cdot)\}$ , have been replaced by approximations,  $\{\widehat{\nabla_w J_\ell}(\cdot)\}$ —compare with (9.139) and (9.146). The significance of the alternative forms (9.156) and (9.157) is that they are applicable to optimization problems involving more general local costs  $J_\ell(w)$  that are not necessarily quadratic, as detailed in [1–3]; see also Section 3.09.10.4. In the next section, we examine how gradient noise affects the performance of the diffusion strategies and how close the successive estimates  $\{w_{k,i}\}$  get to the desired optimal solution  $w^o$ . Table 9.3 lists several of the adaptive diffusion algorithms derived in this section.

The operation of the adaptive diffusion strategies is similar to the operation of the steepest-descent diffusion strategies of the previous section. Thus, note that at every time instant  $i$ , the ATC strategy (9.153) performs two steps; as illustrated in Figure 9.12. The first step is an *information exchange* step where node  $k$  receives from its neighbors their information  $\{d_\ell(i), u_{\ell,i}\}$ . Node  $k$  combines this information and uses it to update its existing estimate  $w_{k,i-1}$  to an intermediate value  $\psi_{k,i}$ . All other nodes in the network are performing a similar step and updating their existing estimates  $\{w_{\ell,i-1}\}$  into intermediate estimates  $\{\psi_{\ell,i}\}$  by using information from their neighbors. The second step in (9.153) is an *aggregation* or consultation step where node  $k$  combines the intermediate estimates  $\{\psi_{\ell,i}\}$  of its neighbors to obtain its updated estimate  $w_{k,i}$ . Again, all other nodes in the network are simultaneously performing a similar step. In the special case when  $C = I$ , so that no information exchange is performed but only the aggregation step, the ATC strategy (9.153) reduces to:

$$\begin{aligned} \text{(adaptive ATC strategy} \quad \psi_{k,i} &= w_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} w_{k,i-1}] \\ \text{without information exchange)} \quad w_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{aligned} . \quad (9.158)$$

Likewise, at every time instant  $i$ , the CTA strategy (9.154) also consists of two steps—see Figure 9.13. The first step is an aggregation step where node  $k$  combines the existing estimates of its neighbors to obtain the intermediate estimate  $\psi_{k,i-1}$ . All other nodes in the network are simultaneously performing a

**Table 9.3** Summary of Adaptive Diffusion Strategies for the Distributed Optimization of Problems of the form (9.92), and their Specialization to the Case of Mean-Square-Error (MSE) Individual Cost Functions Given by (9.93). These Adaptive Solutions Rely on Stochastic Approximations

Algorithm	Recursions	Equation
<b>Adaptive ATC strategy</b> (general case)	$\psi_{k,i} = w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\widehat{\nabla_w J_\ell}(w_{k,i-1})]^*$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$	(9.156)
<b>Adaptive ATC strategy</b> (MSE costs)	$\psi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}]$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$	(9.153)
<b>Adaptive ATC strategy</b> (MSE costs) (no information exchange)	$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} w_{k,i-1}]$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$	(9.158)
<b>Adaptive CTA strategy</b> (general case)	$\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\widehat{\nabla_w J_\ell}(\psi_{k,i-1})]^*$	(9.157)
<b>Adaptive CTA strategy</b> (MSE costs)	$\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \psi_{k,i-1}]$	(9.154)
<b>Adaptive CTA strategy</b> (MSE costs) (no information exchange)	$\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} \psi_{k,i-1}]$	(9.159)

similar step and aggregating the estimates of their neighbors. The second step in (9.154) is an information exchange step where node  $k$  receives from its neighbors their information  $\{d_\ell(i), u_{\ell,i}\}$  and uses this information to update its intermediate estimate to  $w_{k,i}$ . Again, all other nodes in the network are

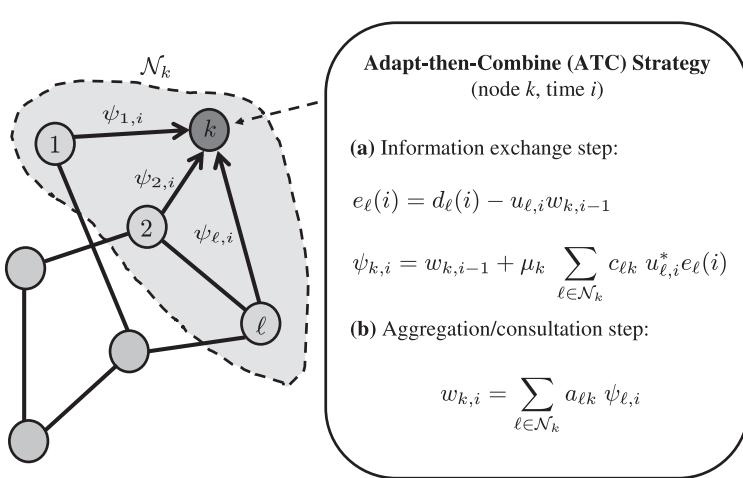
**FIGURE 9.12**

Illustration of the adaptive ATC strategy, which involves two steps: information exchange followed by aggregation.

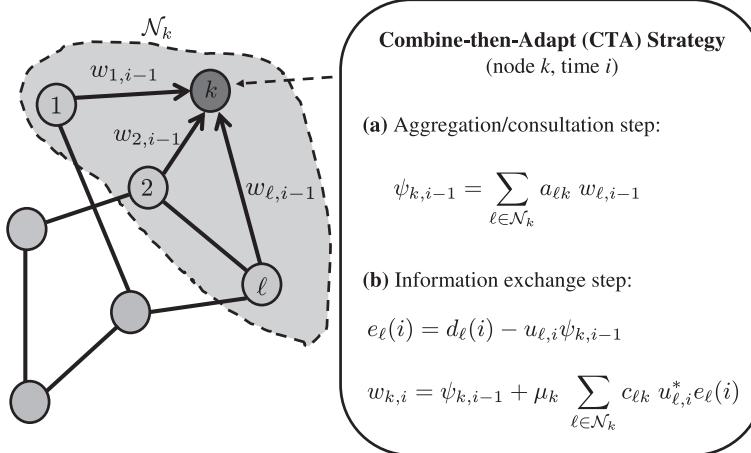
**FIGURE 9.13**

Illustration of the adaptive CTA strategy, which involves two steps: aggregation followed by information exchange.

simultaneously performing a similar information exchange step. In the special case when  $C = I$ , so that no information exchange is performed but only the aggregation step, the CTA strategy (9.154) reduces to:

$$\begin{array}{l} \text{(adaptive CTA strategy} \\ \text{without information exchange)} \end{array} \boxed{\begin{aligned} \psi_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ w_{k,i} &= \psi_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} \psi_{k,i-1}] \end{aligned}} . \quad (9.159)$$

We further note that the adaptive ATC and CTA strategies (9.153) and (9.154) reduce to the non-cooperative adaptive solution (9.22) and (9.23), where each node  $k$  runs its own individual LMS filter, when the coefficients  $\{a_{\ell k}, c_{\ell k}\}$  are selected as

$$a_{\ell k} = \delta_{\ell k} = c_{\ell k} \quad (\text{non-cooperative case}), \quad (9.160)$$

where  $\delta_{\ell k}$  denotes the Kronecker delta function:

$$\delta_{\ell k} \triangleq \begin{cases} 1, & \ell = k, \\ 0, & \text{otherwise.} \end{cases} \quad (9.161)$$

In terms of the combination matrices  $A$  and  $C$ , this situation corresponds to setting

$$\boxed{A = I_N = C} \quad (\text{non-cooperative case}). \quad (9.162)$$

### 3.09.5 Performance of steepest-descent diffusion strategies

Before studying in some detail the mean-square performance of the adaptive diffusion implementations (9.153) and (9.154), and the influence of gradient noise, we examine first the convergence behavior of the steepest-descent diffusion strategies (9.134) and (9.142), which employ the true gradient vectors. Doing so will help introduce the necessary notation and highlight some features of the analysis in preparation for the more challenging treatment of the adaptive strategies in Section 3.09.6.

#### 3.09.5.1 General diffusion model

Rather than study the performance of the ATC and CTA steepest-descent strategies (9.134) and (9.142) separately, it is useful to introduce a more general description that includes the ATC and CTA recursions as special cases. Thus, consider a distributed steepest-descent diffusion implementation of the following general form for  $i \geq 0$ :

$$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} w_{\ell,i-1}, \quad (9.163)$$

$$\psi_{k,i} = \phi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [r_{du,\ell} - R_{u,\ell} \phi_{k,i-1}], \quad (9.164)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \psi_{\ell,i}, \quad (9.165)$$

where the scalars  $\{a_{1,\ell k}, c_{\ell k}, a_{2,\ell k}\}$  denote three sets of nonnegative real coefficients corresponding to the  $(\ell, k)$  entries of  $N \times N$  combination matrices  $\{A_1, C, A_2\}$ , respectively. These matrices are assumed to satisfy the conditions:

$$A_1^T \mathbb{1} = \mathbb{1}, \quad C \mathbb{1} = \mathbb{1}, \quad A_2^T \mathbb{1} = \mathbb{1}, \quad (9.166)$$

so that  $\{A_1, A_2\}$  are left stochastic and  $C$  is right-stochastic, i.e.,

$$\begin{aligned} & \text{for } k = 1, 2, \dots, N : \\ & c_{\ell k} \geq 0, \quad \sum_{k=1}^N c_{\ell k} = 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ & a_{1,\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{1,\ell k} = 1, \quad a_{1,\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ & a_{2,\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{2,\ell k} = 1, \quad a_{2,\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad (9.167)$$

As indicated in Table 9.4, different choices for  $\{A_1, C, A_2\}$  correspond to different cooperation modes. For example, the choice  $A_1 = I_N$  and  $A_2 = A$  corresponds to the ATC implementation (9.134), while the choice  $A_1 = A$  and  $A_2 = I_N$  corresponds to the CTA implementation (9.142). Likewise, the choice  $C = I_N$  corresponds to the case in which the nodes only share weight estimates and the distributed diffusion recursions (9.163) to (9.165) become:

$$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} w_{\ell,i-1}, \quad (9.168)$$

$$\psi_{k,i} = \phi_{k,i-1} + \mu_k (r_{du,k} - R_{u,k} \phi_{k,i-1}), \quad (9.169)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \psi_{\ell,i}. \quad (9.170)$$

Furthermore, the choice  $A_1 = A_2 = C = I_N$  corresponds to the non-cooperative mode of operation, in which case the recursions reduce to the classical (stand-alone) steepest-descent recursion [4–7], where each node minimizes individually its own quadratic cost  $J_k(w)$ , defined earlier in (9.97):

$$w_{k,i} = w_{k,i-1} + \mu_k [r_{du,k} - R_{u,k} w_{k,i-1}], \quad i \geq 0. \quad (9.171)$$

**Table 9.4** Different Choices for the Combination Matrices  $\{A_1, A_2, C\}$  in (9.163)–(9.165) Correspond to Different Cooperation Strategies

<b><math>A_1</math></b>	<b><math>A_2</math></b>	<b><math>C</math></b>	<b>Cooperation Mode</b>
$I_N$	$A$	$C$	ATC strategy (9.134)
$I_N$	$A$	$I_N$	ATC strategy (9.137) without information exchange
$A$	$I_N$	$C$	CTA strategy (9.142)
$A$	$I_N$	$I_N$	CTA strategy (9.145) without information exchange
$I_N$	$I_N$	$I_N$	non-cooperative steepest-descent (9.171)

### 3.09.5.2 Error recursions

Our objective is to examine whether, and how fast, the weight estimates  $\{w_{k,i}\}$  from the distributed implementation (9.163)–(9.165) converge towards the solution  $w^o$  of (9.92) and (9.93). To do so, we introduce the  $M \times 1$  error vectors:

$$\tilde{\phi}_{k,i} \triangleq w^o - \phi_{k,i}, \quad (9.172)$$

$$\tilde{\psi}_{k,i} \triangleq w^o - \psi_{k,i}, \quad (9.173)$$

$$\tilde{w}_{k,i} \triangleq w^o - w_{k,i}. \quad (9.174)$$

Each of these error vectors measures the residual relative to the desired minimizer  $w^o$ . Now recall from (9.100) that

$$r_{du,k} = R_{u,k} w^o. \quad (9.175)$$

Then, subtracting  $w^o$  from both sides of relations in (9.163)–(9.165) we get

$$\tilde{\phi}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \tilde{w}_{\ell,i-1}, \quad (9.176)$$

$$\tilde{\psi}_{k,i} = \left( I_M - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u,\ell} \right) \tilde{\phi}_{k,i-1}, \quad (9.177)$$

$$\tilde{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \tilde{\psi}_{\ell,i}. \quad (9.178)$$

We can describe these relations more compactly by collecting the information from across the network into block vectors and matrices. We collect the error vectors from across all nodes into the following  $N \times 1$  *block* vectors, whose individual entries are of size  $M \times 1$  each:

$$\tilde{\psi}_i \triangleq \begin{bmatrix} \tilde{\psi}_{1,i} \\ \tilde{\psi}_{2,i} \\ \vdots \\ \tilde{\psi}_{N,i} \end{bmatrix}, \quad \tilde{\phi}_i \triangleq \begin{bmatrix} \tilde{\phi}_{1,i} \\ \tilde{\phi}_{2,i} \\ \vdots \\ \tilde{\phi}_{N,i} \end{bmatrix}, \quad \tilde{w}_i \triangleq \begin{bmatrix} \tilde{w}_{1,i} \\ \tilde{w}_{2,i} \\ \vdots \\ \tilde{w}_{N,i} \end{bmatrix}. \quad (9.179)$$

The block quantities  $\{\tilde{\psi}_i, \tilde{\phi}_i, \tilde{w}_i\}$  represent the state of the errors across the network at time  $i$ . Likewise, we introduce the following  $N \times N$  *block* diagonal matrices, whose individual entries are of size  $M \times M$  each:

$$\mathcal{M} \triangleq \text{diag}\{\mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M\}, \quad (9.180)$$

$$\mathcal{R} \triangleq \text{diag} \left\{ \sum_{\ell \in \mathcal{N}_1} c_{\ell 1} R_{u,\ell}, \sum_{\ell \in \mathcal{N}_2} c_{\ell 2} R_{u,\ell}, \dots, \sum_{\ell \in \mathcal{N}_N} c_{\ell N} R_{u,\ell} \right\}. \quad (9.181)$$

Each block diagonal entry of  $\mathcal{R}$ , say, the  $k$ th entry, contains the combination of the covariance matrices in the neighborhood of node  $k$ . We can simplify the notation by denoting these neighborhood combinations as follows:

$$R_k \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u,\ell}, \quad (9.182)$$

so that  $\mathcal{R}$  becomes

$$\mathcal{R} \triangleq \text{diag}\{R_1, R_2, \dots, R_N\} \quad (\text{when } C \neq I). \quad (9.183)$$

In the special case when  $C = I_N$ , the matrix  $\mathcal{R}$  reduces to

$$\mathcal{R}_u = \text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\} \quad (\text{when } C = I) \quad (9.184)$$

with the individual covariance matrices appearing on its diagonal; we denote  $\mathcal{R}$  by  $\mathcal{R}_u$  in this special case. We further introduce the Kronecker products

$$\mathcal{A}_1 \triangleq A_1 \otimes I_M, \quad \mathcal{A}_2 \triangleq A_2 \otimes I_M. \quad (9.185)$$

The matrix  $\mathcal{A}_1$  is an  $N \times N$  *block* matrix whose  $(\ell, k)$  block is equal to  $a_{1,\ell k} I_M$ . Likewise, for  $\mathcal{A}_2$ . In other words, the Kronecker transformation defined above simply replaces the matrices  $\{A_1, A_2\}$  by block matrices  $\{\mathcal{A}_1, \mathcal{A}_2\}$  where each entry  $\{a_{1,\ell k}, a_{2,\ell k}\}$  in the original matrices is replaced by the diagonal matrices  $\{a_{1,\ell k} I_M, a_{2,\ell k} I_M\}$ . For ease of reference, Table 9.5 lists the various symbols that have been defined so far, and others that will be defined in the sequel.

Returning to (9.176)–(9.178), we conclude that the following relations hold for the block quantities:

$$\tilde{\phi}_{i-1} = \mathcal{A}_1^T \tilde{w}_{i-1}, \quad (9.186)$$

$$\tilde{\psi}_i = (I_{NM} - \mathcal{M}\mathcal{R})\tilde{\phi}_{i-1}, \quad (9.187)$$

$$\tilde{w}_i = \mathcal{A}_2^T \tilde{\psi}_i, \quad (9.188)$$

so that the network weight error vector,  $\tilde{w}_i$ , ends up evolving according to the following dynamics:

$$\tilde{w}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}) \mathcal{A}_1^T \tilde{w}_{i-1}, \quad i \geq 0 \quad (\text{diffusion strategy}). \quad (9.189)$$

For comparison purposes, if each node in the network minimizes its own cost function,  $J_k(w)$ , separately from the other nodes and uses the non-cooperative steepest-descent strategy (9.171), then the weight error vector across all  $N$  nodes would evolve according to the following alternative dynamics:

$$\tilde{w}_i = (I_{NM} - \mathcal{M}\mathcal{R}_u) \tilde{w}_{i-1}, \quad i \geq 0 \quad (\text{non-cooperative strategy}), \quad (9.190)$$

where the matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$  do not appear, and  $\mathcal{R}$  is replaced by  $\mathcal{R}_u$  from (9.184). This recursion is a special case of (9.189) when  $A_1 = A_2 = C = I_N$ .

### 3.09.5.3 Convergence behavior

Note from (9.189) that the evolution of the weight error vector involves block vectors and block matrices; this will be characteristic of the distributed implementations that we consider in this chapter. To examine the stability and convergence properties of recursions that involve such block quantities, it becomes useful to rely on a certain block vector norm. In Appendix D, we describe a so-called *block maximum norm* and establish some of its useful properties. The results of the appendix will be used extensively in our exposition. It is therefore advisable for the reader to review the properties stated in the appendix at this stage.

Using the result of Lemma D.6, we can establish the following useful statement about the convergence of the steepest-descent diffusion strategy (9.163)–(9.165). The result establishes that all nodes end up converging to the optimal solution  $w^o$  if the nodes employ positive step-sizes  $\mu_k$  that are small enough; the lemma provides a sufficient bound on the  $\{\mu_k\}$ .

**Theorem 9.5.1 (Convergence to Optimal Solution).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a right stochastic matrix  $C$  and left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) or (9.167); these matrices define the network topology and how information is shared over neighborhoods. Assume each node in the network runs the (distributed) steepest-descent diffusion algorithm (9.163)–(9.165). Then, all estimates  $\{w_{k,i}\}$  across*

**Table 9.5** Definitions of Network Variables Used Throughout the Analysis

Variable	Equation
$A_1 = A_1 \otimes I_M$	(9.185)
$A_2 = A_2 \otimes I_M$	(9.185)
$C = C \otimes I_M$	(9.245)
$R_k = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u,\ell}$	(9.182)
$\mathcal{R} = \text{diag}\{R_1, R_2, \dots, R_N\}$	(9.183)
$\mathcal{R}_u = \text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\}$	(9.241)
$\mathcal{R}_v = \text{diag}\{\sigma_{v,1}^2, \sigma_{v,2}^2, \dots, \sigma_{v,N}^2\}$	(9.319)
$\mathcal{M} = \text{diag}\{\mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M\}$	(9.180)
$\mathcal{S} = \text{diag}\{\sigma_{v,1}^2 R_{u,1}, \sigma_{v,2}^2 R_{u,2}, \dots, \sigma_{v,N}^2 R_{u,N}\}$	(9.241)
$\mathcal{G} = A_2^T \mathcal{M} C^T$	(9.263)
$\mathcal{B} = A_2^T (I_{NM} - \mathcal{M} \mathcal{R}) A_1^T$	(9.264)
$\mathcal{Y} = \mathcal{G} \mathcal{S} \mathcal{G}^T$	(9.280)
$\mathcal{F} \approx \mathcal{B}^T \otimes \mathcal{B}^*$	(9.277)
$\mathcal{J}_k = \text{diag}\{0_M, \dots, 0_M, I_M, 0_M, \dots, 0_M\}$	(9.294)
$\mathcal{T}_k = \text{diag}\{0_M, \dots, 0_M, R_{u,k}, 0_M, \dots, 0_M\}$	(9.298)

the network converge to the optimal solution  $w^o$  if the positive step-size parameters  $\{\mu_k\}$  satisfy

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}, \quad (9.191)$$

where the neighborhood covariance matrix  $R_k$  is defined by (9.182).

**Proof.** The weight error vector  $\tilde{w}_i$  converges to zero if, and only if, the coefficient matrix  $\mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R})\mathcal{A}_1^T$  in (9.189) is a stable matrix (meaning that all its eigenvalues lie strictly inside the unit disc). From property (9.605) established in Appendix D, we know that  $\mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R})\mathcal{A}_1^T$  is stable if the block diagonal matrix  $(I_{NM} - \mathcal{M}\mathcal{R})$  is stable. It is now straightforward to verify that condition (9.191) ensures the stability of  $(I_{NM} - \mathcal{M}\mathcal{R})$ . It follows that

$$\tilde{w}_i \rightarrow 0 \quad \text{as } i \rightarrow \infty. \quad (9.192)$$

□

Observe that the stability condition (9.191) does not depend on the specific combination matrices  $A_1$  and  $A_2$ . Thus, as long as these matrices are chosen to be left-stochastic, the weight-error vectors will converge to zero under condition (9.191) no matter what  $\{A_1, A_2\}$  are. Only the combination matrix  $C$  influences the condition on the step-size through the neighborhood covariance matrices  $\{R_k\}$ . Observe further that the statement of the lemma does not require the network to be connected. Moreover, when  $C = I$ , in which case the nodes only share weight estimates and do not share the neighborhood moments  $\{r_{du,\ell}, R_{u,\ell}\}$ , as in (9.168)–(9.170), condition (9.191) becomes

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}} \quad (\text{cooperation with } C = I), \quad (9.193)$$

in terms of the actual covariance matrices  $\{R_{u,k}\}$ . Conditions (9.191) and (9.193) are reminiscent of a classical result for stand-alone steepest-descent algorithms, as in the non-cooperative case (9.171), where it is known that the estimate by each individual node in this case will converge to  $w^o$  if, and only if, its positive step-size satisfies

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}} \quad (\text{non-cooperative case (9.171) with } A_1 = A_2 = C = I_N). \quad (9.194)$$

This is the same condition as (9.193) for the case  $C = I$ .

The following statement provides a *bi-directional* statement that ensures convergence of the (distributed) steepest-descent diffusion strategy (9.163)–(9.165) for *any* choice of left-stochastic combination matrices  $A_1$  and  $A_2$ .

**Theorem 9.5.2 (Convergence for Arbitrary Combination Matrices).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a right stochastic matrix  $C$  satisfying (9.166). Then, the estimates  $\{w_{k,i}\}$  generated by (9.163)–(9.165) converge to  $w^o$ , for all choices of left-stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) if, and only if,*

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}. \quad (9.195)$$

**Proof.** The result follows from property (b) of Corollary D.1, which is established in Appendix D.  $\square$

More importantly, we can verify that under fairly general conditions, employing the steepest-descent diffusion strategy (9.163)–(9.165) enhances the convergence rate of the error vector towards zero relative to the non-cooperative strategy (9.171). The next three results establish this fact when  $C$  is a doubly stochastic matrix, i.e., it has nonnegative entries and satisfies

$$C\mathbf{1} = \mathbf{1}, \quad C^T\mathbf{1} = \mathbf{1} \quad (9.196)$$

with both its rows and columns adding up to one. Compared to the earlier right-stochastic condition on  $C$  in (9.105), we are now requiring

$$\sum_{\ell \in \mathcal{N}_k} c_{k\ell} = 1, \quad \sum_{\ell \in \mathcal{N}_k} c_{\ell k} = 1. \quad (9.197)$$

For example, these conditions are satisfied when  $C$  is right stochastic and symmetric. They are also automatically satisfied for  $C = I$ , when only weight estimates are shared as in (9.168)–(9.170); this latter case covers the ATC and CTA diffusion strategies (9.137) and (9.145), which do not involve information exchange.

**Theorem 9.5.3 (Convergence Rate is Enhanced: Uniform Step-Sizes).** Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a doubly stochastic matrix  $C$  satisfying (9.196) and left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166). Consider two modes of operation. In one mode, each node in the network runs the (distributed) steepest-descent diffusion algorithm (9.163)–(9.165). In the second mode, each node operates individually and runs the non-cooperative steepest-descent algorithm (9.171). In both cases, the positive step-sizes used by all nodes are assumed to be the same, say,  $\mu_k = \mu$  for all  $k$ , and the value of  $\mu$  is chosen to satisfy the required stability conditions (9.191) and (9.194), which are met by selecting

$$\mu < \min_{1 \leq k \leq N} \left\{ \frac{2}{\lambda_{\max}(R_{u,k})} \right\}. \quad (9.198)$$

It then holds that the magnitude of the error vector,  $\|\tilde{w}_i\|$ , in the diffusion case decays to zero more rapidly than in the non-cooperative case. In other words, diffusion cooperation enhances convergence rate.

**Proof.** Let us first establish that any positive step-size  $\mu$  satisfying (9.198) will satisfy both stability conditions (9.191) and (9.194). It is obvious that (9.194) is satisfied. We verify that (9.191) is also satisfied when  $C$  is doubly stochastic. In this case, each neighborhood covariance matrix,  $R_k$ , becomes a convex combination of individual covariance matrices  $\{R_{u,\ell}\}$ , i.e.,

$$R_k = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u,\ell},$$

where now

$$\sum_{\ell \in \mathcal{N}_k} c_{\ell k} = 1 \quad (\text{when } C \text{ is doubly stochastic}).$$

To proceed, we recall that the spectral norm (maximum singular value) of any matrix  $X$  is a convex function of  $X$  [54]. Moreover, for Hermitian matrices  $X$ , their spectral norms coincide with their spectral radii (largest eigenvalue magnitude). Then, Jensen's inequality [54] states that for any convex function  $f(\cdot)$  it holds that

$$f\left(\sum_m \theta_m X_m\right) \leq \sum_m \theta_m f(X_m)$$

for Hermitian matrices  $X_m$  and nonnegative scalars  $\theta_m$  that satisfy

$$\sum_m \theta_m = 1.$$

Choosing  $f(\cdot)$  as the spectral radius function, and applying it to the definition of  $R_k$  above, we get

$$\begin{aligned} \rho(R_k) &= \rho\left(\sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u,\ell}\right) \\ &\leq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \cdot \rho(R_{u,\ell}) \\ &\leq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \cdot \left[ \max_{1 \leq \ell \leq N} \rho(R_{u,\ell}) \right] \\ &= \max_{1 \leq \ell \leq N} \rho(R_{u,\ell}). \end{aligned}$$

In other words,

$$\lambda_{\max}(R_k) \leq \max_{1 \leq k \leq N} \{\lambda_{\max}(R_{u,k})\}.$$

It then follows from (9.198) that

$$\mu < \frac{2}{\lambda_{\max}(R_k)}, \quad \text{for all } k = 1, 2, \dots, N,$$

so that (9.191) is satisfied as well.

Let us now examine the convergence rate. To begin with, we note that the matrix  $(I_{NM} - \mathcal{MR})$  that appears in the weight-error recursion (9.189) is block diagonal:

$$(I_{NM} - \mathcal{MR}) = \text{diag}\{(I_M - \mu R_1), (I_M - \mu R_2), \dots, (I_M - \mu R_N)\}$$

and each individual block entry,  $(I_M - \mu R_k)$ , is a stable matrix since  $\mu$  satisfies (9.191). Moreover, each of these entries can be written as

$$I_M - \mu R_k = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (I_M - \mu R_{u,\ell}),$$

which expresses  $(I_M - \mu R_k)$  as a convex combination of stable terms  $(I_M - \mu R_{u,\ell})$ . Applying Jensen's inequality again we get

$$\rho \left( \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (I_M - \mu R_{u,\ell}) \right) \leq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \rho(I_M - \mu R_{u,\ell}).$$

Now, we know from (9.189) that the rate of decay of  $\tilde{w}_i$  to zero in the diffusion case is determined by the spectral radius of the coefficient matrix  $\mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T$ . Likewise, we know from (9.190) that the rate of decay of  $\tilde{w}_i$  to zero in the non-cooperative case is determined by the spectral radius of the coefficient matrix  $(I_{NM} - \mathcal{MR}_u)$ . Then, note that

$$\begin{aligned} \rho \left( \mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T \right) &\stackrel{(9.605)}{\leq} \rho(I_{NM} - \mathcal{MR}) \\ &= \max_{1 \leq k \leq N} \rho(I_M - \mu R_k) \\ &= \max_{1 \leq k \leq N} \rho \left( \sum_{\ell \in \mathcal{N}_k} c_{\ell k} (I_M - \mu R_{u,\ell}) \right) \\ &\leq \max_{1 \leq k \leq N} \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \rho(I_M - \mu R_{u,\ell}) \\ &\leq \max_{1 \leq k \leq N} \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \left( \max_{1 \leq \ell \leq N} \rho(I_M - \mu R_{u,\ell}) \right) \\ &= \max_{1 \leq k \leq N} \left\{ \left( \max_{1 \leq \ell \leq N} \rho(I_M - \mu R_{u,\ell}) \right) \cdot \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \right\} \\ &= \max_{1 \leq k \leq N} \left( \max_{1 \leq \ell \leq N} \rho(I_M - \mu R_{u,\ell}) \right) \\ &= \max_{1 \leq \ell \leq N} \rho(I_M - \mu R_{u,\ell}) \\ &= \rho(I_{NM} - \mathcal{MR}_u). \end{aligned}$$

Therefore, the spectral radius of  $\mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T$  is at most as large as the largest individual spectral radius in the non-cooperative case.  $\square$

The argument can be modified to handle different step-sizes across the nodes if we assume uniform covariance data across the network, as stated below.

**Theorem 9.5.4 (Convergence Rate is Enhanced: Uniform Covariance Data).** *Consider the same setting of Theorem 9.5.3. Assume the covariance data are uniform across all nodes, say,  $R_{u,k} = R_u$  is independent of  $k$ . Assume further that the nodes in both modes of operation employ steps-sizes  $\mu_k$  that are chosen to satisfy the required stability conditions (9.191) and (9.194), which in this case are met by:*

$$\mu_k < \frac{2}{\lambda_{\max}(R_u)}, \quad k = 1, 2, \dots, N. \quad (9.199)$$

*It then holds that the magnitude of the error vector,  $\|\tilde{w}_i\|$ , in the diffusion case decays to zero more rapidly than in the non-cooperative case. In other words, diffusion enhances convergence rate.*

**Proof.** Since  $R_{u,\ell} = R_u$  for all  $\ell$  and  $C$  is doubly stochastic, we get  $R_k = R_u$  and  $I_{NM} - \mathcal{MR} = I_{NM} - \mathcal{MR}_u$ . Then,

$$\begin{aligned}\rho \left( \mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T \right) &\stackrel{(9.605)}{\leq} \rho(I_{NM} - \mathcal{MR}) \\ &= \rho(I_{NM} - \mathcal{MR}_u).\end{aligned}$$
 $\square$

The next statement considers the case of ATC and CTA strategies (9.137) and (9.145) without information exchange, which corresponds to the case  $C = I_N$ . The result establishes that these strategies always enhance the convergence rate over the non-cooperative case, without the need to assume uniform step-sizes or uniform covariance data.

**Theorem 9.5.5 (Convergence Rate is Enhanced when  $C = I$ ).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) and set  $C = I_N$ . This situation covers the ATC and CTA strategies (9.137) and (9.145), which do not involve information exchange. Consider two modes of operation. In one mode, each node in the network runs the (distributed) steepest-descent diffusion algorithm (9.168)–(9.170). In the second mode, each node operates individually and runs the non-cooperative steepest-descent algorithm (9.171). In both cases, the positive step-sizes are chosen to satisfy the required stability conditions (9.193) and (9.194), which in this case are met by*

$$\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}, \quad k = 1, 2, \dots, N. \quad (9.200)$$

*It then holds that the magnitude of the error vector,  $\|\tilde{w}_i\|$ , in the diffusion case decays to zero more rapidly than in the non-cooperative case. In other words, diffusion cooperation enhances convergence rate.*

**Proof.** When  $C = I_N$ , we get  $R_k = R_{u,k}$  and, therefore,  $\mathcal{R} = \mathcal{R}_u$  and  $I_{NM} - \mathcal{MR} = I_{NM} - \mathcal{MR}_u$ . Then,

$$\begin{aligned}\rho \left( \mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T \right) &\stackrel{(9.605)}{\leq} \rho(I_{NM} - \mathcal{MR}) \\ &= \rho(I_{NM} - \mathcal{MR}_u).\end{aligned}$$
 $\square$

The results of the previous theorems highlight the following important facts about the role of the combination matrices  $\{A_1, A_2, C\}$  in the convergence behavior of the diffusion strategy (9.163)–(9.165):

- a. The matrix  $C$  influences the stability of the network through its influence on the bound in (9.191). This is because the matrices  $\{R_k\}$  depend on the entries of  $C$ . The matrices  $\{A_1, A_2\}$  do not influence network stability in that they can be chosen arbitrarily and the network will remain stable under (9.191).
- b. The matrices  $\{A_1, A_2, C\}$  influence the rate of convergence of the network since they influence the spectral radius of the matrix  $\mathcal{A}_2^T (I_{NM} - \mathcal{MR}) \mathcal{A}_1^T$ , which controls the dynamics of the weight error vector in (9.189).

### 3.09.6 Performance of adaptive diffusion strategies

We now move on to examine the behavior of the *adaptive* diffusion implementations (9.153) and (9.154), and the influence of both gradient noise and measurement noise on convergence and steady-state performance. Due to the random nature of the perturbations, it becomes necessary to evaluate the behavior of the algorithms on average, using mean-square convergence analysis. For this reason, we shall study the convergence of the weight estimates both in the mean and mean-square sense. To do so, we will again consider a general diffusion structure that includes the ATC and CTA strategies (9.153) and (9.154) as special cases. We shall further resort to the boldface notation to refer to the measurements and weight estimates in order to highlight the fact that they are now being treated as random variables. In this way, the update equations become stochastic updates. Thus, consider the following general adaptive diffusion strategy for  $i \geq 0$ :

$$\boldsymbol{\phi}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \boldsymbol{w}_{\ell,i-1}, \quad (9.201)$$

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{\phi}_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \boldsymbol{u}_{\ell,i}^* [\mathbf{d}_{\ell}(i) - \boldsymbol{u}_{\ell,i} \boldsymbol{\phi}_{k,i-1}], \quad (9.202)$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \boldsymbol{\psi}_{\ell,i}. \quad (9.203)$$

As before, the scalars  $\{a_{1,\ell k}, c_{\ell k}, a_{2,\ell k}\}$  are nonnegative real coefficients corresponding to the  $(\ell, k)$  entries of  $N \times N$  combination matrices  $\{A_1, C, A_2\}$ , respectively. These matrices are assumed to satisfy the same conditions (9.166) or (9.167). Again, different choices for  $\{A_1, C, A_2\}$  correspond to different cooperation modes. For example, the choice  $A_1 = I_N$  and  $A_2 = A$  corresponds to the adaptive ATC implementation (9.153), while the choice  $A_1 = A$  and  $A_2 = I_N$  corresponds to the adaptive CTA implementation (9.154). Likewise, the choice  $C = I_N$  corresponds to the case in which the nodes only share weight estimates and the distributed diffusion recursions (9.201)–(9.203) become

$$\boldsymbol{\phi}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \boldsymbol{w}_{\ell,i-1}, \quad (9.204)$$

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{\phi}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* [\mathbf{d}_k(i) - \boldsymbol{u}_{k,i} \boldsymbol{\phi}_{k,i-1}], \quad (9.205)$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \boldsymbol{\psi}_{\ell,i}. \quad (9.206)$$

Furthermore, the choice  $A_1 = A_2 = C = I_N$  corresponds to the non-cooperative mode of operation, where each node runs the classical (stand-alone) least-mean-squares (LMS) filter independently of the other nodes [4–7]:

$$\boldsymbol{w}_{k,i} = \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i} [\mathbf{d}_k(i) - \boldsymbol{u}_{k,i} \boldsymbol{w}_{k,i-1}], \quad i \geq 0. \quad (9.207)$$

#### 3.09.6.1 Data model

When we studied the performance of the steepest-descent diffusion strategy (9.163)–(9.165) we exploited result (9.175), which indicated how the moments  $\{r_{du,k}, R_{u,k}\}$  that appeared in the recursions related

to the optimal solution  $w^o$ . Likewise, in order to be able to analyze the performance of the *adaptive* diffusion strategy (9.201)–(9.203), we need to know how the data  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  across the network relate to  $w^o$ . Motivated by the several examples presented earlier in Section 3.09.2, we shall assume that the data satisfy a linear model of the form:

$$\boxed{\mathbf{d}_k(i) = \mathbf{u}_{k,i} w^o + \mathbf{v}_k(i)}, \quad (9.208)$$

where  $\mathbf{v}_k(i)$  is measurement noise with variance  $\sigma_{v,k}^2$ :

$$\sigma_{v,k}^2 \triangleq \mathbb{E}|\mathbf{v}_k(i)|^2 \quad (9.209)$$

and where the stochastic processes  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  are assumed to be jointly wide-sense stationary with moments:

$$\sigma_{d,k}^2 \triangleq \mathbb{E}|\mathbf{d}_k(i)|^2 \quad (\text{scalar}), \quad (9.210)$$

$$R_{u,k} \triangleq \mathbb{E}\mathbf{u}_{k,i}^* \mathbf{u}_{k,i} > 0 \quad (M \times M), \quad (9.211)$$

$$r_{du,k} \triangleq \mathbb{E}\mathbf{d}_k(i)\mathbf{u}_{k,i}^* \quad (M \times 1). \quad (9.212)$$

All variables are assumed to be zero-mean. Furthermore, the noise process  $\{\mathbf{v}_k(i)\}$  is assumed to be temporally white and spatially independent, as described earlier by (9.6), namely,

$$\begin{cases} \mathbb{E}\mathbf{v}_k(i)\mathbf{v}_k^*(j) = 0, & \text{for all } i \neq j \text{ (temporal whiteness),} \\ \mathbb{E}\mathbf{v}_k(i)\mathbf{v}_m^*(j) = 0, & \text{for all } i, j \text{ whenever } k \neq m \text{ (spatial whiteness).} \end{cases} \quad (9.213)$$

The noise process  $\mathbf{v}_k(i)$  is further assumed to be independent of the regression data  $\mathbf{u}_{m,j}$  for all  $k, m$  and  $i, j$  so that:

$$\mathbb{E}\mathbf{v}_k(i)\mathbf{u}_{m,j}^* = 0, \quad \text{for all } k, m, i, j. \quad (9.214)$$

We shall also assume that the regression data are temporally white and spatially independent so that:

$$\mathbb{E}\mathbf{u}_{k,i}^* \mathbf{u}_{\ell,j} = R_{u,k} \delta_{k\ell} \delta_{ij}. \quad (9.215)$$

Although we are going to derive performance measures for the network under this independence assumption on the regression data, it turns out that the resulting expressions continue to match well with simulation results for sufficiently small step-sizes, even when the independence assumption does not hold (in a manner similar to the behavior of stand-alone adaptive filters) [4,5].

### 3.09.6.2 Performance measures

Our objective is to analyze whether, and how fast, the weight estimates  $\{\mathbf{w}_{k,i}\}$  from the adaptive diffusion implementation (9.201)–(9.203) converge towards  $w^o$ . To do so, we again introduce the  $M \times 1$  weight error vectors:

$$\tilde{\phi}_{k,i} \triangleq w^o - \phi_{k,i}, \quad (9.216)$$

$$\tilde{\psi}_{k,i} \triangleq w^o - \psi_{k,i}, \quad (9.217)$$

$$\tilde{\mathbf{w}}_{k,i} \triangleq w^o - \mathbf{w}_{k,i}. \quad (9.218)$$

Each of these error vectors measures the residual relative to the desired  $w^o$  in (9.208). We further introduce two scalar error measures:

$$\mathbf{e}_k(i) \triangleq \mathbf{d}_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i-1} \quad (\text{output error}), \quad (9.219)$$

$$\mathbf{e}_{a,k}(i) \triangleq \mathbf{u}_{k,i} \tilde{\mathbf{w}}_{k,i-1} \quad (\text{a priori error}). \quad (9.220)$$

The first error measures how well the term  $\mathbf{u}_{k,i} \mathbf{w}_{k,i-1}$  approximates the measured data,  $\mathbf{d}_k(i)$ ; in view of (9.208), this error can be interpreted as an estimator for the noise term  $\mathbf{v}_k(i)$ . If node  $k$  is able to estimate  $w^o$  well, then  $\mathbf{e}_k(i)$  would get close to  $\mathbf{v}_k(i)$ . Therefore, under ideal conditions, we would expect the variance of  $\mathbf{e}_k(i)$  to tend towards the variance of  $\mathbf{v}_k(i)$ . However, as remarked earlier in (9.31), there is generally an offset term for adaptive implementations that is captured by the variance of the *a priori* error,  $\mathbf{e}_{a,k}(i)$ . This second error measures how well  $\mathbf{u}_{k,i} \mathbf{w}_{k,i-1}$  approximates the uncorrupted term  $\mathbf{u}_{k,i} w^o$ . Using the data model (9.208), we can relate  $\{\mathbf{e}_k(i), \mathbf{e}_{a,k}(i)\}$  as

$$\mathbf{e}_k(i) = \mathbf{e}_{a,k} + \mathbf{v}_k(i). \quad (9.221)$$

Since the noise component,  $\mathbf{v}_k(i)$ , is assumed to be zero-mean and independent of all other random variables, we recover (9.31):

$$\boxed{\mathbb{E}|\mathbf{e}_k(i)|^2 = \mathbb{E}|\mathbf{e}_{a,k}(i)|^2 + \sigma_{v,k}^2}. \quad (9.222)$$

This relation confirms that the variance of the output error,  $\mathbf{e}_k(i)$ , is at least as large as  $\sigma_{v,k}^2$  and away from it by an amount that is equal to the variance of the *a priori* error,  $\mathbf{e}_{a,k}(i)$ . Accordingly, in order to quantify the performance of any particular node in the network, we define the mean-square-error (MSE) and excess-mean-square-error (EMSE) for node  $k$  as the following steady-state measures:

$$\text{MSE}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}|\mathbf{e}_k(i)|^2, \quad (9.223)$$

$$\text{EMSE}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}|\mathbf{e}_{a,k}(i)|^2. \quad (9.224)$$

Then, it holds that

$$\boxed{\text{MSE}_k = \text{EMSE}_k + \sigma_{v,k}^2}. \quad (9.225)$$

Therefore, the EMSE term quantifies the size of the offset in the MSE performance of each node. We also define the mean-square-deviation (MSD) of each node as the steady-state measure:

$$\text{MSD}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|^2, \quad (9.226)$$

which measures how far  $\mathbf{w}_{k,i}$  is from  $w^o$  in the mean-square-error sense.

We indicated earlier in (9.36) and (9.37) how the MSD and EMSE of stand-alone LMS filters in the non-cooperative case depend on  $\{\mu_k, \sigma_v^2, R_{u,k}\}$ . In this section, we examine how cooperation among the nodes influences their performance. Since cooperation couples the operation of the nodes, with data originating from one node influencing the behavior of its neighbors and their neighbors, the study of the network performance requires more effort than in the non-cooperative case. Nevertheless, when all is said and done, we will arrive at expressions that approximate well the network performance and reveal some interesting conclusions.

### 3.09.6.3 Error recursions

Using the data model (9.208) and subtracting  $w^o$  from both sides of the relations in (9.201)–(9.203) we get

$$\tilde{\phi}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \tilde{w}_{\ell,i-1}, \quad (9.227)$$

$$\tilde{\psi}_{k,i} = \left( I_M - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i} \right) \tilde{\phi}_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell,i}^* \mathbf{v}_{\ell}(i), \quad (9.228)$$

$$\tilde{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \tilde{\psi}_{\ell,i}. \quad (9.229)$$

Comparing the second recursion with the corresponding recursion in the steepest-descent case (9.176)–(9.178), we see that two new effects arise: the effect of gradient noise, which replaces the covariance matrices  $R_{u,\ell}$  by the instantaneous approximation  $\mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i}$ , and the effect of measurement noise,  $\mathbf{v}_{\ell}(i)$ .

We again describe the above relations more compactly by collecting the information from across the network in block vectors and matrices. We collect the error vectors from across all nodes into the following  $N \times 1$  block vectors, whose individual entries are of size  $M \times 1$  each:

$$\tilde{\psi}_i \triangleq \begin{bmatrix} \tilde{\psi}_{1,i} \\ \tilde{\psi}_{2,i} \\ \vdots \\ \tilde{\psi}_{N,i} \end{bmatrix}, \quad \tilde{\phi}_i \triangleq \begin{bmatrix} \tilde{\phi}_{1,i} \\ \tilde{\phi}_{2,i} \\ \vdots \\ \tilde{\phi}_{N,i} \end{bmatrix}, \quad \tilde{w}_i \triangleq \begin{bmatrix} \tilde{w}_{1,i} \\ \tilde{w}_{2,i} \\ \vdots \\ \tilde{w}_{N,i} \end{bmatrix}. \quad (9.230)$$

The block quantities  $\{\tilde{\psi}_i, \tilde{\phi}_i, \tilde{w}_i\}$  represent the state of the errors across the network at time  $i$ . Likewise, we introduce the following  $N \times N$  block diagonal matrices, whose individual entries are of size  $M \times M$  each:

$$\mathcal{M} \triangleq \text{diag}\{\mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M\}, \quad (9.231)$$

$$\mathcal{R}_i \triangleq \text{diag} \left\{ \sum_{\ell \in \mathcal{N}_1} c_{\ell 1} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i}, \sum_{\ell \in \mathcal{N}_2} c_{\ell 2} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i}, \dots, \sum_{\ell \in \mathcal{N}_N} c_{\ell N} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i} \right\}. \quad (9.232)$$

Each block diagonal entry of  $\mathcal{R}_i$ , say, the  $k$ th entry, contains a combination of rank-one regression terms collected from the neighborhood of node  $k$ . In this way, the matrix  $\mathcal{R}_i$  is now stochastic *and* dependent on time, in contrast to the matrix  $\mathcal{R}$  in the steepest-descent case in (9.181), which was a constant matrix. Nevertheless, it holds that

$$\mathbb{E} \mathcal{R}_i = \mathcal{R}, \quad (9.233)$$

so that, on average,  $\mathcal{R}_i$  agrees with  $\mathcal{R}$ . We can simplify the notation by denoting the neighborhood combinations as follows:

$$\mathbf{R}_{k,i} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell,i}^* \mathbf{u}_{\ell,i}, \quad (9.234)$$

so that  $\mathcal{R}_i$  becomes

$$\mathcal{R}_i \triangleq \text{diag}\{\mathbf{R}_{1,i}, \mathbf{R}_{2,i}, \dots, \mathbf{R}_{N,i}\} \quad (\text{when } C \neq I). \quad (9.235)$$

Again, compared with the matrix  $R_k$  defined in (9.182), we find that  $\mathbf{R}_{k,i}$  is now both stochastic and time-dependent. Nevertheless, it again holds that

$$\mathbb{E}\mathbf{R}_{k,i} = R_k. \quad (9.236)$$

In the special case when  $C = I$ , the matrix  $\mathcal{R}_i$  reduces to

$$\mathcal{R}_{u,i} \triangleq \text{diag}\{\mathbf{u}_{1,i}^* \mathbf{u}_{1,i}, \mathbf{u}_{2,i}^* \mathbf{u}_{2,i}, \dots, \mathbf{u}_{N,i}^* \mathbf{u}_{N,i}\} \quad (\text{when } C = I) \quad (9.237)$$

with

$$\mathbb{E}\mathcal{R}_{u,i} = \mathcal{R}_u, \quad (9.238)$$

where  $\mathcal{R}_u$  was defined earlier in (9.184).

We further introduce the following  $N \times 1$  block column vector, whose entries are of size  $M \times 1$  each:

$$\mathbf{s}_i \triangleq \text{col}\{\mathbf{u}_{1,i}^* \mathbf{v}_1(i), \mathbf{u}_{2,i}^* \mathbf{v}_2(i), \dots, \mathbf{u}_{N,i}^* \mathbf{v}_N(i)\}. \quad (9.239)$$

Obviously, given that the regression data and measurement noise are zero-mean and independent of each other, we have

$$\mathbb{E}\mathbf{s}_i = 0 \quad (9.240)$$

and the covariance matrix of  $\mathbf{s}_i$  is  $N \times N$  block diagonal with blocks of size  $M \times M$ :

$$\mathcal{S} \triangleq \mathbb{E}\mathbf{s}_i \mathbf{s}_i^* = \text{diag}\{\sigma_{v,1}^2 R_{u,1}, \sigma_{v,2}^2 R_{u,2}, \dots, \sigma_{v,N}^2 R_{u,N}\}. \quad (9.241)$$

Returning to (9.227)–(9.229), we conclude that the following relations hold for the block quantities:

$$\tilde{\phi}_{i-1} = \mathcal{A}_1^T \tilde{\mathbf{w}}_{i-1}, \quad (9.242)$$

$$\tilde{\psi}_i = (I_{NM} - \mathcal{M}\mathcal{R}_i)\tilde{\phi}_{i-1} - \mathcal{M}\mathcal{C}^T \mathbf{s}_i, \quad (9.243)$$

$$\tilde{\mathbf{w}}_i = \mathcal{A}_2^T \tilde{\psi}_i, \quad (9.244)$$

where

$$\mathcal{C} \triangleq C \otimes I_M, \quad (9.245)$$

so that the network weight error vector,  $\tilde{\mathbf{w}}_i$ , ends up evolving according to the following *stochastic* recursion:

$$\tilde{\mathbf{w}}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}_i) \mathcal{A}_1^T \tilde{\mathbf{w}}_{i-1} - \mathcal{A}_2^T \mathcal{M}\mathcal{C}^T \mathbf{s}_i, \quad i \geq 0 \quad (\text{diffusion strategy}). \quad (9.246)$$

For comparison purposes, if each node operates individually and uses the non-cooperative LMS recursion (9.207), then the weight error vector across all  $N$  nodes would evolve according to the following stochastic recursion:

$$\tilde{\mathbf{w}}_i = (I_{NM} - \mathcal{M}\mathcal{R}_{u,i}) \tilde{\mathbf{w}}_{i-1} - \mathcal{M}\mathbf{s}_i, \quad i \geq 0 \quad (\text{non-cooperative strategy}), \quad (9.247)$$

where the matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$  do not appear, and  $\mathcal{R}_i$  is replaced by  $\mathcal{R}_{u,i}$  from (9.237).

### 3.09.6.4 Convergence in the mean

Taking expectations of both sides of (9.246) we find that:

$$\mathbb{E}\tilde{\mathbf{w}}_i = \mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R})\mathcal{A}_1^T \cdot \mathbb{E}\tilde{\mathbf{w}}_{i-1}, \quad i \geq 0 \quad (\text{diffusion strategy}), \quad (9.248)$$

where we used the fact that  $\tilde{\mathbf{w}}_{i-1}$  and  $\mathcal{R}_i$  are independent of each other in view of our earlier assumptions on the regression data and noise in Section 3.09.6.1. Comparing with the error recursion (9.189) in the steepest-descent case, we find that both recursions are identical with  $\tilde{\mathbf{w}}_i$  replaced by  $\mathbb{E}\tilde{\mathbf{w}}_i$ . Therefore, the convergence statements from the steepest-descent case can be extended to the adaptive case to provide conditions on the step-size to ensure stability in the mean, i.e., to ensure

$$\mathbb{E}\tilde{\mathbf{w}}_i \rightarrow 0 \quad \text{as } i \rightarrow \infty. \quad (9.249)$$

When (9.249) is guaranteed, we say that the adaptive diffusion solution is asymptotically unbiased. The following statements restate the results of Theorems 9.5.1–9.5.5 in the context of mean error analysis.

**Theorem 9.6.1 (Convergence in the Mean).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a right stochastic matrix  $C$  and left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) or (9.167). Assume each node in the network measures data that satisfy the conditions described in Section 3.09.6.1, and runs the adaptive diffusion algorithm (9.201)–(9.203). Then, all estimators  $\{\mathbf{w}_{k,i}\}$  across the network converge in the mean to the optimal solution  $w^o$  if the positive step-size parameters  $\{\mu_k\}$  satisfy*

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}, \quad (9.250)$$

where the neighborhood covariance matrix  $R_k$  is defined by (9.182). In other words,  $\mathbb{E}\mathbf{w}_{k,i} \rightarrow w^o$  for all nodes  $k$  as  $i \rightarrow \infty$ .

Observe again that the mean stability condition (9.250) does not depend on the specific combination matrices  $A_1$  and  $A_2$  that are being used. Only the combination matrix  $C$  influences the condition on the step-size through the neighborhood covariance matrices  $\{R_k\}$ . Observe further that the statement of the lemma does not require the network to be connected. Moreover, when  $C = I_N$ , in which case the nodes only share weight estimators and do not share neighborhood data  $\{\mathbf{d}_\ell(i), \mathbf{u}_{\ell,i}\}$  as in (9.204)–(9.206), condition (9.250) becomes

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}} \quad (\text{adaptive cooperation with } C = I_N). \quad (9.251)$$

Conditions (9.250) and (9.251) are reminiscent of a classical result for the stand-alone LMS algorithm, as in the non-cooperative case (9.207), where it is known that the estimator by each individual node in this case would converge in the mean to  $w^o$  if, and only if, its step-size satisfies

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}} \quad (\text{non-cooperative adaptation}). \quad (9.252)$$

The following statement provides a bi-directional result that ensures the mean convergence of the adaptive diffusion strategy for *any* choice of left-stochastic combination matrices  $A_1$  and  $A_2$ .

**Theorem 9.6.2 (Mean Convergence for Arbitrary Combination Matrices).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a right stochastic matrix  $C$  satisfying (9.166). Assume each node in the network measures data that satisfy the conditions described in Section 3.09.6.1. Then, the estimators  $\{\mathbf{w}_{k,i}\}$  generated by the adaptive diffusion strategy (9.201)–(9.203), converge in the mean to  $\mathbf{w}^o$ , for all choices of left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) if, and only if,*

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}. \quad (9.253)$$

As was the case with steepest-descent diffusion strategies, the adaptive diffusion strategy (9.201)–(9.203) also enhances the convergence rate of the mean of the error vector towards zero relative to the non-cooperative strategy (9.207). The next results restate Theorems 9.5.3–9.5.5; they assume  $C$  is a doubly stochastic matrix.

**Theorem 9.6.3 (Mean Convergence Rate is Enhanced: Uniform Step-Sizes).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a doubly stochastic matrix  $C$  satisfying (9.196) and left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166). Assume each node in the network measures data that satisfy the conditions described in Section 3.09.6.1. Consider two modes of operation. In one mode, each node in the network runs the adaptive diffusion algorithm (9.201)–(9.203). In the second mode, each node operates individually and runs the non-cooperative LMS algorithm (9.207). In both cases, the positive step-sizes used by all nodes are assumed to be the same, say,  $\mu_k = \mu$  for all  $k$ , and the value of  $\mu$  is chosen to satisfy the required mean stability conditions (9.250) and (9.252), which are met by selecting*

$$\mu < \min_{1 \leq k \leq N} \left\{ \frac{2}{\lambda_{\max}(R_{u,k})} \right\}. \quad (9.254)$$

*It then holds that the magnitude of the mean error vector,  $\|\mathbb{E}\tilde{\mathbf{w}}_i\|$  in the diffusion case decays to zero more rapidly than in the non-cooperative case. In other words, diffusion enhances convergence rate.*

**Theorem 9.6.4 (Mean Convergence Rate is Enhanced: Uniform Covariance Data).** *Consider the same setting of Theorem 9.6.3. Assume the covariance data are uniform across all nodes, say,  $R_{u,k} = R_u$  is independent of  $k$ . Assume further that the nodes in both modes of operation employ steps-sizes  $\mu_k$  that are chosen to satisfy the required stability conditions (9.250) and (9.252), which in this case are met by:*

$$\mu_k < \frac{2}{\lambda_{\max}(R_u)}, \quad k = 1, 2, \dots, N. \quad (9.255)$$

*It then holds that the magnitude of the mean error vector,  $\|\mathbb{E}\tilde{\mathbf{w}}_i\|$ , in the diffusion case also decays to zero more rapidly than in the non-cooperative case. In other words, diffusion enhances convergence rate.*

The next statement considers the case of ATC and CTA strategies (9.204)–(9.206) without information exchange, which correspond to the choice  $C = I_N$ . The result establishes that these strategies always

enhance the convergence rate over the non-cooperative case, without the need to assume uniform step-sizes or uniform covariance data.

**Theorem 9.6.5 (Mean Convergence Rate is Enhanced when  $C = I$ ).** Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166) and set  $C = I_N$ . This situation covers the ATC and CTA strategies (9.204)–(9.206) that do not involve information exchange. Assume each node in the network measures data that satisfy the conditions described in Section 3.09.6.1. Consider two modes of operation. In one mode, each node in the network runs the adaptive diffusion algorithm (9.163)–(9.165). In the second mode, each node operates individually and runs the non-cooperative LMS algorithm (9.207). In both cases, the positive step-sizes are chosen to satisfy the required stability conditions (9.251) and (9.252), which in this case are met by

$$\mu_k < \frac{2}{\lambda_{\max}(R_{u,k})}, \quad k = 1, 2, \dots, N. \quad (9.256)$$

It then holds that the magnitude of the mean error vector,  $\|\mathbb{E}\tilde{\mathbf{w}}_i\|$ , in the diffusion case decays to zero more rapidly than in the non-cooperative case. In other words, diffusion cooperation enhances convergence rate.

The results of the previous theorems again highlight the following important facts about the role of the combination matrices  $\{A_1, A_2, C\}$  in the convergence behavior of the adaptive diffusion strategy (9.201)–(9.203):

- a. The matrix  $C$  influences the mean stability of the network through its influence on the bound in (9.250). This is because the matrices  $\{R_k\}$  depend on the entries of  $C$ . The matrices  $\{A_1, A_2\}$  do not influence network mean stability in that they can be chosen arbitrarily and the network will remain stable under (9.250).
- b. The matrices  $\{A_1, A_2, C\}$  influence the rate of convergence of the mean weight-error vector over the network since they influence the spectral radius of the matrix  $\mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R})\mathcal{A}_1^T$ , which controls the dynamics of the weight error vector in (9.248).

### 3.09.6.5 Mean-square stability

It is not sufficient to ensure the stability of the weight-error vector in the mean sense. The error vectors,  $\tilde{\mathbf{w}}_{k,i}$ , may be converging on average to zero but they may have large fluctuations around the zero value. We therefore need to examine how small the error vectors get. To do so, we perform a mean-square-error analysis. The purpose of the analysis is to evaluate how the variances  $\mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|^2$  evolve with time and what their steady-state values are, for each node  $k$ .

In this section, we are particularly interested in evaluating the evolution of two mean-square-errors, namely,

$$\mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|^2 \quad \text{and} \quad \mathbb{E}|\mathbf{e}_{a,k}(i)|^2. \quad (9.257)$$

The steady-state values of these quantities determine the MSD and EMSE performance levels at node  $k$  and, therefore, convey critical information about the performance of the network. Under the independence assumption on the regression data from Section 3.09.6.1, it can be verified that the EMSE

variance can be written as:

$$\begin{aligned}
 \mathbb{E}|\mathbf{e}_{a,k}(i)|^2 &\triangleq \mathbb{E}|\mathbf{u}_{k,i}\tilde{\mathbf{w}}_{k,i-1}|^2 \\
 &= \mathbb{E}\tilde{\mathbf{w}}_{k,i-1}^*\mathbf{u}_{k,i}^*\mathbf{u}_{k,i}\tilde{\mathbf{w}}_{k,i-1} \\
 &= \mathbb{E}[\mathbb{E}(\tilde{\mathbf{w}}_{k,i-1}^*\mathbf{u}_{k,i}^*\mathbf{u}_{k,i}\tilde{\mathbf{w}}_{k,i-1}|\tilde{\mathbf{w}}_{k,i-1})] \\
 &= \mathbb{E}\tilde{\mathbf{w}}_{k,i-1}^*[\mathbb{E}\mathbf{u}_{k,i}^*\mathbf{u}_{k,i}]\tilde{\mathbf{w}}_{k,i-1} \\
 &= \mathbb{E}\tilde{\mathbf{w}}_{k,i-1}^*R_{u,k}\tilde{\mathbf{w}}_{k,i-1} \\
 &= \mathbb{E}\|\tilde{\mathbf{w}}_{k,i-1}\|_{R_{u,k}}^2,
 \end{aligned} \tag{9.258}$$

in terms of a weighted square measure with weighting matrix  $R_{u,k}$ . Here we are using the notation  $\|x\|_\Sigma^2$  to denote the weighted square quantity  $x^*\Sigma x$ , for any column vector  $x$  and matrix  $\Sigma$ . Thus, we can evaluate mean-square-errors of the form (9.257) by evaluating the means of weighted square quantities of the following form:

$$\mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2 \tag{9.259}$$

for an arbitrary Hermitian nonnegative-definite weighting matrix  $\Sigma_k$  that we are free to choose. By setting  $\Sigma_k$  to different values (say,  $\Sigma_k = I$  or  $\Sigma_k = R_{u,k}$ ), we can extract various types of information about the nodes and the network, as the discussion will reveal. The approach we follow is based on the energy conservation framework of [4, 5, 55].

So, let  $\Sigma$  denote an arbitrary  $N \times N$  block Hermitian nonnegative-definite matrix that we are free to choose, with  $M \times M$  block entries  $\{\Sigma_{\ell k}\}$ . Let  $\sigma$  denote the  $(NM)^2 \times 1$  vector that is obtained by stacking the columns of  $\Sigma$  on top of each other, written as

$$\sigma \triangleq \text{vec}(\Sigma). \tag{9.260}$$

In the sequel, it will become more convenient to work with the vector representation  $\sigma$  than with the matrix  $\Sigma$  itself.

We start from the weight-error vector recursion (9.246) and re-write it more compactly as:

$$\boxed{\tilde{\mathbf{w}}_i = \mathcal{B}_i \tilde{\mathbf{w}}_{i-1} - \mathcal{G}s_i, \quad i \geq 0}, \tag{9.261}$$

where the coefficient matrices  $\mathcal{B}_i$  and  $\mathcal{G}$  are short-hand representations for

$$\boxed{\mathcal{B}_i \triangleq \mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R}_i)\mathcal{A}_1^T} \tag{9.262}$$

and

$$\boxed{\mathcal{G} \triangleq \mathcal{A}_2^T \mathcal{M} \mathcal{C}^T}. \tag{9.263}$$

Note that  $\mathcal{B}_i$  is stochastic and time-variant, while  $\mathcal{G}$  is constant. We denote the mean of  $\mathcal{B}_i$  by

$$\boxed{\mathcal{B} \triangleq \mathbb{E}\mathcal{B}_i = \mathcal{A}_2^T(I_{NM} - \mathcal{M}\mathcal{R})\mathcal{A}_1^T}, \tag{9.264}$$

where  $\mathcal{R}$  is defined by (9.181). Now equating weighted square measures on both sides of (9.261) we get

$$\|\tilde{\mathbf{w}}_i\|_\Sigma^2 = \|\mathcal{B}_i \tilde{\mathbf{w}}_{i-1} - \mathcal{G}s_i\|_\Sigma^2. \tag{9.265}$$

Expanding the right-hand side we find that

$$\|\tilde{\mathbf{w}}_i\|_{\Sigma}^2 = \tilde{\mathbf{w}}_{i-1}^* \mathcal{B}_i^* \Sigma \mathcal{B}_i \tilde{\mathbf{w}}_{i-1} + \mathbf{s}_i^* \mathcal{G}^T \Sigma \mathcal{G} \mathbf{s}_i - \tilde{\mathbf{w}}_{i-1}^* \mathcal{B}_i^* \Sigma \mathcal{G} \mathbf{s}_i - \mathbf{s}_i^* \mathcal{G}^T \Sigma \mathcal{B}_i \tilde{\mathbf{w}}_{i-1}. \quad (9.266)$$

Under expectation, the last two terms on the right-hand side evaluate to zero so that

$$\mathbb{E}\|\tilde{\mathbf{w}}_i\|_{\Sigma}^2 = \mathbb{E}(\tilde{\mathbf{w}}_{i-1}^* \mathcal{B}_i^* \Sigma \mathcal{B}_i \tilde{\mathbf{w}}_{i-1}) + \mathbb{E}(\mathbf{s}_i^* \mathcal{G}^T \Sigma \mathcal{G} \mathbf{s}_i). \quad (9.267)$$

Let us evaluate each of the expectations on the right-hand side. The last expectation is given by

$$\begin{aligned} \mathbb{E}(\mathbf{s}_i^* \mathcal{G}^T \Sigma \mathcal{G} \mathbf{s}_i) &= \text{Tr}(\mathcal{G}^T \Sigma \mathcal{G} \mathbb{E} \mathbf{s}_i \mathbf{s}_i^*) \\ &\stackrel{(9.241)}{=} \text{Tr}(\mathcal{G}^T \Sigma \mathcal{G} \mathcal{S}) \\ &= \text{Tr}(\Sigma \mathcal{G} \mathcal{S} \mathcal{G}^T), \end{aligned} \quad (9.268)$$

where  $\mathcal{S}$  is defined by (9.241) and where we used the fact that  $\text{Tr}(AB) = \text{Tr}(BA)$  for any two matrices  $A$  and  $B$  of compatible dimensions. With regards to the first expectation on the right-hand side of (9.267), we have

$$\begin{aligned} \mathbb{E}(\tilde{\mathbf{w}}_{i-1}^* \mathcal{B}_i^* \Sigma \mathcal{B}_i \tilde{\mathbf{w}}_{i-1}) &= \mathbb{E}[\mathbb{E}(\tilde{\mathbf{w}}_{i-1}^* \mathcal{B}_i^* \Sigma \mathcal{B}_i \tilde{\mathbf{w}}_{i-1} | \tilde{\mathbf{w}}_{i-1})] \\ &= \mathbb{E}\tilde{\mathbf{w}}_{i-1}^* [\mathbb{E}(\mathcal{B}_i^* \Sigma \mathcal{B}_i)] \tilde{\mathbf{w}}_{i-1} \\ &\triangleq \mathbb{E}\tilde{\mathbf{w}}_{i-1}^* \Sigma' \tilde{\mathbf{w}}_{i-1} \\ &= \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{\Sigma'}^2, \end{aligned} \quad (9.269)$$

where we introduced the nonnegative-definite weighting matrix

$$\begin{aligned} \Sigma' &\triangleq \mathbb{E}\mathcal{B}_i^* \Sigma \mathcal{B}_i \\ &\stackrel{(9.270)}{=} \mathbb{E}\mathcal{A}_1(I_{NM} - \mathcal{R}_i \mathcal{M})\mathcal{A}_2 \Sigma \mathcal{A}_2^T (I_{NM} - \mathcal{M} \mathcal{R}_i) \mathcal{A}_1^T \\ &= \mathcal{A}_1 \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{A}_1^T - \mathcal{A}_1 \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{M} \mathcal{R} \mathcal{A}_1^T - \mathcal{A}_1 \mathcal{R} \mathcal{M} \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{A}_1^T + O(\mathcal{M}^2), \end{aligned} \quad (9.270)$$

where  $\mathcal{R}$  is defined by (9.181) and the term  $O(\mathcal{M}^2)$  denotes the following factor, which depends on the square of the step-sizes,  $\{\mu_k^2\}$ :

$$O(\mathcal{M}^2) = \mathbb{E}(\mathcal{A}_1 \mathcal{R}_i \mathcal{M} \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{M} \mathcal{R}_i \mathcal{A}_1^T). \quad (9.271)$$

The evaluation of the above expectation depends on higher-order moments of the regression data. While we can continue with the analysis by taking this factor into account, as was done in [4, 5, 18, 55], it is sufficient for the exposition in this chapter to focus on the case of sufficiently small step-sizes where terms involving higher powers of the step-sizes can be ignored. Therefore, we continue our discussion by letting

$$\boxed{\Sigma' \triangleq \mathcal{A}_1 \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{A}_1^T - \mathcal{A}_1 \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{M} \mathcal{R} \mathcal{A}_1^T - \mathcal{A}_1 \mathcal{R} \mathcal{M} \mathcal{A}_2 \Sigma \mathcal{A}_2^T \mathcal{A}_1^T}. \quad (9.272)$$

The weighting matrix  $\Sigma'$  is fully defined in terms of the step-size matrix,  $\mathcal{M}$ , the network topology through the matrices  $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{C}\}$ , and the regression statistical profile through  $\mathcal{R}$ . Expression (9.272)

tells us how to construct  $\Sigma'$  from  $\Sigma$ . The expression can be transformed into a more compact and revealing form if we instead relate the vector forms  $\sigma' = \text{vec}(\Sigma')$  and  $\sigma = \text{vec}(\Sigma)$ . Using the following equalities for arbitrary matrices  $\{U, W, \Sigma\}$  of compatible dimensions [5]:

$$\text{vec}(U\Sigma W) = (W^T \otimes U)\sigma, \quad (9.273)$$

$$\text{Tr}(\Sigma W) = [\text{vec}(W^T)]^T \sigma, \quad (9.274)$$

and applying the vec operation to both sides of (9.272) we get

$$\sigma' = (\mathcal{A}_1 \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{A}_2)\sigma - (\mathcal{A}_1 \mathcal{R}^T \mathcal{M} \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{A}_2)\sigma - (\mathcal{A}_1 \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{R} \mathcal{M} \mathcal{A}_2)\sigma.$$

That is,

$$\boxed{\sigma' \triangleq \mathcal{F}\sigma}, \quad (9.275)$$

where we are introducing the coefficient matrix of size  $(NM)^2 \times (NM)^2$ :

$$\boxed{\mathcal{F} \triangleq (\mathcal{A}_1 \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{A}_2) - (\mathcal{A}_1 \mathcal{R}^T \mathcal{M} \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{A}_2) - (\mathcal{A}_1 \mathcal{A}_2 \otimes \mathcal{A}_1 \mathcal{R} \mathcal{M} \mathcal{A}_2)}. \quad (9.276)$$

A reasonable approximate expression for  $\mathcal{F}$  for sufficiently small step-sizes is

$$\boxed{\mathcal{F} \approx \mathcal{B}^T \otimes \mathcal{B}^*}. \quad (9.277)$$

Indeed, if we replace  $\mathcal{B}$  from (9.264) into (9.277) and expand terms, we obtain the same factors that appear in (9.276) plus an additional term that depends on the square of the step-sizes,  $\{\mu_k^2\}$ , whose effect can be ignored for sufficiently small step-sizes.

In this way, and using in addition property (9.274), we find that relation (9.267) becomes:

$$\boxed{\mathbb{E}\|\tilde{\mathbf{w}}_i\|_{\Sigma}^2 = \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{\Sigma'}^2 + [\text{vec}(\mathcal{G}\mathcal{S}^T\mathcal{G}^T)]^T \sigma}. \quad (9.278)$$

The last term is dependent on the network topology through the matrix  $\mathcal{G}$ , which is defined in terms of  $\{\mathcal{A}_2, \mathcal{C}, \mathcal{M}\}$ , and the noise and regression data statistical profile through  $\mathcal{S}$ . It is convenient to introduce the alternative notation  $\|x\|_{\sigma}^2$  to refer to the weighted square quantity  $\|x\|_{\Sigma}^2$ , where  $\sigma = \text{vec}(\Sigma)$ . We shall use these two notations interchangeably. The convenience of the vector notation is that it allows us to exploit the simpler linear relation (9.275) between  $\sigma'$  and  $\sigma$  to rewrite (9.278) as shown in (9.279) below, with the *same* weight vector  $\sigma$  appearing on both sides.

**Theorem 9.6.6 (Variance Relation).** *Consider the data model of Section 3.09.6.1 and the independence statistical conditions imposed on the noise and regression data, including (9.208)–(9.215). Assume further sufficiently small step-sizes are used so that terms that depend on higher-powers of the step-sizes can be ignored. Pick left stochastic matrices  $A_1$  and  $A_2$  and a right stochastic matrix  $C$  satisfying (9.166). Under these conditions, the weight-error vector  $\tilde{\mathbf{w}}_i = \text{col}\{\tilde{\mathbf{w}}_{k,i}\}_{k=1}^N$  associated with a network running the adaptive diffusion strategy (9.201)–(9.203) satisfies the following variance relation:*

$$\boxed{\mathbb{E}\|\tilde{\mathbf{w}}_i\|_{\sigma}^2 = \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{\mathcal{F}\sigma}^2 + [\text{vec}(\mathcal{Y}^T)]^T \sigma} \quad (9.279)$$

for any Hermitian nonnegative-definite matrix  $\Sigma$  with  $\sigma = \text{vec}(\Sigma)$ , and where  $\{\mathcal{S}, \mathcal{G}, \mathcal{F}\}$  are defined by (9.241), (9.263), and (9.277), and

$$\mathcal{Y} \triangleq \mathcal{G}\mathcal{S}\mathcal{G}^T . \quad (9.280)$$

□

Note that relation (9.279) is not an actual recursion; this is because the weighting matrices  $\{\sigma, \mathcal{F}\sigma\}$  on both sides of the equality are different. The relation can be transformed into a true recursion by expanding it into a convenient state-space model; this argument was pursued in [4, 5, 18, 55] and is not necessary for the exposition here, except to say that stability of the matrix  $\mathcal{F}$  ensures the mean-square stability of the filter—this fact is also established further ahead through relation (9.327). By mean-square stability we mean that each term  $\mathbb{E}\|\tilde{w}_{k,i}\|^2$  remains bounded over time and converges to a steady-state  $\text{MSD}_k$  value. Moreover, the spectral radius of  $\mathcal{F}$  controls the rate of convergence of  $\mathbb{E}\|\tilde{w}_i\|^2$  towards its steady-state value.

**Theorem 9.6.7 (Mean-Square Stability).** *Consider the same setting of Theorem 9.6.6. The adaptive diffusion strategy (9.201)–(9.203) is mean-square stable if, and only if, the matrix  $\mathcal{F}$  defined by (9.276), or its approximation (9.277), is stable (i.e., all its eigenvalues lie strictly inside the unit disc). This condition is satisfied by sufficiently small positive step-sizes  $\{\mu_k\}$  that are also smaller than the following bound:*

$$\mu_k < \frac{2}{\lambda_{\max}(R_k)}, \quad (9.281)$$

where the neighborhood covariance matrix  $R_k$  is defined by (9.182). Moreover, the convergence rate of the algorithm is determined by the value  $[\rho(\mathcal{B})]^2$  (the square of the spectral radius of  $\mathcal{B}$ ).

**Proof.** Recall that, for two arbitrary matrices  $A$  and  $B$  of compatible dimensions, the eigenvalues of the Kronecker product  $A \otimes B$  is formed of all product combinations  $\lambda_i(A)\lambda_j(B)$  of the eigenvalues of  $A$  and  $B$  [19]. Therefore, for sufficiently small step-sizes, we can use expression (9.277) to note that  $\rho(\mathcal{F}) = [\rho(\mathcal{B})]^2$ . It follows that  $\mathcal{F}$  is stable if, and only if,  $\mathcal{B}$  is stable. We already noted earlier in Theorem 9.6.1 that condition (9.281) ensures the stability of  $\mathcal{B}$ . Therefore, step-sizes that ensure stability in the mean and are sufficiently small will also ensure mean-square stability. □

**Remark.** More generally, had we not ignored the second-order term (9.271), the expression for  $\mathcal{F}$  would have been the following. Starting from the definition  $\Sigma' = \mathbb{E}\mathcal{B}_i^*\Sigma\mathcal{B}_i$ , we would get

$$\sigma' = (\mathbb{E}\mathcal{B}_i^T \otimes \mathcal{B}_i^*)\sigma,$$

so that

$$\begin{aligned} \mathcal{F} &\triangleq \mathbb{E}(\mathcal{B}_i^T \otimes \mathcal{B}_i^*) \quad (\text{for general step-sizes}) \\ &= (\mathcal{A}_1 \otimes \mathcal{A}_1) \cdot \left\{ I - (\mathcal{R}^T \mathcal{M} \otimes I) - (I \otimes \mathcal{R} \mathcal{M}) + \mathbb{E}(\mathcal{R}_i^T \mathcal{M} \otimes \mathcal{R}_i \mathcal{M}) \right\} \cdot (\mathcal{A}_2 \otimes \mathcal{A}_2). \end{aligned} \quad (9.282)$$

Mean-square stability of the filter would then require the step-sizes  $\{\mu_k\}$  to be chosen such that they ensure the stability of this matrix  $\mathcal{F}$  (in addition to condition (9.281) to ensure mean stability). □

### 3.09.6.6 Network mean-square performance

We can now use the variance relation (9.279) to evaluate the network performance, as well as the performance of the individual nodes, in steady-state. Since the dynamics is mean-square stable for sufficiently small step-sizes, we take the limit of (9.279) as  $i \rightarrow \infty$  and write:

$$\lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_i\|_\sigma^2 = \lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{\mathcal{F}\sigma}^2 + [\text{vec}(\mathcal{Y}^T)]^T \sigma. \quad (9.283)$$

Grouping terms leads to the following result.

**Corollary 9.6.1 (Steady-State Variance Relation).** *Consider the same setting of Theorem 9.6.6. The weight-error vector,  $\tilde{\mathbf{w}}_i = \text{col}\{\tilde{\mathbf{w}}_{k,i}\}_{k=1}^N$ , of the adaptive diffusion strategy (9.201)–(9.203) satisfies the following relation in steady-state:*

$$\boxed{\lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_i\|_{(I-\mathcal{F})\sigma}^2 = [\text{vec}(\mathcal{Y}^T)]^T \sigma} \quad (9.284)$$

for any Hermitian nonnegative-definite matrix  $\Sigma$  with  $\sigma = \text{vec}(\Sigma)$ , and where  $\{\mathcal{F}, \mathcal{Y}\}$  are defined by (9.277) and (9.280).  $\square$

Expression (9.284) is a very useful relation; it allows us to evaluate the network MSD and EMSE through proper selection of the weighting vector  $\sigma$  (or, equivalently, the weighting matrix  $\Sigma$ ). For example, the network MSD is defined as the average value:

$$\text{MSD}^{\text{network}} \triangleq \lim_{i \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|^2, \quad (9.285)$$

which amounts to averaging the MSDs of the individual nodes. Therefore,

$$\text{MSD}^{\text{network}} = \lim_{i \rightarrow \infty} \frac{1}{N} \mathbb{E}\|\tilde{\mathbf{w}}_i\|^2 = \lim_{i \rightarrow \infty} \mathbb{E}\|\tilde{\mathbf{w}}_i\|_{1/N}^2. \quad (9.286)$$

This means that in order to recover the network MSD from relation (9.284), we should select the weighting vector  $\sigma$  such that

$$(I - \mathcal{F})\sigma = \frac{1}{N} \text{vec}(I_{NM}).$$

Solving for  $\sigma$  and substituting back into (9.284) we arrive at the following expression for the network MSD:

$$\boxed{\text{MSD}^{\text{network}} = \frac{1}{N} \cdot [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(I_{NM})}. \quad (9.287)$$

Likewise, the network EMSE is defined as the average value

$$\begin{aligned} \text{EMSE}^{\text{network}} &\triangleq \lim_{i \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \mathbb{E}|\mathbf{e}_{a,k}(i)|^2 \\ &= \lim_{i \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|_{R_{u,k}}^2, \end{aligned} \quad (9.288)$$

which amounts to averaging the EMSEs of the individual nodes. Therefore,

$$\text{EMSE}^{\text{network}} = \lim_{i \rightarrow \infty} \frac{1}{N} \mathbb{E} \|\tilde{\mathbf{w}}_i\|_{\text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\}}^2 = \lim_{i \rightarrow \infty} \frac{1}{N} \mathbb{E} \|\tilde{\mathbf{w}}_i\|_{\mathcal{R}_u}^2, \quad (9.289)$$

where  $\mathcal{R}_u$  is the matrix defined earlier by (9.184), and which we repeat below for ease of reference:

$$\mathcal{R}_u = \text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\}. \quad (9.290)$$

This means that in order to recover the network EMSE from relation (9.284), we should select the weighting vector  $\sigma$  such that

$$(I - \mathcal{F})\sigma = \frac{1}{N} \text{vec}(\mathcal{R}_u). \quad (9.291)$$

Solving for  $\sigma$  and substituting into (9.284) we arrive at the following expression for the network EMSE:

$$\text{EMSE}^{\text{network}} = \frac{1}{N} \cdot [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(\mathcal{R}_u) . \quad (9.292)$$

### 3.09.6.7 Mean-square performance of individual nodes

We can also assess the mean-square performance of the individual nodes in the network from (9.284). For instance, the MSD of any particular node  $k$  is defined by

$$\text{MSD}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{k,i}\|^2. \quad (9.293)$$

Introduce the  $N \times N$  block diagonal matrix with blocks of size  $M \times M$ , where all blocks on the diagonal are zero except for an identity matrix on the diagonal block of index  $k$ , i.e.,

$$\mathcal{J}_k \triangleq \text{diag}\{0_M, \dots, 0_M, I_M, 0_M, \dots, 0_M\}. \quad (9.294)$$

Then, we can express the node MSD as follows:

$$\text{MSD}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_i\|_{\mathcal{J}_k}^2. \quad (9.295)$$

The same argument that was used to obtain the network MSD then leads to

$$\text{MSD}_k = [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(\mathcal{J}_k) . \quad (9.296)$$

Likewise, the EMSE of node  $k$  is defined by

$$\begin{aligned} \text{EMSE}_k &\triangleq \lim_{i \rightarrow \infty} \mathbb{E} |\mathbf{e}_{a,k}(i)|^2 \\ &= \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{k,i}\|_{R_{u,k}}^2. \end{aligned} \quad (9.297)$$

Introduce the  $N \times N$  block diagonal matrix with blocks of size  $M \times M$ , where all blocks on the diagonal are zero except for the diagonal block of index  $k$  whose value is  $R_{u,k}$ , i.e.,

$$\mathcal{T}_k \triangleq \text{diag}\{0_M, \dots, 0_M, R_{u,k}, 0_M, \dots, 0_M\}. \quad (9.298)$$

Then, we can express the node EMSE as follows:

$$\text{EMSE}_k \triangleq \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{w}_i\|_{T_k}^2. \quad (9.299)$$

The same argument that was used to obtain the network EMSE then leads to

$$\boxed{\text{EMSE}_k = [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(T_k)}. \quad (9.300)$$

We summarize the results in the following statement.

**Theorem 9.6.8 (Network Mean-Square Performance).** *Consider the same setting of Theorem 9.6.6. Introduce the  $1 \times (NM)^2$  row vector  $h^T$  defined by*

$$h^T \triangleq [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1}, \quad (9.301)$$

where  $\{\mathcal{F}, \mathcal{Y}\}$  are defined by (9.277) and (9.280). Then the network MSD and EMSE and the individual node performance measures are given by

$$\text{MSD}^{\text{network}} = h^T \cdot \text{vec}(I_{NM})/N, \quad (9.302)$$

$$\text{EMSE}^{\text{network}} = h^T \cdot \text{vec}(\mathcal{R}_u)/N, \quad (9.303)$$

$$\text{MSD}_k = h^T \cdot \text{vec}(\mathcal{J}_k), \quad (9.304)$$

$$\text{EMSE}_k = h^T \cdot \text{vec}(T_k), \quad (9.305)$$

where  $\{\mathcal{J}_k, T_k\}$  are defined by (9.294) and (9.298).  $\square$

We can recover from the above expressions the performance of the nodes in the non-cooperative implementation (9.207), where each node performs its adaptation individually, by setting  $A_1 = A_2 = C = I_N$ .

We can express the network MSD, and its EMSE if desired, in an alternative useful form involving a series representation.

**Corollary 9.6.2 (Series Representation for Network MSD).** *Consider the same setting of Theorem 9.6.6. The network MSD can be expressed in the following alternative series form:*

$$\boxed{\text{MSD}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr}(\mathcal{B}^j \mathcal{Y} \mathcal{B}^{*j})}, \quad (9.306)$$

where

$$\mathcal{Y} = \mathcal{G} \mathcal{S} \mathcal{G}^T, \quad (9.307)$$

$$\mathcal{G} = \mathcal{A}_2^T \mathcal{M} \mathcal{C}^T, \quad (9.308)$$

$$\mathcal{B} = \mathcal{A}_2^T (I - \mathcal{M} \mathcal{R}) \mathcal{A}_1^T. \quad (9.309)$$

**Proof.** Since  $\mathcal{F}$  is stable when the filter is mean-square stable, we can expand  $(I - \mathcal{F})^{-1}$  as

$$\begin{aligned} (I - \mathcal{F})^{-1} &= I + \mathcal{F} + \mathcal{F}^2 + \dots \\ &\stackrel{(9.277)}{=} I + (\mathcal{B}^T \otimes \mathcal{B}^*) + (\mathcal{B}^T \otimes \mathcal{B}^*)^2 + \dots \end{aligned}$$

Substituting into (9.287) and using property (9.274), we obtain the desired result.  $\square$

### 3.09.6.8 Uniform data profile

We can simplify expressions (9.307)–(9.309) for  $\{\mathcal{Y}, \mathcal{G}, \mathcal{B}\}$  in the case when the regression covariance matrices are uniform across the network and all nodes employ the same step-size, i.e., when

$$R_{u,k} = R_u, \quad \text{for all } k \quad (\text{uniform covariance profile}), \quad (9.310)$$

$$\mu_k = \mu, \quad \text{for all } k \quad (\text{uniform step-sizes}), \quad (9.311)$$

and when the combination matrix  $C$  is doubly stochastic, so that

$$C\mathbf{1} = \mathbf{1}, \quad C^T\mathbf{1} = \mathbf{1}. \quad (9.312)$$

We refer to conditions (9.310)–(9.312) as corresponding to a *uniform data profile* environment. The noise variances,  $\{\sigma_{v,k}^2\}$ , do not need to be uniform so that the signal-to-noise ratio (SNR) across the network can still vary from node to node. The simplified expressions derived in the sequel will be useful in Section 3.09.7 when we compare the performance of various cooperation strategies.

Thus, under conditions (9.310)–(9.312), expressions (9.180), (9.181), and (9.263) for  $\{\mathcal{M}, \mathcal{R}, \mathcal{G}\}$  simplify to

$$\mathcal{M} = \mu I_{NM}, \quad (9.313)$$

$$\mathcal{R} = I_N \otimes R_u, \quad (9.314)$$

$$\mathcal{G} = \mu \mathcal{A}_2^T \mathcal{C}^T. \quad (9.315)$$

Substituting these values into expression (9.309) for  $\mathcal{B}$  we get

$$\begin{aligned} \mathcal{B} &= \mathcal{A}_2^T (I - \mathcal{M}\mathcal{R}) \mathcal{A}_1^T \\ &= (A_2^T \otimes I) (I - \mu(I \otimes R_u)) (A_1^T \otimes I) \\ &= (A_2^T \otimes I) (A_1^T \otimes I) - \mu (A_2^T \otimes I) (I \otimes R_u) (A_1^T \otimes I) \\ &= (A_2^T A_1^T \otimes I) - \mu (A_2^T A_1^T \otimes R_u) \\ &= A_2^T A_1^T \otimes (I - \mu R_u), \end{aligned} \quad (9.316)$$

where we used the useful Kronecker product identities:

$$(X + Y) \otimes Z = (X \otimes Z) + (Y \otimes Z), \quad (9.317)$$

$$(X \otimes Y)(W \otimes Z) = (XW \otimes YZ), \quad (9.318)$$

for any matrices  $\{X, Y, Z, W\}$  of compatible dimensions. Likewise, introduce the  $N \times N$  diagonal matrix with noise variances:

$$R_v \triangleq \text{diag} \left\{ \sigma_{v,1}^2, \sigma_{v,2}^2, \dots, \sigma_{v,N}^2 \right\}. \quad (9.319)$$

Then, expression (9.241) for  $\mathcal{S}$  becomes

$$\begin{aligned} \mathcal{S} &= \text{diag} \left\{ \sigma_{v,1}^2 R_u, \sigma_{v,2}^2 R_u, \dots, \sigma_{v,N}^2 R_u \right\} \\ &= R_v \otimes R_u. \end{aligned} \quad (9.320)$$

It then follows that we can simplify expression (9.307) for  $\mathcal{Y}$  as:

$$\begin{aligned}\mathcal{Y} &= \mu^2 A_2^T C^T S C A_2 \\ &= \mu^2 \cdot (A_2^T \otimes I) \cdot (C^T \otimes I) \otimes (R_v \otimes R_u) \cdot (C \otimes I) \cdot (A_2 \otimes I) \\ &= \mu^2 \left( A_2^T C^T R_v C A_2 \otimes R_u \right).\end{aligned}\quad (9.321)$$

**Corollary 9.6.3 (Network MSD for Uniform Data Profile).** *Consider the same setting of Theorem 9.6.6 with the additional requirement that conditions (9.310)–(9.312) for a uniform data profile hold. The network MSD is still given by the same series representation (9.306) where now*

$$\mathcal{Y} = \mu^2 \left( A_2^T C^T R_v C A_2 \otimes R_u \right), \quad (9.322)$$

$$\mathcal{B} = A_2^T A_1^T \otimes (I - \mu R_u). \quad (9.323)$$

Using these expressions, we can decouple the network MSD expression (9.306) into two separate factors: one is dependent on the step-size and data covariance  $\{\mu, R_u\}$ , and the other is dependent on the combination matrices and noise profile  $\{A_1, A_2, C, R_v\}$ :

$$\text{MSD}^{\text{network}} = \frac{\mu^2}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \left[ \left( A_2^T A_1^T \right)^j \left( A_2^T C^T R_v C A_2 \right) (A_1 A_2)^j \right] \otimes [(I - \mu R_u)^j R_u (I - \mu R_u)^j] \right). \quad (9.324)$$

□

**Proof.** Using (9.306) and the given expressions (9.322) and (9.323) for  $\{\mathcal{Y}, \mathcal{B}\}$ , we get

$$\begin{aligned}\text{MSD}^{\text{network}} &= \frac{\mu^2}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \left[ \left( A_2^T A_1^T \right)^j \otimes (I - \mu R_u)^j \right] \times \left( A_2^T C^T R_v C A_2 \otimes R_u \right) \left[ (A_1 A_2)^j \otimes (I - \mu R_u)^j \right] \right).\end{aligned}$$

Result (9.324) follows from property (9.317). □

### 3.09.6.9 Transient mean-square performance

Before comparing the mean-square performance of various cooperation strategies, we pause to comment that the variance relation (9.279) can also be used to characterize the transient behavior of the network, and not just its steady-state performance. To see this, iterating (9.279) starting from  $i = 0$ , we find that

$$\mathbb{E} \|\tilde{w}_i\|_{\sigma}^2 = \mathbb{E} \|\tilde{w}_{-1}\|_{\mathcal{F}^{i+1}\sigma}^2 + [\text{vec}(\mathcal{Y}^T)]^T \cdot \left( \sum_{j=0}^i \mathcal{F}^j \sigma \right), \quad (9.325)$$

where

$$\tilde{\mathbf{w}}_{-1} \triangleq \mathbf{w}^o - \mathbf{w}_{-1} \quad (9.326)$$

in terms of the initial condition,  $\mathbf{w}_{-1}$ . If this initial condition happens to be  $\mathbf{w}_{-1} = 0$ , then  $\tilde{\mathbf{w}}_{-1} = \mathbf{w}^o$ . Comparing expression (9.325) at time instants  $i$  and  $i - 1$  we can relate  $\mathbb{E}\|\tilde{\mathbf{w}}_i\|_\sigma^2$  and  $\mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_\sigma^2$  as follows:

$$\boxed{\mathbb{E}\|\tilde{\mathbf{w}}_i\|_\sigma^2 = \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_\sigma^2 + [\text{vec}(\mathcal{Y}^T)]^T \cdot \mathcal{F}^i \sigma - \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{(I-\mathcal{F})\mathcal{F}^i\sigma}^2}. \quad (9.327)$$

This recursion relates the same weighted square measures of the error vectors  $\{\tilde{\mathbf{w}}_i, \tilde{\mathbf{w}}_{i-1}\}$ . It therefore describes how these weighted square measures evolve over time. It is clear from this relation that, for mean-square stability, the matrix  $\mathcal{F}$  needs to be stable so that the terms involving  $\mathcal{F}^i$  do not grow unbounded.

The learning curve of the network is the curve that describes the evolution of the network EMSE over time. At any time  $i$ , the network EMSE is denoted by  $\zeta(i)$  and measured as:

$$\begin{aligned} \zeta(i) &\triangleq \frac{1}{N} \sum_{k=1}^N \mathbb{E}|\mathbf{e}_{a,k}(i)|^2 \\ &= \frac{1}{N} \sum_{k=1}^N \mathbb{E}\|\tilde{\mathbf{w}}_{k,i}\|_{R_{u,k}}^2. \end{aligned} \quad (9.328)$$

The above expression indicates that  $\zeta(i)$  is obtained by averaging the EMSE of the individual nodes at time  $i$ . Therefore,

$$\zeta(i) = \frac{1}{N} \mathbb{E}\|\tilde{\mathbf{w}}_i\|_{\text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\}}^2 = \frac{1}{N} \mathbb{E}\|\tilde{\mathbf{w}}_i\|_{\mathcal{R}_u}^2, \quad (9.329)$$

where  $\mathcal{R}_u$  is the matrix defined by (9.290). This means that in order to evaluate the evolution of the network EMSE from relation (9.327), we simply select the weighting vector  $\sigma$  such that

$$\sigma = \frac{1}{N} \text{vec}(\mathcal{R}_u). \quad (9.330)$$

Substituting into (9.327) we arrive at the learning curve for the network.

**Corollary 9.6.4 (Network Learning Curve).** *Consider the same setting of Theorem 9.6.6. Let  $\zeta(i)$  denote the network EMSE at time  $i$ , as defined by (9.328). Then, the learning curve of the network corresponds to the evolution of  $\zeta(i)$  with time and is described by the following recursion over  $i \geq 0$ :*

$$\boxed{\zeta(i) = \zeta(i-1) + \frac{1}{N} [\text{vec}(\mathcal{Y}^T)]^T \cdot \mathcal{F}^i \cdot \text{vec}(\mathcal{R}_u) - \frac{1}{N} \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{(I-\mathcal{F})\mathcal{F}^i\text{vec}(\mathcal{R}_u)}^2}, \quad (9.331)$$

where  $\{\mathcal{F}, \mathcal{Y}, \mathcal{R}_u\}$  are defined by (9.277), (9.280), and (9.290).  $\square$

### 3.09.7 Comparing the performance of cooperative strategies

Using the expressions just derived for the MSD of the network, we can compare the performance of various cooperative and non-cooperative strategies. Table 9.6 further ahead summarizes the results derived in this section and the conditions under which they hold.

### 3.09.7.1 Comparing ATC and CTA strategies

We first compare the performance of the adaptive ATC and CTA diffusion strategies (9.153) and (9.154) when they employ a *doubly* stochastic combination matrix  $A$ . That is, let us consider the two scenarios:

$$C, \quad A_1 = A, \quad A_2 = I_N \quad (\text{adaptive CTA strategy}), \quad (9.332)$$

$$C, \quad A_1 = I_N, \quad A_2 = A \quad (\text{adaptive ATC strategy}), \quad (9.333)$$

where  $A$  is now assumed to be doubly stochastic, i.e.,

$$A\mathbb{1} = \mathbb{1}, \quad A^T\mathbb{1} = \mathbb{1} \quad (9.334)$$

with its rows and columns adding up to one. For example, these conditions are satisfied when  $A$  is left stochastic and symmetric. Then, expressions (9.307) and (9.309) give:

$$\mathcal{B}_{\text{cta}} = (I - \mathcal{M}\mathcal{R})\mathcal{A}^T, \quad \mathcal{Y}_{\text{cta}} = \mathcal{M}\mathcal{C}^T\mathcal{S}\mathcal{C}\mathcal{M}, \quad (9.335)$$

$$\mathcal{B}_{\text{atc}} = \mathcal{A}^T(I - \mathcal{M}\mathcal{R}), \quad \mathcal{Y}_{\text{atc}} = \mathcal{A}^T\mathcal{M}\mathcal{C}^T\mathcal{S}\mathcal{C}\mathcal{M}\mathcal{A}, \quad (9.336)$$

where

$$\mathcal{A} = A \otimes I_M. \quad (9.337)$$

Following [18], introduce the auxiliary nonnegative-definite matrix

$$\mathcal{H}_j \triangleq [(I - \mathcal{M}\mathcal{R})\mathcal{A}^T]^j \cdot \mathcal{M}\mathcal{C}^T\mathcal{S}\mathcal{C}\mathcal{M} \cdot [(I - \mathcal{M}\mathcal{R})\mathcal{A}^T]^{*j}. \quad (9.338)$$

Then, it is immediate to verify from (9.306) that

$$\text{MSD}_{\text{cta}}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr}(\mathcal{H}_j), \quad (9.339)$$

$$\text{MSD}_{\text{atc}}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr}(\mathcal{A}^T \mathcal{H}_j \mathcal{A}), \quad (9.340)$$

so that

$$\text{MSD}_{\text{cta}}^{\text{network}} - \text{MSD}_{\text{atc}}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr}(\mathcal{H}_j - \mathcal{A}^T \mathcal{H}_j \mathcal{A}). \quad (9.341)$$

Now, since  $A$  is doubly stochastic, it also holds that the enlarged matrix  $\mathcal{A}$  is doubly stochastic. Moreover, for any doubly stochastic matrix  $\mathcal{A}$  and any nonnegative-definite matrix  $\mathcal{H}$  of compatible dimensions, it holds that (see part (f) of Theorem C.3):

$$\text{Tr}(\mathcal{A}^T \mathcal{H} \mathcal{A}) \leq \text{Tr}(\mathcal{H}). \quad (9.342)$$

Applying result (9.342) to (9.341) we conclude that

$\text{MSD}_{\text{atc}}^{\text{network}} \leq \text{MSD}_{\text{cta}}^{\text{network}}$	(doubly stochastic $A$ ),	(9.343)
------------------------------------------------------------------------------------------	---------------------------	---------

so that the adaptive ATC strategy (9.153) outperforms the adaptive CTA strategy (9.154) for doubly stochastic combination matrices  $A$ .

### 3.09.7.2 Comparing strategies with and without information exchange

We now examine the effect of information exchange ( $C \neq I$ ) on the performance of the adaptive ATC and CTA diffusion strategies (9.153) and (9.154) under conditions (9.310)–(9.312) for *uniform data profile*.

#### 3.09.7.2.1 CTA Strategies

We start with the adaptive CTA strategy (9.154), and consider two scenarios with and without information exchange. These scenarios correspond to the following selections in the general description (9.201)–(9.203):

$$C \neq I, \quad A_1 = A, \quad A_2 = I_N \quad (\text{adaptive CTA with information exchange}), \quad (9.344)$$

$$C = I, \quad A_1 = A, \quad A_2 = I_N \quad (\text{adaptive CTA without information exchange}). \quad (9.345)$$

Then, expressions (9.322) and (9.323) give:

$$\mathcal{B}_{\text{cta}, C \neq I} = A^T \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{cta}, C \neq I} = \mu^2 (C^T R_v C \otimes R_u), \quad (9.346)$$

$$\mathcal{B}_{\text{cta}, C=I} = A^T \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{cta}, C=I} = \mu^2 (R_v \otimes R_u), \quad (9.347)$$

where the matrix  $R_v$  is defined by (9.319). Note that  $\mathcal{B}_{\text{cta}, C \neq I} = \mathcal{B}_{\text{cta}, C=I}$ , so we denote them simply by  $\mathcal{B}$  in the derivation that follows. Then, from expression (9.306) for the network MSD we get:

$$\text{MSD}_{\text{cta}, C=I}^{\text{network}} - \text{MSD}_{\text{cta}, C \neq I}^{\text{network}} = \frac{\mu^2}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}^j \left[ (R_v - C^T R_v C) \otimes R_u \right] \mathcal{B}^{*j} \right). \quad (9.348)$$

It follows that the difference in performance between both CTA implementations depends on how the matrices  $R_v$  and  $C^T R_v C$  compare to each other:

- When  $R_v - C^T R_v C \geq 0$ , we obtain

$\text{MSD}_{\text{cta}, C=I}^{\text{network}} \geq \text{MSD}_{\text{cta}, C \neq I}^{\text{network}}$

(when  $C^T R_v C \leq R_v$ ), (9.349)

so that a CTA implementation with information exchange performs better than a CTA implementation without information exchange. Note that the condition on  $\{R_v, C\}$  corresponds to requiring

$$C^T R_v C \leq R_v, \quad (9.350)$$

which can be interpreted to mean that the cooperation matrix  $C$  should be such that it does not amplify the effect of measurement noise. For example, this situation occurs when the noise profile is uniform across the network, in which case  $R_v = \sigma_v^2 I_M$ . This is because it would then hold that

$$R_v - C^T R_v C = \sigma_v^2 (I - C^T C) \geq 0 \quad (9.351)$$

in view of the fact that  $(I - C^T C) \geq 0$  since  $C$  is doubly stochastic (cf. property (e) in Lemma C.3).

2. When  $R_v - C^T R_v C \leq 0$ , we obtain

$$\boxed{\text{MSD}_{\text{cta}, C=I}^{\text{network}} \leq \text{MSD}_{\text{cta}, C \neq I}^{\text{network}}} \quad (\text{when } C^T R_v C \geq R_v), \quad (9.352)$$

so that a CTA implementation without information exchange performs better than a CTA implementation with information exchange. In this case, the condition on  $\{R_v, C\}$  indicates that the combination matrix  $C$  ends up amplifying the effect of noise.

### 3.09.7.2.2 ATC Strategies

We can repeat the argument for the adaptive ATC strategy (9.153), and consider two scenarios with and without information exchange. These scenarios correspond to the following selections in the general description (9.201)–(9.203):

$$C \neq I, \quad A_1 = I_N, \quad A_2 = A \quad (\text{adaptive ATC with information exchange}), \quad (9.353)$$

$$C = I, \quad A_1 = I_N, \quad A_2 = A \quad (\text{adaptive ATC without information exchange}). \quad (9.354)$$

Then, expressions (9.322) and (9.323) give:

$$\mathcal{B}_{\text{atc}, C \neq I} = A^T \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{atc}, C \neq I} = \mu^2 (A^T C^T R_v C A \otimes R_u), \quad (9.355)$$

$$\mathcal{B}_{\text{atc}, C=I} = A^T \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{atc}, C=I} = \mu^2 (A^T R_v A \otimes R_u). \quad (9.356)$$

Note again that  $\mathcal{B}_{\text{atc}, C \neq I} = \mathcal{B}_{\text{atc}, C=I}$ , so we denote them simply by  $\mathcal{B}$ . Then,

$$\text{MSD}_{\text{atc}, C=I}^{\text{network}} - \text{MSD}_{\text{atc}, C \neq I}^{\text{network}} = \frac{\mu^2}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}^j \left[ A^T (R_v - C^T R_v C) A \otimes R_u \right] \mathcal{B}^{*j} \right). \quad (9.357)$$

It again follows that the difference in performance between both ATC implementations depends on how the matrices  $R_v$  and  $C^T R_v C$  compare to each other and we obtain:

$$\boxed{\text{MSD}_{\text{atc}, C=I}^{\text{network}} \geq \text{MSD}_{\text{atc}, C \neq I}^{\text{network}}} \quad (\text{when } C^T R_v C \leq R_v) \quad (9.358)$$

and

$$\boxed{\text{MSD}_{\text{atc}, C=I}^{\text{network}} \leq \text{MSD}_{\text{atc}, C \neq I}^{\text{network}}} \quad (\text{when } C^T R_v C \geq R_v). \quad (9.359)$$

### 3.09.7.3 Comparing diffusion strategies with the non-cooperative strategy

We now compare the performance of the adaptive CTA strategy (9.154) to the non-cooperative LMS strategy (9.207) assuming conditions (9.310)–(9.312) for uniform data profile. These scenarios correspond to the following selections in the general description (9.201)–(9.203):

$$C, \quad A_1 = A, \quad A_2 = I \quad (\text{adaptive CTA}), \quad (9.360)$$

$$C = I, \quad A_1 = I, \quad A_2 = I \quad (\text{non-cooperative LMS}), \quad (9.361)$$

where  $A$  is further assumed to be doubly stochastic (along with  $C$ ) so that

$$A\mathbb{1} = \mathbb{1}, \quad A^T\mathbb{1} = \mathbb{1}. \quad (9.362)$$

Then, expressions (9.322) and (9.323) give:

$$\mathcal{B}_{\text{cta}} = A^T \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{cta}} = \mu^2 \left( C^T R_v C \otimes R_u \right), \quad (9.363)$$

$$\mathcal{B}_{\text{lms}} = I \otimes (I - \mu R_u), \quad \mathcal{Y}_{\text{lms}} = \mu^2 (R_v \otimes R_u). \quad (9.364)$$

Now recall that

$$\mathcal{C} = C \otimes I_M, \quad (9.365)$$

so that, using the Kronecker product property (9.317),

$$\begin{aligned} \mathcal{Y}_{\text{cta}} &= \mu^2 (C^T R_v C \otimes R_u) \\ &= \mu^2 (C^T \otimes I_M) (R_v \otimes R_u) (C \otimes I_M) \\ &= \mu^2 \mathcal{C}^T (R_v \otimes R_u) \mathcal{C} \\ &= \mathcal{C}^T \mathcal{Y}_{\text{lms}} \mathcal{C}. \end{aligned} \quad (9.366)$$

Then,

$$\begin{aligned} \text{MSD}_{\text{lms}}^{\text{network}} - \text{MSD}_{\text{cta}}^{\text{network}} &= \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}_{\text{lms}}^j \mathcal{Y}_{\text{lms}} \mathcal{B}_{\text{lms}}^{*j} \right) - \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}_{\text{cta}}^j \mathcal{C}^T \mathcal{Y}_{\text{lms}} \mathcal{C} \mathcal{B}_{\text{cta}}^{*j} \right) \\ &= \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}_{\text{lms}}^{*j} \mathcal{B}_{\text{lms}}^j \mathcal{Y}_{\text{lms}} \right) - \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{C} \mathcal{B}_{\text{cta}}^{*j} \mathcal{B}_{\text{cta}}^j \mathcal{C}^T \mathcal{Y}_{\text{lms}} \right) \\ &= \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left[ \left( \mathcal{B}_{\text{lms}}^{*j} \mathcal{B}_{\text{lms}}^j - \mathcal{C} \mathcal{B}_{\text{cta}}^{*j} \mathcal{B}_{\text{cta}}^j \mathcal{C}^T \right) \mathcal{Y}_{\text{lms}} \right]. \end{aligned} \quad (9.367)$$

Let us examine the difference:

$$\begin{aligned} \mathcal{B}_{\text{lms}}^{*j} \mathcal{B}_{\text{lms}}^j - \mathcal{C} \mathcal{B}_{\text{cta}}^{*j} \mathcal{B}_{\text{cta}}^j \mathcal{C}^T &= (I \otimes (I - \mu R_u)^{2j}) - (CA^j \otimes (I - \mu R_u)^j)(A^{jT} C^T \otimes (I - \mu R_u)^j) \\ &\stackrel{(9.317)}{=} (I \otimes (I - \mu R_u)^{2j}) - (CA^j A^{jT} C^T \otimes (I - \mu R_u)^{2j}) \\ &= (I - CA^j A^{jT} C^T) \otimes (I - \mu R_u)^{2j}. \end{aligned} \quad (9.368)$$

Now, due to the even power, it always holds that  $(I - \mu R_u)^{2j} \geq 0$ . Moreover, since  $A^j$  and  $C$  are doubly stochastic, it follows that  $CA^j A^{jT} C^T$  is also doubly stochastic. Therefore, the matrix  $(I - CA^j A^{jT} C^T)$  is nonnegative-definite as well (cf. property (e) of Lemma C.3). It follows that

$$\mathcal{B}_{\text{lms}}^{*j} \mathcal{B}_{\text{lms}}^j - \mathcal{C} \mathcal{B}_{\text{cta}}^{*j} \mathcal{B}_{\text{cta}}^j \mathcal{C}^T \geq 0. \quad (9.369)$$

But since  $\mathcal{Y}_{\text{lms}} \geq 0$ , we conclude from (9.367) that

$\boxed{\text{MSD}_{\text{lms}}^{\text{network}} \geq \text{MSD}_{\text{cta}}^{\text{network}}}. \quad (9.370)$

This is because for any two Hermitian nonnegative-definite matrices  $A$  and  $B$  of compatible dimensions, it holds that  $\text{Tr}(AB) \geq 0$ ; indeed if we factor  $B = XX^*$  with  $X$  full rank, then  $\text{Tr}(AB) = \text{Tr}(X^*AX) \geq 0$ . We conclude from this analysis that adaptive CTA diffusion performs better than non-cooperative LMS under uniform data profile conditions *and* doubly stochastic  $A$ . If we refer to the earlier result (9.343), we conclude that the following relation holds:

$$\boxed{\text{MSD}_{\text{atc}}^{\text{network}} \leq \text{MSD}_{\text{cta}}^{\text{network}} \leq \text{MSD}_{\text{lms}}^{\text{network}}} . \quad (9.371)$$

Table 9.6 lists the comparison results derived in this section and lists the conditions under which the conclusions hold.

**Table 9.6** Comparison of the MSD Performance of Various Cooperative Strategies

Comparison	Conditions
$\text{MSD}_{\text{atc}}^{\text{network}} \leq \text{MSD}_{\text{cta}}^{\text{network}}$	$A$ doubly stochastic, $C$ right stochastic
$\text{MSD}_{\text{cta}, C \neq I}^{\text{network}} \leq \text{MSD}_{\text{cta}, C = I}^{\text{network}}$	$C^T R_V C \leq R_V$ , $C$ doubly stochastic, $R_{U,k} = R_U$ , $\mu_k = \mu$
$\text{MSD}_{\text{cta}, C = I}^{\text{network}} \leq \text{MSD}_{\text{cta}, C \neq I}^{\text{network}}$	$C^T R_V C \geq R_V$ , $C$ doubly stochastic, $R_{U,k} = R_U$ , $\mu_k = \mu$
$\text{MSD}_{\text{atc}, C \neq I}^{\text{network}} \leq \text{MSD}_{\text{atc}, C = I}^{\text{network}}$	$C^T R_V C \leq R_V$ , $C$ doubly stochastic, $R_{U,k} = R_U$ , $\mu_k = \mu$
$\text{MSD}_{\text{atc}, C = I}^{\text{network}} \leq \text{MSD}_{\text{atc}, C \neq I}^{\text{network}}$	$C^T R_V C \geq R_V$ , $C$ doubly stochastic, $R_{U,k} = R_U$ , $\mu_k = \mu$
$\text{MSD}_{\text{atc}}^{\text{network}} \leq \text{MSD}_{\text{cta}}^{\text{network}} \leq \text{MSD}_{\text{lms}}^{\text{network}}$	$\{A, C\}$ doubly stochastic, $R_{U,k} = R_U$ , $\mu_k = \mu$

### 3.09.8 Selecting the combination weights

The adaptive diffusion strategy (9.201)–(9.203) employs combination weights  $\{a_{1,\ell k}, a_{2,\ell k}, c_{\ell k}\}$  or, equivalently, combination matrices  $\{A_1, A_2, C\}$ , where  $A_1$  and  $A_2$  are left-stochastic matrices and  $C$  is a right-stochastic matrix. There are several ways by which these matrices can be selected. In this section, we describe constructions that result in left-stochastic or doubly-stochastic combination matrices,  $A$ . When a right-stochastic combination matrix is needed, such as  $C$ , then it can be obtained by transposition of the left-stochastic constructions shown below.

### 3.09.8.1 Constant combination weights

Table 9.7 lists a couple of common choices for selecting constant combination weights for a network with  $N$  nodes. Several of these constructions appeared originally in the literature on graph theory. In the table, the symbol  $n_k$  denotes the degree of node  $k$ , which refers to the size of its neighborhood.

**Table 9.7** Selections for Combination Matrices  $A = [a_{\ell k}]$

Entries of Combination Matrix $A$	Type of $A$
<b>1. Averaging rule [56]:</b> $a_{\ell k} = \begin{cases} 1/n_k, & \text{if } k \neq \ell \text{ are neighbors or } k = \ell \\ 0, & \text{otherwise} \end{cases}$	left-stochastic
<b>2. Laplacian rule [50,57]:</b> $A = I_N - \gamma \mathcal{L}, \gamma > 0$	symmetric and doubly-stochastic
<b>3. Laplacian rule</b> using $\gamma = 1/n_{\max}$ : $a_{\ell k} = \begin{cases} 1/n_{\max}, & \text{if } k \neq \ell \text{ are neighbors} \\ 1 - (n_k - 1)/n_{\max}, & k = \ell \\ 0, & \text{otherwise} \end{cases}$	symmetric and doubly-stochastic
<b>4. Laplacian rule</b> using $\gamma = 1/N$ (maximum-degree rule [51]): $a_{\ell k} = \begin{cases} 1/N, & \text{if } k \neq \ell \text{ are neighbors} \\ 1 - (n_k - 1)/N, & k = \ell \\ 0, & \text{otherwise} \end{cases}$	symmetric and doubly-stochastic
<b>5. Metropolis rule [50,58,59]:</b> $a_{\ell k} = \begin{cases} 1/\max\{n_k, n_\ell\}, & \text{if } k \neq \ell \text{ are neighbors} \\ 1 - \sum_{m \in \mathcal{N}_k \setminus \{k\}} a_{mk}, & k = \ell \\ 0, & \text{otherwise} \end{cases}$	symmetric and doubly-stochastic
<b>6. Relative-degree rule [39]:</b> $a_{\ell k} = \begin{cases} n_\ell / \left( \sum_{m \in \mathcal{N}_k} n_m \right), & \text{if } k \text{ and } \ell \text{ are neighbors or } k = \ell \\ 0, & \text{otherwise} \end{cases}$	left-stochastic

Likewise, the symbol  $n_{\max}$  refers to the maximum degree across the network, i.e.,

$$n_{\max} = \max_{1 \leq k \leq N} \{n_k\}. \quad (9.372)$$

The Laplacian rule, which appears in the second line of the table, relies on the use of the Laplacian matrix  $\mathcal{L}$  of the network and a positive scalar  $\gamma$ . The Laplacian matrix is defined by (9.574) in Appendix B, namely, it is a symmetric matrix whose entries are constructed as follows [60–62]:

$$[\mathcal{L}]_{k\ell} = \begin{cases} n_k - 1, & \text{if } k = \ell, \\ -1, & \text{if } k \neq \ell \text{ and nodes } k \text{ and } \ell \text{ are neighbors,} \\ 0, & \text{otherwise.} \end{cases} \quad (9.373)$$

The Laplacian rule can be reduced to other forms through the selection of the positive parameter  $\gamma$ . One choice is  $\gamma = 1/n_{\max}$ , while another choice is  $\gamma = 1/N$  and leads to the maximum-degree rule. Obviously, it always holds that  $n_{\max} \leq N$  so that  $1/n_{\max} \geq 1/N$ . Therefore, the choice  $\gamma = 1/n_{\max}$  ends up assigning larger weights to neighbors than the choice  $\gamma = 1/N$ . The averaging rule in the first row of the table is one of the simplest combination rules whereby nodes simply average data from their neighbors.

In the constructions in Table 9.7, the values of the weights  $\{a_{\ell k}\}$  are largely dependent on the degree of the nodes. In this way, the number of connections that each node has influences the combination weights with its neighbors. While such selections may be appropriate in some applications, they can nevertheless degrade the performance of adaptation over networks [63]. This is because such weighting schemes ignore the noise profile across the network. And since some nodes can be noisier than others, it is not sufficient to rely solely on the amount of connectivity that nodes have to determine the combination weights to their neighbors. It is important to take into account the amount of noise that is present at the nodes as well. Therefore, designing combination rules that are aware of the variation in noise profile across the network is an important task. It is also important to devise strategies that are able to *adapt* these combination weights in response to variations in network topology and data statistical profile. For this reason, following [64, 65], we describe in the next subsection one adaptive procedure for adjusting the combination weights. This procedure allows the network to assign more or less relevance to nodes according to the quality of their data.

### 3.09.8.2 Optimizing the combination weights

Ideally, we would like to select  $N \times N$  combination matrices  $\{A_1, A_2, C\}$  in order to minimize the network MSD given by (9.302) or (9.306). In [18], the selection of the combination weights was formulated as the following optimization problem:

$$\min_{\{A_1, A_2, C\}} \text{MSD}^{\text{network}} \text{ given by (9.302) or (9.306)}$$

over left and right-stochastic matrices with nonnegative entries:

$$A_1^T \mathbb{1} = \mathbb{1}, \quad a_{1,\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k,$$

$$A_2^T \mathbb{1} = \mathbb{1}, \quad a_{2,\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k,$$

$$C \mathbb{1} = \mathbb{1}, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k.$$

(9.374)

We can pursue a numerical solution to (9.374) in order to search for optimal combination matrices, as was done in [18]. Here, however, we are interested in an adaptive solution that becomes part of the learning process so that the network can adapt the weights on the fly in response to network conditions. We illustrate an approximate approach from [64, 65] that leads to one adaptive solution that performs reasonably well in practice.

We illustrate the construction by considering the ATC strategy (9.158) without information exchange where  $A_1 = I_N$ ,  $A_2 = A$ , and  $C = I$ . In this case, recursions (9.204)–(9.206) take the form:

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} w_{k,i-1}], \quad (9.375)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}, \quad (9.376)$$

and, from (9.306), the corresponding network MSD performance is:

$$\text{MSD}_{\text{atc}}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}_{\text{atc}}^j \mathcal{Y}_{\text{atc}} \mathcal{B}_{\text{atc}}^{*j} \right), \quad (9.377)$$

where

$$\mathcal{B}_{\text{atc}} = \mathcal{A}^T (I - \mathcal{M} \mathcal{R}_u), \quad (9.378)$$

$$\mathcal{Y}_{\text{atc}} = \mathcal{A}^T \mathcal{M} \mathcal{S} \mathcal{M} \mathcal{A}, \quad (9.379)$$

$$\mathcal{R}_u = \text{diag}\{R_{u,1}, R_{u,2}, \dots, R_{u,N}\}, \quad (9.380)$$

$$\mathcal{S} = \text{diag}\{\sigma_{v,1}^2 R_{u,1}, \sigma_{v,2}^2 R_{u,2}, \dots, \sigma_{v,N}^2 R_{u,N}\}, \quad (9.381)$$

$$\mathcal{M} = \text{diag}\{\mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M\}, \quad (9.382)$$

$$\mathcal{A} = A \otimes I_M. \quad (9.383)$$

Minimizing the MSD expression (9.377) over left-stochastic matrices  $A$  is generally non-trivial. We pursue an approximate solution.

To begin with, for compactness of notation, let  $r$  denote the spectral radius of the  $N \times N$  block matrix  $I - \mathcal{M} \mathcal{R}_u$ :

$$r \triangleq \rho(I - \mathcal{M} \mathcal{R}_u). \quad (9.384)$$

We already know, in view of the mean and mean-square stability of the network, that  $|r| < 1$ . Now, consider the series that appears in (9.377) and whose trace we wish to minimize over  $A$ . Note that its block maximum norm can be bounded as follows:

$$\begin{aligned} \left\| \sum_{j=0}^{\infty} \mathcal{B}_{\text{atc}}^j \mathcal{Y}_{\text{atc}} \mathcal{B}_{\text{atc}}^{*j} \right\|_{b,\infty} &\leq \sum_{j=0}^{\infty} \left\| \mathcal{B}_{\text{atc}}^j \right\|_{b,\infty} \cdot \|\mathcal{Y}_{\text{atc}}\|_{b,\infty} \cdot \left\| \mathcal{B}_{\text{atc}}^{*j} \right\|_{b,\infty} \\ &\stackrel{(a)}{\leq} N \cdot \left( \sum_{j=0}^{\infty} \left\| \mathcal{B}_{\text{atc}}^j \right\|_{b,\infty}^2 \cdot \|\mathcal{Y}_{\text{atc}}\|_{b,\infty} \right) \end{aligned}$$

$$\begin{aligned}
 &\leq N \cdot \left( \sum_{j=0}^{\infty} \|\mathcal{B}_{\text{atc}}\|_{b,\infty}^{2j} \cdot \|\mathcal{Y}_{\text{atc}}\|_{b,\infty} \right) \\
 &\stackrel{(b)}{\leq} N \cdot \left( \sum_{j=0}^{\infty} r^{2j} \cdot \|\mathcal{Y}_{\text{atc}}\|_{b,\infty} \right) \\
 &= \frac{N}{1-r^2} \cdot \|\mathcal{Y}_{\text{atc}}\|_{b,\infty},
 \end{aligned} \tag{9.385}$$

where for step (b) we use result (9.602) to conclude that

$$\begin{aligned}
 \|\mathcal{B}_{\text{atc}}\|_{b,\infty} &= \left\| \mathcal{A}^T (I - \mathcal{M}\mathcal{R}_u) \right\|_{b,\infty} \\
 &\leq \left\| \mathcal{A}^T \right\|_{b,\infty} \cdot \left\| I - \mathcal{M}\mathcal{R}_u \right\|_{b,\infty} \\
 &= \|I - \mathcal{M}\mathcal{R}_u\|_{b,\infty} \\
 &\stackrel{(9.602)}{=} r.
 \end{aligned} \tag{9.386}$$

To justify step (a), we use result (9.584) to relate the norms of  $\mathcal{B}_{\text{atc}}^j$  and its complex conjugate,  $\left[\mathcal{B}_{\text{atc}}^j\right]^*$ , as

$$\left\| \mathcal{B}_{\text{atc}}^{*j} \right\|_{b,\infty} \leq N \cdot \left\| \mathcal{B}_{\text{atc}}^j \right\|_{b,\infty}. \tag{9.387}$$

Expression (9.385) then shows that the norm of the series appearing in (9.377) is bounded by a scaled multiple of the norm of  $\mathcal{Y}_{\text{atc}}$ , and the scaling constant is independent of  $A$ . Using property (9.586) we conclude that there exists a positive constant  $c$ , also independent of  $A$ , such that

$$\text{Tr} \left( \sum_{j=0}^{\infty} \mathcal{B}_{\text{atc}}^j \mathcal{Y}_{\text{atc}} \mathcal{B}_{\text{atc}}^{*j} \right) \leq c \cdot \text{Tr}(\mathcal{Y}_{\text{atc}}). \tag{9.388}$$

Therefore, instead of attempting to minimize the trace of the series, the above result motivates us to minimize an upper bound to the trace. Thus, we consider the alternative problem of minimizing the first term of the series (9.377), namely,

$$\begin{aligned}
 \min_A \quad &\text{Tr}(\mathcal{Y}_{\text{atc}}) \\
 \text{subject to } &A^T \mathbb{1} = \mathbb{1}, \quad a_{\ell k} \geq 0, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k.
 \end{aligned} \tag{9.389}$$

Using (9.379), the trace of  $\mathcal{Y}_{\text{atc}}$  can be expressed in terms of the combination coefficients as follows:

$$\text{Tr}(\mathcal{Y}_{\text{atc}}) = \sum_{k=1}^N \sum_{\ell=1}^N \mu_\ell^2 a_{\ell k}^2 \sigma_{v,\ell}^2 \text{Tr}(R_{u,\ell}), \tag{9.390}$$

so that problem (9.389) can be decoupled into  $N$  separate optimization problems of the form:

$$\boxed{\begin{aligned} \min_{\{a_{\ell k}\}_{\ell=1}^N} & \sum_{\ell=1}^N \mu_{\ell}^2 a_{\ell k} \sigma_{v,\ell}^2 \text{Tr}(R_{u,\ell}), \quad k = 1, \dots, N \\ \text{subject to} & \\ a_{\ell k} \geq 0, \quad & \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \end{aligned}}. \quad (9.391)$$

With each node  $\ell$ , we associate the following nonnegative noise-data-dependent measure:

$$\boxed{\gamma_{\ell}^2 \triangleq \mu_{\ell}^2 \sigma_{v,\ell}^2 \text{Tr}(R_{u,\ell})}. \quad (9.392)$$

This measure amounts to scaling the noise variance at node  $\ell$  by  $\mu_{\ell}^2$  and by the power of the regression data (measured through the trace of its covariance matrix). We shall refer to  $\gamma_{\ell}^2$  as the noise-data variance product (or *variance product*, for simplicity) at node  $\ell$ . Then, the solution of (9.391) is given by:

$$\boxed{a_{\ell k} = \begin{cases} \frac{\gamma_{\ell}^{-2}}{\sum_{m \in \mathcal{N}_k} \gamma_m^{-2}}, & \text{if } \ell \in \mathcal{N}_k \\ 0, & \text{otherwise} \end{cases}} \quad (\text{relative-variance rule}). \quad (9.393)$$

We refer to this combination rule as the *relative-variance combination rule* [64]; it leads to a left-stochastic matrix  $A$ . In this construction, node  $k$  combines the intermediate estimates  $\{\psi_{\ell,i}\}$  from its neighbors in (9.376) in proportion to the inverses of their variance products,  $\{\gamma_m^{-2}\}$ . The result is physically meaningful. Nodes with smaller variance products will generally be given larger weights. In comparison, the following *relative-degree-variance rule* was proposed in [18] (a typo appears in Table III in [18], where the noise variances appear written in the table instead of their inverses):

$$\boxed{a_{\ell k} = \begin{cases} \frac{n_{\ell} \sigma_{v,\ell}^{-2}}{\sum_{m \in \mathcal{N}_k} n_m \sigma_{v,m}^{-2}}, & \text{if } \ell \in \mathcal{N}_k \\ 0, & \text{otherwise} \end{cases}} \quad (\text{relative degree-variance rule}). \quad (9.394)$$

This second form also leads to a left-stochastic combination matrix  $A$ . However, rule (9.394) does not take into account the covariance matrices of the regression data across the network. Observe that in the special case when the step-sizes, regression covariance matrices, and noise variances are uniform across the network, i.e.,  $\mu_k = \mu$ ,  $R_{u,k} = R_u$  and  $\sigma_{v,k}^2 = \sigma_v^2$  for all  $k$ , expression (9.393) reduces to the simple averaging rule (first line of Table 9.7). In contrast, expression (9.394) reduces the relative degree rule (last line of Table 9.7).

### 3.09.8.3 Adaptive combination weights

To evaluate the combination weights (9.393), the nodes need to know the variance products,  $\{\gamma_m^2\}$ , of their neighbors. According to (9.392), the factors  $\{\gamma_m^2\}$  are defined in terms of the noise variances,  $\{\sigma_{v,m}^2\}$ ,

and the regression covariance matrices,  $\{\text{Tr}(R_{u,m})\}$ , and these quantities are not known beforehand. The nodes only have access to realizations of  $\{\mathbf{d}_m(i), \mathbf{u}_{m,i}\}$ . We now describe one procedure that allows every node  $k$  to learn the variance products of its neighbors in an adaptive manner. Note that if a particular node  $\ell$  happens to belong to two neighborhoods, say, the neighborhood of node  $k_1$  and the neighborhood of node  $k_2$ , then each of  $k_1$  and  $k_2$  need to evaluate the variance product,  $\gamma_\ell^2$ , of node  $\ell$ . The procedure described below allows each node in the network to estimate the variance products of its neighbors in a recursive manner.

To motivate the algorithm, we refer to the ATC recursion (9.375), (9.376) and use the data model (9.208) to write for node  $\ell$ :

$$\boldsymbol{\psi}_{\ell,i} = \mathbf{w}_{\ell,i-1} + \mu_\ell \mathbf{u}_{\ell,i}^* [\mathbf{u}_{\ell,i} \tilde{\mathbf{w}}_{\ell,i-1} + \mathbf{v}_\ell(i)], \quad (9.395)$$

so that, in view of our earlier assumptions on the regression data and noise in Section 3.09.6.1, we obtain in the limit as  $i \rightarrow \infty$ :

$$\lim_{i \rightarrow \infty} \mathbb{E} \|\boldsymbol{\psi}_{\ell,i} - \mathbf{w}_{\ell,i-1}\|^2 = \mu_\ell^2 \cdot \left( \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{\ell,i-1}\|^2 \right)_{\mathbb{E}(\mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i})} + \mu_\ell^2 \cdot \sigma_{v,\ell}^2 \cdot \text{Tr}(R_{u,\ell}). \quad (9.396)$$

We can evaluate the limit on the right-hand side by using the steady-state result (9.284). Indeed, we select the vector  $\sigma$  in (9.284) to satisfy

$$(I - \mathcal{F})\sigma = \text{vec} \left[ \mathbb{E} \left( \mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i} \right) \right]. \quad (9.397)$$

Then, from (9.284),

$$\lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{\ell,i-1}\|^2_{\mathbb{E}(\mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i})} = [\text{vec}(\mathcal{Y}^T)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec} \left[ \mathbb{E} \left( \mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i} \right) \right]. \quad (9.398)$$

Now recall from expression (9.379) for  $\mathcal{Y}$  that for the ATC algorithm under consideration we have

$$\mathcal{Y} = \mathcal{A}^T \mathcal{M} \mathcal{S} \mathcal{M} \mathcal{A}, \quad (9.399)$$

so that the entries of  $\mathcal{Y}$  depend on combinations of the squared step-sizes,  $\{\mu_m^2, m = 1, 2, \dots, N\}$ . This fact implies that the first term on the right-hand side of (9.396) depends on products of the form  $\{\mu_\ell^2 \mu_m^2\}$ ; these fourth-order factors can be ignored in comparison to the second-order factor  $\mu_\ell^2$  for small step-sizes so that

$$\begin{aligned} \lim_{i \rightarrow \infty} \mathbb{E} \|\boldsymbol{\psi}_{\ell,i} - \mathbf{w}_{\ell,i-1}\|^2 &\approx \mu_\ell^2 \cdot \sigma_{v,\ell}^2 \cdot \text{Tr}(R_{u,\ell}) \\ &= \gamma_\ell^2 \end{aligned} \quad (9.400)$$

in terms of the desired variance product,  $\gamma_\ell^2$ . Using the following instantaneous approximation at node  $k$  (where  $w_{\ell,i-1}$  is replaced by  $w_{k,i-1}$ ):

$$\mathbb{E} \|\boldsymbol{\psi}_{\ell,i} - \mathbf{w}_{\ell,i-1}\|^2 \approx \|\boldsymbol{\psi}_{\ell,i} - w_{k,i-1}\|^2. \quad (9.401)$$

We can now motivate an algorithm that enables node  $k$  to estimate the variance product,  $\gamma_\ell^2$ , of its neighbor  $\ell$ . Thus, let  $\gamma_{\ell k}^2(i)$  denote an estimate for  $\gamma_\ell^2$  that is computed by node  $k$  at time  $i$ . Then, one way to evaluate  $\gamma_{\ell k}^2(i)$  is through the recursion:

$$\boxed{\gamma_{\ell k}^2(i) = (1 - v_k) \cdot \gamma_{\ell k}^2(i - 1) + v_k \cdot \|\psi_{\ell,i} - w_{k,i-1}\|^2}, \quad (9.402)$$

where  $0 < v_k \ll 1$  is a positive coefficient smaller than one. Note that under expectation, expression (9.402) becomes

$$\mathbb{E}\gamma_{\ell k}^2(i) = (1 - v_k) \cdot \mathbb{E}\gamma_{\ell k}^2(i - 1) + v_k \mathbb{E}\|\psi_{\ell,i} - w_{k,i-1}\|^2, \quad (9.403)$$

so that in steady-state, as  $i \rightarrow \infty$ ,

$$\lim_{i \rightarrow \infty} \mathbb{E}\gamma_{\ell k}^2(i) \approx (1 - v_k) \cdot \lim_{i \rightarrow \infty} \mathbb{E}\gamma_{\ell k}^2(i - 1) + v_k \gamma_\ell^2. \quad (9.404)$$

Hence, we obtain

$$\lim_{i \rightarrow \infty} \mathbb{E}\gamma_{\ell k}^2(i) \approx \gamma_\ell^2. \quad (9.405)$$

That is, the estimator  $\gamma_{\ell k}^2(i)$  converges on average close to the desired variance product  $\gamma_\ell^2$ . In this way, we can replace the optimal weights (9.393) by the adaptive construction:

$$\boxed{a_{\ell k}(i) = \begin{cases} \frac{\gamma_{\ell k}^{-2}(i)}{\sum_{m \in \mathcal{N}_k} \gamma_{mk}^{-2}(i)}, & \text{if } \ell \in \mathcal{N}_k \\ 0, & \text{otherwise} \end{cases}}. \quad (9.406)$$

Equations (9.402) and (9.406) provide one adaptive construction for the combination weights  $\{a_{\ell k}\}$ .

### 3.09.9 Diffusion with noisy information exchanges

The adaptive diffusion strategy (9.201)–(9.203) relies on the fusion of local information collected from neighborhoods through the use of combination matrices  $\{A_1, A_2, C\}$ . In the previous section, we described several constructions for selecting such combination matrices. We also motivated and developed an adaptive scheme for the ATC mode of operation (9.375) and (9.376) that computes combination weights in a manner that is aware of the variation of the variance-product profile across the network. Nevertheless, in addition to the measurement noises  $\{v_k(i)\}$  at the individual nodes, we also need to consider the effect of perturbations that are introduced during the exchange of information among neighboring nodes. Noise over the communication links can be due to various factors including thermal noise and imperfect channel information. Studying the degradation in mean-square performance that results from these noisy exchanges can be pursued by straightforward extension of the mean-square analysis of Section 3.09.6, as we proceed to illustrate. Subsequently, we shall use the results to show how the combination weights can also be adapted in the presence of noisy exchange links.

### 3.09.9.1 Noise sources over exchange links

To model noisy links, we introduce an additive noise component into each of the steps of the diffusion strategy (9.201)–(9.203) during the operations of information exchange among the nodes. The notation becomes a bit cumbersome because we need to account for both the source and destination of the information that is being exchanged. For example, the same signal  $\mathbf{d}_\ell(i)$  that is generated by node  $\ell$  will be broadcast to all the neighbors of node  $\ell$ . When this is done, a different noise will interfere with the exchange of  $\mathbf{d}_\ell(i)$  over each of the edges that link node  $\ell$  to its neighbors. Thus, we will need to use a notation of the form  $\mathbf{d}_{\ell k}(i)$ , with two subscripts  $\ell$  and  $k$ , to indicate that this is the noisy version of  $\mathbf{d}_\ell(i)$  that is received by node  $k$  from node  $\ell$ . The subscript  $\ell k$  indicates that  $\ell$  is the source and  $k$  is the sink, i.e., information is moving from  $\ell$  to  $k$ . For the reverse situation where information flows from node  $k$  to  $\ell$ , we would use instead the subscript  $k \ell$ .

With this notation in mind, we model the noisy data received by node  $k$  from its neighbor  $\ell$  as follows:

$$\mathbf{w}_{\ell k, i-1} = \mathbf{w}_{\ell, i-1} + \mathbf{v}_{\ell k, i-1}^{(w)}, \quad (9.407)$$

$$\boldsymbol{\psi}_{\ell k, i} = \boldsymbol{\psi}_{\ell, i} + \mathbf{v}_{\ell k, i}^{(\psi)}, \quad (9.408)$$

$$\mathbf{u}_{\ell k, i} = \mathbf{u}_{\ell, i} + \mathbf{v}_{\ell k, i}^{(u)}, \quad (9.409)$$

$$\mathbf{d}_{\ell k}(i) = \mathbf{d}_\ell(i) + \mathbf{v}_{\ell k}^{(d)}(i), \quad (9.410)$$

where  $\mathbf{v}_{\ell k, i-1}^{(w)} (M \times 1)$ ,  $\mathbf{v}_{\ell k, i}^{(\psi)} (M \times 1)$ , and  $\mathbf{v}_{\ell k, i}^{(u)} (1 \times M)$  are vector noise signals, and  $\mathbf{v}_{\ell k}^{(d)}(i)$  is a scalar noise signal. These are the noise signals that perturb exchanges over the edge linking source  $\ell$  to sink  $k$  (i.e., for data sent from node  $\ell$  to node  $k$ ). The superscripts  $\{(w), (\psi), (u), (d)\}$  in each case refer to the variable that these noises perturb. Figure 9.14 illustrates the various noise sources that perturb the exchange of information from node  $\ell$  to node  $k$ . The figure also shows the measurement noises  $\{\mathbf{v}_\ell(i), \mathbf{v}_k(i)\}$  that exist locally at the nodes in view of the data model (9.208).

We assume that the following noise signals, which influence the data received by node  $k$ ,

$$\left\{ \mathbf{v}_k(i), \mathbf{v}_{\ell k}^{(d)}(i), \mathbf{v}_{\ell k, i-1}^{(w)}, \mathbf{v}_{\ell k, i}^{(\psi)}, \mathbf{v}_{\ell k, i}^{(u)} \right\} \quad (9.411)$$

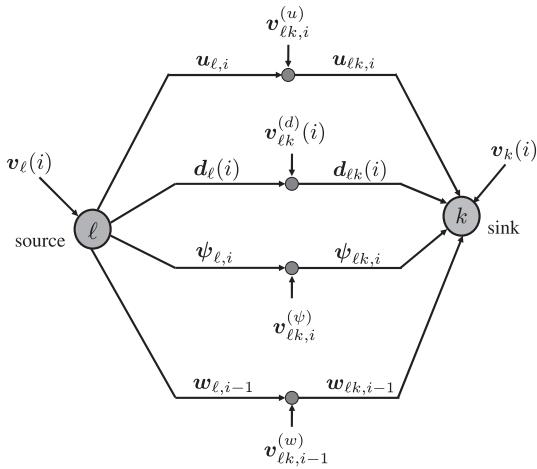
are temporally white and spatially independent random processes with zero mean and variances or covariances given by

$$\boxed{\left\{ \sigma_{v,k}^2, \sigma_{v,\ell k}^2, R_{v,\ell k}^{(w)}, R_{v,\ell k}^{(\psi)}, R_{v,\ell k}^{(u)} \right\}}. \quad (9.412)$$

Obviously, the quantities

$$\left\{ \sigma_{v,\ell k}^2, R_{v,\ell k}^{(w)}, R_{v,\ell k}^{(\psi)}, R_{v,\ell k}^{(u)} \right\} \quad (9.413)$$

are all zero if  $\ell \notin \mathcal{N}_k$  or when  $\ell = k$ . We further assume that the noise processes (9.411) are independent of each other and of the regression data  $\mathbf{u}_{m,j}$  for all  $k, \ell, m$  and  $i, j$ .


**FIGURE 9.14**

Additive noise sources perturb the exchange of information from node  $\ell$  to node  $k$ . The subscript  $\ell k$  in this illustration indicates that  $\ell$  is the source node and  $k$  is the sink node so that information is flowing from  $\ell$  to  $k$ .

### 3.09.9.2 Error recursion

Using the perturbed data (9.407)–(9.410), the adaptive diffusion strategy (9.201)–(9.203) becomes

$$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \mathbf{w}_{\ell k,i-1}, \quad (9.414)$$

$$\psi_{k,i} = \phi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell k,i}^* [\mathbf{d}_{\ell k}(i) - \mathbf{u}_{\ell k,i} \phi_{k,i-1}], \quad (9.415)$$

$$\mathbf{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \psi_{\ell k,i}. \quad (9.416)$$

Observe that the perturbed quantities  $\{\mathbf{w}_{\ell k,i-1}, \mathbf{u}_{\ell k,i}, \mathbf{d}_{\ell k}(i), \psi_{\ell k,i}\}$ , with subscripts  $\ell k$ , appear in (9.414)–(9.416) in place of the original quantities  $\{\mathbf{w}_{\ell,i-1}, \mathbf{u}_{\ell,i}, \mathbf{d}_{\ell}(i), \psi_{\ell,i}\}$  that appear in (9.201)–(9.203). This is because these quantities are now subject to exchange noises. As before, we are still interested in examining the evolution of the weight-error vectors:

$$\tilde{\mathbf{w}}_{k,i} \triangleq \mathbf{w}^o - \mathbf{w}_{k,i}, \quad k = 1, 2, \dots, N. \quad (9.417)$$

For this purpose, we again introduce the following  $N \times 1$  block vector, whose entries are of size  $M \times 1$  each:

$$\tilde{\mathbf{w}}_i \triangleq \begin{bmatrix} \tilde{\mathbf{w}}_{1,i} \\ \tilde{\mathbf{w}}_{2,i} \\ \vdots \\ \tilde{\mathbf{w}}_{N,i} \end{bmatrix} \quad (9.418)$$

and proceed to determine a recursion for its evolution over time. The arguments are largely similar to what we already did before in Section 3.09.6.3 and, therefore, we shall emphasize the differences that arise. The main deviation is that we now need to account for the presence of the new noise signals; they will contribute additional terms to the recursion for  $\tilde{w}_i$ —see (9.442) further ahead. We may note that some studies on the effect of imperfect data exchanges on the performance of adaptive diffusion algorithms are considered in [66–68]. These earlier investigations were limited to particular cases in which only noise in the exchange of  $w_{\ell,i-1}$  was considered (as in (9.407)), in addition to setting  $C = I$  (in which case there is *no* exchange of  $\{d_\ell(i), u_{\ell,i}\}$ ), and by focusing on the CTA case for which  $A_2 = I$ . Here, we consider instead the general case that accounts for the additional sources of imperfections shown in (9.408)–(9.410), in addition to the general diffusion strategy (9.201)–(9.203) with combination matrices  $\{A_1, A_2, C\}$ .

To begin with, we introduce the aggregate  $M \times 1$  zero-mean noise signals:

$$\mathbf{v}_{k,i-1}^{(w)} \triangleq \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} \mathbf{v}_{\ell k,i-1}^{(w)}, \quad \mathbf{v}_{k,i}^{(\psi)} \triangleq \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \mathbf{v}_{\ell k,i}^{(\psi)}. \quad (9.419)$$

These noises represent the aggregate effect on node  $k$  of all exchange noises from the neighbors of node  $k$  while exchanging the estimates  $\{w_{\ell,i-1}, \psi_{\ell,i}\}$  during the two combination steps (9.201) and (9.203). The  $M \times M$  covariance matrices of these noises are given by

$$R_{v,k}^{(w)} \triangleq \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k}^2 R_{v,\ell k}^{(w)}, \quad R_{v,k}^{(\psi)} \triangleq \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k}^2 R_{v,\ell k}^{(\psi)}. \quad (9.420)$$

These expressions aggregate the exchange noise covariances in the neighborhood of node  $k$ ; the covariances are scaled by the squared coefficients  $\{a_{1,\ell k}^2, a_{2,\ell k}^2\}$ . We collect these noise signals, and their covariances, from across the network into  $N \times 1$  block vectors and  $N \times N$  block diagonal matrices as follows:

$$\mathbf{v}_{i-1}^{(w)} \triangleq \text{col} \left\{ \mathbf{v}_{1,i-1}^{(w)}, \mathbf{v}_{2,i-1}^{(w)}, \dots, \mathbf{v}_{N,i-1}^{(w)} \right\}, \quad (9.421)$$

$$\mathbf{v}_i^{(\psi)} \triangleq \text{col} \left\{ \mathbf{v}_{1,i}^{(\psi)}, \mathbf{v}_{2,i}^{(\psi)}, \dots, \mathbf{v}_{N,i}^{(\psi)} \right\}, \quad (9.422)$$

$$R_v^{(w)} \triangleq \text{diag} \left\{ R_{v,1}^{(w)}, R_{v,2}^{(w)}, \dots, R_{v,N}^{(w)} \right\}, \quad (9.423)$$

$$R_v^{(\psi)} \triangleq \text{diag} \left\{ R_{v,1}^{(\psi)}, R_{v,2}^{(\psi)}, \dots, R_{v,N}^{(\psi)} \right\}. \quad (9.424)$$

We further introduce the following scalar zero-mean noise signal:

$$\mathbf{v}_{\ell k}(i) \triangleq \mathbf{v}_\ell(i) + \mathbf{v}_{\ell k}^{(d)}(i) - \mathbf{v}_{\ell k,i}^{(u)} w^o, \quad (9.425)$$

whose variance is

$$\sigma_{\ell k}^2 = \sigma_{v,\ell}^2 + \sigma_{v,\ell k}^2 + w^{o*} R_{v,\ell k}^{(u)} w^o. \quad (9.426)$$

In the absence of exchange noises for the data  $\{d_\ell(i), u_{\ell,i}\}$ , the signal  $\mathbf{v}_{\ell k}(i)$  would coincide with the measurement noise  $\mathbf{v}_\ell(i)$ . Expression (9.425) is simply a reflection of the aggregate effect of the noises

in exchanging  $\{\mathbf{d}_\ell(i), \mathbf{u}_{\ell,i}\}$  on node  $k$ . Indeed, starting from the data model (9.208) and using (9.409) and (9.410), we can easily verify that the noisy data  $\{\mathbf{d}_{\ell k}(i), \mathbf{u}_{\ell k,i}\}$  are related via:

$$\mathbf{d}_{\ell k}(i) = \mathbf{u}_{\ell k,i} w^o + \mathbf{v}_{\ell k}(i). \quad (9.427)$$

We also define (compare with (9.234) and (9.235) and note that we are now using the perturbed regression vectors  $\{\mathbf{u}_{\ell k,i}\}$ ):

$$\mathbf{R}'_{k,i} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell k,i}^* \mathbf{u}_{\ell k,i}, \quad (9.428)$$

$$\mathcal{R}'_i \triangleq \text{diag} \{ \mathbf{R}'_{1,i}, \mathbf{R}'_{2,i}, \dots, \mathbf{R}'_{N,i} \}. \quad (9.429)$$

It holds that

$$\boxed{\mathbb{E} \mathbf{R}'_{k,i} = R'_k}, \quad (9.430)$$

where

$$\boxed{R'_k \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \left[ R_{u,\ell} + R_{v,\ell k}^{(u)} \right].} \quad (9.431)$$

When there is no noise during the exchange of the regression data, i.e., when  $R_{v,\ell k}^{(u)} = 0$ , the expressions for  $\{\mathbf{R}'_{k,i}, \mathcal{R}'_i, R'_k\}$  reduce to expressions (9.234), (9.235), and (9.182) for  $\{\mathbf{R}_{k,i}, \mathcal{R}_i, R_k\}$ .

Likewise, we introduce (compare with (9.239)):

$$\mathbf{z}_{k,i} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbf{u}_{\ell k,i}^* \mathbf{v}_{\ell k}(i), \quad (9.432)$$

$$\mathbf{z}_i \triangleq \text{col}\{\mathbf{z}_{1,i}, \mathbf{z}_{2,i}, \dots, \mathbf{z}_{N,i}\}. \quad (9.433)$$

Compared with the earlier definition for  $s_i$  in (9.239) when there is no noise over the exchange links, we see that we now need to account for the various noisy versions of the same regression vector  $\mathbf{u}_{\ell,i}$  and the same signal  $\mathbf{d}_\ell(i)$ . For instance, the vectors  $\mathbf{u}_{\ell k,i}$  and  $\mathbf{u}_{\ell m,i}$  would denote two noisy versions received by nodes  $k$  and  $m$  for the *same* regression vector  $\mathbf{u}_{\ell,i}$  transmitted from node  $\ell$ . Likewise, the scalars  $\mathbf{d}_{\ell k}(i)$  and  $\mathbf{d}_{\ell m}(i)$  would denote two noisy versions received by nodes  $k$  and  $m$  for the *same* scalar  $\mathbf{d}_\ell(i)$  transmitted from node  $\ell$ . As a result, the quantity  $\mathbf{z}_i$  is not zero mean any longer (in contrast to  $s_i$ , which had zero mean). Indeed, note that

$$\begin{aligned} \mathbb{E} \mathbf{z}_{k,i} &= \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbb{E} \mathbf{u}_{\ell k,i}^* \mathbf{v}_{\ell k}(i) \\ &= \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \mathbb{E} \left( \left[ \mathbf{u}_{\ell,i} + \mathbf{v}_{\ell k,i}^{(u)} \right]^* \cdot \left[ \mathbf{v}_{\ell}(i) + \mathbf{v}_{\ell k}^{(d)}(i) - \mathbf{v}_{\ell k,i}^{(u)} w^o \right] \right) \\ &= - \left( \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{v,\ell k}^{(u)} \right) w^o. \end{aligned} \quad (9.434)$$

It follows that

$$\mathbb{E}z_i = - \begin{bmatrix} \sum_{\ell \in \mathcal{N}_1} c_{\ell 1} R_{v, \ell 1}^{(u)} \\ \sum_{\ell \in \mathcal{N}_2} c_{\ell 2} R_{v, \ell 2}^{(u)} \\ \vdots \\ \sum_{\ell \in \mathcal{N}_N} c_{\ell N} R_{v, \ell N}^{(u)} \end{bmatrix} w^o. \quad (9.435)$$

Although we can continue our analysis by studying this general case in which the vectors  $z_i$  do not have zero-mean (see [65, 69]), we shall nevertheless limit our discussion in the sequel to the case in which there is no noise during the exchange of the regression data, i.e., we henceforth assume that:

$$\boxed{v_{\ell k, i}^{(u)} = 0, \quad R_{v, \ell k}^{(u)} = 0, \quad u_{\ell k, i} = u_{\ell, i}} \quad (\text{assumption from this point onwards}). \quad (9.436)$$

We maintain all other noise sources, which occur during the exchange of the weight estimates  $\{w_{\ell, i-1}, \psi_{\ell, i}\}$  and the data  $\{d_{\ell}(i)\}$ . Under condition (9.436), we obtain

$$\mathbb{E}z_i = 0, \quad (9.437)$$

$$\sigma_{\ell k}^2 = \sigma_{v, \ell}^2 + \sigma_{v, \ell k}^2, \quad (9.438)$$

$$R'_k = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} R_{u, \ell} \stackrel{(9.182)}{=} R_k. \quad (9.439)$$

Then, the covariance matrix of each term  $z_{k, i}$  is given by

$$\boxed{R_{z, k} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k}^2 \sigma_{\ell k}^2 R_{u, \ell}} \quad (9.440)$$

and the covariance matrix of  $z_i$  is  $N \times N$  block diagonal with blocks of size  $M \times M$ :

$$\boxed{\mathcal{Z} \triangleq \mathbb{E}z_i z_i^* = \text{diag}\{R_{z, 1}, R_{z, 2}, \dots, R_{z, N}\}.} \quad (9.441)$$

Now repeating the argument that led to (9.246) we arrive at the following recursion for the weight-error vector:

$$\boxed{\tilde{w}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}'_i) \mathcal{A}_1^T \tilde{w}_{i-1} - \mathcal{A}_2^T \mathcal{M}z_i - \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}'_i) v_{i-1}^{(w)} - v_i^{(\psi)} \quad (\text{noisy links})} \quad (9.442)$$

For comparison purposes, we repeat recursion (9.246) here (recall that this recursion corresponds to the case when the exchanges over the links are not subject to noise):

$$\tilde{w}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}_i) \mathcal{A}_1^T \tilde{w}_{i-1} - \mathcal{A}_2^T \mathcal{M}\mathcal{C}^T s_i \quad (\text{perfect links}). \quad (9.443)$$

Comparing (9.442) and (9.443) we find that:

1. The covariance matrix  $\mathcal{R}_i$  in (9.443) is replaced by  $\mathcal{R}'_i$ . Recall from (9.429) that  $\mathcal{R}'_i$  contains the influence of the noises that arise during the exchange of the regression data, i.e., the  $\{R_{v,\ell k}^{(u)}\}$ . But since we are now assuming that  $R_{v,\ell k}^{(u)} = 0$ , then  $\mathcal{R}'_i = \mathcal{R}_i$ .
2. The term  $\mathcal{C}^T s_i$  in (9.443) is replaced by  $z_i$ . Recall from (9.432) that  $z_i$  contains the influence of the noises that arise during the exchange of the measurement data and the regression data, i.e., the  $\{\sigma_{v,\ell k}^2, R_{v,\ell k}^{(u)}\}$ .
3. Two new driving terms appear involving  $v_{i-1}^{(w)}$  and  $v_i^{(\psi)}$ . These terms reflect the influence of the noises during the exchange of the weight estimates  $\{w_{\ell,i-1}, \psi_{\ell,i}\}$ .
4. Observe further that:
  - a. The term involving  $v_{i-1}^{(w)}$  accounts for noise introduced at the information-exchange step (9.414) *before* adaptation.
  - b. The term involving  $z_i$  accounts for noise introduced during the adaptation step (9.415).
  - c. The term involving  $v_i^{(\psi)}$  accounts for noise introduced at the information-exchange step (9.416) *after* adaptation.

Therefore, since we are not considering noise during the exchange of the regression data, the weight-error recursion (9.442) simplifies to:

$$\tilde{w}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}_i) \mathcal{A}_1^T \tilde{w}_{i-1} - \mathcal{A}_2^T \mathcal{M}z_i - \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}_i) v_{i-1}^{(w)} - v_i^{(\psi)} \quad (\text{noisy links}), \quad (9.444)$$

where we used the fact that  $\mathcal{R}'_i = \mathcal{R}_i$  under these conditions.

### 3.09.9.3 Convergence in the mean

Taking expectations of both sides of (9.444) we find that the mean error vector evolves according to the following recursion:

$$\mathbb{E}\tilde{w}_i = \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}) \mathcal{A}_1^T \cdot \mathbb{E}\tilde{w}_{i-1}, \quad i \geq 0 \quad (9.445)$$

with  $\mathcal{R}$  defined by (9.181). This is the same recursion encountered earlier in (9.248) during perfect data exchanges. Note that had we considered noises during the exchange of the regression data, then the vector  $z_i$  in (9.444) would not be zero mean and the matrix  $\mathcal{R}'_i$  will have to be used instead of  $\mathcal{R}_i$ . In that case, the recursion for  $\mathbb{E}\tilde{w}_i$  will be different from (9.445); i.e., the presence of noise during the exchange of regression data alters the dynamics of the mean error vector in an important way—see [65, 69] for details on how to extend the arguments to this general case with a driving non-zero bias term. We can now extend Theorem 9.6.1 to the current scenario.

**Theorem 9.9.1 (Convergence in the Mean).** *Consider the problem of optimizing the global cost (9.92) with the individual cost functions given by (9.93). Pick a right stochastic matrix  $C$  and left stochastic matrices  $A_1$  and  $A_2$  satisfying (9.166). Assume each node in the network runs the perturbed adaptive diffusion algorithm (9.414)–(9.416). Assume further that the exchange of the variables  $\{w_{\ell,i-1}, \psi_{\ell,i}, d_{\ell}(i)\}$  is subject to additive noises as in (9.407), (9.408), and (9.410). We assume that the*

regressors are exchanged unperturbed. Then, all estimators  $\{\mathbf{w}_{k,i}\}$  across the network will still converge in the mean to the optimal solution  $w^o$  if the step-size parameters  $\{\mu_k\}$  satisfy

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}, \quad (9.446)$$

where the neighborhood covariance matrix  $R_k$  is defined by (9.182). That is,  $\mathbb{E}\mathbf{w}_{k,i} \rightarrow w^o$  as  $i \rightarrow \infty$ .  $\square$

### 3.09.9.4 Mean-square convergence

Recall from (9.264) that we introduced the matrix:

$$\mathcal{B} \triangleq \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}) \mathcal{A}_1^T. \quad (9.447)$$

We further introduce the  $N \times N$  block matrix with blocks of size  $M \times M$  each:

$$\mathcal{H} \triangleq \mathcal{A}_2^T (I_{NM} - \mathcal{M}\mathcal{R}). \quad (9.448)$$

Then, starting from (9.444) and repeating the argument that led to (9.279) we can establish the validity of the following variance relation:

$$\boxed{\mathbb{E}\|\tilde{\mathbf{w}}_i\|_\sigma^2 = \mathbb{E}\|\tilde{\mathbf{w}}_{i-1}\|_{\mathcal{F}\sigma}^2 + \left[ \text{vec}(\mathcal{A}_2^T \mathcal{M} \mathcal{Z}^T \mathcal{M} \mathcal{A}_2) + \text{vec}((\mathcal{H}\mathcal{R}_v^{(w)^T} \mathcal{H}^*)^T) + \text{vec}(\mathcal{R}_v^{(\psi)^T}) \right]^T \sigma} \quad (9.449)$$

for an arbitrary nonnegative-definite weighting matrix  $\Sigma$  with  $\sigma = \text{vec}(\Sigma)$ , and where  $\mathcal{F}$  is the same matrix defined earlier either by (9.276) or (9.277). We can therefore extend the statement of Theorem 9.6.7 to the present scenario.

**Theorem 9.9.2 (Mean-Square Stability).** Consider the same setting of Theorem 9.9.1. Assume sufficiently small step-sizes to justify ignoring terms that depend on higher powers of the step-sizes. The perturbed adaptive diffusion algorithm (9.414)–(9.416) is mean-square stable if, and only if, the matrix  $\mathcal{F}$  defined by (9.276), or its approximation (9.277), is stable (i.e., all its eigenvalues are strictly inside the unit disc). This condition is satisfied by sufficiently small step-sizes  $\{\mu_k\}$  that also satisfy:

$$\boxed{\mu_k < \frac{2}{\lambda_{\max}(R_k)}}, \quad (9.450)$$

where the neighborhood covariance matrix  $R_k$  is defined by (9.182). Moreover, the convergence rate of the algorithm is determined by  $[\rho(\mathcal{B})]^2$ .  $\square$

We conclude from the previous two theorems that the conditions for the mean and mean-square convergence of the adaptive diffusion strategy are not affected by the presence of noises over the exchange links (under the assumption that the regression data are exchanged without perturbation; otherwise,

the convergence conditions would be affected). The mean-square performance, on the other hand, is affected as follows. Introduce the  $N \times N$  block matrix:

$$\boxed{\mathcal{Y}_{\text{imperfect}} \triangleq \mathcal{A}_2^T \mathcal{M} \mathcal{Z} \mathcal{M} \mathcal{A}_2 + \mathcal{H} \mathcal{R}_v^{(w)} \mathcal{H}^* + \mathcal{R}_v^{(\psi)}} \quad (\text{imperfect exchanges}), \quad (9.451)$$

which should be compared with the corresponding quantity defined by (9.280) for the perfect exchanges case, namely,

$$\mathcal{Y}_{\text{perfect}} = \mathcal{A}_2^T \mathcal{M} \mathcal{C}^T \mathcal{S} \mathcal{C} \mathcal{M} \mathcal{A}_2 \quad (\text{perfect exchanges}). \quad (9.452)$$

When perfect exchanges occur, the matrix  $\mathcal{Z}$  reduces to  $\mathcal{C}^T \mathcal{S} \mathcal{C}$ . We can relate  $\mathcal{Y}_{\text{imperfect}}$  and  $\mathcal{Y}_{\text{perfect}}$  as follows. Let

$$\mathcal{R}^{(du)} \triangleq \text{diag} \left\{ \sum_{\ell \in \mathcal{N}_1} c_{\ell 1}^2 \sigma_{v, \ell 1}^2 R_{u, \ell}, \sum_{\ell \in \mathcal{N}_2} c_{\ell 2}^2 \sigma_{v, \ell 2}^2 R_{u, \ell}, \dots, \sum_{\ell \in \mathcal{N}_N} c_{\ell N}^2 \sigma_{v, \ell N}^2 R_{u, \ell} \right\}. \quad (9.453)$$

Then, using (9.438) and (9.441), it is straightforward to verify that

$$\mathcal{Z} = \mathcal{C}^T \mathcal{S} \mathcal{C} + \mathcal{R}^{(du)} \quad (9.454)$$

and it follows that:

$$\begin{aligned} \mathcal{Y}_{\text{imperfect}} &= \mathcal{Y}_{\text{perfect}} + \mathcal{A}_2^T \mathcal{M} \mathcal{R}^{(du)} \mathcal{M} \mathcal{A}_2 + \mathcal{H} \mathcal{R}_v^{(w)} \mathcal{H}^* + \mathcal{R}_v^{(\psi)} \\ &\triangleq \mathcal{Y}_{\text{perfect}} + \Delta \mathcal{Y}. \end{aligned} \quad (9.455)$$

Expression (9.455) reflects the influence of the noises  $\{R_v^{(w)}, R_v^{(\psi)}, \sigma_{v, \ell k}^2\}$ . Substituting the definition (9.451) into (9.449), and taking the limit as  $i \rightarrow \infty$ , we obtain from the latter expression that:

$$\boxed{\lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_i\|_{(I - \mathcal{F})\sigma}^2 = \left[ \text{vec}(\mathcal{Y}_{\text{imperfect}}^T) \right]^T \sigma}, \quad (9.456)$$

which has the same form as (9.284); therefore, we can proceed analogously to obtain:

$$\boxed{\text{MSD}_{\text{imperfect}}^{\text{network}} = \frac{1}{N} \cdot \left[ \text{vec}(\mathcal{Y}_{\text{imperfect}}^T) \right]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(I_{NM})} \quad (9.457)$$

and

$$\boxed{\text{EMSE}_{\text{imperfect}}^{\text{network}} = \frac{1}{N} \cdot \left[ \text{vec}(\mathcal{Y}_{\text{imperfect}}^T) \right]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(\mathcal{R}_u)}. \quad (9.458)$$

Using (9.455), we see that the network MSD and EMSE deteriorate as follows:

$$\text{MSD}_{\text{imperfect}}^{\text{network}} = \text{MSD}_{\text{perfect}}^{\text{network}} + \frac{1}{N} \cdot \left[ \text{vec}(\Delta \mathcal{Y}^T) \right]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(I_{NM}), \quad (9.459)$$

$$\text{EMSE}_{\text{imperfect}}^{\text{network}} = \text{EMSE}_{\text{perfect}}^{\text{network}} + \frac{1}{N} \cdot \left[ \text{vec}(\Delta \mathcal{Y}^T) \right]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(\mathcal{R}_u). \quad (9.460)$$

### 3.09.9.5 Adaptive combination weights

We can repeat the discussion from Sections 3.09.8.2 and 3.09.8.3 to devise one adaptive scheme to adjust the combination coefficients in the noisy exchange case. We illustrate the construction by considering the ATC strategy corresponding to  $A_1 = I_N$ ,  $A_2 = A$ ,  $C = I_N$ , so that only weight estimates are exchanged and the update recursions are of the form:

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} w_{k,i-1}], \quad (9.461)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell k,i}, \quad (9.462)$$

where from (9.408):

$$\psi_{\ell k,i} = \psi_{\ell,i} + v_{\ell k,i}^{(\psi)}. \quad (9.463)$$

In this case, the network MSD performance (9.457) becomes

$$\text{MSD}_{\text{atc}, C=I, \text{imperfect}}^{\text{network}} = \frac{1}{N} \sum_{j=0}^{\infty} \text{Tr} \left( \mathcal{B}_{\text{atc}, C=I}^j \mathcal{Y}_{\text{atc}, \text{imperfect}} \mathcal{B}_{\text{atc}, C=I}^{*j} \right), \quad (9.464)$$

where, since now  $\mathcal{Z} = \mathcal{S}$  and  $\mathcal{R}_v^{(w)} = 0$ , we have

$$\mathcal{B}_{\text{atc}, C=I} = \mathcal{A}^T (I - \mathcal{M} \mathcal{R}_u), \quad (9.465)$$

$$\mathcal{Y}_{\text{atc}, \text{imperfect}} = \mathcal{A}^T \mathcal{M} \mathcal{S} \mathcal{M} \mathcal{A} + \mathcal{R}_v^{(\psi)}, \quad (9.466)$$

$$R_v^{(\psi)} = \text{diag} \left\{ R_{v,1}^{(\psi)}, R_{v,2}^{(\psi)}, \dots, R_{v,N}^{(\psi)} \right\}, \quad (9.467)$$

$$R_{v,k}^{(\psi)} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k}^2 R_{v,\ell k}^{(\psi)}, \quad (9.468)$$

$$\mathcal{R}_u = \text{diag} \left\{ R_{u,1}, R_{u,2}, \dots, R_{u,N} \right\}, \quad (9.469)$$

$$\mathcal{S} = \text{diag} \left\{ \sigma_{v,1}^2 R_{u,1}, \sigma_{v,2}^2 R_{u,2}, \dots, \sigma_{v,N}^2 R_{u,N} \right\}, \quad (9.470)$$

$$\mathcal{M} = \text{diag} \{ \mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M \}, \quad (9.471)$$

$$\mathcal{A} = A \otimes I_M. \quad (9.472)$$

To proceed, as was the case with (9.389), we consider the following simplified optimization problem:

$$\begin{aligned} & \min_A \text{Tr}(\mathcal{Y}_{\text{atc}, \text{imperfect}}) \\ & \text{subject to } A^T \mathbb{1} = \mathbb{1}, \quad a_{\ell k} \geq 0, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad (9.473)$$

Using (9.466), the trace of  $\mathcal{Y}_{\text{atc}, \text{imperfect}}$  can be expressed in terms of the combination coefficients as follows:

$$\text{Tr}(\mathcal{Y}_{\text{atc}, \text{imperfect}}) = \sum_{k=1}^N \sum_{\ell=1}^N a_{\ell k}^2 \left[ \mu_\ell^2 \sigma_{v,\ell}^2 \text{Tr}(R_{u,\ell}) + \text{Tr} \left( R_{v,\ell k}^{(\psi)} \right) \right], \quad (9.474)$$

so that problem (9.473) can be decoupled into  $N$  separate optimization problems of the form:

$$\boxed{\begin{aligned} \min_{\{a_{\ell k}\}_{\ell=1}^N} \quad & \sum_{\ell=1}^N a_{\ell k}^2 \left[ \mu_\ell^2 \sigma_{v,\ell}^2 \text{Tr}(R_{u,\ell}) + \text{Tr} \left( R_{v,\ell k}^{(\psi)} \right) \right], \quad k = 1, \dots, N \\ \text{subject to} \quad & a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \end{aligned}} . \quad (9.475)$$

With each node  $\ell$ , we associate the following nonnegative variance products:

$$\boxed{\gamma_{\ell k}^2 \triangleq \mu_\ell^2 \cdot \sigma_{v,\ell}^2 \cdot \text{Tr}(R_{u,\ell}) + \text{Tr} \left( R_{v,\ell k}^{(\psi)} \right), \quad k \in \mathcal{N}_\ell} . \quad (9.476)$$

This measure now incorporates information about the exchange noise covariances  $\{R_{v,\ell k}^{(\psi)}\}$ . Then, the solution of (9.475) is given by:

$$\boxed{a_{\ell k} = \begin{cases} \frac{\gamma_{\ell k}^{-2}}{\sum_{m \in \mathcal{N}_k} \gamma_{mk}^{-2}}, & \text{if } \ell \in \mathcal{N}_k \\ 0, & \text{otherwise} \end{cases} \quad (\text{relative-variance rule})} . \quad (9.477)$$

We continue to refer to this combination rule as the *relative-variance combination rule* [64]; it leads to a left-stochastic matrix  $A$ . To evaluate the combination weights (9.477), the nodes need to know the variance products,  $\{\gamma_{mk}^2\}$ , of their neighbors. As before, we can motivate one adaptive construction as follows.

We refer to the ATC recursion (9.461)–(9.463) and use the data model (9.208) to write for node  $\ell$ :

$$\boldsymbol{\psi}_{\ell k,i} = \mathbf{w}_{\ell,i-1} + \mu_\ell \mathbf{u}_{\ell,i}^* [\mathbf{u}_{\ell,i} \tilde{\mathbf{w}}_{\ell,i-1} + \mathbf{v}_\ell(i)] + \mathbf{v}_{\ell k,i}^{(\psi)}, \quad (9.478)$$

so that, in view of our earlier assumptions on the regression data and noise in Sections 3.09.6.1 and 3.09.9.1, we obtain in the limit as  $i \rightarrow \infty$ :

$$\lim_{i \rightarrow \infty} \mathbb{E} \|\boldsymbol{\psi}_{\ell k,i} - \mathbf{w}_{\ell,i-1}\|^2 = \mu_\ell^2 \cdot \left( \lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{\ell,i-1}\|_{\mathbb{E}(\mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i})}^2 \right) + \mu_\ell^2 \cdot \sigma_{v,\ell}^2 \cdot \text{Tr}(R_{u,\ell}) + \text{Tr} \left( R_{v,\ell k}^{(\psi)} \right) . \quad (9.479)$$

In a manner similar to what was done before for (9.396), we can evaluate the limit on the right-hand side by using the corresponding steady-state result (9.456). We select the vector  $\sigma$  in (9.456) to satisfy:

$$(I - \mathcal{F})\sigma = \text{vec} \left[ \mathbb{E} \left( \mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i} \right) \right] . \quad (9.480)$$

Then, from (9.456),

$$\lim_{i \rightarrow \infty} \mathbb{E} \|\tilde{\mathbf{w}}_{\ell,i-1}\|_{\mathbb{E}(\mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i})}^2 = \left[ \text{vec} \left( \mathcal{Y}_{\text{atc,imperfect}}^T \right) \right]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec} \left[ \mathbb{E} \left( \mathbf{u}_{\ell,i}^* \|\mathbf{u}_{\ell,i}\|^2 \mathbf{u}_{\ell,i} \right) \right] . \quad (9.481)$$

Now recall from expression (9.466) that the entries of  $\mathcal{Y}_{\text{atc},\text{imperfect}}$  depend on combinations of the squared step-sizes,  $\{\mu_m^2, m = 1, 2, \dots, N\}$ , and on terms involving  $\{\text{Tr}(R_{v,m}^{(\psi)})\}$ . This fact implies that the first term on the right-hand side of (9.479) depends on products of the form  $\{\mu_\ell^2 \mu_m^2\}$ ; these fourth-order factors can be ignored in comparison to the second-order factor  $\mu_\ell^2$  for small step-sizes. Moreover, the same first term on the right-hand side of (9.479) depends on products of the form  $\{\mu_\ell^2 \text{Tr}(R_{v,m}^{(\psi)})\}$ , which can be ignored in comparison to the last term,  $\text{Tr}(R_{v,\ell k}^{(\psi)})$ , in (9.479), which does not appear multiplied by a squared step-size. Therefore, we can approximate:

$$\begin{aligned}\lim_{i \rightarrow \infty} \mathbb{E} \|\boldsymbol{\psi}_{\ell k,i} - \mathbf{w}_{\ell,i-1}\|^2 &\approx \mu_\ell^2 \cdot \sigma_{v,\ell}^2 \cdot \text{Tr}(R_{u,\ell}) + \text{Tr}\left(R_{v,\ell k}^{(\psi)}\right) \\ &= \gamma_{\ell k}^2\end{aligned}\quad (9.482)$$

in terms of the desired variance product,  $\gamma_{\ell k}^2$ . Using the following instantaneous approximation at node  $k$  (where  $w_{\ell,i-1}$  is replaced by  $w_{k,i-1}$ ):

$$\mathbb{E} \|\boldsymbol{\psi}_{\ell k,i} - \mathbf{w}_{\ell,i-1}\|^2 \approx \|\boldsymbol{\psi}_{\ell k,i} - w_{k,i-1}\|^2, \quad (9.483)$$

we can motivate an algorithm that enables node  $k$  to estimate the variance products  $\gamma_{\ell k}^2$ . Thus, let  $\hat{\gamma}_{\ell k}^2(i)$  denote an estimate for  $\gamma_{\ell k}^2$  that is computed by node  $k$  at time  $i$ . Then, one way to evaluate  $\hat{\gamma}_{\ell k}^2(i)$  is through the recursion:

$$\boxed{\hat{\gamma}_{\ell k}^2(i) = (1 - \nu_k) \cdot \hat{\gamma}_{\ell k}^2(i-1) + \nu_k \cdot \|\boldsymbol{\psi}_{\ell k,i} - w_{k,i-1}\|^2}, \quad (9.484)$$

where  $\nu_k$  is a positive coefficient smaller than one. Indeed, it can be verified that

$$\lim_{i \rightarrow \infty} \mathbb{E} \hat{\gamma}_{\ell k}^2(i) \approx \gamma_{\ell k}^2, \quad (9.485)$$

so that the estimator  $\hat{\gamma}_{\ell k}^2(i)$  converges on average close to the desired variance product  $\gamma_{\ell k}^2$ . In this way, we can replace the weights (9.477) by the adaptive construction:

$$\boxed{a_{\ell k}(i) = \begin{cases} \frac{\hat{\gamma}_{\ell k}^{-2}(i)}{\sum_{m \in \mathcal{N}_k} \hat{\gamma}_{mk}^{-2}(i)}, & \text{if } \ell \in \mathcal{N}_k \\ 0, & \text{otherwise} \end{cases}}. \quad (9.486)$$

Equations (9.484) and (9.486) provide one adaptive construction for the combination weights  $\{a_{\ell k}\}$ .

### 3.09.10 Extensions and further considerations

Several extensions and variations of diffusion strategies are possible. Among those variations we mention strategies that endow nodes with temporal processing abilities, in addition to their spatial cooperation abilities. We can also apply diffusion strategies to solve recursive least-squares and state-space estimation problems in a distributed manner. In this section, we highlight select contributions in these and related areas.

### 3.09.10.1 Adaptive diffusion strategies with smoothing mechanisms

In the ATC and CTA adaptive diffusion strategies (9.153) and (9.154), each node in the network shares information locally with its neighbors through a process of spatial cooperation or combination. In this section, we describe briefly an extension that adds a temporal dimension to the processing at the nodes. For example, in the ATC implementation (9.153), rather than have each node  $k$  rely solely on current data,  $\{d_\ell(i), u_{\ell,i}, \ell \in \mathcal{N}_k\}$ , and on current weight estimates received from the neighbors,  $\{\psi_{\ell,i}, \ell \in \mathcal{N}_k\}$ , node  $k$  can be allowed to store and process present and past weight estimates, say,  $P$  of them as in  $\{\psi_{\ell,j}, j = i, i-1, \dots, i-P+1\}$ . In this way, previous weight estimates can be smoothed and used more effectively to help enhance the mean-square-deviation performance especially in the presence of noise over the communication links.

To motivate diffusion strategies with smoothing mechanisms, we continue to assume that the random data  $\{d_k(i), u_{k,i}\}$  satisfy the modeling assumptions of Section 3.09.6.1. The global cost (9.92) continues to be the same but the individual cost functions (9.93) are now replaced by

$$J_k(w) = \sum_{j=0}^{P-1} q_{kj} \mathbb{E} |d_k(i-j) - u_{k,i-j} w|^2 \quad (9.487)$$

so that

$$J^{\text{glob}}(w) = \sum_{k=1}^N \left( \sum_{j=0}^{P-1} q_{kj} \mathbb{E} |d_k(i-j) - u_{k,i-j} w|^2 \right), \quad (9.488)$$

where each coefficient  $q_{kj}$  is a nonnegative scalar representing the weight that node  $k$  assigns to data from time instant  $i-j$ . The coefficients  $\{q_{kj}\}$  are assumed to satisfy the normalization condition:

$$q_{ko} > 0, \quad \sum_{j=0}^{P-1} q_{kj} = 1, \quad k = 1, 2, \dots, N. \quad (9.489)$$

When the random processes  $d_k(i)$  and  $u_{k,i}$  are jointly wide-sense stationary, as was assumed in Section 3.09.6.1, the optimal solution  $w^o$  that minimizes (9.488) is still given by the same normal equations (9.40). We can extend the arguments of Sections 3.09.3 and 3.09.4 to (9.488) and arrive at the following version of a diffusion strategy incorporating temporal processing (or smoothing) of the intermediate weight estimates [70, 71]:

$$\phi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} q_{\ell o} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}] \quad (\text{adaptation}), \quad (9.490)$$

$$\psi_{k,i} = \sum_{j=0}^{P-1} f_{kj} \phi_{k,i-j} \quad (\text{temporal processing or smoothing}), \quad (9.491)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \quad (\text{spatial processing}), \quad (9.492)$$

where the nonnegative coefficients  $\{c_{\ell k}, a_{\ell k}, f_{kj}, q_{\ell o}\}$  satisfy:

for  $k = 1, 2, \dots, N$ :

$$c_{\ell k} \geq 0, \quad \sum_{k=1}^N c_{\ell k} = 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad (9.493)$$

$$a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad (9.494)$$

$$f_{kj} \geq 0, \quad \sum_{j=0}^{P-1} f_{kj} = 1, \quad (9.495)$$

$$0 < q_{\ell o} \leq 1. \quad (9.496)$$

Since only the coefficients  $\{q_{\ell o}\}$  are needed, we alternatively denote them by the simpler notation  $\{q_\ell\}$  in the listing in Table 9.8. These are simply chosen as nonnegative coefficients:

$$0 < q_\ell \leq 1, \quad \ell = 1, 2, \dots, N. \quad (9.497)$$

Note that algorithm (9.490)–(9.492) involves three steps: (a) an adaptation step (A) represented by (9.490); (b) a temporal filtering or smoothing step (T) represented by (9.491), and a spatial cooperation step (S) represented by (9.492). These steps are illustrated in Figure 9.15. We use the letters (A, T, S) to label these steps; and we use the sequence of letters (A, T, S) to designate the order of the steps. According to this convention, algorithm (9.490)–(9.492) is referred to as the ATS diffusion strategy since adaptation is followed by temporal processing, which is followed by spatial processing. In total, we can obtain six different combinations of diffusion algorithms by changing the order by which the temporal and spatial combination steps are performed in relation to the adaptation step. The resulting variations are summarized in Table 9.8. When we use only the most recent weight vector in the temporal filtering step (i.e., set  $\psi_{k,i} = \phi_{k,i}$ ), which corresponds to the case  $P = 1$ , the algorithms of Table 9.8 reduce to the ATC and CTA diffusion algorithms (9.153) and (9.154). Specifically, the variants TSA, STA, and SAT (where spatial processing S precedes adaptation A) reduce to CTA, while the variants TAS, ATS, and AST (where adaptation A precedes spatial processing S) reduce to ATC.

The mean-square performance analysis of the smoothed diffusion strategies can be pursued by extending the arguments of Section 3.09.6. This step is carried out in [70, 71] for doubly stochastic combination matrices  $A$  when the filtering coefficients  $\{f_{kj}\}$  do not change with  $k$ . For instance, it is shown in [71] that whether temporal processing is performed before or after adaptation, the strategy that performs adaptation before spatial cooperation is always better. Specifically, the six diffusion variants can be divided into two groups with the respective network MSDs satisfying the following relations:

$$\text{Group #1 : } \text{MSD}_{\text{TSA}}^{\text{network}} = \text{MSD}_{\text{STA}}^{\text{network}} \geq \text{MSD}_{\text{TAS}}^{\text{network}}, \quad (9.498)$$

$$\text{Group #2 : } \text{MSD}_{\text{SAT}}^{\text{network}} > \text{MSD}_{\text{ATS}}^{\text{network}} = \text{MSD}_{\text{AST}}^{\text{network}}. \quad (9.499)$$

Note that within groups 1 and 2, the order of the A and T operations is the same: in group 1, T precedes A and in group 2, A precedes T. Moreover, within each group, the order of the A and S operations determines performance; the strategy that performs A before S is better. Note further that when  $P = 1$ ,

**Table 9.8** Six Diffusion Strategies with Temporal Smoothing Steps

<b>TSA Diffusion:</b>	<b>TAS Diffusion:</b>
$\phi_{k,i-1} = \sum_{j=0}^{P-1} f_{kj} w_{k,i-j-1}$ $\psi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \phi_{\ell,i-1}$ $w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \psi_{k,i-1}]$	$\phi_{k,i-1} = \sum_{j=0}^{P-1} f_{kj} w_{k,i-j-1}$ $\psi_{k,i} = \phi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \phi_{k,i-1}]$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$
<b>STA Diffusion:</b>	<b>ATS Diffusion:</b>
$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $\psi_{k,i-1} = \sum_{j=0}^{P-1} f_{kj} \phi_{k,i-j-1}$ $w_{k,i} = \psi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \psi_{k,i-1}]$	$\phi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}]$ $\psi_{k,i} = \sum_{j=0}^{P-1} f_{kj} \phi_{k,i-j}$ $w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$
<b>SAT Diffusion:</b>	<b>AST Diffusion:</b>
$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}$ $\psi_{k,i} = \phi_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} \phi_{k,i-1}]$ $w_{k,i} = \sum_{j=0}^{P-1} f_{kj} \psi_{k,i-j}$	$\phi_{k,i} = w_{k,i-1} + \mu_k \sum_{\ell \in \mathcal{N}_k} q_\ell c_{\ell k} u_{\ell,i}^* [d_\ell(i) - u_{\ell,i} w_{k,i-1}]$ $\psi_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \phi_{\ell,i}$ $w_{k,i} = \sum_{j=0}^{P-1} f_{kj} \psi_{k,i-j}$

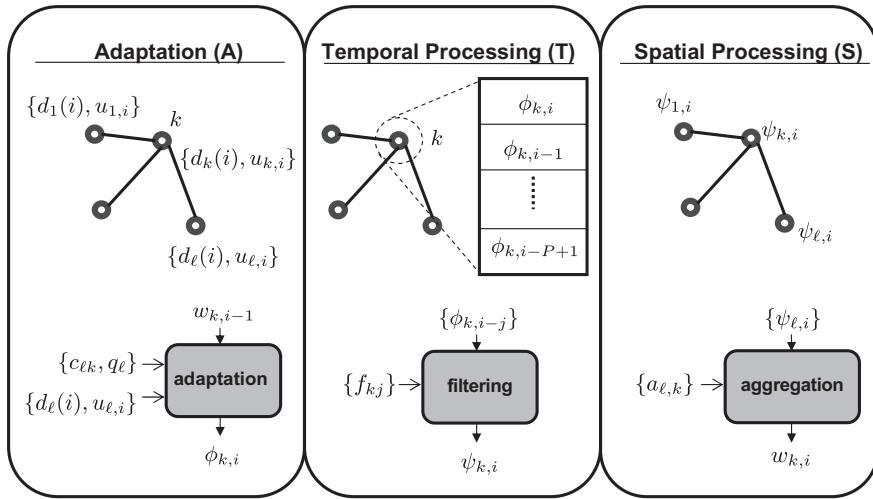


FIGURE 9.15

Illustration of the three steps involved in an ATS diffusion strategy: adaptation, followed by temporal processing or smoothing, followed by spatial processing.

so that temporal processing is not performed, then TAS reduces to ATC and TSA reduces to CTA. This conclusion is consistent with our earlier result (9.343) that ATC outperforms CTA.

In related work, reference [72] started from the CTA algorithm (9.159) without information exchange and added a useful projection step to it between the combination step and the adaptation step; i.e., the work considered an algorithm with an STA structure (with spatial combination occurring first, followed by a projection step, and then adaptation). The projection step uses the current weight estimate,  $\phi_{k,i}$ , at node  $k$  and projects it onto hyperslabs defined by the current and past raw data. Specifically, the algorithm from [72] has the following form:

$$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1}, \quad (9.500)$$

$$\psi_{k,i-1} = \mathcal{P}'_{k,i}[\phi_{k,i-1}], \quad (9.501)$$

$$w_{k,i} = \psi_{k,i-1} - \mu_k \left\{ \psi_{k,i-1} - \sum_{j=0}^{P-1} f_{kj} \cdot \mathcal{P}_{k,i-j}[\phi_{k,i-1}] \right\}, \quad (9.502)$$

where the notation  $\psi = \mathcal{P}_{k,i}[\phi]$  refers to the act of projecting the vector  $\phi$  onto the hyperslab  $P_{k,i}$  that consists of all  $M \times 1$  vectors  $z$  satisfying (similarly for the projection  $\mathcal{P}'_{k,i}$ ):

$$P_{k,i} \triangleq \{z \text{ such that } |d_k(i) - u_{k,i} z| \leq \epsilon_k\}, \quad (9.503)$$

$$P'_{k,i} \triangleq \{z \text{ such that } |d_k(i) - u_{k,i} z| \leq \epsilon'_k\}, \quad (9.504)$$

where  $\{\epsilon_k, \epsilon'_k\}$  are positive (tolerance) parameters chosen by the designer to satisfy  $\epsilon'_k > \epsilon_k$ . For generic values  $\{d, u, \epsilon\}$ , where  $d$  is a scalar and  $u$  is a row vector, the projection operator is described analytically

by the following expression [73]:

$$\mathcal{P}[\phi] = \phi + \begin{cases} \frac{u^*}{\|u\|^2}[d - \epsilon - u\phi], & \text{if } d - \epsilon > u\phi, \\ 0, & \text{if } |d - u\phi| \leq \epsilon, \\ \frac{u^*}{\|u\|^2}[d + \epsilon - u\phi], & \text{if } d + \epsilon < u\phi. \end{cases} \quad (9.505)$$

The projections that appear in (9.501) and (9.502) can be regarded as another example of a temporal processing step. Observe from the middle plot in Figure 9.15 that the temporal step that we are considering in the algorithms listed in Table 9.8 is based on each node  $k$  using its current and past weight estimates, such as  $\{\phi_{k,i}, \phi_{k,i-1}, \dots, \phi_{k,i-P+1}\}$ , rather than only  $\phi_{k,i}$  and current and past raw data  $\{d_k(i), d_k(i-1), \dots, d_k(i-P+1), u_{k,i}, u_{k,i-1}, \dots, u_{k,i-P+1}\}$ . For this reason, the temporal processing steps in Table 9.8 tend to exploit information from across the network more broadly and the resulting mean-square error performance is generally improved relative to (9.500)–(9.502).

### 3.09.10.2 Diffusion recursive least-squares

Diffusion strategies can also be applied to recursive least-squares problems to enable distributed solutions of least-squares designs [38, 39]; see also [74]. Thus, consider again a set of  $N$  nodes that are spatially distributed over some domain. The objective of the network is to collectively estimate some unknown column vector of length  $M$ , denoted by  $w^o$ , using a least-squares criterion. At every time instant  $i$ , each node  $k$  collects a scalar measurement,  $d_k(i)$ , which is assumed to be related to the unknown vector  $w^o$  via the linear model:

$$d_k(i) = u_{k,i} w^o + v_k(i). \quad (9.506)$$

In the above relation, the vector  $u_{k,i}$  denotes a row regression vector of length  $M$ , and  $v_k(i)$  denotes measurement noise. A snapshot of the data in the network at time  $i$  can be captured by collecting the measurements and noise samples,  $\{d_k(i), v_k(i)\}$ , from across all nodes into column vectors  $y_i$  and  $v_i$  of sizes  $N \times 1$  each, and the regressors  $\{u_{k,i}\}$  into a matrix  $H_i$  of size  $N \times M$ :

$$y_i = \begin{bmatrix} d_1(i) \\ d_2(i) \\ \vdots \\ d_N(i) \end{bmatrix} (N \times 1), \quad v_i = \begin{bmatrix} v_1(i) \\ v_2(i) \\ \vdots \\ v_N(i) \end{bmatrix} (N \times 1), \quad H_i = \begin{bmatrix} u_{1,i} \\ u_{2,i} \\ \vdots \\ u_{N,i} \end{bmatrix} (N \times M). \quad (9.507)$$

Likewise, the history of the data across the network up to time  $i$  can be collected into vector quantities as follows:

$$\mathcal{Y}_i = \begin{bmatrix} y_i \\ y_{i-1} \\ \vdots \\ y_0 \end{bmatrix}, \quad \mathcal{V}_i = \begin{bmatrix} v_i \\ v_{i-1} \\ \vdots \\ v_0 \end{bmatrix}, \quad \mathcal{H}_i = \begin{bmatrix} H_i \\ H_{i-1} \\ \vdots \\ H_0 \end{bmatrix}. \quad (9.508)$$

Then, one way to estimate  $w^o$  is by formulating a global least-squares optimization problem of the form:

$$\boxed{\min_w \|w\|_{\Pi_i}^2 + \|\mathcal{Y}_i - \mathcal{H}_i w\|_{\mathcal{W}_i}^2}, \quad (9.509)$$

where  $\Pi_i > 0$  represents a Hermitian regularization matrix and  $\mathcal{W}_i \geq 0$  represents a Hermitian weighting matrix. Common choices for  $\Pi_i$  and  $\mathcal{W}_i$  are

$$\mathcal{W}_i = \text{diag} \left\{ I_N, \lambda I_N, \dots, \lambda^i I_N \right\}, \quad (9.510)$$

$$\Pi_i = \lambda^{i+1} \delta^{-1}, \quad (9.511)$$

where  $\delta > 0$  is usually a large number and  $0 \ll \lambda \leq 1$  is a forgetting factor whose value is generally very close to one. In this case, the global cost function (9.509) can be written in the equivalent form:

$$\boxed{\min_w \lambda^{i+1} \|w\|^2 + \sum_{j=0}^i \lambda^{i-j} \left( \sum_{k=1}^N |d_k(j) - u_{k,j} w|^2 \right)}, \quad (9.512)$$

which is an exponentially weighted least-squares problem. We see that, for every time instant  $j$ , the squared errors,  $|d_k(j) - u_{k,j} w|^2$ , are summed across the network and scaled by the exponential weighting factor  $\lambda^{i-j}$ . The index  $i$  denotes current time and the index  $j$  denotes a time instant in the past. In this way, data occurring in the remote past are scaled more heavily than data occurring closer to present time. The global solution of (9.509) is given by [5]:

$$w_i = [\Pi_i + \mathcal{H}_i \mathcal{W}_i \mathcal{H}_i]^ {-1} \mathcal{H}_i^* \mathcal{W}_i \mathcal{Y}_i \quad (9.513)$$

and the notation  $w_i$ , with a subscript  $i$ , is meant to indicate that the estimate  $w_i$  is based on all data collected from across the network up to time  $i$ . Therefore, the  $w_i$  that is computed via (9.513) amounts to a global construction.

In [38, 39] a diffusion strategy was developed that allows nodes to approach the global solution  $w_i$  by relying solely on local interactions. Let  $w_{k,i}$  denote a local estimate for  $w^o$  that is computed by node  $k$  at time  $i$ . The diffusion recursive-least-squares (RLS) algorithm takes the following form. For every node  $k$ , we start with the initial conditions  $w_{k,-1} = 0$  and  $P_{k,-1} = \delta I_M$ , where  $P_{k,-1}$  is an  $M \times M$  matrix. Then, for every time instant  $i$ , each node  $k$  performs an incremental step followed by a diffusion step as follows:

### Diffusion RLS.

#### Step 1 (incremental update)

$$\psi_{k,i} \leftarrow w_{k,i-1}$$

$$P_{k,i} \leftarrow \lambda^{-1} P_{k,i-1}$$

for every neighboring node  $\ell \in \mathcal{N}_k$ , update:

$$\begin{aligned} \psi_{k,i} &\leftarrow \psi_{k,i} + \frac{c_{\ell k} P_{k,i} u_{\ell,i}^*}{1 + c_{\ell k} u_{\ell,i} P_{k,i} u_{\ell,i}^*} [d_{\ell,i} - u_{\ell,i} \psi_{k,i}] \\ P_{k,i} &\leftarrow P_{k,i} - \frac{c_{\ell k} P_{k,i} u_{\ell,i}^* u_{\ell,i} P_{k,i}}{1 + c_{\ell k} u_{\ell,i} P_{k,i} u_{\ell,i}^*} \end{aligned} \quad (9.514)$$

end

#### Step 2 (diffusion update)

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i},$$

where the symbol  $\leftarrow$  denotes a sequential assignment, and where the scalars  $\{a_{\ell k}, c_{\ell k}\}$  are nonnegative coefficients satisfying:

for  $k = 1, 2, \dots, N$ :

$$c_{\ell k} \geq 0, \quad \sum_{k=1}^N c_{\ell k} = 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \quad (9.515)$$

$$a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \quad (9.516)$$

The above algorithm requires that at every instant  $i$ , nodes communicate to their neighbors their measurements  $\{d_\ell(i), u_{\ell,i}\}$  for the incremental update, and the intermediate estimates  $\{\psi_{\ell,i}\}$  for the diffusion update. During the incremental update, node  $k$  cycles through its neighbors and incorporates their data contributions represented by  $\{d_\ell(i), u_{\ell,i}\}$  into  $\{\psi_{k,i}, P_{k,i}\}$ . Every other node in the network is performing similar steps. At the end of the incremental step, neighboring nodes share their intermediate estimates  $\{\psi_{\ell,i}\}$  to undergo diffusion. Thus, at the end of both steps, each node  $k$  would have updated the quantities  $\{w_{k,i-1}, P_{k,i-1}\}$  to  $\{w_{k,i}, P_{k,i}\}$ . The quantities  $P_{k,i}$  are matrices of size  $M \times M$  each. Observe that the diffusion RLS implementation (9.514) does not require the nodes to share their matrices  $\{P_{\ell,i}\}$ , which would amount to a substantial burden in terms of communications resources since each of these matrices has  $M^2$  entries. Only the quantities  $\{d_\ell(i), u_{\ell,i}, \psi_{\ell,i}\}$  are shared. The mean-square performance and convergence of the diffusion RLS strategy are studied in some detail in [39].

The incremental step of the diffusion RLS strategy (9.514) corresponds to performing a number of  $|\mathcal{N}_k|$  successive least-squares updates starting from the initial conditions  $\{w_{k,i-1}, P_{k,i-1}\}$  and ending with the values  $\{\psi_{k,i}, P_{k,i}\}$  that move onto the diffusion update step. It can be verified from the properties of recursive least-squares solutions [4,5] that these variables satisfy the following equations at the *end* of the incremental stage (step 1):

$$P_{k,i}^{-1} = \lambda P_{k,i-1}^{-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* u_{\ell,i}, \quad (9.517)$$

$$P_{k,i}^{-1} \psi_{k,i} = \lambda P_{k,i-1}^{-1} w_{k,i-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* d_\ell(i). \quad (9.518)$$

Introduce the auxiliary  $M \times 1$  variable:

$$q_{k,i} \triangleq P_{k,i}^{-1} \psi_{k,i}. \quad (9.519)$$

Then, the above expressions lead to the following alternative form of the diffusion RLS strategy (9.514).

Alternative form of diffusion RLS.

$$\begin{aligned} w_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i-1} \\ P_{k,i}^{-1} &= \lambda P_{k,i-1}^{-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* u_{\ell,i} \\ q_{k,i} &= \lambda P_{k,i-1}^{-1} w_{k,i-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* d_\ell(i) \\ \psi_{k,i} &= P_{k,i} q_{k,i} \end{aligned} \quad (9.520)$$

Under some approximations, and for the special choices  $A = C$  and  $\lambda = 1$ , the diffusion RLS strategy (9.520) can be reduced to a form given in [75] and which is described by the following equations:

$$P_{k,i}^{-1} = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \left[ P_{\ell,i-1}^{-1} + u_{\ell,i}^* u_{\ell,i} \right], \quad (9.521)$$

$$q_{k,i} = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \left[ q_{\ell,i-1} + u_{\ell,i}^* d_{\ell}(i) \right], \quad (9.522)$$

$$\psi_{k,i} = P_{k,i} q_{k,i}. \quad (9.523)$$

Algorithm (9.521)–(9.523) is motivated in [75] by using consensus-type arguments. Observe that the algorithm requires the nodes to share the variables  $\{d_{\ell}(i), u_{\ell,i}, q_{\ell,i-1}, P_{\ell,i-1}\}$ , which corresponds to more communications overburden than required by diffusion RLS; the latter only requires that nodes share  $\{d_{\ell}(i), u_{\ell,i}, \psi_{\ell,i-1}\}$ . In order to illustrate how a special case of diffusion RLS (9.520) can be related to this scheme, let us set

$$A = C \quad \text{and} \quad \lambda = 1. \quad (9.524)$$

Then, Eqs. (9.520) give:

Special form of diffusion RLS when  $A = C$  and  $\lambda = 1$ .

$$\begin{aligned} w_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \psi_{\ell,i-1} \\ P_{k,i}^{-1} &= P_{k,i-1}^{-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* u_{\ell,i} \\ q_{k,i} &= P_{k,i-1}^{-1} w_{k,i-1} + \sum_{\ell \in \mathcal{N}_k} c_{\ell k} u_{\ell,i}^* d_{\ell}(i) \\ \psi_{k,i} &= P_{k,i} q_{k,i} \end{aligned} \quad (9.525)$$

Comparing these equations with (9.521)–(9.523), we find that algorithm (9.521)–(9.523) of [75] would relate to the diffusion RLS algorithm (9.520) when the following approximations are justified:

$$\sum_{\ell \in \mathcal{N}_k} c_{\ell k} P_{\ell,i-1}^{-1} \approx P_{k,i-1}^{-1}, \quad (9.526)$$

$$\begin{aligned} \sum_{\ell \in \mathcal{N}_k} c_{\ell k} q_{\ell,i-1} &= \sum_{\ell \in \mathcal{N}_k} c_{\ell k} P_{\ell,i-1}^{-1} \psi_{\ell,i-1} \\ &\approx \sum_{\ell \in \mathcal{N}_k} c_{\ell k} P_{k,i-1}^{-1} \psi_{\ell,i-1} \end{aligned}$$

$$= P_{k,i-1}^{-1} \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \psi_{\ell,i-1} \quad (9.527)$$

$$= P_{k,i-1}^{-1} w_{k,i-1}. \quad (9.528)$$

It was indicated in [39] that the diffusion RLS implementation (9.514) or (9.520) leads to enhanced performance in comparison to the consensus-based update (9.521)–(9.523).

### 3.09.10.3 Diffusion Kalman filtering

Diffusion strategies can also be applied to the solution of distributed state-space filtering and smoothing problems [35, 40, 41]. Here, we describe briefly the diffusion version of the Kalman filter; other variants and smoothing filters can be found in [35]. We assume that some system of interest is evolving according to linear state-space dynamics, and that every node in the network collects measurements that are linearly related to the unobserved state vector. The objective is for every node to track the state of the system over time based solely on local observations and on neighborhood interactions.

Thus, consider a network consisting of  $N$  nodes observing the state vector,  $\mathbf{x}_i$ , of size  $n \times 1$  of a linear state-space model. At every time  $i$ , every node  $k$  collects a measurement vector  $\mathbf{y}_{k,i}$  of size  $p \times 1$ , which is related to the state vector as follows:

$$\mathbf{x}_{i+1} = F_i \mathbf{x}_i + G_i \mathbf{n}_i, \quad (9.529)$$

$$\mathbf{y}_{k,i} = H_{k,i} \mathbf{x}_i + v_{k,i}, \quad k = 1, 2, \dots, N. \quad (9.530)$$

The signals  $\mathbf{n}_i$  and  $v_{k,i}$  denote state and measurement noises of sizes  $n \times 1$  and  $p \times 1$ , respectively, and they are assumed to be zero-mean, uncorrelated and white, with covariance matrices denoted by

$$\mathbb{E} \begin{bmatrix} \mathbf{n}_i \\ v_{k,i} \end{bmatrix} \begin{bmatrix} \mathbf{n}_j \\ v_{k,j} \end{bmatrix}^* \triangleq \begin{bmatrix} Q_i & 0 \\ 0 & R_{k,i} \end{bmatrix} \delta_{ij}. \quad (9.531)$$

The initial state vector,  $\mathbf{x}_o$ , is assumed to be zero-mean with covariance matrix

$$\mathbb{E} \mathbf{x}_o \mathbf{x}_o^* = \Pi_o > 0 \quad (9.532)$$

and is uncorrelated with  $\mathbf{n}_i$  and  $v_{k,i}$ , for all  $i$  and  $k$ . We further assume that  $R_{k,i} > 0$ . The parameter matrices  $\{F_i, G_i, H_{k,i}, Q_i, R_{k,i}, \Pi_o\}$  are assumed to be known by node  $k$ .

Let  $\hat{\mathbf{x}}_{k,i|j}$  denote a local estimator for  $\mathbf{x}_i$  that is computed by node  $k$  at time  $i$  based solely on local observations and on neighborhood data up to time  $j$ . The following diffusion strategy was proposed in [35, 40, 41] to approximate predicted and filtered versions of these local estimators in a distributed manner for data satisfying model (9.529)–(9.532). For every node  $k$ , we start with  $\hat{\mathbf{x}}_{k,0|-1} = 0$  and  $P_{k,0|-1} = \Pi_o$ , where  $P_{k,0|-1}$  is an  $M \times M$  matrix. At every time instant  $i$ , every node  $k$  performs an incremental step followed by a diffusion step:

---

Time and measurement-form of the diffusion Kalman filter.

---

**Step 1** (incremental update)

$$\boldsymbol{\psi}_{k,i} \leftarrow \hat{\mathbf{x}}_{k,i|i-1}$$

$$P_{k,i} \leftarrow P_{k,i|i-1}$$

for every neighboring node  $\ell \in \mathcal{N}_k$ , update:

$$\begin{aligned} R_e &\leftarrow R_{\ell,i} + H_{\ell,i} P_{k,i} H_{\ell,i}^* \\ \boldsymbol{\psi}_{k,i} &\leftarrow \boldsymbol{\psi}_{k,i} + P_{k,i} H_{\ell,i}^* R_e^{-1} [\mathbf{y}_{\ell,i} - H_{\ell,i} \boldsymbol{\psi}_{k,i}] \\ P_{k,i} &\leftarrow P_{k,i} - P_{k,i} H_{\ell,i}^* R_e^{-1} H_{\ell,i} P_{k,i} \end{aligned} \quad (9.533)$$

end

**Step 2** (diffusion update)

$$\hat{\mathbf{x}}_{k,i|i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i}$$

$$P_{k,i|i} = P_{k,i}$$

$$\hat{\mathbf{x}}_{k,i+1|i} = F_i \hat{\mathbf{x}}_{k,i|i}$$

$$P_{k,i+1|i} = F_i P_{k,i|i} F_i^* + G_i Q_i G_i^*.$$


---

where the symbol  $\leftarrow$  denotes a sequential assignment, and where the scalars  $\{a_{\ell k}\}$  are nonnegative coefficients satisfying:

for  $k = 1, 2, \dots, N$ :

$$a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \quad (9.534)$$

The above algorithm requires that at every instant  $i$ , nodes communicate to their neighbors their measurement matrices  $H_{\ell,i}$ , the noise covariance matrices  $R_{\ell,i}$ , and the measurements  $\mathbf{y}_{\ell,i}$  for the incremental update, and the intermediate estimators  $\boldsymbol{\psi}_{\ell,i}$  for the diffusion update. During the incremental update, node  $k$  cycles through its neighbors and incorporates their data contributions represented by  $\{\mathbf{y}_{\ell,i}, H_{\ell,i}, R_{\ell,i}\}$  into  $\{\boldsymbol{\psi}_{k,i}, P_{k,i}\}$ . Every other node in the network is performing similar steps. At the end of the incremental step, neighboring nodes share their updated intermediate estimators  $\{\boldsymbol{\psi}_{\ell,i}\}$  to undergo diffusion. Thus, at the end of both steps, each node  $k$  would have updated the quantities  $\{\hat{\mathbf{x}}_{k,i|i-1}, P_{k,i|i-1}\}$  to  $\{\hat{\mathbf{x}}_{k,i+1|i}, P_{k,i+1|i}\}$ . The quantities  $P_{k,i|i-1}$  are  $n \times n$  matrices. It is important to note that even though the notation  $P_{k,i|i}$  and  $P_{k,i|i-1}$  has been retained for these variables, as in the standard Kalman filtering notation [5, 76], these matrices do *not* represent any longer the covariances of the state estimation errors,  $\tilde{\mathbf{x}}_{k,i|i-1} = \mathbf{x}_i - \hat{\mathbf{x}}_{k,i|i-1}$ , but can be related to them [35].

An alternative representation of the diffusion Kalman filter may be obtained in information form by further assuming that  $P_{k,i|i-1} > 0$  for all  $k$  and  $i$ ; a sufficient condition for this fact to hold is to require the matrices  $\{F_i\}$  to be invertible [76]. Thus, consider again data satisfying model (9.529)–(9.532). For every node  $k$ , we start with  $\hat{\mathbf{x}}_{k,0|-1} = 0$  and  $P_{k,0|-1}^{-1} = \Pi_o^{-1}$ . At every time instant  $i$ , every node  $k$  performs an incremental step followed by a diffusion step:

---

Information form of the diffusion Kalman filter.

---

**Step 1** (incremental update)

$$\begin{aligned} S_{k,i} &= \sum_{\ell \in \mathcal{N}_k} H_{\ell,i}^* R_{\ell,i}^{-1} H_{\ell,i} \\ \mathbf{q}_{k,i} &= \sum_{\ell \in \mathcal{N}_k} H_{\ell,i}^* R_{\ell,i}^{-1} \mathbf{y}_{\ell,i} \\ P_{k,i|i}^{-1} &= P_{k,i|i-1}^{-1} + S_{k,i} \\ \boldsymbol{\psi}_{k,i} &= \hat{\mathbf{x}}_{k,i|i-1} + P_{k,i|i} [\mathbf{q}_{k,i} - S_{k,i} \hat{\mathbf{x}}_{k,i|i-1}] \end{aligned} \quad (9.535)$$

**Step 2** (diffusion update)

$$\begin{aligned} \hat{\mathbf{x}}_{k,i|i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i} \\ \hat{\mathbf{x}}_{k,i+1|i} &= F_i \hat{\mathbf{x}}_{k,i|i} \\ P_{k,i+1|i} &= F_i P_{k,i|i} F_i^* + G_i Q_i G_i^* \end{aligned}$$


---

The incremental update in (9.535) is similar to the update used in the distributed Kalman filter derived in [49]. An important difference in the algorithms is in the diffusion step. Reference [49] starts from a continuous-time consensus implementation and discretizes it to arrive at the following update relation:

$$\hat{\mathbf{x}}_{k,i|i} = \boldsymbol{\psi}_{k,i} + \epsilon \sum_{\ell \in \mathcal{N}_k} (\boldsymbol{\psi}_{\ell,i} - \boldsymbol{\psi}_{k,i}), \quad (9.536)$$

which, in order to facilitate comparison with (9.535), can be equivalently rewritten as:

$$\hat{\mathbf{x}}_{k,i|i} = (1 + \epsilon - n_k \epsilon) \cdot \boldsymbol{\psi}_{k,i} + \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} \epsilon \cdot \boldsymbol{\psi}_{\ell,i}, \quad (9.537)$$

where  $n_k$  denotes the degree of node  $k$  (i.e., the size of its neighborhood,  $\mathcal{N}_k$ ). In comparison, the diffusion step in (9.535) can be written as:

$$\hat{\mathbf{x}}_{k,i|i} = a_{kk} \cdot \boldsymbol{\psi}_{k,i} + \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} a_{\ell k} \cdot \boldsymbol{\psi}_{\ell,i}. \quad (9.538)$$

Observe that the weights used in (9.537) are  $(1 + \epsilon - n_k \epsilon)$  for the node's estimator,  $\boldsymbol{\psi}_{k,i}$ , and  $\epsilon$  for all other estimators,  $\{\boldsymbol{\psi}_{\ell,i}\}$ , arriving from the neighbors of node  $k$ . In contrast, the diffusion step (9.538) employs a convex combination of the estimators  $\{\boldsymbol{\psi}_{\ell,i}\}$  with generally different weights  $\{a_{\ell k}\}$  for different neighbors; this choice is motivated by the desire to employ combination coefficients that enhance the fusion of information at node  $k$ , as suggested by the discussion in Appendix D of [35]. It was verified in [35] that the diffusion implementation (9.538) leads to enhanced performance in comparison

to the consensus-based update (9.537). Moreover, the weights  $\{a_{\ell k}\}$  in (9.538) can also be adjusted over time in order to further enhance performance, as discussed in [77]. The mean-square performance and convergence of the diffusion Kalman filtering implementations are studied in some detail in [35], along with other diffusion strategies for smoothing problems including fixed-point and fixed-lag smoothing.

### 3.09.10.4 Diffusion distributed optimization

The ATC and CTA steepest-descent diffusion strategies (9.134) and (9.142) derived earlier in Section 3.09.3 provide distributed mechanisms for the solution of global optimization problems of the form:

$$\min_w \sum_{k=1}^N J_k(w), \quad (9.539)$$

where the individual costs,  $J_k(w)$ , were assumed to be quadratic in  $w$ , namely,

$$J_k(w) = \sigma_{d,k}^2 - w^* r_{du,k} - r_{du,k}^* w + w^* R_{u,k} w \quad (9.540)$$

for given parameters  $\{\sigma_{d,k}^2, r_{du,k}, R_{u,k}\}$ . Nevertheless, we remarked in that section that similar diffusion strategies can be applied to more general cases involving individual cost functions,  $J_k(w)$ , that are not necessarily quadratic in  $w$  [1–3]. We restate below, for ease of reference, the general ATC and CTA diffusion strategies (9.139) and (9.146) that can be used for the distributed solution of global optimization problems of the form (9.539) for more general convex functions  $J_k(w)$ :

$$\begin{aligned} \psi_{k,i} &= w_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(w_{k,i-1})]^* \\ (\text{ATC strategy}) \quad w_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \end{aligned} \quad (9.541)$$

and

$$\begin{aligned} \psi_{k,i-1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,i-1} \\ (\text{CTA strategy}) \quad w_{k,i} &= \psi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_\ell(\psi_{k,i-1})]^* \end{aligned} \quad (9.542)$$

for positive step-sizes  $\{\mu_k\}$  and for nonnegative coefficients  $\{c_{\ell k}, a_{\ell k}\}$  that satisfy:

$$\begin{aligned} &\text{for } k = 1, 2, \dots, N: \\ c_{\ell k} &\geq 0, \quad \sum_{k=1}^N c_{\ell k} = 1, \quad c_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k, \\ a_{\ell k} &\geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad (9.543)$$

That is, the matrix  $A = [a_{\ell k}]$  is left-stochastic while the matrix  $C = [c_{\ell k}]$  is right-stochastic:

$$C\mathbb{1} = \mathbb{1}, \quad A^T \mathbb{1} = \mathbb{1}. \quad (9.544)$$

We can again regard the above ATC and CTA strategies as special cases of the following general diffusion scheme:

$$\phi_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{1,\ell k} w_{\ell,i-1}, \quad (9.545)$$

$$\psi_{k,i} = \phi_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell k} [\nabla_w J_{\ell}(\phi_{k,i-1})]^*, \quad (9.546)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{2,\ell k} \psi_{\ell,i}, \quad (9.547)$$

where the coefficients  $\{a_{1,\ell k}, a_{2,\ell k}, c_{\ell k}\}$  are nonnegative coefficients corresponding to the  $(l, k)$ th entries of combination matrices  $\{A_1, A_2, C\}$  that satisfy:

$$A_1^T \mathbb{1} = \mathbb{1}, \quad A_2^T \mathbb{1} = \mathbb{1}, \quad C \mathbb{1} = \mathbb{1}. \quad (9.548)$$

The convergence behavior of these diffusion strategies can be examined under both conditions of noiseless updates (when the gradient vectors are available) and noisy updates (when the gradient vectors are subject to gradient noise). The following properties can be proven for the diffusion strategies (9.545)–(9.547) [2]. The statements that follow assume, for convenience of presentation, that all data are *real-valued*; the conditions would need to be adjusted for complex-valued data.

#### 3.09.10.4.1 Noiseless updates

Let

$$J^{\text{glob}}(w) = \sum_{k=1}^N J_k(w) \quad (9.549)$$

denote the global cost function that we wish to minimize. Assume  $J^{\text{glob}}(w)$  is strictly convex so that its minimizer  $w^o$  is unique. Assume further that each individual cost function  $J_k(w)$  is convex and has a minimizer at the *same*  $w^o$ . This case is common in practice; situations abound where nodes in a network need to work cooperatively to attain a common objective (such as tracking a target, locating the source of chemical leak, estimating a physical model, or identifying a statistical distribution). The case where the  $\{J_k(w)\}$  have different individual minimizers is studied in [1,3], where it is shown that the same diffusion strategies of this section are still applicable and nodes would converge instead to a Pareto-optimal solution.

**Theorem 9.10.1 (Convergence to Optimal Solution: Noise-Free Case).** *Consider the problem of minimizing the strictly convex global cost (9.549), with the individual cost functions  $\{J_k(w)\}$  assumed to be convex with each having a minimizer at the same  $w^o$ . Assume that all data are real-valued and suppose the Hessian matrices of the individual costs are bounded from below and from above as follows:*

$$\lambda_{\ell,\min} I_M \leq \nabla_w^2 J_{\ell}(w) \leq \lambda_{\ell,\max} I_M, \quad \ell = 1, 2, \dots, N \quad (9.550)$$

for some positive constants  $\{\lambda_{\ell,\min}, \lambda_{\ell,\max}\}$ . Let

$$\sigma_{k,\min} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \lambda_{\ell,\min}, \quad \sigma_{k,\max} \triangleq \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \lambda_{\ell,\max}. \quad (9.551)$$

Assume further that  $\sigma_{k,\min} > 0$  and that the positive step-sizes are chosen such that:

$$\mu_k \leq \frac{2}{\sigma_{k,\max}}, \quad k = 1, \dots, N. \quad (9.552)$$

Then, it holds that  $w_{k,i} \rightarrow w^o$  as  $i \rightarrow \infty$ . That is, the weight estimates generated by (9.545)–(9.547) at all nodes will tend towards the desired global minimizer.  $\square$

We note that in works on distributed sub-gradient methods (e.g., [36, 78]), the norms of the sub-gradients are usually required to be uniformly bounded. Such a requirement is restrictive in the unconstrained optimization of differentiable functions. Condition (9.550) is more relaxed since it allows the gradient vector  $\nabla_w J_\ell(w)$  to have *unbounded* norm. This extension is important because requiring bounded gradient norms, as opposed to bounded Hessian matrices, would exclude the possibility of using quadratic costs for the  $J_\ell(w)$  (since the gradient vectors would then be unbounded). And, as we saw in the body of the chapter, quadratic costs play a critical role in adaptation and learning over networks.

### 3.09.10.4.2 Updates with gradient noise

It is often the case that we do not have access to the exact gradient vectors to use in (9.546), but to noisy versions of them, say,

$$\widehat{\nabla_w J_\ell(\phi_{k,i-1})} \triangleq \nabla_w J_\ell(\phi_{k,i-1}) + v_\ell(\tilde{\phi}_{k,i-1}), \quad (9.553)$$

where the random vector variable  $v_\ell(\cdot)$  refers to gradient noise; its value is generally dependent on the weight-error vector realization,

$$\tilde{\phi}_{k,i-1} \triangleq w^o - \phi_{k,i-1} \quad (9.554)$$

at which the gradient vector is being evaluated. In the presence of gradient noise, the weight estimates at the various nodes become random quantities and we denote them by the boldface notation  $\{\mathbf{w}_{k,i}\}$ . We assume that, conditioned on the past history of the weight estimators at all nodes, namely,

$$\mathcal{F}_{i-1} \triangleq \{\mathbf{w}_{m,j}, m = 1, 2, \dots, N, j < i\} \quad (9.555)$$

the gradient noise has zero mean and its variance is upper bounded as follows:

$$\mathbb{E}\{v_\ell(\tilde{\phi}_{k,i-1})|\mathcal{F}_{i-1}\} = 0, \quad (9.556)$$

$$\mathbb{E}\{\|v_\ell(\tilde{\phi}_{k,i-1})\|^2|\mathcal{F}_{i-1}\} \leq \alpha \|\tilde{\phi}_{k,i-1}\|^2 + \sigma_v^2 \quad (9.557)$$

for some  $\alpha > 0$  and  $\sigma_v^2 \geq 0$ . Condition (9.557) allows the variance of the gradient noise to be time-variant, so long as it does not grow faster than  $\mathbb{E}\|\tilde{\phi}_{k,i-1}\|^2$ . This condition on the noise is more general than the “uniform-bounded assumption” that appears in [36], which required instead:

$$\mathbb{E}\{\|v_\ell(\tilde{\phi}_{k,i-1})\|^2\} \leq \sigma_v^2, \quad \mathbb{E}\{\|v_\ell(\tilde{\phi}_{k,i-1})\|^2|\mathcal{F}_{i-1}\} \leq \sigma_v^2. \quad (9.558)$$

These two requirements are special cases of (9.557) for  $\alpha = 0$ . Furthermore, condition (9.557) is similar to condition (4.3) in [79], which requires the noise variance to satisfy:

$$\mathbb{E}\{\|v_\ell(\tilde{\phi}_{k,i-1})\|^2|\mathcal{F}_{i-1}\} \leq \alpha[\|\nabla_w J_\ell(\phi_{k,i-1})\|^2 + 1]. \quad (9.559)$$

This requirement can be verified to be a combination of the “relative random noise” and the “absolute random noise” conditions defined in [22]—see [2].

Now, introduce the column vector:

$$\mathbf{z}_i \triangleq \sum_{\ell=1}^N \text{col}\{c_{\ell 1} \mathbf{v}_{\ell}(w^o), c_{\ell 2} \mathbf{v}_{\ell}(w^o), \dots, c_{\ell N} \mathbf{v}_{\ell}(w^o)\} \quad (9.560)$$

and let

$$\mathcal{Z} = \mathbb{E} \mathbf{z}_i \mathbf{z}_i^* \quad (9.561)$$

Let further

$$\tilde{\mathbf{w}}_i \triangleq \text{col}\{\tilde{\mathbf{w}}_{i,1}, \tilde{\mathbf{w}}_{i,2}, \dots, \tilde{\mathbf{w}}_{i,N}\}, \quad (9.562)$$

where

$$\tilde{\mathbf{w}}_{k,i} \triangleq w^o - \mathbf{w}_{k,i}. \quad (9.563)$$

Then, the following result can be established [2]; it characterizes the network mean-square deviation in steady-state, which is defined as

$$\text{MSD}^{\text{network}} \triangleq \lim_{i \rightarrow \infty} \left( \frac{1}{N} \sum_{k=1}^N \mathbb{E} \|\tilde{\mathbf{w}}_{k,i}\|^2 \right). \quad (9.564)$$

**Theorem 9.10.2 (Mean-Square Stability: Noisy Case).** *Consider the problem of minimizing the strictly convex global cost (9.549), with the individual cost functions  $\{J_k(w)\}$  assumed to be convex with each having a minimizer at the same  $w^o$ . Assume all data are real-valued and suppose the Hessian matrices of the individual costs are bounded from below and from above as stated in (9.550). Assume further that the diffusion strategy (9.545)–(9.547) employs noisy gradient vectors, where the noise terms are zero mean and satisfy conditions (9.557) and (9.561). We select the positive step-sizes to be sufficiently small and to satisfy:*

$$\mu_k < \min \left\{ \frac{2\sigma_{k,\max}}{\sigma_{k,\max}^2 + \alpha \|C\|_1^2}, \frac{2\sigma_{k,\min}}{\sigma_{k,\min}^2 + \alpha \|C\|_1^2} \right\} \quad (9.565)$$

for  $k = 1, 2, \dots, N$ . Then, the diffusion strategy (9.545)–(9.547) is mean-square stable and the mean-square-deviation of the network is given by:

$$\text{MSD}^{\text{network}} \approx \frac{1}{N} [\text{vec}(\mathcal{A}_2^T \mathcal{M} \mathcal{Z}^T \mathcal{M} \mathcal{A}_2)]^T \cdot (I - \mathcal{F})^{-1} \cdot \text{vec}(I_{NM}), \quad (9.566)$$

where

$$\mathcal{A}_2 = \mathcal{A}_2 \otimes I_M, \quad (9.567)$$

$$\mathcal{M} = \text{diag}\{\mu_1 I_M, \mu_2 I_M, \dots, \mu_N I_M\}, \quad (9.568)$$

$$\mathcal{F} \approx \mathcal{B}^T \otimes \mathcal{B}^*, \quad (9.569)$$

$$\mathcal{B} = \mathcal{A}_2^T (I - \mathcal{M} \mathcal{R}) \mathcal{A}_1^T, \quad (9.570)$$

$$\mathcal{R} = \sum_{\ell=1}^N \text{diag} \left\{ c_{\ell 1} \nabla_w^2 J_{\ell}(w^o), c_{\ell 2} \nabla_w^2 J_{\ell}(w^o), \dots, c_{\ell N} \nabla_w^2 J_{\ell}(w^o) \right\}. \quad (9.571)$$

□

## Appendices

### A Properties of Kronecker products

For ease of reference, we collect in this appendix some useful properties of Kronecker products. All matrices are assumed to be of compatible dimensions; all inverses are assumed to exist whenever necessary. Let  $E = [e_{ij}]_{i,j=1}^n$  and  $B = [b_{ij}]_{i,j=1}^m$  be  $n \times n$  and  $m \times m$  matrices, respectively. Their Kronecker product is denoted by  $E \otimes B$  and is defined as the  $nm \times nm$  matrix whose entries are given by [20]:

$$E \otimes B = \begin{bmatrix} e_{11}B & e_{12}B & \dots & e_{1n}B \\ e_{21}B & e_{22}B & \dots & e_{2n}B \\ \vdots & & \vdots & \\ e_{n1}B & e_{n2}B & \dots & e_{nn}B \end{bmatrix}. \quad (9.572)$$

In other words, each entry of  $E$  is replaced by a scaled multiple of  $B$ . Let  $\{\lambda_i(E), i = 1, \dots, n\}$  and  $\{\lambda_j(B), j = 1, \dots, m\}$  denote the eigenvalues of  $E$  and  $B$ , respectively. Then, the eigenvalues of  $E \otimes B$  will consist of all  $nm$  product combinations  $\{\lambda_i(E)\lambda_j(B)\}$ . Table 9.9 lists some well-known properties of Kronecker products.

**Table 9.9** Properties of Kronecker Products

$$(E + B) \otimes C = (E \otimes C) + (B \otimes C)$$

$$(E \otimes B)(C \otimes D) = (EC \otimes BD)$$

$$(E \otimes B)^T = E^T \otimes B^T$$

$$(E \otimes B)^* = E^* \otimes B^*$$

$$(E \otimes B)^{-1} = E^{-1} \otimes B^{-1}$$

$$(E \otimes B)^\ell = E^\ell \otimes B^\ell$$

$$\{\lambda(E \otimes B)\} = \{\lambda_i(E)\lambda_j(B)\}_{i=1,j=1}^{n,m}$$

$$\det(E \otimes B) = (\det E)^m(\det B)^n$$

$$\text{Tr}(E \otimes B) = \text{Tr}(E)\text{Tr}(B)$$

$$\text{Tr}(EB) = [\text{vec}(B^T)]^T \text{vec}(E)$$

$$\text{vec}(ECB) = (B^T \otimes E)\text{vec}(C)$$

### B Graph Laplacian and network connectivity

Consider a network consisting of  $N$  nodes and  $L$  edges connecting the nodes to each other. In the constructions below, we only need to consider the edges that connect distinct nodes to each other; these

edges do not contain any self-loops that may exist in the graph and which connect nodes to themselves directly. In other words, when we refer to the  $L$  edges of a graph, we are excluding self-loops from this set; but we are still allowing loops of at least length 2 (i.e., loops generated by paths covering at least 2 edges).

The neighborhood of any node  $k$  is denoted by  $\mathcal{N}_k$  and it consists of all nodes that node  $k$  can share information with; these are the nodes that are connected to  $k$  through edges, in addition to node  $k$  itself. The degree of node  $k$ , which we denote by  $n_k$ , is defined as the positive integer that is equal to the size of its neighborhood:

$$n_k \triangleq |\mathcal{N}_k|. \quad (9.573)$$

Since  $k \in \mathcal{N}_k$ , we always have  $n_k \geq 1$ . We further associate with the network an  $N \times N$  Laplacian matrix, denoted by  $\mathcal{L}$ . The matrix  $\mathcal{L}$  is symmetric and its entries are defined as follows [60–62]:

$$[\mathcal{L}]_{k\ell} = \begin{cases} n_k - 1, & \text{if } k = \ell, \\ -1, & \text{if } k \neq \ell \text{ and nodes } k \text{ and } \ell \text{ are neighbors,} \\ 0, & \text{otherwise.} \end{cases} \quad (9.574)$$

Note that the term  $n_k - 1$  measures the number of edges that are incident on node  $k$ , and the locations of the  $-1$ 's on row  $k$  indicate the nodes that are connected to node  $k$ . We also associate with the graph an  $N \times L$  incidence matrix, denoted by  $\mathcal{I}$ . The entries of  $\mathcal{I}$  are defined as follows. Every column of  $\mathcal{I}$  represents one edge in the graph. Each edge connects two nodes and its column will display two nonzero entries at the rows corresponding to these nodes: one entry will be  $+1$  and the other entry will be  $-1$ . For directed graphs, the choice of which entry is positive or negative can be used to identify the nodes from which edges emanate (source nodes) and the nodes at which edges arrive (sink nodes). Since we are dealing with undirected graphs, we shall simply assign positive values to lower indexed nodes and negative values to higher indexed nodes:

$$[\mathcal{I}]_{ke} = \begin{cases} +1, & \text{if node } k \text{ is the lower-indexed node connected to edge } e, \\ -1, & \text{if node } k \text{ is the higher-indexed node connected to edge } e, \\ 0, & \text{otherwise.} \end{cases} \quad (9.575)$$

Figure 9.16 shows the example of a network with  $N = 6$  nodes and  $L = 8$  edges. Its Laplacian and incidence matrices are also shown and these have sizes  $6 \times 6$  and  $6 \times 8$ , respectively. Consider, for example, column 6 in the incidence matrix. This column corresponds to edge 6, which links nodes 3 and 5. Therefore, at location  $\mathcal{I}_{36}$  we have a  $+1$  and at location  $\mathcal{I}_{56}$  we have  $-1$ . The other columns of  $\mathcal{I}$  are constructed in a similar manner.

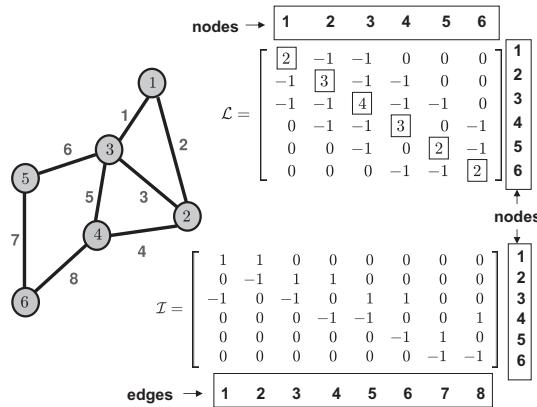
Observe that the Laplacian and incidence matrices of a graph are related as follows:

$$\mathcal{L} = \mathcal{I}\mathcal{I}^T. \quad (9.576)$$

The Laplacian matrix conveys useful information about the topology of the graph. The following is a classical result from graph theory [60–62, 80].

**Lemma B.1 (Laplacian and Network Connectivity).** *Let*

$$\theta_1 \geq \theta_2 \geq \cdots \geq \theta_N \quad (9.577)$$

**FIGURE 9.16**

A network with  $N = 6$  nodes and  $L = 8$  edges. The nodes are marked 1 through 6 and the edges are marked 1 through 8. The corresponding Laplacian and incidence matrices  $\mathcal{L}$  and  $\mathcal{I}$  are  $6 \times 6$  and  $6 \times 8$ .

denote the ordered eigenvalues of  $\mathcal{L}$ . Then the following properties hold:

- a.  $\mathcal{L}$  is symmetric nonnegative-definite so that  $\theta_i \geq 0$ .
- b. The rows of  $\mathcal{L}$  add up to zero so that  $\mathcal{L}\mathbb{1} = 0$ . This means that  $\mathbb{1}$  is a right eigenvector of  $\mathcal{L}$  corresponding to the eigenvalue zero.
- c. The smallest eigenvalue is always zero,  $\theta_N = 0$ . The second smallest eigenvalue,  $\theta_{N-1}$ , is called the algebraic connectivity of the graph.
- d. The number of times that zero is an eigenvalue of  $\mathcal{L}$  (i.e., its multiplicity) is equal to the number of connected subgraphs.
- e. The algebraic connectivity of a connected graph is nonzero, i.e.,  $\theta_{N-1} \neq 0$ . In other words, a graph is connected if, and only if, its algebraic connectivity is nonzero.

**Proof.** Property (a) follows from the identity  $\mathcal{L} = \mathcal{I}\mathcal{I}^T$ . Property (b) follows from the definition of  $\mathcal{L}$ . Note that for each row of  $\mathcal{L}$ , the entries on the row add up to zero. Property (c) follows from properties (a) and (b) since  $\mathcal{L}\mathbb{1} = 0$  implies that zero is an eigenvalue of  $\mathcal{L}$ . For part (d), assume the network consists of two separate connected subgraphs. Then, the Laplacian matrix would have a block diagonal structure, say, of the form  $\mathcal{L} = \text{diag}\{\mathcal{L}_1, \mathcal{L}_2\}$ , where  $\mathcal{L}_1$  and  $\mathcal{L}_2$  are the Laplacian matrices of the smaller subgraphs. The smallest eigenvalue of each of these Laplacian matrices would in turn be zero and unique by property (e). More generally, if the graph consists of  $m$  connected subgraphs, then the multiplicity of zero as an eigenvalue of  $\mathcal{L}$  would be  $m$ . To establish property (e), first observe that if the algebraic connectivity is nonzero then it is obvious that the graph must be connected; otherwise, the Laplacian matrix would be block diagonal and  $\theta_{N-1}$  would be zero (a contradiction). For the converse statement, assume the graph is connected and let  $x$  denote an arbitrary eigenvector of  $\mathcal{L}$  corresponding to the eigenvalue at zero. Then,  $x^T \mathcal{L} x = 0$ , from which we can conclude that all entries of  $x$  must be equal. Therefore, the algebraic multiplicity of the eigenvalue of  $\mathcal{L}$  at zero is equal to one and the algebraic connectivity of the graph must be nonzero.  $\square$

---

## C Stochastic matrices

Consider  $N \times N$  matrices  $A$  with nonnegative entries,  $\{a_{\ell k} \geq 0\}$ . The matrix  $A = [a_{\ell k}]$  is said to be right-stochastic if it satisfies

$$A\mathbb{1} = \mathbb{1} \quad (\text{right-stochastic}), \quad (9.578)$$

in which case each row of  $A$  adds up to one. The matrix  $A$  is said to be left-stochastic if it satisfies

$$A^T\mathbb{1} = \mathbb{1} \quad (\text{left-stochastic}), \quad (9.579)$$

in which case each column of  $A$  adds up to one. And the matrix is said to be doubly stochastic if both conditions hold so that both its columns and rows add up to one:

$$A\mathbb{1} = \mathbb{1}, \quad A^T\mathbb{1} = \mathbb{1} \quad (\text{doubly-stochastic}). \quad (9.580)$$

Stochastic matrices arise frequently in the study of adaptation over networks. This appendix lists some of their properties.

**Lemma C.1 (Spectral Norm of Stochastic Matrices).** *Let  $A$  be an  $N \times N$  right or left or doubly stochastic matrix. Then,  $\rho(A) = 1$  and, therefore, all eigenvalues of  $A$  lie inside the unit disc, i.e.,  $|\lambda(A)| \leq 1$ .*

**Proof.** We prove the result for right stochastic matrices; a similar argument applies to left or doubly stochastic matrices. Let  $A$  be a right-stochastic matrix. Then,  $A\mathbb{1} = \mathbb{1}$ , so that  $\lambda = 1$  is one of the eigenvalues of  $A$ . Moreover, for any matrix  $A$ , it holds that  $\rho(A) \leq \|A\|_\infty$ , where  $\|\cdot\|_\infty$  denotes the maximum absolute row sum of its matrix argument. But since all rows of  $A$  add up to one, we have  $\|A\|_\infty = 1$ . Therefore,  $\rho(A) \leq 1$ . And since we already know that  $A$  has an eigenvalue at  $\lambda = 1$ , we conclude that  $\rho(A) = 1$ .  $\square$

The above result asserts that the spectral radius of a stochastic matrix is unity and that  $A$  has an eigenvalue at  $\lambda = 1$ . The result, however, does not rule out the possibility of multiple eigenvalues at  $\lambda = 1$ , or even other eigenvalues with magnitude equal to one. Assume, in addition, that the stochastic matrix  $A$  is *regular*. This means that there exists an integer power  $j_o$  such that all entries of  $A^{j_o}$  are *strictly* positive, i.e.,

$$\text{for all } (\ell, k), \text{ it holds that } [A^{j_o}]_{\ell k} > 0, \quad \text{for some } j_o > 0. \quad (9.581)$$

Then a result in matrix theory known as the Perron-Frobenius Theorem [20] leads to the following stronger characterization of the eigen-structure of  $A$ .

**Lemma C.2 (Spectral Norm of Regular Stochastic Matrices).** *Let  $A$  be an  $N \times N$  right stochastic and regular matrix. Then:*

- a.  $\rho(A) = 1$ .
- b. All other eigenvalues of  $A$  are strictly inside the unit circle (and, hence, have magnitude strictly less than one).

- c. The eigenvalue at  $\lambda = 1$  is simple, i.e., it has multiplicity one. Moreover, with proper sign scaling, all entries of the corresponding eigenvector are positive. For a right-stochastic  $A$ , this eigenvector is the vector  $\mathbb{1}$  since  $A\mathbb{1} = \mathbb{1}$ .
- d. All other eigenvectors associated with the other eigenvalues will have at least one negative or complex entry.

**Proof.** Part (a) follows from Lemma C.1. Parts (b) and (d) follow from the Perron-Frobenius Theorem when  $A$  is regular [20].  $\square$

**Lemma C.3 (Useful Properties of Doubly Stochastic Matrices).** *Let  $A$  be an  $N \times N$  doubly stochastic matrix. Then the following properties hold:*

- a.  $\rho(A) = 1$ .
- b.  $AA^T$  and  $A^T A$  are doubly stochastic as well.
- c.  $\rho(AA^T) = \rho(A^T A) = 1$ .
- d. The eigenvalues of  $AA^T$  or  $A^T A$  are real and lie inside the interval  $[0, 1]$ .
- e.  $I - AA^T \geq 0$  and  $I - A^T A \geq 0$ .
- f.  $\text{Tr}(A^T H A) \leq \text{Tr}(H)$ , for any  $N \times N$  nonnegative-definite Hermitian matrix  $H$ .

**Proof.** Part (a) follows from Lemma C.1. For part (b), note that  $AA^T$  is symmetric and  $AA^T\mathbb{1} = A\mathbb{1} = \mathbb{1}$ . Therefore,  $AA^T$  is doubly stochastic. Likewise for  $A^T A$ . Part (c) follows from part (a) since  $AA^T$  and  $A^T A$  are themselves doubly stochastic matrices. For part (d), note that  $AA^T$  is symmetric and nonnegative-definite. Therefore, its eigenvalues are real and nonnegative. But since  $\rho(AA^T) = 1$ , we must have  $\lambda(AA^T) \in [0, 1]$ . Likewise for the matrix  $A^T A$ . Part (e) follows from part (d). For part (f), since  $AA^T \geq 0$  and its eigenvalues lie within  $[0, 1]$ , the matrix  $AA^T$  admits an eigen-decomposition of the form:

$$AA^T = U\Lambda U^T,$$

where  $U$  is orthogonal (i.e.,  $U^{-1} = U^T$ ) and  $\Lambda$  is diagonal with entries in the range  $[0, 1]$ . It then follows that

$$\begin{aligned} \text{Tr}(A^T H A) &= \text{Tr}(AA^T H) \\ &= \text{Tr}(U\Lambda U^T H) \\ &= \text{Tr}(\Lambda U^T H U) \\ &\stackrel{(*)}{\leq} \text{Tr}(U^T H U) \\ &= \text{Tr}(U U^T H) \\ &= \text{Tr}(H), \end{aligned}$$

where step (\*) is because  $U^T H U = U^{-1} H U$  and, by similarity, the matrix  $U^{-1} H U$  has the same eigenvalues as  $H$ . Therefore,  $U^T H U \geq 0$ . This means that the diagonal entries of  $U^T H U$  are nonnegative. Multiplying  $U^T H U$  by  $\Lambda$  ends up scaling the nonnegative diagonal entries to smaller values so that (\*) is justified.  $\square$

---

## D Block maximum norm

Let  $x = \text{col}\{x_1, x_2, \dots, x_N\}$  denote an  $N \times 1$  block column vector whose individual entries are of size  $M \times 1$  each. Following [23, 63, 81], the block maximum norm of  $x$  is denoted by  $\|x\|_{b,\infty}$  and is defined as

$$\|x\|_{b,\infty} \triangleq \max_{1 \leq k \leq N} \|x_k\|, \quad (9.582)$$

where  $\|\cdot\|$  denotes the Euclidean norm of its vector argument. Correspondingly, the induced block maximum norm of an arbitrary  $N \times N$  block matrix  $\mathcal{A}$ , whose individual block entries are of size  $M \times M$  each, is defined as

$$\|\mathcal{A}\|_{b,\infty} \triangleq \max_{x \neq 0} \frac{\|\mathcal{A}x\|_{b,\infty}}{\|x\|_{b,\infty}}. \quad (9.583)$$

The block maximum norm inherits the unitary invariance property of the Euclidean norm, as the following result indicates [63].

**Lemma D.1 (Unitary Invariance).** *Let  $\mathcal{U} = \text{diag}\{U_1, U_2, \dots, U_N\}$  be an  $N \times N$  block diagonal matrix with  $M \times M$  unitary blocks  $\{U_k\}$ . Then, the following properties hold:*

- a.  $\|\mathcal{U}x\|_{b,\infty} = \|x\|_{b,\infty}$
- b.  $\|\mathcal{U}\mathcal{A}\mathcal{U}^*\|_{b,\infty} = \|\mathcal{A}\|_{b,\infty}$

for all block vectors  $x$  and block matrices  $\mathcal{A}$  of appropriate dimensions.  $\square$

The next result provides useful bounds for the block maximum norm of a block matrix.

**Lemma D.2 (Useful Bounds).** *Let  $\mathcal{A}$  be an arbitrary  $N \times N$  block matrix with blocks  $A_{\ell k}$  of size  $M \times M$  each. Then, the following results hold:*

- a. *The norms of  $\mathcal{A}$  and its complex conjugate are related as follows:*

$$\|\mathcal{A}^*\|_{b,\infty} \leq N \cdot \|\mathcal{A}\|_{b,\infty}. \quad (9.584)$$

- b. *The norm of  $\mathcal{A}$  is bounded as follows:*

$$\max_{1 \leq \ell, k \leq N} \|A_{\ell k}\| \leq \|\mathcal{A}\|_{b,\infty} \leq N \cdot \left( \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\| \right), \quad (9.585)$$

where  $\|\cdot\|$  denotes the 2-induced norm (or maximum singular value) of its matrix argument.

- c. *If  $\mathcal{A}$  is Hermitian and nonnegative-definite ( $\mathcal{A} \geq 0$ ), then there exist finite positive constants  $c_1$  and  $c_2$  such that*

$$c_1 \cdot \text{Tr}(\mathcal{A}) \leq \|\mathcal{A}\|_{b,\infty} \leq c_2 \cdot \text{Tr}(\mathcal{A}). \quad (9.586)$$

**Proof.** Part (a) follows directly from part (b) by noting that

$$\begin{aligned} \|\mathcal{A}^*\|_{b,\infty} &\leq N \cdot \left( \max_{1 \leq \ell, k \leq N} \|A_{\ell k}^*\| \right) \\ &= N \cdot \left( \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\| \right) \\ &\leq N \cdot \|\mathcal{A}\|_{b,\infty}, \end{aligned}$$

where the equality in the second step is because  $\|A_{\ell k}^*\| = \|A_{\ell k}\|$ ; i.e., complex conjugation does not alter the 2-induced norm of a matrix.

To establish part (b), we consider arbitrary  $N \times 1$  block vectors  $x$  with entries  $x = \text{col}\{x_1, x_2, \dots, x_N\}$  and where each  $x_k$  is  $M \times 1$ . Then, note that

$$\begin{aligned}\|\mathcal{A}x\|_{b,\infty} &= \max_{1 \leq \ell \leq N} \left\| \sum_{k=1}^N A_{\ell k} x_k \right\| \\ &\leq \max_{1 \leq \ell \leq N} \left( \sum_{k=1}^N \|A_{\ell k}\| \cdot \|x_k\| \right) \\ &\leq \left( \max_{1 \leq \ell \leq N} \sum_{k=1}^N \|A_{\ell k}\| \right) \cdot \max_{1 \leq k \leq N} \|x_k\| \\ &\leq \left( \max_{1 \leq \ell \leq N} \sum_{k=1}^N \max_{1 \leq k \leq N} \|A_{\ell k}\| \right) \cdot \|x\|_{b,\infty} \\ &\leq N \cdot \left( \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\| \right) \cdot \|x\|_{b,\infty},\end{aligned}$$

so that

$$\|\mathcal{A}\|_{b,\infty} \triangleq \max_{x \neq 0} \frac{\|\mathcal{A}x\|_{b,\infty}}{\|x\|_{b,\infty}} \leq N \cdot \left( \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\| \right),$$

which establishes the upper bound in (9.585).

To establish the lower bound, we assume without loss of generality that  $\max_{1 \leq \ell, k \leq N} \|A_{\ell k}\|$  is attained at  $\ell = 1$  and  $k = 1$ . Let  $\sigma_1$  denote the largest singular value of  $A_{11}$  and let  $\{v_1, u_1\}$  denote the corresponding  $M \times 1$  right and left singular vectors. That is,

$$\|A_{11}\| = \sigma_1, \quad A_{11}v_1 = \sigma_1 u_1, \tag{9.587}$$

where  $v_1$  and  $u_1$  have unit norms. We now construct an  $N \times 1$  block vector  $x^o$  as follows:

$$x^o \triangleq \text{col}\{v_1, 0_M, 0_M, \dots, 0_M\}. \tag{9.588}$$

Then, obviously,

$$\|x^o\|_{b,\infty} = 1 \tag{9.589}$$

and

$$\mathcal{A}x^o = \text{col}\{A_{11}v_1, A_{21}v_1, \dots, A_{N1}v_1\}. \tag{9.590}$$

It follows that

$$\begin{aligned}
\|\mathcal{A}x^o\|_{b,\infty} &= \max\{\|A_{11}v_1\|, \|A_{21}v_1\|, \dots, \|A_{N1}v_1\|\} \\
&\geq \|A_{11}v_1\| \\
&= \|\sigma_1 u_1\| \\
&= \sigma_1 \\
&= \|A_{11}\| \\
&= \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\|. 
\end{aligned} \tag{9.591}$$

Therefore, by the definition of the block maximum norm,

$$\begin{aligned}
\|\mathcal{A}\|_{b,\infty} &\triangleq \max_{x \neq 0} \left( \frac{\|\mathcal{A}x\|_{b,\infty}}{\|x\|_{b,\infty}} \right) \\
&\geq \frac{\|\mathcal{A}x^o\|_{b,\infty}}{\|x^o\|_{b,\infty}} \\
&= \|\mathcal{A}x^o\|_{b,\infty} \\
&\geq \max_{1 \leq \ell, k \leq N} \|A_{\ell k}\|,
\end{aligned} \tag{9.592}$$

which establishes the lower bound in (9.585).

To establish part (c), we start by recalling that all norms on finite-dimensional vector spaces are equivalent [20, 21]. This means that if  $\|\cdot\|_a$  and  $\|\cdot\|_d$  denote two different matrix norms, then there exist positive constants  $c_1$  and  $c_2$  such that for any matrix  $X$ ,

$$c_1 \cdot \|X\|_a \leq \|X\|_d \leq c_2 \cdot \|X\|_a. \tag{9.593}$$

Now, let  $\|\mathcal{A}\|_*$  denote the nuclear norm of the square matrix  $\mathcal{A}$ . It is defined as the sum of its singular values:

$$\|\mathcal{A}\|_* \triangleq \sum_m \sigma_m(\mathcal{A}). \tag{9.594}$$

Since  $\mathcal{A}$  is Hermitian and nonnegative-definite, its eigenvalues coincide with its singular values and, therefore,

$$\|\mathcal{A}\|_* = \sum_m \lambda_m(\mathcal{A}) = \text{Tr}(\mathcal{A}).$$

Now applying result (9.593) to the two norms  $\|\mathcal{A}\|_{b,\infty}$  and  $\|\mathcal{A}\|_*$  we conclude that

$$c_1 \cdot \text{Tr}(\mathcal{A}) \leq \|\mathcal{A}\|_{b,\infty} \leq c_2 \cdot \text{Tr}(\mathcal{A}), \tag{9.595}$$

as desired.  $\square$

The next result relates the block maximum norm of an extended matrix to the  $\infty$ -norm (i.e., maximum absolute row sum) of the originating matrix. Specifically, let  $A$  be an  $N \times N$  matrix with bounded entries and introduce the block matrix

$$\mathcal{A} \triangleq A \otimes I_M. \tag{9.596}$$

The extended matrix  $\mathcal{A}$  has blocks of size  $M \times M$  each.

**Lemma D.3 (Relation to Maximum Absolute Row Sum).** *Let  $\mathcal{A}$  and  $A$  be related as in (9.596). Then, the following properties hold:*

- a.  $\|\mathcal{A}\|_{b,\infty} = \|A\|_\infty$ , where the notation  $\|\cdot\|_\infty$  denotes the maximum absolute row sum of its argument.
- b.  $\|\mathcal{A}^*\|_{b,\infty} \leq N \cdot \|\mathcal{A}\|_{b,\infty}$ .

**Proof.** The results are obvious for a zero matrix  $A$ . So assume  $A$  is nonzero. Let  $x = \text{col}\{x_1, x_2, \dots, x_N\}$  denote an arbitrary  $N \times 1$  block vector whose individual entries  $\{x_k\}$  are vectors of size  $M \times 1$  each. Then,

$$\begin{aligned} \|\mathcal{A}x\|_{b,\infty} &= \max_{1 \leq k \leq N} \left\| \sum_{\ell=1}^N a_{k\ell} x_\ell \right\| \\ &\leq \max_{1 \leq k \leq N} \left( \sum_{\ell=1}^N |a_{k\ell}| \cdot \|x_\ell\| \right) \\ &\leq \left( \max_{1 \leq k \leq N} \sum_{\ell=1}^N |a_{k\ell}| \right) \cdot \max_{1 \leq \ell \leq N} \|x_\ell\| \\ &= \|A\|_\infty \cdot \|x\|_{b,\infty}, \end{aligned} \tag{9.597}$$

so that

$$\|\mathcal{A}\|_{b,\infty} \triangleq \max_{x \neq 0} \frac{\|\mathcal{A}x\|_{b,\infty}}{\|x\|_{b,\infty}} \leq \|A\|_\infty. \tag{9.598}$$

The argument so far establishes that  $\|\mathcal{A}\|_{b,\infty} \leq \|A\|_\infty$ . Now, let  $k_o$  denote the row index that corresponds to the maximum absolute row sum of  $A$ , i.e.,

$$\|A\|_\infty = \sum_{\ell=1}^N |a_{k_o\ell}|.$$

We construct an  $N \times 1$  block vector  $z = \text{col}\{z_1, z_2, \dots, z_N\}$ , whose  $M \times 1$  entries  $\{z_\ell\}$  are chosen as follows:

$$z_\ell = \text{sign}(a_{k_o\ell}) \cdot e_1,$$

where  $e_1$  is the  $M \times 1$  basis vector:

$$e_1 = \text{col}\{1, 0, 0, \dots, 0\}$$

and the sign function is defined as

$$\text{sign}(a) = \begin{cases} 1, & \text{if } a \geq 0, \\ -1, & \text{otherwise.} \end{cases}$$

Then, note that  $z \neq 0$  for any nonzero matrix  $A$ , and

$$\|z\|_{b,\infty} = \max_{1 \leq \ell \leq N} \|z_\ell\| = 1.$$

Moreover,

$$\begin{aligned}
 \|\mathcal{A}\|_{b,\infty} &\triangleq \max_{x \neq 0} \frac{\|\mathcal{A}x\|_{b,\infty}}{\|x\|_{b,\infty}} \\
 &\geq \frac{\|\mathcal{A}z\|_{b,\infty}}{\|z\|_{b,\infty}} \\
 &= \|\mathcal{A}z\|_{b,\infty} \\
 &= \max_{1 \leq k \leq N} \left\| \sum_{\ell=1}^N a_{k\ell} z_\ell \right\| \\
 &\geq \left\| \sum_{\ell=1}^N a_{k_o \ell} z_\ell \right\| \\
 &= \left\| \sum_{\ell=1}^N a_{k_o \ell} \cdot \text{sign}(a_{k_o \ell}) e_1 \right\| \\
 &= \sum_{\ell=1}^N |a_{k_o \ell}| \cdot \|e_1\| \\
 &= \sum_{\ell=1}^N |a_{k_o \ell}| \\
 &= \|A\|_\infty. \tag{9.599}
 \end{aligned}$$

Combining this result with (9.598) we conclude that  $\|\mathcal{A}\|_{b,\infty} = \|A\|_\infty$ , which establishes part (a). Part (b) follows from the statement of part (a) in Lemma D.2.  $\square$

The next result establishes a useful property for the block maximum norm of right or left stochastic matrices; such matrices arise as combination matrices for distributed processing over networks as in (9.166) and (9.185).

**Lemma D.4 (Right and Left Stochastic Matrices).** *Let  $C$  be an  $N \times N$  right stochastic matrix, i.e., its entries are nonnegative and it satisfies  $C\mathbb{1} = \mathbb{1}$ . Let  $A$  be an  $N \times N$  left stochastic matrix, i.e., its entries are nonnegative and it satisfies  $A^T\mathbb{1} = \mathbb{1}$ . Introduce the block matrices*

$$\mathcal{A}^T \triangleq A^T \otimes I_M, \quad \mathcal{C} \triangleq C \otimes I_M. \tag{9.600}$$

*The matrices  $\mathcal{A}$  and  $\mathcal{C}$  have blocks of size  $M \times M$  each. It holds that*

$$\boxed{\|\mathcal{A}^T\|_{b,\infty} = 1, \quad \|\mathcal{C}\|_{b,\infty} = 1}. \tag{9.601}$$

**Proof.** Since  $A^T$  and  $C$  are right stochastic matrices, it holds that  $\|A^T\|_\infty = 1$  and  $\|C\|_\infty = 1$ . The desired result then follows from part (a) of Lemma D.3.  $\square$

The next two results establish useful properties for the block maximum norm of a block diagonal matrix transformed by stochastic matrices; such transformations arise as coefficient matrices that control the evolution of weight error vectors over networks, as in (9.189).

**Lemma D.5 (Block Diagonal Hermitian Matrices).** Consider an  $N \times N$  block diagonal Hermitian matrix  $\mathcal{D} = \text{diag}\{D_1, D_2, \dots, D_N\}$ , where each  $D_k$  is  $M \times M$  Hermitian. It holds that

$$\boxed{\rho(\mathcal{D}) = \max_{1 \leq k \leq N} \rho(D_k) = \|\mathcal{D}\|_{b,\infty}}, \quad (9.602)$$

where  $\rho(\cdot)$  denotes the spectral radius (largest eigenvalue magnitude) of its argument. That is, the spectral radius of  $\mathcal{D}$  agrees with the block maximum norm of  $\mathcal{D}$ , which in turn agrees with the largest spectral radius of its block components.

**Proof.** We already know that the spectral radius of any matrix  $\mathcal{X}$  satisfies  $\rho(\mathcal{X}) \leq \|\mathcal{X}\|$ , for any induced matrix norm [19, 20]. Applying this result to  $\mathcal{D}$  we readily get that  $\rho(\mathcal{D}) \leq \|\mathcal{D}\|_{b,\infty}$ . We now establish the reverse inequality, namely,  $\|\mathcal{D}\|_{b,\infty} \leq \rho(\mathcal{D})$ . Thus, pick an arbitrary  $N \times 1$  block vector  $x$  with entries  $\{x_1, x_2, \dots, x_N\}$ , where each  $x_k$  is  $M \times 1$ . From definition (9.583) we have

$$\begin{aligned} \|\mathcal{D}\|_{b,\infty} &\triangleq \max_{x \neq 0} \frac{\|\mathcal{D}x\|_{b,\infty}}{\|x\|_{b,\infty}} \\ &= \max_{x \neq 0} \left( \frac{1}{\|x\|_{b,\infty}} \cdot \max_{1 \leq k \leq N} \|D_k x_k\| \right) \\ &\leq \max_{x \neq 0} \left( \frac{1}{\|x\|_{b,\infty}} \cdot \max_{1 \leq k \leq N} (\|D_k\| \cdot \|x_k\|) \right) \\ &= \max_{x \neq 0} \max_{1 \leq k \leq N} \left( \|D_k\| \cdot \frac{\|x_k\|}{\|x\|_{b,\infty}} \right) \\ &\leq \max_{1 \leq k \leq N} \|D_k\| \\ &= \max_{1 \leq k \leq N} \rho(D_k), \end{aligned} \quad (9.603)$$

where the notation  $\|D_k\|$  denotes the 2-induced norm of  $D_k$  (i.e., its largest singular value). But since  $D_k$  is assumed to be Hermitian, its 2-induced norm agrees with its spectral radius, which explains the last equality.  $\square$

**Lemma D.6 (Block Diagonal Matrix Transformed by Left Stochastic Matrices).** Consider an  $N \times N$  block diagonal Hermitian matrix  $\mathcal{D} = \text{diag}\{D_1, D_2, \dots, D_N\}$ , where each  $D_k$  is  $M \times M$  Hermitian. Let  $A_1$  and  $A_2$  be  $N \times N$  left stochastic matrices, i.e., their entries are nonnegative and they satisfy  $A_1^T \mathbb{1} = \mathbb{1}$  and  $A_2^T \mathbb{1} = \mathbb{1}$ . Introduce the block matrices

$$\mathcal{A}_1^T = A_1^T \otimes I_M, \quad \mathcal{A}_2^T \triangleq A_2^T \otimes I_M. \quad (9.604)$$

The matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$  have blocks of size  $M \times M$  each. Then it holds that

$$\boxed{\rho(\mathcal{A}_2^T \cdot \mathcal{D} \cdot \mathcal{A}_1^T) \leq \rho(\mathcal{D})}. \quad (9.605)$$

**Proof.** Since the spectral radius of any matrix never exceeds any induced norm of the same matrix, we have that

$$\begin{aligned}
 \rho(\mathcal{A}_2^T \cdot \mathcal{D} \cdot \mathcal{A}_1^T) &\leq \|\mathcal{A}_2^T \cdot \mathcal{D} \cdot \mathcal{A}_1^T\|_{b,\infty} \\
 &\leq \|\mathcal{A}_2^T\|_{b,\infty} \cdot \|\mathcal{D}\|_{b,\infty} \cdot \|\mathcal{A}_1^T\|_{b,\infty} \\
 &\stackrel{(9.601)}{=} \|\mathcal{D}\|_{b,\infty} \\
 &\stackrel{(9.602)}{=} \rho(\mathcal{D}). \tag{9.606}
 \end{aligned}$$

□

In view of the result of Lemma D.5, we also conclude from (9.605) that

$$\boxed{\rho(\mathcal{A}_2^T \cdot \mathcal{D} \cdot \mathcal{A}_1^T) \leq \max_{1 \leq k \leq N} \rho(D_k)}. \tag{9.607}$$

It is worth noting that there are choices for the matrices  $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{D}\}$  that would result in strict inequality in (9.605). Indeed, consider the special case:

$$\mathcal{D} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathcal{A}_1^T = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \end{bmatrix}, \quad \mathcal{A}_2^T = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \end{bmatrix}.$$

This case corresponds to  $N = 2$  and  $M = 1$  (scalar blocks). Then,

$$\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T = \begin{bmatrix} \frac{2}{3} & \frac{2}{3} \\ \frac{2}{3} & 1 \end{bmatrix}$$

and it is easy to verify that

$$\rho(\mathcal{D}) = 2, \quad \rho(\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T) \approx 1.52.$$

The following conclusions follow as corollaries to the statement of Lemma D.6, where by a stable matrix  $X$  we mean one whose eigenvalues lie strictly inside the unit circle.

**Corollary D.1 (Stability Properties).** *Under the same setting of Lemma D.6, the following conclusions hold:*

- a. *The matrix  $\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T$  is stable whenever  $\mathcal{D}$  is stable.*
- b. *The matrix  $\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T$  is stable for all possible choices of left stochastic matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$  if, and only if,  $\mathcal{D}$  is stable.*

**Proof.** Since  $\mathcal{D}$  is block diagonal, part (a) follows immediately from (9.605) by noting that  $\rho(\mathcal{D}) < 1$  whenever  $\mathcal{D}$  is stable. [This statement fixes the argument that appeared in Appendix I of [18] and Lemma 2 of [35]. Since the matrix  $X$  in Appendix I of [18] and the matrix  $\mathcal{M}$  in Lemma 2 of [35] are block diagonal, the  $\|\cdot\|_{b,\infty}$  norm should replace the  $\|\cdot\|_\rho$  norm used there, as in the proof that led to (9.606)]

and as already done in [63].] For part (b), assume first that  $\mathcal{D}$  is stable, then  $\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T$  will also be stable by part (a) for any left-stochastic matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$ . To prove the converse, assume that  $\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T$  is stable for any choice of left stochastic matrices  $\mathcal{A}_1$  and  $\mathcal{A}_2$ . Then,  $\mathcal{A}_2^T \mathcal{D} \mathcal{A}_1^T$  is stable for the particular choice  $\mathcal{A}_1 = I = \mathcal{A}_2$  and it follows that  $\mathcal{D}$  must be stable.  $\square$

## E Comparison with consensus strategies

Consider a connected network consisting of  $N$  nodes. Each node has a state or measurement value  $x_k$ , possibly a vector of size  $M \times 1$ . All nodes in the network are interested in evaluating the average value of their states, which we denote by

$$w^o \triangleq \frac{1}{N} \sum_{k=1}^N x_k. \quad (9.608)$$

A centralized solution to this problem would require each node to transmit its measurement  $x_k$  to a fusion center. The central processor would then compute  $w^o$  using (9.608) and transmit it back to all nodes. This centralized mode of operation suffers from at least two limitations. First, it requires communications and power resources to transmit the data back and forth between the nodes and the central processor; this problem is compounded if the fusion center is stationed at a remote location. Second, the architecture has a critical point of failure represented by the central processor; if it fails, then operations would need to be halted.

### E.1 Consensus recursion

The consensus strategy provides an elegant distributed solution to the same problem, whereby nodes interact locally with their neighbors and are able to converge to  $w^o$  through these interactions. Thus, consider an arbitrary node  $k$  and assign nonnegative weights  $\{a_{\ell k}\}$  to the edges linking  $k$  to its neighbors  $\ell \in \mathcal{N}_k$ . For each node  $k$ , the weights  $\{a_{\ell k}\}$  are assumed to add up to one so that

$$\begin{aligned} & \text{for } k = 1, 2, \dots, N : \\ & a_{\ell k} \geq 0, \quad \sum_{\ell=1}^N a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k. \end{aligned} \quad (9.609)$$

The resulting combination matrix is denoted by  $A$  and its  $k$ th column consists of the entries  $\{a_{\ell k}, \ell = 1, 2, \dots, N\}$ . In view of (9.609), the combination matrix  $A$  is seen to satisfy  $A^T \mathbb{1} = \mathbb{1}$ . That is,  $A$  is left-stochastic. The consensus strategy can be described as follows. Each node  $k$  operates repeatedly on the data from its neighbors and updates its state iteratively according to the rule:

$w_{k,n} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} w_{\ell,n-1}, \quad n > 0,$

(9.610)

where  $w_{\ell,n-1}$  denotes the state of node  $\ell$  at iteration  $n - 1$ , and  $w_{k,n}$  denotes the updated state of node  $k$  after iteration  $n$ . The initial conditions are

$$w_{k,o} = x_k, \quad k = 1, 2, \dots, N. \quad (9.611)$$

If we collect the states of all nodes at iteration  $n$  into a column vector, say,

$$z_n \triangleq \text{col}\{w_{1,n}, w_{2,n}, \dots, w_{N,n}\}. \quad (9.612)$$

Then, the consensus iteration (9.610) can be equivalently rewritten in vector form as follows:

$$\boxed{z_n = \mathcal{A}^T z_{n-1}, \quad n > 0}, \quad (9.613)$$

where

$$\mathcal{A}^T = A^T \otimes I_M. \quad (9.614)$$

The initial condition is

$$z_0 \triangleq \text{col}\{x_1, x_2, \dots, x_N\}. \quad (9.615)$$

## E.2 Error recursion

Note that we can express the average value,  $w^o$ , from (9.608) in the form

$$\boxed{w^o = \frac{1}{N}(\mathbb{1}^T \otimes I_M)z_o}, \quad (9.616)$$

where  $\mathbb{1}$  is the vector of size  $M \times 1$  and whose entries are all equal to one. Let

$$\tilde{w}_{k,n} = w^o - w_{k,n} \quad (9.617)$$

denote the weight error vector for node  $k$  at iteration  $n$ ; it measures how far the iterated state is from the desired average value  $w^o$ . We collect all error vectors across the network into an  $N \times 1$  block column vector whose entries are of size  $M \times 1$  each:

$$\tilde{w}_n \triangleq \begin{bmatrix} \tilde{w}_{1,n} \\ \tilde{w}_{2,n} \\ \vdots \\ \tilde{w}_{N,n} \end{bmatrix}. \quad (9.618)$$

Then,

$$\boxed{\tilde{w}_n = (\mathbb{1} \otimes I_M)w^o - z_n}. \quad (9.619)$$

### E.2.1 Convergence conditions

The following result is a classical result on consensus strategies [43–45]. It provides conditions under which the state of all nodes will converge to the desired average,  $w^o$ , so that  $\tilde{w}_n$  will tend to zero.

**Theorem E.1 (Convergence to Consensus).** *For any initial states  $\{x_k\}$ , the successive iterates  $w_{k,n}$  generated by the consensus iteration (9.610) converge to the network average value  $w^o$  as  $n \rightarrow \infty$  if, and only if, the following three conditions are met:*

$$A^T \mathbb{1} = \mathbb{1}, \quad (9.620)$$

$$A \mathbb{1} = \mathbb{1}, \quad (9.621)$$

$$\rho \left( A^T - \frac{1}{N} \mathbb{1} \mathbb{1}^T \right) < 1. \quad (9.622)$$

That is, the combination matrix  $A$  needs to be doubly stochastic, and the matrix  $A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T$  needs to be stable.

**Proof (Sufficiency).** Assume first that the three conditions stated in the theorem hold. Since  $A$  is doubly stochastic, then so is any power of  $A$ , say,  $A^n$  for any  $n \geq 0$ , so that

$$[A^n]^T \mathbb{1} = \mathbb{1}, \quad A^n \mathbb{1} = \mathbb{1}. \quad (9.623)$$

Using this fact, it is straightforward to verify by induction the validity of the following equality:

$$\left( A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T \right)^n = [A^n]^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T. \quad (9.624)$$

Likewise, using the Kronecker product identities

$$(E + B) \otimes C = (E \otimes C) + (B \otimes C), \quad (9.625)$$

$$(E \otimes B)(C \otimes D) = (EC \otimes BD), \quad (9.626)$$

$$(E \otimes B)^n = E^n \otimes B^n, \quad (9.627)$$

for matrices  $\{E, B, C, D\}$  of compatible dimensions, we observe that

$$\begin{aligned} (\mathcal{A}^n)^T - \frac{1}{N} \cdot (\mathbb{1} \otimes I_M) \cdot (\mathbb{1}^T \otimes I_M) &= \left[ (A^n)^T \otimes I_M \right] - \frac{1}{N} \cdot \left( \mathbb{1}\mathbb{1}^T \otimes I_M \right) \\ &= \left[ (A^n)^T - \frac{1}{N} \cdot \mathbb{1}\mathbb{1}^T \right] \otimes I_M \\ &\stackrel{(9.624)}{=} \left( A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T \right)^n \otimes I_M \\ &= \left[ \left( A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T \right) \otimes I_M \right]^n. \end{aligned} \quad (9.628)$$

Iterating (9.613) we find that

$$z_n = [\mathcal{A}^n]^T z_o \quad (9.629)$$

and, hence, from (9.616) and (9.619),

$$\begin{aligned} \tilde{w}_n &= - \left[ (\mathcal{A}^n)^T - \frac{1}{N} \cdot (\mathbb{1} \otimes I_M) \cdot (\mathbb{1}^T \otimes I_M) \right] \cdot z_o \\ &\stackrel{(9.628)}{=} - \left[ \left( A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T \right)^n \otimes I_M \right] \cdot z_o. \end{aligned} \quad (9.630)$$

Now recall that, for two arbitrary matrices  $C$  and  $D$  of compatible dimensions, the eigenvalues of the Kronecker product  $C \otimes D$  is formed of all product combinations  $\lambda_i(C)\lambda_j(D)$  of the eigenvalues of  $C$  and  $D$  [19]. We conclude from this property, and from the fact that  $A^T - \frac{1}{N}\mathbb{1}\mathbb{1}^T$  is stable, that the coefficient matrix

$$\left( A^T - \frac{1}{N} \cdot \mathbb{1}\mathbb{1}^T \right) \otimes I_M$$

is also stable. Therefore,

$$\tilde{w}_n \rightarrow 0 \text{ as } n \rightarrow \infty. \quad (9.631)$$

(Necessity). In order for  $z_n$  in (9.629) to converge to  $(\mathbb{1} \otimes I_M)w^o$ , for any initial state  $z_o$ , it must hold that

$$\lim_{n \rightarrow \infty} (\mathcal{A}^n)^T \cdot z_o = \frac{1}{N} \cdot (\mathbb{1} \otimes I_M) \cdot (\mathbb{1}^T \otimes I_M) \cdot z_o \quad (9.632)$$

for any  $z_o$ . This implies that we must have

$$\lim_{n \rightarrow \infty} (\mathcal{A}^n)^T = \frac{1}{N} \cdot (\mathbb{1}\mathbb{1}^T \otimes I_M) \quad (9.633)$$

or, equivalently,

$$\lim_{n \rightarrow \infty} (A^n)^T = \frac{1}{N} \mathbb{1}\mathbb{1}^T. \quad (9.634)$$

This in turn implies that we must have

$$\lim_{n \rightarrow \infty} A^T \cdot (A^n)^T = A^T \cdot \frac{1}{N} \mathbb{1}\mathbb{1}^T. \quad (9.635)$$

But since

$$\lim_{n \rightarrow \infty} A^T \cdot (A^n)^T = \lim_{n \rightarrow \infty} (A^{n+1})^T = \lim_{n \rightarrow \infty} (A^n)^T, \quad (9.636)$$

we conclude from (9.634) and (9.635) that it must hold that

$$\frac{1}{N} \mathbb{1}\mathbb{1}^T = \frac{1}{N} A^T \cdot \mathbb{1}\mathbb{1}^T. \quad (9.637)$$

That is,

$$\frac{1}{N} (A^T \mathbb{1} - \mathbb{1}) \cdot \mathbb{1}^T = 0 \quad (9.638)$$

from which we conclude that we must have  $A^T \mathbb{1} = \mathbb{1}$ . Similarly, we can show that  $A\mathbb{1} = \mathbb{1}$  by studying the limit of  $(A^n)^T A^T$ . Therefore,  $A$  must be a doubly stochastic matrix. Now using the fact that  $A$  is doubly stochastic, we know that (9.624) holds. It follows that in order for condition (9.634) to be satisfied, we must have

$$\rho \left( A^T - \frac{1}{N} \mathbb{1}\mathbb{1}^T \right) < 1. \quad (9.639)$$

□

### E.2.2 Rate of convergence

From (9.630) we conclude that the rate of convergence of the error vectors  $\{\tilde{w}_{k,n}\}$  to zero is determined by the spectrum of the matrix

$$A^T - \frac{1}{N} \mathbb{1}\mathbb{1}^T. \quad (9.640)$$

Now since  $A$  is a doubly stochastic matrix, we know that it has an eigenvalue at  $\lambda = 1$ . Let us denote the eigenvalues of  $A$  by  $\lambda_k(A)$  and let us order them in terms of their magnitudes as follows:

$$0 \leq |\lambda_M(A)| \leq \dots \leq |\lambda_3(A)| \leq |\lambda_2(A)| \leq 1, \quad (9.641)$$

where  $\lambda_1(A) = 1$ . Then, the eigenvalues of the coefficient matrix  $(A^T - \frac{1}{N}\mathbf{1}\mathbf{1}^T)$  are equal to

$$\{\lambda_M(A), \dots, \lambda_3(A), \lambda_2(A), 0\}. \quad (9.642)$$

It follows that the magnitude of  $\lambda_2(A)$  becomes the spectral radius of  $A^T - \frac{1}{N}\mathbf{1}\mathbf{1}^T$ . Then condition (9.639) ensures that  $|\lambda_2(A)| < 1$ . We therefore arrive at the following conclusion.

**Corollary E.1 (Rate of Convergence of Consensus).** *Under conditions (9.620)–(9.622), the rate of convergence of the successive iterates  $\{w_{k,n}\}$  towards the network average  $w^o$  in the consensus strategy (9.610) is determined by the second largest eigenvalue magnitude of  $A$ , i.e., by  $|\lambda_2(A)|$  as defined in (9.641).  $\square$*

It is worth noting that doubly stochastic matrices  $A$  that are also *regular* satisfy conditions (9.620)–(9.622). This is because, as we already know from Lemma C.2, the eigenvalues of such matrices satisfy  $|\lambda_m(A)| < 1$ , for  $m = 2, 3, \dots, N$ , so that condition (9.622) is automatically satisfied.

**Corollary E.2 (Convergence for Regular Combination Matrices).** *Any doubly-stochastic and regular matrix  $A$  satisfies the three conditions (9.620) – (9.622) and, therefore, ensures the convergence of the consensus iterates  $\{w_{k,n}\}$  generated by (9.610) towards  $w^o$  as  $n \rightarrow \infty$ .  $\square$*

A regular combination matrix  $A$  would result when the two conditions listed below are satisfied by the graph connecting the nodes over which the consensus iteration is applied.

**Corollary E.3 (Sufficient Condition for Regularity).** *Assume the combination matrix  $A$  is doubly stochastic and that the graph over which the consensus iteration (9.610) is applied satisfies the following two conditions:*

- a. *The graph is connected. This means that there exists a path connecting any two arbitrary nodes in the network. In terms of the Laplacian matrix that is associated with the graph (see Lemma B.1), this means that the second smallest eigenvalue of the Laplacian is nonzero.*
- b.  *$a_{\ell k} = 0$  if, and only if,  $\ell \notin \mathcal{N}_k$ . That is, the combination weights are strictly positive between any two neighbors, including  $a_{kk} > 0$ .*

*Then, the corresponding matrix  $A$  will be regular and, therefore, the consensus iterates  $\{w_{k,n}\}$  generated by (9.610) will converge towards  $w^o$  as  $n \rightarrow \infty$ .*

**Proof.** We first establish that conditions (a) and (b) imply that  $A$  is a regular matrix, namely, that there should exist an integer  $j_o > 0$  such that

$$[A^{j_o}]_{\ell k} > 0 \quad (9.643)$$

for all  $(\ell, k)$ . To begin with, by the rules of matrix multiplication, the  $(\ell, k)$  entry of the  $i$ th power of  $A$  is given by

$$[A^i]_{\ell k} = \sum_{m_1=1}^N \sum_{m_2=1}^N \dots \sum_{m_{i-1}=1}^N a_{\ell m_1} a_{m_1 m_2} \dots a_{m_{i-1} k}. \quad (9.644)$$

The summand in (9.644) is nonzero if, and only if, there is some sequence of indices  $(\ell, m_1, \dots, m_{i-1}, k)$  that forms a path from node  $\ell$  to node  $k$ . Since the network is assumed to be connected, there exists a minimum (and finite) integer value  $i_{\ell k}$  such that a path exists from node  $\ell$  to node  $k$  using  $i_{\ell k}$  edges and that

$$[A^{i_{\ell k}}]_{\ell k} > 0.$$

In addition, by induction, if  $[A^{i_{\ell k}}]_{\ell k} > 0$ , then

$$\begin{aligned}[A^{i_{\ell k}+1}]_{\ell k} &= \sum_{m=1}^N [A^{i_{\ell k}}]_{\ell m} a_{mk} \\ &\geq [A^{i_{\ell k}}]_{\ell k} a_{kk} \\ &> 0.\end{aligned}$$

Let

$$j_o = \max_{1 \leq k, \ell \leq N} \{i_{\ell k}\}.$$

Then, property (9.643) holds for all  $(\ell, k)$ . And we conclude from (9.581) that  $A$  is a regular matrix. It then follows from Corollary E.2 that the consensus iterates  $\{w_{k,n}\}$  converge to the average network value  $w^o$ .  $\square$

### E.2.3 Comparison with diffusion strategies

Observe that in comparison to diffusion strategies, such as the ATC strategy (9.153), the consensus iteration (9.610) employs the same quantities  $w_{k,\cdot}$  on both sides of the iteration. In other words, the consensus construction keeps iterating on the same set of vectors until they converge to the average value  $w^o$ . Moreover, the index  $n$  in the consensus algorithm is an iteration index. In contrast, diffusion strategies employ different quantities on both sides of the combination step in (9.153), namely,  $w_{k,i}$  and  $\{\psi_{\ell,i}\}$ ; the latter variables have been processed through an information exchange step and are updated (or filtered) versions of the  $w_{\ell,i-1}$ . In addition, each step of the diffusion strategy (9.153) can incorporate new data,  $\{d_\ell(i), u_{\ell,i}\}$ , that are collected by the nodes at every time instant. Moreover, the index  $i$  in the diffusion implementation is a time index (and not an iteration index); this is because diffusion strategies are inherently adaptive and perform online learning. Data keeps streaming in and diffusion incorporates the new data into the update equations at every time instant. As a result, diffusion strategies are able to respond to data in an adaptive manner, and they are also able to solve general optimization problems: the vector  $w^o$  in adaptive diffusion iterations is the minimizer of a global cost function (cf. (9.92)), while the vector  $w^o$  in consensus iterations is the average value of the initial states of the nodes (cf. (9.608)).

It turns out that diffusion strategies influence the evolution of the network dynamics in an interesting and advantageous manner in comparison to consensus strategies. We illustrate this point by means of an example. Consider initially the ATC strategy (9.158) without information exchange, whose update equation we repeat below for ease of reference:

$$\psi_{k,i} = w_{k,i-1} + \mu_k u_{k,i}^* [d_k(i) - u_{k,i} w_{k,i-1}], \quad (9.645)$$

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i} \quad (\text{ATC diffusion}). \quad (9.646)$$

These recursions were derived in the body of the chapter as an effective distributed solution for optimizing (9.92) and (9.93). Note that they involve two steps, where the weight estimator  $\mathbf{w}_{k,i-1}$  is first updated to the intermediate estimator  $\boldsymbol{\psi}_{k,i}$ , before the intermediate estimators from across the neighborhood are combined to obtain  $\mathbf{w}_{k,i}$ . Both steps of ATC diffusion (9.645) and (9.646) can be combined into a single update as follows:

$$\boxed{\mathbf{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} [\mathbf{w}_{\ell,i-1} + \mu_{\ell} \mathbf{u}_{\ell,i}^* (\mathbf{d}_{\ell}(i) - \mathbf{u}_{\ell,i} \mathbf{w}_{\ell,i-1})]} \quad (\text{ATC diffusion}). \quad (9.647)$$

Likewise, consider the CTA strategy (9.159) without information exchange, whose update equation we also repeat below:

$$\boldsymbol{\psi}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \mathbf{w}_{\ell,i-1} \quad (\text{CTA diffusion}), \quad (9.648)$$

$$\mathbf{w}_{k,i} = \boldsymbol{\psi}_{k,i-1} + \mu_k \mathbf{u}_{k,i}^* [\mathbf{d}_k(i) - \mathbf{u}_{k,i} \boldsymbol{\psi}_{k,i-1}]. \quad (9.649)$$

Again, the CTA strategy involves two steps: the weight estimators  $\{\mathbf{w}_{\ell,i-1}\}$  from the neighborhood of node  $k$  are first combined to yield the intermediate estimator  $\boldsymbol{\psi}_{k,i-1}$ , which is subsequently updated to  $\mathbf{w}_{k,i}$ . Both steps of CTA diffusion can also be combined into a single update as follows:

$$\boxed{\mathbf{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \mathbf{w}_{\ell,i-1} + \mu_k \mathbf{u}_{k,i}^* \left[ \mathbf{d}_k(i) - \mathbf{u}_{k,i} \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \mathbf{w}_{\ell,i-1} \right]} \quad (\text{CTA diffusion}). \quad (9.650)$$

Now, motivated by the consensus iteration (9.610), and based on a procedure for distributed optimization suggested in [23] (see expression (7.1) in that reference), some works in the literature (e.g., [46, 53, 82–88]) considered distributed strategies that correspond to the following form for the optimization problem under consideration (see, e.g., expression (1.20) in [53] and expression (9) in [87]):

$$\boxed{\mathbf{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \mathbf{w}_{\ell,i-1} + \mu_k \mathbf{u}_{k,i}^* [\mathbf{d}_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i-1}]} \quad (\text{consensus strategy}). \quad (9.651)$$

This strategy can be derived by following the same argument we employed earlier in Sections 3.09.3.2 and 3.09.4 to arrive at the diffusion strategies, namely, we replace  $w^o$  in (9.127) by  $\mathbf{w}_{\ell,i-1}$  and then apply the instantaneous approximations (9.150). Note that the *same* variable  $\mathbf{w}_{k,i}$  appears on both sides of the equality in (9.651). Thus, compared with the ATC diffusion strategy (9.647), the update from  $\mathbf{w}_{k,i-1}$  to  $\mathbf{w}_{k,i}$  in the consensus implementation (9.651) is only influenced by data  $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$  from node  $k$ . In contrast, the ATC diffusion structure (9.645), (9.646) helps incorporate the influence of the data  $\{\mathbf{d}_{\ell}(i), \mathbf{u}_{\ell,i}\}$  from across the neighborhood of node  $k$  into the update of  $\mathbf{w}_{k,i}$ , since these data are reflected in the intermediate estimators  $\{\boldsymbol{\psi}_{\ell,i}\}$ . Likewise, the contrast with the CTA diffusion strategy (9.650) is clear, where the right-most term in (9.650) relies on a combination of all estimators from across the neighborhood of node  $k$ , and not only on  $\mathbf{w}_{k,i-1}$  as in the consensus strategy (9.651). These facts have desirable implications on the evolution of the weight-error vectors across diffusion networks.

Some simple algebra, similar to what we did in Section 3.09.6, will show that the mean of the extended error vector for the consensus strategy (9.651) evolves according to the recursion:

$$\mathbb{E}\tilde{\mathbf{w}}_i = (\mathcal{A}^T - \mathcal{M}\mathcal{R}_u) \cdot \mathbb{E}\tilde{\mathbf{w}}_{i-1}, \quad i \geq 0 \quad (\text{consensus strategy}), \quad (9.652)$$

where  $\mathcal{R}_u$  is the block diagonal covariance matrix defined by (9.184) and  $\tilde{\mathbf{w}}_i$  is the aggregate error vector defined by (9.230). We can compare the above mean error dynamics with the ones that correspond to the ATC and CTA diffusion strategies (9.645), (9.646) and (9.648)–(9.650); their error dynamics follow as special cases from (9.248) by setting  $A_1 = I = C$  and  $A_2 = A$  for ATC and  $A_2 = I = C$  and  $A_1 = A$  for CTA:

$$\mathbb{E}\tilde{\mathbf{w}}_i = \mathcal{A}^T(I_{NM} - \mathcal{M}\mathcal{R}_u) \cdot \mathbb{E}\tilde{\mathbf{w}}_{i-1}, \quad i \geq 0 \quad (\text{ATC diffusion}) \quad (9.653)$$

and

$$\mathbb{E}\tilde{\mathbf{w}}_i = (I_{NM} - \mathcal{M}\mathcal{R}_u)\mathcal{A}^T \cdot \mathbb{E}\tilde{\mathbf{w}}_{i-1}, \quad i \geq 0 \quad (\text{CTA diffusion}). \quad (9.654)$$

We observe that the coefficient matrices that control the evolution of  $\mathbb{E}\tilde{\mathbf{w}}_i$  are different in all three cases. In particular,

$$\text{consensus strategy (9.652) is stable in the mean } \iff \rho(\mathcal{A}^T - \mathcal{M}\mathcal{R}_u) < 1, \quad (9.655)$$

$$\text{ATC diffusion (9.653) is stable in the mean } \iff \rho[\mathcal{A}^T(I_{NM} - \mathcal{M}\mathcal{R}_u)] < 1, \quad (9.656)$$

$$\text{CTA diffusion (9.654) is stable in the mean } \iff \rho[(I_{NM} - \mathcal{M}\mathcal{R}_u)\mathcal{A}^T] < 1. \quad (9.657)$$

It follows that the mean stability of the consensus network is sensitive to the choice of the combination matrix  $A$ . This is not the case for the diffusion strategies. This is because from property (9.605) established in Appendix D, we know that the matrices  $\mathcal{A}^T(I_{NM} - \mathcal{M}\mathcal{R}_u)$  and  $(I_{NM} - \mathcal{M}\mathcal{R}_u)\mathcal{A}^T$  are stable if  $(I_{NM} - \mathcal{M}\mathcal{R}_u)$  is stable. Therefore, we can select the step-sizes to satisfy  $\mu_k < 2/\lambda_{\max}(R_{u,k})$  for the ATC or CTA diffusion strategies and ensure their mean stability regardless of the combination matrix  $A$ . This also means that the diffusion networks will be mean stable whenever the individual nodes are mean stable, regardless of the topology defined by  $A$ . In contrast, for consensus networks, the network can exhibit unstable mean behavior even if all its individual nodes are stable in the mean. For further details and other results on the mean-square performance of diffusion networks in relation to consensus networks, the reader is referred to [89, 90].

## Acknowledgments

The development of the theory and applications of diffusion adaptation over networks has benefited greatly from the insights and contributions of several UCLA Ph.D. students, and several visiting graduate students to the UCLA Adaptive Systems Laboratory (<http://www.ee.ucla.edu/asl>). The assistance and contributions of all students are hereby gratefully acknowledged, including Cassio G. Lopes, Federico S. Cattivelli, Sheng-Yuan Tu, Jianshu Chen, Xiaochuan Zhao, Zaid Towfic, Chung-Kai Yu, Noriyuki Takahashi, Jae-Woo Lee, Alexander Bertrand, and Paolo Di Lorenzo. The author is also particularly thankful to S.-Y. Tu, J. Chen, X. Zhao, Z. Towfic, and C.-K. Yu for their assistance in reviewing an earlier draft of this chapter.

---

## References

- [1] J. Chen, A.H. Sayed, On the limiting behavior of distributed optimization strategies, in: Proc. 50th Annual Allerton Conference on Communication, Control, and Computing, Allerton, IL, October 2012, pp. 1535–1542.
- [2] J. Chen, A.H. Sayed, Diffusion adaptation strategies for distributed optimization and learning over networks, *IEEE Trans. Signal Process.* 60 (8) (2012) 4289–4305.
- [3] J. Chen, A.H. Sayed, Distributed Pareto optimization via diffusion strategies, *IEEE J. Sel. Top. Signal Process.* 7 (2) (2013) 205–220.
- [4] A.H. Sayed, *Fundamentals of Adaptive Filtering*, Wiley, NJ, 2003.
- [5] A.H. Sayed, *Adaptive Filters*, Wiley, NJ, 2008.
- [6] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, NJ, 2002.
- [7] B. Widrow, S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, NJ, 1985.
- [8] S.-Y. Tu, A.H. Sayed, Mobile adaptive networks, *IEEE J. Sel. Top. Signal Process.* 5 (4) (2011) 649–664.
- [9] F. Cattivelli, A.H. Sayed, Modeling bird flight formations using diffusion adaptation, *IEEE Trans. Signal Process.* 59 (5) (2011) 2038–2051.
- [10] J. Li, A.H. Sayed, Modeling bee swarming behavior through diffusion adaptation with asymmetric information sharing, *EURASIP J. Adv. Signal Process.* 2012 (18) 2012, doi:10.1186/1687-6180-2012-18.
- [11] J. Chen, A.H. Sayed, Bio-inspired cooperative optimization with application to bacteria motility, in: Proceedings of the ICASSP, Prague, Czech Republic, May 2011, pp. 5788–5791.
- [12] A.H. Sayed, F.A. Sayed, Diffusion adaptation over networks of particles subject to Brownian fluctuations, in: Proceedings of the Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, November 2011, pp. 685–690.
- [13] J. Mitola, G.Q. Maguire, Cognitive radio: making software radios more personal, *IEEE Personal Commun.* 6 (1999) 13–18.
- [14] S. Haykin, Cognitive radio: brain-empowered wireless communications, *IEEE J. Sel. Areas Commun.* 23 (2) (2005) 201–220.
- [15] Z. Quan, W. Zhang, S.J. Shellhammer, A.H. Sayed, Optimal spectral feature detection for spectrum sensing at very low SNR, *IEEE Trans. Commun.* 59 (1) (2011) 201–212.
- [16] Q. Zou, S. Zheng, A.H. Sayed, Cooperative sensing via sequential detection, *IEEE Trans. Signal Process.* 58 (12) (2010) 6266–6283.
- [17] P. Di Lorenzo, S. Barbarossa, A.H. Sayed, Bio-inspired swarming for dynamic radio access based on diffusion adaptation, in: Proceedings of the EUSIPCO, Barcelona, Spain, August 2011, pp. 402–406.
- [18] F.S. Cattivelli, A.H. Sayed, Diffusion LMS strategies for distributed estimation, *IEEE Trans. Signal Process.* 58 (3) (2010) 1035–1048.
- [19] G.H. Golub, C.F. Van Loan, *Matrix Computations*, third ed., The John Hopkins University Press, Baltimore, 1996.
- [20] R.A. Horn, C.R. Johnson, *Matrix Analysis*, Cambridge University Press, 2003.
- [21] E. Kreyszig, *Introductory Functional Analysis with Applications*, Wiley, NY, 1989.
- [22] B. Poljak, *Introduction to Optimization*, Optimization Software, NY, 1987.
- [23] D.P. Bertsekas, J.N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, first ed., Athena Scientific, Singapore, 1997.
- [24] D.P. Bertsekas, A new class of incremental gradient methods for least squares problems, *SIAM J. Optim.* 7 (4) (1997) 913–926.
- [25] A. Nedic, D.P. Bertsekas, Incremental subgradient methods for nondifferentiable optimization, *SIAM J. Optim.* 12 (1) (2001) 109–138.

- [26] M.G. Rabbat, R.D. Nowak, Quantized incremental algorithms for distributed optimization, *IEEE J. Sel. Areas Commun.* 23 (4) (2005) 798–808.
- [27] C.G. Lopes, A.H. Sayed, Incremental adaptive strategies over distributed networks, *IEEE Trans. Signal Process.* 55 (8) (2007) 4064–4077.
- [28] F.S. Cattivelli, A.H. Sayed, Diffusion LMS algorithms with information exchange, in: Proceedings of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, November 2008, pp. 251–255.
- [29] C.G. Lopes, A.H. Sayed, Distributed processing over adaptive networks, in: Proceedings of the Adaptive Sensor Array Processing Workshop, MIT Lincoln Laboratory, MA, June 2006, pp. 1–5.
- [30] A.H. Sayed, C.G. Lopes, Adaptive processing over distributed networks, *IEICE Trans. Fund. Electron. Commun. Comput. Sci.* E90-A (8) (2007) 1504–1510.
- [31] C.G. Lopes, A.H. Sayed, Diffusion least-mean-squares over adaptive networks, in: Proceedings of the IEEE ICASSP, Honolulu, Hawaii, vol. 3, 2007, pp. 917–920.
- [32] C.G. Lopes, A.H. Sayed, Steady-state performance of adaptive diffusion least-mean squares, in: Proceedings of the IEEE Workshop on Statistical Signal Processing (SSP), Madison, WI, August 2007, pp. 136–140.
- [33] C.G. Lopes, A.H. Sayed, Diffusion least-mean squares over adaptive networks: Formulation and performance analysis, *IEEE Trans. Signal Process.* 56 (7) (2008) 3122–3136.
- [34] A.H. Sayed, F. Cattivelli, Distributed adaptive learning mechanisms, in: S. Haykin, K.J. Ray Liu (Eds.), *Handbook on Array Processing and Sensor Networks*, Wiley, NJ, 2009, pp. 695–722.
- [35] F. Cattivelli, A.H. Sayed, Diffusion strategies for distributed Kalman filtering and smoothing, *IEEE Trans. Autom. Control* 55 (9) (2010) 2069–2084.
- [36] S.S. Ram, A. Nedic, V.V. Veeravalli, Distributed stochastic subgradient projection algorithms for convex optimization, *J. Optim. Theory Appl.* 147 (3) (2010) 516–545.
- [37] P. Bianchi, G. Fort, W. Hachem, J. Jakubowicz, Convergence of a distributed parameter estimator for sensor networks with local averaging of the estimates, in: Proceedings of the IEEE ICASSP, Prague, Czech, May 2011, pp. 3764–3767.
- [38] F.S. Cattivelli, C.G. Lopes, A.H. Sayed, A diffusion RLS scheme for distributed estimation over adaptive networks, in: Proceedings of the IEEE Workshop on Signal Processing Advances Wireless Communications (SPAWC), Helsinki, Finland, June 2007, pp. 1–5.
- [39] F.S. Cattivelli, C.G. Lopes, A.H. Sayed, Diffusion recursive least-squares for distributed estimation over adaptive networks, *IEEE Trans. Signal Process.* 56 (5) (2008) 1865–1877.
- [40] F.S. Cattivelli, C.G. Lopes, A.H. Sayed, Diffusion strategies for distributed Kalman filtering: formulation and performance analysis, in: Proceedings of the IAPR Workshop on Cognitive Information Process. (CIP), Santorini, Greece, June 2008, pp. 36–41.
- [41] F.S. Cattivelli, A.H. Sayed, Diffusion mechanisms for fixed-point distributed Kalman smoothing, in: Proceedings of the EUSIPCO, Lausanne, Switzerland, August 2008, pp. 1–4.
- [42] S.S. Stankovic, M.S. Stankovic, D.S. Stipanovic, Decentralized parameter estimation by consensus based stochastic approximation, *IEEE Trans. Autom. Control* 56 (3) (2011) 531–543.
- [43] M.H. DeGroot, Reaching a consensus, *J. Am. Stat. Assoc.* 69 (345) (1974) 118–121.
- [44] R.L. Berger, A necessary and sufficient condition for reaching a consensus using DeGroot’s method, *J. Am. Stat. Assoc.* 76 (374) (1981) 415–418.
- [45] J. Tsitsiklis, M. Athans, Convergence and asymptotic agreement in distributed decision problems, *IEEE Trans. Autom. Control* 29 (1) (1984) 42–50.
- [46] A. Jadbabaie, J. Lin, A.S. Morse, Coordination of groups of mobile autonomous agents using nearest neighbor rules, *IEEE Trans. Autom. Control* 48 (6) (2003) 988–1001.
- [47] R. Olfati-Saber, R.M. Murray, Consensus problems in networks of agents with switching topology and time-delays, *IEEE Trans. Autom. Control* 49 (2004) 1520–1533.

- [48] R. Olfati-Saber, Distributed Kalman filter with embedded consensus filters, in: Proceedings of the 44th IEEE Conference Decision Control, Sevilla, Spain, December 2005, pp. 8179–8184.
- [49] R. Olfati-Saber, Distributed Kalman filtering for sensor networks, in: Proceedings of the 46th IEEE Conference Decision Control, New Orleans, LA, December 2007, pp. 5492–5498.
- [50] L. Xiao, S. Boyd, Fast linear iterations for distributed averaging, *Syst. Control Lett.* 53 (1) (2004) 65–78.
- [51] L. Xiao, S. Boyd, S. Lall, A scheme for robust distributed sensor fusion based on average consensus, in: Proceedings of the IPSN, Los Angeles, CA, April 2005, pp. 63–70.
- [52] U.A. Khan, J.M.F. Moura, Distributing the Kalman filter for large-scale systems, *IEEE Trans. Signal Process.* 56 (10) (2008) 4919–4935.
- [53] A. Nedic, A. Ozdaglar, Cooperative distributed multi-agent optimization, in: Y. Eldar, D. Palomar (Eds.), *Convex Optimization in Signal Processing and Communications*, Cambridge University Press, 2010, pp. 340–386.
- [54] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [55] T.Y. Al-Naffouri, A.H. Sayed, Transient analysis of data-normalized adaptive filters, *IEEE Trans. Signal Process.* 51 (3) (2003) 639–652.
- [56] V.D. Blondel, J.M. Hendrickx, A. Olshevsky, J.N. Tsitsiklis, Convergence in multiagent coordination, consensus, and flocking, in: Proceedings of the Joint 44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), Seville, Spain, December 2005, pp. 2996–3000.
- [57] D.S. Scherber, H.C. Papadopoulos, Locally constructed algorithms for distributed computations in ad-hoc networks, in: Proceedings of the Information Processing in Sensor Networks (IPSN), Berkeley, CA, April 2004, pp. 11–19.
- [58] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculations by fast computing machines, *J. Chem. Phys.* 21 (6) (1953) 1087–1092.
- [59] W.K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* 57 (1) (1970) 97–109.
- [60] D.M. Cvetković, M. Doob, H. Sachs, *Spectra of Graphs: Theory and Applications*, Wiley, NY, 1998.
- [61] B. Bollobas, *Modern Graph Theory*, Springer, 1998.
- [62] W. Kocay, D.L. Kreher, *Graphs, Algorithms and Optimization*, Chapman & Hall/CRC Press, Boca Raton, 2005.
- [63] N. Takahashi, I. Yamada, A.H. Sayed, Diffusion least-mean-squares with adaptive combiners: formulation and performance analysis, *IEEE Trans. Signal Process.* 9 (2010) 4795–4810.
- [64] S.-Y. Tu, A.H. Sayed, Optimal combination rules for adaptation and learning over networks, in: Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), San Juan, Puerto Rico, December 2011, pp. 317–320.
- [65] X. Zhao, S.-Y. Tu, A.H. Sayed, Diffusion adaptation over networks under imperfect information exchange and non-stationary data, *IEEE Trans. Signal Process.* 60 (7) (2012) 3460–3475.
- [66] R. Abdolee, B. Champagne, Diffusion LMS algorithms for sensor networks over non-ideal inter-sensor wireless channels, in: Proceedings of the IEEE Int. Conference on Distributed Computing in Sensor Systems (DCOSS), Barcelona, Spain, June 2011, pp. 1–6.
- [67] A. Khalili, M.A. Tinati, A. Rastegarnia, J.A. Chambers, Steady state analysis of diffusion LMS adaptive networks with noisy links, *IEEE Trans. Signal Process.* 60 (2) (2012) 974–979.
- [68] S.-Y. Tu, A.H. Sayed, Adaptive networks with noisy links, in: Proceedings of the IEEE Globecom, Houston, TX, December 2011, pp. 1–5.
- [69] X. Zhao, A.H. Sayed, Combination weights for diffusion strategies with imperfect information exchange, in: Proceedings of the IEEE ICC, Ottawa, Canada, June 2012, pp. 1–5.
- [70] J.-W. Lee, S.-E. Kim, W.-J. Song, A.H. Sayed, Spatio-temporal diffusion mechanisms for adaptation over networks, in: Proceedings of the EUSIPCO, Barcelona, Spain, August–September 2011, pp. 1040–1044.

- [71] J.-W. Lee, S.-E. Kim, W.-J. Song, A.H. Sayed, Spatio-temporal diffusion strategies for estimation and detection over networks, *IEEE Trans. Signal Process.* 60 (8) (2012) 4017–4034.
- [72] S. Chouvardas, K. Slavakis, S. Theodoridis, Adaptive robust distributed learning in diffusion sensor networks, *IEEE Trans. Signal Process.* 59 (10) (2011) 4692–4707.
- [73] K. Slavakis, Y. Kopsinis, S. Theodoridis, Adaptive algorithm for sparse system identification using projections onto weighted  $\ell_1$  balls, in: Proceedings of the IEEE ICASSP, Dallas, TX, March 2010, pp. 3742–3745.
- [74] A.H. Sayed, C.G. Lopes, Distributed recursive least-squares strategies over adaptive networks, in: Proceedings of Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, October–November 2006, pp. 233–237.
- [75] L. Xiao, S. Boyd, S. Lall, A space-time diffusion scheme peer-to-peer least-squares-estimation, in: Proceedings of the Information Processing in Sensor Networks (IPSN), Nashville, TN, April 2006, pp. 168–176.
- [76] T. Kailath, A.H. Sayed, B. Hassibi, *Linear Estimation*, Prentice Hall, NJ, 2000.
- [77] F. Cattivelli, A.H. Sayed, Diffusion distributed Kalman filtering with adaptive weights, in: Proceedings of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, November 2009, pp. 908–912.
- [78] A. Nedic, A. Ozdaglar, Distributed subgradient methods for multi-agent optimization, *IEEE Trans. Autom. Control* 54 (1) (2009) 48–61.
- [79] D.P. Bertsekas, J.N. Tsitsiklis, Gradient convergence in gradient methods with errors, *SIAM J. Optim.* 10 (3) (2000) 627–642.
- [80] M. Fiedler, Algebraic connectivity of graphs, *Czech. Math. J.* 23 (1973) 298–305.
- [81] N. Takahashi, I. Yamada, Parallel algorithms for variational inequalities over the cartesian product of the intersections of the fixed point sets of nonexpansive mappings, *J. Approx. Theory* 153 (2) (2008) 139–160.
- [82] S. Barbarossa, G. Scutari, Bio-inspired sensor network design, *IEEE Signal Process. Mag.* 24 (3) (2007) 26–35.
- [83] R. Olfati-Saber, Kalman-consensus filter: optimality, stability, and performance, in: Proceedings of the IEEE CDC, Shanghai, China, 2009, pp. 7036–7042.
- [84] I.D. Schizas, G. Mateos, G.B. Giannakis, Distributed LMS for consensus-based in-network adaptive processing, *IEEE Trans. Signal Process.* 57 (6) (2009) 2365–2382.
- [85] G. Mateos, I.D. Schizas, G.B. Giannakis, Performance analysis of the consensus-based distributed LMS algorithm, *EURASIP J. Adv. Signal Process.* (2009) 1–19.
- [86] S. Kar, J.M.F. Moura, Distributed consensus algorithms in sensor networks: link failures and channel noise, *IEEE Trans. Signal Process.* 57 (1) (2009) 355–369.
- [87] S. Kar, J.M.F. Moura, Convergence rate analysis of distributed gossip (linear parameter) estimation: fundamental limits and tradeoffs, *IEEE J. Sel. Top. Signal Process.* 5 (4) (2011) 674–690.
- [88] A.G. Dimakis, S. Kar, J.M.F. Moura, M.G. Rabbat, A. Scaglione, Gossip algorithms for distributed signal processing, *Proc. IEEE* 98 (11) (2010) 1847–1864.
- [89] S.-Y. Tu, A.H. Sayed, Diffusion networks outperform consensus networks, in: Proceedings of the IEEE Statistical Signal Processing Workshop, Ann Arbor, Michigan, August 2012, pp. 313–316.
- [90] S.-Y. Tu, A.H. Sayed, Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks, *IEEE Trans. Signal Process.* 60 (12) (2012) 6217–6234.

# Array Signal Processing: Overview of the Included Chapters

# 10

Mats Viberg

*Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden*

---

## 3.10.1 Some history

The first Radio Direction Finding system was patented in the early 20th century [1]. The idea was to compare the signal strength from a direction-sensitive antenna when it was “steered” in different directions. The main application of the technology was maritime and aircraft navigation, but being able to localize incoming signal energy also found other civilian and military uses. It was soon realized that the performance could be significantly enhanced by employing an array of spatially separated antennas. The outputs of the antennas were combined to increase the directionality of the antenna system, which also enabled localization of multiple spatially separated signal sources. During the 1930s, the radio localization was further developed by actively transmitting signal energy in narrow lobes [2]. The return signal was monitored, and a high energy indicated that some object was present. The system is known as RAdio Detection And Ranging (RADAR), and the use of coherently combined array antennas can be considered the birth of *Array Signal Processing*. Since the early days, the technology has spread to a variety of sensor types as well as application areas. Electronically steered antenna arrays are used in mobile communication systems, to provide both increased range and directional filtering (sectorization). During the last decade or so, more “intelligent” use of adaptive antenna technology at both the transmit and the receiving ends (MIMO systems) has started to revolutionize wireless communications by offering both vastly improved capacity and interference resilience. Acoustic sensors are used in the air (microphone arrays), underwater (SONAR), as well as in the ground for earthquake monitoring. Furthermore, the mathematical algorithms developed in array signal processing have found use in several other fields, not necessarily involving sensor arrays. All these developments motivate a separate section on Array Signal Processing in the present electronic reference publication.

---

## 3.10.2 Summary of the included chapters

Array Signal Processing is a rather generic area, and it is not clear how to draw the borderlines to neighboring areas such as Statistical Signal Processing, Wireless Communications, Radar, and Acoustic Signal Processing. Some coordination between the corresponding sections has been necessary, and much related work will therefore be found also in other sections. We have tried to make a balanced selection between theory and applications and not focusing on any particular sensor type. Thus, most of the

chapters present generic array signal processing techniques that are broadly applicable. Yet, we are happy to have the two last chapters giving some hints on how these methods need to be modified when faced with various practical aspects as well as specific application demands.

### 3.10.2.1 Introduction to Array Processing

The first chapter by Mats Viberg gives an expanded introduction to the area of Array Signal Processing. The mathematical modeling of coherent array data is given special attention as well as the basic properties of sensor arrays from a spatial filtering point of view. This includes the inherent properties of a given sensor array as well as shaping of the response using beamforming. Several array geometries in one and two dimensions are considered as well as the effect of bandwidth. The chapter then gives a short introduction to Direction-Of-Arrival (DOA) estimation using a passive listening array followed by some examples of non-coherent array signal processing applications.

### 3.10.2.2 Adaptive and Robust Beamforming

The chapter by Sergiy A. Vorobyov further elaborates on the issues of adaptive and robust beamforming. The chapter is dedicated to the memory of Alex B. Gershman, who was a major contributor in these fields as well as several other array signal processing related topics. The received data is assumed to contain a desired signal corrupted by interference and noise, and most designs are based on maximizing the Signal-to-Interference-plus-Noise Ratio (SINR) or some equivalent measure. Time-recursive solutions are introduced as approximations of the exact data-adaptive beamformers. The chapter also presents several formulations to make beamforming less susceptible to small sample support (data-adaptive approaches), to correlated interference as well as to imperfectly known array responses. Classical constrained beamforming techniques are considered as well as more recent approaches based on convex optimization.

### 3.10.2.3 Broadband Beamforming and Optimization

This chapter by Sven E. Nordholm, Hai H. Dam, Chiong C. Lai, and Eric A. Lehmann explores the concept of spatial filtering using beamforming in some detail. A particular concern is applications where there is a demand for broadband operation, such as acoustic microphone and SONAR arrays. The chapter gives an insightful physical background to modeling of broadband data using two different approaches: aperture theory and solution of the wave equation in the frequency domain. For both of these models, various optimization approaches are presented for broadband beamforming, which is actually a multi-channel filter design. Robustness to deviations from the assumed ideal model is also considered. The chapter gives several illustrative design examples.

### 3.10.2.4 DOA Estimation Methods and Algorithms

Originating in passive radio direction finding more than a century ago, Direction-Of-Arrival estimation is now a fairly mature research area, and a vast plethora of algorithms has appeared in the literature. The chapter by Pei-Jung Chung, Mats Viberg, and Jia Yu presents a comprehensive overview of both classical spectral-based methods and parametric approaches. Since the latter often involves solving a computationally expensive optimization problem, the most popular algorithmic solutions are briefly

surveyed. The practical issues of wideband data and number of signals detection are addressed in separate sections. The chapter ends with a list of special topics with references to more detailed treatments.

### 3.10.2.5 Subspace Methods and Exploitation of Special Array Structures

One of the most influential outcomes of Array Signal Processing research is subspace methods. These exploit an inherent low-rank structure of the array covariance matrix in order to enhance the SNR prior to further processing, thus significantly improving the DOA estimation performance. The methods have been extended to several other research areas, most notably subspace-based system identification and blind channel estimation and equalization to name a few. The chapter by Martin Haardt, Marius Pesavento, Florian Roemer, and M. Nabil El Korso gives a comprehensive exposure of subspace methods, with a special emphasis of computationally efficient algorithms that exploit special array geometries. The chapter also introduces the emerging area of tensor-based array processing, which involves modeling and processing higher dimensional data.

### 3.10.2.6 Performance Bounds and Statistical Analysis of DOA Estimation

In most practical estimation problems involving a finite set of noisy data, the statistical properties of a given DOA estimate are of major concern. Indeed, the choice of an estimator typically involves trading performance for computational simplicity. The chapter by Jean Pierre Delmas focuses on fundamental performance bounds for a given data model, and it introduces the main tools for analyzing a given DOA estimation algorithm. The analysis assumes a large enough data set, so that the DOA estimate can be linearly related to a certain finite statistic, whose statistical moments can be easily derived. The analysis is carried out for a few key algorithms, including beamforming, maximum likelihood, and some subspace-based techniques. A key issue in DOA estimation is resolution of closely spaced sources, since most methods perform similarly when the sources are well separated. Thus, the chapter finishes with a special study of DOA estimation performance in the context of two closely spaced sources, and shows how statistical analysis can be useful for predicting the resolution capability of a given algorithm.

### 3.10.2.7 DOA Estimation of Nonstationary Signals

Many applications of practical interest involve nonstationary signals. An illustrative example is the ultrasonic localization system employed by bats, which uses a narrowband signal with time-varying frequency. Similarly, many practical radar systems use linear frequency modulation to provide resolution in both range and speed (Doppler). The chapter by Moeness G. Amin and Yimin D. Zhang focuses on DOA estimation using nonstationary signal models. The main tool is the Spatial Time-Frequency Distribution (STFD), which is a generalization of the classical Cohen class of time-frequency distributions. It is shown how nonstationary space-time filtering using the STFD can enhance the SNR, much in the same way as subspace methods do for stationary data. The chapter shows how “standard” DOA estimation methods can be applied using the STFD framework, resulting in significantly enhanced performance as compared to not taking the nonstationary nature of the data into account. A more recent development of joint DOA and Direction-Of-Departure estimation using Multiple Input Multiple Output (MIMO) radar data is also presented.

### 3.10.2.8 Source Localization and Tracking

The classical DOA estimation problem uses coherent data sampled by an array of sensors. The chapter by Yu-Hen Hu considers the related but different problems of source localization using non-coherent sensor data. Each sensor, which itself could be an array of coherent sensor elements, locally processes the received data, producing measurements of the received signal strength, distance to the signal source, and/or angle of arrival. These local measurements are then used at some fusion center for estimating the  $xy$ -coordinates of the transmitter. The simplest and most well-known technique is triangulation, based on angle-of-arrival measurements. More generally, techniques based on linear or non-linear least-squares and Bayesian estimation are employed. The chapter also introduces the problem of tracking of nonstationary targets.

### 3.10.2.9 Array Processing in the Face of Nonidealities

All high resolution DOA estimation methods are based on a precise mathematical model of the received array data. In practice, such a model is often obtained by measuring the array response to distant sources at known locations, i.e., *array calibration*, and the model is at best a good approximation of the reality. The chapter by Visa Koivunen, Mário Costa, and Mats Viberg considers DOA estimation in the absence of a perfectly known array model. The most common sources of error are first introduced and their detrimental effects are explained. Given array calibration data, two main approaches are then presented. The first assumes a parametric model for the array response, and the unknown parameters are estimated using the calibration data. The second approach is based on array interpolation and wave field modeling. A particularly attractive feature of the latter approach is that an approximate model of the array response is obtained that resembles that of a Uniform Linear Array (ULA). Thus, the various computationally attractive methods for DOA estimation using a ULA (see Section 3.10.2.5) can be applied to arbitrary arrays using only samples of the array response at known directions.

### 3.10.2.10 Applications of Array Signal Processing

The fact that Array Signal Processing has been highly influential in other fields than DOA estimation using passive listening arrays is very well illustrated in the chapter by A. Lee Swindlehurst, Brian D. Jeffs, Gonzalo Seco-Granados, and Jian Li. Though not claiming to be exhaustive, the chapter presents an impressive set of practical engineering applications where array signal processing plays a key role. First, active sensing using radar is considered, and several special issues such as Space-Time Adaptive Processing (STAP) and MIMO radar are explored. Next, applications in radio astronomy are considered, where the objective is to observe signals from distant stars, pulsars, galaxies, and even black holes. The “sensor arrays” may in this case cover a whole continent. The chapter then presents the problem of positioning and navigation using a Global Navigation Satellite System (GNSS) from an array processing perspective. Next, the use of space-time coding technology in MIMO wireless communication for improving the communication performance by diversity and/or multiplexing is explained. Several biomedical applications are then introduced, using ultrasonic data as well as EEG/MEG. The chapter then surveys acoustic applications underwater (SONAR) and in the air (microphone arrays). Finally, a more recent application of array signal processing using chemical sensor arrays is described. The objective is to detect, for example, a chemical spill, and the main difference to “normal” array processing is that the propagation is governed by diffusion rather than the wave equation.

### 3.10.3 Outlook

While array signal processing has been around for several decades, there is still much important ongoing research, see, e.g., [3]. Much of the theoretical work is related to sparse signal modeling and estimation. This can fit into the standard array processing framework by sampling the array response on a dense grid. Given a limited number of point sources, the received data can then be modeled as a linear combination of a relatively small set of basis functions, i.e., a sparse representation. We may also expect to see more applications of compressive sampling technology in array signal processing, just as subspace methods and other array signal processing algorithms are used to recover data in compressed sensing (see, e.g., [4]).

An area which has seen much theoretical advancements in the last few decades is distributed processing and localization (see also Section 3.10.2.8). This has been much driven by military applications, but more recently the potential to use wireless sensor networks in vehicle safety applications has been explored [5]. Besides short-term safety systems, one can also expect this to be useful for large-scale traffic control in the future. Other areas where theoretical work has paved the way for engineering applications are MIMO systems, both for wireless communication and radar. In communication, MIMO is already part of the recent standards, but an area yet to be explored in practice is large-scale MIMO [6]. Similarly, the theoretical advantages of MIMO radar over standard phased array radar [7] are yet to manifest itself in commercial products.

Finally, as illustrated in the chapter by Swindlehurst et al., theory and methods from array signal processing have found their way into a rich plethora of other areas, and we can expect this quest to continue well into the future. An emerging area of great interest is microwave-based imaging, which has been used in biomedical applications such as breast cancer detection [8]. The same technology may prove useful in a variety of monitoring applications in the process industry [9]. This has already resulted in commercial products for the food industry, but we look forward to an interesting development in other branches with similar needs for resolution in time and space and at the same time penetration beneath the surface of the matter under investigation.

---

## References

- [1] J. Dellinger, Principles of Radio Transmission and Reception with Antenna and Coil Aerials, Govt. Print. Off., 1919, No. 330–368 (Online). <<http://books.google.se/books?id=7rrA5ZFH3gUC>>.
- [2] L. Brown, A Radar History of World War II: Technical and Military Perspectives, IOP Publishing Ltd., Bristol, UK, 1999.
- [3] J. Li, B. Sadler, M. Viberg, Sensor array and multichannel signal processing [in the spotlight], *IEEE Signal Process. Mag.* 28 (5) (2011) 157–158.
- [4] K. Gedalyahu, Y. Eldar, Time-delay estimation from low-rate samples: a union of subspaces approach, *IEEE Trans. Signal Process.* 58 (6) (2010) 3017–3031.
- [5] S. Biswas, R. Tatchikou, F. Dion, Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety, *IEEE Commun. Mag.* 44 (1) (2006) 74–82.
- [6] F. Rusek, D. Persson, B. Lau, E. Larsson, T. Marzetta, O. Edfors, F. Tufvesson, Scaling up MIMO: opportunities and challenges with very large arrays, *IEEE Signal Process. Mag.* 30 (1) (2013) 40–60.
- [7] J. Li, P. Stoica, MIMO radar with colocated antennas, *IEEE Signal Process. Mag.* 24 (5) (2007) 106–114.

- [8] E. Fear, S. Hagness, P. Meaney, M. Okoniewski, M. Stuchly, Enhancing breast tumor detection with near-field imaging, *IEEE Microwave Mag.* 3 (1) (2002) 48–56.
- [9] Z. Wu, A.H. Boughriet, H. McCann, L.E. Davis, A.T. Nugroho, Investigation of microwave tomographic imaging techniques for industrial processes, in: *Proceedings of the SPIE 4188, Process Imaging for Automatic Control*, Boston, MA, November 2001, pp. 151–158.

# Introduction to Array Processing\*

# 11

Mats Viberg

*Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden*

## 3.11.1 Introduction

Array processing generally deals with signal processing applications using an array of spatially separated sensors of the same type. These sensors typically sample an incoming wave-field generated by far-field emitters. As such, the area dates back to the early use of radars about a century ago, and in particular to microwave Phased Arrays in the 1950s [1]. Today, the processing of sensor array data is mainly done using Digital Signal Processing (DSP). This has opened up a vast flora of opportunities, and the area has found a rich plethora of engineering applications over the past several decades. Just like radar uses electromagnetic (EM) antennas in the air, sonar arrays are used underwater to localize distant objects [2,3], either by passive listening or by actively transmitting waveforms and analyzing the return echo. Both antenna arrays and sonar arrays are also employed in communication applications, where the purpose is to transmit and receive messages over long distances in the presence of severe interference [4,5]. Other types of sensors include microphone, ultrasound and infrared sensors used in air. An emerging technology over the past decade or so is sensor networks, where sensors are distributed over a wide area. The purpose is generally to monitor the environment and/or to localize a certain signal source [6].

In classical radar and sonar applications, the sensor array is used to focus and steer the energy of a signal in the spatial domain, so that a potential target return is enhanced in favor of surrounding interference, clutter and noise. On transmit, this is done by distributing the signal waveform over the various sensor elements with suitable time delays, which will make the energy from all sensors add coherently in the desired (look) direction, while it is attenuated in other directions. The same technique is used on receive, by first applying the same time delays to the sensor outputs, and then summing the result. The result can be interpreted as a matched filter to a hypothesized signal arriving from the look direction. This “scalar” way of processing the signal is referred to as beam forming, and it is the spatial equivalent of temporal Finite Impulse Response (FIR) filtering. The performance of this approach in terms of resolution and interference suppression capability depends critically on the number of sensor elements and the physical size (aperture) of the array. More elaborate filter functions can be designed to avoid some of the drawbacks of the classical beamformer or matched filter, and this is the subject of Chapters 13 and 20 of this book. Although this is very useful, in particular in applications involving

\*This work was supported in part by the Swedish Foundation for Strategic Research within the Strategic Research Center Charmant.

transmission or reception of scalar signals, more flexibility is offered by treating the sensor elements separately. Sampling the individual sensor outputs allows for applying more sophisticated digital signal processing algorithms that take full advantage of the multidimensional nature of the array outputs. In particular, a physical model of the array output can be exploited to enable, for example, accurate localization of multiple closely-spaced signal sources [7,8]. In particular, the invention of so-called *subspace methods* [9,10] sparked a tremendous interest in Direction-of-Arrival (DOA) estimation, and a plethora of algorithms were developed. Given the importance of this topic in array signal processing, several chapters in the current book are devoted to DOA estimation: Chapter 15 gives more details on subspace methods and in particular how to take advantage of special array structures, Chapter 14 gives a general survey of DOA estimation algorithms, Chapter 17 presents methods designed for non-stationary signals, Chapter 19 treats estimation with real-world arrays and Chapter 16 concerns statistical accuracy aspects.

In applications involving active transmission, the classical way is to transmit a coherent wavefront as alluded to above. However, the sensors can be treated individually also in transmit arrays. Thus, in the so-called MIMO (Multiple Input Multiple Output) technology, the sensors transmit different (often orthogonal) signal waveforms, and the individual sensor outputs are then digitized at a receiving array. This technique has revolutionized wireless communication in terms of offering substantially increased capacity without requiring additional bandwidth or power [11], while the advantages of MIMO radar include enhanced resolution capability and diversity [12,13]. Although array processing methods are designed for a multichannel signal model, they are often useful even in single sensor applications. An important example is time series modeling (or spectral analysis), where an artificial sensor array is created by a tapped-delay line so that the different “array elements” contain the same signal with different time delays. Indeed, this is the case also if physical arrays are used and a plane wave is propagating across the array. Thus, it comes as no surprise that array processing methods are useful also for time series analysis. See, e.g., [14] for spectral analysis and [15,16] for subspace-based system identification methods. Chapter 20 of this book presents more examples where Array Signal Processing technology can be applied other than the archetypical problem of DOA estimation using a passive listening array, and in Chapter 18 the particular area of source localization in sensor networks is surveyed.

The present chapter gives a brief introduction to the general area of array signal processing as well as to the other chapters in this book. It should be stressed that we do not attempt to give a survey of the area—all chapters together may serve that purpose. Rather the ambition is to illustrate the nature of array signal processing methodology, and exemplify by some typical problem formulations and algorithms. Towards this goal, in Section 3.11.2 we go in some detail into the underlying mathematical modeling of sensor array data in the context of narrowband signals from far-field emitters arriving at an array of closely spaced (in relation to the wavelength) sensors. The nature of the problem is illustrated by investigating the properties of this model in the form of spatial filtering functions (beam patterns) for some standard array configurations in Section 3.11.3.1. The connection to temporal Finite Impulse Response (FIR) filtering and Discrete Fourier Transform (DFT) give some additional insight. Next, in Section 3.11.4 a short introduction to beam forming and signal waveform estimation is given, followed in Section 3.11.5 by some examples of Direction-of-Arrival (DOA) estimation methods. The chapter ends in Section 3.11.6 with an outlook towards applications that do not necessarily involve sampling a wave-field coherently using closely spaced sensors. Indeed, the rich activities in array processing has lead to

applications reaching far beyond the generic DOA estimation using EM sensors as is also evident from [Chapter 20](#) of this book.

### 3.11.2 Geometric data model

This section presents the underlying modeling ideas in array processing. The sensor array data are introduced as ideal samples of a transmitted wave in a linear and non-dispersive medium. The influence of imperfect sensors and other sources of perturbation are briefly mentioned, referring to [Chapter 2](#) of this book for more details.

#### 3.11.2.1 Wave propagation

The generic problem in array processing is to estimate the parameters of incoming waves from distant sources. This can, for example, be electromagnetic energy transmitted and captured by antennas or acoustic waves propagated by transducers under water or loudspeakers in the air. In either case, provided the medium of propagation is homogenous and non-dispersive, the “signal”  $E(t, \mathbf{r})$  (e.g., EM field or acoustic pressure) at a time  $t$  and location  $\mathbf{r} = (x, y, z)^T$  is governed by the wave equation

$$\frac{\partial^2 E(t, \mathbf{r})}{\partial x^2} + \frac{\partial^2 E(t, \mathbf{r})}{\partial y^2} + \frac{\partial^2 E(t, \mathbf{r})}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 E(t, \mathbf{r})}{\partial t^2}, \quad (11.1)$$

where  $c$  represents the speed of propagation. In general,  $E(t, \mathbf{r})$  can be vector-valued, for example due to polarization (see, e.g., [\[17\]](#)), but for simplicity we only consider a scalar field here. Some more details about vector fields and vector sensors are provided in [Chapters 14](#) and [16](#) in this book. We are particularly interested in signals transmitted from a point source. It can be shown that the solutions of the wave equation then depend only on the distance to the emitter, and not on the direction. It is well-known that such solutions must satisfy the so-called spherical wave equation:

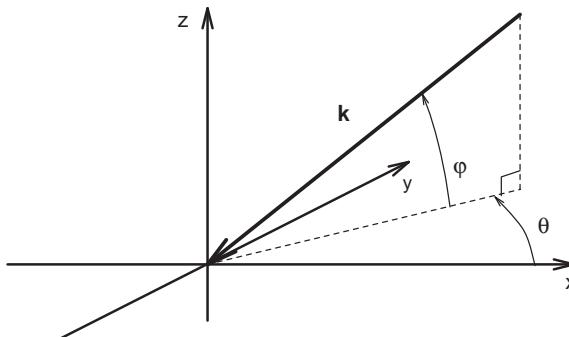
$$\frac{\partial^2 \{r E(t, \mathbf{r})\}}{\partial r^2} = \frac{1}{c^2} \frac{\partial^2 \{r E(t, \mathbf{r})\}}{\partial t^2}, \quad (11.2)$$

where  $r$  is the distance between the observation point  $\mathbf{r}$  and the source. It is easy to see that the following general form satisfies (11.2):

$$E(t, \mathbf{r}) = \frac{1}{r} s_+(t - r/c) + \frac{1}{r} s_-(t + r/c),$$

where  $s_+(t - r/c)$  and  $s_-(t + r/c)$  are two arbitrary functions, whose shape will depend on the initial conditions. Due to the dependence on  $t \pm r/c$ , these functions are interpreted as waves traveling out from and in towards the point source, respectively. In our case, we are mainly concerned with the outgoing wave  $s_+(t - r/c)$ . Assuming a relatively small array of sensors situated in the far-field, the  $1/r$  scaling, being approximately constant, can be absorbed into the function itself, leading to the form

$$E(t, \mathbf{r}) = s(t - r/c).$$

**FIGURE 11.1**

Coordinate system used in the definition of (11.4). Note that azimuth is measured counter-clockwise relative to the positive  $x$ -axis, and elevation is defined relative to the  $xy$ -plane.

With some abuse of notation, let us define  $s(t)$  as the signal at some origin which is near the sensor array and far from the signal source. Then  $r$  above is replaced by the difference in travel distance from the source to the origin and to the observation point respectively. This is given by  $\mathbf{r} \cdot \mathbf{u}_r$ , where  $\mathbf{u}_r$  is a unit vector pointing in the direction of propagation and  $\mathbf{r}$  is the coordinates of the observation point. A case of special interest is a mono-chromatic signal  $s(t) = Ae^{j\omega t}$ , which leads to

$$E(t, \mathbf{r}) = Ae^{j\omega(t - \mathbf{r} \cdot \mathbf{u}_r/c)} = Ae^{j(\omega t - \mathbf{r} \cdot \mathbf{k})}, \quad (11.3)$$

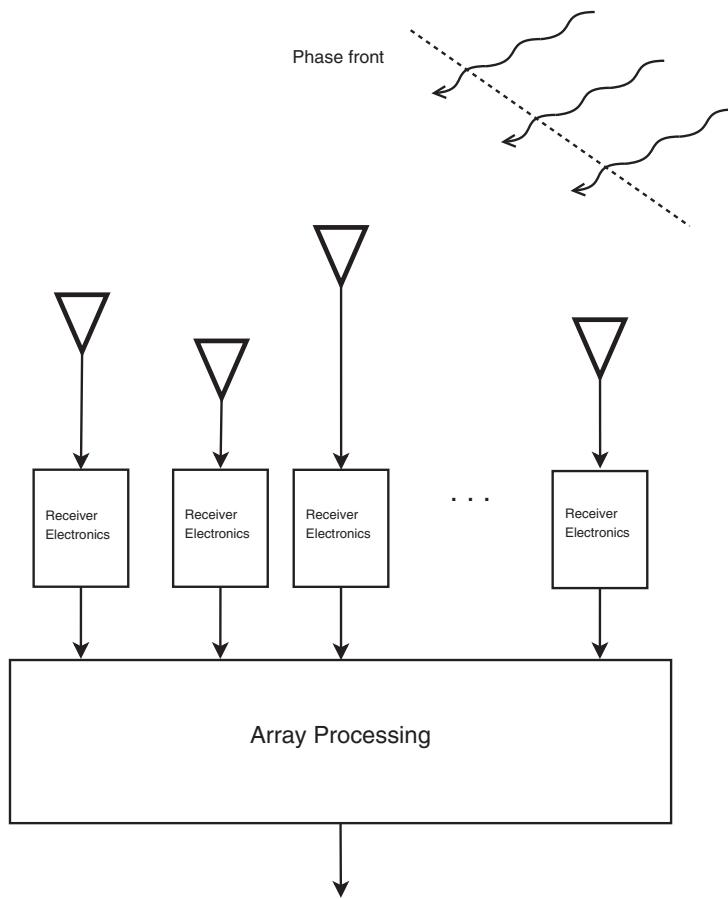
where  $k = \omega/c$  is termed the wave number and  $\mathbf{k} = k\mathbf{u}_r$  is the wave vector. The latter can be expressed in Cartesian coordinates as

$$\mathbf{k} = \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} = -k \begin{bmatrix} \cos \phi \cos \theta \\ \cos \phi \sin \theta \\ \sin \phi \end{bmatrix}, \quad (11.4)$$

where  $\theta$  and  $\phi$  denote, respectively, the azimuth and elevation angles with respect to the coordinate system, see Figure 11.1. Note that these are the same for all sensors due to the far-field assumption. For the same reason, the wave vector is perpendicular to a plane where  $E(t, \mathbf{r})$  depends only on time, called the phase front and defined by  $\mathbf{k} \cdot \mathbf{r} = 0$ . A single-frequency propagating wave in the far-field is therefore termed a plane wave.

### 3.11.2.2 Ideal data model

In many applications of interest, the transmitted signal occupies a very small bandwidth  $B$  as compared to its center frequency  $\omega$ . Using a complex-valued representation, such a signal can be expressed as  $A(t)e^{j\omega t}$ , where the complex amplitude  $A(t)$  varies much slower than  $e^{j\omega t}$ , so that it can be modeled as constant during the propagation of the wave across the array. Suppose a narrowband signal is received by an array of  $M$  sensors at positions  $\mathbf{r}_m$ ,  $m = 1, \dots, M$  relative to the origin, see Figure 11.2.

**FIGURE 11.2**

Plane waves from far-field sources arrive at an array of antennas. The problem of interest here is to estimate the directions to the sources.

Assuming  $A(t - r_m/c) \approx A(t - r/c)$ ,  $\forall m$ , where  $r$  is the distance between the source and the origin, the field at the  $m$ th sensor is expressed using (11.3) as

$$E(t, \mathbf{r}_m) = A(t - r/c)e^{j(\omega t - \mathbf{k} \cdot \mathbf{r}_m)}.$$

If the Radio-Frequency (RF) signal  $E(t, \mathbf{r}_m)$  is captured by an ideal sensor, and the resulting signal is down-converted to baseband,<sup>1</sup> the resulting output of the  $m$ th sensor is given by

$$x_m(t) = e^{-j\mathbf{k} \cdot \mathbf{r}_m} s(t), \quad (11.5)$$

---

<sup>1</sup>In this complex-valued representation, down-conversion is performed by multiplication by  $e^{-j\omega t}$ .

where we have defined the complex envelope signal at the origin as  $s(t) = A(t - r/c)$ . Thus, in the narrowband case, the sensors outputs are all coherent and differ only by a phase shift. In reality, both the sensor and the receiver electronics, see Figure 11.2, may introduce distortions to the ideal model (11.5). Some more details regarding this are given in below, and especially in Chapter 19 of this book. The simple geometrical model (11.5) implies that we can collect the sensor outputs into an  $M$ -vector  $\mathbf{x}(t)$ , termed the *array output*, modeled by

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_M(t) \end{bmatrix} = \mathbf{a}(\theta, \phi)s(t), \quad (11.6)$$

where

$$\mathbf{a}(\theta, \phi) = \begin{bmatrix} e^{-j\mathbf{k}(\theta, \phi) \cdot \mathbf{r}_1} \\ \vdots \\ e^{-j\mathbf{k}(\theta, \phi) \cdot \mathbf{r}_M} \end{bmatrix} \quad (11.7)$$

is termed the *steering vector* (propagation vector or array response vector are other popular names). Note that we have stressed the dependence of the wave vector on the Direction-of-Arrival (DOA), here expressed in a 3D space by the azimuth and elevation angles  $\theta$  and  $\phi$  respectively. The steering vector models the relative phase shifts of a narrowband signal at the various sensor locations due to their different spatial locations. For this reason the models (11.6) and (11.7) are termed the *geometrical data model*.

In many applications, the scenario is two-dimensional, so that the DOA is determined by only one parameter. The by far most studied sensor configuration is that of a Uniform Linear Array (ULA). Using a linear array, it is of course not possible to determine the direction in 3D due to inherent ambiguities. It is then commonly assumed that the sensors and the sources all reside in a plane, for instance the plane  $z = 0$ . Thus, setting the elevation in (11.4) to  $\phi = 0^\circ$  and placing the sensors along the  $y$ -axis, so that  $\mathbf{r}_m = (0, (m-1)\Delta, 0)^T$ , where  $\Delta$  is the inter-element separation, (11.4) and (11.7) yield to  $\mathbf{k} \cdot \mathbf{r}_m = k(m-1)\Delta \sin \theta$ , resulting in

$$\mathbf{a}_{\text{ULA}}(\theta) = \left[ 1, e^{jk\Delta \sin \theta}, \dots, e^{jk(M-1)\Delta \sin \theta} \right]^T. \quad (11.8)$$

If the elevation is unknown,  $\sin \theta$  in (11.8) corresponds to  $\sin \theta \cos \phi$  in (11.4), which can be interpreted as a cone angle with respect to the array axis. A so-called *standard ULA* has  $k\Delta = \pi$ , i.e.,  $\Delta = \pi/k = \lambda/2$ , where  $\lambda = c/f$  is the wavelength. This is the maximum element separation to avoid ambiguities, or *grating lobes*, see Figure 11.6. Ambiguities arise because two or more different DOAs give the same array response, which is normally undesired. In the sequel, we will use the shorter notation  $\mathbf{a}(\theta)$  for the steering vector, keeping in mind that the DOA parameter  $\theta$  is two-dimensional in the 3D case. In general, the steering vector may also be parameterized by other parameters, for example related to the array geometry or to polarization.

In the presence of multiple emitters we can apply the superposition principle for linear sensors, resulting in the ubiquitous data model

$$\mathbf{x}(t) = \sum_{p=1}^P \mathbf{a}(\theta_p) s_p(t) + \mathbf{n}(t), \quad (11.9)$$

where  $\theta_p$  represents the DOA of the  $p$ th signal source,  $s_p(t)$  the corresponding signal waveform, and  $\mathbf{n}(t)$  is a vector-valued additive noise term. The noise represents all un-modeled phenomena in the simple data model (11.6), such as hardware imperfections and various internal and external noise sources. For a well calibrated array,  $\mathbf{n}(t)$  is often assumed to be dominated by thermal noise in the receivers, which can be well-modeled as a stationary white (temporally as well as spatially) and circularly symmetric Gaussian random process. A compact matrix form for (11.9) is obtained by defining the  $M \times P$  steering matrix  $\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_P)]$ , where  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_P]^T$ , and the signal vector  $\mathbf{s}(t) = [s_1(t), \dots, s_P(t)]^T$ . This results in

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t) + \mathbf{n}(t). \quad (11.10)$$

Whether the main objective is to determine the DOAs or to reconstruct one or more of the incoming signals, the estimation is typically based on a finite set of  $N$  samples of  $\mathbf{x}(t)$ , taken at arbitrary time intervals  $t_n$ ,  $n = 1, \dots, N$ . Replacing  $x(t_n)$  by the discrete-time notation  $x(n)$ , we express the available data as

$$\mathbf{x}(n) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n) + \mathbf{n}(n), \quad n = 1, \dots, N. \quad (11.11)$$

Most estimation methods use only second-order properties of the data. The array correlation matrix is defined as  $\mathbf{R}_x = E[\mathbf{x}(n)\mathbf{x}^H(n)]$ , where  $E[\cdot]$  is statistical expectation and  $(\cdot)^H$  denotes complex conjugate and transpose (the Hermitian operator). If the signal and noise vectors are assumed to be independent zero-mean stationary random processes with correlation matrices  $\mathbf{R}_s = E[\mathbf{s}(n)\mathbf{s}^H(n)]$  and  $\mathbf{R}_n = E[\mathbf{n}(n)\mathbf{n}^H(n)]$  respectively, we obtain

$$\mathbf{R}_x = \mathbf{A}(\boldsymbol{\theta})\mathbf{R}_s\mathbf{A}^H(\boldsymbol{\theta}) + \mathbf{R}_n. \quad (11.12)$$

As previously alluded to, the noise is often modeled as spatially white, i.e.,  $\mathbf{R}_n = \sigma^2\mathbf{I}$ , where  $\mathbf{I}$  is the  $M \times M$  identity matrix and  $\sigma^2$  the noise power. If this is not the case, but  $\mathbf{R}_n$  is known, then  $\mathbf{x}(n)$  can be pre-multiplied by an inverse square-root factor  $\mathbf{R}_n^{-1/2}$  of  $\mathbf{R}_n$ , which renders the resulting noise white (and also alters the steering vectors in a predictable way). The array correlation matrix is estimated from the available data by the sample correlation matrix

$$\widehat{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}^H(n). \quad (11.13)$$

Under mild assumptions on the involved random processes (see Chapter 16 in this book for more details regarding large sample analysis), it holds that<sup>2</sup>  $\widehat{\mathbf{R}}_x \rightarrow \mathbf{R}_x$  as  $N \rightarrow \infty$ . Many estimation methods are based on properties of the “true” array correlation matrix  $\mathbf{R}_x$ , but applied to the sample correlation. If the sample size is large enough and the data model is sufficiently good, such an approach can result in highly accurate DOA estimates.

A number of problem formulations can now be stated based on the available data. The most basic problem is that of signal detection, i.e., to determine if there is anything except noise in the data. Formally, this is stated as deciding between the two hypotheses:

$$\begin{aligned} \mathcal{H}_0 : \mathbf{x}(n) &= \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n) + \mathbf{n}(n), \\ \mathcal{H}_1 : \mathbf{x}(n) &= \mathbf{n}(n). \end{aligned} \quad (11.14)$$

<sup>2</sup>Strictly speaking, the convergence is random and needs to be given a more precise statistical meaning. We avoid such technical details in this introductory chapter.

The main difficulty in this formulation is of course that the signal parameters (if there is a signal) are unknown. Closely related to this is the problem to determine the number of signals  $P$ . This is often also referred to as the detection problem, although in some literature the term source enumeration is used. In communication applications, the problem of main interest is to disentangle the signal waveforms  $s(n)$ . One of these may be the signal-of-interest while the others are due to co-channel interference. In this case, there is usually a rich information regarding the signal structure available, such as each  $s_p(n)$  belonging to a finite set of constellation symbols (digital modulation). Finally, the problem that has received the most interest by far is that of estimating the DOA parameters from the given data. The archetypical problem is that of a passive listening array, where (11.11), and (11.12) is the only information available about the model. However, in applications such as Doppler radar, there may be additional signal structure information available, for example that the signal is a complex exponential with an unknown frequency. All these problem formulations call for different treatment, and this has lead to a rich flora of publications over the last several decades. See [7,8,18–20] for some overview treatments. As illustrated in Section 3.11.6 (see also Chapters 18 and 20 of this book), there are several important applications of multi-sensor array processing techniques where the data does not originate from an array of coherent sensors, but where models and ideas from array processing have inspired new theory and methods.

### 3.11.2.3 Non-ideal data models

In any parametric estimation problem, it is of course crucial to have an accurate model of the received data. There are many reasons why a real-world antenna array would deviate from the idealistic sampling of a wave-field as given above. In reality, both the sensor and the receiver electronics, see Figure 11.2, may introduce perturbations. In particular, each sensor has its own characteristics depending on its position in the array, and its presence affects the wave-field. This leads to *mutual coupling* between the sensors, which, if not properly accounted for, can have detrimental effects on the estimation performance. It is common to model the mutual coupling using a so-called coupling matrix  $\mathbf{C}$  (see, e.g., [21]), which modifies the steering vectors from the ideal  $\mathbf{a}(\theta)$  to  $\mathbf{Ca}(\theta)$ . For simple geometries and sensors, such as a ULA of dipoles operating in free space, it is possible to compute  $\mathbf{C}$  theoretically. However, in more practical scenarios one has to resort to numerical computation or experimental characterization. The latter has the additional advantage that any other imperfections not accounted for in the ideal model will also be captured. Some of the more commonly encountered non-idealities include:

- Uncertain element positions or orientations.
- Channel imbalances, leading to gain and phase errors at the different sensors.
- Imbalances between the I and Q channels (i.e., real and imaginary parts of the received data).
- Non-linearities in amplifiers, A/D converters, modulators and other hardware.
- Near-field scattering and other diffuse multipath phenomena.

In addition to this, the color of the background noise may not be perfectly known. In this introductory chapter, we mainly raise the awareness of these practical issues. It is clear that accurate calibration is necessary for a successful implementation of real-world direction finding. More details are given in Chapter 19, whereas Chapter 20 considers beam forming methods that are robust to calibration errors.

### 3.11.3 Spatial filtering and beam patterns

Just as in temporal signal processing, a basic operation in sensor array (i.e., spatial) signal processing is that of linear filtering. The purpose of this section is to give some insight into spatial filtering and into the basic properties of the data model (11.11). We also introduce some common 1D and 2D array structures.

#### 3.11.3.1 Spatial filtering

A linear spatial filter is simply obtained by weighting and summing the sensor outputs, see Figure 11.3. Defining the weights<sup>3</sup> as  $\{w_m^*\}_{m=1}^M$ , the output of the so-called beamformer is given by

$$y(n) = \sum_{m=1}^M w_m^* x_m(n) = \mathbf{w}^H \mathbf{x}(n), \quad (11.15)$$

where  $\mathbf{w} = [w_1, \dots, w_M]^T$  is termed the weight vector (or beamforming vector). In the presence of a single source with DOA parameter(s)  $\theta$ , the array output is given by

$$y(n) = \mathbf{w}^H \mathbf{a}(\theta) s(n). \quad (11.16)$$

Thus, we can think of  $\mathbf{w}^H \mathbf{a}(\theta)$  as the spatial transfer function from  $s(n)$ , at the direction  $\theta$  (and at the frequency  $\omega$ ), to  $y(n)$ . Its magnitude  $G(\theta) = |\mathbf{w}^H \mathbf{a}(\theta)|$  is the gain of the spatial filter towards a signal coming in from the direction  $\theta$ . It is instructive to compare to temporal Finite Impulse Response (FIR) filtering. Thus, if  $s(n)$  is the discrete-time input to an FIR filter with coefficients  $\{h_m\}_{m=0}^{M-1}$ , the output is given by (see, e.g., [22])

$$y(n) = \sum_{m=0}^{M-1} h_m s(n-mT),$$

where  $T$  is the sampling interval. Defining  $\mathbf{h} = [h_0, \dots, h_{M-1}]^T$  and  $\mathbf{s}(n) = [s(n), s(n-T), \dots, s(n-(M-1)T)]^T$ , we can express this in compact form as

$$y(n) = \mathbf{h}^T \mathbf{s}(n). \quad (11.17)$$

Now, the spatial gain function  $G(\theta) = |\mathbf{w}^H \mathbf{a}(\theta)|$  is the response to a point source at the DOA  $\theta$ . This can be interpreted as a spatial line spectrum. The corresponding case in temporal processing is a single frequency input. Putting  $s(n) = e^{j\omega n}$  we have, with some abuse of notation,

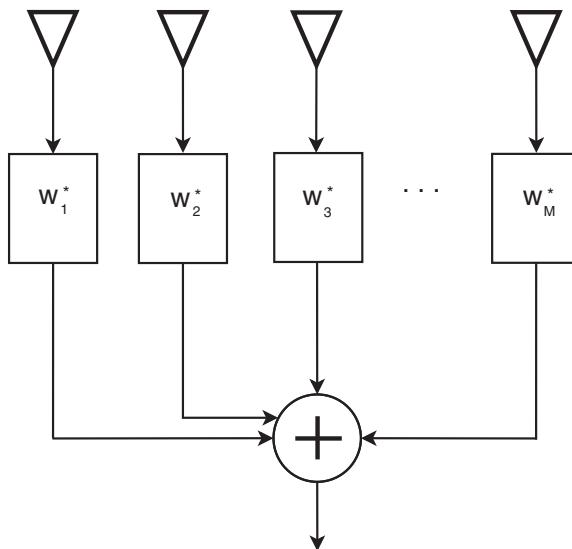
$$\mathbf{s}(n) = [1, e^{-j\omega T}, \dots, e^{-j\omega(M-1)T}]^T e^{j\omega n} = \mathbf{a}(\omega) s(n).$$

Thus, (11.17) yields the FIR-filter response to a pure sinusoid as

$$y(n) = \mathbf{h}^T \mathbf{a}(\omega) s(n) = \sum_{m=0}^{M-1} h_m e^{-j\omega m T} s(n). \quad (11.18)$$

---

<sup>3</sup>The weights are commonly defined with complex conjugate, so as to obtain a proper inner product in (11.15).

**FIGURE 11.3**

Linear spatial filtering through beamforming. The output is the weighted sum of the various antenna signals.

As expected, the filter transfer function  $\sum_{m=0}^{M-1} h_m e^{-j\omega mT}$  is simply given by the Discrete-Time Fourier Transform (DTFT) of the FIR-filter coefficients  $h_m$ . A similar relation is obtained in the case of uniform linear sampling in space. From (11.8), the ideal ULA beamformer output is given by

$$y_{\text{ULA}}(n) = \mathbf{w}^H \mathbf{a}_{\text{ULA}}(\theta) s(n) = \sum_{m=0}^{M-1} w_m^* e^{jm k \Delta \sin \theta} s(n).$$

Defining the *electrical angle* as  $\xi = k \Delta \sin \theta$ , the ULA beamforming gain is expressed as

$$G(\xi) = \left| \sum_{m=0}^{M-1} w_m^* e^{jm \xi} \right| = \left| \sum_{m=0}^{M-1} w_m e^{-jm \xi} \right|.$$

Thus, also in this case the gain is obtained using the DTFT of the filter coefficients. The only difference is that the weights are applied to spatially sampled data at the same time instant, rather than to temporally sampled data at the same spatial location as in the FIR case. In both cases, the choice of weights influence the gain as a function of frequency (spatial or temporal). In particular, a spatial filter can be used to block interference from certain directions in favor of a desired signal. This aspect will be explored in more detail in Section 3.11.4, and in particular in Chapters 13 and 20 of this book.

### 3.11.3.2 One-dimensional arrays

In the previous section, the beamforming gain is defined as  $G(\theta) = |\mathbf{w}^H \mathbf{a}(\theta)|$ , where  $\mathbf{w}$  is the weight vector and  $\mathbf{a}(\theta)$  the array steering vector. We will now introduce the important concept of array beam pattern.

This is simply given by the beamforming gain  $|\mathbf{w}^H \mathbf{a}(\theta)|$  when the beamforming weights are selected as  $\mathbf{w} = \mathbf{a}(\theta_0)$ , where  $\theta_0$  is the *look direction*. Since the weights depend on  $\theta_0$ , we use the notation

$$G(\theta, \theta_0) = |\mathbf{a}^H(\theta_0) \mathbf{a}(\theta)|. \quad (11.19)$$

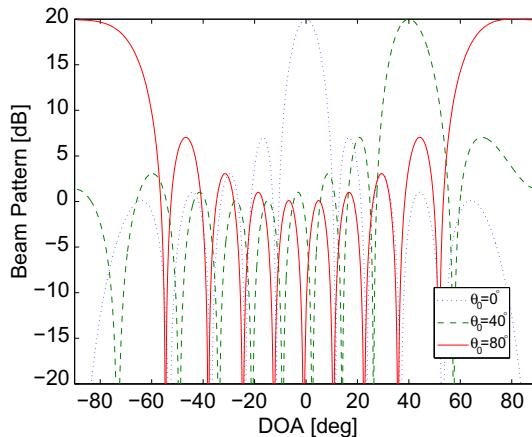
By Cauchy-Schwartz's inequality,  $|\mathbf{a}^H(\theta_0) \mathbf{a}(\theta)| \leq \|\mathbf{a}^H(\theta_0)\| \times \|\mathbf{a}(\theta)\|$ , with equality for  $\theta = \theta_0$ . Thus, when  $\mathbf{w} = \mathbf{a}(\theta_0)$  the gain  $G(\theta, \theta_0)$  will have a maximum in the direction of  $\theta = \theta_0$ . We say that the array is steered to the direction  $\theta_0$ . Conversely, the choice  $\mathbf{w} = \mathbf{a}(\theta_0)$  maximizes the gain  $|\mathbf{w}^H \mathbf{a}(\theta_0)|$  in the direction  $\theta_0$ , subject to the norm constraint  $\|\mathbf{w}\|^2 = \|\mathbf{a}(\theta)\|^2 = M$ . Since it equalizes all phases in  $\mathbf{a}(\theta_0)$  before summing up the gains, we call this a *matched filter*. For a ULA, (11.8) gives the beam pattern as

$$G_{\text{ULA}}(\theta, \theta_0) = \left| \sum_{m=0}^{M-1} e^{jmk\Delta(\sin \theta - \sin \theta_0)} \right| = \frac{|\sin [Mk\Delta(\sin \theta - \sin \theta_0)/2]|}{|\sin [k\Delta(\sin \theta - \sin \theta_0)/2]|}. \quad (11.20)$$

Figure 11.4 shows the beam pattern for a 10-element ULA with different look directions. Due to symmetry, the beam pattern is plotted only for  $-90^\circ \leq \theta < 90^\circ$ . Because of the  $\sin \theta$ -dependence in (11.20), the spatial filter is broadened as  $\theta_0$  approaches  $90^\circ$ , which corresponds to array endfire. This is expected, since the baseline of a linear array vanishes from this angle. In terms of the electrical angle  $\xi = k\Delta \sin \theta$ , the beam pattern is a function only of the difference  $\xi - \xi_0$ ,

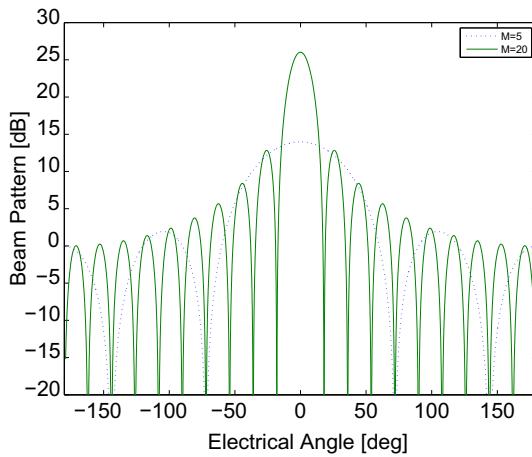
$$G_{\text{ULA}}(\xi - \xi_0) = \left| \sum_{m=0}^{M-1} e^{jm(\xi - \xi_0)} \right| = \frac{|\sin [M(\xi - \xi_0)/2]|}{|\sin [(\xi - \xi_0)/2]|}. \quad (11.21)$$

It is now easy to see that the maximum is given by  $G_{\text{ULA}}(0) = M$  and that  $G_{\text{ULA}}(\xi)$  has nulls when  $\sin(M\xi/2) = 0$ , i.e., for  $\xi = \pm 2\pi l/M$  where  $l$  is an integer. The location of the first null,  $\xi_{\text{BW}} = 2\pi/M$



**FIGURE 11.4**

Beam pattern versus DOA for a 10-element ULA with  $k\Delta = \pi$  for different look directions  $\theta_0$ .

**FIGURE 11.5**

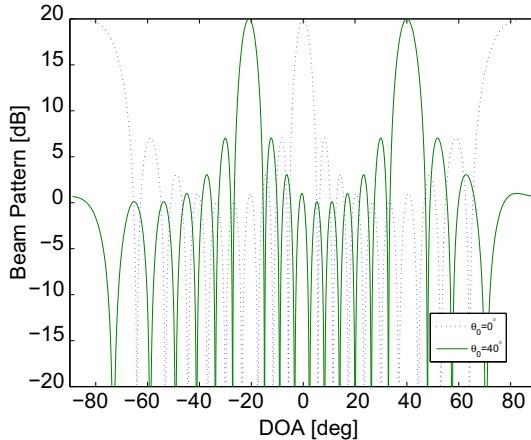
Beam pattern versus electrical angle  $\xi$  for a uniform linear arrays with  $M = 5$  and  $M = 20$  sensors, respectively.

is termed the (Rayleigh) beamwidth of the array. It is clear that the array beam pattern is sharper the larger  $M$  is, as illustrated in Figure 11.5. A narrow main lobe implies a better capability to discriminate between multiple sources at different positions, i.e., *resolution*. Besides the main lobe, the beam pattern also displays sidelobes, the largest being about 13 dB below that of the main lobe peak. The peak sidelobe level is approximately independent of the number of sensors. For large  $M$ , we can approximate the Rayleigh beamwidth in terms of the DOA parameter  $\theta$  as

$$\theta_{\text{BW}} \approx \frac{2\pi/M}{k\Delta \cos \theta_0} = \frac{1}{(\Delta/\lambda)M \cos \theta_0}, \quad (11.22)$$

where in the second equality we used that  $k = 2\pi/\lambda$ , where  $\lambda$  is the wavelength. This demonstrates clearly that the beamwidth is widened as  $\cos \theta_0$  approaches 0, as previously seen in Figure 11.4. Equation (11.22) also shows that the beamwidth is inversely proportional to  $d = \Delta/\lambda$ , which is the element separation normalized to the wavelength. Thus, a larger element separation yields a sharper beam pattern. However, if  $d > 1/2$ , i.e., half wavelength element separation, we may have so-called *grating lobes* in the beam pattern. This means that  $G_{\text{ULA}}(\theta, \theta_0)$  in (11.20) reaches its maximum value of  $M$  at more than one location. This effect is illustrated in Figure 11.6 for different look directions  $\theta_0$ , using the element separation  $d = 1$ . Using the analogy to temporal filtering, introduced e.g., in (11.18), we see that the phenomenon is similar to aliasing in temporal sampling. The element separation  $d = 1/2$ , or  $\Delta = \lambda/2$ , is the spatial analogy of the well-known Nyquist frequency. A ULA with  $d = 1/2$  is hereafter referred to as a *standard ULA*.

It is possible to increase the physical extent of a linear array of  $M$  sensors beyond that of a standard ULA, without causing grating lobes, namely by using a non-uniform element separation. A commonly used geometry is that of a so-called Minimum Redundancy Array (MRA) [23]. The idea of this design

**FIGURE 11.6**

Beam pattern versus DOA for a 10-element ULA with one wavelength element separation using the look directions  $\theta_0 = 0^\circ$  and  $\theta_0 = 40^\circ$  respectively.

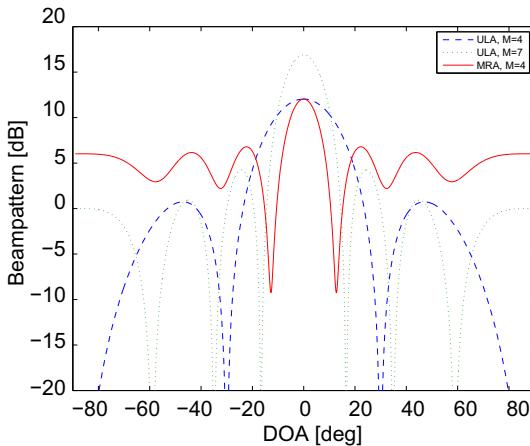
is to choose the pairwise element separations as multiples of  $d = 1/2$  in such a way that all separations  $l/2$ , for  $l = 1, \dots, 2M - 1$  are achieved exactly once by some element pair. For  $M = 4$ , the element separations are  $\{1, 3, 2\}$ , resulting in a total aperture (array length) of  $3\lambda$ , which is twice that of a 4-element standard ULA. It has been shown that no perfect MRA exists for  $M > 4$ , so one has to give up either the requirement to cover all element separations or allow more than one pair for some  $l$ s. In fact, finding good sparse (or “thinned”) array designs has been the subject of a significant research effort over several decades, see, e.g., [1, 24]. The  $M = 4$  MRA beam pattern is compared to that of standard ULAs with  $M = 4$  and  $M = 7$ , the latter having the same aperture as the MRA, in Figure 11.7. It is seen that the 4-element MRA offers significantly increased resolution, in fact similar to that of the 7-element ULA, at the expense of increased sidelobes. The latter can be a drawback in the presence of multiple signal sources and/or interference. In Section 3.11.4 we will study how weight vector design can be used to shape the beam pattern of any array, thus alleviating the problem of high sidelobes to some extent.

### 3.11.3.3 Two-dimensional arrays

To localize a source in 3D, it is necessary to employ a 2D (or even 3D) array. In this case, the steering vector (11.7) is a function of both azimuth and elevation, and consequently so is the beam pattern. Similar to (11.19), we define the 2D beam pattern as

$$G(\theta, \phi) = \left| \mathbf{a}^H(\theta_0, \phi_0) \mathbf{a}(\theta, \phi) \right|, \quad (11.23)$$

where the dependence of  $G(\theta, \phi)$  on the look direction  $(\theta_0, \phi_0)$  has been suppressed for notational convenience. Below, we illustrate the shape of the beam pattern for three popular 2D array structures,

**FIGURE 11.7**

Beam pattern versus DOA for a 4- and 7-element ULA as well as a 4-element Maximum Redundancy Array (MRA).

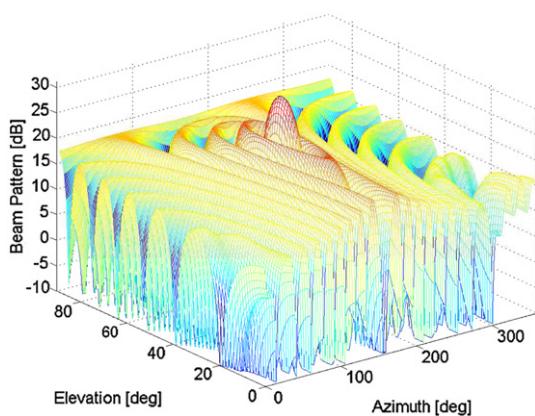
namely the Uniform Circular Array (UCA), Uniform Rectangular Array (URA), and the L-shaped array. For simplicity, we assume that the array elements are placed in the  $xy$ -plane, although the emitters may be in 3D space. The UCA has its elements uniformly placed along a circle of radius  $R$ . Placing the origin of the coordinate system at the center of the array, the coordinates are given by

$$(x_m, y_m, z_m) = (R \cos \eta_m, R \sin \eta_m, 0), \quad m = 1, \dots, M,$$

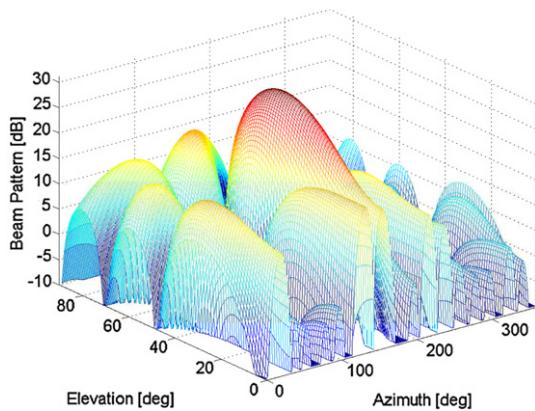
where  $\eta_m = 2\pi(m - 1)/M$ . Since also this array may suffer from grating lobes, it is customary to choose the radius such that the minimum element separation is close to  $\lambda/2$ , as in the case of a ULA. Figure 11.8 shows the beam pattern for a  $M = 36$  element UCA with radius  $R = 3\lambda$ , steered to the look direction  $(\theta_0, \phi_0) = (160^\circ, 45^\circ)$ .

The URA is another natural extension of the ULA to two dimensions. In this case, the element positions are placed in the  $xy$ -plane on a uniform rectangular grid of size  $\sqrt{M} \times \sqrt{M}$  (it is assumed that  $M$  is an even square in this case). Figure 11.9 shows the beam pattern of a URA with  $\lambda/2$  element separation. Again,  $M = 36$  and the array is steered to the “look direction”  $(\theta_0, \phi_0) = (160^\circ, 45^\circ)$ .

The uniform L-shaped array consists of two uniform linear arrays, one along the  $x$  axis and the other along the  $y$  axis. Using the same parameters as for the UCA and URA, the beam pattern for the L-shaped array is illustrated in Figure 11.10. Comparing the three 2D arrays illustrated here, we see that for a given number of sensors, the L-shaped array has the narrowest main beam whereas the URA has the widest. In return, the L-shaped array has a ridge of high sidelobes (peak value approximately  $-6$  dB). One may conclude that the UCA has the most “balanced” beam pattern, offering good resolution and nearly equi-ripple sidelobes, but this must of course be given a more precise meaning depending on the application.

**FIGURE 11.8**

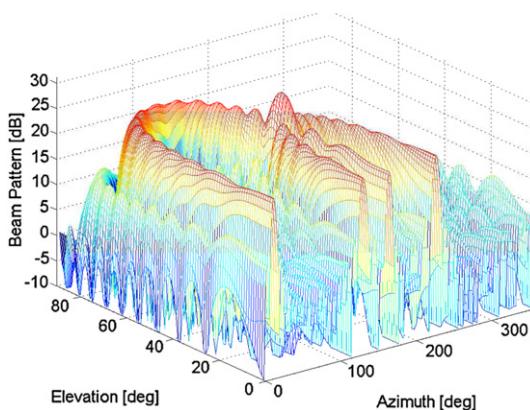
Beam pattern versus azimuth and elevation angle for a  $M = 36$  element uniform circular array steered to  $(\theta_0, \phi_0) = (160^\circ, 45^\circ)$ .

**FIGURE 11.9**

Beam pattern versus azimuth and elevation angle for a  $M = 36$  element standard uniform rectangular array steered to  $(\theta_0, \phi_0) = (160^\circ, 45^\circ)$ .

### 3.11.3.4 Wideband array response

The development this far relies heavily on the assumption of narrowband signals. To arrive at (11.5) it is necessary that the baseband waveform remains approximately constant as the signal traverses the array. The result is that the time-delay at different sensor locations translates to a phase shift due to change in carrier phase. In some applications, in particular those involving acoustic waves, this is a poor approximation due to the inherent wideband nature of the signal. Also in other applications there

**FIGURE 11.10**

Beam pattern versus azimuth and elevation angle for a  $M = 36$  element L-shaped array steered to  $(\theta_0, \phi_0) = (160^\circ, 45^\circ)$ .

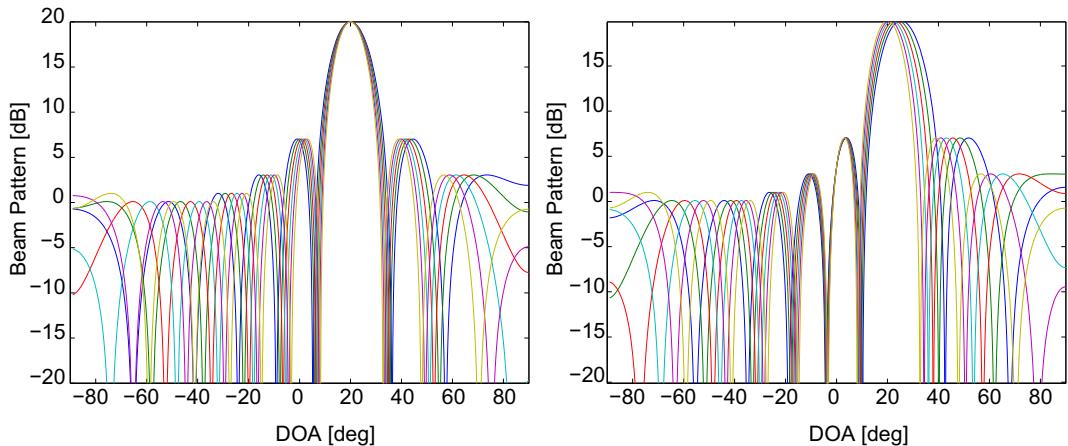
is a constant interest to increase the bandwidth in order to enable more information to be transmitted or to increase the range resolution. In these cases, the wideband characteristics of the signal needs to be accounted for. A natural extension of the narrowband model is to apply the Fourier transform to the signal (after demodulation in case there is a carrier) and express the model in the frequency domain. The counterpart of (11.5) is then

$$X_m(\omega) = e^{-j\mathbf{k}(\omega) \cdot \mathbf{r}_m} S(\omega), \quad (11.24)$$

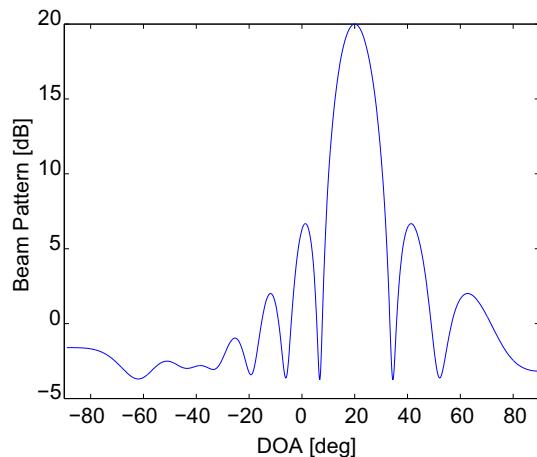
where  $S(\omega)$  denotes the Fourier transform of the transmitted signal,  $X_m(\omega)$  is the received signal in the frequency domain and the dependency of  $\mathbf{k}$  on the frequency has been stressed. Thus, the frequency-domain array output is modeled by

$$\mathbf{X}(\omega) = \mathbf{a}(\theta, \omega) S(\omega). \quad (11.25)$$

It is clear that the only difference to the time-domain model is that the steering vector is now a function of frequency, which needs to be taken into account when designing beamformers and DOA estimation algorithms. This is illustrated in Figure 11.11, which shows the beam patterns for two different beam forming strategies. In the left plot, a so-called *true time-delay* beam former is used, i.e., the weight vector  $\mathbf{w}(\omega) = \mathbf{a}(\theta_0, \omega)$  is a function of frequency. In contrast, in the right plot a fixed weight vector  $\mathbf{w} = \mathbf{a}(\theta_0, \omega_H)$  is used, designed for the highest frequency  $\omega_H$ . The relatively high bandwidth (20% in this case) causes an effect known as *beam squinting* when a frequency-independent beamformer is used, meaning that the peak of the beam pattern appears at the intended look direction only for the design frequency, and is shifted away from  $\theta_0$  at other frequencies. The true time-delay beamformer can be approximately implemented either by pre-processing the digital sensor outputs using FFT and applying frequency-dependent beamformers, or by applying fractional time-delay filters at each digital sensor output to approximate the required time-delay (see, e.g., [25]). While true time-delay beam forming maintains a constant look direction, the beam width and the region outside the main beam still depends

**FIGURE 11.11**

Beam patterns at different frequencies versus azimuth for an  $M = 10$  element wideband ULA steered to  $\theta_0 = 20^\circ$ . The relative bandwidth is 20%. Left: true time-delay beamforming, right: narrowband beamforming.

**FIGURE 11.12**

Wideband beam pattern versus azimuth for an  $M = 10$  element wideband ULA steered to  $\theta_0 = 20^\circ$ . The plot shows the average beam pattern over the 20% relative bandwidth.

on frequency. One can define a *wideband beam pattern*, for example by averaging the power response of the array over frequency, as illustrated in Figure 11.12. The averaging has the effect of a smoothing of the beam pattern outside the main beam so that the wideband beam pattern does not display any sharp nulls as in the narrowband case.

### 3.11.4 Beam forming and signal detection

The previous section introduced spatial filtering as an inherent property of an antenna array. In this section, we consider the design of a spatial filter, or beamformer, to achieve a certain goal. The reader is referred to e.g., [19] for a good overview on the topic, as well as to Chapters 13 and 20 of this book.

#### 3.11.4.1 Beamforming as spatial filter design

A scalar spatial filter is introduced in (11.15) as a weighted linear combination of the sensor outputs

$$y(n) = \mathbf{w}^H \mathbf{x}(n). \quad (11.26)$$

Assuming a single source from the direction  $\theta$ , so that  $\mathbf{x}(n) = \mathbf{a}(\theta)s(n)$ , the gain of the spatial filter is  $G(\theta) = |\mathbf{w}^H \mathbf{a}(\theta)|$ . The spatial filter design problem is then to choose the weight vector  $\mathbf{w}$  to meet some design objectives. We have previously seen that the matched filter  $\mathbf{w} = \mathbf{a}(\theta_0)$  maximizes the gain in the direction  $\theta_0$ . This is usually referred to as conventional, or Bartlett, beamforming. As illustrated in (11.21) and in Figure 11.5, it inherits the properties of the DFT for ULAs, and similarly for other geometries. Among all possible spatial filters it has the narrowest beamwidth, but this comes at the expense of sidelobes ( $-13$  dB in the ULA case). In analogy with Fourier-based spectral analysis, one can trade beamwidth for reduced sidelobes by applying an amplitude window (sometimes referred to as tapering) to the elements of  $\mathbf{a}(\theta_0)$ . Thus, the weighting is chosen as

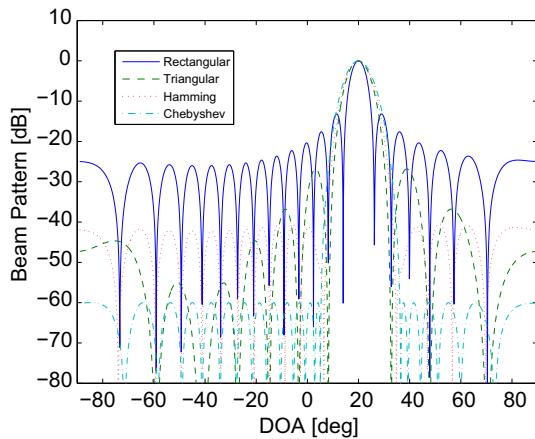
$$\mathbf{w} = \mathbf{w}_{\text{win}} \odot \mathbf{a}(\theta_0), \quad (11.27)$$

where  $\odot$  is the Schur product (elementwise multiplication), and  $\mathbf{w}_{\text{win}}$  is the desired window, see, e.g., [26, 27]. Figure 11.13 illustrates the application of different window functions to an  $M = 20$  element ULA. A 60 dB damping is specified for the Chebyshev window, which then clearly has the lowest peak sidelobe. Although the analogy to spectral analysis is for ULAs, extensions have been proposed for other array geometries, see, e.g., [1, 28] and also [29]. In general, the idea is to transform the given array response to that of a ULA using the concept of phase modes or a similar approach. Another simple generalization of the window-based beamformer design is to enforce a null in the direction of a strong jammer, say  $\theta_j$ , in case the suppression offered by the window is not sufficient. This can easily be achieved by a projection, resulting in the beamformer

$$\mathbf{w} = \boldsymbol{\Pi}_j^\perp (\mathbf{w}_{\text{win}} \odot \mathbf{a}(\theta_0)), \quad (11.28)$$

where  $\boldsymbol{\Pi}_j^\perp = \mathbf{I} - \mathbf{a}(\theta_j)[\mathbf{a}^H(\theta_j)\mathbf{a}(\theta_j)]^{-1}\mathbf{a}^H(\theta_j)$  is a projection matrix onto the orthogonal complement of  $\mathbf{a}(\theta_j)$ .

A more general approach to beamforming or spatial filter design is to formulate the choice of weight vector as a formal optimization problem. Such a design can be very flexible, in that an arbitrary desired beam pattern can be approximated, and the functional form  $\mathbf{a}(\theta)$  of the steering vectors can be arbitrary (it may even be known only in the form of a table lookup). Several formulations have been proposed in the literature, see, e.g., [1], some of them leading to high computational complexity. Fortunately, most design objectives can be formulated as a convex optimization problem [30, 31], implying that an optimal

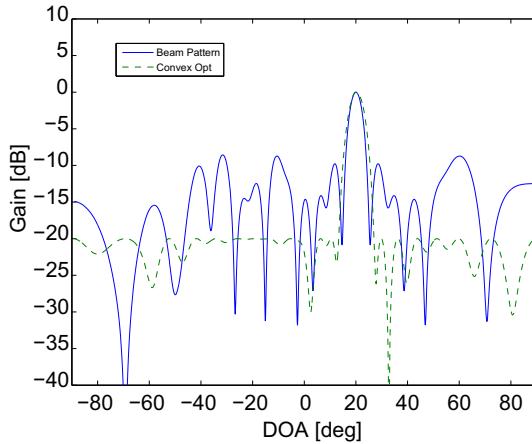
**FIGURE 11.13**

Spatial filter gain versus DOA for different window functions. ULA with look direction  $\theta_0 = 20^\circ$ ,  $M = 20$ . Filter responses are normalized to unit gain in the look direction.

solution can always be found with an acceptable computational effort. As an example, the following Chebyshev-like approach is of practical relevance. Let  $G(\theta) = |\mathbf{w}^H \mathbf{a}(\theta)|$  and introduce the stop-band region  $\theta \in \Theta_s$ . It is assumed that unit gain in the look direction is desired, expressed as  $\mathbf{w}^H \mathbf{a}(\theta_0) = 1$ , and that an upper bound (peak sidelobe)  $\epsilon(\theta)$  on  $G(\theta)$  is specified in the stop-band. Given these constraints, the optimization problem is to minimize the white noise gain, which is given by  $\|\mathbf{w}\|^2$ . To see this, let  $\mathbf{x}(n) = \mathbf{n}(n)$  with  $E[\mathbf{n}(n)\mathbf{n}^H(n)] = \sigma^2 \mathbf{I}$ . The output power is then  $E[|\mathbf{w}^H \mathbf{x}(n)|^2] = \sigma^2 \|\mathbf{w}\|^2$ . The optimization problem is now formulated as:

$$\begin{aligned} & \min_{\mathbf{w}} \|\mathbf{w}\|^2 \\ & \text{s.t. } \mathbf{w}^H \mathbf{a}(\theta_0) = 1 \\ & \quad G(\theta) \leq \epsilon(\theta), \quad \theta \in \Theta_s, \end{aligned} \tag{11.29}$$

where  $\epsilon(\theta) > 0$  is an arbitrary application-specific function (or constant). This problem is easily implemented in publicly available software, such as CVX [32] (in fact, (11.29) is readily available as an application example in the CVX code). The choice of stop-band damping and stop-band region is left to the discretion of the user, and in general it requires some trial and error. Note that the problem (11.29) may lack any feasible solution at all if the specifications are too tight. To illustrate the generality of this approach, Figure 11.14 shows the conventional and the optimized beam pattern for a 20-element random linear array. It is not possible to achieve more than about 20 dB damping for this array, but the response is still considerably improved by the optimization. In closing this section, we point to the possibility to include robustness to certain sources of error into the optimization formulation. This can be, for example, that the position of the desired source (look direction) is not perfectly known, or that the array response is subject to errors. Chapter 20 of this book addresses this issue, and the reader is also referred to [33].

**FIGURE 11.14**

Conventional beam pattern and optimized filter response for a random linear array. Look direction  $\theta_0 = 20^\circ$ ,  $M = 20$ . Filter responses are normalized to unit gain in the look direction.

### 3.11.4.2 Adaptive beamforming

The approach in the previous section is essentially to shape the beamforming response of the beamformer to meet certain requirements. In many situations, the scenario is not precisely known, and it might be impossible to design a filter to work under all circumstances. An attractive alternative is then a beamformer that is adapted to the statistical properties of the desired signal and possible interference and noise sources. Such a filter can also be automatically tuned using the received data (adaptive filtering). A commonly used criterion for an adaptive filter is the Mean-Squared Error (MSE) of the filter output. The latter is given by  $\mathbf{w}^H \mathbf{x}(n)$ , and denoting the desired signal by  $d(n)$ , the objective is to minimize the MSE:

$$\text{MSE} = E[(d(n) - \mathbf{w}^H \mathbf{x}(n))^2]. \quad (11.30)$$

Assuming that  $d(n)$  and  $\mathbf{x}(n)$  are jointly stationary, this is the well-known Wiener Filter (WF), or Linear Minimum Mean-Square Error (LMMSE) problem (see, e.g., [34]). The minimizing weight vector is given by

$$\mathbf{w}_{\text{WF}} = \mathbf{R}_x^{-1} \mathbf{r}_{xd}, \quad (11.31)$$

where  $\mathbf{R}_x = E[\mathbf{x}(n)\mathbf{x}^H(n)]$  is the array correlation matrix and  $\mathbf{r}_{xd} = E[\mathbf{x}(n)d^H(n)]$  is the cross-correlation between the array output and the desired signal. If the array output is modeled by  $\mathbf{x}(n) = \mathbf{a}(\theta_0)s(n) + \mathbf{n}(n)$ , where the desired signal  $d(n) = s(n)$  is independent of the interference-plus-noise term  $\mathbf{n}(n)$ , we get  $\mathbf{r}_{xd} = \mathbf{a}(\theta_0)\sigma_s^2$  so that  $\mathbf{w}_{\text{WF}} = \sigma_s^2 \mathbf{R}_x^{-1} \mathbf{a}(\theta_0)$ . Here,  $\sigma_s^2$  denotes the signal power.

An alternative performance criterion is the Signal-to-Interference-plus-Noise Ratio (SINR). If  $\mathbf{x}(n) = \mathbf{a}(\theta_0)s(n) + \mathbf{z}(n)$ , where  $\mathbf{z}(n)$  represents interference plus noise, we can define the signal part of the beamformer output as  $\mathbf{w}^H \mathbf{a}(\theta_0)s(n)$  and the interference-plus-noise part as  $\mathbf{w}^H \mathbf{z}(n)$ . Thus, the SINR is obtained as

$$\text{SINR} = \frac{\mathbb{E}|\mathbf{w}^H \mathbf{a}(\theta_0) s(n)|^2}{\mathbb{E}|\mathbf{w}^H \mathbf{z}(n)|^2} = \frac{\sigma_s^2 |\mathbf{w}^H \mathbf{a}(\theta_0)|^2}{\mathbf{w}^H \mathbf{R}_z \mathbf{w}}. \quad (11.32)$$

It is easy to show, e.g., using the Cauchy-Schwartz inequality, that the SINR is maximized by the choice

$$\mathbf{w}_{\text{SINR}} = \mu \mathbf{R}_z^{-1} \mathbf{a}(\theta_0), \quad (11.33)$$

where  $\mu$  is an arbitrary scalar (i.e., it does not affect the SINR). As expected,  $\mathbf{w}_{\text{WF}}$  and  $\mathbf{w}_{\text{SINR}}$  are closely related, and if  $\mu = r_s/[1 + r_s \mathbf{a}^H(\theta_0) \mathbf{R}_z^{-1} \mathbf{a}(\theta_0)]$  it holds that  $\mathbf{w}_{\text{WF}} = \mathbf{w}_{\text{SINR}}$ . Thus, the Wiener Filter solution also maximizes the SINR for the case of one signal observed in independent but (possibly) spatially colored noise.

Another possibility worth mentioning is the Minimum Variance Distortionless Response (MVDR) beamformer. In this case, the filter design is again cast as an optimization problem:

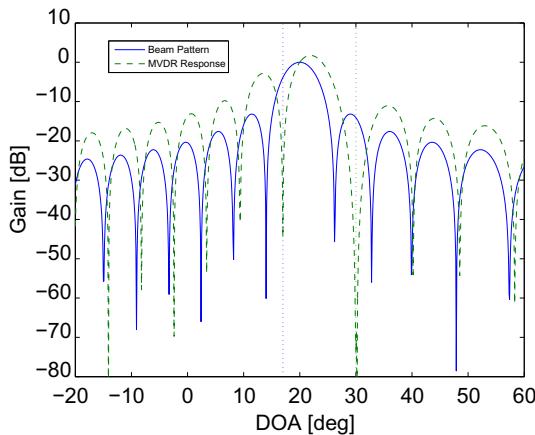
$$\begin{aligned} & \min_{\mathbf{w}} \mathbb{E}|\mathbf{w}^H \mathbf{x}(n)|^2 \\ & \text{s.t. } \mathbf{w}^H \mathbf{a}(\theta_0) = 1. \end{aligned} \quad (11.34)$$

The idea of minimizing the output power is that the beamformer should automatically put nulls in the direction of possible interfering sources, in contrast to e.g., (11.29) where all directions are suppressed. The constraint  $\mathbf{w}^H \mathbf{a}(\theta_0) = 1$  guarantees that the desired signal is preserved despite the power minimization. As is well-known, the solution to this problem is given by

$$\mathbf{w}_{\text{MVDR}} = \frac{1}{\mathbf{a}^H(\theta_0) \mathbf{R}_x^{-1} \mathbf{a}(\theta_0)} \mathbf{R}_x^{-1} \mathbf{a}(\theta_0). \quad (11.35)$$

Clearly, also  $\mathbf{w}_{\text{MVDR}}$  maximizes the SINR, and hence all of the mentioned design principles lead to the same weight vector up to a scaling. Figure 11.15 shows the MVDR response together with the conventional beam pattern for a difficult case with two jammers, of which one (10 dB above the noise floor) is within the main beam of the array and the other (20 dB above the noise) is near the peak sidelobe. As expected the MVDR gain has a deep null at the location of the strongest jammer, and a somewhat lesser null towards the mainbeam jammer. As a result of the latter nulling, the main beam of the MVDR beamformer is somewhat shifted towards the other end. We remark here that the constraint in (11.34) can easily be generalized to an arbitrary linear constraint, say  $\mathbf{C}^H \mathbf{w} = \mathbf{b}$ . The solution (11.35) is then replaced by  $\mathbf{w}_{\text{LCMV}} = \mathbf{R}_x^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}_x^{-1} \mathbf{C})^{-1} \mathbf{b}$ , where LCMV stands for Linearly Constrained Minimum Variance. The extra constraints can be used e.g., to enforce nulls in certain directions and/or to increase the robustness to certain types of errors.

The WF, Max SINR, and MVDR approaches all yield weight vectors of the form  $\mathbf{w} = \alpha \mathbf{R}_x^{-1} \mathbf{a}(\theta_0)$  for some scalar  $\alpha$ , which we take as unity for the sake of brevity. In a practical situation, the array correlation matrix  $\mathbf{R}_x$  may not be known. An obvious approach is then to replace  $\mathbf{R}_x$  by the sample correlation matrix  $\hat{\mathbf{R}}_x$ , defined in (11.13). This is commonly referred to as Sample Matrix Inversion (SMI) [35]. For large enough number of samples  $N$ , the performance of the SMI beamformer is identical to that of the optimal MVDR. A rule of thumb according to [35] is that  $N$  has to be at least twice the number of jammers in order for the expected value of the SINR to be within 3 dB of the optimal one, see also [18]. However, for small number of samples, the resulting filter response may have an unacceptably

**FIGURE 11.15**

Conventional beam pattern and MVDR filter response for a 20-element ULA. A 0 dB desired source is present at the look direction  $\theta_0 = 20^\circ$ , a 10 dB jammer is located in the main beam at  $17^\circ$  and a 20 dB jammer near the peak sidelobe at  $30^\circ$ .

high variance. To alleviate this, it is popular to use regularization (known as diagonal loading in the radar literature). The correlation matrix in (11.35) is then replaced by the regularized version

$$\widehat{\mathbf{R}}_{x,\lambda} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}^H(n) + \lambda \mathbf{I}, \quad (11.36)$$

where  $\lambda$  is the regularization parameter. The addition of the  $\lambda \mathbf{I}$  term tends to stabilize the inverse  $\widehat{\mathbf{R}}_{x,\lambda}^{-1}$ , and the procedure is standard when dealing with ill-conditioned least-squares problems. An additional benefit of the regularization is that it also reduces the sensitivity to various mismatches, such as imprecise knowledge of the look direction and/or array response. Choosing an optimal regularization parameter is not a trivial task, though, and several approaches have been proposed in the literature, see, e.g., [36,37] and Chapter 20 of this book. In radar applications, it is often possible to collect samples of the interference-plus-noise only. Thus, the formulation (11.33) can be used, with  $\mathbf{R}_z$  replaced by the sample covariance matrix taken from the signal-free data. Provided these secondary data are indeed representative of the interference environment in the primary data, the use of  $\widehat{\mathbf{R}}_z^{-1}$  instead of  $\widehat{\mathbf{R}}_x^{-1}$  is known to improve the performance for small samples as well as the robustness to signal-related modeling errors, see, e.g., [38].

In many cases it is desired to update the adaptive beamformer on-line, as new data becomes available. This is similar to other adaptive filtering applications, and a multitude of algorithms are available, see, e.g., [39,40]. Besides simplicity of implementation, such a recursive implementation can cope with a non-stationary signal environment. Tuning an adaptive filter requires some additional information to enable distinguishing signal from noise. If the look direction is known, one can apply a recursive implementation of the weight vector  $\mathbf{w}(n)$  at time  $n$ :

$$\mathbf{w}(n) = \widehat{\mathbf{R}}^{-1}(n)\mathbf{a}(\theta_0). \quad (11.37)$$

This can be achieved by a Recursive Least Squares (RLS) type update of the matrix inverse, see, e.g., [39, 40] for details. Gradient-based solutions are also possible, see, e.g., [41]. In a communication or radar scenario, a so-called reference signal  $d(n)$  is often available, which is identical to or highly correlated with the signal transmitted from the desired look direction (which in this case may be unknown). Given such a reference, one can base the update on the MSE criterion (11.30). The well-known Least Mean Squares (LMS) algorithm (see, e.g., [42]) is based on an instantaneous gradient search of (11.30), with the expectation dropped. This leads to the formula

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \alpha \mathbf{x}(n) \left( d(n) - \mathbf{w}^H(n) \mathbf{x}(n) \right)^*, \quad (11.38)$$

where  $\mathbf{w}(n)$  is the weight vector at time  $n$ , which is updated using the new data  $d(n)$  and  $\mathbf{x}(n)$  to form the next weight vector  $\mathbf{w}(n+1)$ . The steplength parameter  $\alpha$  controls the speed of convergence as well as the stationary variance of the weight vector (for a time-invariant scenario). In general, it should be tuned so that the adaptive algorithm can follow the time variations of the “true” scenario. The LMS algorithm is a stochastic gradient algorithm, since the expectation in (11.30) had to be dropped. This means that the search direction is random, but correct “on average.” Provided the scenario is not too rapidly time-varying, a small steplength should be used to alleviate the random fluctuations. The LMS algorithm is highly popular in practical implementations of adaptive filtering, mainly due to its simplicity and computational efficiency. Other, more complex algorithms such as Recursive Least Squares (RLS) and Kalman Filtering (KF) can provide faster convergence speed at the expense of a higher complexity. There are also even cheaper versions of LMS available, see, e.g., [39, 40]. In concluding this section we remark that given sufficient information, one can estimate all or a subset of the signals that are present in a given scenario at the same time, see, e.g., [43].

### 3.11.4.3 Signal detection

The acronym “radar” stands for radio detection and ranging, and this clarifies that the main purpose of a radar system is to detect the presence of objects by transmitting radio signals and analyze the response. The basic principle is to transmit a signal in a narrow beam, so that the direction to a potential target (if present) can be assumed to be known, and to sample the return signal (after matched filtering) in time, where each sample corresponds to a certain range bin. Several pulses can be used to increase the SNR and/or to determine the speed of a moving target. The response from one pulse at a certain DOA  $\theta_0$  and range bin is modeled by one of the two hypotheses:

$$\begin{aligned} \mathcal{H}_0 : \mathbf{x} &= \mathbf{n}, \\ \mathcal{H}_1 : \mathbf{x} &= \mathbf{a}_0 s + \mathbf{n}, \end{aligned} \quad (11.39)$$

where  $\mathbf{a}_0 = \mathbf{a}(\theta_0)$  and the time index has been dropped for convenience. To decide between the two hypotheses, a statistical model is postulated, and based on this a test statistic is derived. Under  $\mathcal{H}_0$ ,  $\mathbf{x}$  is distributed as  $\mathcal{N}(0, \mathbf{R}_n)$  and under  $\mathcal{H}_1$ , one can use either  $\mathbf{x} \in \mathcal{N}(\mathbf{a}_0 s, \mathbf{R}_n)$  or  $\mathbf{x} \in \mathcal{N}(0, \mathbf{R}_n + \sigma_s^2 \mathbf{a}_0 \mathbf{a}_0^H)$ . We consider only the former model here, which means that the signal amplitude  $s$  is regarded as a deterministic constant. The most powerful test in terms of maximizing the Probability of Detection

(PD) for a given False Alarm (FA) rate is given by the Likelihood Ratio Test (LRT). The logarithm of the ratio of the likelihood functions under  $\mathcal{H}_1$  and  $\mathcal{H}_0$  respectively, is easily found as

$$lr = -(\mathbf{x} - \mathbf{a}_0 s)^H \mathbf{R}_n^{-1} (\mathbf{x} - \mathbf{a}_0 s) + \mathbf{x}^H \mathbf{R}_n^{-1} \mathbf{x}. \quad (11.40)$$

The optimal test is now to compare  $lr$  to a threshold, and decide in favor of  $\mathcal{H}_1$  when  $lr$  exceeds the threshold. The latter is usually selected to achieve a pre-specified FA rate, which is the probability that  $lr$  exceeds the threshold when  $\mathcal{H}_0$  is true. In practice, the signal amplitude  $s$  as well as the noise correlation matrix  $\mathbf{R}_n$  are both unknown. A useful procedure is to replace  $s$  by its Maximum Likelihood (ML) estimate, which leads to the Generalized LRT (GLRT). Maximizing (11.40) w.r.t.  $s$  leads to

$$\hat{s} = \frac{\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{x}}{\mathbf{a}_0^H \mathbf{R}_n \mathbf{a}_0}. \quad (11.41)$$

Substituting (11.41) back into (11.40) and canceling the common term leads to

$$glr = \frac{|\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{x}|^2}{\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}. \quad (11.42)$$

It is clear that  $\mathbf{R}_n$  cannot be estimated from the single data  $\mathbf{x}$  alone. Instead, it is assumed that a secondary data set is available with only interference-plus-noise. This can be taken from adjacent range bins or data at other frequencies for example. Thus, the sample correlation  $\widehat{\mathbf{R}}_n$  is computed from the secondary data and used in (11.42), which gives the so-called Adaptive Matched Filter (AMF) [44]. In view of the fact that  $\mathbf{R}_x^{-1} \mathbf{a}_0 \propto \mathbf{R}_n^{-1} \mathbf{a}_0$ , one can also use data that (possibly) contains the signal, although this may require more data for the same performance. The AMF test is now to decide in favor of  $\mathcal{H}_1$  if  $\widehat{glr} > \gamma$ , and for  $\mathcal{H}_0$  otherwise, where  $\gamma$  is the threshold and  $\widehat{glr}$  is  $glr$  with  $\mathbf{R}_n$  replaced by  $\widehat{\mathbf{R}}_n$ . The AMF enjoys the Constant False Alarm Ratio (CFAR) property, i.e., its FA ratio is independent of the true noise covariance matrix. For a large sample size,  $K$ , of the secondary data set, the FA rate can be approximated by [44,45]

$$\text{Prob}(glr > \gamma | \mathcal{H}_0) = (1 - \gamma)^{(K+1-M)}. \quad (11.43)$$

The power of the test, which is  $\text{Prob}(glr > \gamma | \mathcal{H}_1)$  is not available in closed form, but can easily be numerically evaluated.

### 3.11.5 Direction-of-arrival estimation

In this section we give a brief introduction to the vast area of Direction-of-Arrival (DOA) estimation. For more details, refer to Chapter 14 of this book, as well as, e.g., [7,8]. From Section 3.11.2, the general problem is formulated as follows: Given data  $\mathbf{x}(n)$ ,  $n = 1, \dots, N$ , modeled by the relation

$$\mathbf{x}(n) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n) + \mathbf{n}(n), \quad (11.44)$$

we wish to estimate the parameter vector  $\boldsymbol{\theta}$ , which contains the  $P$  DOAs of interest. The noise is usually assumed to be spatially and temporally white, so that  $E[\mathbf{n}(n)\mathbf{n}^H(m)] = \delta_{n,m}\sigma^2 \mathbf{I}$ , where  $\delta_{n,m}$

is the Kronecker delta. The spatial whiteness can be relaxed, provided the noise correlation matrix  $\mathbf{R}_n$  is known. Temporal correlation is generally not a problem, but might imply that a larger number of samples  $N$  is needed for the same performance. Note that  $\theta$  can contain more than  $P$  parameters, for example azimuth and elevation as well as polarization parameters. However, we will here assume that there is only one parameter per source, referred to as the DOA  $\theta$ . Inherent in the DOA estimation problem is to determine the number of signals  $P$ . The signal waveforms  $\mathbf{s}(n)$  can be regarded as nuisance parameters when estimating the DOAs, although some techniques exploit certain signal properties that might be known, for example constant modulus or cyclo-stationarity [46, 47].

### 3.11.5.1 Beamforming methods

In view of the connection to temporal filtering and DFT (see Section 3.11.3.1), a natural approach to DOA estimation is to apply Fourier-based techniques for spectrum estimation. Thus, a beamforming vector  $\mathbf{w}(\theta)$  is selected, with the objective to steer the array in the direction  $\theta$ . In the classical (Bartlett) beamforming case,  $\mathbf{w}(\theta)$  is taken as the steering vector  $\mathbf{a}(\theta)$ , which means that  $\mathbf{w}(\theta)$  is a matched filter in the direction  $\theta$ . Given the beamforming vector, the output energy at time instant  $n$  is given by  $|\mathbf{w}^H(\theta)\mathbf{x}(n)|^2$ , and the total output power is computed as

$$P(\theta) = \frac{1}{N} \sum_{n=1}^N |\mathbf{w}^H(\theta)\mathbf{x}(n)|^2 = \mathbf{w}^H(\theta)\widehat{\mathbf{R}}\mathbf{w}(\theta), \quad (11.45)$$

where the second equality follows by writing  $|\mathbf{w}^H(\theta)\mathbf{x}(n)|^2 = \mathbf{w}^H(\theta)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}(\theta)$ . Now,  $P(\theta)$  is regarded as a *spatial spectrum*, and it is expected to exhibit peaks near the locations of the true DOAs  $\theta_1, \dots, \theta_P$ . Thus, the DOA estimates are taken as the parameter values  $\hat{\theta}_1, \dots, \hat{\theta}_P$  of the  $P$  highest peaks (isolated) of  $P(\theta)$ . Indeed, in the presence of a single source in temporally and spatially white noise, the peak location of the Conventional Beam Former (CBF) with  $\mathbf{w}(\theta) = \mathbf{a}(\theta)$  coincides with the Maximum Likelihood (ML) estimator, and thus shares its optimality properties. However, the presence of multiple signal sources tends to move the peaks due to signal leakage, and if two sources are too close they will only give rise to a single peak. This limited resolution capability is similar to Fourier-based spectral analysis. According to (11.22), the resolution is given by  $\theta_{\text{BW}} \approx \frac{1}{Md \cos \theta_0}$  for a ULA, where  $\theta_0$  is the true DOA and  $d = \Delta/\lambda$  is the element separation measured in wavelengths. It may also happen that a strong source masks a nearby weaker one through the sidelobes. This frequency masking can be alleviated by the use of windowing, see Section 3.11.4.1, although at the expense of a reduced resolution.

Conventional beamforming-based DOA estimation does not take full advantage of the available data. Indeed, its performance is essentially independent of the Signal-to-Noise Ratio (SNR). The use of adaptive beamforming (11.34), with  $\mathbf{R}_x$  replaced by the sample correlation  $\widehat{\mathbf{R}}_x$  offers an improved resolution at high SNR. Inserting (11.35) into (11.45) leads to the MVDR, or Capon [48] spectrum

$$P_{\text{MVDR}}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\widehat{\mathbf{R}}_x^{-1}\mathbf{a}(\theta)}. \quad (11.46)$$

As before, the DOA estimates are taken as the locations of the  $P$  largest peaks of (11.46). As shown, e.g., in [49], the resolution of the MVDR spectrum improves with increasing SNR. Still, the method is unable not take full advantage of the data. For example, neither the resolution nor the bias is essentially improved even if an infinite amount of data is available, i.e., if the true  $\mathbf{R}_x$  is known.

### 3.11.5.2 Subspace methods

A new class of methods was introduced in the late 1970s to further improve the resolution of spectral-based DOA estimation. The idea is to exploit the geometrical properties of the data model in a more explicit way than, e.g., the adaptive beamforming techniques. From (11.12), the array correlation matrix is given by

$$\mathbf{R}_x = \mathbf{A}(\theta_0)\mathbf{R}_s\mathbf{A}^H(\theta_0) + \sigma^2\mathbf{I}, \quad (11.47)$$

where it is assumed that  $\mathbf{R}_n = \sigma^2\mathbf{I}$  and we have used  $\theta_0$  to stress that  $\mathbf{R}_x$  depends on the true DOA parameter vector. Clearly, if some vector  $\mathbf{e}$  is orthogonal to  $\mathbf{A}(\theta_0)$ , (11.47) shows that  $\mathbf{R}_x\mathbf{e} = \sigma^2\mathbf{e}$ , implying that  $\mathbf{e}$  is an eigenvector of  $\mathbf{R}_x$  with the eigenvalue  $\sigma^2$ . Provided  $P < M$  and  $\mathbf{R}_s$  has full rank, the matrix  $\mathbf{A}(\theta_0)\mathbf{R}_s\mathbf{A}^H(\theta_0)$  is positive semi-definite and has rank  $P$ . Thus, there exists a basis of  $M - P$  vectors  $\{\mathbf{e}_{P+1}, \dots, \mathbf{e}_M\}$  that are orthogonal to  $\mathbf{A}(\theta_0)$ , and all of them are eigenvectors of  $\mathbf{R}_x$  with the same eigenvalue  $\sigma^2$ . We call these the *noise eigenvectors*. Recall that eigenvectors of Hermitean matrices that belong to different eigenvalues are orthogonal. Therefore, the remaining  $P$  eigenvectors  $\{\mathbf{e}_1, \dots, \mathbf{e}_P\}$  that correspond to eigenvalues  $\lambda_1, \dots, \lambda_P$  that are all greater than  $\sigma^2$ , constitute an orthogonal basis for the span of the matrix  $\mathbf{A}(\theta_0)$ , which we call the *Signal Subspace*. We can thus express the eigendecomposition of the array correlation matrix, with eigenvalues in non-increasing order, as

$$\mathbf{R}_x = \sum_{m=1}^M \lambda_m \mathbf{e}_m \mathbf{e}_m^H = \mathbf{E}_s \Lambda_s \mathbf{E}_s^H + \mathbf{E}_n \Lambda_n \mathbf{E}_n^H, \quad (11.48)$$

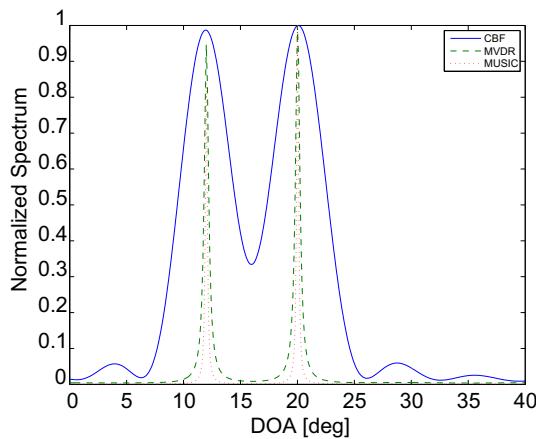
where  $\mathbf{E}_s = [\mathbf{e}_1, \dots, \mathbf{e}_P]$  contains the signal eigenvectors, and the noise eigenvector matrix  $\mathbf{E}_n = [\mathbf{e}_{P+1}, \dots, \mathbf{e}_M]$  obeys  $\mathbf{A}^H(\theta_0)\mathbf{E}_n = 0$ , where 0 is a matrix of suitable dimension. Clearly, the relation  $\mathbf{a}^H(\theta)\mathbf{E}_n = 0$  for  $\theta \in \{\theta_1, \dots, \theta_P\}$  is useful to determine the DOA parameters  $\theta_p$ . Provided the array is *unambiguous*, meaning that any  $M$  steering vectors that correspond to distinct DOA parameters are linearly independent, the relation  $\mathbf{a}^H(\theta)\mathbf{E}_n = 0$  does not have any false solutions. The above observations lead to the invention of the MUSIC (MULTiple SIgnal Characterization) algorithm [9, 10]. First one computes the eigendecomposition of the sample correlation matrix

$$\widehat{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}^H(n) = \widehat{\mathbf{E}}_s \widehat{\Lambda}_s \widehat{\mathbf{E}}_s^H + \widehat{\mathbf{E}}_n \widehat{\Lambda}_n \widehat{\mathbf{E}}_n^H, \quad (11.49)$$

similarly to (11.48). The number of signals can be determined by using the fact that the multiplicity of the smallest eigenvalue is  $M - P$  [50, 51], although this is far from a trivial task in a practical case. Given a finite number of samples only,  $\mathbf{a}^H(\theta)\widehat{\mathbf{E}}_n = 0$  will not hold for any value of  $\theta$ . A visually attractive solution is to search for the  $P$  largest peaks of the so-called MUSIC pseudo-spectrum

$$P_{\text{MUSIC}}(\theta) = \frac{1}{\|\mathbf{a}^H(\theta)\widehat{\mathbf{E}}_n\|^2} = \frac{1}{\mathbf{a}^H(\theta)\widehat{\mathbf{E}}_n \widehat{\mathbf{E}}_n^H \mathbf{a}(\theta)}. \quad (11.50)$$

Note that this is not a spectrum in the sense of, e.g., (11.45), since it does not have the dimension of power. However, it is clear that the locations of the peaks will coincide with the true DOAs when either  $\sigma^2 \rightarrow 0$  or  $N \rightarrow \infty$ , because the noise subspace will then be correctly estimated. Thus, the MUSIC

**FIGURE 11.16**

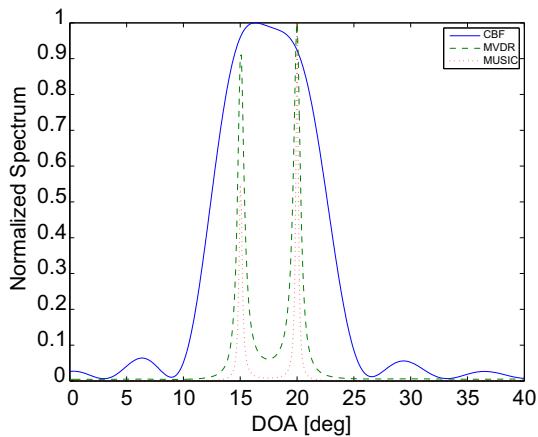
Normalized spectra for the Conventional Beam Former (CBF), MVDR and MUSIC methods, assuming a 20-element ULA and  $N = 100$  samples. Two 10 dB uncorrelated sources are present at  $\theta_1 = 12^\circ$  and  $\theta_2 = 20^\circ$  respectively.

estimator is statistically *consistent*. However, in finite samples it too has a limited resolution. See e.g., [52] or Chapter 16 for details. In addition, the MUSIC algorithm is unable to cope with the presence of *multipath propagation*, since  $\mathbf{R}_s$  is required to have full rank. We refer to Chapter 14 for extensions of MUSIC leading to increased resolution and/or reduced sensitivity to multipath, and also to Chapter 15 for subspace-based methods that exploit special array structures and that work in higher dimensions. Example 1 illustrates the performance of the three spectral based methods presented here in a simple scenario of two uncorrelated sources.

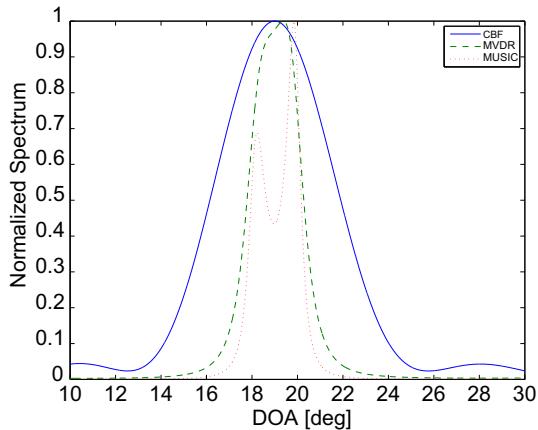
**Example 1 (Resolution of Spectral-Based DOA Estimation Methods).** Suppose a standard ULA with  $M = 20$  elements. Two uncorrelated sources are present, both with 10 dB power above the white noise floor. The Classical Beam Forming (CBF), MVDR and MUSIC spectra are calculated based on  $N = 100$  snapshots of data. Figure 11.16 shows one (typical) realization of the spectra for a DOA separation of  $8^\circ$ . In this case, all three methods are able to resolve the sources with good results. In Figure 11.17, the DOA separation is reduced to  $5^\circ$ . This is within the beamwidth, so CBF can no longer resolve the sources, but both MVDR and MUSIC provide accurate estimates. When the sources are moved even closer, as in Figure 11.18, also the MVDR fails to resolve them in most realizations while MUSIC still succeeds with high probability.

### 3.11.5.3 Parametric methods

Similar to any estimation problem, the data model (11.44) can be cast in a statistical framework in order to arrive at an estimator with certain optimality properties. In most of the literature, the noise term  $\mathbf{n}(n)$  is assumed to be temporally and spatially white with a circularly symmetric Gaussian distribution.

**FIGURE 11.17**

Normalized spectra for the Conventional Beam Former (CBF), MVDR and MUSIC methods, assuming a 20-element ULA and  $N = 100$  samples. Two 10 dB uncorrelated sources are present at  $\theta_1 = 15^\circ$  and  $\theta_2 = 20^\circ$  respectively.

**FIGURE 11.18**

Normalized spectra for the Conventional Beam Former (CBF), MVDR and MUSIC methods, assuming a 20-element ULA and  $N = 100$  samples. Two 10 dB uncorrelated sources are present at  $\theta_1 = 18^\circ$  and  $\theta_2 = 20^\circ$  respectively.

Considering the signal waveforms  $\mathbf{s}(n)$ , they can either be regarded as unknown deterministic parameters to be estimated along with  $\boldsymbol{\theta}$ , or they too can be given a statistical model, which is typically temporally white and  $\mathcal{N}(0, \mathbf{R}_s)$ . In some applications, more structure is available, for example that the signals are complex sinusoids or that their transposed correlation matrix  $E[\mathbf{s}(n)\mathbf{s}^T(n)]$  is non-zero. For the sake of

brevity, we only consider the case where  $\mathbf{s}(n)$  is deterministic and unknown. The corresponding ML estimator is usually referred to as Deterministic ML (DML), or Conditional ML (CML) in the literature. Under the stated assumptions, the data  $\mathbf{x}(n)$  is temporally white and distributed as  $\mathcal{N}(\mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n), \sigma^2 \mathbf{I})$ . Thus, the negative log-likelihood function, ignoring constants, is given by

$$l(\boldsymbol{\theta}, \{\mathbf{s}(n)\}_{n=1}^N, \sigma^2) = NM \log \sigma^2 + \frac{1}{\sigma^2} \sum_{n=1}^N \|\mathbf{x}(n) - \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n)\|^2. \quad (11.51)$$

The ML estimates of  $\boldsymbol{\theta}$ ,  $\{\mathbf{s}(n)\}_{n=1}^N$  and  $\sigma^2$  are the minimizing arguments of  $l(\cdot)$ . It is straightforward to show that for a fixed, although yet unknown, value of  $\boldsymbol{\theta}$ ; minimizing (11.51) w.r.t. the noise variance and the signal waveforms yields

$$\hat{\mathbf{s}}(n) = \mathbf{A}^\dagger(\boldsymbol{\theta})\mathbf{x}(n), \quad (11.52)$$

$$\hat{\sigma}^2 = \frac{1}{M} \text{Tr} \left\{ \boldsymbol{\Pi}_\mathbf{A}^\perp \widehat{\mathbf{R}}_x \right\}, \quad (11.53)$$

where  $\mathbf{A}^\dagger = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  is the Moore-Penrose pseudo-inverse, Tr is the matrix trace operator, and  $\boldsymbol{\Pi}_\mathbf{A}^\perp = \mathbf{I} - \boldsymbol{\Pi}_\mathbf{A} = \mathbf{I} - \mathbf{A}\mathbf{A}^\dagger$  is the projection matrix onto the orthogonal complement of the matrix  $\mathbf{A} = \mathbf{A}(\boldsymbol{\theta})$ . Inserting (11.52) and (11.53) into (11.51) shows that the ML DOA parameters are obtained as the minimizing arguments of the function [53, 54]

$$l(\boldsymbol{\theta}) = \text{Tr} \left\{ \boldsymbol{\Pi}_\mathbf{A}^\perp \widehat{\mathbf{R}}_x \right\}. \quad (11.54)$$

The close relation to the noise variance estimate (11.53) shows a nice interpretation of (11.54). The projection tries to remove all signal components of  $\widehat{\mathbf{R}}_x$ , and then the trace measures the remaining (noise) power. Clearly, this should be smallest when  $\boldsymbol{\theta}$  is close to  $\boldsymbol{\theta}_0$ , but since the noise will have random components along both the signal and the noise subspaces, the estimates will not be exact in finite samples. By the general theory of ML estimation, we expect  $\hat{\boldsymbol{\theta}}_{\text{ML}} = \arg \min l(\boldsymbol{\theta})$  to be an approximately minimum variance unbiased estimate in large samples. However, since the number of parameters increases with increasing  $N$ , this turns out to hold only for large enough  $M$  (or small  $\sigma^2$ ). Indeed, for correlated sources, the ML estimator derived under the Gaussian assumption, termed Stochastic ML (SML) [55], can result in improved DOA estimates, but the difference is often negligible in practical scenarios. We refer to Chapters 14 and 16 of this book for more details and comparisons. Although the DML estimator has a nice formulation (11.54), its computation is far from trivial. The function  $l(\boldsymbol{\theta})$  has in general multiple local minima, and to find the global optimum with certainty may require a full  $P$ -dimensional grid search. Several optimization methods have been proposed in the literature, see [56, 57] and Chapter 14 of this book for more details. In this introductory chapter, we mention briefly a popular iterative procedure related to the SAGE algorithm of [58], and termed RELAX in [59], since it is a relaxed optimization procedure. The idea is to compute the DOA estimates by optimizing  $l(\boldsymbol{\theta}, \{\mathbf{s}(n)\}_{n=1}^N)$  (the noise variance can be ignored, since its value is not needed for the other parameters) with respect to one pair  $\{\theta_p, s_p(n)\}$  at the time, keeping the others fixed. This is achieved by subtracting the already estimated components (if any) from  $\mathbf{x}(n)$ , forming the “cleaned” signal  $\mathbf{x}_p(n) = \mathbf{x}(n) - \sum_{k \neq p} \mathbf{a}(\hat{\theta}_k) \hat{s}_k(n)$ . Then, updated parameter estimates for signal  $p$  are computed as

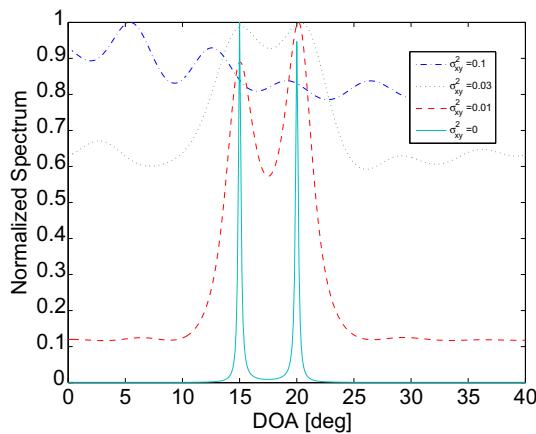
$$\hat{\theta}_p = \arg \min_{\theta} \text{Tr} \left\{ \boldsymbol{\Pi}_{\mathbf{a}(\theta)}^{\perp} \widehat{\mathbf{R}}_{x_p} \right\} = \arg \max_{\theta} \frac{\mathbf{a}^H(\theta) \widehat{\mathbf{R}}_{x_p} \mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \mathbf{a}(\theta)},$$

$$\hat{s}_p(n) = \mathbf{a}^{\dagger} (\hat{\theta}_p) \mathbf{x}_p(n),$$

where  $\widehat{\mathbf{R}}_{x_p}$  is the sample correlation matrix of  $\mathbf{x}_p(n)$ . One iteration of the algorithm now constitutes applying the above procedure for  $p = \{1, 2, \dots, P\}$ , and the iterations continue until some convergence criterion is met, such as that the DOA estimates do not change significantly between two iterations. The SAGE/RELAX algorithm and its variations (mostly varying the order in which the estimates are computed) has been found to be quite successful for “nice” arrays, that do not exhibit “too high” sidelobes in their beam pattern.

### 3.11.5.4 Modeling errors and array calibration

It is clear that any model-based estimator requires a reliable description of the sensor response as a function of the parameter to be estimated. Concerning DOA estimation, this means that the steering vector  $\mathbf{a}(\theta)$  must be a known function of the DOA parameter. As explained in Section 3.11.2.3, the response of a real-world sensor array can be quite different from the ideal wave-field model introduced in Section 3.11.2.2. This will inevitably lead to a performance degradation of the DOA estimator. To illustrate this effect, consider the same scenario as in Example 1. The source locations are fixed at  $\theta_1 = 15^\circ$  and  $\theta_2 = 20^\circ$ . However, the sensor positions in the  $xy$ -plane (normalized to the wavelength) are perturbed by i.i.d. Gaussian random variables with zero mean and variance  $\sigma_{xy}^2$ . A “typical” realization of the resulting MUSIC spectra is plotted in Figure 11.19, for several values of  $\sigma_{xy}^2$ . Although MUSIC is



**FIGURE 11.19**

Normalized spectra for the MUSIC method, assuming  $N = 100$  samples and two 10 dB uncorrelated sources at  $\theta_1 = 15^\circ$  and  $\theta_2 = 20^\circ$ . The  $x$  and  $y$  coordinates of the nominal  $M = 20$ -element standard ULA are perturbed by i.i.d.  $\mathcal{N}(0, \sigma_{xy}^2)$  random variables, where  $\sigma_{xy}^2$  is varied.

reasonably robust to certain types of modeling errors [60], the performance degrades as  $\sigma_{xy}^2$  is increased, and in particular the ability to resolve closely spaced sources rapidly decreases. For  $\sigma_{xy} > 0.2$ , there is a significant probability that the sensors switch place (the nominal sensor separations are 0.5 wavelengths), leading to a breakdown of any DOA estimation procedure. This example clearly illustrates the need for array calibration when large modeling errors are present. There are two main approaches to calibration. The first attempt, termed auto-calibration, is to estimate certain array parameters along with the unknown DOAs. The array parameters can include individual sensor gains, phases and positions and/or mutual coupling parameters. In general, this leads to a large number of unknown parameters and thus high sensitivity to noise, or even non-uniqueness. The other approach is to measure the array response in a controlled experiment, using sources at known positions. A possible alternative to generate such data artificially is a full-scale electromagnetic (or acoustic) simulation, provided sufficient details of the sensors and receiver circuitry as well as the environment surrounding the array is available. Chapter 19 of this book treats DOA estimation using real-world sensor arrays in more detail.

### 3.11.6 Non-coherent array applications

Classical DOA estimation deals with signals that are perfectly coherent at the different sensors. Yet, the data model (11.11) and hence the associated estimation methods have found application in a vast array of applications that do not involve coherent multi-sensor array data. This section gives a brief introduction to such generalizations of DOA estimation, starting with the concept of spatially spread sources.

#### 3.11.6.1 Spread sources

In certain applications of array signal processing, the received signal from one source is not perfectly coherent in all sensors, as predicted by the point-source model (11.6). This may be due to multipath or to propagation through a random medium. In effect, the signal then appears to be coming from a cluster of arrival angles, rather than from just one. A commonly adopted signal model for such situations is that of a randomly time-varying “spatial signature” vector  $\mathbf{v}(n)$ , which has zero mean due to the random phase of the scattering, and a correlation matrix that contains information of the scenario at hand. Thus, the received signal at the sensor array due to a transmitted signal  $s(n)$  is modeled by

$$\mathbf{x}(n) = \mathbf{v}(n)s(n). \quad (11.55)$$

Due to the random nature of the propagation, the individual realizations of the spatial signatures are not useful, but their statistics may reveal important information that can be used, e.g., to localize the signal source (by the mean angle) and to characterize the nature of the propagation channel. A natural model for the correlation matrix is given by

$$\mathbf{R}_v = E[\mathbf{v}(n)\mathbf{v}^H(n)] = \int_{\theta} \mathbf{a}(\theta)\mathbf{a}^H(\theta)p_{\theta}(\theta)d\theta, \quad (11.56)$$

where  $\mathbf{a}(\theta)$  is the steering vector, i.e., the response to a signal from the direction  $\theta$ , and  $p_{\theta}(\theta)d\theta$  represents the part of the signal power that comes from the direction in question. Thus, one can view

$p_\theta(\theta)$  as a Probability Density Function (PDF) over the DOAs in a cluster. In general, the precise form of  $p_\theta(\theta)$  is not important, and also very difficult to predict, but its mean angle and spread parameter is of interest. Therefore,  $p_\theta(\theta)$  is often taken as Gaussian  $\mathcal{N}(\theta_0, \sigma_\theta)$ , and  $\theta_0$  and  $\sigma_\theta$  become the parameters to be estimated. In the presence of spatially white noise, the array output  $\mathbf{x}(n)$  will also be zero mean, and its correlation matrix is given by

$$\mathbf{R}_x = \sigma_s^2 \mathbf{R}_v(\theta_0, \sigma_\theta) + \sigma^2 \mathbf{I}, \quad (11.57)$$

where  $\sigma_s^2$  is the signal power. Depending on the sampling time,  $\mathbf{x}(n)$  may or not be temporally correlated, but it is clear that a “sufficiently long” observation time is necessary in order for  $\mathbf{v}(n)$  to reveal enough information of  $\mathbf{R}_v$ . Thus, assume that a batch of  $N$  samples of  $\mathbf{x}(n)$  are available, where  $N$  is large enough so that the sample correlation  $\widehat{\mathbf{R}}_x$  is “close to” the true  $\mathbf{R}_x$  in (11.57).

A very simple yet effective technique in the one-cluster case is proposed in [61]. The idea is to exploit a spatial spectrum estimate  $\widehat{P}(\theta)$ , as given, e.g., by the conventional or the MVDR beamformer. Provided the resolution is sufficient, the spatial spectrum is approximately proportional to  $p_\theta(\theta)$ . Thus, the mean DOA and the spread can be estimated as

$$\hat{\theta}_0 = \frac{\int_{\theta \in \Theta} \theta \widehat{P}(\theta) d\theta}{\int_{\theta \in \Theta} \widehat{P}(\theta) d\theta}, \quad (11.58)$$

$$\hat{\sigma}^2 = \frac{\int_{\theta \in \Theta} (\theta - \hat{\theta}_0)^2 \widehat{P}(\theta) d\theta}{\int_{\theta \in \Theta} \widehat{P}(\theta) d\theta}, \quad (11.59)$$

where  $\Theta$  is the expected support of the DOA cluster. In general, the above non-parametric estimate of  $\hat{\theta}_0$  is accurate, whereas  $\hat{\sigma}^2$  is more sensitive to the choice of  $\Theta$  and to the resolution of the beamformer.

For multiple clusters, one may extend (11.57) to

$$\mathbf{R}_x = \sum_{p=1}^P \sigma_p^2 \mathbf{R}_v(\theta_p, \sigma_{\theta_p}) + \sigma^2 \mathbf{I}, \quad (11.60)$$

where  $P$  is the number of clusters. In this case, a parametric technique is preferable. The most effective methods are based on covariance matching [62]. Since these are computationally costly, suboptimal techniques have been proposed, that only require searching over the mean DOA and spread parameters. An interesting class of methods is based on generalizations of spectral-based techniques for point sources, e.g., the MVDR spectrum [63, 64]. In particular, in the method of [64], the parameter estimates are found by localizing the  $P$  smallest minima of the 2D function

$$P(\theta, \sigma_\theta) = \left\| \widehat{\mathbf{R}}_x^{-1} \mathbf{R}_v(\theta, \sigma_\theta) \right\|_F = \text{Tr} \left\{ \widehat{\mathbf{R}}_x^{-2} \mathbf{R}_v^2(\theta, \sigma_\theta) \right\}, \quad (11.61)$$

where  $\|\cdot\|_F$  denotes the Frobenius matrix norm. Clearly, for a point source model,  $p_\theta(\theta) = \delta(\theta)$  and  $\mathbf{R}_v = \mathbf{a}(\theta_0)\mathbf{a}^H(\theta_0)$ , so if  $\|\mathbf{a}(\theta)\|^2 = M$ , (11.61) reduces to a non-parametric version of the Pisarenko family of methods [65].

An important special case of (11.55) is the case of a multiplicative noise, for example due to a randomly time-varying medium. This is of relevance, e.g., in sonar applications. In this case, the received signal is modeled as

$$\mathbf{x}(n) = (\mathbf{a}(\theta) \odot \mathbf{g}(n)) s(n) + \mathbf{n}(n), \quad (11.62)$$

where  $\odot$  denotes the Schur product (elementwise multiplication). The ML estimator to this problem is derived in [66], assuming  $\mathbf{g}(n)$  to be a sequence of deterministic quantities to be estimated along with  $\theta$ . The resulting estimator has a surprisingly simple form, reminiscent of beamforming using the squared data  $\mathbf{x}(n) \odot \mathbf{x}(n)$  with the corresponding steering vector  $\mathbf{a}(\theta) \odot \mathbf{a}(\theta)$ .

### 3.11.6.2 Time series modeling

We have previously noted the close relation between spatial filtering using a ULA and the DTFT. Thus, it should not come as a surprise that methods derived for array processing are also useful for modeling time series, even if only one sensor is available. The closest case is that of a pure complex sinusoidal signal

$$s(n) = A e^{j\omega t}. \quad (11.63)$$

Suppose a batch of  $K$  samples of  $s(n)$ ,  $n = 1, \dots, K$  are available. Choosing an equivalent “array size”  $M$ ,  $1 < M \leq K$ , we can form  $N = K - M$  vector-valued “array outputs”

$$\mathbf{x}(n) = \begin{bmatrix} s(n) \\ s(n+1) \\ \vdots \\ s(n+M-1) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{j\omega} \\ \vdots \\ e^{j(M-1)\omega} \end{bmatrix} s(n) = \mathbf{a}(\omega) s(n). \quad (11.64)$$

Clearly, this model is identical to (11.6), here with  $\mathbf{a}(\omega)$  representing a DTFT vector. In the presence of multiple sinusoids and noise, the model is identical to the spatial data model (11.11), with the only difference that the noise vector samples will be temporally correlated due to the overlapping noise samples in adjacent array outputs. Provided the scalar measurement noise is temporally white, it still holds that the vector-valued noise is spatially white. Thus, all methods for DOA estimation are readily available also for estimating the frequencies of superimposed sinusoids in white noise. Since the steering vector has the form of a Vandermonde vector, as in the ULA case, the vast flora of computationally efficient methods developed for such arrays are applicable, see Chapter 15 for details. It should be noted that the same model (11.64) can be derived also for damped sinusoids, although the steering vector is then a function of both the frequency and the damping, and the signal is obviously no longer stationary.

Next, consider the problem of blind multi-channel FIR modeling. Thus, an unknown signal  $s_n$  is transmitted and captured at an  $m$ -element antenna array. The receiver wishes to decode the transmitted message, and therefore needs an estimate of the propagation channel. The received signal  $\mathbf{x}_n \in \mathcal{C}^m$  is assumed to be an FIR filtered version of the transmitted signal,

$$\mathbf{x}_n = \sum_{l=0}^{L-1} \mathbf{h}_l s_{n-l} + \mathbf{n}_n, \quad (11.65)$$

where  $\{\mathbf{h}_l\}_{l=0}^L$  are the vector-valued channel taps to be estimated and  $\mathbf{n}_n$  is the receiver noise. It should be noted that the same model can be obtained also for a scalar receiving antenna, provided over-sampling by a factor of  $m$  is used (see [67] for details). Similar to (11.64), we choose an equivalent array size  $M$  and form the array output vector (this time sampling backwards, due to the causality in (11.65))

$$\mathbf{x}(n) = \begin{bmatrix} \mathbf{x}_n \\ \mathbf{x}_{n-1} \\ \vdots \\ \mathbf{x}_{n-M+1} \end{bmatrix} = \mathbf{A}(\mathbf{h})\mathbf{s}(n) + \mathbf{n}(n), \quad (11.66)$$

where we have defined the signal vector

$$\mathbf{s}(n) = [s_n, s_{n-1}, \dots, s_{n-L-M+2}]^T, \quad (11.67)$$

and the “steering matrix”  $\mathbf{A}(\mathbf{h})$  is given by

$$\mathbf{A}(\mathbf{h}) = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \dots & \mathbf{h}_{L-1} & 0 & \dots & 0 \\ 0 & \mathbf{h}_1 & \ddots & & \mathbf{h}_{L-1} & & \vdots \\ \vdots & & \ddots & & & & 0 \\ 0 & \dots & 0 & \mathbf{h}_1 & \mathbf{h}_2 & \dots & \mathbf{h}_{L-1} \end{bmatrix}. \quad (11.68)$$

It is clear that (11.66) resembles the DOA estimation model (11.11), except that  $\mathbf{A}(\mathbf{h})$  has a rather different structure. The fact that  $\mathbf{A}(\mathbf{h})$  is linear in its parameters,  $\mathbf{h} = [\mathbf{h}_0^T, \dots, \mathbf{h}_{L-1}^T]^T$ , can of course be exploited. Note that  $\mathbf{A}(\mathbf{h})$  is full column rank by construction, as long as  $\mathbf{h}_0$  and  $\mathbf{h}_{L-1}$  are both different from the null vector (implying that the FIR filter order must be properly chosen). Its dimensions are  $Mm \times (M + L - 1)$ , so if  $Mm > M + L - 1$ , i.e.,  $M > (L - 1)/(m - 1)$ , then  $\mathbf{A}^H(\mathbf{h})$  has an  $M - (L - 1)/(m - 1)$ -dimensional null-space. Similarly to (11.49), an estimate  $\widehat{\mathbf{E}}_n$  of that nullspace can be computed from the minor eigenvectors of the sample covariance matrix of  $\mathbf{x}(n)$ . Ideally  $\mathbf{A}(\mathbf{h}) \perp \mathbf{E}_n$ , so the following blind FIR channel estimator is now natural [67],

$$\hat{\mathbf{h}} = \arg \min_{\mathbf{h}} \left\| \mathbf{A}^H(\mathbf{h}) \widehat{\mathbf{E}}_n \right\|_F^2. \quad (11.69)$$

The linearity of  $\mathbf{A}(\mathbf{h})$  in  $\mathbf{h}$  means that (11.69) is nothing but a linear least-squares problem, which can be solved very efficiently (although its dimensions can be large).

Subspace methods have also been successfully applied in a more general system identification setting, see, e.g., [15, 68, 69]. The problem then is to determine a general finite-dimensional model of a linear multi-variable input-output system along with a stochastic model of the disturbances and noise from measurements of input-output data. The case of purely stochastic modeling (output-data only), similar to the multi-channel blind FIR modeling outlined above, is also treated. The subspace techniques are capable of providing consistent estimates of a multi-variable state-space model of the data, without resorting to iterative non-linear search procedures as required by traditional approaches. The methods are now standard components in system identification software, such as Matlab’s System Identification Toolbox.

### 3.11.6.3 Source localization in sensor networks

Conventional DOA estimation methods are based on data collected by a relatively small sensor array in the far-field of the sources. The received data due to a single point source is usually assumed to be perfectly coherent among the different sensors. In contrast, a sensor network consists of a set of widely separated sensors, where each sensor can be a small array itself. Thus, it is not realistic to assume perfect phase synchronization between the sensors and the processing cannot be done coherently among all sensors. In return, the wide separation of sensors implies that the complete 2D (or even 3D) positions of signal sources can be determined, and not just the directions of incoming sources, provided sufficiently many of the sensors receive the transmitted signal. The sensors may be of many different kinds, such as electromagnetic antennas, acoustic sensors, infrared or other environmental monitoring systems. In most cases, they communicate by wireless communication links. Common to these applications is the desire to reduce energy consumption and thus to increase battery lifetime. Thus, GPS may not be available and a major problem can be to localize the sensors themselves [6].

In general, source localization in sensor networks is based on one or more of the following phenomena:

- Direction-of-Arrival (DOA) from several sensor arrays.
- Time Difference-of-Arrival (TDOA) between sensor pairs.
- Signal strength measurements.

DOA-based estimation is possible by clustering sensors into coherent groups (subarrays), and measuring the DOA from each such subarray. The estimated DOAs are then transmitted to a central node, that performs a triangulation-type fusion of the various sensor data to determine the source position. DOA-based estimation can be done using narrowband or wideband data. In contrast, TDOA-based source localization requires wideband data to determine the relative time-delays between the signals received at the various sensors. We refer to [70] for a review of such methods. A practical approach that requires relatively little inter-sensor communication is to use the signal strength received at the various sensors. If a model of the power loss as a function of the distance to the transmitter is known, this can be used to combine the various power measurements at a central node. See [71] for details, and Chapter 18 of this book for a more complete overview of the source localization problem.

### 3.11.6.4 Microwave and ultrasound imaging

Classical radar assumes that targets are in the far-field, and thus can be modeled as point sources. The situation is quite different in ultrasonic imaging of a human body, see, e.g., [72]. In recent years there has also been an increasing interest in near-field radar applications, including microwave imaging for breast cancer [73, 74]. Although in most cases one is still concerned with one or a few strong sources of backscattered energy, the medium of propagation causes the received signal to be severely distorted. Thus, DOA estimation techniques designed for point sources are therefore not applicable. The spread source modeling approach of Section 3.11.6.1 can be useful, but in general the presence of severe background noise requires special treatment. The most common approach is to view the problem as a spatial spectrum estimation and apply beamforming techniques similar to the MVDR

beamformer (11.35) [72–74]. A promising emerging technique is to apply a full-scale electromagnetic (or acoustic) model of the scenario, including sensor and receiver characteristics as well as the “target” in its environment (e.g., a cancer tumor) [75]. Although this is computationally very costly, the rapid progress in numerical computation and parallel processing makes it a promising candidate for future diagnostic imaging systems.

An application facing similar problems is through-the-wall imaging using a wideband radar technique. For the case of a relatively simple scenario with a known obstacle (wall), a geometric beamforming-like approach is taken in [76]. In more complicated scenarios one can envision that numerical approaches similar to [75] can give more accurate localization and increased resolution, but the physical extent of the scenario (in wavelengths) makes the computational load challenging. In interferometric Synthetic Aperture Radar (SAR) imaging, two slightly displaced antenna arrays are used to create a topographic map of the earth. In this case the narrowband assumption can be considered to hold, and the effect of the random medium can therefore be modeled as a multiplicative noise, similar to (11.62) but with  $\mathbf{g}(n)$  a constant. In [77], several beamforming techniques are evaluated together with parametric techniques. A combination of using MVDR (11.46) DOA estimation and Least-Squares amplitude estimation (11.52) is favored.

---

### 3.11.7 Concluding remarks

Originating from the use of antenna arrays in radar more than a century ago, array signal processing has emerged as an active field of research which has influenced a variety of application areas. We have mentioned several of these in this introductory chapter, and more are presented in Chapter 20. One obvious reason for the success is that multi-sensor processing is becoming increasingly more important, and the power of DSP is readily available at low cost and ease of use. In addition, as we have seen for example in Section 3.11.6, the generic sensor array model (11.10) finds applications way beyond localization of plane waves arriving at an array of coherent sensors. See also [78] for an extensive list of applications of the separable Non-Linear Least-Squares (NLLS) formulation (11.51). All of these are candidates for trying other array signal processing methods than NLLS, some of which are mentioned herein. Some of the emerging topics in sensor array and multi-channel signal processing are surveyed in [79]. Thus, we may expect to see more influence of MIMO technology in practical systems, both for wireless communication (where it is already included in the LTE and WiMAX standards) and radar. Not the least important is near-field microwave tomography using MIMO radar, which so far has focused mostly on biomedical applications. Measurements that are resolved both in space and time are sought for in the entire process industry (chemical processes, forestry, food processing etc.), and microwave tomography has the potential to revolutionize this area. Techniques for distributed processing are also an emerging topic that may walk into our everyday lives as the world becomes more and more connected. In terms of mathematical methods, convex optimization has finally become a ubiquitous tool for array signal processing researchers, and we foresee more attempts to marry “conventional” model-based estimation techniques with machine learning. A particularly promising arena for such attempts is sparse signal modeling, see, e.g., [80].

### *Relevant Theory:* Signal Processing Theory

- See [Vol. 1, Chapter 1](#) Introduction: Signal Processing Theory  
 See [Vol. 1, Chapter 2](#) Continuous-Time Signals and Systems  
 See [Vol. 1, Chapter 3](#) Discrete-Time Signals and Systems  
 See [Vol. 1, Chapter 4](#) Random Signals and Stochastic Processes  
 See [Vol. 1, Chapter 9](#) Discrete Multi-Scale Transforms in Signal Processing  
 See [Vol. 1, Chapter 11](#) Parametric Estimation  
 See [Vol. 1, Chapter 12](#) Adaptive Filters

---

## References

- [1] R.J. Mailloux, *Phased Array Antenna Handbook*, second ed., Artech House, Boston and London, UK, 2005.
- [2] N. Owsley, Sonar array processing, in: S. Haykin (Ed.), *Array Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.
- [3] A. Farina, *Antenna-Based Signal Processing Techniques for Radar Systems*, Artech House, Norwood, MA, 1992.
- [4] A. Paulraj, R. Nabar, D. Gore, *Introduction to Space-Time Wireless Communications*, Cambridge University Press, Cambridge, UK, 2003.
- [5] A. Singer, J. Nelson, S. Kozat, Signal processing for underwater acoustic communications, *IEEE Commun. Mag.* 47 (1) (2009) 90–96.
- [6] N. Patwari, J. Ash, S. Kyperountas, A.O. Hero, R. Moses, N. Correal, Locating the nodes: cooperative localization in wireless sensor networks, *IEEE Signal Process. Mag.* 22 (4) (2005) 54–69.
- [7] H. Krim, M. Viberg, Two decades of array signal processing research: the parametric approach, *IEEE Signal Process. Mag.* 13 (4) (1996) 67–94.
- [8] H.V. Trees, *Optimum Array Processing*, John Wiley & Sons, Canada, 2002.
- [9] G. Bienvenu, L. Kopp, Principle de la goniometrie passive adaptive, in: Proceedings of 7'eme Colloque GRESTIT, Nice, France, 1979, pp. 106/1–106/10.
- [10] R. Schmidt, Multiple emitter location and signal parameter estimation, in: Proceedings of RADC Spectrum Estimation Workshop, Rome, NY, 1979, pp. 243–258.
- [11] G. Foschini, M. Gans, On limits of wireless communications in a fading environment when using multiple antennas, *Wireless Personal Commun.* 6 (1998) 311–335.
- [12] J. Li, P. Stoica, MIMO radar with colocated antennas, *IEEE Signal Process. Mag.* 24 (5) (2007) 106–114.
- [13] A. Haimovich, R. Blum, L.J. Cimini, MIMO radar with widely separated antennas, *IEEE Signal Process. Mag.* 25 (1) (2008) 116–129.
- [14] P. Stoica, R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, NJ, 2005.
- [15] M. Viberg, On subspace-based methods for the identification of linear time-invariant systems, *Automatica* 31 (12) (1995) 1835–1851.
- [16] P. Van Overschee, B. De Moor, *Subspace Identification of Linear Systems: Theory, Implementation, Applications*, Kluwer Academic Publishers, 1996.
- [17] A. Nehorai, E. Paldi, Vector-sensor array processing for electromagnetic source localization, *IEEE Trans. Signal Process.* 42 (2) (1994) 376–398.
- [18] J. Hudson, *Adaptive Array Principles*, Peter Peregrinus, 1981.
- [19] B.V. Veen, K. Buckley, Beamforming: a versatile approach to spatial filtering, *IEEE Acoust. Speech Signal Process. Mag.* (1988) 4–24.

- [20] R. Monzingo, T. Miller, *Introduction to Adaptive Arrays*, SciTech Publishing, Inc., Raleigh, NC, 2004.
- [21] H. Steyskal, J. Herd, Mutual coupling compensation in small array antennas, *IEEE Trans. Antennas Propag.* 38 (1990).
- [22] S. Mitra, *Digital Signal Processing—A Computer-Based Approach*, second ed., McGraw-Hill, New York, 2001.
- [23] A. Moffet, Minimum-redundancy linear arrays, *IEEE Trans. Antennas Propag.* 16 (2) (1968) 172–175.
- [24] H. Pumphrey, Design of sparse arrays in one, two, and three dimensions, *J. Acoust. Soc. Am.* 93 (3) (1993) 1620–1628.
- [25] T. Laakso, V. Välimäki, M. Karjalainen, U. Laine, Splitting the unit delay, *IEEE Signal Process. Mag.* 13 (1996) 30–60.
- [26] F. Harris, On the use of windows for harmonic analysis with the discrete Fourier transform, *Proc. IEEE* 66 (1) (1978) 51–83.
- [27] P. Stoica, R. Moses, *Introduction to Spectral Analysis*, Prentice Hall, Upper Saddle River, NJ, 1997.
- [28] B. Lau, Y. Leung, A Dolph-Chebyshev approach to the synthesis of array patterns for uniform circular arrays, in: *Proceedings ISCAS*, Geneva, Switzerland, vol. 1, 2000, pp. 124–127.
- [29] M. Doron, E. Doron, Wavefield modeling and array processing, part I—spatial sampling, *IEEE Trans. Signal Process.* 42 (10) (1994) 2549–2559.
- [30] H. Lebret, S. Boyd, Antenna array pattern synthesis via convex optimization, *IEEE Trans. Signal Process.* 45 (3) (1997) 526–532.
- [31] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.
- [32] M. Grant, S. Boyd, Y. Ye, CVX: Matlab software for disciplined convex programming, 2008. <<http://www.stanford.edu/boyd/cvx/>>.
- [33] J. Li, P. Stoica, *Robust Adaptive Beamforming*, John Wiley & Sons, Inc., Hoboken, NJ, 2006.
- [34] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall International Editions, Englewood Cliffs, NJ, 1998.
- [35] I. Reed, J. Mallett, L. Brennan, Rapid convergence rate in adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 10 (6) (1974) 853–863.
- [36] J. Li, P. Stoica, Z. Wang, On robust capon beamforming and diagonal loading, *IEEE Trans. Signal Process.* 51 (7) (2003) 1702–1715.
- [37] S. Vorobyov, A. Gershman, Z.-Q. Luo, Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem, *IEEE Trans. Signal Process.* 51 (2) (2003) 313–324.
- [38] D. Boroson, Sample size considerations for adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 16 (4) (1980) 446–451.
- [39] F. Gustavsson, *Adaptive Filtering and Change Detection*, John Wiley & Sons, Ltd., Chichester, UK, 2000.
- [40] A. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, Ltd., Hoboken, NJ, 2003.
- [41] O.L. Frost III, An algorithm for linearly constrained adaptive array processing, *Proc. IEEE* 60 (8) (1972) 926–935.
- [42] B. Widrow, P. Mantey, L. Griffiths, B. Goode, Adaptive antenna systems, *Proc. IEEE* 55 (1967) 2143–2159.
- [43] B. Ottersten, R. Roy, T. Kailath, Signal waveform estimation in sensor array processing, in: *Proceedings of 23rd Asilomar Conference on Circuits, Systems and Computers*, November 1989, pp. 787–791.
- [44] F. Robey, D. Fuhrmann, E. Kelly, R. Nitzberg, A CFAR adaptive matched filter detector, *IEEE Trans. Aerosp. Electron. Syst.* 28 (1) (1992) 208–216.
- [45] E. Kelly, An adaptive detection algorithm, *IEEE Trans. Aerosp. Electron. Syst.* 22 (2) (1986) 115–127.
- [46] A. Leshem, A.-J.V. der Veen, Direction-of-arrival estimation for constant modulus signals, *IEEE Trans. Signal Process.* 47 (11) (1999) 3125–3129.

- [47] G. Xu, T. Kailath, Direction-of-arrival estimation via exploitation of cyclostationary—a combination of temporal and spatial processing, *IEEE Trans. Signal Process.* 40 (7) (1992) 1775–1786.
- [48] J. Capon, High resolution frequency wave number spectrum analysis, *Proc. IEEE* 57 (1969) 1408–1418.
- [49] C. Richmond, The CAPON-MVDR algorithm: threshold SNR prediction and the probability of resolution, in: *Proceedings of ICASSP 2004*, Montreal, Canada, vol. 2, May 2004, pp. 217–220.
- [50] M. Wax, T. Kailath, Detection of signals by information theoretic criteria, *IEEE Trans. Acoust. Speech Signal Process.* 33 (2) (1985) 387–392.
- [51] L. Zhao, P. Krishnaiah, Z. Bai, On detection of the number of signals in presence of white noise, *J. Multivariate Anal.* 20 (1) (1986) 1–25.
- [52] M. Kaveh, A.J. Barabell, The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise, *IEEE Trans. Acoust. Speech Signal Process.* 34 (1986) 331–341.
- [53] J. Böhme, Estimation of source parameters by maximum likelihood and nonlinear regression, in: *Proc. ICASSP 84*, vol. 9, 1984, pp. 271–274.
- [54] M. Wax, Detection and estimation of superimposed signals, Ph.D. Dissertation, Stanford University, Stanford, CA, March 1985.
- [55] J. Böhme, Estimation of spectral parameters of correlated signals in wavefields, *Signal Process.* 11 (1986) 329–337.
- [56] P. Stoica, R. Moses, B. Friedlander, T. Söderström, Maximum likelihood estimation of the parameters of multiple sinusoids from noisy measurements, *IEEE Trans. Acoust. Speech Signal Process.* 37 (1989) 378–392.
- [57] M. Viberg, B. Ottersten, Sensor array processing based on subspace fitting, *IEEE Trans. Signal Process.* 39 (5) (1991) 1110–1121.
- [58] J. Fessler, A. Hero, Space-alternating generalized expectation-maximization algorithm, *IEEE Trans. Signal Process.* 42 (10) (1994) 2664–2677.
- [59] J. Li, D. Zheng, P. Stoica, Angle and waveform estimation via RELAX, *IEEE Trans. Aerosp. Electron. Syst.* 33 (3) (1997) 1077–1087.
- [60] A. Swindlehurst, T. Kailath, A performance analysis of subspace-based methods in the presence of model errors—Part I: The MUSIC algorithm, *IEEE Trans. Signal Process.* 40 (7) (1992) 1758–1774.
- [61] M. Tapiro, On the use of beamforming for estimation of spatially distributed signals, in: *Proc. ICASSP 03*, Hong Kong, vol. 3, May 2003, pp. 3005–3008.
- [62] B. Ottersten, P. Stoica, R. Roy, Covariance matching estimation techniques for array signal processing applications, *Digital Signal Process.* 8 (3) (1998) 185–210.
- [63] A. Hassanien, S. Shahbazpanahi, A. Gershman, A generalized Capon estimator for multiple spread sources, *IEEE Trans. Signal Process.* 52 (1) (2004) 280–283.
- [64] A. Zoubir, Y. Wang, P. Charge, Efficient subspace-based estimator for localization of multiple incoherently distributed sources, *IEEE Trans. Signal Process.* 56 (2) (2008) 532–542.
- [65] V.F. Pisarenko, On the estimation of spectra by means of non-linear functions of the covariance matrix, *Geophys. J. Roy. Astron. Soc.* 28 (1972) 522–531.
- [66] P. Stoica, O. Besson, A. Gershman, Direction-of-arrival estimation of an amplitude-distorted wavefront, *IEEE Trans. Signal Process.* 49 (2) (2001) 269–276.
- [67] E. Moulines, P. Duhamel, J.-F. Cardoso, S. Mayrargue, Subspace methods for the blind identification of multichannel FIR filters, *IEEE Trans. Signal Process.* 43 (2) (1995) 516–525.
- [68] P. Van Overschee, B. De Moor, N4SID: subspace algorithms for the identification of combined deterministic-stochastic systems, *Automatica* 30 (1) (1994) 75–93 (special issue on Statistical Signal Processing and Control).

- [69] T. Katayama, Subspace Methods for System Identification, Springer-Verlag, London, UK, 2005.
- [70] J. Chen, K. Yao, R. Hudson, Source localization and beamforming, *IEEE Signal Process. Mag.* 19 (2) (2002) 30–39.
- [71] X. Sheng, Y.-H. Hu, Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks, *IEEE Trans. Signal Process.* 53 (1) (2005) 44–53.
- [72] J.-F. Synnevag, A. Austeng, S. Holm, Adaptive beamforming applied to medical ultrasound imaging, *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* 54 (8) (2007) 1606–1613.
- [73] E. Fear, S. Hagness, P. Meaney, M. Okoniewski, M. Stuchly, Enhancing breast tumor detection with near-field imaging, *IEEE Microwave Mag.* 3 (1) (2002) 48–56.
- [74] Y. Xie, B. Guo, L. Xu, J. Li, P. Stoica, Multistatic adaptive microwave imaging for early breast cancer detection, *IEEE Trans. Biomed. Eng.* 53 (8) (2006) 1647–1657.
- [75] A. Phager, P. Hashemzadeh, M. Persson, Reconstruction quality and spectral content of an electromagnetic time-domain inversion algorithm, *IEEE Trans. Biomed. Eng.* 53 (8) (2006) 1594–1604.
- [76] M. Amin, F. Ahmad, Wideband synthetic aperture beamforming for through-the-wall imaging [lecture notes], *IEEE Signal Process. Mag.* 25 (4) (2008) 110–113.
- [77] F. Gini, F. Lombardini, Multibaseline cross-track SAR interferometry: a signal processing perspective, *IEEE Aerosp. Electron. Syst. Mag.* 20 (8) (2005) 71–93.
- [78] G. Golub, V. Pereyra, Separable nonlinear least squares: the variable projection method and its applications, *Inverse Prob.* 19 (2) (2003) R1–R26 (topical review).
- [79] J. Li, B. Sadler, M. Viberg, Sensor array and multichannel signal processing [in the spotlight], *IEEE Signal Process. Mag.* 28 (5) (2011) 157–158.
- [80] R. Chartrand, R.G. Baraniuk, Y.C. Eldar, M.A.T. Figueiredo, J. Tanner, Introduction to the issue on compressive sensing, *IEEE J. Sel. Top. Signal Process.* 4 (2) (2010) 241–243.

# Adaptive and Robust Beamforming\*

# 12

Sergiy A. Vorobyov

*Department of Signal Processing and Acoustics, Aalto University, FI-00076 AALTO, Finland and Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada*

## 3.12.1 Introduction

Adaptive beamforming is a versatile approach to detect and estimate the signal-of-interest (SOI) at the output of sensor array using data adaptive spatial or spatio-temporal filtering and interference cancellation [1–3]. Being a very central problem of array processing (see [4]), adaptive beamforming has found numerous application to radar [5,6], sonar [7], speech processing [8], radio astronomy [9,10], biomedicine [11,12], wireless communications [13–15], cognitive communications [16], and other fields. The connection of adaptive beamforming to adaptive filtering is emphasized in [4]. The major differences, however, come from the fact that adaptive filtering is based on temporal processing of a signal, while adaptive beamforming stresses on spatial processing. The latter indicates also that the signal is sampled in space, i.e., the signal is measured/observed by an array of spatially distributed antenna elements/sensors. Electronic beamforming design problem consists of computing optimal (in some sense that will be specified) complex beamforming weights for sensor measurements of the signal. If such complex beamforming weights are optimized based on the input/output array data/measurements, the corresponding beamforming is called adaptive to distinguish it from the conventional beamforming where the beamforming weights do not depend on input/output array data.

The traditional approach to the design of adaptive beamforming is to maximize the beamformer output signal-to-interference-plus-noise ratio (SINR) assuming that there is no SOI component in the beamforming training data [2,3]. Although such SOI-free data assumption may be relevant to certain radar applications, in typical practical applications, the beamforming training snapshots also include the SOI [17,18]. In the latter case, the SINR performance of adaptive beamforming can severely degrade even in the presence of small signal steering vector errors/mismatches, because the SOI component in the beamformer training data can be mistakenly interpreted by the adaptive beamforming algorithm as an interferer and, consequently, it can be suppressed rather than being protected. The steering vector errors are, however, very common in practice and can be caused by a number of reasons such as signal look direction/pointing errors; array calibration imperfections; non-linearities in amplifiers, A/D converters, modulators and other hardware; distorted antenna shape;

\*Dedicated to the memory of Professor Alex B. Gershman.

unknown wavefront distortions/fluctuations; signal fading; near-far wavefront mismodeling; local scattering; and many other effects. The performance degradation of adaptive beamformer can also take place even when the SOI steering vector is precisely known, but the sample size (the number of samples at the training stage) is small [18]. One more reason for performance degradation is the environmental non-stationarities because of the fast variations of the propagation channel and rapid motion of interfering sources or antenna array [19]. As a result, the environment can significantly change from the beamforming training stage, at which the adaptive beamforming weights are computed, to the beamforming testing stage, at which the beamforming weights are used. This may severely limit the training sample size and increase the required frequency of beamforming weights updates. To protect against the aforementioned imperfections, the robust adaptive beamforming is considered.

This chapter is dedicated to the review of the main results in the fields of adaptive beamforming and robust adaptive beamforming. We start by introducing the array data and beamforming models for both cases on narrowband and wideband signals. Adaptive beamforming techniques are then reviewed including the basic principles of adaptive beamforming design, minimum variance distortionless response adaptive beamforming technique, analysis of optimal SINR, adaptive beamforming technique for general rank sources. The general numerical algorithms for solving the adaptive beamforming problem such as the gradient algorithm, the sample matrix inversion algorithm, and the projection adaptive beamforming algorithm are also reviewed. Finally, the reduced complexity approaches to adaptive beamforming and some techniques for wideband adaptive beamforming are explained. The motivations for robust adaptive beamforming then follow. The particular robust adaptive beamforming techniques explained in this chapter include the diagonally loaded sample matrix inversion beamforming technique, the robust adaptive beamforming techniques with point and derivative mainbeam constraints, the generalized sidelobe canceler, the adaptive beamforming techniques robust against the correlation between the SOI and interferences such as spatial and forward-backward smoothing, the adaptive beamforming techniques robust against rapidly moving interferences. A unified principle to minimum variance distortionless response robust adaptive beamforming design is given and several most popular robust adaptive beamforming techniques based on this principle are explained including the eigenspace-based beamforming technique, the worst-case-based and doubly constrained robust adaptive beamforming techniques, the probabilistically constrained robust adaptive beamforming, and the recently proposed robust adaptive beamforming that uses as little as possible prior information, and others. Robust adaptive beamforming for general-rank source model and robust adaptive wideband beamforming are also considered.

### 3.12.2 Data and beamforming models

In this chapter, the discussion is focussed on adaptive and robust adaptive beamforming and is based on the assumptions of linear antenna geometry consisting of omni-directional antenna elements. Other considerations, which are not directly related to the adaptive beamforming problem, such as non-linear multi-dimensional antenna geometries and antenna elements with directional beampattern stay outside of the scope of this chapter.

### 3.12.2.1 Narrowband case

#### 3.12.2.1.1 Point source

Consider an antenna array with  $M$  omni-directional antenna elements see also the introduction to array processing in this encyclopedia [4]. The narrowband signal received by the antenna array at the time instant  $k$  can be mathematically represented as

$$\mathbf{x}(k) = \mathbf{x}_s(k) + \mathbf{x}_i(k) + \mathbf{x}_n(k), \quad (12.1)$$

where  $\mathbf{x}_s(k)$ ,  $\mathbf{x}_i(k)$ , and  $\mathbf{x}_n(k)$  denote the  $M \times 1$  vectors of the SOI, interference, and noise, respectively. The interference signal is generated by other than SOI sources that are not of interest (interferers and possibly a jammer). For simplicity, all these components of the received signal (12.1) are assumed to be statistically independent to each other. This assumption is fairly practical since the SOI and the signals from interferers (other objects or users) are typically independent. The case of correlated/coherent SOI and interference signals, however, can occur in practice, for example, because of the scattering effect. This case will be considered separately in the chapter as well. The noise is typically isotropic or diffuse and it can be accurately modeled as spatially white Gaussian noise (i.e., the noise components are spatially uncorrelated at different antenna elements with the same noise power at each antenna element). In other words, the  $M \times M$  covariance matrix of the noise at the antenna array can be expressed as  $\mathbf{R}_n \triangleq E[\mathbf{x}_n(k)\mathbf{x}_n^H(k)] = \sigma_n^2\mathbf{I}$ , where  $\sigma_n^2$  is the noise variance/power at a single antenna element,  $\mathbf{I}$  denotes the identity matrix of the same size as the number of antenna elements in the array, and  $(\cdot)^H$  and  $E[\cdot]$  stand for the Hermitian transpose and mathematical expectation, respectively. As such, the noise is statistically independent from the SOI and interference signals.

In the case of point source, it is assumed that the SOI  $\mathbf{x}_s(k)$  arrives at the antenna array as a single plane wave and it can be mathematically represented as

$$\mathbf{x}_s(k) = s(k)\mathbf{a}(\theta_s), \quad (12.2)$$

where  $s(k)$  is the signal waveform,  $\mathbf{a}(\theta_s)$  is the  $M \times 1$  steering vector associated with the SOI, and  $\theta_s$  is the direction-of-arrival (DOA) of the SOI. Although the steering vector  $\mathbf{a}(\theta_s)$  is expressed only as a function of the DOA  $\theta_s$ , which is the source characteristic in the case of far distant point source, one should keep in mind that it is in fact also a function of array geometry as well as propagation media characteristics. The covariance matrix of the SOI for the case of point source can be, therefore, expressed in the form of the following  $M \times M$  rank-one matrix:  $\mathbf{R}_s \triangleq E[\mathbf{x}_s(k)\mathbf{x}_s^H(k)] = E[|s(k)|^2\mathbf{a}(\theta_s)\mathbf{a}^H(\theta_s)] = \sigma_s^2\mathbf{a}(\theta_s)\mathbf{a}^H(\theta_s)$ , where  $\sigma_s^2 \triangleq E[|s(k)|^2]$  is the SOI power.

The beamformer output is a weighted (with complex weights) linear combination of the signals received by different antenna elements (see also Figure 1.3 in Chapter 1 of this book [4]) at the time instant  $k$  and it can be mathematically expressed as

$$y(k) \triangleq \sum_{m=1}^M w_m^* x_m(k) = \mathbf{w}^H \mathbf{x}(k), \quad (12.3)$$

where  $w_m$  is the complex weight corresponding to the  $m$ th antenna element,  $x_m(k)$  in the signal received by the  $m$ th antenna element at the time instant  $k$ ,  $\mathbf{w} \triangleq [w_1, \dots, w_M]^T$  is the  $M \times 1$  complex weight

(beamforming) vector of the antenna array, and  $(\cdot)^T$  and  $(\cdot)^*$  denote the transpose and conjugate, respectively. The expression (12.3) is in fact a linear spatial filter. The beamforming complex weights  $\{w_m^*\}_{m=1}^M$  can be applied to the signals received by the correspondent antenna elements right at these antenna elements or at the receiver electronics. The weights  $\{w_m^*\}_{m=1}^M$  must be designed so that the SOI would be presumed/amplified at the beamformer output, the interference signals would be canceled, and the noise would be suppressed.

If only the SOI component is present, the beamformer output in the case of point source becomes  $y(k) = \mathbf{w}^H \mathbf{a}(\theta_s) s(k)$ . From the latter expression the interpretation of the beamformer in terms of a special filter becomes intuitive. Indeed,  $\mathbf{w}^H \mathbf{a}(\theta_s)$  can be thought as the spatial transfer function from  $s(k)$  at the direction  $\theta_s$  to  $y(k)$ . The magnitude  $G(\theta_s) \triangleq |\mathbf{w}^H \mathbf{a}(\theta_s)|$  is the gain of the spatial filter towards the SOI. It is similar to the finite impulse response (FIR) filtering in the temporal domain where instead of the spatial steering vector  $\mathbf{a}(\theta_s)$  we have a vector of time-delayed values of the input signal. For more details see the introduction to array processing in this encyclopedia [4].

Under the assumption that the SOI steering vector  $\mathbf{a}(\theta_s)$  is known precisely, the optimal beamforming vector  $\mathbf{w}$  can be obtained by maximizing the beamformer output signal-to-noise-plus-interference ratio (SINR) given as

$$\text{SINR} \triangleq \frac{E[|\mathbf{w}^H \mathbf{x}_s(k)|^2]}{E[|\mathbf{w}^H (\mathbf{x}_i(k) + \mathbf{x}_n(k))|^2]} = \frac{\sigma_s^2 |\mathbf{w}^H \mathbf{a}|^2}{\mathbf{w}^H \mathbf{R}_{i+n} \mathbf{w}}, \quad (12.4)$$

where  $\mathbf{R}_{i+n} \triangleq E[(\mathbf{x}_i(k) + \mathbf{x}_n(k))(\mathbf{x}_i(k) + \mathbf{x}_n(k))^H]$  is the  $M \times M$  interference-plus-noise covariance matrix.

Because of the fact that  $\mathbf{R}_{i+n}$  is unknown in practice, it is typically substituted in (12.4) by the following data sample covariance matrix

$$\widehat{\mathbf{R}} \triangleq \frac{1}{K} \sum_{k=1}^K \mathbf{x}(k) \mathbf{x}^H(k), \quad (12.5)$$

where  $K$  is the number of training data samples which also include the desired signal component. Other estimates of the data covariance matrix than (12.5) can be used [20]. It is worth mentioning here that since the noise is spatially white Gaussian and uncorrelated with the SOI and interference signals, the actual data covariance matrix can be found as

$$\mathbf{R} \triangleq E[\mathbf{x}(k) \mathbf{x}^H(k)] = \mathbf{A} \mathbf{S} \mathbf{A}^H + \sigma_n^2 \mathbf{I}, \quad (12.6)$$

where  $\mathbf{A} \triangleq [\mathbf{a}(\theta_s), \mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_L}]$  is the  $M \times (L + 1)$  matrix of steering vectors of the SOI and the interference sources under the assumption that all sources are the point sources,  $L$  is the number of interference sources,  $\mathbf{S}$  is the  $(L + 1) \times (L + 1)$  source covariance matrix. The matrix  $\mathbf{S}$  is diagonal if the SOI and all interference signals are uncorrelated.

### 3.12.2.1.2 General-rank source

Typical situations in practice, however, are when the source signal is incoherently scattered (spatially distributed) [21,22] and/or when it is characterized by fluctuating (randomly distorted) wavefronts [23,24]. Such situations are very typical, for example, for sonar and wireless communications. Particularly in sonar, effects of signal propagation through a randomly inhomogeneous underwater channel

lead to a substantial perturbation of a regular wakefield in a random way and cause its coherence loss. The result of such coherence loss is that the SOI may be subject to fast fluctuations that destroy the point source structure (12.2). In wireless communications, the common situation is the fast fading due to local scattering in the vicinity of the mobile user. Local scattering also destroys the point source structure (12.2). In such applications, the SOI can no longer be viewed by the antenna array as a point source and the source model needs to be modified. Typically, the SOI is modeled as a spatially distributed source with some central angle and angular spread. The source covariance matrix is, therefore, no longer a rank-one matrix and, for example, in the incoherently scattered source case can be given as [25]

$$\mathbf{R}_s = \int_{-\pi/2}^{\pi/2} \rho(\theta) \mathbf{a}(\theta) \mathbf{a}^H(\theta) d\theta, \quad (12.7)$$

where  $\rho(\theta)$  is the normalized angular power density (i.e.,  $\int_{-\pi/2}^{\pi/2} \rho(\theta) d\theta = 1$ ). The name “general rank source” is reflecting the fact that the covariance matrix (12.7) can have any rank from 1 in a degenerate case to  $M$ .

In the case of general-rank SOI, the SINR expression is given as

$$\text{SINR} = \frac{\mathbf{w}^H \mathbf{R}_s \mathbf{w}}{\mathbf{w}^H \mathbf{R}_{i+n} \mathbf{w}}. \quad (12.8)$$

Since the matrix  $\mathbf{R}_{i+n}$  is not known in practice it is substituted by the data sample covariance matrix (12.5) in practice.

### 3.12.2.2 Wideband case

In the wideband case, the SOI and/or the interference signals are widely spread in the frequency domain. As a result, it is not possible to factorize the processing in temporal and spatial parts. Therefore, joint space-time adaptive processing (STAP) has to be performed. The name STAP stresses on the fact that the adaptive beamforming in the wideband case is no longer a spatial filtering technique as for the narrowband case, but rather a joint spatial and temporal filtering. For more details see the chapter on broadband beamforming in this encyclopedia [26].

Let the number of taps in the time domain be denoted as  $P$ . Let also the  $M$  array sensors be uniformly spaced with the inter-element spacing less than or equal to  $c/2f_u$ , where  $f_u = f_c + B_s/2$  is the maximum frequency of the SOI/maximum passband frequency,  $f_c$  is the carrier frequency,  $B_s$  is the signal bandwidth, and  $c$  is the wave propagation speed. The general case of not necessarily uniform linear array (ULA) is considered in a specialized chapter on broadband beamforming of this encyclopedia [26]. The received signal at the  $m$ th antenna element goes to a wideband presteering delay filter with the delay  $\Delta_m$ . Let the output of the wideband presteering delay filter be sampled with the sampling frequency  $f_s = 1/\tau$ , where  $\tau$  is the sampling time and  $f_s$  is greater than or equal to  $2f_u$ . Then the  $MP \times 1$  stacked snapshot vector containing  $P$  delayed prestereed data vectors is the data vector  $\mathbf{x}(k)$ . The beamformer output  $y(k)$  is then given by [27]

$$y(k) = \mathbf{w}^H \mathbf{x}(k) = \mathbf{w}^T \mathbf{x}(k), \quad (12.9)$$

where  $\mathbf{w}$  is the real-valued  $MP \times 1$  beamformer weight vector, i.e.,  $w_{M(p-1)+m} = w_{m,p}$  and, thus,  $\mathbf{w}^H$  is equivalently substituted by  $\mathbf{w}^T$ .

In the wideband case, the steering vector also depends on frequency and in the case of a ULA is given as

$$\mathbf{a}(f, \theta) = [e^{j2\pi f z_1 \sin(\theta)/c}, \dots, e^{j2\pi f z_M \sin(\theta)/c}]^T, \quad (12.10)$$

where  $z_m$  is the  $m$ th antenna element location that for ULA is given as  $z_m = (m - 1)d$  with  $d$  denoting the inter-element spacing. The overall  $MP \times 1$  steering vector can be expressed as

$$\bar{\mathbf{a}}(f, \theta) = \mathbf{d}(f) \otimes (\mathbf{B}(f)\mathbf{a}(f, \theta)), \quad (12.11)$$

where  $\mathbf{d}(f) \triangleq [1, e^{-j2\pi f\tau}, \dots, e^{-j2\pi f(P-1)\tau}]^T$ ,  $\mathbf{B}(f) \triangleq \text{diag}\{e^{-j2\pi f\Delta_1}, \dots, e^{-j2\pi f\Delta_M}\}$ , and  $\otimes$  denotes the Kronecker product. Then the array response to a plane wave with the frequency  $f$  and angle or arrival  $\theta$  is

$$H(f, \theta) = \mathbf{w}^T \bar{\mathbf{a}}(f, \theta). \quad (12.12)$$

The presteering delays are selected so that the SOI arriving from the look direction  $\theta_0$  appears coherently at the output of the  $M$  presteering filters so that [27]

$$\mathbf{B}(f)\mathbf{a}(f, \theta_0) = \mathbf{1}_M, \quad (12.13)$$

where  $\mathbf{1}_M$  is the  $M \times 1$  vector containing all ones. Then the steering vector towards the look direction  $\theta_0$  becomes

$$\bar{\mathbf{a}}(f, \theta_0) = \mathbf{d}(f) \otimes \mathbf{1}_M \quad (12.14)$$

and the array response towards such signal becomes

$$H(f, \theta_0) = \mathbf{w}^T \bar{\mathbf{a}}(f, \theta_0) = \mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f), \quad (12.15)$$

where  $\mathbf{C}_0 \triangleq \mathbf{I}_P \otimes \mathbf{1}_M$ .

### 3.12.3 Adaptive beamforming

#### 3.12.3.1 Basic principles

The signal-to-noise ratio (SNR) gain due to coherent processing of the signal  $\mathbf{x}(k)$  received at the antenna array, i.e., due to receive beamforming, is proportional to the quantity  $|\mathbf{w}^H \mathbf{a}(\theta_s)|$  in the case of a point source. Here  $\theta_s$  is the presumed SOI DOA. Using the Cauchy-Schwarz inequality, it can be easily found that  $|\mathbf{w}^H \mathbf{a}(\theta_s)| \leq \|\mathbf{w}\| \cdot \|\mathbf{a}(\theta_s)\|$ , where equality holds when

$$\mathbf{w} = \mathbf{a}(\theta_s). \quad (12.16)$$

The expression (12.16) is referred to as the conventional nonadaptive beamforming. In the case when a single point source signal is observed in the background of white Gaussian noise, the conventional nonadaptive beamformer (12.16) is known to be optimal in the sense that it provides the highest possible output SNR gain [3]. The idealistic condition of a single point source (no interferences) is, however, impractical. Moreover, the precise estimate of the SOI steering vector  $\mathbf{a}(\theta_s)$  is required in (12.16). In the presence of interferences, (12.16) is no longer optimal and, thus, adaptive beamforming technique are of interest.

The goal of adaptive beamforming as a spatial adaptive filter is to filter out (suppress) the undesired interference and noise components  $\mathbf{x}_i(k)$  and  $\mathbf{x}_n(k)$  as much as possible, and to detect and obtain as good as possible approximation/estimation of the desired signal  $\mathbf{x}_s(k)$ , the estimate is denoted as  $\hat{\mathbf{x}}_s(k)$ . The beamforming weight vector  $\mathbf{w}$  is optimized based on the received data  $\mathbf{x}(k)$  for a number of time instants  $k = 1, \dots, K$  during the training interval. Since the adaptive beamforming problem consists of optimizing the beamforming weight vector  $\mathbf{w}$ , the optimization criterion must be defined.

One of the standard in filter design and estimation theory criteria is the mean-square error (MSE). In the context of adaptive beamforming design, the MSE criterion can be expressed as

$$\text{MSE} \triangleq E[|d(k) - \mathbf{w}^H \mathbf{x}(k)|^2], \quad (12.17)$$

where  $d(k)$  is the desired signal copy. The corresponding optimization problem is then formulated as follows:

$$\min_{\mathbf{w}} E[|d(k) - \mathbf{w}^H \mathbf{x}(k)|^2]. \quad (12.18)$$

The solution of the minimum MSE problem is well known to be the Wiener-Hopf equation, which for the optimization problem (12.18) becomes

$$\mathbf{w}_{\text{MSE}} = (E[\mathbf{x}(k)\mathbf{x}^H(k)])^{-1} E[\mathbf{x}^H(k)d(k)] = \mathbf{R}^{-1} \mathbf{r}_{xd}, \quad (12.19)$$

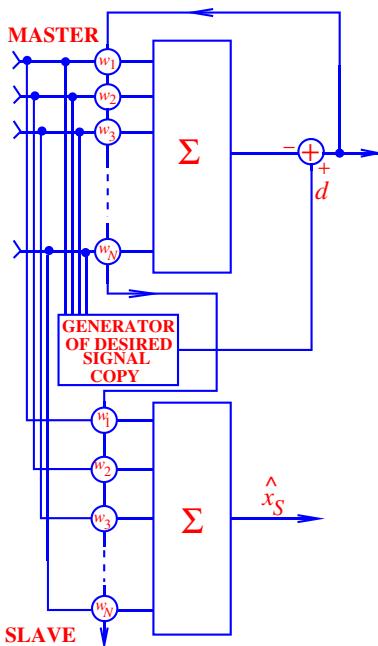
where  $\mathbf{R} \triangleq E[\mathbf{x}(k)\mathbf{x}^H(k)]$  is the data covariance matrix and  $\mathbf{r}_{xd} \triangleq E[\mathbf{x}^H(k)d(k)]$  in the correlation vector between the data vector  $\mathbf{x}$  and the reference signal  $d$ .

The block scheme of the adaptive beamformer based on MSE minimization (12.19) is shown in Figure 12.1. The adaptive beamformer consists of the “master” and “slave” beamformers. The beamforming weights are adjusted at the “master” beamformer based on minimizing the difference between the desired signal copy and the computed (using the antenna array measurements) output of the adaptive beamformer. These weights are then passed to the “slave” beamformer for computing the estimate of the desired signal  $\hat{\mathbf{x}}_s$ . The main limitation of such adaptive beamformer is the necessity to know the desired signal copy  $d(k)$ . In Figure 12.1, this necessity is reflected by introducing the generator of desired signal copy. Although the knowledge of the desired signal copy is common in adaptive filtering, in adaptive beamforming the SOI is unknown. Thus, the adaptive beamformer based on MSE minimization is impractical in most of the situations of interest.

The practically appealing criterion for adaptive beamforming design is the SINR (12.4) for the case of a point source or (12.8) for the case of a general-rank source. Obviously, the SINR does not depend on re-scaling of the beamforming vector  $\mathbf{w}$ , that is, if  $\mathbf{w}_{\text{opt}}$  is an optimal weight vector then  $\alpha \mathbf{w}_{\text{opt}}$  is another optimal weight vector as well. Here  $\alpha$  is a scaling factor. Therefore, in the case of point source, the maximization of the SINR (12.4) is equivalent to the following constrained optimization problem

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R}_{i+n} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \mathbf{a}(\theta_s) = \text{const}, \quad (12.20)$$

where “const” is any constant, for example,  $\text{const} = 1$ . The optimization problem (12.20) and its solution are known under the name of minimum variance distortionless response (MVDR) adaptive beamforming. Here the “minimum variance” stands for the fact that the objective of the optimization problem (12.20) corresponds to the variance minimization of the signal at the output of the adaptive

**FIGURE 12.1**

Adaptive beamforming based on MSE minimization.

beamformer. The term “distortionless response” refers to the constraint of the optimization problem (12.20), which requires the response of the adaptive beamformer towards the direction of the SOI steering vector  $\mathbf{a}(\theta_s)$  to be fixed and undistorted.

The optimization problem (12.20) can be solved in closed-form using the Lagrange multiplier method. Specifically, the Lagrangian for the problem (12.20) is given as

$$L(\mathbf{w}, \lambda) = \mathbf{w}^H \mathbf{R}_{i+n} \mathbf{w} + \lambda(1 - \mathbf{w}^H \mathbf{a}(\theta_s)), \quad (12.21)$$

where  $\lambda$  is a Lagrange multiplier. The solution of (12.20) is then obtained by finding the gradient of the Lagrangian (12.21), equating it to zero, and solving the so-obtained equation. This equation is

$$\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = \mathbf{R}_{i+n} \mathbf{w} - \lambda \mathbf{a}(\theta_s) = 0 \quad (12.22)$$

and it can be rewritten equivalently as

$$\mathbf{R}_{i+n} \mathbf{w} = \lambda \mathbf{a}(\theta_s). \quad (12.23)$$

Then, the solution of (12.23) can be easily found as

$$\mathbf{w}_{\text{opt}} = \lambda \mathbf{R}_{i+n}^{-1} \mathbf{a}(\theta_s). \quad (12.24)$$

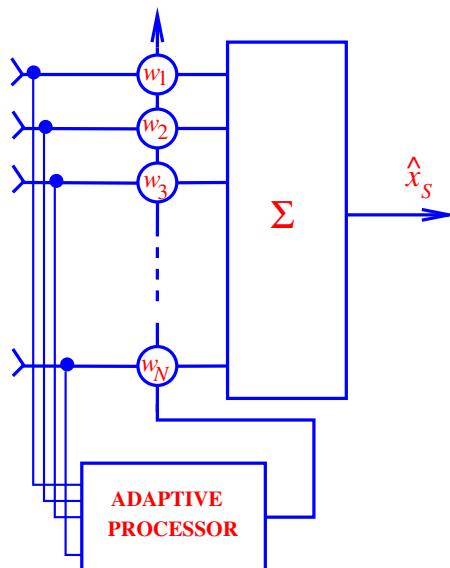
This is a *spatial version* of the Wiener-Hopf equation. Compared to (12.19), there is the SOI spatial signature/steering vector  $\mathbf{a}(\theta_s)$  in (12.24) instead of the correlation vector  $\mathbf{r}_{xd}$ . Moreover, there is the interference-plus-noise covariance matrix  $\mathbf{R}_{i+n}$  instead of the data covariance matrix  $\mathbf{R}$ . The Lagrange multiplier  $\lambda$  can be easily found by substituting (12.24) in the distortionless response constraint of the original optimization problem (12.20) and solving the corresponding equation for  $\lambda$ . The result is

$$\lambda = \frac{1}{\mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)}. \quad (12.25)$$

Finally, substituting (12.25) in (12.24), the closed-form expression for the MVDR beamforming can be obtained in the following form:

$$\mathbf{w}_{MVDR} = \frac{1}{\mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)}\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s). \quad (12.26)$$

The block scheme of the adaptive beamformer based on SINR maximization is shown in Figure 12.2. According to this block scheme, the beamforming weights are computed at the adaptive processor, which implements the estimation of the covariance matrix  $\mathbf{R}_{i+n}$  and then computes the beamforming weight vector according to (12.26). The input data for the adaptive processor are the antenna array measurements  $\mathbf{x}(k)$ , while the output, which is passed to the antenna elements, is the vector of optimal beamforming weights  $\mathbf{w}$ . If the received signal is free of the desired signal component, the sample estimate of the



**FIGURE 12.2**

Adaptive beamforming based on SINR maximization.

covariance matrix  $\mathbf{R}_{i+n}$  can be obtained based on the expression (12.5). Otherwise, only the sample estimate of the data covariance matrix  $\widehat{\mathbf{R}}$  can be found by using (12.5). The latter case when the signal of interest is present in the data vector  $\mathbf{x}$  is, however, common in practice.

### 3.12.3.2 MVDR beamforming with data covariance matrix

Even if the SOI is present in the data vector  $\mathbf{x}(k)$ , but the estimate of the data covariance matrix is perfect and the steering vector of the SOI  $\mathbf{a}(\theta_s)$  is known precisely, the resulting beamformer that uses the data covariance matrix instead of the interference-plus-noise covariance matrix is equivalent to the MVDR beamformer of (12.26). Indeed, the data covariance matrix in the case of point source can be represented by explicitly using the interference-plus-noise covariance matrix as

$$\mathbf{R} \triangleq \mathbb{E}[\mathbf{x}(k)\mathbf{x}^H(k)] = \sigma_s^2 \mathbf{a}(\theta_s) \mathbf{a}^H(\theta_s) + \mathbf{R}_{i+n}. \quad (12.27)$$

Ignoring the immaterial for the SINR at the output of the adaptive beamformer coefficient  $1/\mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)$  in (12.26), using the data covariance matrix (12.27) instead of the interference-plus-noise covariance matrix, and applying consequently the matrix inversion lemma, it can be shown that

$$\begin{aligned} \mathbf{R}^{-1}\mathbf{a}(\theta_s) &= \left( \mathbf{R}_{i+n} + \sigma_s^2 \mathbf{a}(\theta_s) \mathbf{a}^H(\theta_s) \right)^{-1} \mathbf{a}(\theta_s) = \left( \mathbf{R}_{i+n}^{-1} - \frac{\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s) \mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1}}{1/\sigma_s^2 + \mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)} \right) \mathbf{a}(\theta_s) \\ &= \mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s) - \frac{\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s) \mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)}{1/\sigma_s^2 + \mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)} = \left( 1 - \frac{\mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)}{1/\sigma_s^2 + \mathbf{a}^H(\theta_s)\mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s)} \right) \mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s) \\ &= \alpha \mathbf{R}_{i+n}^{-1}\mathbf{a}(\theta_s), \end{aligned} \quad (12.28)$$

where the coefficient  $\alpha \triangleq 1 / \left( 1 + \sigma_s^2 \mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1} \mathbf{a}(\theta_s) \right)$  is immaterial for the output SINR of the adaptive beamformer.

### 3.12.3.3 Optimal SINR

The optimal output SINR is the maximum SINR obtained by substituting the optimal MVDR beamforming vector (12.26) in the SINR expression (12.4). Specifically, the optimal SINR in the case of a point source is given by

$$\text{SINR}_{\text{opt}} = \frac{\sigma_s^2 \left( \mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1} \mathbf{a}(\theta_s) \right)^2}{\mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1} \mathbf{R}_{i+n} \mathbf{R}_{i+n}^{-1} \mathbf{a}(\theta_s)} = \sigma_s^2 \mathbf{a}^H(\theta_s) \mathbf{R}_{i+n}^{-1} \mathbf{a}(\theta_s). \quad (12.29)$$

The expression (12.29) is in fact an upper bound for the output SINR, obtained for the case of no interference.

For rough estimation of the optimal SINR in the case when there are only a few uncorrelated interferences and the signal is well separated from them, the interference-plus-noise covariance matrix can be approximated by a scaled identity matrix with a scaling coefficient representing the aggregate

power of the interferences and noise denoted as  $\sigma^2$ . Then the upper bound for the optimal SINR (12.29) is

$$\text{SINR}_{\text{opt}} \simeq \frac{\sigma_s^2}{\sigma^2} \mathbf{a}^H(\theta_s) \mathbf{a}(\theta_s) = M \frac{\sigma_s^2}{\sigma^2}, \quad (12.30)$$

where for obtaining the last equality, the fact that the squared norm of the steering vector equals to the number of sensors in the antenna array, i.e.,  $\|\mathbf{a}(\theta_s)\|^2 = M$ , has been used. Thus, roughly, the optimal SINR is upper bounded by the product of the input SINRs at the individual antenna elements and the total number of antenna elements in the antenna array.

### 3.12.3.4 Adaptive beamforming for general-rank source

In the case of general-rank source, the SINR expression (12.8) is the one that has to be used. The corresponding MVDR-type optimization problem can be then formulated as

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R}_{i+n} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \mathbf{R}_s \mathbf{w} = 1. \quad (12.31)$$

The solution of the optimization problem (12.31) is well known to be the principal eigenvector of the matrix product  $\mathbf{R}_{i+n}^{-1} \mathbf{R}_s$ , that is mathematically expressed as

$$\mathbf{w}_{\text{opt}} = \mathcal{P}[\mathbf{R}_{i+n}^{-1} \mathbf{R}_s], \quad (12.32)$$

where  $\mathcal{P}[\cdot]$  denotes the operator that computes the principal eigenvector of a matrix. The solution (12.32) is of a limited practical use because in most applications, the matrix  $\mathbf{R}_s$  is unknown, and often no reasonable estimate of it is available. However, if the estimate of  $\mathbf{R}_s$  is available as well as the estimate of  $\mathbf{R}_{i+n}$ , (12.32) provides a simple solution to the adaptive beamforming problem for the general-rank source. The solution of (12.32) can be equivalently found as the solution of the characteristic equation for the matrix  $\mathbf{R}_s^{-1} \mathbf{R}_{i+n}$ , that is,  $\mathbf{R}_s^{-1} \mathbf{R}_{i+n} \mathbf{w} = \lambda \mathbf{w}$ , if the matrix  $\mathbf{R}_s$  is full-rank invertible. In practice, however, the rank of the desired source can be smaller than the number of sensors in the antenna array and the source covariance matrix  $\mathbf{R}_s$  may not be invertible, while the matrix  $\mathbf{R}_{i+n}$  is guaranteed to be invertible due to the presence of the noise component. Therefore, the solution (12.32) is always preferred practically.

### 3.12.3.5 Gradient adaptive beamforming algorithms

The interference-plus-noise and data covariance matrices are unknown in practice. Assuming that there is a finite number of training snapshots  $\mathbf{x}(k)$  that do not contain the SOI component and that the SOI steering vector  $\mathbf{a}(\theta_s)$  is known precisely, the historically first adaptive beamforming method is the gradient algorithm developed back in the 1960s of the last century [28]. Similar to the least-mean square (LMS) adaptive filtering, the gradient adaptive beamforming algorithm can be mathematically expressed as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu (\mathbf{a}(\theta_s) - \mathbf{x}(k) \mathbf{x}^H(k) \mathbf{w}(k)), \quad (12.33)$$

where  $\mathbf{w}(k)$  stands for the beamforming weight vector at the  $k$ th iteration, i.e., after processing the  $k$ th data snapshot, and  $\mu$  is the step size of the LMS algorithm. The convergence condition for the gradient

adaptive beamforming algorithm is similar to that of the LMS convergence condition and is formulated as follows. The beamforming vector  $\mathbf{w}(k)$  converges to the MVDR beamforming solution (12.26) if

$$0 < \mu < \frac{2}{\lambda_{\max}[\mathbf{R}_{i+n}]}, \quad (12.34)$$

where  $\lambda_{\max}[\cdot]$  denotes the maximum eigenvalue of a square matrix. Finding the maximum eigenvalue required in (12.34) is computationally complex. Hence, using the property that the maximum eigenvalue of a positive semi-definite square matrix is smaller or equal to the trace of such matrix, (12.34) can be simplified as

$$0 < \mu < \frac{2}{\text{Tr}(\mathbf{R}_{i+n})}, \quad (12.35)$$

where  $\text{Tr}(\cdot)$  stands for the trace of a square matrix.

The covariance matrix  $\mathbf{R}_{i+n}$  is, however, not known in practice. Thus, the choice of the step size  $\mu$  that guarantees the convergence of the algorithm (12.33) is a nontrivial practical issue. Another main disadvantage of the gradient adaptive beamforming algorithm is that the convergence depends on eigenvalue spread of the matrix  $\mathbf{R}_{i+n}$  and may be very slow. To demonstrate it, the following simulation example is considered.

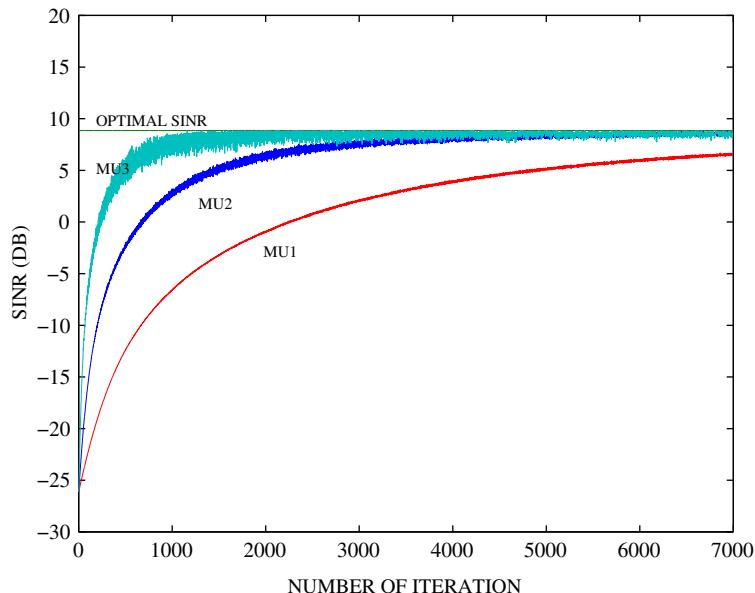
A ULA consists of  $M = 8$  omni-directional sensors spaced half-wavelength apart from each other. A single SOI impinges on the antenna array form the direction  $\theta_s = 0^\circ$  with SNR = 0 dB, while a single interference impinges on the antenna array form the direction  $\theta_i = 30^\circ$  with interference-to-noise ratio (INR) = 40 dB. The gradient adaptive beamforming algorithm (12.33) is tested for three different values of the step size:  $\mu_1 = 1/50 \text{ Tr}(\mathbf{R}_{i+n})$ ,  $\mu_2 = 1/15 \text{ Tr}(\mathbf{R}_{i+n})$ , and  $\mu_3 = 1/5 \text{ Tr}(\mathbf{R}_{i+n})$ . The results are shown in Figure 12.3 which demonstrates the convergence of (12.33) for different values of  $\mu$  in terms of the output SINR in (dB) versus the number of snapshots, i.e., the number of algorithm iterations. The optimal SINR (12.29) that provides an absolute upper bound for the output SINR of an adaptive beamformer is also shown. It can be seen from Figure 12.3 that the convergence is faster for larger  $\mu$ , but the variance of the output SINR values distribution is significantly higher compared to the case of small  $\mu$ . Moreover, even in the case of fastest convergence, the number of iterations required for convergence, i.e., the required number of training snapshots is well above 1000 which is too large number in most practical applications. As an extreme example, in radar field only a single snapshot may be available.

### 3.12.3.6 Sample matrix inversion adaptive beamformer

The sample matrix inversion (SMI) adaptive beamformer [29] is obtained by replacing the interference-plus-noise covariance matrix  $\mathbf{R}_{i+n}$  in the MVDR beamformer (12.26) with the sample estimate of the data covariance matrix (12.5). Then the expression for the corresponding beamformer is given as

$$\mathbf{w}_{\text{SMI}} = \widehat{\mathbf{R}}^{-1} \mathbf{a}(\theta_s). \quad (12.36)$$

Under the assumption shared by all traditional adaptive beamforming techniques that the SOI component is not present in the training data, the requirement of the SMI beamformer on the number of training snapshots is given by the so-called *Reed-Mallett-Brennan (RMB) rule* [29]. It states that the

**FIGURE 12.3**

SINR versus the number of training snapshots (number of iterations) for the gradient adaptive beamforming algorithm with different choices of the algorithm step size.

mean losses (relative to the optimal SINR) due to the SMI approximation of  $\mathbf{w}_{\text{MVDR}}$  (12.26) do not exceed 3 dB if

$$K \geq 2M. \quad (12.37)$$

Hence, the SMI beamformer has in general fast convergence rate that is much faster than that of the gradient adaptive beamforming algorithm.

### 3.12.3.7 Projection adaptive beamforming methods

Although the RMB rule for the SMI beamformer provides a significantly faster convergence rate compared to the gradient adaptive beamforming algorithm, the number of required training snapshots may be still quite significant especially for large arrays. The so-called Hung-Turner or projection adaptive beamformer allows to reduce the number of training snapshots even further [30].

Under the standard for traditional adaptive beamforming techniques assumption that the SOI component is not present in the training data and also under the assumption that the noise power is negligible, the inverse of the data covariance matrix  $\mathbf{R}^{-1}$  can be closely approximated by the orthogonal projection matrix  $\mathbf{P}_A^\perp \triangleq \mathbf{I} - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  where the matrix  $\mathbf{A}$  in the absence of the SOI becomes  $\mathbf{A} \triangleq [\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_L}]$ , i.e., it only consists of  $L$  interference steering vectors. The interference steering vectors are unknown in practice and, thus,  $\mathbf{P}_A^\perp$  is also unknown. However, under the aforementioned assumptions of no SOI and negligible noise power,  $\mathbf{P}_A^\perp$  can be closely approximated by the data-orthogonal

projection matrix  $\mathbf{P}_{\mathbf{X}}^{\perp} \triangleq \mathbf{I} - \mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H$ , where  $\mathbf{X}$  is the matrix of available training snapshots. Thus, the following train of approximate equalities holds:

$$\mathbf{R}_{i+n}^{-1} \simeq \mathbf{P}_A^{\perp} \simeq \mathbf{P}_{\mathbf{X}}^{\perp}. \quad (12.38)$$

Replacing  $\mathbf{R}_{i+n}^{-1}$  in (12.26) by the data-orthogonal projection matrix as in (12.38), the Hung-Turner adaptive beamforming algorithm can be written as

$$\mathbf{w}_{HT} = (\mathbf{I} - \mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H) \mathbf{a}(\theta_s). \quad (12.39)$$

For this method, a satisfactory performance can be achieved with [30]

$$K \geq L. \quad (12.40)$$

The optimal value of  $K$  is [30]

$$K_{opt} = \sqrt{(M+1)L} - 1 \quad (12.41)$$

which may be significantly smaller than the value given by the RMB rule for the SMI beamformer especially for large antenna arrays and for the scenarios with small number of interferences. The drawback of the projection adaptive beamformer is, however, that the number of interference sources should be known a priori.

### 3.12.3.8 Reduced complexity approaches to adaptive beamforming

The Hung-Turner adaptive beamforming algorithm (12.39) is especially efficient when the number of sensors in the array is much larger than the number of interferences. However, in some applications the number of sensors in the array, or equivalently, the number of adaptive degrees of freedom (adaptive beamforming weights) is so large that the computational complexity of the beamformer (12.39) becomes high. For example, the over-the-horizon radar may consists of hundreds and thousands of antenna elements [31], while the number of interferences may be relatively few. In such cases, *partially adaptive arrays* can be used to reduce the amount of computations [3].

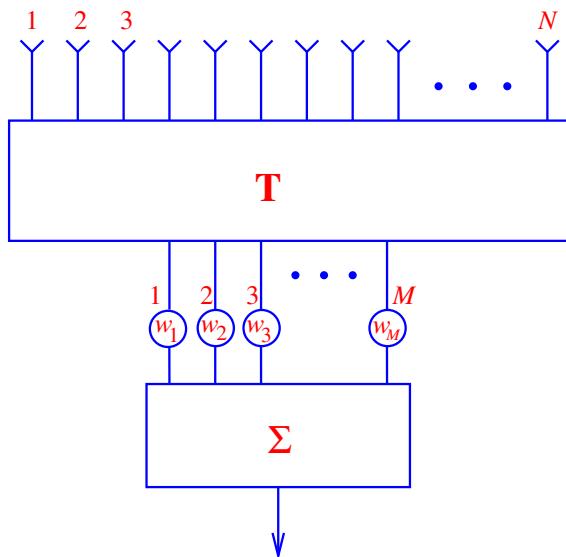
The idea of partially adaptive array is to use nonadaptive (data-independent) preprocessor to reduce the number of adaptive channels. Mathematically, such nonadaptive preprocessor can be expressed as

$$\mathbf{y}(k) = \mathbf{T}^H \mathbf{x}(k), \quad (12.42)$$

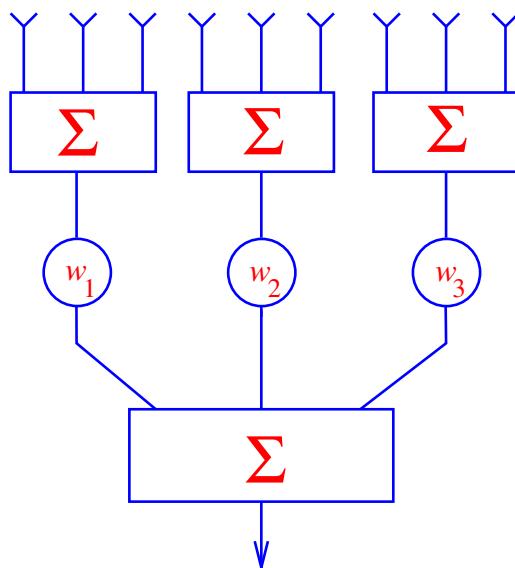
where  $\mathbf{T}$  is an  $M \times N$  ( $N < M$ ) fixed preprocessing full-rank matrix and  $\mathbf{y}(k)$  has a reduced dimension of  $N \times 1$  relative to  $M \times 1$  for the original data vector  $\mathbf{x}(k)$ . The block scheme of the partially adaptive beamformer is shown in Figure 12.4 where the  $M$  measurements of the antenna array are first preprocessed by multiplying the vector  $\mathbf{x}(k)$  to the preprocessing matrix  $\mathbf{T}$ . Then the adaptive beamformer is applied to the preprocessed vector  $\mathbf{y}(k)$ .

There are two type of preprocessors: subarray preprocessing and beamspace preprocessing. An example of partially adaptive beamformer with subarray preprocessor is shown in Figure 12.5. In this example, the matrix  $\mathbf{T}$  takes a form of

$$\mathbf{T}^T = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}. \quad (12.43)$$

**FIGURE 12.4**

Block scheme of the partially adaptive beamformer.

**FIGURE 12.5**

An example of partially adaptive beamformer based on subarray preprocessing.

It can be easily seen that  $\mathbf{T}^T \mathbf{T} = \mathbf{I}$  for (12.43). It is a desired property since the preprocessing may lead to colored noise if  $\mathbf{T}^T \mathbf{T} \neq \mathbf{I}$ . However, the noise remains spatially white if  $\mathbf{T}^T \mathbf{T} = \mathbf{I}$ .

The preprocessing of type (12.43) or a general preprocessing that follows (12.42) changes the array manifold. We say that the element-space of the antenna array is transformed into the beam-space of a smaller dimension to stress on the fact that the resulting array manifold is changed and, thus, the new SOI steering vector is

$$\tilde{\mathbf{a}}(\theta_s) = \mathbf{T}^H \mathbf{a}(\theta_s). \quad (12.44)$$

The relationship between the element-space and beam-space is also shown in Figure 12.6 for a certain partially adaptive beamformer based on subarray preprocessing. For an arbitrary preprocessor, the covariance matrix of the preprocessed data  $\mathbf{y}(k)$  can be expressed as

$$\mathbf{R}_y \triangleq E[\mathbf{y}(k)\mathbf{y}^H(k)] = \mathbf{T}^H E[\mathbf{x}(k)\mathbf{x}^H(k)]\mathbf{T} = \mathbf{T}^H \mathbf{R} \mathbf{T}. \quad (12.45)$$

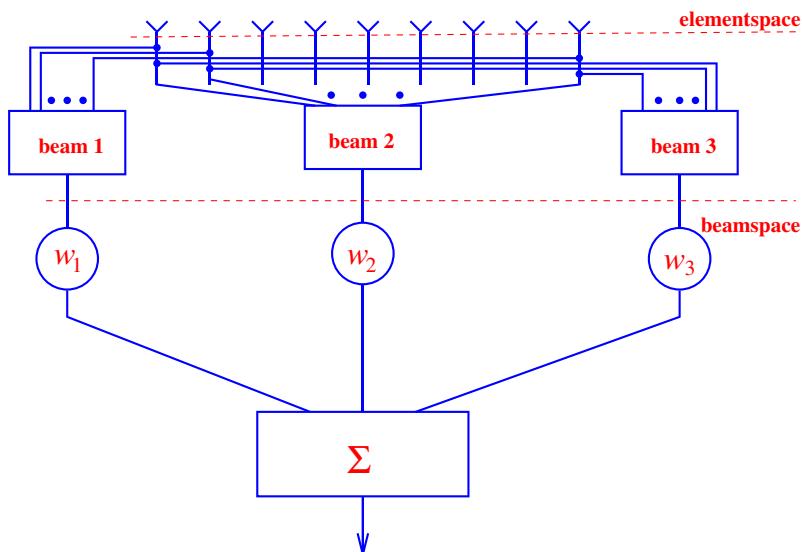
Substituting the expression (12.6) for the actual data covariance matrix in (12.45), we obtain

$$\mathbf{R}_y = \mathbf{T}^H \mathbf{A} \mathbf{S} \mathbf{A}^H \mathbf{T} + \sigma_n^2 \mathbf{T}^H \mathbf{T} = \tilde{\mathbf{A}} \tilde{\mathbf{S}} \tilde{\mathbf{A}}^H + \mathbf{Q}, \quad (12.46)$$

where

$$\tilde{\mathbf{A}} \triangleq \mathbf{T}^H \mathbf{A}, \quad (12.47)$$

$$\mathbf{Q} \triangleq \sigma_n^2 \mathbf{T}^H \mathbf{T}, \quad (12.48)$$



**FIGURE 12.6**

Element-space and beam-space of a partially adaptive beamformer based on subarray preprocessing.

and the noise covariance matrix for the preprocessed data  $\mathbf{Q}$  may not be a scaled identity matrix in general. Thus, while designing the preprocessing matrix the condition

$$\mathbf{T}^H \mathbf{T} = \mathbf{I} \quad (12.49)$$

has to be ensured.

Existing designs for the preprocessing matrix  $\mathbf{T}$  that satisfy the condition (12.49) are the discrete Fourier transform (DFT)-based beamspace preprocessing technique and the spheroidal sequences technique [3, 32, 33]. Both techniques consider an angular sector  $[\theta_{\min}, \theta_{\max}]$  where the SOI is likely to be located, i.e.,  $\theta_s \in [\theta_{\min}, \theta_{\max}]$ , and attempt to design a set of vectors that are orthonormal in this sector. Such orthonormal vectors form the preprocessing matrix  $\mathbf{T}$  and guarantee that the property (12.49) is satisfied.

The DFT-based beamspace preprocessing matrix is expressed as

$$\mathbf{T} = [\mathbf{a}(\theta_{\min}), \mathbf{a}(\theta_{\min} + \Delta\theta), \dots, \mathbf{a}(\theta_{\max} - \Delta\theta), \mathbf{a}(\theta_{\max})], \quad (12.50)$$

where all vectors are DFT orthonormal vectors covering the angular sector  $[\theta_{\min}, \theta_{\max}]$  with an angular sampling interval  $\Delta\theta$ .

The essence of the spheroidal sequence technique [33] to the design of the preprocessing matrix  $\mathbf{T}$  (beamspace transformation) [32] is to take the principal eigenvectors of the matrix

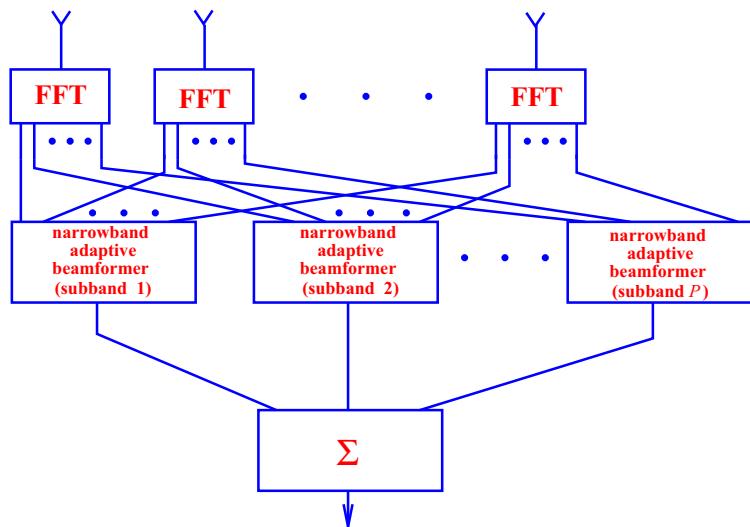
$$\int_{\theta_{\min}}^{\theta_{\max}} \mathbf{a}(\theta) \mathbf{a}^H(\theta) d\theta \quad (12.51)$$

as columns of  $\mathbf{T}$ . Since these columns are the eigenvectors, they will be orthonormal as desired.

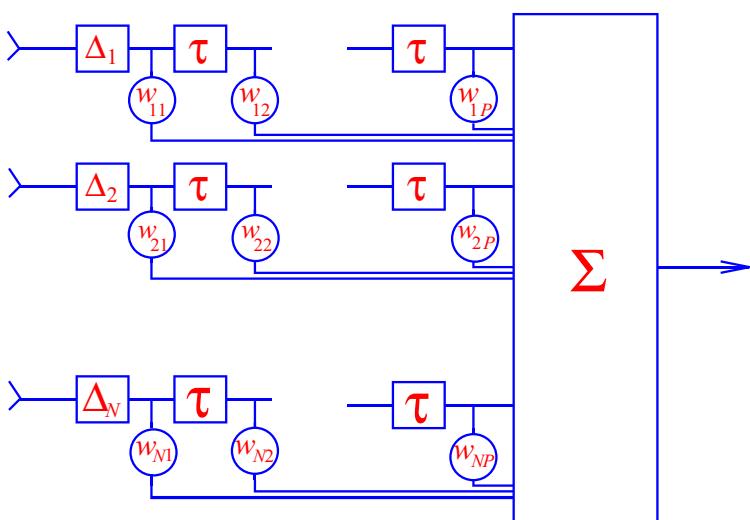
### 3.12.3.9 Wideband adaptive beamforming

One popular approach to wideband beamforming is to decompose the baseband waveforms into narrowband frequency components by the means of fast Fourier transform (FFT) [34, 35]. Subsequently, the subbands can be processed independently from each other using narrowband beamforming techniques as it is shown in Figure 12.7. Then any of the above discussed adaptive beamforming methods can be used to solve each narrowband beamforming problem. Thus,  $P$  adaptive beamforming problems, each for the beamforming vector of length  $M$ , are needed to be solved. The time-domain beamformer output samples are obtained by applying an inverse FFT (IFFT) of the output samples of the individual narrowband beamformers. However, such FFT-based wideband beamforming technique is not optimal, since correlations between the frequency domain snapshot vectors of different subbands are not taken into account. Although these correlations can be reduced by increasing the FFT length, the latter requires a larger training data set [34].

Based on the wideband data and beamforming models introduced in Section 3.12.2.2, another approach to wideband beamforming that does not require subband decomposition has been developed [27]. The block scheme of such adaptive beamformer is shown in Figure 12.8. As explained in Section 3.12.2.2, this beamformer uses a presteering delay front-end consisting of presteering delay filters to time-align the desired signal components in different sensors. Then the presteering delays are

**FIGURE 12.7**

Subband processing scheme for wideband adaptive beamforming.

**FIGURE 12.8**

Block scheme of the presteered wideband adaptive beamformer.

followed by FIR filters, each of length  $P$ . The beamformer output is then the sum of the filtered waveforms. The weights of such spatial-temporal filter for the wideband MVDR beamformer are designed to minimize the output power subject to the distortionless response constraint for the SOI. Multiple mainbeam constraints are required to protect the SOI in the frequency band of interest. The distortionless response constraint is formulated for the steering vector (12.14) after the SOI components in different sensors are made identical at the presteering stage. Then the narrowband adaptive beamforming algorithms introduced in this section can be extended relatively straightforwardly for the STAP shown in Figure 12.8. Moreover, the so-called generalized sidelobe canceler-type of techniques that will be explained in Section 3.12.4.4 can be straightforwardly used [27]. For more details and designs for wideband adaptive beamforming see also the specialized chapter on broadband beamforming in this encyclopedia [26].

## 3.12.4 Robust adaptive beamforming

### 3.12.4.1 Motivations

The result (12.28) on the equivalence between the MVDR adaptive beamformer with the interference-to-noise covariance matrix and the one with the data covariance matrix holds true only under the conditions that

- there is infinite number of snapshots available at the training stage and the data covariance matrix can be estimated exactly or at least with high accuracy,
- the SOI steering vector  $\mathbf{a}(\theta_s)$  is known precisely.

However, these conditions are not satisfied in practice since the data covariance matrix  $\mathbf{R}$  cannot be known exactly and its estimate  $\widehat{\mathbf{R}}$  typically contains the SOI component where the desired signal steering vector  $\mathbf{a}(\theta_s)$  may be known imprecisely. The applications where the SOI component is always present in the training data include mobile communications, passive source location, microphone array speech processing, medical imaging, radio astronomy, etc. The inaccuracies in the knowledge of the SOI steering vector may appear for multiple reasons associated with imperfect knowledge of the source characteristics, propagation media or antenna array itself. For example, even small look direction/signal pointing errors can lead to significant degradation of the adaptive beamformer performance [36,37]. Similarly, an imperfect array calibration and distorted antenna shape can also lead to significant degradations [38]. Other common causes of the adaptive beamformer's performance degradation are the array manifold mismodeling due to source waveform distortions resulting from environmental inhomogeneities [39], nea-far problem [40], source spreading and local scattering [41–43], and so on.

All the aforementioned issues are addressed in the field of robust adaptive beamforming. One of the earlier excellent reviews of the field is [44]. However, many new techniques and approaches have been developed since this review. This section aims at revising the most significant robust adaptive beamforming techniques.

### 3.12.4.2 Diagonally loaded SMI beamformer

Even in the ideal case when the SOI steering vector is precisely known, the SOI presence in the training data may dramatically reduce the convergence rates of adaptive beamforming algorithms as compared with the SOI-free training data case [18]. This may cause a much more substantial degradation of the performance of adaptive beamforming techniques in situations of small training sample size compared to the prediction given, for example, by the RMB rule (12.37) for the SMI adaptive beamformer (12.36).

By adding a regularization term in the objective function of the optimization problem (12.20) that penalizes the imperfections in the data covariance matrix estimate due to small sample size and other effects, the problem (12.20) can be reformulated as

$$\min_{\mathbf{w}} \mathbf{w}^H \widehat{\mathbf{R}} \mathbf{w} + \gamma \|\mathbf{w}\|^2 \quad \text{subject to} \quad \mathbf{w}^H \mathbf{a}(\theta_s) = 1, \quad (12.52)$$

where  $\gamma$  is some penalty parameter. The solution to the problem (12.52) is given by the well known diagonally loaded or shortly just loaded SMI (LSMI) beamformer [17,45,46]

$$\mathbf{w}_{\text{LSMI}} = \widehat{\mathbf{R}}_{\text{DL}}^{-1} \mathbf{a}(\theta_s), \quad \widehat{\mathbf{R}}_{\text{DL}} \triangleq \widehat{\mathbf{R}} + \gamma \mathbf{I}, \quad (12.53)$$

where the empirically-optimal penalty weight  $\gamma$  equals to double the noise power [17]. LSMI beamformer allows to converge faster than in  $2M$  snapshots suggested by the RMB rule.

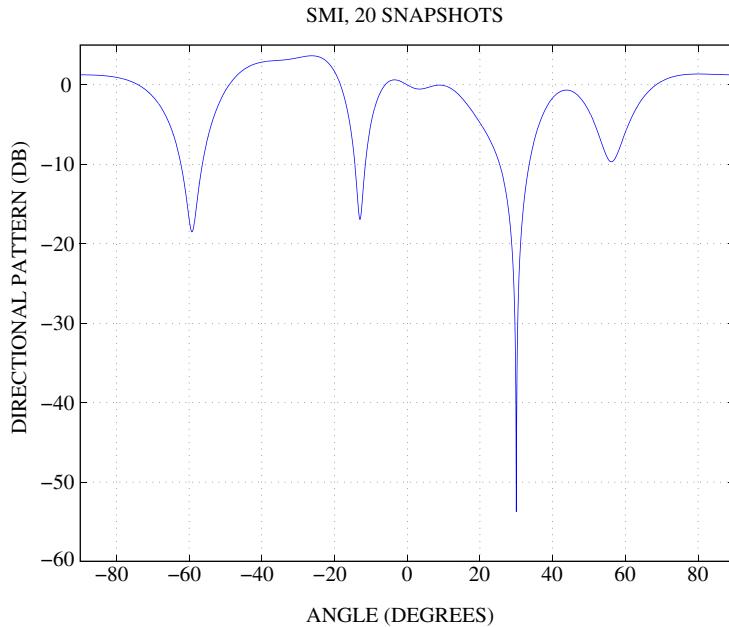
*LSMI convergence rule:* the mean losses (relative to the optimal SINR) due to the LSMI approximation of  $\mathbf{w}_{\text{MVDR}}$  in (12.26) do not exceed a few dB's if

$$K \geq L. \quad (12.54)$$

Interestingly, for properly selected  $\gamma$ , the LSMI beamformer is also efficient in the case when the desired signal steering vector is mismatched. This fact will be explained in details later. However, the choice of  $\gamma$  is not a trivial problem for the LSMI beamformer. Another important observation is that the convergence rule for the LSMI beamformer coincides with that of the Hung-Turner beamformer. Thus, the Hung-Turner beamformer can also be classified as robust against small sample size.

To demonstrate the efficiency of the LSMI beamformer compared to the SMI beamformer, the following simulation example is considered. A ULA consists of 10 omni-directional sensors spaced half wavelength apart from each other. The DOA of a single SOI is  $\theta_s = 0^\circ$  and SNR = 0 dB, while the DOA of a single interference is  $\theta_i = 30^\circ$  and INR = 40 dB. Figures 12.9 and 12.10 show the beampatterns of the SMI and LSMI beamformers, respectively. The number of training snapshots for the SMI beamformer equals to  $K = 20$  that satisfies the RMB rule (12.37), while the number of training snapshots for the LSMI beamformer equals only  $K = 2$  that satisfies the LSMI convergence rule (12.54). It can be seen from the figures that despite the fact that the number of training snapshots for the LSMI beamformer is 10 times smaller than that for the SMI beamformer, the beampattern corresponding to the LSMI beamformer has a significantly higher mainlobe and lower sidelobes. The parameter  $\gamma$  for the LSMI beamformer has been selected as double the noise power.

In addition, Figure 12.11 demonstrates the convergence rate for the SMI beamformer for two cases when the SOI component is not present in the training snapshots and when it is present. The same simulation set up as above has been used. It can be seen from this figure that the presence of the SOI component in the training snapshots significantly slows down the convergence of the SMI beamformer. The same conclusion is true for the LSMI beamformer with fixed diagonal loading factor  $\gamma$  that is selected as double the noise power.



**FIGURE 12.9**

Beampattern of the SMI beamformer for the number of snapshots  $K = 2M = 20$  that satisfies the RMB rule (12.37).

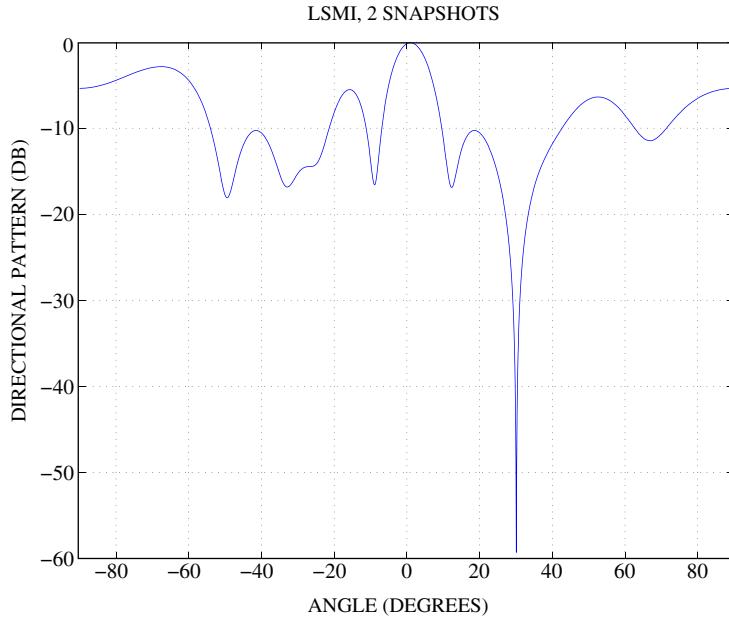
### 3.12.4.3 Look direction mismatch (pointing error) problem

Although the mismatch in the desired signal steering vector can be caused by a number of reasons, the look direction mismatch (pointing error) has been considered historically first. Even a very slight look direction mismatch can lead to the effect that is known as the signal cancellation phenomenon. This phenomenon is schematically demonstrated in Figure 12.12 where the presumed DOA of the SOI differs from the real DOA by few degrees. The adaptive beamformer misinterprets the desired signal with an interference and puts the null in the direction of the SOI. The signal cancellation phenomenon may cause a performance breakdown for adaptive beamformer and, thus, robust adaptive beamforming techniques become vital.

To stabilize the mainbeam response of adaptive beamformer in the case of pointing error, additional constraints are required. If all additional constraints are of the same type as the distortionless response constraint, i.e., linear constraints, the optimization problem can be reformulated as

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to} \quad \mathbf{C}^H \mathbf{w} = \mathbf{f}, \quad (12.55)$$

where  $\mathbf{C}$  and  $\mathbf{f}$  are some  $Q \times M$  and  $Q \times 1$  matrix and vector, respectively. Depending on the choice of  $\mathbf{C}$  or  $\mathbf{f}$ , we may have point or derivative mainbeam constraints [27, 47].

**FIGURE 12.10**

Beampattern of the LSMI beamformer for the number of snapshots  $K = 2L = 2$  that satisfies the LSMI convergence rule (12.54).

*Point mainbeam constraints:* In this case, the matrix of constrained directions is given as

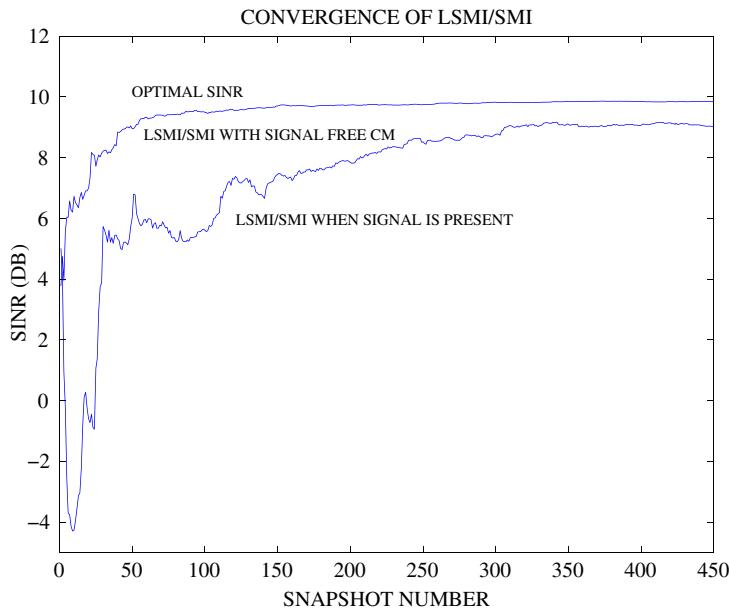
$$\mathbf{C} = [\mathbf{a}(\theta_{s,1}), \mathbf{a}(\theta_{s,2}), \dots, \mathbf{a}(\theta_{s,Q})], \quad (12.56)$$

where  $\mathbf{a}(\theta_{s,q})$ ,  $q = 1, \dots, Q$  are all taken in the neighborhood of the steering vector in the presumed direction  $\mathbf{a}(\theta_s)$  and include the steering vector in the presumed direction as well. Then the vector of constraints  $\mathbf{f}$  is

$$\mathbf{f} = [1, 1, \dots, 1]^T. \quad (12.57)$$

The constraint in the optimization problem (12.55) consists of multiple point constraints similar to the distortionless response constraints, but covers not only the presumed direction, but also the directions in the neighborhood of the presumed direction. The work principle of the point mainlobe constraint is demonstrated in Figure 12.13.

The disadvantage of using multiple distortionless response constraints is that additional degrees of freedom are used by the beamformer in order to satisfy these constraints. Since for an antenna array of  $M$  sensors, the number of degrees of freedom is  $M$ , the use of each additional degree of freedom for satisfying additional distortionless response constraints limits the remaining degrees of freedom that may be needed for suppressing interference signals.



**FIGURE 12.11**

SINR versus the number of training snapshots for the SMI beamformer in the SOI-free scenario and in the case when SOI is present in the training data.

*Derivative mainbeam constraints:* In this case, the matrix of constrained directions is given as

$$\mathbf{C} = \left[ \mathbf{a}(\theta_s), \frac{\partial \mathbf{a}(\theta)}{\partial \theta} \Big|_{\theta=\theta_s}, \dots, \frac{\partial^{M-1} \mathbf{a}(\theta)}{\partial \theta^{M-1}} \Big|_{\theta=\theta_s} \right] \quad (12.58)$$

and the vector of constraints is

$$\mathbf{f} = [1, 0, \dots, 0]^T. \quad (12.59)$$

Here

$$\frac{\partial^k \mathbf{a}(\theta)}{\partial \theta^k} \Big|_{\theta=\theta_s} = \mathbf{D}^k \mathbf{a}_s, \quad (12.60)$$

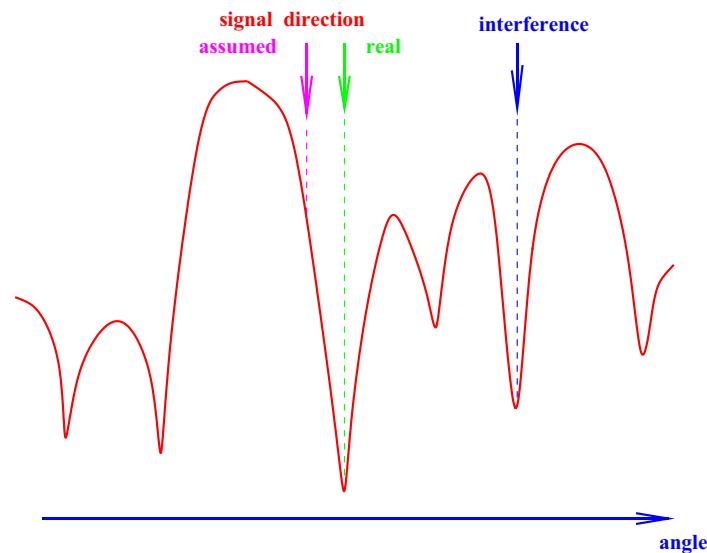
where  $\mathbf{D}$  is the matrix that depends on the SOI presumed DOA  $\theta_s$  and the array geometry.

The solution of the optimization problem can be found in a similar way as the solution of the MVDR beamformer, and it can be written as

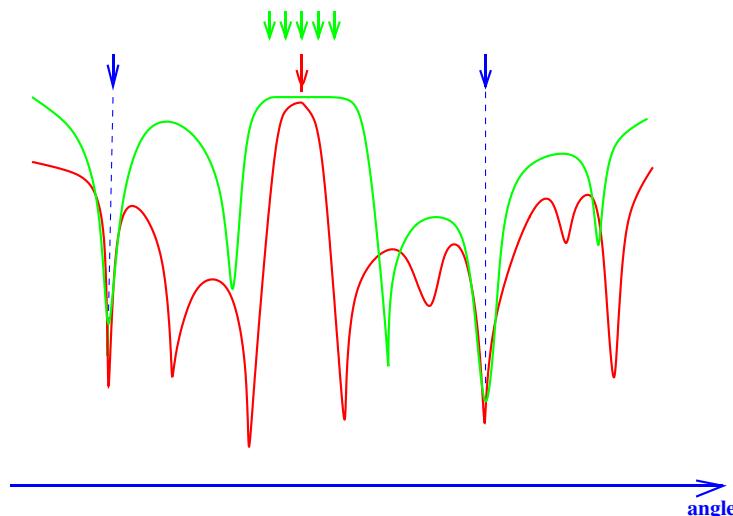
$$\mathbf{w}_{\text{opt}} = \mathbf{R}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C})^{-1} \mathbf{f}. \quad (12.61)$$

Since the data covariance matrix is unknown in practice, its sample estimate has to be used. Then the SMI version of the beamformer (12.61) is

$$\mathbf{w}_{\text{SMI}} = \widehat{\mathbf{R}}^{-1} \mathbf{C} (\mathbf{C}^H \widehat{\mathbf{R}}^{-1} \mathbf{C})^{-1} \mathbf{f}. \quad (12.62)$$

**FIGURE 12.12**

Look direction mismatch (pointing error) problem. The SOI arrives from a different direction than the presumed direction.

**FIGURE 12.13**

Pointing error. Effect of point mainlobe constraints.

### 3.12.4.4 Generalized sidelobe canceler

The solution (12.61) can be decomposed into two components, one in the constrained subspace and the other in the orthogonal subspace to the constrained subspace, as follows [27]:

$$\begin{aligned} \mathbf{w}_{\text{opt}} &= \underbrace{(\mathbf{P}_C + \mathbf{P}_C^\perp)}_{\mathbf{I}} \mathbf{w}_{\text{opt}} \\ &= \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \underbrace{\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C})^{-1}}_{\mathbf{I}} \mathbf{f} \\ &\quad + \mathbf{P}_C^\perp \mathbf{R}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C})^{-1} \mathbf{f}, \end{aligned} \quad (12.63)$$

where  $\mathbf{P}_C \triangleq \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H$  and  $\mathbf{P}_C^\perp \triangleq \mathbf{I} - \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{C}^H$  are the projection matrix on the constrained subspace and the orthogonal projection matrix on the constrained subspace, respectively.

The decomposition (12.63) can be written in a general form as

$$\mathbf{w}_{\text{GSC}} = \mathbf{w}_q - \mathbf{B} \mathbf{w}_a, \quad (12.64)$$

where

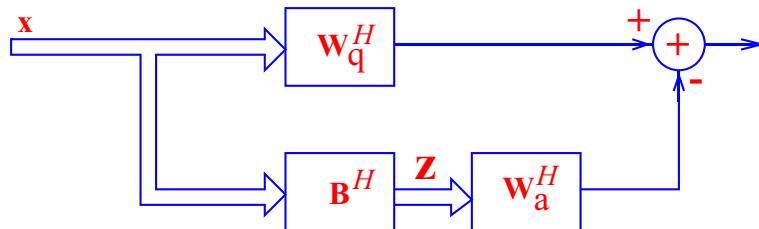
$$\mathbf{w}_q \triangleq \mathbf{C}(\mathbf{C}^H \mathbf{C})^{-1} \mathbf{f} \quad (12.65)$$

is the so-called *quiescent* beamforming vector, which is independent of the input/output data of the antenna array. The matrix  $\mathbf{B}$  in (12.64) must be selected so that

$$\mathbf{B}^H \mathbf{C} = \mathbf{0} \quad (12.66)$$

and it is called the *blocking matrix*. The vector  $\mathbf{w}_a$  is the new adaptive weight vector, while  $\mathbf{w}_q$  is non-adaptive. The beamformer (12.64) is called the *generalized sidelobe canceler* (GSC). Its block scheme is shown in Figure 12.14 and it consists of the non-adaptive branch and adaptive branch, in which the adaptive beamforming vector is applied to the data vector  $\mathbf{z}(k)$  after the blocking matrix  $\mathbf{B}$  that blocks the constrained directions.

The choice of the blocking matrix  $\mathbf{B}$  in the GSC (12.64) is not unique. In (12.63), for example, the blocking matrix  $\mathbf{B} \triangleq \mathbf{P}_C^\perp$  is used. However, in this case,  $\mathbf{B}$  is not a full-rank matrix. Therefore, it is



**FIGURE 12.14**

The block scheme of generalized sidelobe canceler.

more common to select an  $M \times (M - N)$  full-rank matrix  $\mathbf{B}$ . Then, the vectors  $\mathbf{z}(k) \triangleq \mathbf{B}^H \mathbf{x}(k)$  and  $\mathbf{w}_a$  both have shorter length of  $(M - N) \times 1$  relative to the  $M \times 1$  vectors  $\mathbf{x}(k)$  and  $\mathbf{w}_q$ . Since the non-adaptive component  $\mathbf{w}_q$  is data independent and has to be pre-computed only once, the GSC reduces the computational complexity by requiring to compute only the adaptive component  $\mathbf{w}_a$  of a shorter length. Moreover, the blocking matrix can be interpreted as a spatial filter and designed accordingly, which is a very fruitful approach especially in non-ideal situations when the assumptions of the plane waves and identical channels from air into digital processor do not hold [48].

In order to find the adaptive component  $\mathbf{w}_a$ , it can be observed that since the constrained directions are blocked by the matrix  $\mathbf{B}$ , it is guaranteed that the SOI cannot be suppressed and, therefore, the weight vector  $\mathbf{w}_a$  can adapt freely to suppress interferences by minimizing the output GSC power

$$\begin{aligned} P_{\text{GSC}} &= \mathbf{w}_{\text{opt}}^H \mathbf{R} \mathbf{w}_{\text{opt}} = (\mathbf{w}_q - \mathbf{B} \mathbf{w}_a)^H \mathbf{R} (\mathbf{w}_q - \mathbf{B} \mathbf{w}_a) \\ &= \mathbf{w}_q^H \mathbf{R} \mathbf{w}_q - \mathbf{w}_q^H \mathbf{R} \mathbf{B} \mathbf{w}_a - \mathbf{w}_a^H \mathbf{B}^H \mathbf{R} \mathbf{w}_q + \mathbf{w}_a^H \mathbf{B}^H \mathbf{R} \mathbf{B} \mathbf{w}_a. \end{aligned} \quad (12.67)$$

The unconstrained minimization of (12.67) results in the following expression for the adaptive component of the GSC:

$$\mathbf{w}_{a,\text{opt}} = (\mathbf{B}^H \mathbf{R} \mathbf{B})^{-1} \mathbf{B}^H \mathbf{R} \mathbf{w}_q. \quad (12.68)$$

Noting that

$$y(k) \triangleq \mathbf{w}_q^H \mathbf{x}(k), \quad \mathbf{z}(k) \triangleq \mathbf{B}^H \mathbf{x}(k) \quad (12.69)$$

the following covariance matrix of the data vector  $\mathbf{z}(k)$  and the correlation vector between  $\mathbf{z}(k)$  and  $y(k)$  can be introduced:

$$\begin{aligned} \mathbf{R}_z &\triangleq E[\mathbf{z}(k) \mathbf{z}^H(k)] = \mathbf{B}^H E[\mathbf{x}(k) \mathbf{x}^H(k)] \mathbf{B} \\ &= \mathbf{B}^H \mathbf{R} \mathbf{B}, \end{aligned} \quad (12.70)$$

$$\begin{aligned} \mathbf{r}_{yz} &\triangleq E[\mathbf{z}(k) y^*(k)] = \mathbf{B}^H E[\mathbf{x}(k) \mathbf{x}^H(k)] \mathbf{w}_q \\ &= \mathbf{B}^H \mathbf{R} \mathbf{w}_q. \end{aligned} \quad (12.71)$$

Using the notations (12.70) and (12.71), the expression (12.68) can be finally written as

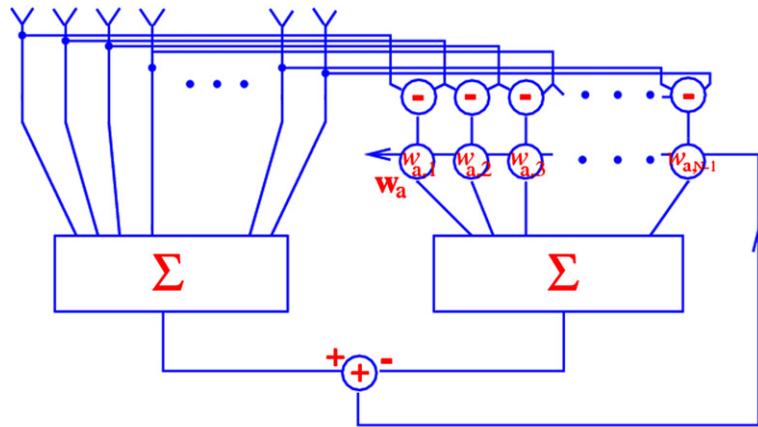
$$\mathbf{w}_{a,\text{opt}} = \mathbf{R}_z^{-1} \mathbf{r}_{yz} \quad (12.72)$$

which is again the Wiener-Hopf equation for finding optimal  $\mathbf{w}_a$  of a shorter length than  $\mathbf{w}$ .

The remaining question is how to choice the blocking matrix  $\mathbf{B}$ , if it is different from the projection matrix  $\mathbf{P}_C^\perp$ . The blocking matrix  $\mathbf{B}$  must satisfy the condition (12.66). In addition, it is desired that the dimension of the data vector at the output of  $\mathbf{B}$ , i.e., the dimension of the vector  $\mathbf{z}(k)$ , be smaller than the dimension of the data vector  $\mathbf{x}(k)$ . Thus, the matrix  $\mathbf{B}$  should be composed by linearly independent vectors  $\mathbf{b}_i$  so that  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_{M-N}]$  and the condition (12.66) becomes

$$\mathbf{b}_i \perp \mathbf{c}_k, \quad i = 1, \dots, M - N; \quad k = 1, \dots, N, \quad (12.73)$$

where  $\mathbf{c}_k$  is the  $k$ th column of the matrix  $\mathbf{C}$ .

**FIGURE 12.15**

The GSC in the case of normal direction and a single distortionless constraint for a particular choice of blocking matrix.

There are many possible choices of  $\mathbf{B}$ . For example, for the GSC shown in Figure 12.15, the matrix  $\mathbf{C}$  becomes a vector

$$\mathbf{C} = [1, 1, \dots, 1]^T, \quad (12.74)$$

while the blocking matrix  $\mathbf{B}$  is of the form

$$\mathbf{B}^H = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -1 \end{bmatrix}. \quad (12.75)$$

The corresponding vectors  $\mathbf{x}(k)$  and  $\mathbf{z}(k)$  are

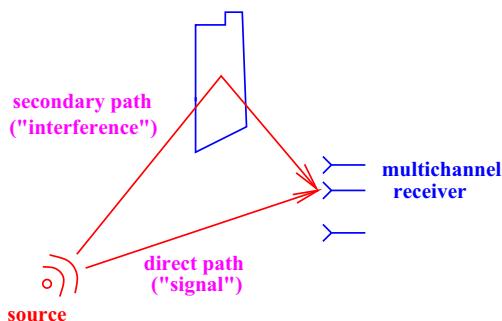
$$\mathbf{x}(k) = [x_1(k), x_2(k), \dots, x_M(k)]^T, \quad (12.76)$$

$$\mathbf{z}(k) = [x_1(k) - x_2(k), x_2(k) - x_3(k), \dots, x_{M-1}(k) - x_M(k)]^T \quad (12.77)$$

and the vector  $\mathbf{z}(k)$  has a shorter length than the vector  $\mathbf{x}(k)$  by one element.

### 3.12.4.5 Correlated (coherent) SOI and interferences: spatial smoothing

Correlation between the SOI and interferences can occur, for example, because of signal multipath propagation (this effect is shown in Figure 12.16) or because of “smart” jammers [49]. The correlation between the SOI and interferences leads to a strong signal cancellation effect. It is because the optimal beamforming vector is obtained by minimizing the array output power subject to the SOI distortionless

**FIGURE 12.16**

Correlated (coherent) signal and interferences occurring because of multipath propagation.

response constraint. If an interference is correlated (coherent) with the SOI, the minimum will be achieved if the array gain toward the interference is such that the interfering source exactly cancels the SOI. The distortionless response constraint is of no help in such a situation, since the array output does not have already the SOI component. As a result, robust techniques which would specifically address the situation of such correlation have been developed [49,50].

The following example visualizes the destructive effect of coherence (when the SOI and interference are correlated with the correlation coefficient 1). A ULA with  $M = 10$  omni-directional sensors spaced half-wavelength apart from each other is assumed. The DOA of a single SOI is  $\theta_s = 0^\circ$  and SNR = 0 dB, while the DOA of a single interference is  $\theta_i = 30^\circ$  and INR = 20 dB. Figure 12.17 depicts the beampattern of the SMI adaptive beamformer for two cases of no correlation between the SOI and interference and full coherence between the SOI and interference. It can be seen that in the incoherent case, the directional pattern of the SMI beamformer has perfect mainlobe, low sidelobes and a deep null in the direction of the interference. However, in the coherent case, the directional pattern is completely destroyed.

The main idea of adaptive beamforming techniques robust against the SOI and interferences correlation is a decorrelation of the SOI and interferences by the means of electronic subaperture motion. Such technique is called *spatial smoothing* and it is demonstrated in Figure 12.18. On the left, the antenna array partitioned into subarrays is shown, while on the right, the blocks of the data covariance matrix that correspond to different subarrays are singled out.

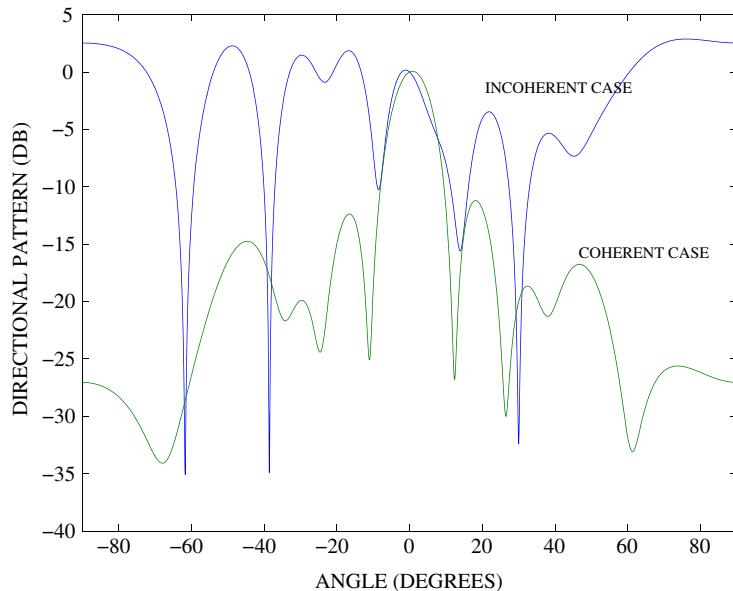
Recall that the snapshot model is

$$\begin{aligned} \mathbf{x}(k) &= s(k)\mathbf{a}(\theta_s) + \mathbf{x}_i(k) + \mathbf{x}_n(k) \\ &= \underbrace{\mathbf{As}(k)}_{\text{signal + interference}} + \mathbf{x}_n(k), \end{aligned} \quad (12.78)$$

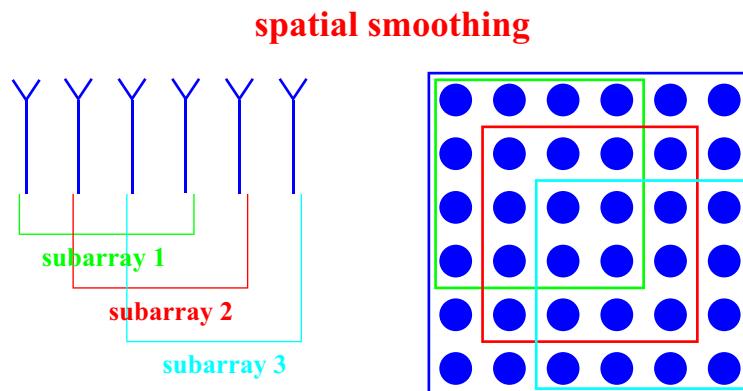
where  $s(k)$  is the vector of the waveforms of the SOI and the interferences.

According to Figure 12.18, the data vector in the  $p$ th subarray is

$$\tilde{\mathbf{x}}_p(k) = \tilde{\mathbf{A}}\mathbf{D}^{p-1}\mathbf{s}(k) + \tilde{\mathbf{x}}_{n,p}(k), \quad (12.79)$$

**FIGURE 12.17**

Directional patterns of SMI beamformer for incoherent and coherent cases.

**FIGURE 12.18**

Decorrelation of the desired signal and interference by the means of electronic subaperture motion. The array consists of 6 sensors which form 3 subarrays of 4 sensors (on the left). A block of the data covariance matrix corresponds to each subarray (on the right). There are 3 covariance matrices for 3 subarrays.

where  $\tilde{\mathbf{A}}$  has a reduced dimension relative to  $\mathbf{A}$  and for the ULA configuration

$$\mathbf{D} \triangleq \text{diag} \left\{ e^{j \frac{\omega}{c} d \sin \theta_s}, e^{j \frac{\omega}{c} d \sin \theta_{i_2}}, \dots, e^{j \frac{\omega}{c} d \sin \theta_{i_L}} \right\} \quad (12.80)$$

is a diagonal matrix, where  $\theta_s, \theta_{i_2}, \dots, \theta_{i_L}$  are the DOA's of the SOI and  $L$  interferences and  $d$  is the distance between any two adjacent antenna elements in ULA.

Let the number of subarrays be

$$P = M - J + 1, \quad (12.81)$$

where  $J$  is the subarray dimension, i.e., the number of antenna elements in one subarray. Then the  $J \times J$  spatially smoothed covariance matrix can be determined as

$$\tilde{\mathbf{R}} \triangleq \frac{1}{P} \sum_{p=1}^P \tilde{\mathbf{R}}_p, \quad (12.82)$$

where

$$\tilde{\mathbf{R}}_p \triangleq \mathbb{E}\{\tilde{\mathbf{x}}_p(k)\tilde{\mathbf{x}}_p^H(k)\} \quad (12.83)$$

is the covariance matrix for the  $p$ th subarray.

Substituting (12.79) in (12.83) and then substituting the result in (12.82), we obtain that

$$\begin{aligned} \tilde{\mathbf{R}} &\triangleq \frac{1}{P} \sum_{p=1}^P \tilde{\mathbf{R}}_p = \frac{1}{P} \sum_{p=1}^P \mathbb{E}[\tilde{\mathbf{x}}_p(k)\tilde{\mathbf{x}}_p^H(k)] \\ &= \frac{1}{P} \sum_{p=1}^P \tilde{\mathbf{A}} \mathbf{D}^{p-1} \underbrace{\mathbb{E}[\mathbf{s}(k)\mathbf{s}^H(k)]}_{\mathbf{S}} (\mathbf{D}^*)^{p-1} \tilde{\mathbf{A}}^H + \sigma_n^2 \mathbf{I} \\ &= \tilde{\mathbf{A}} \left[ \frac{1}{P} \sum_{p=1}^P \mathbf{D}^{p-1} \mathbf{S} (\mathbf{D}^*)^{p-1} \right] \tilde{\mathbf{A}}^H + \sigma_n^2 \mathbf{I} \\ &= \tilde{\mathbf{A}} \tilde{\mathbf{S}} \tilde{\mathbf{A}}^H + \sigma_n^2 \mathbf{I}, \end{aligned} \quad (12.84)$$

where the new notation

$$\tilde{\mathbf{S}} \triangleq \frac{1}{P} \sum_{p=1}^P \mathbf{D}^{p-1} \mathbf{S} (\mathbf{D}^*)^{p-1} \quad (12.85)$$

is introduced.

Coherence between the SOI and interferences leads to a singular source covariance matrix  $\mathbf{S}$ . It is straightforward to see from (12.85) that even in the case of singular  $\mathbf{S}$ , the matrix  $\tilde{\mathbf{S}}$  becomes nonsingular if the number of subarrays is greater than the number of sources, that is,

$$P > L + 1. \quad (12.86)$$

The decorrelation factor between sources  $i$  and  $l$  due to spatial smoothing can also be found and it is expressed as [51]

$$|g_{il}| = \left| \frac{\sin\{P(\omega d/2c)(\sin \theta_i - \sin \theta_l)\}}{P \sin\{(\omega d/2c)(\sin \theta_i - \sin \theta_l)\}} \right|. \quad (12.87)$$

Then the  $(i, l)$ th elements of the matrices  $\mathbf{S}$  and  $\tilde{\mathbf{S}}$  are given, respectively, as

$$[\mathbf{S}]_{i,l} = \sigma_i \sigma_l \rho_{i,l}, \quad [\tilde{\mathbf{S}}]_{i,l} = \sigma_i \sigma_l \rho_{i,l} g_{i,l}, \quad (12.88)$$

where  $\sigma_i^2$  and  $\sigma_l^2$  are the variances of  $i$ th and  $l$ th sources, respectively, and  $\rho_{i,l}$  is the correlation coefficient between the  $i$ th and  $l$ th sources.

### 3.12.4.6 Forward-backward averaging and spatial smoothing

The spatial smoothing source decorrelation method, however, severely reduces the antenna array length because the number of subarrays must be larger than the number of correlated sources according to (12.81). Moreover, the spatial smoothing method does not exploit the structure of ULA or, in general, any array with centro-symmetric geometry, in a full measure. The antenna array length can be enlarged by the means of the so-called *forward-backward (FB) spatial smoothing* [52]. It is based on the observation that for any array with centro-symmetric geometry, we have

$$\mathbf{J}\mathbf{a}(\theta) = \begin{bmatrix} 0 & \dots & 0 & 0 & 1 \\ 0 & \dots & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ e^{j\frac{\omega}{c}d \sin \theta} \\ \vdots \\ e^{j\frac{\omega}{c}(N-1)d \sin \theta} \end{bmatrix} = \begin{bmatrix} e^{j\frac{\omega}{c}(N-1)d \sin \theta} \\ \vdots \\ e^{j\frac{\omega}{c}d \sin \theta} \\ 1 \end{bmatrix} = e^{j\frac{\omega}{c}(N-1)d \sin \theta}, \quad (12.89)$$

where  $\mathbf{J}$  is the exchange matrix.

In application to the covariance matrix for uncorrelated sources, the observation (12.89) leads to the following interesting result:

$$\begin{aligned} \mathbf{J}\mathbf{R}^*\mathbf{J} &= \mathbf{J} \left[ \sum_{l=1}^{L+1} \sigma_l^2 \mathbf{a}(\theta_l) \mathbf{a}^H(\theta_l) \right]^* \mathbf{J} + \sigma_n^2 \underbrace{\mathbf{J}^2}_{\mathbf{I}} \\ &= \left[ \sum_{l=1}^{L+1} \sigma_l^2 \mathbf{a}^*(\theta_l) \mathbf{a}^T(\theta_l) \right]^* + \sigma_n^2 \mathbf{I} \\ &= \sum_{l=1}^{L+1} \sigma_l^2 \mathbf{a}(\theta_l) \mathbf{a}^H(\theta_l) + \sigma_n^2 \mathbf{I} = \mathbf{R} \end{aligned} \quad (12.90)$$

that is, the covariance matrix  $\mathbf{R}$  is the so-called *centro-Hermitian* matrix, satisfying  $\mathbf{R} = \mathbf{J}\mathbf{R}^*\mathbf{J}$ .

Using (12.90), the idea of the FB averaging is to decorrelate the coherent/correlated sources by the means of enforcing the centro-Hermitian property, i.e., by computing the following FB covariance matrix:

$$\mathbf{R}_{\text{FB}} \triangleq \frac{1}{2} (\mathbf{R} + \mathbf{J}\mathbf{R}^*\mathbf{J}). \quad (12.91)$$

Combining the FB averaging with spatial smoothing, it is possible to double the number of subarrays while keeping the subarray length the same as in the conventional spatial smoothing. Then the covariance matrix for the FB spatial smoothing is defined as

$$\tilde{\mathbf{R}}_{\text{FB}} \triangleq \frac{1}{2P} \sum_{p=1}^P \left( \tilde{\mathbf{R}}_p + \mathbf{J} \tilde{\mathbf{R}}_p^* \mathbf{J} \right). \quad (12.92)$$

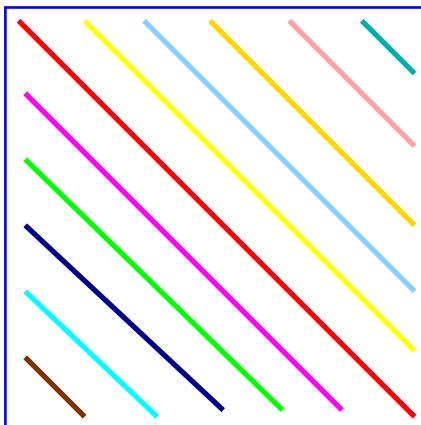
As a result, the same decorrelation factor can be archived by the FB spatial smoothing with the use of subarrays of a bigger size as the one achieved by the conventional spatial smoothing with subarrays of a smaller size.

The FB spatial smoothing method can be further generalized by introducing the weights  $c_p$  ( $p = 1, \dots, P$ ) as follows

$$\tilde{\mathbf{R}}_{\text{wFB}} \triangleq \frac{1}{2P} \sum_{p=1}^P c_p \left( \tilde{\mathbf{R}}_p + \mathbf{J} \tilde{\mathbf{R}}_p^* \mathbf{J} \right) \quad (12.93)$$

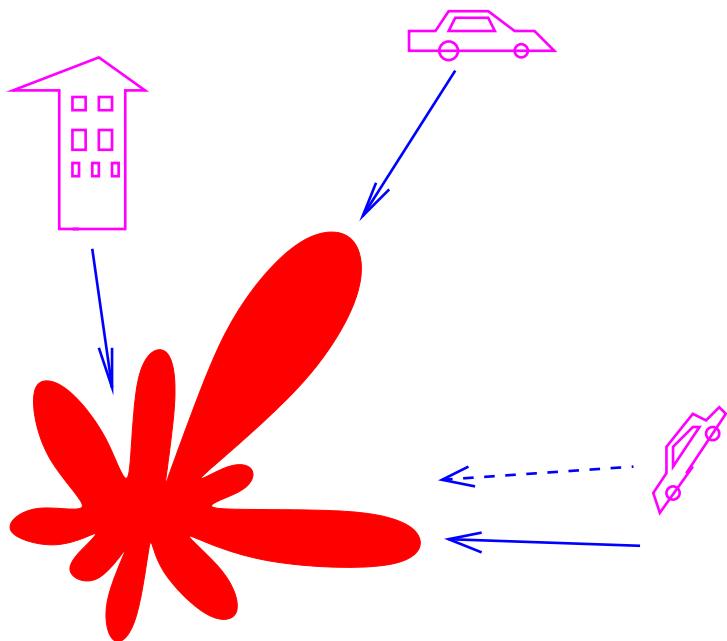
and optimizing these weights to minimize the source decorrelation factor further.

One more simple source decorrelation method, named *redundancy averaging* [53], is based on the fact that the true data covariance matrix in a ULA must be a Toeplitz matrix. The sample estimate of the data covariance matrix is in general not a Toeplitz matrix. However, the Toeplitz structure can be enforced, for example, by averaging the diagonals of the sample covariance matrix as it is shown in Figure 12.19. In addition to enforcing the Toeplitz structure, such redundancy averaging also leads to source decorrelation.



**FIGURE 12.19**

Redundancy averaging source decorrelation method. All diagonals are averaged and each element in the same diagonal takes the corresponding average value. It also leads to source decorrelation.

**FIGURE 12.20**

Example of rapidly moving interference.

### 3.12.4.7 Rapidly moving interferences

If the rate/vilosity of interference motion is faster than the rate of adaptation of the adaptive array, the antenna array may be unable to follow such rapid changes of the interference position. The situation of fast moving interference is, however, common in a large number of applications [19]. An example of such situation is shown in Figure 12.20. The nulls of the beampattern are very narrow and even relatively slow interference motion may lead to the situation when the interference leaks to the output of adaptive beamformer through a sidelobe that may significantly reduce the output SINR of adaptive beamformer.

The situation of rapidly moving interferences is typically addressed in terms of broadening the adaptive pattern nulls towards the interfering sources. The main difficulty here is that the DOAs of interfering sources are unknown. However, the null width even towards interfering sources with unknown DOAs can be increased by replacing the sample covariance matrix with the modified covariance matrix of the form [19,54]

$$\widehat{\mathbf{R}}_{\text{tap}} = \widehat{\mathbf{R}} \odot \mathbf{T}, \quad (12.94)$$

where  $\odot$  stands for element-wise Hadamard product of matrices and the matrix  $\mathbf{T}$  is a positive semi-definite matrix that is called a *taper matrix*. The choice of taper is not unique. In the most popular

tapper proposed in [54], the  $(i, l)$ th element of the taper matrix  $\mathbf{T}$  is given by

$$[\mathbf{T}]_{i,l} = \frac{\sin(|i - l|\Delta)}{|i - l|\Delta}, \quad (12.95)$$

where  $\Delta$  defines the required null width.

Broadening the adaptive pattern nulls can also be archived by the means of enforcing the so-called data-dependent derivative constraints [55]. The resulting covariance matrix takes the form

$$\widehat{\mathbf{R}}_{\text{ddc}} = \widehat{\mathbf{R}} + \xi \mathbf{D} \widehat{\mathbf{R}} \mathbf{D}, \quad (12.96)$$

where  $\mathbf{D}$  is the known diagonal matrix of sensor coordinates and the parameter  $\xi$  defines the tradeoff between null depth and width. It has been shown that the null broadening method based on (12.96) can be also interpreted in terms of the covariance tapering method of (12.94) (see [55]).

### 3.12.4.8 Unified principle to MVDR robust adaptive beamforming design

As we have seen earlier, the SMI beamformer is not robust to an imperfect knowledge of the SOI steering vector. Different robust adaptive beamforming techniques which use different specific notions of robustness such as robustness against small sample size, pointing error, coherence between the SOI and interferences, and rapid interference motion have been revised. Each of these notions of robustness is very specific. For example, the point or derivative mainbeam constraint-based beamforming technique is very useful for overcoming pointing error problem, but it does not help in the general case of mismatched SOI wavefront and finite sample size when the SOI is present in the antenna array measurements.

The general meaning of robustness for any robust adaptive beamforming technique can be, however, defined as the ability of such technique to compute the beamforming vector so that the SINR is maximized despite possibly imperfect and incomplete knowledge of required prior information. More specifically, the aforementioned signal cancellation effect for the SMI beamformer occurs in the situation when the SOI steering vector is misinterpreted with any of the interference steering vectors of their linear combinations. Thus, if with incomplete and/or imperfect prior information, a robust adaptive beamforming technique is able to estimate the SOI steering vector so that the estimate does not converge to any of the interferences and their linear combinations, such technique is called robust. Using this notion of robustness, the unified principle to robust adaptive beamforming design based on MVDR beamformer can be formulated as follows. Use the standard SMI beamformer (12.36) in tandem with SOI steering vector estimation performed based on some possibly incomplete and inaccurate prior information. The difference between different MVDR robust adaptive beamforming techniques can be then shown to boil down to the differences in the assumed prior information, the specific notions of robustness, and the corresponding steering vector estimation techniques used.

Hereafter, the imperfectly known presumed SOI steering vector is denoted as  $\mathbf{p}$ , while  $\mathbf{a}$  stands for the actual SOI steering vector that is different from  $\mathbf{p}$ , i.e.,  $\mathbf{a} \neq \mathbf{p}$ . The estimate of the actual SOI steering vector is denoted as  $\hat{\mathbf{a}}$ . In the techniques that follow, the estimate  $\hat{\mathbf{a}}$  is found by using different prior information and based on different principles. Other than that, all the techniques are based on the aforementioned unified principle.

Many modern robust adaptive beamforming techniques are based on convex optimization theory [16, 56, 57]. Most of such robust beamformers cannot be expressed in closed-form, i.e., cannot be written in terms of the covariance matrix inversion or eigenvalue decomposition. However, the complexity of solving optimization problems that correspond to such beamforming techniques is comparable to the complexity of matrix inversion. Thus, there is no significant difference in terms of computational complexity between the so-called closed-form solutions and numerical solutions of convex problems.

### 3.12.4.9 Eigenspace-based beamformer

Using the a priori knowledge of the presumed SOI steering vector  $\mathbf{p}$ , the eigenspace-based beamformer computes and uses the projection of  $\mathbf{p}$  onto the sample signal-plus-interference subspace as a corrected estimate of the actual SOI steering vector. The eigendecomposition of (12.5) yields

$$\widehat{\mathbf{R}} = \mathbf{E}\Lambda\mathbf{E}^H + \mathbf{G}\Gamma\mathbf{G}^H, \quad (12.97)$$

where the  $M \times (L + 1)$  matrix  $\mathbf{E}$  and  $M \times (M - L - 1)$  matrix  $\mathbf{G}$  contain the signal-plus-interference subspace eigenvectors of  $\widehat{\mathbf{R}}$  and the noise subspace eigenvectors, respectively, while the  $(L + 1) \times (L + 1)$  matrix  $\Lambda$  and  $(M - L - 1) \times (M - L - 1)$  matrix  $\Gamma$  contain the eigenvalues corresponding to  $\mathbf{E}$  and  $\mathbf{G}$ , respectively, and as before  $L$  stands for the number of interfering signals.

The estimate of the actual SOI steering vector is found as

$$\hat{\mathbf{a}} = \mathbf{E}\mathbf{E}^H\mathbf{p}, \quad (12.98)$$

where  $\mathbf{E}\mathbf{E}^H$  is the projection matrix to the desired signal-plus-interference subspace. Then the eigenspace-based beamformer is obtained by substituting the so-obtained estimate of the steering vector to the SMI beamformer (12.36), and it can be expressed as [58]

$$\mathbf{w}_{\text{eig}} = \widehat{\mathbf{R}}^{-1}\hat{\mathbf{a}} = \widehat{\mathbf{R}}^{-1}\mathbf{E}\mathbf{E}^H\mathbf{p} = \mathbf{E}\Lambda^{-1}\mathbf{E}^H\mathbf{p}, \quad (12.99)$$

where the fact that

$$\widehat{\mathbf{R}}^{-1}\mathbf{E}\mathbf{E}^H = (\mathbf{E}\Lambda\mathbf{E}^H + \mathbf{G}\Gamma\mathbf{G}^H)^{-1}\mathbf{E}\mathbf{E}^H = \mathbf{E}\Lambda^{-1}\mathbf{E}^H \quad (12.100)$$

has been used for obtaining the last equality and  $\mathbf{G}^H\mathbf{E} = \mathbf{0}$  because  $\mathbf{G}$  and  $\mathbf{E}$  are orthogonal (see the decomposition (12.97)).

Summarizing, the essence of the eigenspace-based beamforming technique is to project the presumed SOI steering vector onto the measured signal-plus-interference subspace prior to processing in order to reduce the signal wavefront mismatch. Then, the estimate of the actual SOI steering vector is plugged to the standard SMI beamformer. The interference rejection part remains unchanged for this beamformer as compared to the SMI beamformer. The prior information used is the presumed steering vector  $\mathbf{p}$  and the number of interfering sources  $L$ . The notion of robustness is the projection of the presumed steering vector to the signal-plus-interference subspace. It is, however, well known that at low SNR, the eigenspace-based beamformer suffers from a high probability of subspace swap and incorrect estimation of the signal-plus-interference subspace dimension [59].

### 3.12.4.10 Worst-case-based robust adaptive beamforming

This approach is based on explicitly modeling of the actual SOI steering vector  $\mathbf{a}$  as a sum of the presumed steering vector and a deterministic norm bounded mismatch vector  $\boldsymbol{\delta}$ , that is,

$$\mathbf{a} \triangleq \mathbf{p} + \boldsymbol{\delta}, \quad \|\boldsymbol{\delta}\| \leq \varepsilon, \quad (12.101)$$

where  $\varepsilon$  is some a priori known bound. Thus, the worst-case-based robust adaptive beamformer uses the prior information about the presumed steering vector and the information that the mismatch vector is norm bounded [60]. An ellipsoidal uncertainty region can also be considered instead of the mentioned in (12.101) spherical uncertainty [61]. However, a more sophisticated prior information has to be available in the case of ellipsoidal uncertainty. Assuming spherical uncertainty for  $\boldsymbol{\delta}$ , i.e., introducing the uncertainty set

$$\mathcal{A}(\boldsymbol{\delta}) \triangleq \{\mathbf{a} = \mathbf{p} + \boldsymbol{\delta} \mid \|\boldsymbol{\delta}\| \leq \varepsilon\} \quad (12.102)$$

the worst-case-based robust adaptive beamforming aims at solving the following optimization problem [60]

$$\min_{\mathbf{w}} \mathbf{w}^H \widehat{\mathbf{R}} \mathbf{w} \text{ subject to } \min_{\hat{\mathbf{a}} \in \mathcal{A}(\boldsymbol{\delta})} |\mathbf{w}^H \hat{\mathbf{a}}| \geq 1. \quad (12.103)$$

The optimization problem (12.103) is equivalent to the following second-order cone (SOC) programming problem [60]

$$\min_{\mathbf{w}} \mathbf{w}^H \widehat{\mathbf{R}} \mathbf{w} \text{ subject to } \mathbf{w}^H \mathbf{p} \geq \varepsilon \|\mathbf{w}\| + 1, \quad (12.104)$$

which can be solved efficiently using standard numerical optimization methods with complexity comparable to the complexity of matrix inversion.

The worst-case-based robust adaptive beamforming technique (12.103) can be equivalently interpreted as the standard SMI beamformer used in tandem with the SOI steering vector estimate obtained by solving the following covariance fitting problem [62]

$$\min_{\sigma^2, \hat{\mathbf{a}}} \sigma^2 \text{ subject to } \widehat{\mathbf{R}} - \sigma^2 \hat{\mathbf{a}} \hat{\mathbf{a}}^H \geq 0 \text{ for any } \hat{\mathbf{a}} \text{ satisfying } \|\boldsymbol{\delta}\| \leq \varepsilon. \quad (12.105)$$

Summarizing, the prior information used in the worst-case-based robust adaptive beamforming techniques is the presumed steering vector and the value  $\varepsilon$ , which may be difficult to obtain in practice. The notion of robustness is the uncertainty region for the presumed steering vector. The robustness to the rapidly moving interference sources can also be added to the worst-case-based robust adaptive beamforming [63].

### 3.12.4.11 Relationship between the worst-case-based and the LSMI adaptive beamformers

Note that the constraint in the optimization problem (12.103) must be satisfied with equality at optimality. Indeed, if the constraint is not satisfied with equality, then the minimum of the objective function in (12.103) is achieved when  $\kappa \triangleq \min_{\hat{\mathbf{a}} \in \mathcal{A}(\boldsymbol{\delta})} |\mathbf{w}^H \hat{\mathbf{a}}| > 1$ . However, by replacing  $\mathbf{w}$  with  $\mathbf{w}/\sqrt{\kappa}$ , the objective function of (12.103) can be decreased by the factor of  $\kappa > 1$ , whereas the constraint in (12.103) will be still satisfied. This contradicts the original statement that the objective function is

minimized when  $\kappa > 1$ . Therefore, the minimum of the objective function is achieved at  $\kappa = 1$ , and the inequality constraint in (12.103) is equivalent to the equality constraint. This also means that  $\mathbf{w}^H \hat{\mathbf{a}}$  is real-valued and positive. Using these facts, the problem (12.103) can be rewritten as

$$\min_{\mathbf{w}} \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} \quad \text{subject to} \quad (\mathbf{w}^H \mathbf{p} - 1)^2 = \varepsilon^2 \mathbf{w}^H \mathbf{w}. \quad (12.106)$$

The solution to (12.106) can be found by using the method Lagrange multipliers, i.e., by optimizing the following Lagrangian:

$$L(\mathbf{w}, \lambda) = \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} + \lambda(\varepsilon^2 \mathbf{w}^H \mathbf{w} - (\mathbf{w}^H \mathbf{p} - 1)^2), \quad (12.107)$$

where  $\lambda$  is a Lagrange multiplier. Taking the gradient of (12.107) and equating it to zero, it can be found that

$$\mathbf{w} = -\lambda(\hat{\mathbf{R}} + \lambda\varepsilon^2 \mathbf{I} - \lambda \mathbf{p} \mathbf{p}^H)^{-1} \mathbf{p}. \quad (12.108)$$

Furthermore, applying the matrix inversion lemma to (12.108), the beamforming vector can be expressed as [60]

$$\mathbf{w} = \frac{\lambda}{\lambda \mathbf{p}^H (\hat{\mathbf{R}} + \lambda \varepsilon^2 \mathbf{I})^{-1} \mathbf{p} - 1} (\hat{\mathbf{R}} + \lambda \varepsilon^2 \mathbf{I})^{-1} \mathbf{p}, \quad (12.109)$$

which is the LSMI beamformer with adaptive diagonal loading factor. The expression (12.109) cannot be used practically since the optimal value of  $\lambda$  has to be first found. The numerical algorithms designed in [61] are particularly based on finding  $\lambda$  numerically, while the general SOC programming is used in [60]. The complexity of both type of methods is, however, the same and is comparable to the matrix inversion as in SMI and LSMI beamformers.

### 3.12.4.12 Doubly constrained robust adaptive beamforming

The doubly constrained robust adaptive beamforming [64] is similar to the worst-case-based one (12.103) (equivalently (12.105)), but it exposes also an additional constraint to the norm of the steering vector estimate, that is,  $\|\hat{\mathbf{a}}\|^2 = M$ . Then the corresponding optimization problem for finding  $\hat{\mathbf{a}}$  is

$$\begin{aligned} \min_{\sigma^2, \hat{\mathbf{a}}} \sigma^2 \quad &\text{subject to} \quad \hat{\mathbf{R}} - \sigma^2 \hat{\mathbf{a}} \hat{\mathbf{a}}^H \geq 0, \\ &\text{for any } \hat{\mathbf{a}} \text{ satisfying } \|\delta\| \leq \varepsilon, \quad \|\hat{\mathbf{a}}\|^2 = M. \end{aligned} \quad (12.110)$$

This method uses the same prior information as the worst-case-based robust adaptive beamforming method and obviously fits under the aforementioned unified framework. Although the spherical uncertainty region is considered in [64], it can be relatively straightforwardly extended to the ellipsoidal uncertainty region [65]. Clearly, the notion of robustness for this method is the same as for the worst-case-based one. Due to the constraint  $\|\hat{\mathbf{a}}\|^2 = M$ , the doubly constrained robust adaptive beamforming provides a better estimate of the SOI than the worst-case-based robust adaptive beamforming. It can be important in the applications where such estimate is needed.

### 3.12.4.13 Probabilistically constrained robust adaptive beamforming

Another approach to robust adaptive beamforming is based on the assumption that the mismatch vector  $\delta$  is random. Then the problem has to be formulated in probabilistic terms in contrast to the deterministic terms used in the worst-case-based design. Specifically, the probabilistically constrained robust adaptive beamforming problem is formulated as [66]

$$\min_{\mathbf{w}} \mathbf{w}^H \widehat{\mathbf{R}} \mathbf{w} \text{ subject to } \Pr\{|\mathbf{w}^H \mathbf{a}| \geq 1\} \geq p_0, \quad (12.111)$$

where  $\Pr\{\cdot\}$  denotes probability and  $p_0$  is preselected probability value. In this case, the prior information is the presumed steering vector  $\mathbf{p}$  as before, but since the steering vector mismatch is assumed to be random, the other prior information is the distribution type and the distribution variance of  $\delta$  as well as the non-outage probability  $p_0$  for the distortionless response constraint. In two cases when  $\delta$  is Gaussian distributed and the distribution of  $\delta$  is unknown and assumed to be the worst possible, it has been shown that the problem (12.111) can be closely approximated by the following problem [66]:

$$\min_{\mathbf{w}} \mathbf{w}^H \widehat{\mathbf{R}} \mathbf{w} \text{ subject to } \tilde{\varepsilon} \|\mathbf{Q}_\delta^{1/2} \mathbf{w}\| \leq \mathbf{w}^H \mathbf{p} - 1, \quad (12.112)$$

where  $\mathbf{Q}_\delta$  is the covariance matrix of the random mismatch vector  $\delta$  and  $\tilde{\varepsilon} = \sqrt{-\ln(1-p_0)}$  if  $\delta$  is Gaussian distributed and  $\tilde{\varepsilon} = 1/\sqrt{1-p_0}$  if the distribution of  $\delta$  is unknown. Thus, the latter problem boils down mathematically to the same form as the worst-case-based robust adaptive beamforming problem and can also be considered as a part of the earlier explained unified framework. However, the prior information required for the probabilistically constrained robust adaptive beamforming may be easier to obtain than that for the worst-case-based approach since it is typically easier to estimate the statistics of the mismatch distribution reliably, while  $p_0$  has a clear physical meaning. The non-outage probability  $p_0$  for the distortionless response constraint is the specific notion of robustness used in this approach.

### 3.12.4.14 Sequential quadratic programming-based robust adaptive beamforming

The title of this approach refers to the optimization technique used, but its essence is significantly different from the above approaches that are based on the same aforementioned unified principle to robust MVDR beamforming design. According to this approach the estimate of the actual steering vector  $\mathbf{a}$  is found so that the beamformer output power is maximized while the convergence of the estimate  $\hat{\mathbf{a}}$  to any interference steering vector is prohibited [67]. The rationale behind maximization of the beamformer output power is the following. In the steering vector mismatched case, the solution (12.36) can be re-written as a function of unknown  $\delta$ , that is,  $\mathbf{w}(\delta) = \alpha \widehat{\mathbf{R}}^{-1}(\mathbf{p} + \delta)$ . Using  $\mathbf{w}(\delta)$ , the beamformer output power can be also written as a function of the mismatch  $\delta$  as

$$P(\delta) = \frac{1}{(\mathbf{p} + \delta)^H \widehat{\mathbf{R}}^{-1}(\mathbf{p} + \delta)}. \quad (12.113)$$

Thus, the estimate of  $\delta$  or, equivalently, the estimate of  $\mathbf{a}$  that maximizes (12.113) will be the best estimate of the actual steering vector  $\mathbf{a}$  under the constraints that the norm of  $\hat{\mathbf{a}}$  equals  $\sqrt{M}$  and  $\hat{\mathbf{a}}$  does

not converge to any of the interference steering vectors. The latter is guaranteed in this method by requiring that

$$\mathbf{P}^\perp(\mathbf{p} + \hat{\boldsymbol{\delta}}) = \mathbf{P}^\perp\hat{\mathbf{a}} = 0, \quad (12.114)$$

where  $\mathbf{P}^\perp \triangleq \mathbf{I} - \mathbf{U}\mathbf{U}^H$ ,  $\mathbf{U} \triangleq [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_T]$ ,  $\mathbf{u}_l, l = 1, \dots, T$  are the  $T$  dominant eigenvectors of the matrix  $\mathbf{C} \triangleq \int_{\Theta} \mathbf{d}(\theta)\mathbf{d}^H(\theta)d\theta$ ,  $\mathbf{d}(\theta)$  is the steering vector associated with direction  $\theta$  and having the structure defined by the antenna geometry,  $\Theta$  is the angular sector in which the SOI is located,  $\hat{\boldsymbol{\delta}}$  and  $\hat{\mathbf{a}}$  stand for the estimates of the steering vector mismatch and the actual SOI steering vector, respectively. The optimization problem for finding the estimate  $\hat{\mathbf{a}}$  can be written as [67]

$$\begin{aligned} & \min_{\hat{\mathbf{a}}} \hat{\mathbf{a}}^H \hat{\mathbf{R}}^{-1} \hat{\mathbf{a}} \\ \text{subject to } & \mathbf{P}^\perp \hat{\mathbf{a}} = \mathbf{0}, \quad \|\hat{\mathbf{a}}\|^2 = M, \\ & \hat{\mathbf{a}}^H \tilde{\mathbf{C}} \hat{\mathbf{a}} \leq \mathbf{p}^H \mathbf{C} \mathbf{p}, \end{aligned} \quad (12.115)$$

where  $\tilde{\mathbf{C}} \triangleq \int_{\tilde{\Theta}} \mathbf{d}(\theta)\mathbf{d}^H(\theta)d\theta$  and the sector  $\tilde{\Theta}$  is the complement of the sector  $\Theta$ . The last constraint in (12.115) limits the noise power collected in  $\tilde{\Theta}$ .

Since the optimization problem (12.115) is non-convex and difficult to solve, it is modified so that the orthogonal component of  $\boldsymbol{\delta}$  (here  $\boldsymbol{\delta}$  is decomposed to colinear and orthogonal components) is estimated iteratively as shown in Figure 12.21, while at each iteration the following quadratic (convex) optimization problem is solved

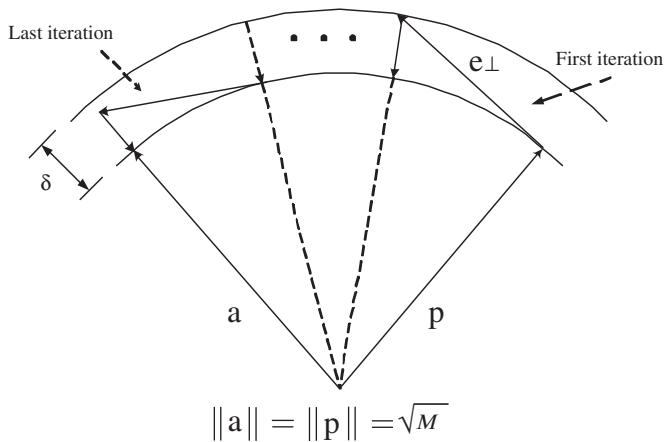
$$\begin{aligned} & \min_{\hat{\boldsymbol{\delta}}_\perp} (\mathbf{p} + \hat{\boldsymbol{\delta}}_\perp)^H \hat{\mathbf{R}}^{-1} (\mathbf{p} + \hat{\boldsymbol{\delta}}_\perp) \\ \text{subject to } & \mathbf{P}^\perp(\mathbf{p} + \hat{\boldsymbol{\delta}}_\perp) = \mathbf{0}, \\ & \|\mathbf{p} + \hat{\boldsymbol{\delta}}_\perp\|^2 \leq M, \\ & \mathbf{p}^H \hat{\boldsymbol{\delta}}_\perp = 0, \quad \|\hat{\mathbf{a}}\|^2 = M, \\ & \mathbf{a}^H \tilde{\mathbf{C}} \hat{\mathbf{a}} \leq \mathbf{p}^H \mathbf{C} \mathbf{p}, \end{aligned} \quad (12.116)$$

where  $\hat{\boldsymbol{\delta}}_\perp$  is the component of  $\hat{\boldsymbol{\delta}}$  that is orthogonal to  $\mathbf{p}$  and the orthogonality between  $\hat{\boldsymbol{\delta}}$  and  $\mathbf{p}$  is imposed by adding the constraint  $\mathbf{p}^H \hat{\boldsymbol{\delta}}_\perp = 0$ . Because the quadratic programming problem (12.116) has to be solved sequentially, the corresponding method is called the sequential quadratic programming (SQP)-based robust adaptive beamforming.

It can be seen that the prior information used in this approach is the presumed steering vector and the angular sector  $\Theta$  in which the desired signal is located. Note that if the constraint (12.114) is replaced by the constraint  $\|\boldsymbol{\delta}\| \leq \varepsilon$ , the convergence to an interference steering vector is also be avoided, but the problem then becomes equivalent to the worst-case-based robust adaptive beamforming (see [64]). This technique can be simplified for more structured uncertainties, for example, when it is known that the array is partially calibrated [68]. However, the amount of required prior information about the uncertainty then increases.

### 3.12.4.15 Eigenvalue beamforming using multi-rank MVDR beamformer

If the desired signal and interference steering vectors lie in known signal subspaces and the rank of the signal correlation matrix is known, the eigenvalue beamforming using multi-rank MVDR beamformer

**FIGURE 12.21**

Convergence trajectory of the iterative robust adaptive beamforming algorithm.

can be efficient [69]. The multi-rank beamformer matrix is computed as

$$\mathbf{W} = \widehat{\mathbf{R}}^{-1} \boldsymbol{\Psi} (\boldsymbol{\Psi}^H \widehat{\mathbf{R}}^{-1} \boldsymbol{\Psi})^{-1} \mathbf{Q}, \quad (12.117)$$

where  $\mathbf{Q}$  is a data dependent left-orthogonal matrix, i.e.,  $\mathbf{Q}^H \mathbf{Q} = \mathbf{I}$ , and  $\boldsymbol{\Psi}$  is the matrix with the columns that span the linear subspace in which the SOI lies. For example, for resolving a signal with a rank-one covariance matrix and an unknown but fixed DOA, the columns of  $\mathbf{Q}$  should be selected as the dominant eigenvectors of the mismatch covariance matrix, i.e.,

$$\mathbf{R}_\delta = (\boldsymbol{\Psi}^H \mathbf{R}^{-1} \boldsymbol{\Psi})^{-1}. \quad (12.118)$$

If it is assumed that the signal lies in a known subspace, but the DOA is unknown and unfixed (randomly changes from snapshot to snapshot), it is the subdominant eigenvectors of the mismatch covariance matrix that should be used as the columns of the matrix  $\mathbf{Q}$ .

The prior information required for this beamforming technique is the linear subspace in which the desired signal lies and the rank of the desired signal covariance matrix. The main disadvantages are that a very specific modeling of the covariance matrix is used and the signal subspace has to be known.

### 3.12.4.16 Robust adaptive beamforming based on steering vector estimation with as little as possible prior information

The essence of robustness can be practically viewed as an ability of adaptive beamformer to achieve acceptably high output SINR despite imprecise and perhaps very limited prior information. This beamforming technique aims at fulfilling such most general notion of robustness. Assume that the SOI lies in the known angular sector  $\Theta = [\theta_{\min}, \theta_{\max}]$  that is distinguishable from general locations of the interfering signals. The estimate  $\hat{\mathbf{a}}$  can be forced not to converge to any vector with DOAs within the complement of  $\Theta$  including the interference steering vectors and their linear combinations by the means

of the following constraint [70, 71]:

$$\hat{\mathbf{a}}^H \tilde{\mathbf{C}} \hat{\mathbf{a}} \leq \Delta_0, \quad (12.119)$$

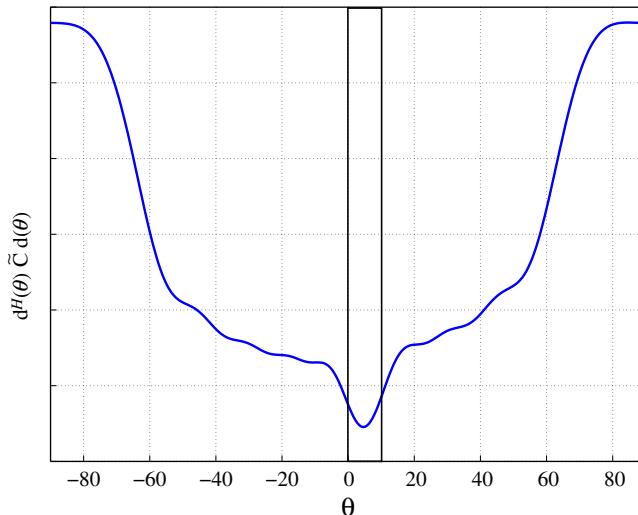
where  $\Delta_0$  is a uniquely selected value for a given angular sector  $\Theta$ , that is,

$$\Delta_0 \triangleq \max_{\theta \in \Theta} \mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta). \quad (12.120)$$

It is worth stressing that no restrictions/assumptions on the structure of the interferences are needed. Moreover, the interferences do not need to have the same structure as the SOI.

In order to illustrate how the quadratic constraint (12.119) works, let us consider the following example. Consider ULA of 10 omni-directional antenna elements spaced half wavelength apart from each other. Let the range of the SOI angular locations be  $\Theta = [0^\circ, 10^\circ]$ . Figure 12.22 depicts the values of the quadratic term  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta)$  for different angles. The rectangular bar in the figure marks the directions within the angular sector  $\Theta$ . It can be observed from this figure that the term  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta)$  takes the smallest values within the angular sector  $\Theta$  and increases outside of the sector. Therefore, if  $\Delta_0$  is selected to be equal to the maximum value of the term  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta)$  within the angular sector  $\Theta$ , the constraint (12.119) guarantees that the estimate of the desired signal steering vector does not converge to any of the interference steering vectors and their linear combinations. The equality  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta) = \Delta_0$  must occur at one of the edges of  $\Theta$ . However, the value of the quadratic term might be smaller than  $\Delta_0$  at the other edge of  $\Theta$ . Therefore, a possibly larger sector  $\Theta_a \geq \Theta$  has to be defined, at which the equality  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta) = \Delta_0$  holds at both edges.

Although for computing the matrix  $\tilde{\mathbf{C}}$ , the presumed knowledge of the antenna array geometry is used, an inaccurate information about the antenna array geometry is sufficient. It further stresses on the



**FIGURE 12.22**

Values of the term  $\mathbf{d}^H(\theta) \tilde{\mathbf{C}} \mathbf{d}(\theta)$  in the constraint (12.119) for different angles.

robustness of such beamforming design to the imperfect prior information [71]. Taking into account the normalization constraint and the constraint (12.119), the problem of estimating the SOI steering vector based on the knowledge of the sector  $\Theta$  can be formulated as the following optimization problem:

$$\begin{aligned} \min_{\hat{\mathbf{a}}} \quad & \hat{\mathbf{a}}^H \hat{\mathbf{R}}^{-1} \hat{\mathbf{a}} \\ \text{subject to} \quad & \|\hat{\mathbf{a}}\|^2 = M, \\ & \hat{\mathbf{a}}^H \tilde{\mathbf{C}} \hat{\mathbf{a}} \leq \Delta_0. \end{aligned} \quad (12.121)$$

Compared to the other MVDR robust adaptive beamforming methods, which require the knowledge of the presumed steering vector and, thus, the knowledge of the presumed antenna array geometry and propagation media and source characteristics, only imprecise knowledge of the antenna array geometry and approximate knowledge of the angular sector  $\Theta$  are needed for the robust adaptive beamformer (12.121).

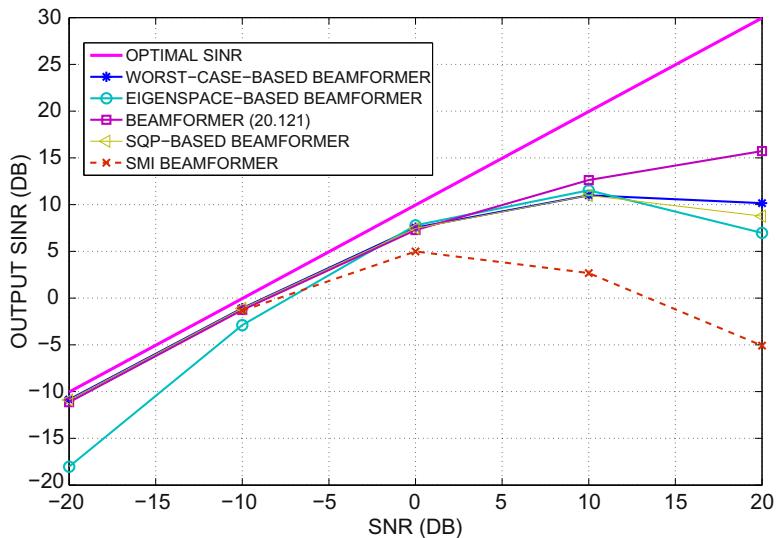
As cooperated to the SQP-based beamformer (12.116), where the constraint  $\mathbf{P}^\perp \hat{\mathbf{a}} = \mathbf{0}$  enforces the estimated steering vector to be a linear combination of  $T$  dominant eigenvectors  $\mathbf{U} \triangleq [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_T]$ , the steering vector in (12.121) is not restricted by such linear combination requirement, while the convergence to any of the interference steering vectors and their linear combinations is avoided by the means of the constraint (12.119). As a result, the beamformer (12.121) has more degrees of freedom compared to the SQP-based beamformer. Thus, it is expected that it outperform the latter one. Finally, due to the non-convex equality constraint, the problem (12.121) is non-convex and NP-hard in general. The efficient polynomial-time solution to this problem is developed in [71] based on the semi-definite programming relaxation theory [57, 72, 73].

### 3.12.4.17 Comparison by simulation

To compare a number of aforementioned MVDR robust adaptive beamforming methods based on the unified approach, the following example is considered. A ULA of 10 omni-directional sensors with the inter-element spacing of half wavelength is used. Additive noise in antenna elements is modeled as spatially and temporally independent complex Gaussian noise with zero mean and unit variance. Two interfering sources are assumed to impinge on the antenna array from the directions  $30^\circ$  and  $50^\circ$ , while the presumed direction towards the SOI is assumed to be  $3^\circ$ . The INR equals 30 dB and the desired signal is always present in the training data.

The robust adaptive beamforming (12.121) is compared with the eigenspace-based, the worst-case-based, the SQP-based, and the LSMI robust adaptive beamforming techniques. For the beamformer (12.121) and the SQP-based one, the angular sector of interest  $\Theta$  is assumed to be  $\Theta = [\theta_p - 5^\circ, \theta_p + 5^\circ]$ , where  $\theta_p$  is the presumed DOA of the SOI. The difference between the presumed and actual positions of each antenna element is modeled as a uniform random variable distributed in the interval  $[-0.05, 0.05]$  measured in wavelength. In addition to the antenna element displacements, the signal steering vector is distorted by wave propagation effects in an inhomogeneous medium. Independent-increment phase distortions are accumulated by the components of the presumed steering vector. It is assumed that the phase increments remain fixed in each simulation run.

Figure 12.23 depict the output SINR performance of the aforementioned robust adaptive beamforming techniques tested versus the SNR for fixed training data size  $K = 30$ . As it can be observed from

**FIGURE 12.23**

Output SINR versus SNR for training data size of  $K = 30$  and INR = 30 dB for the case of perturbations in antenna array geometry.

the figures, the beamformer (12.121) has a better performance even if there is an error in the knowledge of the antenna array geometry.

### 3.12.4.18 Robust adaptive beamforming for the general-rank signal model

Robust adaptive beamforming techniques for general-rank signal model address the situation when the desired signal covariance matrix  $\mathbf{R}_s$  is not known precisely as well as the sample estimate of the data covariance matrix (12.5) is inaccurate because of small sample size.

In order to provide robustness against the norm-bounded mismatches  $\|\Delta_1\| \leq \epsilon$  and  $\|\Delta_2\| \leq \gamma$  (where  $\epsilon$  and  $\gamma$  are some preselected bounds) in the SOI and data sample covariance matrices, respectively, the following worst-case-based robust adaptive beamformer has been derived [74, 75]

$$\mathbf{w} = \mathcal{P}\{(\widehat{\mathbf{R}} + \gamma \mathbf{I})^{-1}(\mathbf{R}_s - \epsilon \mathbf{I})\}. \quad (12.122)$$

Although it is a simple closed-form solution, it is overly conservative due to the fact that the negatively diagonally loaded signal covariance matrix  $\mathbf{R}_s - \epsilon \mathbf{I}$  can be indefinite. A less conservative robust adaptive beamforming problem formulation, which enforces the matrix  $\mathbf{R}_s + \Delta_1$  to be positive semi-definite has been considered in [76]. Defining  $\mathbf{R}_s = \mathbf{Q}^H \mathbf{Q}$ , which is for example the Cholesky decomposition, the corresponding robust adaptive beamforming problem for a norm bounded-mismatch  $\|\Delta\| \leq \eta$  (where

$\eta$  is some value found based on the bound value  $\epsilon$ ) to the matrix  $\mathbf{Q}$  is given as [76]

$$\begin{aligned} & \min_{\mathbf{w}} \max_{\|\Delta_2\| \leq \gamma} \mathbf{w}^H (\widehat{\mathbf{R}} + \Delta_2) \mathbf{w} \\ & \text{subject to } \min_{\|\Delta\| \leq \eta} \mathbf{w}^H (\mathbf{Q} + \Delta)^H (\mathbf{Q} + \Delta) \mathbf{w} \geq 1. \end{aligned} \quad (12.123)$$

If the mismatch of the signal covariance matrix is small enough, the optimization problem (12.123) can be equivalently recast as

$$\min_{\mathbf{w}} \mathbf{w}^H (\widehat{\mathbf{R}} + \gamma \mathbf{I}) \mathbf{w} \quad \text{subject to } \|\mathbf{Q}\mathbf{w}\| - \eta \|\mathbf{w}\| \geq 1. \quad (12.124)$$

Due to the non-convex (difference-of-convex functions (DC)) constraint, the problem (12.124) is non-convex. Although the DC programming problems are believed to be NP-hard in general, the problem (12.124) is shown to have very efficient polynomial-time solution [77].

### 3.12.4.19 Wideband robust adaptive beamforming

In the wideband case (see Figure 12.8), the SOI components at different frequencies are typically not perfectly phased-aligned by the presteering delays because of multiple practical imperfections. The reasons for imperfections are accentually the same as in the narrowband case with an addition of more error sources such as the presteering delay quantization effects. Therefore, there are errors that can be modeled in terms of the phase error vector  $\delta(f)$  that is the function of the frequency  $f$ . Then the actual components of the SOI arriving from DOA  $\theta_s$  after the presteering delay filter are [78]

$$\mathbf{B}(f)\mathbf{a}(f, \theta_s) = e^{j\pi f \varsigma} \mathbf{1}_M + \delta(f), \quad \forall f \in [f_l, f_u] \quad (12.125)$$

instead of (12.13) in the case of no mismatch. Here  $\varsigma$  is a common time delay at each of the  $M$  sensors and  $f_l$  is the minimum frequency of the SOI.

Defining the mismatch set that contains all possible phase error vectors at the frequency  $f$  as  $\mathcal{A}_\varepsilon(f) \triangleq \{\delta(f) \in \mathbb{C}^M | \|\delta(f)\| \leq \varepsilon(f)\}$ , the wideband robust adaptive beamforming problem can be written as

$$\min_{\delta(f) \in \mathcal{A}_\varepsilon(f)} |H(f, \theta_s)| \geq 1 \quad \forall f \in [f_l, f_u]. \quad (12.126)$$

Using (12.15) and (12.125), the array response towards DOA  $\theta_s$  can be written as [78]

$$H(f, \theta_s) = e^{j\pi f \varsigma} \mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f) + \mathbf{w}^T \mathbf{Q}(f) \delta(f), \quad (12.127)$$

where  $\mathbf{Q}(f) \triangleq \mathbf{d}(f) \otimes \mathbf{I}_M$  is  $MP \times M$  matrix.

Using the triangular and then Cauchy-Schwarz inequalities, the magnitude of the lower bound for the array response (12.127) can be found as

$$\begin{aligned} |H(f, \theta_s)| &= |e^{j\pi f \varsigma} \mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f) + \mathbf{w}^T \mathbf{Q}(f) \delta(f)| \\ &\geq |\mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f)| - |\mathbf{w}^T \mathbf{Q}(f) \delta(f)| \\ &\geq |\mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f)| - \varepsilon(f) \|\mathbf{Q}^T(f) \mathbf{w}\|. \end{aligned} \quad (12.128)$$

Finally, using the lower bound (12.128) for the constraint  $|H(f, \theta_s)| \geq 1$  in (12.126) and imposing a linear phase constraint on each of the  $M$  FIR filters of the array processor Figure 12.8, the optimization problem (12.126) can be reformulated as the following output power minimization problem:

$$\begin{aligned} & \min_{\mathbf{w}} \quad \mathbf{w}^T \mathbf{R} \mathbf{w} \\ \text{subject to} \quad & |\mathbf{w}^T \mathbf{C}_0 \mathbf{d}(f)| - \varepsilon(f) \|\mathbf{Q}^T(f) \mathbf{w}\| \geq 1, \quad f \in [f_l, f_u], \\ & w_{m,l} = w_{m,P-l+1}, \quad \forall m \in \mathbb{Z}_1^M, \quad l \in \mathbb{Z}_1^{P_c-1}, \end{aligned} \quad (12.129)$$

where  $\mathbf{R}$  is the covariance matrix of the stacked snapshot vectors,  $P_c = (P + 1)/2$ , and  $\mathbb{Z}_i^j$  denotes the ring of integers from  $i$  to  $j$ . The last constraint in the optimization problem (12.129) ensures the linear phase at each of the  $M$  FIR filters and it provides additional robustness against presteering errors [78]. The problem (12.129) is non-convex, but it can be reformulated to a convex problem that can be solved efficiently [78]. The disadvantage is, however, that the constraint on the magnitude of the array response is strengthened by using the triangular and Cauchy-Schwarz inequalities (see (12.128)). More sophisticated wideband robust adaptive beamforming designs can be also found in [79, 80].

### 3.12.4.20 Summary

The applicability of different robust adaptive beamforming techniques is mainly defined by the corresponding notions of robustness used for designing a particular method and by the required prior information needed to run a method. A majority of the existing robust adaptive beamforming techniques such as the above mentioned techniques as eigenspace-based, worst-case-based, doubly constrained, probabilistically constrained techniques as well as the eigenvalue beamforming using multi-rank MVDR beamformer and their various modifications require the knowledge of the presumed steering vector. In turn, the availability of this knowledge implies that the source and propagation media characteristics as well as antenna geometry are known with a certain accuracy. Each method also requires some additional information. For example, the eigenspace-based beamformer needs to know the number of interferences, which may be a challenging practical problem. The worst-case-based and the doubly constrained beamforming techniques need to know the upper-bound to the norm of the steering vector mismatch, which is fortunately irrelevant to specific causes of mismatch and which is practically easy to guess or estimate in a particular application. It is important that the performance of these methods is not very sensitive to the over- or under-estimation of upper-bound to the norm of the steering vector mismatch that makes these approaches practically attractive and widely applicable. The probabilistically constrained robust adaptive beamforming enables to quantify the upper-bound to the norm of the steering vector mismatch in terms of the variance of the steering vector estimation and the practically tolerable outage probability that the distortionless response constraint is satisfied. This may be an advantage in a number of applications especially when the variance of the steering vector/channel estimation is already the existing information that does not require any additional efforts to obtain. However, the least restrictive in terms of the required prior information is the robust adaptive beamforming technique based on steering vector estimation with as little as possible prior information. It does not need the information about the presumed steering vector, but only needs a very approximate knowledge of the array geometry, which is easy to have even in such challenging applications as sonar. Similarly, it does not need any nearly accurate estimates of the source characteristics, but rather needs

only the very approximate knowledge of a sector where the source of interest is located. In this respect, the latter technique can be most appropriately called “robust.” Moreover, it outperforms other existing technique in terms of the beamformer output SINR. However, the complexity of the latter technique is equivalent to the complexity of solving SDP problem that may be higher than the complexity of matrix inversion, i.e., the complexity of SMI and LSMI beamformers, and nearly the complexity of the worst-case-based and other aforementioned beamforming techniques. Finally, the notion of robustness used by the robust adaptive beamforming technique based on steering vector estimation with as little as possible prior information is the most general that makes its applicability essentially unlimited (limited only by the source model as the source is assumed to be narrowband). The extension of this technique to the wideband case is the topic of future promising research.

The field of robust adaptive beamforming is an actively developing research field which is strongly connected to the progress in optimization theory. While the notion of robustness used in [71] is the most general as mentioned above, new methods have been actively developing within the other approaches to robust adaptive beamforming design with more specific notions of robustness. As an example, within the worst-case-based approach, it has been recently noticed in [81] that although the above described worst-case-based beamforming designs can be formulated as 1D covariance fitting problems (as explained in this section), these beamformers lead to inherently non-optimum results in the presence of interferers. To mitigate the detrimental effect of interferers, the 1D covariance fitting approach is extended to multi-dimensional (MD) covariance fitting in [81].

The adaptive and robust beamforming problem is originated from array processing, but it has found a number of very fruitful applications in other actively developing fields which successfully applied the ideas and designs developed first in array processing framework. To mention just a few of such applications we refer the reader to such wireless communication problems as downlink beamforming in cellular wireless networks [82], code-division multiple-access (CDMA) multiuser detection [83–85], linear receiver design for multi-access space-time codded systems [86,87], multicast beamforming [15,88], secondary multicast beamforming for spectrum sharing in cognitive radio systems [16], relay network beamforming [89], etc. For more details on such applications see [90] and other sections of the Encyclopedia.

---

## Acknowledgments

The author of this chapter would like to acknowledge the input of Prof. Alex B. Gershman (formerly of Darmstadt University of Technology, Germany). While still alive, Prof. Gershman has shared with the author some materials on adaptive beamforming including a number of figures used in this chapter. Without these figures, the presentation would be significantly less illustrative.

*Relevant Theory:* Array Signal Processing

See this volume, [Chapter 13](#), Broadband Beamforming and Optimisation

See this volume, [Chapter 15](#), Subspace Methods and Exploitation of Special Array Structures

See this volume, [Chapter 17](#), DOA Estimation of Nonstationary Signals

See this volume, [Chapter 19](#), Array Processing in the Face of Nonidealities

See this volume, [Chapter 20](#), Applications of Array Signal Processing

---

## References

- [1] B. Widrow, P.E. Mantey, J.L. Griffiths, B.B. Goode, Adaptive antenna systems, Proc. IEEE 55 (1967) 2143–2159.
- [2] R.A. Monzingo, T.W. Miller, Introduction to Adaptive Arrays, Wiley, NY, 1980.
- [3] H.L. Van Trees, Optimum Array Processing, Wiley, NY, 2002.
- [4] M. Viberg, Introduction to array processing, Signal Processing Encyclopedia, 2013.
- [5] S. Haykin, J. Litva, T. Shepherd (Eds.), Radar Array Processing, Springer-Verlag, 1992.
- [6] A. Farina, Antenna-Based Signal Processing Techniques for Radar Systems, Artech House, Norwood, MA, 1992.
- [7] R.J. Vaccaro (Ed.), The Past, Present, and the Future of Underwater Acoustic Signal Processing, IEEE Signal Process. Mag. 15 (1998) 21–51.
- [8] Y. Kameda, J. Ohga, Adaptive microphone-array system for noise reduction, IEEE Trans. Acoust. Speech Signal Process. 34 (1986) 1391–1400.
- [9] S.W. Ellingson, G.A. Hampson, A subspace-tracking approach to interference nulling for phased array-based radio telescopes, IEEE Trans. Antennas Propag. 50 (2002) 25–30.
- [10] A. Leshem, J. Christou, B.D. Jeffs, E. Kuruoglu, A.J. van der Veen (Eds.), Issue on Signal Processing for Space Research and Astronomy, IEEE J. Sel. Top. Signal Process. 2 (5) (2008).
- [11] K. Sekihara, Performance of an MEG adaptive beamformer source reconstruction technique in the presence of adaptive low-rank interference, IEEE Trans. Biomed. Eng. 51 (2004) 90–99.
- [12] K.-M. Chen, D. Misra, H. Wang, H.-R. Chuang, E. Postow, An X-band microwave life-detection system, IEEE Trans. Biomed. Eng. 33 (1986) 697–701.
- [13] T.S. Rapaport (Ed.), Smart Antennas: Adaptive Arrays, Algorithms, and Wireless Position Location, IEEE Press, 1998.
- [14] A.B. Gershman, N.D. Sidiropoulos (Eds.), Space-Time Processing for MIMO Communications, John Wiley and Sons, 2005.
- [15] N.D. Sidiropoulos, T.N. Davidson, Z.-Q. Luo, Transmit beamforming for physical-layer multicasting, IEEE Trans. Signal Process. 54 (2006) 2239–2251.
- [16] K.T. Phan, S.A. Vorobyov, N.D. Sidiropoulos, C. Tellambura, Spectrum sharing in wireless networks via QoS-aware secondary multicast beamforming, IEEE Trans. Signal Process. 57 (2009) 2323–2335.
- [17] H. Cox, R.M. Zeskind, M.H. Owen, Robust adaptive beamforming, IEEE Trans. Acoust. Speech Signal Process. 35 (1987) 1365–1376.
- [18] D.D. Feldman, L.J. Griffiths, A projection approach to robust adaptive beamforming, IEEE Trans. Signal Process. 42 (1994) 867–876.
- [19] J.R. Guerci, Theory and application of covariance matrix tapers to robust adaptive beamforming, IEEE Trans. Signal Process. 47 (2000) 977–985.
- [20] O. Leodit, M. Wolf, Improved estimation of the covariance matrix of the stock returns with an application to portfolio selection, J. Empirical Finance 10 (2003) 603–621.
- [21] K.I. Pedersen, P.E. Mogensen, B.H. Fleury, A stochastic model of the temporal and azimuthal dispersion seen at the base station in outdoor propagation environments, IEEE Trans. Veh. Technol. 49 (2000) 437–447.
- [22] O. Besson, P. Stoica, Decoupled estimation of DOA and angular spread for a spatially distributed source, IEEE Trans. Signal Process. 48 (2000) 1872–1882.
- [23] H. Cox, Line array performance when the signal coherence is spatially dependent, J. Acoust. Soc. Am. 54 (1973) 1743–1746.

- [24] A.B. Gershman, V.I. Turchin, V.A. Zverev, Experimental results of localization of moving underwater signal by adaptive beamforming, *IEEE Trans. Signal Process.* 43 (1995) 2249–2257.
- [25] A.B. Gershman, C.F. Mecklenbräuker, J.F. Boehme, Matrix fitting approach to direction of arrival estimation with imperfect spatial coherence of wavefronts, *IEEE Trans. Signal Process.* 45 (1997) 1894–1899.
- [26] S.E. Nordholm, H.H. Dam, C.C. Lai, E.A. Lehmann, Broadband Beamforming and Optimization, *Signal Process. Encyclopedia*, 2013.
- [27] O.L. Frost, An algorithm for linearly constrained adaptive array processing, *Proc. IEEE* 60 (1972) 926–935.
- [28] B. Widrow, S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1985.
- [29] I.S. Reed, J.D. Mallett, L.E. Brennan, Rapid convergence rate in adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 10 (1974) 853–863.
- [30] E.K. Hung, R.M. Turner, A fast beamforming algorithm for large arrays, *IEEE Trans. Aerosp. Electron. Syst.* 19 (1983) 598–607.
- [31] G.A. Fabrizio, A.B. Gershman, M.D. Turley, Robust adaptive beamforming for HF surface wave over-the-horizon radar, *IEEE Trans. Aerosp. Electron. Syst.* 40 (2004) 510–625.
- [32] P. Forster, G. Vezzosi, Application of spheroidal sequences to array processing, in: *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Dallas, TX, May 1987, pp. 2268–2271.
- [33] D. Slepian, Prolate spheroidal wave functions, Fourier analysis, and uncertainty. V—the discrete case, *Bell Syst. Tech. J.* (1978) 1371–1430.
- [34] T.J. Cornwell, A novel principle for optimization of the instantaneous Fourier plane coverage of correlation arrays, *IEEE Trans. Antennas Propag.* 36 (8) (1988) 1165–1167.
- [35] Z. Wang, J. Li, P. Stoica, T. Nishida, M. Sheplak, Constantbeamwidth and constant-powerwidth wideband robust Capon beamformers for acoustic imaging, *J. Acoust. Soc. Am.* 116 (3) (2004) 1621–1631.
- [36] L.C. Godara, The effect of phase-shift errors on the performance of an antenna-array beamformer, *IEEE J. Ocean. Eng.* 10 (1985) 278–284.
- [37] J.W. Kim, C.K. Un, An adaptive array robust to beam pointing error, *IEEE Trans. Signal Process.* 40 (1992) 1582–1584.
- [38] N.K. Jablon, Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections, *IEEE Trans. Antennas Propag.* 34 (1986) 996–1012.
- [39] A.B. Gershman, V.I. Turchin, V.A. Zverev, Experimental results of localization of moving underwater signal by adaptive beamforming, *IEEE Trans. Signal Process.* 43 (1995) 2249–2257.
- [40] Y.J. Hong, C.C. Yeh, D.R. Ucci, The effect of a finite-distance signal source on a far-field steering Applebaum array: two dimensional array case, *IEEE Trans. Antennas Propag.* 36 (1988) 468–475.
- [41] J. Goldberg, H. Messer, Inherent limitations in the localization of a coherently scattered source, *IEEE Trans. Signal Process.* 46 (1998) 3441–3444.
- [42] O. Besson, P. Stoica, Decoupled estimation of DOA and angular spread for a spatially distributed source, *IEEE Trans. Signal Process.* 48 (2000) 1872–1882.
- [43] D. Astely, B. Ottersten, The effects of local scattering on direction of arrival estimation with MUSIC, *IEEE Trans. Signal Process.* 47 (1999) 3220–3234.
- [44] A.B. Gershman, Robust adaptive beamforming in sensor arrays, *Int. J. Electron. Commun.* 53 (1999) 305–314 (invited paper).
- [45] Y.I. Abramovich, Controlled method for adaptive optimization of filters using the criterion of maximum SNR, *Radio Eng. Electron. Phys.* 26 (1981) 87–95.
- [46] B.D. Carlson, Covariance matrix estimation errors and diagonal loading in adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 24 (7) (1988) 397–401.
- [47] W.F. Gabriel (Ed.), Special issue on adaptive antennas, *IEEE Trans. Antennas Propag.* 24 (1976).
- [48] I. Claesson, S. Nordholm, A spatial filtering approach to robust adaptive beamforming, *IEEE Trans. Antennas Propag.* 40 (1992) 1093–1096.

- [49] T.J. Shan, T. Kailath, Adaptive beamforming for coherent signals and interference, *IEEE Trans. Acoust. Speech Signal Process.* 33 (1985) 527–536.
- [50] M.D. Zoltowski, On the performance of the MVDR beamformer in the presence of correlated interference, *IEEE Trans. Acoust. Speech Signal Process.* 36 (1988) 945–947.
- [51] A.B. Gershman, V.T. Ermolaev, Optimal subarray size for spatial smoothing, *IEEE Signal Process. Lett.* 2 (1995) 28–30.
- [52] R.T. Williams, S. Prasad, A.K. Mahalanabis, L. Sibul, An improved spatial smoothing technique for bearing estimation in a multipath environment, *IEEE Trans. Acoust. Speech Signal Process.* 36 (1988) 425–432.
- [53] K. Takao, N. Kikuma, T. Yano, Toeplitzization of correlation matrix in multipath environment, in: Proc. ICASSP'86, Tokyo, Japan, 1986, pp. 1873–1876.
- [54] R.J. Mallioux, Covariance matrix augmentation to produce adaptive array pattern throughs, *Electron. Lett.* 31 (1995) 771–772.
- [55] A.B. Gershman, U. Nickel, J.F. Boehme, Adaptive beamforming algorithms with robustness against jammer motion, *IEEE Trans. Signal Process.* 45 (1997) 1878–1885.
- [56] Z.-Q. Luo, W. Yu, An introduction to convex optimization for communications and signal processing, *IEEE J. Sel. Areas Commun.* 24 (2006) 20–34.
- [57] Z.-Q. Luo, W.-K. Ma, A.M.-C. So, Y. Ye, S. Zhang, Semidefinite relaxation of quadratic optimization problems, *IEEE Signal Process. Mag.* 27 (3) (2010) 20–34.
- [58] L. Chang, C.C. Yeh, Performance of DMI and eigenspace-based beamformers, *IEEE Trans. Antennas Propag.* 40 (1992) 1336–1347.
- [59] J.K. Thomas, L.L. Scharf, D.W. Tufts, The probability of a subspace swap in the SVD, *IEEE Trans. Signal Process.* 43 (1995) 730–736.
- [60] S.A. Vorobyov, A.B. Gershman, Z.-Q. Luo, Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem, *IEEE Trans. Signal Process.* 51 (2003) 313–324.
- [61] R.G. Lorenz, S.P. Boyd, Robust minimum variance beamforming, *IEEE Trans. Signal Process.* 53 (2005) 1684–1696.
- [62] J. Li, P. Stoica, Z. Wang, On robust Capon beamforming and diagonal loading, *IEEE Trans. Signal Process.* 51 (2003) 1702–1715.
- [63] S.A. Vorobyov, A.B. Gershman, Z.-Q. Luo, N. Ma, Adaptive beamforming with joint robustness against mismatched signal steering vector and interference nonstationarity, *IEEE Signal Process. Lett.* 11 (2004) 108–111.
- [64] J. Li, P. Stoica, and Z. Wang, Doubly constrained robust capon beamformer, *IEEE Trans. Signal Process.* 52 (2004) 2407–2423.
- [65] A. Beck, Y. Eldar, Doubly constrained robust Capon beamformer with ellipsoidal uncertainty set, *IEEE Trans. Signal Process.* 55 (2007) 753–758.
- [66] S.A. Vorobyov, H. Chen, A.B. Gershman, On the relationship between robust minimum variance beamformers with probabilistic and worst-case distortionless response constraints, *IEEE Trans. Signal Process.* 56 (2008) 5719–5724.
- [67] A. Hassanien, S.A. Vorobyov, K.M. Wong, Robust adaptive beamforming using sequential programming: an iterative solution to the mismatch problem, *IEEE Signal Process. Lett.* 15 (2008) 733–736.
- [68] L. Lei, J.P. Lie, A.B. Gershman, C.M.S. See, Robust adaptive beamforming in partly calibrated sparse sensor arrays, *IEEE Trans. Signal Process.* 58 (2010) 1661–1667.
- [69] A. Pezeshki, B.D. Van Veen, L.L. Scharf, H. Cox, M. Lundberg, Eigenvalue beamforming using a multi-rank MVDR beamformer and subspace selection, *IEEE Trans. Signal Process.* 56 (5) (2008) 1954–1967.
- [70] A. Khabbazibasmenj, S.A. Vorobyov, A. Hassanien, Robust adaptive beamforming via estimating steering vector based on semidefinite relaxation, in: Proc. 44th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, California, November 2010, pp. 1102–1106.

- [71] A. Khabbazibasmenj, S.A. Vorobyov, A. Hassani, Robust adaptive beamforming based on steering vector estimation with as little as possible prior information, *IEEE Trans. Signal Process.* 60 (2012) 2974–2987.
- [72] Y.S. Nesterov, Semidefinite relaxation and nonconvex quadratic optimization, *Optim. Methods Softw.* 9 (1–3) (1998) 141–160.
- [73] S. Zhang, Quadratic maximization and semidefinite relaxation, *Math. Program. A* 87 (2000) 453–465.
- [74] S. Shahbazpanahi, A.B. Gershman, Z.-Q. Luo, K.M. Wong, Robust adaptive beamforming for general-rank signal models, *IEEE Trans. Signal Process.* 51 (2003) 2257–2269.
- [75] A.B. Gershman, Z.-Q. Luo, S. Shahbazpanahi, Robust adaptive beamforming based on worst-case performance optimization, in: P. Stoica, J. Li (Eds.), *Robust Adaptive Beamforming*, Wiley, Hoboken, NJ, 2006, pp. 49–89.
- [76] H.H. Chen, A.B. Gershman, Robust adaptive beamforming for general-rank signal models with positive semi-definite constraints, in: Proc. IEEE Int. Conf. Acoustic, Speech, and Signal Processing, Las Vegas, USA, April 2008, pp. 2341–2344.
- [77] A. Khabbazibasmenj, S.A. Vorobyov, A computationally efficient robust adaptive beamforming for general-rank signal model with positive semi-definite constraint, in: Proc. Inter. Workshop Comp. Advances in Multi-Sensor Adaptive Processing, San Juan, Puerto Rico, December 2011, pp. 185–188.
- [78] A. El-Keyi, T. Kirubarajan, A.B. Gershman, Adaptive wideband beamforming with robustness against presteering errors, in: Proc. IEEE Workshop on Sensor Arrays and Multi-Channel Processing, Waltham, MA, USA, July 2006, pp. 11–15.
- [79] M. Ruebsamen, A. El-Keyi, A.B. Gershman, T. Kirubarajan, Robust broadband adaptive beamforming using convex optimization, in: D.P. Palomar, Y.C. Eldar (Eds.), *Convex Optimization in Signal Processing and Communications*, Cambridge University Press, 2010, pp. 315–339 (Chapter 9).
- [80] M. Ruebsamen, Advanced Direction-of-Arrival Estimation and Beamforming Techniques for Multiple Antenna Systems, Ph.D. Thesis, Darmstadt University of Technology, 2011 (Chapter 5).
- [81] M. Ruebsamen, A.B. Gershman, Robust adaptive beamforming using multi0dimentions covariance fitting, *IEEE Trans. Signal Process.* 60 (2012) 740–753.
- [82] M. Bengtsson, B. Ottersten, Optimal and suboptimal transmit beamforming, in: L. Godara (Ed.), *Handbook of Antennas in Wireless Communications*, CRC Press: Boca Raton, FL, 2001 (Chapter 18).
- [83] S. Verdu, *Multiuser Detection*, Cambridge University Press, Cambridge, UK, 1998.
- [84] K. Zarifi, S. Shahbazpanahi, A.B. Gershman, Z.-Q. Luo, Robust blind multiuser detection based on the worst-case performance optimization of the MMSE receiver, *IEEE Trans. Signal Process.* 53 (2005) 295–305.
- [85] S.A. Vorobyov, Robust CDMA multiuser detectors: probability-constrained versus the worst-case based design, *IEEE Signal Process. Lett.* 15 (2008) 273–276.
- [86] S. Shahbazpanahi, M. Beheshti, A.B. Gershman, M. Gharavi-Alkhansari, K.M. Wong, Minimum variance linear receivers for multi-access MIMO wireless systems with space-time block coding, *IEEE Trans. Signal Process.* 52 (2004) 3306–3313.
- [87] Y. Rong, S.A. Vorobyov, A.B. Gershman, Robust linear receivers for multi-access space-time block coded MIMO systems: a probabilistically constrained approach, *IEEE J. Sel. Areas Commun.* 24 (2006) 1560–1570.
- [88] E. Karipidis, N.D. Sidiropoulos, Z.-Q. Luo, Far-field multicast beamforming for uniform linear antenna arrays, *IEEE Trans. Signal Process.* 55 (2007) 4916–4927.
- [89] V. Havary-Nassab, S. Shahbazpanahi, A. Grami, Z.-Q. Luo, Distributed beamforming for relay networks based on second-order statistics of the channel state information, *IEEE Trans. Signal Process.* 56 (2008) 4306–4316.
- [90] A.B. Gershman, N.D. Sidiropoulos, S. Shahbazpanahi, M. Bengtsson, B. Ottersten, Convex optimization-based beamforming, *IEEE Signal Process. Mag.* 27(3) (2010) 62–75.

# Broadband Beamforming and Optimization

# 13

Sven E. Nordholm<sup>\*</sup>, Hai H. Dam<sup>†</sup>, Chiong C. Lai<sup>\*</sup>, and Eric A. Lehmann<sup>‡</sup>

<sup>\*</sup>Department of Electrical and Computer Engineering, Curtin University of Technology, Perth, WA, Australia

<sup>†</sup>Department of Mathematics and Statistics, Curtin University of Technology, Perth, WA, Australia

<sup>‡</sup>Mathematics, Informatics and Statistics, CSIRO, Wembley, WA, Australia

## 3.13.1 Introduction

Broadband beamformers [1–4] have been studied extensively over many years due to their wide applications in many areas such as radar, geology, sonar, biomedicine, radio astronomy, speech acquisition and acoustics. Broadband beamformers, in contrast to single point sensor observations of a signal, employ spatially distributed sensors to gain information on the spatial properties of the incoming waves. These spatially separated sensor observations provide extra information about signal properties and noise characteristics. Using an array of sensors allows the use of spatial domain information as well as time domain information. Accordingly, multidimensional filters can be designed such that a signal of a certain beam-width and bandwidth can be extracted while signals that are not overlapping spatially or in frequency can be suppressed. This multidimensional filter is usually referred to as a broadband beamformer. The beamformer response processes the data such that the gain for each frequency becomes a function of the direction of the incoming wave. This provides an increased gain for the spatially selected incoming wave if it falls within the beam direction and a suppression if the wave falls outside the beam direction. For narrowband beamformers this improvement is usually defined as an array gain. It is also common to provide an array pattern or beampattern which is a function of the angular direction of the incoming wave. In most cases, one is mainly concerned with the array gain in the azimuthal plane and thus the plot becomes two-dimensional. For broadband beamformers, the frequency dependent array pattern is integrated over the bandwidth of interest and is presented as an integrated array pattern plot. As an alternative, a frequency dependent array pattern can be presented in form of a three-dimensional plot over angle and frequency. Thus the beam direction is a function of angle and frequency.

An effective utilization of spatial and temporal processing will lead to an efficient processing solution. There are multiple aspects that need to be considered in this design; the filter weight design, the spatial distribution of the elements, the required spatial selectivity or beamwidth, and the frequency range and propagation speed.

The actual solutions and requirements are application dependent. For microphone arrays, the beamformer design is usually made over many octaves. In high end audio for high quality recordings, the bandwidth can be 100 Hz to 20 kHz. But most applications have a more limited bandwidth such as 300–3400 Hz for audio, and 200–7000 Hz for high quality audio conferencing. In these types of speech

applications one of the significant degradations is room reverberation. To resolve those problems, the array needs to grow in size and number of elements. Another problem with microphone arrays is that the free field assumption usually imposed is not valid. Often the microphone arrays should be operating both in near field and far field. Far field is considered when the distance from the center of the array to the source  $r_s$  is larger than  $\frac{2L^2}{\lambda}$ , where  $L$  is the largest aperture of the array and  $\lambda$  is the shortest wavelength of signals considered.

In other applications such as sonar, the necessary bandwidth is also a few octaves. Radar is usually more narrowband and some modulation technique such as a chirped FM signal is used to create the signal. As the demand for higher resolution increases, higher bandwidth is needed. In radio physics, very broadband antennas are used in combination with narrow steerable beams. The common denominator in the applications is the need for joint space and time-frequency analysis. This imposes design requirements on broadband beamforming, such as the capability to have a similar beamwidth over a wide range of frequencies given an array constellation and also low linear distortion of the signal of interest (SOI). There are usually physical constraints on the array size and this constrains the beamwidth for large wavelengths.

In general terms, the problem that the broadband beamformer shall solve can be defined as follows.

**Problem statement:** *The broadband beamformer shall operate upon the incoming waveforms in such a way that Signal(s) of Interest (SOI), transmitted from point(s) in space, shall be extracted while all other signals shall be suppressed.*

This task is of course not possible in all practical scenarios, but it is this problem formulation that will be discussed in this chapter. Some of the questions that will be discussed are: What methods are available for extraction? What are the limitations for the processing? What *a priori* information is available? How well do available techniques work in realistic environments?

Numerous techniques have been suggested for the design of broadband beamformers. The main challenge in the design is to be able to handle the wide frequency range required. The techniques that will be discussed here are optimized designs, which are non-data-dependent weight design methods, and optimum beamformers, where the weight design is dependent on statistical properties of the presented data.

An optimized beamformer is a design where the weights are designed according to a propagation model, array geometry and a design specification. Common ways to do the design is either to start from a sampled domain where the sensor elements are considered as points in some given spatial pattern (linear, circular, etc.) or from the wave field domain where the receiver is a volume of a certain shape and dimension [4,5]. For the sampled domain design, the problem can be considered as a multidimensional filter design problem with an objective function and specification. The design domain is over space and frequency. In the case of a wave field domain design, the problem is considered over a volume aperture using aperture theory. For certain shapes, such as a sphere or a circle, the wave field solution can naturally be expressed in spherical harmonics which results in modal beamformer design. This allows separation of spatial processing and temporal processing such that beam steering becomes simple. In theory, this gives an efficient means to design a beamformer of relatively low complexity and high quality, but requires sensors of very high quality. There exist commercial microphone arrays available using this wave domain technique for spherical arrays. Optimized beamformers are designed using criteria such as Least Squares (LS), Weighted Least Squares (WLS), Total Least Squares (TLS), or Chebyshev (min-max criteria). These criteria will be discussed in detail in a later section including some design examples.

An optimum beamformer is a beamformer with weights designed based on data and the statistical properties of those data. Since the receiving array only has access to the received signal and not the original signal, it is blind to the actual SOI. To accommodate for this, the design criteria need to include some form of constraint to maintain a desired direction and signal. The challenge when using optimum beamformers is to maintain the signal integrity or to have low signal distortion and thus no source cancellation. Significant work has been done to find ways to maintain signal integrity while still providing capability to suppress noise and directional disturbance. The task for the optimum broadband beamformer is usually twofold, namely to suppress directional disturbing signals, usually called jammers in the radar literature, and to compensate the convolutional effects stemming from the environment. The most common criteria that are used for optimum broadband beamformers are Minimum Mean Square Error (MMSE) or Signal to Noise Ratio (SNR) with distortion constraints.

Another aspect of broadband beamforming is the capability to track the SOI. Accordingly, the array pattern is steered according to tracking information of a source. Thus the direction of the beamformer needs to be steerable. This is usually the case when the overall system is implemented for real scenarios in all of the above applications. In some cases, several SOI need to be tracked in parallel, which requires multi-source tracking. Then the broadband beamformer needs to implement several beams simultaneously.

The problem outlined above can also be posed as a communication problem. The SOI is filtered through an unknown channel which corresponds to the transmission environment and is also disturbed by directional noise and non-directional noise. The classic techniques for source estimation are Minimum Mean Square Error (MMSE) estimation, or Wiener filter. However for most applications, these techniques cannot be directly applied to this problem since the SOI is not available. The unavailability of the SOI differentiates this problem from the classic communication problem where usually a known pilot signal can be transmitted. However, for active radar and active sonar, partial knowledge of the transmitted waveform is available.

It can also be necessary to consider channel modeling since the medium is usually not the ideal free field or anechoic situation. However, one needs to be aware that any model is just a model, and it is imperative to verify broadband beamforming schemes based on real or at least measured data. The better the design models one has, the fewer iterations are needed between the design phase and implementation phase. Research work is pursued with the aim of being able to predict performance in computer simulations without building the actual broadband beamformer. This would allow the use of these advanced models into the actual design phase of the array.

For speech acquisition in particular, microphone arrays are commonly deployed to reduce the level of localized and ambient noise signals as well as undesired reverberation from a desired direction via beamforming. Then the capability to track speakers is necessary. The beamformer acts as a spatial filter and utilizes the spatial domain as well as the time domain. An effective combination of spatial and temporal processing will lead to a computationally efficient solution.

The radar and sonar situations are different in the sense that often only the detection information is needed and not the actual waveforms. At current time, there is also a research interest to obtain contextual information. In such cases, there is a desire to extract more content out of the received signal than a simple decision on the detection or presence of an object. This research is very active at current time since providing a higher level of contextual analysis of the received signal allows to obtain more information on the object. This is particularly the case for active sonar and active radar, which transmit

a known signal. This known signal is subjected to a channel and reflected on a surface. Based on the received signal, one can form an optimum beamformer such that important information can be extracted. The contextual information can be obtained either by training or modeling.

### 3.13.2 Environment and channel modeling

The operational scenario in applications such as radar, geology, sonar, biomedicine, speech acquisition and acoustics, are that the sensor array receives signals in form of acoustic or electromagnetic waves transmitted from sources that are spatially separated in space. Those waves are generated from either a SOI, or undesired sources usually called jammers. In addition to that, independent noise is incumbent on each element. Incoming waves are considered as noise when they do not have a distinct point of origin or when they are too many to be modeled as individual sources. Whereas SOI and interference signals are directional, noise is usually considered to be non-directive or diffuse.

The theory describing the relationship between a source and a sensor is usually called aperture theory [4]. The theoretical background needed to understand those phenomena is vector calculus, linear system theory and wave theory. Aperture theory describes the relationship between a transmitter and a receiver. The study of air and underwater acoustics is dedicated to the mathematical modeling and measurement of the propagation and scattering of sound waves in fluid and elastic media. Since this is not the topic of this chapter, only very limited background information is presented here so as to explain the relevant concepts of beamforming design.

#### 3.13.2.1 Aperture theory

In the acoustics literature [4], the word aperture is used to describe either a single electro-acoustic transducer or an array of transducers. An active transducer emits acoustic energy and is called a transmitter. An electro-acoustic device converts electric signal to acoustic signal. In a similar manner, a passive transducer is operating as a receiver. As such the electro-acoustic receiver converts acoustic signal to electric signal. These transducers are usually called sensors in modern language. They are commonly modeled as omni-directional point sources. The general theory that connects the electrical signal in a transmitter to a receiver is called aperture theory. It combines linear systems theory and wave theory. Only a few main results are discussed here and their details can be found in [4]. The aperture function  $A_T(\omega, \mathbf{r}_T)$ , which is the complex transducer function in frequency domain for each point on the transducer body, determines the far-field directivity function or beampattern and is given by

$$D_T(\omega, \boldsymbol{\alpha}) = \int_{-\infty}^{\infty} A_T(\omega, \mathbf{r}_T) \exp(i2\pi\boldsymbol{\alpha}^T \mathbf{r}_T) d\mathbf{r}_T, \quad (13.1)$$

where  $\boldsymbol{\alpha} = [\frac{f}{c} \sin \theta \cos \phi, \frac{f}{c} \sin \theta \sin \phi, \frac{f}{c} \cos \theta]$  is the unit vector in directions  $\theta$  and  $\psi$  according to Figure 13.1, and  $(\cdot)^T$  denotes transpose. The integral is taken over the transducer body. It is however more common to work directly on the sampled space assuming ideal point source elements. These elements sample the aperture function  $A_T(\omega, \mathbf{r}_T)$ , so the aperture function can be expressed in terms of those sample values

$$A_T(\omega, \mathbf{r}_T) = \sum_{n=1}^L c_n(\omega) \delta(\mathbf{r}_T - \mathbf{r}_n), \quad (13.2)$$

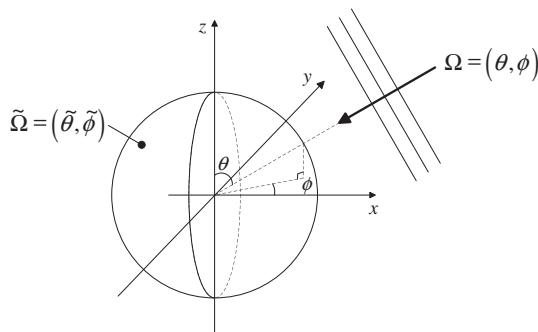
**FIGURE 13.1**

Figure illustrating the choice of 3-D variables.

where  $\mathbf{r}_n$  gives the element positions. Inserting this expression (13.2) in (13.1) yields

$$\begin{aligned} D_T(\omega, \boldsymbol{\alpha}) &= \int_{-\infty}^{\infty} \sum_{n=1}^L c_n(\omega) \delta(\mathbf{r}_T - \mathbf{r}_n) \exp(i2\pi\boldsymbol{\alpha}^T \mathbf{r}_T) d\mathbf{r}_T \\ &= \sum_{n=1}^L c_n(\omega) \exp(i2\pi\boldsymbol{\alpha}^T \mathbf{r}_n) = \mathbf{c}^H(\omega) \mathbf{d}_a(\boldsymbol{\alpha}, r_n). \end{aligned} \quad (13.3)$$

where  $(\cdot)^H$  denotes the Hermitian transpose, i.e., transpose and conjugate.

Thus the beampattern is determined by a vector of complex weights  $\mathbf{c}(\omega)$  and an array response vector  $\mathbf{d}(\boldsymbol{\alpha}, \mathbf{r}_n)$ . The array response vector is defined by the frequency, the unit directivity vector  $\boldsymbol{\alpha}$  and placements of elements. The expression (13.3) gives the far field response in a free field for identical omni-directional elements. This equation can then be used as a design equation for the far field beamformer weight design. For near field designs, a near field model needs to be considered. The Green function for free field can be used to model near field response

$$H_M(\omega, \mathbf{r}_M, \mathbf{r}_T, v) = H_M(\omega, \rho) = -\frac{\exp(-i\frac{\omega}{c}|\mathbf{r}_M - \mathbf{r}_T|)}{4\pi|\mathbf{r}_M - \mathbf{r}_T|} = -\frac{\exp(-i\frac{\omega}{c}\rho)}{4\pi\rho}. \quad (13.4)$$

This expression is valid for time invariant and space invariant settings (unbounded and homogeneous medium). In a more general modeling case, one needs to consider a propagation model and also include reflections.

### 3.13.2.2 Array geometries

An array can be considered as a sampled aperture where the elements are chosen as sample values on a surface of various shape. In the previous section, the far-field directivity pattern (beam pattern) was connected to the aperture function Eq. (13.1). That gives the general far field response for any given set of sample points. When one considers certain geometries such as a linear array, planar array, circular array, cylindric array or spherical array, the array response vector will have a different structure. The positions

**Table 13.1** Array Element Positions for Some Common Array Geometries

Structure	Element position
Linear array	$\mathbf{r}_n = [(n-1)d_x, 0, 0]$
2-D rectangular	$\mathbf{r}_{k,n} = [(k-1)d_x, (n-1)d_y, 0]$
Multi-ring circular	$\mathbf{r}_{k,n} = \left[ r_n \cos\left(\frac{2\pi(kN+n)}{KN}\right), r_n \sin\left(\frac{2\pi(kN+n)}{KN}\right), 0 \right]$
Cylindrical	$\mathbf{r}_{k,n} = \left[ r \cos\left(\frac{2\pi k}{K}\right), r \sin\left(\frac{2\pi k}{K}\right), z_n \right]$
Spherical	$\mathbf{r}_{k,n} = r \left[ \cos\left(\frac{2\pi k}{K}\right) \sin\left(\frac{2\pi n}{N}\right), \sin\left(\frac{2\pi k}{K}\right) \sin\left(\frac{2\pi n}{N}\right), \cos\left(\frac{2\pi n}{N}\right) \right]$

**Table 13.2** Array Response for Some Common Array Geometries

Structure	$[\mathbf{d}_a(\omega, \phi, \theta)]_{k,n}$
Linear array	$d_{0,n}(\omega, \phi) = \exp\left(i \frac{\omega d \sin(\theta) \cos(\phi)(n-1)}{c}\right)$
2-D rectangular	$d_{k,n}(\omega, \phi) = \exp\left(j \frac{\omega}{c} (d_x \sin(\theta) \cos(\phi)(k-1) + d_y \sin(\theta) \sin(\phi)(n-1))\right)$
Multi-ring circular	$d_{k,n}(\omega, \phi, \theta) = \exp\left(j \frac{\omega r_p}{c} \sin(\theta) \cos\left(\phi - \frac{2\pi(kN+n)}{KN}\right)\right)$
Cylindrical	$d_{k,n}(\omega, \phi, \theta) = \exp\left(j \frac{\omega}{c} \left( r \sin(\theta) \cos\left(\phi - \frac{2\pi k}{K}\right) + z_n \cos(\theta) \right)\right)$
Spherical	$d_{k,n}(\omega, \phi, \theta) = \exp\left(j \frac{\omega r}{c} \left( \sin(\theta) \sin\left(\frac{\pi(n-1)}{N-1}\right) \cos\left(\phi - \frac{2\pi k}{K}\right) + \cos(\theta) \cos\left(\frac{\pi(n-1)}{N-1}\right) \right)\right)$

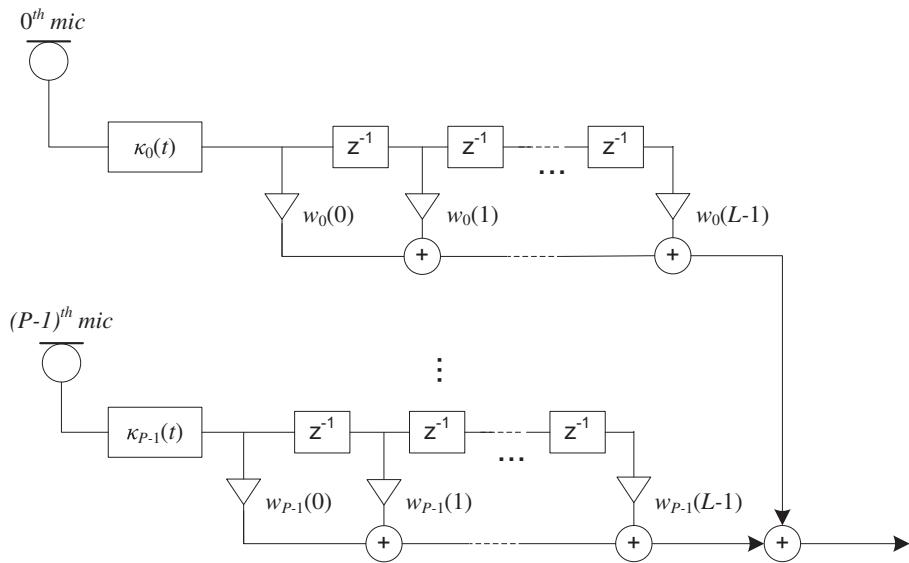
of the array elements for some common geometries, assuming that the origin of the coordinate system has been chosen as one of the points, can be found in Table 13.1.

From the array element positions given in Table 13.1 and utilizing the unit direction vector  $\alpha$ , the array response vector for the far field can be calculated. They have been tabulated in Table 13.2.

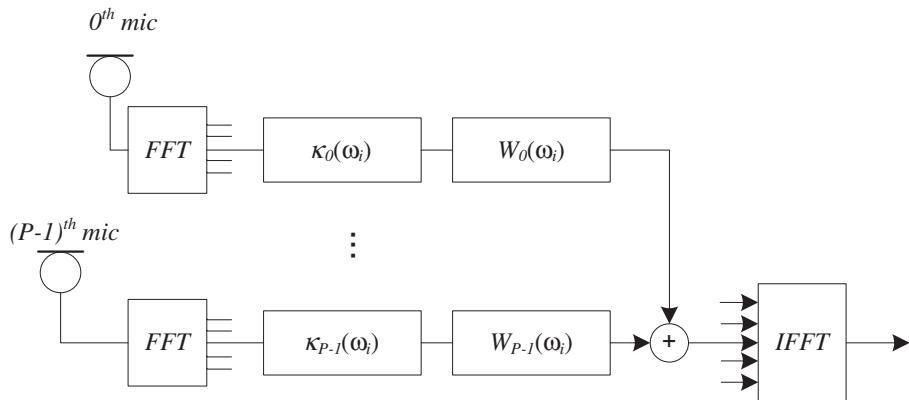
The array response vector is connected to the far field response via the weighted sum in (13.3). Using this equation, a design model of the broadband beamformer can be found. It is also possible to design beamformers using wave propagation modeling, which is considered in a separate section.

### 3.13.3 Broadband beamformer design in element space

In this section, we consider the broadband beamformer design in element space, thus in a spatially sampled domain. The design can be considered for time domain implementation using conventional FIR filters or possibly IIR filters, according to Figure 13.2. Pre-steering delays  $\kappa_p(t)$  have also been included. These delays steer the array to a prescribed direction. This direction is usually determined by some tracking algorithm which is outside the scope of this chapter. Alternatively, the design can be considered in transform domain using multi-rate filtering techniques, see Figure 13.3. This element space design contrasts to the wave propagation domain which considers orthonormal expansion of


**FIGURE 13.2**

Broadband beamformer, time domain.


**FIGURE 13.3**

Broadband beamformer, frequency domain.

the wave field on the receiver shape. Then the orthonormal expansion is sampled in such a way that necessary orthogonality properties are maintained. Using that technique allows the spatial and temporal domain to be separated such that steering of beams is independent of the beamforming. This topic will be discussed in a separate section.

Now consider the frequency domain output of an element space broadband beamformer for a point source in position  $\mathbf{r}_m$

$$Y_m(\omega) = S_m(\omega)\mathbf{W}^H(\omega)\mathbf{H}(\omega, \mathbf{r}_m), \quad (13.5)$$

where  $S(\omega)$  is the spectral density of the source, and  $\mathbf{H}(\omega, \mathbf{r}_m)$  is the array response vector consisting of the Green function Eq. (13.4) describing the individual channel between the source and each of the sensor array elements

$$\mathbf{H}(\omega, \mathbf{r}_m) = \begin{pmatrix} H_1(\omega, \mathbf{r}_m) \\ H_2(\omega, \mathbf{r}_m) \\ \vdots \\ H_P(\omega, \mathbf{r}_m) \end{pmatrix}. \quad (13.6)$$

Furthermore,  $\mathbf{W}(\omega)$  is a vector with the frequency function of the beamformer weights for each element and is described by

$$\mathbf{W}(\omega_s) = \begin{pmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_P^T \end{pmatrix} \mathbf{e}(\omega_s), \quad (13.7)$$

where  $\mathbf{w}_p = [w_p(0), w_p(1), \dots, w_p(L-1)]^T$  is the weight vector with real valued weights for the  $p$ th FIR filter in the broadband array, and the filter response vector is given by  $\mathbf{e}(\omega_s) = [1, e^{-j\omega_s}, \dots, e^{-j(L-1)\omega_s}]^T$ , where  $\omega_s = \omega T_s$  denotes discrete frequency. The assumption is that the signal at every element is bandlimited and sampled with sample frequency  $f_s = 1/T_s$ . Now, instead of using matrix form for the filter weights, stack the weight vectors  $\mathbf{w}_p$  into an  $PL \times 1$  weight vector  $\mathbf{w}$  and also combining the array response from the source to every element with the FIR response of every filter

$$\mathbf{d}(\omega, \mathbf{r}_m) = \mathbf{H}(\omega, \mathbf{r}_m) \otimes \mathbf{e}(\omega_s), \quad (13.8)$$

where  $\otimes$  defines Kronecker product. The overall response from a position  $\mathbf{r}_m$  is given by

$$G(\omega, \mathbf{r}_m) = \mathbf{w}^T \mathbf{d}(\omega, \mathbf{r}_m), \quad (13.9)$$

where  $\mathbf{d}(\omega, \mathbf{r}_m)$  is a column vector of length  $N = PL$ . The optimized beamformer is designed by finding the coefficients of the filters such that the actual response  $G(\omega, \mathbf{r}_m) = \mathbf{w}^T \mathbf{d}(\omega, \mathbf{r}_m)$  fits a given desired response  $G_d(\omega, \mathbf{r}_m)$ . The performance measure used here is the error between an actual frequency response  $G(\omega, \mathbf{r}_m)$  and a desired frequency response  $G_d(\omega, \mathbf{r}_m)$ ,

$$\xi(\omega, \mathbf{r}_m) = G(\omega, \mathbf{r}_m) - G_d(\omega, \mathbf{r}_m). \quad (13.10)$$

The coefficients of the filters are found such that this error measure is minimized. This expression is now formed so it is valid for a general channel model for a homogeneous medium. Under simplified modeling considerations (free and far field), it is possible to express Eq. (13.4) using the far field array response vector  $\mathbf{d}_a(\omega, \phi, \theta)$ . This relationship normally assumes a choice of coordinate system such that the origin is located in one of the array elements. In that case, one has one common Green

function to one element and then the relative delay to the other sensors; these response functions can be found in Table 13.2 for various geometries. Then the response will be a function of  $(\omega, \theta, \phi)$ . Since the response is dependent on direction and frequency, directivity patterns for each frequency can be plotted accordingly. In the near field, the response is dependent on every point and not just the direction.

The minimization of  $\xi(\omega, \mathbf{r}_m)$  can be performed by employing standard filter design methods such as Weighted Least-Squares (WLS) or Chebyshev (min-max) optimization. However, there has also been suggested solutions based on Total Least Squares (TLS) and Eigen-Filter (EF) methods. These techniques are similar to those used for multi-dimensional filter design, except that the main difference for broadband beamforming design is convolutive channel and sensor amplitude and phase uncertainty. Accordingly, that uncertainty also needs to be considered in the design which leads to various robust design techniques. In the sequel, we will discuss those methods in details.

### 3.13.3.1 Weighted Least Squares design

The nominal Weighted Least Square (WLS) design is the standard way to find optimal weights for the broadband beamformer. The details of the method are as follows. The transfer function from a spatial point with position vector  $\mathbf{r}_T$  to the  $n$ th weight  $w_n$  of the broadband beamformer is denoted by  $d_n(\omega, \mathbf{r}_T)$ . Let  $G_d(\omega, \mathbf{r}_T)$  and  $G(\omega, \mathbf{r}_T)$  be the specified desired response and the nominal response of the broadband beamformer, respectively. The response vectors are defined in space and frequency. The nominal array response vector  $\mathbf{d}(\omega, \mathbf{r}_m)$  is assumed to be known. The WLS formulation is given by

$$\begin{aligned} \min_{\mathbf{w}} J_{\text{WLS}}(\mathbf{w}) &= \min_{\mathbf{w}} \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) |\xi(\omega, \mathbf{r}_T)|^2 d\mathbf{r}_T d\omega \\ &= \min_{\mathbf{w}} (\mathbf{w}^T \mathbf{A}_{\text{WLS}} \mathbf{w} - 2\text{Re} \left\{ \mathbf{a}_{\text{WLS}}^H \mathbf{w} \right\} + b_{\text{WLS}}), \end{aligned} \quad (13.11)$$

where  $\Omega$  is the range of frequencies and  $R$  is spatial area of the design and

$$\mathbf{A}_{\text{WLS}} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \mathbf{d}(\omega, \mathbf{r}_T) \mathbf{d}^H(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega, \quad (13.12)$$

$$\mathbf{a}_{\text{WLS}} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \mathbf{d}(\omega, \mathbf{r}_T) G_d^H(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega, \quad (13.13)$$

$$b_{\text{WLS}} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) |G_d(\omega, \mathbf{r}_T)|^2 d\mathbf{r}_T d\omega. \quad (13.14)$$

The optimal solution is found as

$$\mathbf{w} = \mathbf{A}_{\text{WLS}}^{-1} \mathbf{a}_{\text{WLS}}. \quad (13.15)$$

One property of the WLS optimization is that one has little control of the sidelobes of the beampattern. To achieve such control, sidelobe constraints can be included into (13.11), yielding

$$\min_{\mathbf{w}} J_{\text{WLS}}(\mathbf{w}) = \mathbf{w}^T \mathbf{A}_{\text{WLS}} \mathbf{w} - 2\text{Re} \left\{ \mathbf{a}_{\text{WLS}}^H \mathbf{w} \right\} + b_{\text{WLS}} \quad (13.16)$$

$$\text{subject to } |\mathbf{d}^H(\omega, \mathbf{r}_T) \mathbf{w}|^2 \leq \epsilon \quad (\omega, \mathbf{r}_T) \in \text{stop region},$$

where  $\mathbf{A}_{\text{WLS}}$ ,  $\mathbf{a}_{\text{WLS}}$ , and  $b_{\text{WLS}}$  have already been defined and  $\epsilon$  gives the sidelobe tolerance. The stop region is formed by the spectral stop-band region  $\Omega_{st}$  and the spatial stop-band region  $R_{st}$ , respectively.

This problem is a quadratic semi-definite program which can be solved directly by discretizing the constraints. However, it becomes a large-scale problem, especially when the spatial domain is a two or three dimensional region. For large-scale constrained optimization problems, the most efficient algorithm is the interior point algorithm, which has polynomial time computational complexity and has been applied successfully for linear programming, linear semi-definite programming, second-order cone programming and convex programming. As the constraints are quadratic functions, this problem can be transformed into a linear semi-definite programming problem and the interior point algorithm can be applied. For references, see the section on further reading.

### 3.13.3.2 Total Least Squares design and Eigen-Filters

Another popular way to design beamformer coefficients is to use Eigen-Filters. The Eigen-Filter cost function is obtained by using the following expression:

$$\max_{\mathbf{w}} J'_{\text{Eig}}(\mathbf{w}) = \max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{A}_{\text{ROI}} \mathbf{w}}{\mathbf{w}^T \mathbf{A}_{\text{TOT}} \mathbf{w}}, \quad (13.17)$$

where

$$\mathbf{A}_{\text{ROI}} = \int_{\Omega_{\text{ROI}}} \int_{R_{\text{ROI}}} V(\omega, \mathbf{r}_T) \mathbf{d}(\omega, \mathbf{r}_T) \mathbf{d}^H(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.18)$$

is a power measure defined over the passband region, thus it is defined by the spectral passband region  $\Omega_{\text{ROI}}$  and the spatial passband region  $R_{\text{ROI}}$ , respectively. In a similar manner the power measure for the total region is

$$\mathbf{A}_{\text{TOT}} = \int_{\Omega_{\text{TOT}}} \int_{R_{\text{TOT}}} V(\omega, \mathbf{r}_T) \mathbf{d}(\omega, \mathbf{r}_T) \mathbf{d}^H(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega, \quad (13.19)$$

where  $\Omega_{\text{TOT}}$  and  $R_{\text{TOT}}$  are the spectral total region and the spatial total region, respectively. The denominator matrix  $\mathbf{A}_{\text{TOT}}$  is constructed from all regions. The optimal weights are found by maximizing  $J'_{\text{Eig}}(\mathbf{w})$  with respect to  $\mathbf{w}$ , which is also known as a generalized eigenvalue problem. This optimization usually gives low sidelobes, thus high suppression outside the ROI. However there is no control over the passband region and it does not work well for broadband arrays. This results in high distortion of the input signal from the ROI in the low frequency band. To improve this and still maintain a high noise suppression, an alternative formulation has been suggested. This so called Total Least Squares (TLS) formulation seeks a solution comprising both the WLS error and the Eigen-Filter formulation.

In this case, the response for the region of interest versus the overall region or the region of rejection is optimized with the equation given by

$$\xi_{\text{TLS}}(\omega, \mathbf{r}_T) = \frac{|\xi(\omega, \mathbf{r}_T)|}{\sqrt{\mathbf{w}^T \mathbf{A}_{e,\text{TOT}} \mathbf{w} + 1}}, \quad (13.20)$$

where  $\xi(\omega, \mathbf{r}_T)$  is defined as in Eq. (13.10). Now suppose that the beamformer design is based on (13.20), i.e., define the cost function as

$$\begin{aligned} J_{\text{TLS}}(\mathbf{w}) &= \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \xi_{\text{TLS}}^2(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \\ &= \frac{\mathbf{w}_e^T \mathbf{A}_{\text{TLS}} \mathbf{w}_e}{\mathbf{w}_e^T \mathbf{A}_{e,\text{TOT}} \mathbf{w}_e}, \end{aligned} \quad (13.21)$$

where

$$\mathbf{A}_{\text{TLS}} = \begin{bmatrix} \mathbf{A}_{\text{WLS}} & \mathbf{a}_{\text{WLS}} \\ \mathbf{a}_{\text{WLS}}^H & b_{\text{WLS}} \end{bmatrix}, \quad \mathbf{A}_{e,\text{TOT}} = \begin{bmatrix} \mathbf{A}_{\text{TOT}} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}, \quad \mathbf{w}_e = \begin{bmatrix} \mathbf{w} \\ -1 \end{bmatrix}. \quad (13.22)$$

Thus the Total Least Squares Eigen-Filter design formulation is given by

$$\min_{\mathbf{w}_e} J_{\text{TLS}}(\mathbf{w}) = \min_{\mathbf{w}_e} \frac{\mathbf{w}_e^T \mathbf{A}_{e,\text{TLS}} \mathbf{w}_e}{\mathbf{w}_e^T \mathbf{A}_{e,\text{TOT}} \mathbf{w}_e}. \quad (13.23)$$

The solution to (13.23) is the generalized eigenvector of  $\mathbf{A}_{e,\text{TLS}}$  and  $\mathbf{A}_{e,\text{TOT}}$  that corresponds to the smallest generalized eigenvalue. The filter coefficients  $\mathbf{w}$  can be extracted from  $\mathbf{w}_e$  after scaling it such that the last element is  $-1$ . The advantage of this formulation is that a measure of desired response has been included, thus less distortion can be achieved.

### 3.13.3.3 Chebyshev design

It is also possible to optimize the Chebyshev error or min-max error instead of the Least Squares. This optimization problem minimizes the maximum deviation of the error. In this case the problem is written as follows:

$$J(\mathbf{w}_c) = \min_{\mathbf{w}_c} \max_{(\omega, \mathbf{r}_T) \in S} V(\omega, \mathbf{r}_T) \left| \mathbf{w}_c^T \mathbf{d}(\omega, \mathbf{r}_T) - G_d(\omega, \mathbf{r}_T) \right|, \quad (13.24)$$

where  $G(\omega, \mathbf{r}_T) = \mathbf{w}_c^T \mathbf{d}(\omega, \mathbf{r}_T)$  is the beamformer response function and  $G_d(\omega, \mathbf{r}_T)$  is the desired response vector as defined previously. The frequency and source location vector should belong to the optimization set  $S$ . This set  $S$  is given by the passband region and the stopband region both in frequency and space. The weighting  $V(\omega, \mathbf{r}_T)$  is a positive function. The original Chebyshev problem (13.24) can be rewritten in an equivalent form as follows:

$$\begin{aligned} \min_{\delta} \quad & \delta \\ \text{subject to} \quad & V^2(\omega, \mathbf{r}_T) |\mathbf{w}_c^T \mathbf{d}(\omega, \mathbf{r}_T) - G_d(\omega, \mathbf{r}_T)|^2 \leq \delta, \quad \forall (\omega, \mathbf{r}_T) \in S, \end{aligned} \quad (13.25)$$

where  $\delta$  is the maximum deviation. The equivalence follows from the fact that  $\delta$  is the maximum deviation of the non-linear constraint and this  $\delta$  is minimized. Thus the original Chebyshev problem has been converted to a sequential quadratic programming problem. This problem can also be converted into a linear semi-definite programming problem. The original problem is semi-infinite and thus continuous, but it can be discretized. Although the constrained optimization problem (13.25) can be solved directly, the number of constraints is usually large. For a broadband beamforming case it becomes a large-scale problem since the number of constraints grow, especially when the spatial domain is a two or three-dimensional region. This will be the case for near-field problems. As stated earlier in Section 3.13.3.1, the most efficient algorithm to solve this problem is the interior point algorithm.

### 3.13.3.4 Model and robust formulation

As mentioned earlier, the performance of broadband beamformers designed with the nominal WLS, TLS and min-max optimization methods are severely degraded due to model errors. The design of robust beamformers topic is extensive, see [6], here we concentrate on the problem of designing data

independent broadband beamformers. For adaptive beamformers the problems that occur are things like SOI cancelation or source distortion. In the data independent broadband case the main concern is the deviation between the nominal array response and the actual channel response a source input is subjected to. In many cases this deviation means that the weights designed on the basis of the nominal model are not useful since this deviation become exorbitant. Typically, errors could arise from phase and amplitude mismatch of sensors and amplifiers, as well as sensor placement errors. Another aspect of importance is the difference between the actual environment and propagation model used for the design. Thus, it is important to be able to model those errors and include them in the design. Normally, the errors are considered to be either multiplicative or additive. Accordingly, the perturbed response vector is given either by

$$\tilde{\mathbf{H}}(\omega, \mathbf{r}_T) = \mathbf{H}(\omega, \mathbf{r}_T) \odot \delta_H(\omega, \mathbf{r}_T) \quad (13.26)$$

or

$$\tilde{\mathbf{H}}(\omega, \mathbf{r}_T) = \mathbf{H}(\omega, \mathbf{r}_T) + \delta_H(\omega, \mathbf{r}_T), \quad (13.27)$$

where  $\delta_H(\omega, \mathbf{r}_T)$  is modeled as a complex random vector and  $\mathbf{H}(\omega, \mathbf{r}_T)$  is the nominal array response vector and  $\odot$  denotes direct product.

We first consider the multiplicative case according to Eq. (13.26), the  $p$ th element in the random vector  $\delta_H(\omega, \mathbf{r}_T)|_p = \delta_{H,p}(\omega, \mathbf{r}_T)$  can be characterized by its gain error  $|\delta_{H,p}(\omega, \mathbf{r}_T)|$  and phase error  $\arg(\delta_{H,p}(\omega, \mathbf{r}_T))$ . The gain and phase are assumed to be independent errors. This perturbation model can now be incorporated in the array response vector given in Eq. (13.8) yielding

$$\tilde{\mathbf{d}}(\omega, \mathbf{r}_T) = \tilde{\mathbf{H}}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega), \quad (13.28)$$

which can be rewritten using some simple matrix manipulations into

$$\tilde{\mathbf{d}}(\omega, \mathbf{r}_T) = (\delta_H(\omega, \mathbf{r}_T) \otimes \mathbf{1}) \odot \mathbf{d}(\omega, \mathbf{r}_T),$$

where  $\mathbf{1}$  is a  $L \times 1$  vector with all elements equal to one. Based on the perturbed array response, form the outer product such that

$$\begin{aligned} \tilde{\mathbf{Q}}(\omega, \mathbf{r}_T) &= \tilde{\mathbf{d}}(\omega, \mathbf{r}_T) \tilde{\mathbf{d}}^H(\omega, \mathbf{r}_T) \\ &= \left( (\delta_H(\omega, \mathbf{r}_T) \delta_H^H(\omega, \mathbf{r}_T)) \otimes \mathbf{1} \mathbf{1}^T \right) \odot \mathbf{Q}(\omega, \mathbf{r}_T), \end{aligned} \quad (13.29)$$

where  $\mathbf{Q}(\omega, \mathbf{r}_T) = \mathbf{d}(\omega, \mathbf{r}_T) \mathbf{d}^H(\omega, \mathbf{r}_T)$ . Now also form the cross-term with the desired response

$$\tilde{\mathbf{g}}(\omega, \mathbf{r}_T) = \tilde{\mathbf{d}}(\omega, \mathbf{r}_T) G_d^H(\omega, \mathbf{r}_T), \quad (13.30)$$

which can be rewritten as

$$\tilde{\mathbf{g}}(\omega, \mathbf{r}_T) = (\delta_H(\omega, \mathbf{r}_T) \otimes \mathbf{1}) \odot \mathbf{g}(\omega, \mathbf{r}_T).$$

Note that the sensor gain and phase errors can be considered as random variables and it is the error vector  $\delta_H(\omega, \mathbf{r}_T)$  that is of interest. Let the matrix containing the perturbations be defined according to

$$\Xi(\omega, \mathbf{r}_T) = \delta_H(\omega, \mathbf{r}_T) \delta_H^H(\omega, \mathbf{r}_T). \quad (13.31)$$

These perturbations are considered as a random vector, and to optimize the mean performance, the gain and phase probability density functions (PDFs) are assumed to be known, i.e.,

$$\bar{\boldsymbol{\Xi}}(\omega, \mathbf{r}_T) = E[\boldsymbol{\Xi}(\omega, \mathbf{r}_T)] = E\left[\delta_H(\omega, \mathbf{r}_T)\delta_H^H(\omega, \mathbf{r}_T)\right], \quad (13.32)$$

$$\bar{\boldsymbol{\delta}}_H(\omega, \mathbf{r}_T) = E[\delta_H(\omega, \mathbf{r}_T)]. \quad (13.33)$$

Thus the perturbed quantities for the multiplicative error can be written as

$$\bar{\mathbf{Q}}(\omega, \mathbf{r}_T) = E[\tilde{\mathbf{Q}}(\omega, \mathbf{r}_T)] = (\bar{\boldsymbol{\Xi}}(\omega, \mathbf{r}_T) \otimes \mathbf{1}\mathbf{1}^T) \odot \mathbf{Q}(\omega, \mathbf{r}_T), \quad (13.34)$$

$$\bar{\mathbf{g}}(\omega, \mathbf{r}_T) = E[\tilde{\mathbf{g}}(\omega, \mathbf{r}_T)] = (\bar{\boldsymbol{\delta}}_H(\omega, \mathbf{r}_T) \otimes \mathbf{1}) \odot \mathbf{d}(\omega, \mathbf{r}_T)G_d^H(\omega, \mathbf{r}_T). \quad (13.35)$$

The additive disturbance is the next situation to be studied. One interpretation of this is that the additive disturbance is a random reflection of the incoming wave, local scattering or distributed source. In this case, the perturbed response vector is given by

$$\tilde{\mathbf{H}}(\omega, \mathbf{r}_T) = \mathbf{H}(\omega, \mathbf{r}_T) + \boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T). \quad (13.36)$$

The outer product of the perturbed response is given by

$$\begin{aligned} \tilde{\mathbf{Q}}_a(\omega, \mathbf{r}_T) &= \tilde{\mathbf{d}}_a(\omega, \mathbf{r}_T)\tilde{\mathbf{d}}_a^H(\omega, \mathbf{r}_T) \\ &= (\boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega) + \mathbf{d}(\omega, \mathbf{r}_T))(\boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega) + \mathbf{d}(\omega, \mathbf{r}_T))^H. \end{aligned}$$

This expression can be expanded as follows:

$$\tilde{\mathbf{Q}}_a(\omega, \mathbf{r}_T) = \mathbf{Q}(\omega, \mathbf{r}_T) + \Delta\mathbf{Q}_a(\omega, \mathbf{r}_T),$$

where

$$\Delta\mathbf{Q}_a(\omega, \mathbf{r}_T) = \left( \boldsymbol{\Xi}_a(\omega, \mathbf{r}_T) + \boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T)\mathbf{H}^H(\omega, \mathbf{r}_T) + \mathbf{H}(\omega, \mathbf{r}_T)\boldsymbol{\delta}_{a,H}^H(\omega, \mathbf{r}_T) \right) \otimes \mathbf{e}(\omega)\mathbf{e}^H(\omega)$$

and

$$\begin{aligned} \tilde{\mathbf{g}}_a(\omega, \mathbf{r}_T) &= \tilde{\mathbf{d}}_a(\omega, \mathbf{r}_T)G_d^H(\omega, \mathbf{r}_T) = (\boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega) + \mathbf{d}(\psi, \omega, \mathbf{r}_T))G_d(\omega, \mathbf{r}_T) \\ &= \mathbf{g}(\psi, \omega, \mathbf{r}_T) + \boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega)G_d(\omega, \mathbf{r}_T). \end{aligned}$$

The expectation can be taken on those equations resulting in

$$\bar{\mathbf{Q}}_a(\omega, \mathbf{r}_T) = \mathbf{Q}(\omega, \mathbf{r}_T) + \Delta\bar{\mathbf{Q}}_a(\omega, \mathbf{r}_T), \quad (13.37)$$

$$\bar{\mathbf{g}}(\omega, \mathbf{r}_T) = \mathbf{g}(\omega, \mathbf{r}_T) + \bar{\boldsymbol{\delta}}_{a,H}(\omega, \mathbf{r}_T) \otimes \mathbf{e}(\omega)G_d(\omega, \mathbf{r}_T), \quad (13.38)$$

where the model error is given by

$$\Delta\bar{\mathbf{Q}}_a(\omega, \mathbf{r}_T) = \left( \bar{\boldsymbol{\Xi}}_a(\omega, \mathbf{r}_T) + \bar{\boldsymbol{\delta}}_{a,H}(\omega, \mathbf{r}_T)\mathbf{H}^H(\omega, \mathbf{r}_T) + \mathbf{H}(\omega, \mathbf{r}_T)\bar{\boldsymbol{\delta}}_{a,H}^H(\omega, \mathbf{r}_T) \right) \otimes \mathbf{e}(\omega)\mathbf{e}^H(\omega),$$

the correlation matrix of the perturbation error is

$$\bar{\boldsymbol{\Xi}}_a(\omega, \mathbf{r}_T) = E\left[\boldsymbol{\delta}_{a,H}(\omega, \mathbf{r}_T)\boldsymbol{\delta}_{a,H}^H(\omega, \mathbf{r}_T)\right]$$

and the mean of the perturbation vector is

$$\bar{\delta}_{a,H}(\omega, \mathbf{r}_T) = E[\delta_{a,H}(\omega, \mathbf{r}_T)].$$

These expressions are given in form of an expectation which is the mean of the errors on each element. Thus, to obtain those expressions, one needs a model on the probability density of the errors. It is also possible to study a worst case scenario but this is more difficult to obtain, since it is not easy to find worst case bounds.

### 3.13.3.5 Robust WLS design

In the robust formulated WLS design, a model of the random vector is employed. The robust design is based on the same set of weighting coefficients  $V(\omega, \mathbf{r}_T)$  and desired response  $G_d(\omega)$  as in the nominal design, but with an unknown (random) array response vector added to or multiplied with the nominal array response vector. The robust WLS formulation is based on the averaged response

$$\begin{aligned} \min_{\mathbf{w}} J_{WLS}(\mathbf{w}) &= \min_{\mathbf{w}} E \left[ \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) |\tilde{\xi}(\psi, \omega, \mathbf{r}_T)|^2 d\mathbf{r}_T d\omega \right] \\ &= \min_{\mathbf{w}} \left( \mathbf{w}^H \bar{\mathbf{A}}_{WLS} \mathbf{w} - 2\Re \left\{ \bar{\mathbf{a}}_{WLS}^H \mathbf{w} \right\} + b_{WLS} \right), \end{aligned} \quad (13.39)$$

where  $\Omega$  is the range of frequencies and  $R$  is the spatial region for the design, and

$$\bar{\mathbf{A}}_{WLS} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \bar{\mathbf{Q}}(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.40)$$

$$\bar{\mathbf{a}}_{WLS} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \bar{\mathbf{g}}(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.41)$$

$$b_{WLS} = \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) b(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega, \quad (13.42)$$

where  $V(\omega, \mathbf{r}_T)$  is the weighting function, and  $\Omega$  and  $R$  are the regions of interest for  $\omega$  and  $\mathbf{r}_T$ , respectively.  $\bar{\mathbf{Q}}(\omega, \mathbf{r}_T)$  and  $\bar{\mathbf{g}}(\omega, \mathbf{r}_T)$  are given in (13.34) and (13.35), respectively, for the multiplicative case. In the additive error model case, the structure does not change but  $\bar{\mathbf{Q}}_a(\omega, \mathbf{r}_T)$  and  $\bar{\mathbf{g}}_a(\omega, \mathbf{r}_T)$  are given by (13.37) and (13.38) instead. There is no change to the optimization *per se*, only in the way the problem is set up. In the examples, we will see the difference in results when inserting the robust model.

It is also possible to extend the constrained WLS design as a robust formulation; the design equations are as follows:

$$\min_{\mathbf{w}} J_{CLS}(\mathbf{w}) = \mathbf{w}^H \bar{\mathbf{A}}_{CLS} \mathbf{w} - 2\Re \left\{ \bar{\mathbf{a}}_{CLS}^H \mathbf{w} \right\} + b_{CLS} \quad (13.43)$$

$$\text{subject to } |\bar{\mathbf{d}}^H(\omega, \mathbf{r}_T) \mathbf{w}| \leq \epsilon \quad (\omega, \mathbf{r}_T) \in \text{stop region},$$

where

$$\bar{\mathbf{A}}_{CLS} = \int_{\Omega_{pb}} \int_{R_{pb}} V(\omega, \mathbf{r}_T) \bar{\mathbf{Q}}(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.44)$$

$$\bar{\mathbf{a}}_{CLS} = \int_{\Omega_{pb}} \int_{R_{pb}} V(\omega, \mathbf{r}_T) \bar{\mathbf{g}}(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.45)$$

$$b_{\text{CLS}} = \int_{\Omega_{pb}} \int_{R_{pb}} V(\omega, \mathbf{r}_T) b(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \quad (13.46)$$

$$\bar{\mathbf{d}}(\omega, \mathbf{r}_T) = V(\omega, \mathbf{r}_T) \bar{\mathbf{g}}(\psi, \omega, \mathbf{r}_T) / G_d(\omega, \mathbf{r}_T) \quad (13.47)$$

with  $\epsilon$  the tolerance of the sidelobes, and  $\Omega_{pb}$  and  $R_{pb}$  are the spectral and spatial pass region, respectively.

### 3.13.3.6 Robust Total Least Squares design

The TLS optimization can also be extended to a robust formulation, where its error function is expressed as

$$\tilde{\xi}_{\text{TLS}}^2(\omega, \mathbf{r}_T) = \frac{E[|\tilde{\xi}(\omega, \mathbf{r}_T)|^2]}{\mathbf{w}^H \bar{\mathbf{A}}_{\text{TOT}} \mathbf{w} + 1}. \quad (13.48)$$

This TLS error can be formulated into a cost function as follows

$$\begin{aligned} \min_{\mathbf{w}_e} J_{\text{TLS}}(\mathbf{w}_e) &= \min_{\mathbf{w}_e} E \left[ \int_{\Omega} \int_R V(\omega, \mathbf{r}_T) \tilde{\xi}_{\text{TLS}}^2(\omega, \mathbf{r}_T) d\mathbf{r}_T d\omega \right] \\ &= \min_{\mathbf{w}_e} \frac{\mathbf{w}_e^H \bar{\mathbf{A}}_{\text{TLS}} \mathbf{w}_e}{\mathbf{w}_e^H \bar{\mathbf{A}}_{e,\text{TOT}} \mathbf{w}_e}, \end{aligned} \quad (13.49)$$

where

$$\bar{\mathbf{A}}_{e,\text{TLS}} = \begin{bmatrix} \bar{\mathbf{A}}_{\text{WLS}} & \bar{\mathbf{a}}_{\text{LS}} \\ \bar{\mathbf{a}}_{\text{LS}}^H & b_{\text{LS}} \end{bmatrix}, \quad \bar{\mathbf{A}}_{e,\text{TOT}} = \begin{bmatrix} \bar{\mathbf{A}}_{\text{TOT}} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}, \quad \mathbf{w}_e = \begin{bmatrix} \mathbf{w} \\ -1 \end{bmatrix}. \quad (13.50)$$

Its solution is given by the generalized eigenvector of  $\bar{\mathbf{A}}_{e,\text{TLS}}$  and  $\bar{\mathbf{A}}_{e,\text{TOT}}$  that corresponds to the smallest generalized eigenvalue. Again, the filter coefficients  $\mathbf{w}$  can be extracted from  $\mathbf{w}_e$  after scaling its last element to  $-1$ .

### 3.13.3.7 Robust Chebyshev design

The above non-robust design of the Chebyshev error can be extended to include robustness constraints. The robust array response includes a perturbation in the same way as for the WLS design

$$\begin{aligned} \min_{\mathbf{w}_c} J(\mathbf{w}_c) &= \min_{\mathbf{w}_c} \|(E[\tilde{\xi}(\omega, \mathbf{r}_T)])\| \\ &= \min_{\mathbf{w}_c} \max_{(\omega, \mathbf{r}_T) \in S} V(\omega, \mathbf{r}_T) \sqrt{E \left[ \left| \mathbf{w}_c^H \tilde{\mathbf{d}}(\omega, \mathbf{r}_T) - G_d(\omega, \mathbf{r}_T) \right|^2 \right]}, \end{aligned} \quad (13.51)$$

where  $V(\omega, \mathbf{r}_T)$  is the positive weighting function.

Now take the square of the Chebyshev error and expand

$$V(\omega, \mathbf{r}_T) \sqrt{E \left[ \left| \mathbf{w}_c^H \tilde{\mathbf{d}}(\omega, \mathbf{r}_T) - G_d(\omega, \mathbf{r}_T) \right|^2 \right]},$$

which yields

$$\begin{aligned} V^2(\omega, \mathbf{r}_T) E \left[ \left| \mathbf{w}_c^T \tilde{\mathbf{d}}(\omega, \mathbf{r}_T) - G_d(\omega, \mathbf{r}_T) \right|^2 \right] \\ = V^2(\omega, \mathbf{r}_T) \left( \mathbf{w}_c^T \bar{\mathbf{Q}}(\omega, \mathbf{r}_T) \mathbf{w}_c - 2\Re\{\bar{\mathbf{g}}(\omega, \mathbf{r}_T)\}^H \mathbf{w}_c + |G_d(\omega, \mathbf{r}_T)|^2 \right). \end{aligned}$$

Define  $E(|\varepsilon_\kappa(\mathbf{w}_c)|^2) = E[|\mathbf{w}_c^H \tilde{\mathbf{d}}_\kappa - G_{d,\kappa}|^2]$  where  $\kappa$  is the index set of sample points over the design area given by  $\Omega \in [\omega_{\text{low}}, \omega_{\text{upper}}]$ ,  $R \in [\text{spatial design region}]$ ; the functions are now sampled over that grid. Accordingly, the robust Chebyshev problem can be formulated as

$$\begin{aligned} \min_{\delta} \quad & \delta \\ \text{subject to} \quad & V_\kappa^2 E \left[ \left| \mathbf{w}_c^H \tilde{\mathbf{d}}_\kappa - G_{d,\kappa} \right|^2 \right] \leq \delta, \quad \forall \kappa \in \{1, \dots, \mathcal{K}\}. \end{aligned} \quad (13.52)$$

This problem is the same as Eq. (13.25) above structurally, but with a random perturbation included. Thus, this problem can be reformulated and solved in the same way.

This concludes the discussion on optimal design of broadband beamformers. For more details, there is a lot of literature available. But the main consideration is how to make the design robust such that it can be applied in real world problems. This will be made even clearer in some design examples.

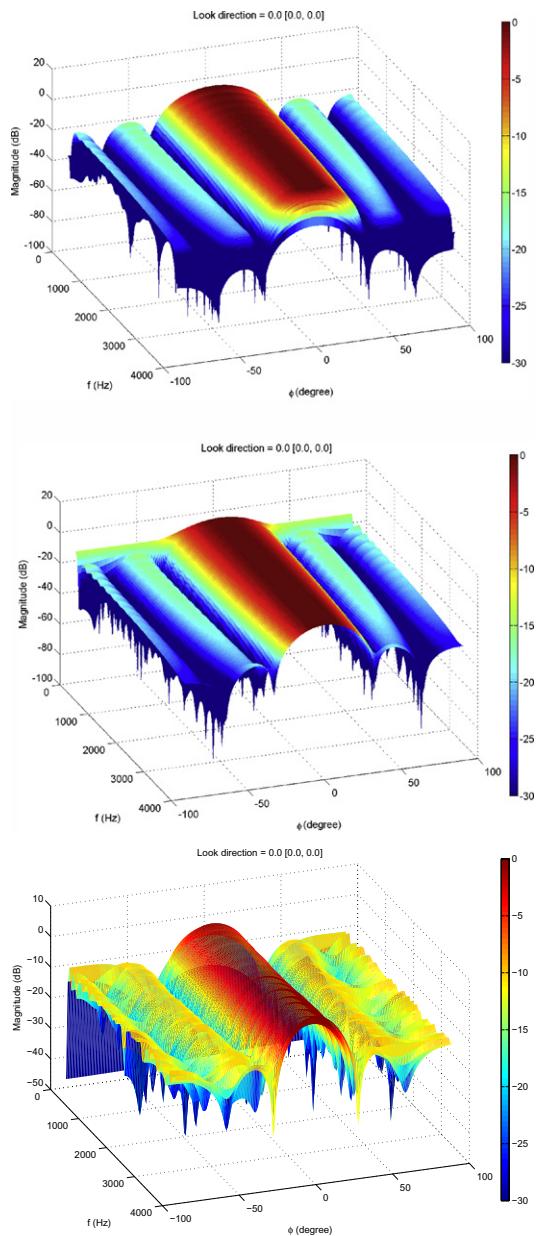
### 3.13.3.8 Design examples

A few design examples are presented to illustrate the formulations discussed. The design examples are for the non-robust WLS, TLS and Chebyshev. A broadside linear array is used with inter-element spacing of 4 cm. The design parameters are as specified in Table 13.3.

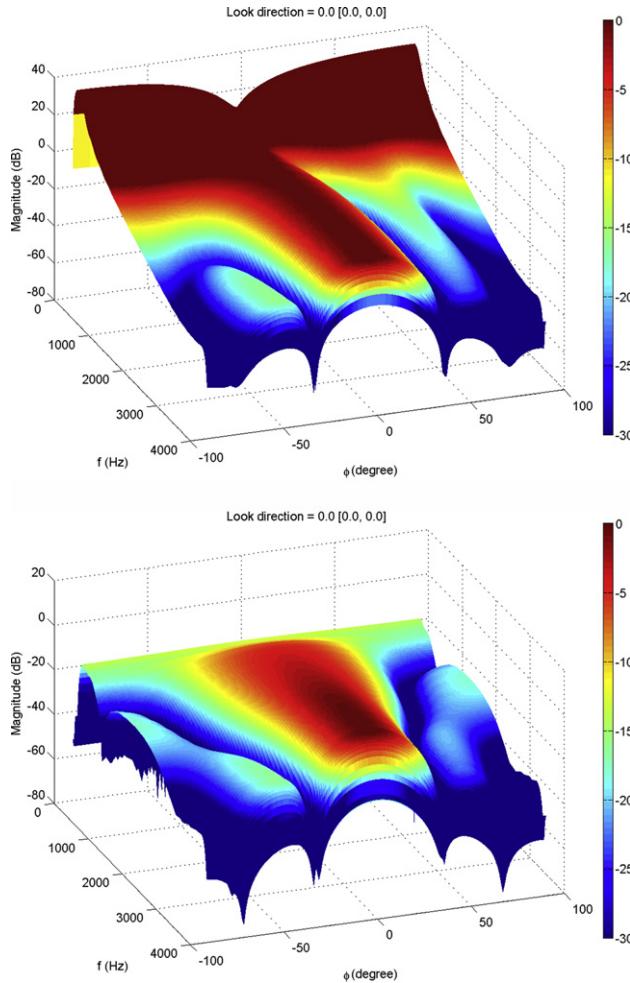
Figure 13.4 shows the normalized beampatterns for non-robust WLS, TLS and Chebyshev designs under ideal conditions. It can be seen from the plots that the TLS gives lower sidelobes than WLS. But the TLS will give more distortion in the passband. The Chebyshev design has equiripple characteristics. This means that the response deviation has been controlled in both passband and stopband. In order to illustrate the benefits with robust formulation, a comparison between normalized beamformer response

**Table 13.3** Common Design Parameters

Parameters	Values
Number of microphones, $K$	5
FIR filter length, $N$	65
Spectral passband, $\Omega_p$	[200, 3800] Hz
Spectral stopband, $\Omega_s$	[0, 100] $\cup$ [3950, 4000] Hz
Sampling frequency, $f_s$	8000 Hz
Speed of sound, $c$	$343 \text{ m s}^{-1}$
Spatial pass region, $\Phi_p$	$[-15^\circ, 15^\circ]$
Spatial stop region, $\Phi_s$	$[-180^\circ, -25^\circ] \cup [25^\circ, 180^\circ]$

**FIGURE 13.4**

Beampatterns for non-robust (a) WLS, (b) TLS, and (c) Chebyshev designs under ideal conditions.

**FIGURE 13.5**

Beampatterns for (a) non-robust and (b) robust WLS designs with perturbation.

for non-robust and robust WLS with perturbation is shown in Figure 13.5, which clearly highlights the robustness. The perturbations are made on phase and amplitude response on each element, the perturbations are made according to a Gaussian distributed in amplitude  $|\delta_H(\omega, \mathbf{r}_T)| \in \Phi(1, 0.05)$  and rectangular distributed in phase  $\arg(\delta_H(\omega, \mathbf{r}_T)) \in R(-0.05, 0.05)$  rad. Similar results can be obtained for the other formulations but only one example is provided as an illustration. This clearly shows the effectiveness of a robust formulation. The WLS problem formulation gives very little extra complexity in the design phase compared to the non-robust but provide a practical beamformer design. In the robust design it has been assumed the same distribution of each individual element as the perturbation above.

Of course in the design case the expectation has been calculated and in the example it is an outcome using that distribution.

### 3.13.3.9 Steerable broadband beamformer

A steerable beamformer can be created in two steps. The first step is to steer the beam using broadband fractional delays on each sensor element, and then broadband beamforming is applied. The fractional delay steering is usually implemented using either a Farrow structure or sub-sampling technique. However, it is also possible to combine the fractional delays directly in the beamforming design, resulting in an integrated system. The steering can then be done by adjusting one single parameter which is a function of the steering direction. From the formulation of the problem, it can be seen that the beamformer weight design can be performed by using any of the previously discussed design techniques.

The broadband steerable Farrow beamformer structure is described in Figure 13.6. The Farrow beamformer consists of  $M$  parallel FIR filters combined with a polynomial interpolation. This allows an approximation of a fractional delay which is used to provide both the capabilities of beam steering and broadband beamforming. For simplicity, we present the case for the horizontal plane, where the steerable beamformer structure using the time domain FIR Farrow filters is shown in Figure 13.6. The beamformer response is given by

$$G(\psi, \omega, \mathbf{r}_T) = \sum_{p=0}^{P-1} \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} w_{p,k}[l] H_p(\omega, \mathbf{r}_T) \psi^k \exp(-j\omega l T_s), \quad (13.53)$$

where  $P$  is the number of microphones,  $K - 1$  is the order of Farrow's structure,  $L$  is the number of filter taps,  $H_p(\omega, \mathbf{r}_T)$  is the array response and  $w_{p,k}(l)$  is the filter coefficients, which are considered to be real. The steering variable  $\psi$ , typically normalized to be between  $-0.5$  and  $0.5$  inclusively, is related to the look direction as in

$$\psi = \frac{\tilde{\psi}}{\tilde{\psi}_{\max}}, \quad (13.54)$$

where  $\tilde{\psi}_{\max}$  is the maximum range of steerable look direction. Note that (13.53) can be written more compactly in matrix form as given by

$$G(\psi, \omega, \mathbf{r}_T) = \mathbf{w}^T \mathbf{d}(\psi, \omega, \mathbf{r}_T), \quad (13.55)$$

where

$$\mathbf{w} = [\mathbf{w}_{0,0}^T \ \dots \ \mathbf{w}_{0,K-1}^T \ | \ \dots \ | \ \mathbf{w}_{P-1,0}^T \ \dots \ \mathbf{w}_{P-1,K-1}^T]^T, \quad (13.56)$$

$$\mathbf{w}_{p,k} = [w_{p,k}[0] \ \dots \ w_{p,k}[L-1]]^T, \quad (13.57)$$

$$\mathbf{d}(\psi, \omega, \mathbf{r}_T) = \mathbf{H}(\omega, \mathbf{r}_T) \otimes \boldsymbol{\psi} \otimes \mathbf{e}(\omega), \quad (13.58)$$

$$\mathbf{H}(\omega, \mathbf{r}_T) = [h_0(\omega, \mathbf{r}_T) \ \dots \ h_{P-1}(\omega, \mathbf{r}_T)]^T, \quad (13.59)$$

$$\boldsymbol{\psi} = [\psi^0 \ \dots \ \psi^{K-1}]^T, \quad (13.60)$$

$$\mathbf{e}(\omega) = [1 \ \dots \ e^{(-j\omega(L-1)T_s)}]^T. \quad (13.61)$$

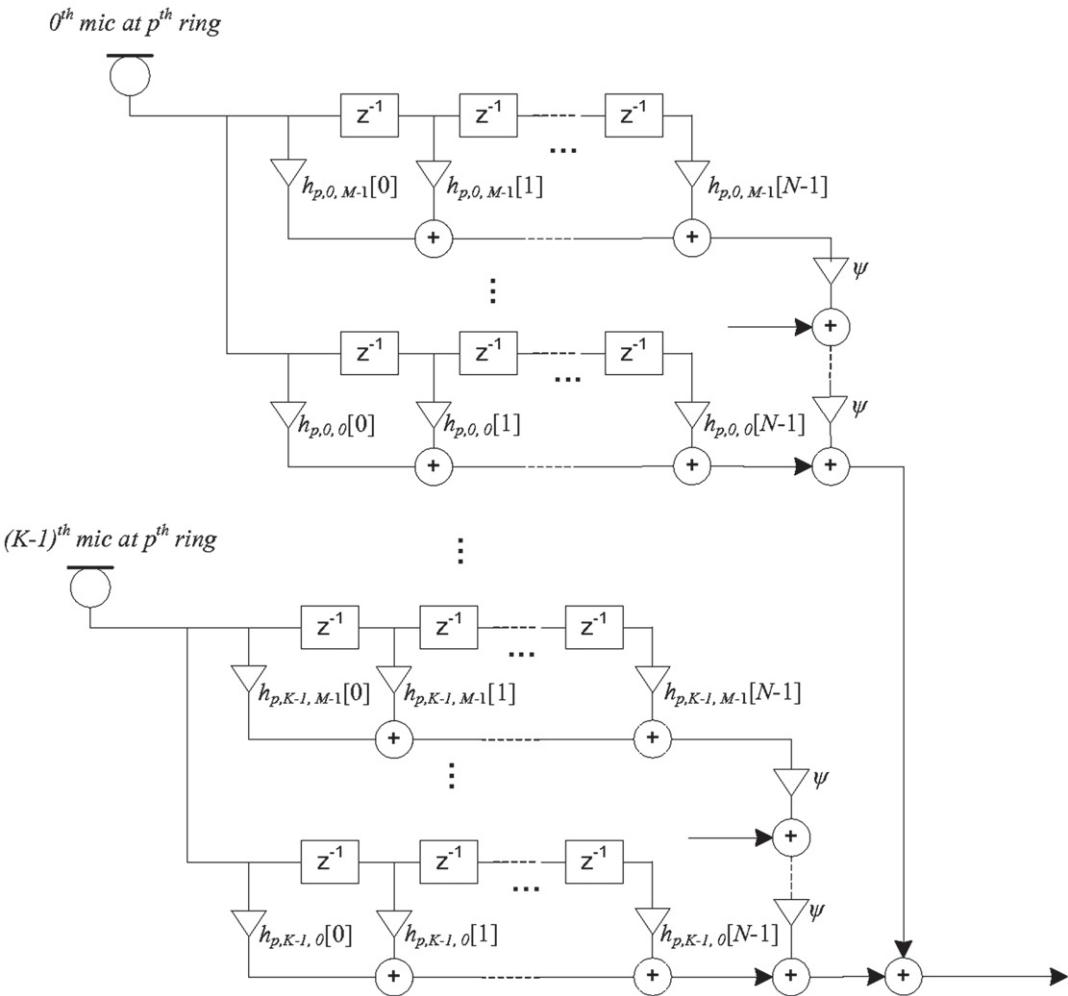


FIGURE 13.6

TD Farrow filter structure for steerable broadband beamformer.

Defining the difference between the designed response  $G(\psi, \omega, \mathbf{r}_T)$  and desired response  $G_d(\psi, \omega, \mathbf{r}_T)$  as

$$\xi(\psi, \omega, \mathbf{r}_T) = G(\psi, \omega, \mathbf{r}_T) - G_d(\psi, \omega, \mathbf{r}_T) \quad (13.62)$$

now define the squared difference as

$$|\xi(\psi, \omega, \mathbf{r}_T)|^2 = \mathbf{w}^T \mathbf{Q}(\psi, \omega, \mathbf{r}_T) \mathbf{w} - 2\Re \left\{ \mathbf{w}^T \mathbf{g}(\psi, \omega, \mathbf{r}_T) \right\} + b(\psi, \omega, \mathbf{r}_T), \quad (13.63)$$

where

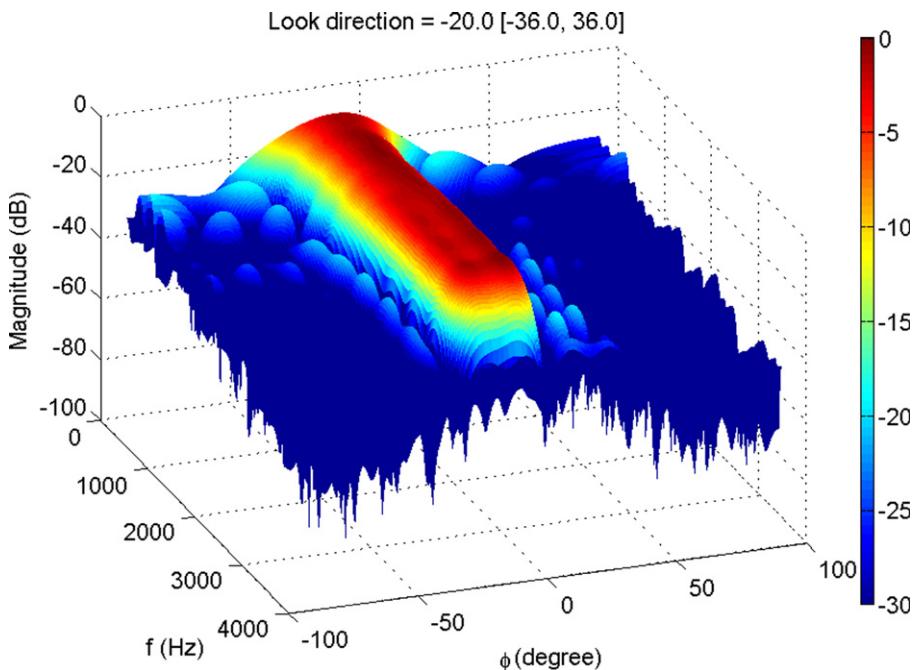
$$\mathbf{Q}(\psi, \omega, \mathbf{r}_T) = \mathbf{d}(\psi, \omega, \mathbf{r}_T) \mathbf{d}^H(\psi, \omega, \mathbf{r}_T), \quad (13.64)$$

$$\mathbf{g}(\psi, \omega, \mathbf{r}_T) = \mathbf{d}(\psi, \omega, \mathbf{r}_T) G_d^H(\psi, \omega, \mathbf{r}_T), \quad (13.65)$$

$$b(\psi, \omega, \mathbf{r}_T) = |G_d(\psi, \omega, \mathbf{r}_T)|^2. \quad (13.66)$$

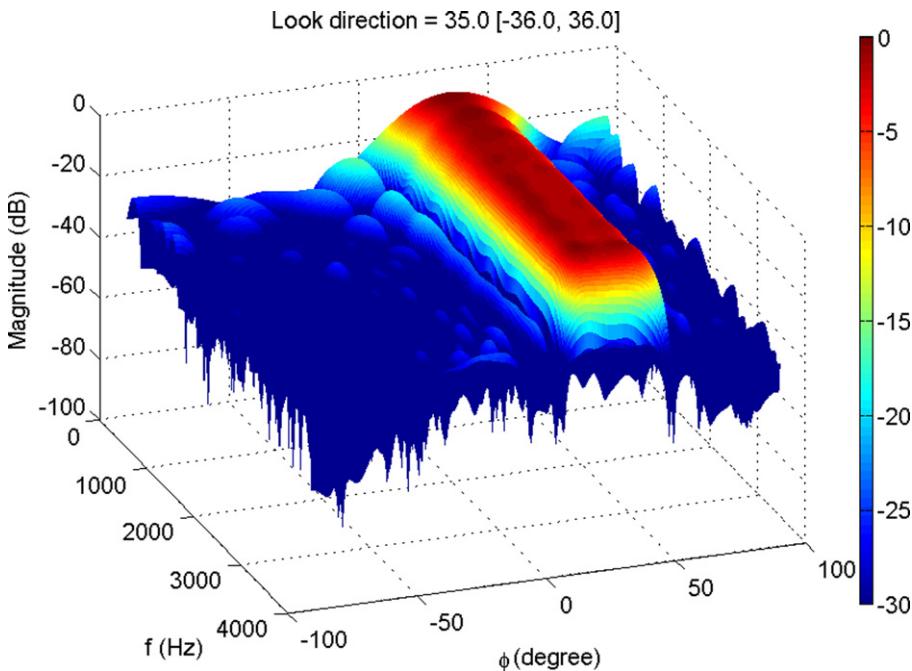
The form of the error function for the Farrow structure is similar as defined in previous sections. Hence, the same optimization techniques can be used. The only difference is that there is a need to expand the equations to optimize over  $\psi$  as well. The usefulness of this formulation will be shown in some examples. It allows the direct steering of broadband beams from one single parameter  $\psi$ . Since the steering is included in the design, the beampattern can more or less be maintained. It is however more challenging to design the beampattern because the number of parameters grows with a factor  $K$ . Furthermore, as the matrix  $\mathbf{Q}(\psi, \omega, \mathbf{r}_T)$  is evaluated for various  $(\psi, \omega, \mathbf{r}_T)$ , there is less variations which leads to numerical difficulty in solving the problem in some cases. The TLS and WLS solutions can still be found. However, the constrained optimizations, Constrained Weighted Least Squares (CWLS) and Chebyshev, result in large scale optimization problems which are a challenge to solve [7].

As an example, a design with the same design specifications as in Table 13.3 is provided with the order of Farrow structure  $M = 4$ . Figures 13.7 and 13.8 show the beampattern steered to  $\psi = -20^\circ$  and  $35^\circ$ , respectively.



**FIGURE 13.7**

Beampattern of steerable beamformer steered to  $\psi = -20^\circ$ .

**FIGURE 13.8**

Beampattern of steerable beamformer steered to  $\psi = 35^\circ$ .

### 3.13.3.10 Discussion on optimization and design of broadband beamformers

The design of broadband beamformers are in many ways equivalent to designing multidimensional filters. The important property to remember however is the connection to the physical room and space. The main difficulty to be faced with in the design stems from the low frequency end of the spectrum. Since broadband beamformers are designed over many octaves, the wavelength is very large for the lower frequency end. But the distance between array elements needs to be spaced such that spatial aliasing is avoided. Accordingly, in the lower frequency end, the array operates almost like a point receiver and observes the same signal at each element resulting in poor spatial selectivity. Thus, if a constant beamwidth is desired for all frequencies, very large beamformer weights are needed to compensate for the small variations in the signals in the low frequencies. Which means that even small modeling errors will have a large impact on the response. For practical use, robustness design is very important for broadband beamformers if they are to be used without extensive calibration.

---

### 3.13.4 Broadband beamformer design using the wave equation

An alternative to the point sensor element view is that one can consider the broadband beamformer as a rigid body and perform an eigenexpansion on that body. The eigenexpansion corresponds to a spatial

Fourier transform [5]. To find these eigenexpansions on the surface of the body, one usually considers very simple geometries, such as a sphere. There are a number of advantages with this approach, one being the fact that the beampattern can be steered to any direction in 3-D space. This follows from the fact that the eigenfunctions are not a function of frequency. It is only the modal coefficients that are frequency dependent. The spherical array naturally couples to the spherical representation of the wave equation. The modal decomposition separates the problem into an orthogonal decomposition. The main results follow from the expression of the time-independent lossless Helmholtz equation in spherical coordinates

$$\nabla^2 \varphi_f(\mathbf{r}) + k^2 \varphi_f(\mathbf{r}) = 0, \quad (13.67)$$

where  $k = 2\pi f/c$  is the wave number. To obtain this equation, it is assumed that the input is a complex sinusoid. The solution to this equation in spherical coordinates [4,5] is obtained by the method of separation of variables. This solution also follows from the spherical Fourier transform.

One main result from this solution is the following relationship:

$$e^{i\alpha\mathbf{r}} = 4\pi \sum_{n=0}^{\infty} i^n j_n(kr) \sum_{m=-n}^n Y_n^m(\theta, \phi) Y_n^{m*}(\theta_s, \phi_s), \quad (13.68)$$

where  $j_n(kr)$ ,  $n \in [0, \infty]$  is the spherical Bessel functions and  $Y_n^m(\theta, \phi)$  are the orthonormal spherical harmonic eigenfunctions. The spherical harmonics are defined as

$$Y_n^m(\theta, \phi) = \sqrt{\frac{(2n+1)}{4\pi} \frac{(n-m)!}{(n+m)!}} P_n^m(\cos(\theta)) e^{im\phi}, \quad (13.69)$$

where  $P_n^m(\cos(\theta))$  are the associated Legendre functions of degree  $m$  and order  $n$  of the first kind. For the particular case of scattering on a sphere of radius  $a$ , the total acoustic pressure on the surface for an impinging plane wave from the direction  $(\theta, \phi)$  at the location  $(a, \theta_s, \phi_s)$  on the sphere is given by

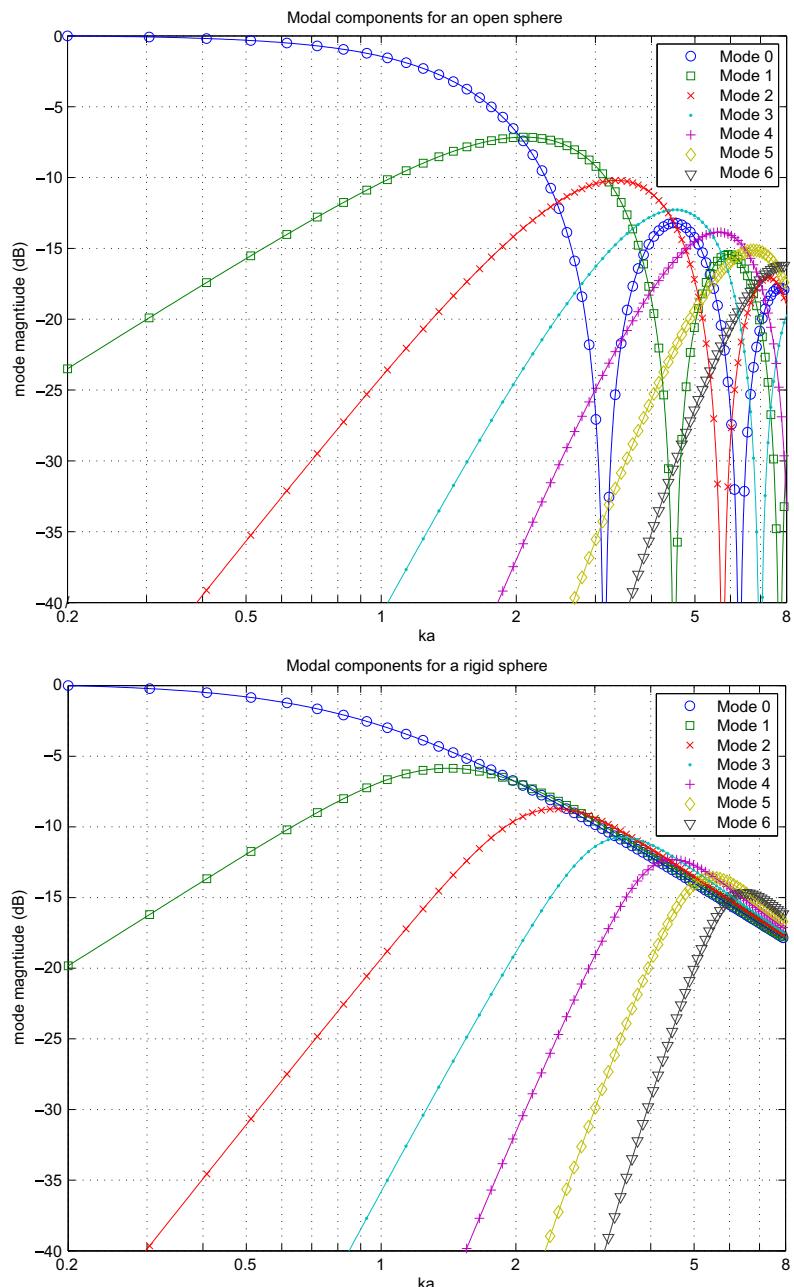
$$\varphi_s(\theta_s, \phi_s, ka, \theta, \phi) = 4\pi \sum_{n=0}^{\infty} i^n b_n(ka) \sum_{m=-n}^n Y_n^m(\theta, \phi) Y_n^{m*}(\theta_s, \phi_s), \quad (13.70)$$

where the normalized modal coefficients are given by

$$b_n(ka) = \begin{cases} j_n(ka) - \frac{j'_n(ka)}{h_n^{(2)'}(ka)} h_n^{(2)}(ka) & \text{for a rigid sphere,} \\ j_n(ka) & \text{for an open sphere,} \end{cases} \quad (13.71)$$

where  $h_n^{(2)}(ka)$  is the spherical Hankel function of second kind and  $(\cdot)'$  denotes the derivative. From the expression (13.70), it can be seen that the frequency dependency is in the modal coefficients. For the rigid sphere, the first term in the modal component describes the incoming wave and the second term describes the scattered wave. The boundary condition for a rigid sphere is that the radial velocity is zero. This condition gives the above modal coefficients for a rigid sphere. For non-rigid spheres, the boundary condition changes but the solution will remain in the same form but with different modal coefficients. The first seven modal functions have been plotted as a function of  $ka$  in Figure 13.9 for an open sphere and a rigid sphere, respectively.

As can be seen from (13.15), the sound pressure field varies with the incoming wave and the position on the sphere. If the spherical harmonic function  $Y_n^m(\Omega_s)$  can be formed on the surface of the sphere, it

**FIGURE 13.9**

Top: Modal components  $b_n(ka)$  for an open sphere. Bottom: Modal components  $b_n(ka)$  for a rigid sphere.

can operate as a eigenfunction beamforming receiver that matches to the incoming wave. Thus consider the output of the eigenfunction beamformer for an incident plane wave, and considering only one spherical harmonic function  $Y_n^m(\Omega_s)$ , then we have

$$\begin{aligned}\varphi_{n,m}(ka, \Omega_0) &= \int_{\Omega_s} \varphi_{s,f}(\Omega_s, ka, \Omega_0) Y_n^m(\Omega_s)^* d\Omega_s \\ &= \int_{\Omega_s} 4\pi \sum_{n'=0}^{\infty} i^n b_n(ka) \sum_{m'=-n'}^{n'} Y_{n'}^{m'}(\Omega_0) [Y_{n'}^{m'}(\Omega_s)]^* Y_n^m(\Omega_s)^* d\Omega_s \\ &= 4\pi (-i)^n b_n(ka) Y_n^m(\Omega_0)^*,\end{aligned}\quad (13.72)$$

where  $\int_{\Omega_s} d\Omega_s$  represents an integration over the surface of the sphere. This equation follows from the orthonormal property of the spherical harmonics functions:

$$\int_{\Omega_s} [Y_{n'}^{m'}(\Omega_s)]^* Y_n^m(\Omega_s) d\Omega_s = \delta(n - n', m - m'), \quad (13.73)$$

where  $\delta(n, m)$  denotes the Kronecker delta function. Also, the impinging wave can be recovered from the inverse of the operation in (13.72), which corresponds to an inverse spherical harmonic Fourier transform:

$$\varphi_s(ka, \Omega_0, \Omega_s) = 4\pi \sum_{n=0}^{\infty} \sum_{m=-n}^n \varphi_{n,m}(ka, \Omega_0) Y_n^m(\Omega_s). \quad (13.74)$$

Now assume that there are  $D$  incoming waves from directions  $\Omega_0, \Omega_1, \dots, \Omega_{D-1}$ , and noise impinging on the sphere. Then the sound pressure on the sphere is given by

$$x(ka, \Omega_s) = \sum_{d=0}^{D-1} \varphi_s(ka, \Omega_d, \Omega_s) S_d(\omega) + N(\omega, \Omega_s), \quad (13.75)$$

where  $S_d(\omega)$  is the spectrum of the wave from direction  $\Omega_d$  and the noise spectrum is denoted  $N(\omega, \Omega_s)$ . Then, the spherical Fourier transform can be taken on the sphere, resulting in

$$\begin{aligned}x_{n,m}(ka) &= \int_{\Omega_s} x(ka, \Omega_s) Y_n^m(\Omega_s) d\Omega_s \\ &= \int_{\Omega_s} \left( \sum_{d=0}^{D-1} \varphi_s(ka, \Omega_d, \Omega_s) S_d(\omega) + N(\omega, \Omega_s) \right) Y_n^m(\Omega_s) d\Omega_s \\ &= \sum_{d=0}^{D-1} \varphi_{n,m}(ka, \Omega_d) S_d(\omega) + N_{n,m}(\omega).\end{aligned}\quad (13.76)$$

We are now ready to discuss the impact of an aperture weighting function. The aperture weighting function on the sphere operates like beamformer weights, where it is applied as a window over the sphere and the output is obtained by integrating over the sphere as follows:

$$y(ka) = \int_{\Omega_s} x(ka, \Omega_s) w^*(k, \Omega_s) d\Omega_s. \quad (13.77)$$

Both  $x(ka, \Omega_s)$  and  $w(k, \Omega_s)$  can be expanded into the spherical harmonics domain using the spherical Fourier transform. This operation results in the following:

$$y(ka) = \sum_{n=0}^{\infty} \sum_{m=-n}^n x_{n,m}(ka) w_{n,m}^*(k). \quad (13.78)$$

This summation is a weighting in the spherical harmonics domain and is usually called phase-mode processing. Furthermore, the directivity pattern for the beamformer is defined as the response of the beamformer to a unit signal from any direction of interest  $\Omega$ , thus

$$\begin{aligned} D(k, \Omega) &= \int_{\Omega_s} \varphi(ka, \Omega, \Omega_s) w^*(k, \Omega_s) d\Omega_s \\ &= \sum_{n=0}^{\infty} \sum_{m=-n}^n \varphi_{n,m}(ka, \Omega) w_{n,m}^*(k). \end{aligned} \quad (13.79)$$

This expression gives the response for a unit complex sinusoid. It is common to split the weighting vector into one frequency dependent component and one spatial component such that

$$w_{n,m}^*(k) = c_n(k) Y_n^m(\Omega_0). \quad (13.80)$$

This choice of weighting results in a rotational symmetric response [8]. Since the expansion of the unit signal from any point of interest  $\Omega$  is given by

$$\varphi_{n,m}(ka, \Omega) = 4\pi(-i)^n b_n(ka) Y_n^m(\Omega)^*$$

we can insert the expression (13.80) in (13.79), which results in

$$\begin{aligned} D(k, \Omega) &= \sum_{n=0}^{\infty} \sum_{m=-n}^n 4\pi i^n b_n(ka) c_n(k) Y_n^m(\Omega)^* Y_n^m(\Omega_0) \\ &= \sum_{n=0}^{\infty} (-i)^n b_n(ka) c_n(k) (2n+1) P_n(\cos(\Theta)). \end{aligned} \quad (13.81)$$

This expression follows from the Legendre polynomial addition theorem [9] which states the following:

$$P_n(\cos(\Theta)) = \frac{4\pi}{2n+1} \sum_{m=-n}^n Y_n^m(\Omega)^* Y_n^m(\Omega_0), \quad (13.82)$$

where  $\Theta$  is the angle between two unit vectors from direction  $\Omega$  and  $\Omega_0$ . Note that when  $\Omega = \Omega_0$ , the response should be one with the correct scaling. Thus, one can steer the beamformer to the position  $\Omega_0$ . The design of a given beampattern can be achieved by designing the coefficients  $c_n(k)$ . One simple way to do that is to set them as the inverse of the modal components. In view of the properties of the modal components, those coefficients need to provide a high gain at low frequencies. To get frequency independent beams over a wide frequency band, one needs to equalize the modal components, and as the frequency becomes lower the gain needs to increase. The zero modal component gives an omnidirectional response. To have high resolution, high order components are needed. From the modal function plots in Figure 13.9, it can be seen that the open sphere is more difficult to control as it has nulls and larger variations.

Another practical aspect is that the sound pressure wave on the sphere needs to be sampled. The sampling is performed by sensors which are usually considered to be point sources. An exact representation between the sampled information and the continuous information is obtained if the spherical Fourier coefficients can be calculated from the spatial samples without errors such as

$$x_{n,m}(ka) = \sum_{r=1}^R \alpha_s x(ka, \Omega_r) Y_n^m(\Omega_r), \quad (13.83)$$

where  $r \in [1, \dots, R]$  are the sample points on the sphere and  $\alpha_s$  is a scaling depending on the sampling. From (13.74), this condition can also be expressed as

$$\sum_{r=1}^R \alpha_s [Y_{n'}^{m'}(\Omega_r)]^* Y_n^m(\Omega_r) = \delta(n - n', m - m'). \quad (13.84)$$

Thus, the sampling points should be chosen such that the orthogonality properties are fulfilled. A common approach which fulfills these properties is the equiangle sampling. This sampling is performed in such a way that the sphere is sampled uniformly in  $\theta$  and  $\phi$ . In this case the parameter is given by  $\alpha_s = 4\pi/R$ . This sampling does not lead to a case with uniform distance between points and will not give the lowest number of points. However it gives rotational invariance which is practical when the array is moved. Now given that (13.84) is fulfilled, the discretized processing can be used.

The actual processing can be separated into four parts:

1. take the STFT, Short-time Fourier Transform, of the A/D converted signal for each sensor element; this gives an approximation of  $x(ka, \Omega_r)$ ;
2. multiply each channel with the sampled spherical harmonic  $Y_n^m(\Omega_r)$ , then sum up the components according to (13.83) to get  $x_{n,m}(ka)$ ;
3. multiply with the steered spherical harmonic  $Y_n^m(\Omega_0)$ ;
4. multiply with the beamformer coefficients and sum according to (13.78);
5. take the ISTFT, Inverse STFT, of the output.

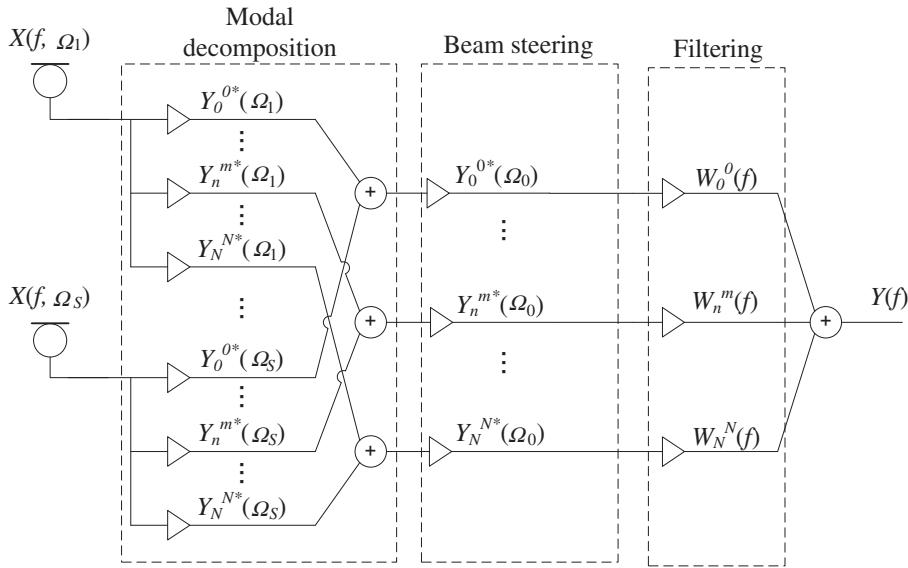
This provides the basic theoretical framework for the design of a broadband beamformer in spherical coordinates. Now, compared to the conventional broadband beamforming case, the difference is that it is performed in frequency domain and that the signals are pre-beamformed using the spherical harmonic functions, which have different patterns depending on the order according to Figure 13.10.

### 3.13.4.1 Broadband beamformer design based on the Wave equation

The design equation used to design a beamformer is given by the directivity pattern

$$D(\omega, \Omega, \Omega_0) = \sum_{n=0}^{\infty} (-i)^n b_n(\omega a/c) c_n(\omega) (2n + 1) P_n(\cos(\Theta)). \quad (13.85)$$

Here, (13.81) was used but the argument was changed from wave number to frequency. The relationship between wave number and frequency is  $k = \omega/c$  and thus  $\omega = kc$ . There are different approaches to find weights for this beamformer. As mentioned, the direct way is just to compensate the modal components; by doing so, one can get a frequency invariant beamformer over a certain bandwidth. However, for a large bandwidth, some of the modal components become very small and thus one needs

**FIGURE 13.10**

Wave domain beamformer.

to have a high gain in  $c_n(\omega)$  for low frequencies. This limits the number of eigenbeams one can use. To be able to stretch the frequency range, one needs low noise sensor elements. Thus, an alternative to the direct approach is to use an optimization method to design the weights. To do this, form a cost function based on a desired response and compare that with the response given by (13.85). This expression is then optimized with respect to the filter weights  $c_n(\omega)$ . The first step is to limit the number of spherical harmonic functions used in the optimization. They are set to be less than  $S \leq (N + 1)^2$ . Since the expression (13.85) is a finite sum, it can be rewritten in vector form as

$$D(\omega, \Omega) = \mathbf{W}^H(\omega) \mathbf{d}(\omega, \Omega, \Omega_0), \quad (13.86)$$

where

$$\begin{aligned} \mathbf{d}(\omega, \Omega, \Omega_0) = & [b_0(\omega a/c)(1)P_0(\cos(\Theta)), (-i)b_1(\omega a/c)3P_1(\cos(\Theta)), \dots, \\ & (-i)^N b_N(\omega a/c)]^T. \end{aligned} \quad (13.87)$$

Also note that the modal components are frequency dependent but the spatial component is only depending on the spatial properties, thus

$$\mathbf{d}(\omega, \Omega, \Omega_0) = \mathbf{b}(\omega) \mathbf{P}(\Theta), \quad (13.88)$$

where

$$\mathbf{b}(\omega) = \begin{bmatrix} b_0(\omega a/c) \\ b_1(\omega a/c) \\ \vdots \\ b_N(\omega a/c) \end{bmatrix}$$

and

$$\mathbf{P}(\Theta) = \begin{bmatrix} P_0(\cos(\Theta)) & 0 & \cdots & 0 \\ 0 & (-i)3P_1(\cos(\Theta)) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (-i)^N(2N+1)P_N(\cos(\Theta)) \end{bmatrix}.$$

Thus,  $D(\omega, \Omega)$  can be rewritten as

$$D(\omega, \Omega) = \mathbf{W}^H(\omega)\mathbf{b}(\omega)\mathbf{P}(\Theta). \quad (13.89)$$

From this equation it can be seen that one term is not depending on frequency but is constant for all frequencies. The other term contains all the frequency dependency. Also, note that the term  $\mathbf{P}(\Theta)$  is depending on the shape of the beamformer and  $\mathbf{b}(\omega)$  determines the array response for frequency  $\omega$ . Based on this simplified beamforming approach, which is very common to use for spherical arrays, it becomes straightforward to formulate the performance measure as the error between an actual frequency response  $D(\omega, \Omega)$  and a desired frequency response  $D_d(\omega, \Omega)$ ,

$$\xi(\omega, \Omega) = D(\omega, \Omega) - D_d(\omega, \Omega). \quad (13.90)$$

Based on this expression, any of the already presented optimization methods can be used to find the optimal weights  $W_n(\omega)$ . Thus from (13.90) it immediately follows that optimum weights in WLS sense can be found as

$$\begin{aligned} \min_{\mathbf{c}} J_{\text{WLS}}(\mathbf{c}) &= \min_{\mathbf{c}} \int_{\omega_R} \int_{\Omega_R} V(\omega, \Omega) |\xi(\omega, \Omega)|^2 d\Omega d\omega \\ &= \min_{\mathbf{c}} \left( \mathbf{c}^H \mathbf{A}_{s, \text{WLS}} \mathbf{c} - 2\Re \left\{ \mathbf{a}_{s, \text{WLS}}^H \mathbf{c} \right\} + b_{s, \text{WLS}} \right), \end{aligned} \quad (13.91)$$

where  $\omega_s$  is the range of frequencies and  $\Omega_s$  is angular region in  $(\theta, \phi)$  of the design, and

$$\mathbf{A}_{s, \text{WLS}} = \int_{\omega_s} \int_{\Omega_s} V(\omega, \Omega) \mathbf{f}(\omega, \Omega, \Omega_0) \mathbf{f}^H(\omega, \Omega, \Omega_0) d\Omega d\omega, \quad (13.92)$$

$$\mathbf{a}_{s, \text{WLS}} = \int_{\omega_s} \int_{\Omega_s} V(\omega, \Omega) \mathbf{f}(\omega, \Omega, \Omega_0) D_d^H(\omega, \Omega) d\Omega d\omega, \quad (13.93)$$

$$b_{s, \text{WLS}} = \int_{\omega_s} \int_{\Omega_s} V(\omega, \Omega) D_d^H(\omega, \Omega) D_d(\omega, \Omega) d\Omega d\omega, \quad (13.94)$$

where  $\Omega_0$  is the steering direction which can be set to  $(0, 0)$  in the design phase. In a similar way, the weight design can be generalized to the other norm criterion including the robust formulations.

### 3.13.4.2 Time domain design of spherical broadband beamformer

As a matter of fact, (13.89) also forms the basis for the design of a time domain beamformer. In this case, express  $W_n(\omega) = \mathbf{w}_n^T \mathbf{e}(\omega)$  where  $\mathbf{w}_n$  is a vector of FIR filter weights and  $\mathbf{e}(\omega)$  is the FIR filter response vector. Since the design equation is the same as for a general broadband beamformer, the same techniques can be used. Thus the following expression can be used:

$$D(\omega, \Omega) = \mathbf{w}^T \mathbf{e}(\omega) \times \mathbf{f}(\omega, \Omega, \Omega_0), \quad (13.95)$$

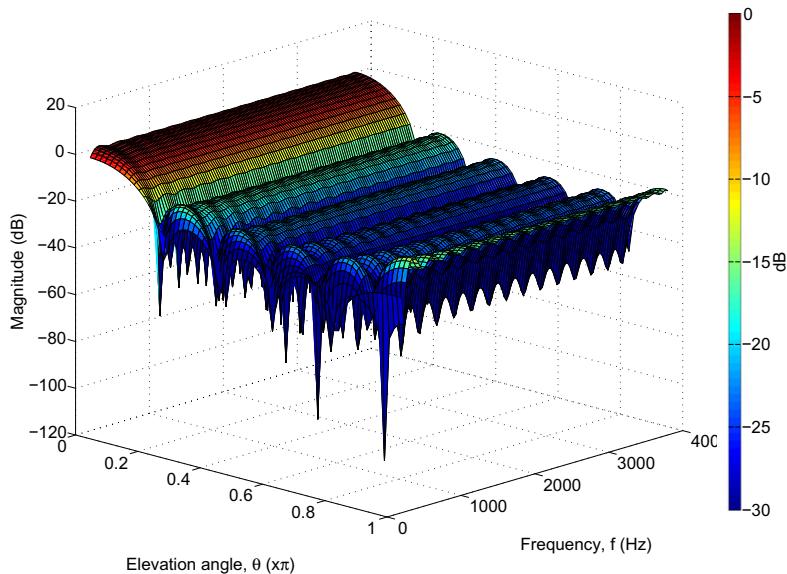
where  $\mathbf{w} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N]^T$ . This expression can now be used in the error equation and consequently in the design equations. Thus, the design of a broadband beamformer using spherical harmonics can be performed either in frequency domain or time domain. Note that the spherical harmonics are independent of frequency the only frequency dependency is in the modal components. This means that the time domain design can be viewed as an approximation of the frequency domain expression.

### 3.13.4.3 Design examples

In order to illustrate the design formulation discussed, a design example is presented. The design parameters used are as listed in Table 13.4, where the microphones are sampled using the equiangle sampling

**Table 13.4** Design Parameters

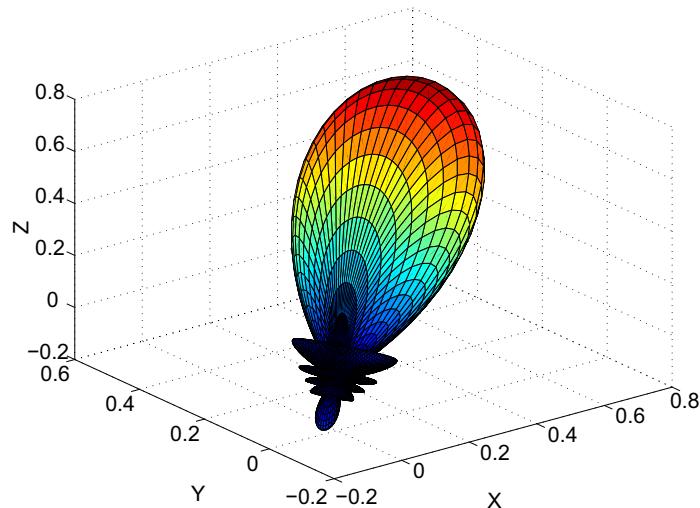
Parameters	Value
Highest spherical harmonics order, $N$	5
Number of sensor, $P$	144
FIR filter length, $L$	64
Sampling frequency, $f_s$	8000 Hz
Spectral passband, $\Omega_{pb}$	[200, 3800] Hz
Radius of spherical array, $a$	4 cm



**FIGURE 13.11**

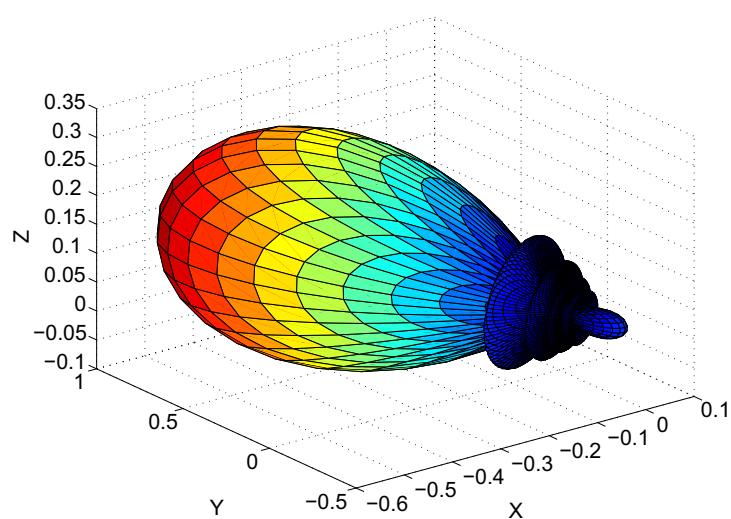
Beampattern steered to  $\Omega_0 = (0, 0)$ .

scheme, which requires a total of 144 microphones to resolve up to 5th order spherical harmonics [8, 10, 11]. Figure 13.11 shows the normalized beampattern steered to  $\Omega_0 = (0, 0)$  (azimuth angle  $\phi$  is not shown in the figure since the beampattern is invariant to azimuth angle). The frequency invariant property of the design is clearly highlighted in this figure. Figures 13.12 and 13.13 show the normalized



**FIGURE 13.12**

Beampattern at  $f = 2$  kHz steered to  $\Omega_0 = (40, 30)$ .



**FIGURE 13.13**

Beampattern at  $f = 2$  kHz steered to  $\Omega_0 = (80, 120)$ .

beampattern (at  $f = 2$  kHz) steered to  $\Omega_0 = (40, 30)$  and  $\Omega_0 = (80, 120)$ , respectively. These figures clearly shows the steerability of the design while at the same time, the beampattern is maintained.

### 3.13.5 Optimum and adaptive broadband beamforming

For the optimum beamformer, the weights are formed based on properties of the data presented at each sensor element. There are a number of methods available for the design of the broadband beamformer weights. Since the design is based on measured data where the SOI is usually not available, some *a priori* information about the source is needed to extract it. This is in contrast to blind signal separation which assumes less *a priori* information, merely independence or sparsity. The most common way to apply optimum beamforming is to use a constrained method. This type of methods relies on geometrical constraints, where the location of the source is known either perfectly or almost perfectly. It also requires a free field or almost free field environment and good calibration of the array. This limits the use of the standard techniques in practical scenarios since model errors are evident. Accordingly, various different robustness constraints have been developed. It is also possible to combine optimum beamforming with the methods discussed above to view the problem as a mix of an optimal and optimum beamforming problem. In this section, some optimum beamforming methods will be reviewed.

#### 3.13.5.1 Common signal modeling

In order to provide a consistent description, a general signal model and propagation model have been used. It is general in the sense that microphone elements and sources can be placed arbitrarily with any spectral content. The scenario is assumed to consist of  $M$  different point signal sources  $s_m(t)$ ,  $m = 1, \dots, M$  with spectral densities  $R_{s_m s_m}(\omega)$ . The sources are assumed to be mutually uncorrelated, i.e., the cross power spectral density  $R_{s_l s_m}(\omega)$  is zero if  $l \neq m$ . The waves from the  $M$  sources impinge on an array of  $P$  microphone elements, each corrupted with non-directional diffuse noise  $n_l(t)$  (the simplest case is when the noise sources are independent). For a single point to single point transmission, the propagation can be modeled using a transfer function (Green function) between source  $m$  and array element  $p$ , denoted  $H_{m,p}(\omega)$ , and is either obtained from measurements or a model. In the model, a spherical source in a free field and homogeneous medium is the simple model usually used. In a real world scenario, it is only possible to work on measured data. The sensor element signals  $x_p(t)$  are

$$x_p(t) = \sum_{m=1}^M s_m(t) * h_{m,p}(t) + n_p(t) \quad p = 1, \dots, P, \quad (13.96)$$

where  $*$  denotes convolution. This means that each source signal is described as a point source and is filtered by a linear time invariant system. By this is meant that the variations of the acoustic channel are small relative to the update of optimal filters or adaptive filters. In the sequel, all the signals are assumed to be band limited and sampled.

#### 3.13.5.2 Linearly Constrained Minimum Variance beamforming

The Linearly Constrained Minimum Variance (LCMV) beamforming minimizes the output of a broadband array while maintaining a constant gain constraint towards the SOI. In the time domain implementation, the signal  $x_p(t)$  at each element is sampled and is filtered by an FIR filter of length  $L$ .

The output of the beamformer is given as

$$y(k) = \sum_{p=0}^{P-1} \mathbf{w}_p^T \mathbf{x}_p(k) = \mathbf{w}^T \mathbf{x}(k), \quad (13.97)$$

where  $P$  is the number of elements and with the weight vector

$$\mathbf{w}_p = [w_{p,0} \ w_{p,1} \ \cdots \ w_{p,L-1}]^T,$$

and the input data vector

$$\mathbf{x}_p(k) = [x_p(k) \ x_p(k-1) \ \cdots \ x_p(k-L+1)]^T$$

has length  $L$ . Stacking weight and data vectors will give vectors of length  $PL \times 1$ .

The expression to be minimized is the Mean Square Error  $E[|y(k)|^2]$  with respect to the filter weights

$$r_{yy}(0) = E[|y(k)|^2] = \mathbf{w}^T E[\mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w}, \quad (13.98)$$

where  $E[\cdot]$  is the expectation operator and the minimum is obviously zero. To avoid this trivial solution, a constraint needs to be included. This constraint can be formed such that  $y_{s_0}(k) = s_0(k - k_0)$ , i.e., the output is distortion free apart from a possible delay. The SOI is assumed to be the zeroth source. To create such a constraint, the impulse response  $h_{m,p}(t)$  between the SOI and each array element needs to be known. To derive the constraint for a broadband LCMV beamformer, assume that the  $P$  impulse responses  $h_{m,p}(t)$  can be represented by FIR responses  $h_{m,p}(k)$  of length  $N$ . By using convolution matrix techniques, the response between the SOI and the output of the beamformer can be expressed as

$$\mathbf{H}\mathbf{w} = \mathbf{g}, \quad (13.99)$$

where the convolutional matrix  $\mathbf{H}$  of size  $(N + L - 1) \times (PL)$  is given by

$$\mathbf{H} = [\mathbf{H}_{0,0} \ \mathbf{H}_{0,1} \ \cdots \ \mathbf{H}_{0,P-2} \ \mathbf{H}_{0,P-1}], \quad (13.100)$$

where the convolutional matrix from the SOI to sensor element  $p$  is given by

$$\mathbf{H}_{0,p} = \begin{bmatrix} h_{0,p}(0) & 0 & \cdots & 0 & 0 \\ h_{0,p}(1) & h_{0,p}(0) & \cdots & 0 & 0 \\ h_{0,p}(2) & h_{0,p}(1) & h_{0,p}(0) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{0,p}(N-1) \end{bmatrix}. \quad (13.101)$$

Here,  $\mathbf{w}$  is the same stacked vector as before and  $\mathbf{g}$  is the resulting response vector. To achieve the distortion free property, the output  $\mathbf{g}$  should have an entry that is one in position  $k_0$  and zero otherwise.

The LCMV problem can now be stated as

$$\min_{\mathbf{w}} \mathbf{w}^T \mathbf{R}_{xx} \mathbf{w} \quad (13.102)$$

subject to  $\mathbf{H}\mathbf{w} = \mathbf{g}$ ,

where

$$\mathbf{R}_{\mathbf{xx}} = E[\mathbf{x}(k)\mathbf{x}^T(k)]. \quad (13.103)$$

The solution to this problem can be found analytically

$$\mathbf{w} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{H}^T(\mathbf{H}\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{H}^T)^{-1}\mathbf{g}. \quad (13.104)$$

As can be seen from (13.99), this constraint may be very restrictive and degrees of freedom are lost. This prevents the beamformer from suppressing directional interference, so called jammers. This is particularly the case when the propagation impulse response has high complexity. In most studies and as a first approximation, one would only consider the direct path and use a pre-steering filter to align the source signal to the correct direction of arrival of the SOI. As the SOI signals on the array are now all aligned, the medium's impulse response can be ignored. This corresponds to the free field scenario as stated in Section 3.13.2.1; this assumption leads to a convolutional matrix  $\mathbf{H}$  of size  $(L - 1) \times (PL)$  and the constraint simplifies to the average of the column weights in the beamformer FIR matrix being zero for all columns apart from the  $k_0$  column. This highlights the difficulty of using this technique in reverberant rooms or when there is a mismatch between model and real signals, since the constraint is only valid under ideal situations. To improve the response for those scenarios, extra constraints such as a derivative constraints or quadratic constraints are usually added. These will lower the degrees of freedom of the optimum beamformer. As an alternative to the time domain approach, the problem can be formulated in frequency domain combined with multi-rate techniques for the processing. That will give a number of narrowband problems which are usually easier to handle in practical scenarios. From this discussion, we can also see the challenge in using broadband beamforming in reverberant environment. For example, if the impulse responses from the SOI to every element can be determined accurately, then for high reverberations the channels are long and thus  $(N + L - 1) > (PL)$  and there is no degrees of freedom to do beamforming, only an inverse filter of the propagation response.

### 3.13.5.3 LCMV in frequency domain

The frequency domain formulation of the LCMV broadband beamformer follows directly from the narrow-band formulation. The input data  $\mathbf{x}_p(k)$  is transformed to a time-frequency representation  $X_p(\omega, l)$ . Assuming enough resolution in the time-frequency representation,  $X_p(\omega, l)$  can be expressed as

$$X_p(\omega, l) = \sum_{m=0}^{M-1} S_m(\omega, l) H_{m,p}(\omega) + N_p(\omega), \quad p = 0, 1, \dots, P-1. \quad (13.105)$$

Based on this assumption, the beamformer is formed in each frequency band

$$Y(\omega, l) = \sum_{p=0}^{P-1} W_p^H(\omega) X_p(\omega, l) = \mathbf{W}^H(\omega) \mathbf{X}(\omega, l), \quad (13.106)$$

where  $\mathbf{W}(\omega)$  is the beamformer response for frequency component  $\omega$ , and  $\mathbf{X}(\omega, l)$  is the input data vector. The LCMV objective is to minimize the following expression:

$$S_{YY}(\omega) = \mathbf{W}^H(\omega) \mathbf{S}_{\mathbf{XX}}(\omega) \mathbf{W}(\omega) \quad (13.107)$$

for each frequency component. The matrix,  $\mathbf{S}_{\mathbf{XX}}(\omega)$  is defined as

$$\mathbf{S}_{\mathbf{XX}}(\omega) = \begin{pmatrix} \mathbf{S}_{X_0 X_0}(\omega) & \mathbf{S}_{X_0 X_1}(\omega) & \cdots & \mathbf{S}_{X_0 X_{P-1}}(\omega) \\ \mathbf{S}_{X_1 X_0}(\omega) & \mathbf{S}_{X_1 X_1}(\omega) & \cdots & \mathbf{S}_{X_1 X_{P-1}}(\omega) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{X_{P-1} X_0}(\omega) & \mathbf{S}_{X_{P-1} X_1}(\omega) & \cdots & \mathbf{S}_{X_{P-1} X_{P-1}}(\omega) \end{pmatrix}. \quad (13.108)$$

The constraint is formed for each frequency component such that the SOI is not distorted. To achieve this, the constraint is given by

$$\mathbf{H}_0^H(\omega)\mathbf{W}(\omega) = G(\omega). \quad (13.109)$$

The frequency domain LCMV problem can now be stated as follows:

$$\begin{aligned} & \min_{\mathbf{W}} \mathbf{W}^H(\omega)\mathbf{S}_{\mathbf{XX}}(\omega)\mathbf{W}(\omega) \\ & \text{subject to } \mathbf{H}_0^H(\omega)\mathbf{W}(\omega) = G(\omega), \end{aligned} \quad (13.110)$$

where the solution to this problem can be found analytically

$$\mathbf{W}(\omega) = \frac{\mathbf{S}_{\mathbf{XX}}^{-1}(\omega)\mathbf{H}_0(\omega)}{G(\omega)\mathbf{H}_0^H(\omega)\mathbf{S}_{\mathbf{XX}}^{-1}(\omega)\mathbf{H}_0(\omega)}. \quad (13.111)$$

In the simplest scenario, it is assumed that the beamformer is aligned to the SOI and the environment is homogeneous and free field. In this case the constraint vector is given by  $\mathbf{H}_0(\omega) = [1, 1, \dots, 1]^T$  and it will be frequency independent. That assumes perfect pre-steering. Thus to make this type of processing more robust, several constraints around the direction where the SOI is located are included; these constraints can either be derivative constraints or several point constraints. Also, quadratic constraints that model uncertainty in the model have been suggested.

### 3.13.5.4 Generalized Sidelobe Canceler

The LCMV problem can be converted to an unconstrained problem. This unconstrained beamformer is usually called a Generalized Sidelobe Canceler (GSC). In its simplest form, it can be viewed as a constrained beamformer that has been converted to a non-constrained design by means of a blocking matrix. Thus the problem has been separated into two parts: one is a fixed beamforming part that determines the response for the desired source, and the other part blocks the desired source from entering into the canceler. The relationship between LCMV and GSC follows from straightforward algebra. It relies on the fact that the weights of the LCMV beamformer can be decomposed into two parts, one part fulfilling the constraint and another part that can be minimized.

The LCMV constraint matrix  $\mathbf{H}$  is of size  $(N + L - 1) \times (PL)$  and should have a rank given by the minimum of  $(N + L - 1)$  and  $(PL)$ . Assume that the rank is  $(N + L - 1)$ , then a new matrix can be constructed

$$\mathbf{U} = [\mathbf{H}^T | \mathbf{H}_d] \quad (13.112)$$

which is full rank and thus is invertible, and  $\mathbf{H}\mathbf{H}_a = \mathbf{0}$ . This means that  $\mathbf{H}_a$  is the orthogonal complement to  $\mathbf{H}^T$ . Accordingly,  $\mathbf{H}_a$  can be found from the following relationship:

$$\mathbf{H}_a = (\mathbf{I} - \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}). \quad (13.113)$$

Now assume the following:

$$\mathbf{w} = \mathbf{U}\mathbf{q}, \quad (13.114)$$

where  $\mathbf{q} = [\mathbf{v} - \mathbf{w}_a]^T$ . Using this relationship with (13.112) yields

$$\mathbf{w} = \mathbf{H}^T\mathbf{v} - \mathbf{H}_a\mathbf{w}_a \quad (13.115)$$

and furthermore  $\mathbf{w}$  fulfills the LCMV constraint, thus

$$\mathbf{H}\mathbf{w} = \mathbf{H}\mathbf{H}^T\mathbf{v} - \mathbf{H}\mathbf{H}_a\mathbf{w}_a = \mathbf{H}\mathbf{H}^T\mathbf{v} = \mathbf{g}. \quad (13.116)$$

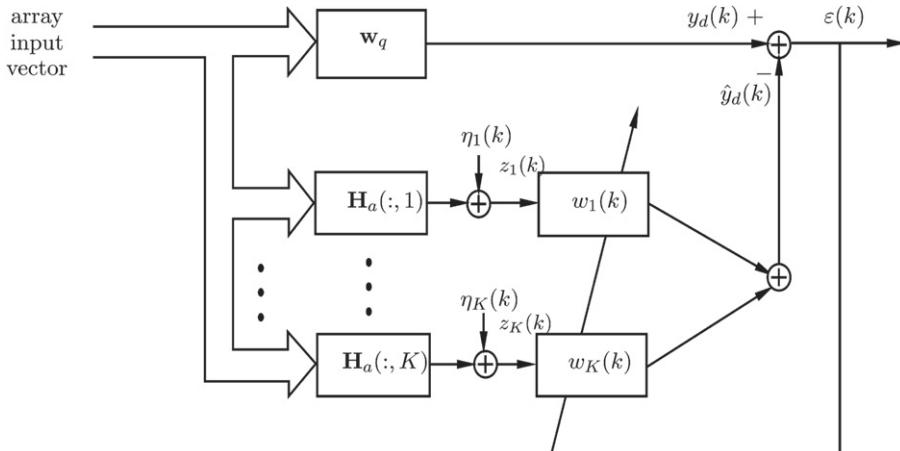
It follows that

$$\mathbf{v} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{g}. \quad (13.117)$$

Inserting (13.117) in (13.115), the final decomposition is found as

$$\mathbf{w} = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{g} - \mathbf{H}_a\mathbf{w}_a = \mathbf{w}_q - \mathbf{H}_a\mathbf{w}_a. \quad (13.118)$$

This is the GSC decomposition where  $\mathbf{w}_q$  are the optimum beamformer weights depending only on the constraint, and  $\mathbf{w}_a$  are the weights that are free to minimize the cost function. Accordingly, the GSC consists of three parts (see Figure 13.14): one upper fixed beamformer  $\mathbf{w}_q$ , a blocking matrix  $\mathbf{H}_a$ , and an interference canceler  $\mathbf{w}_a$ .



**FIGURE 13.14**

Generalized Sidelobe Canceler structure.

It operates in the following way. The input signal vector  $\mathbf{x}(k)$  will be filtered by the upper beamformer  $\mathbf{w}_q$ , which forms a beam towards the SOI, creating an output

$$y_d(k) = \mathbf{w}_q^T \mathbf{x}(k). \quad (13.119)$$

This output  $y_d(k)$  has been filtered towards the SOI and in its simplest form, will correspond to a summing beamformer. In its general form, it will consist of stacked FIR filters which form a broadband optimized beamformer. The blocking matrix  $\mathbf{H}_a$  forms beams that are orthogonal to the desired signal. Thus, the input to the interference canceler should contain only undesired signals,

$$\mathbf{z}(k) = \mathbf{H}_a^T \mathbf{x}(k), \quad (13.120)$$

where  $\mathbf{z}(k)$  is a vector of length  $(P - 1)L - N + 1$ . The weights  $\mathbf{w}_a$  are obtained by minimizing the following expression:

$$\min_{\mathbf{w}_a} E \left[ |y_d(k) - \mathbf{w}_a^T \mathbf{z}(k)|^2 \right]. \quad (13.121)$$

The original optimum beamformer problem has now been turned into an interference cancellation problem. One interpretation of the signal blocking matrix is that it consists of  $PL - N - L + 1$  lower beamformers  $\mathbf{h}_k$  implementing the signal blocking matrix. In theory, this formulation will provide a blocking of the SOI. In practice this is not feasible, since it requires very accurate knowledge of the Green function from the desired source to the input of the lower beamformers. However, based on this unconstrained formulation a number of practical ways to approximate this formulation have been suggested. One such way is to view the problem as a filter design problem where the desired signal should be suppressed below a certain level. Where this level is determined by how much of the desired signal leaks into the cancellation structure.

As an alternative to this fixed constrained approach, one can form various adaptive schemes to obtain the blocking matrix. More details on these will be found in the further reading section.

In summary, the broadband GSC provides a very good suppression of interfering signals but the signal distortion is difficult to control and also the calibration for a combined array is difficult. This is because the design model and real life scenarios need to match very accurately, which is difficult to achieve in practical scenarios.

### 3.13.5.5 Generalized Sidelobe Canceler in frequency domain

Similarly to the LCMV, the GSC can be directly transferred to the frequency domain. The derivations for the time domain are still valid and the weight vector can be reformulated as follows:

$$\begin{aligned} \mathbf{W}(\omega) &= \mathbf{H}_0(\omega) \left( \mathbf{H}_0^H(\omega) \mathbf{H}_0(\omega) \right)^{-1} G(\omega) - \mathbf{H}_a(\omega) \mathbf{W}_a(\omega) \\ &= \mathbf{W}_q(\omega) - \mathbf{H}_a(\omega) \mathbf{W}_a(\omega). \end{aligned} \quad (13.122)$$

The blocking matrix is given by  $\mathbf{H}_a(\omega) = (\mathbf{I} - \mathbf{H}_0(\omega) (\mathbf{H}_0^H(\omega) \mathbf{H}_0(\omega))^{-1} \mathbf{H}_0^H(\omega))$ . Since the response vector of the array for the SOI corresponds to  $\mathbf{H}_0(\omega)$ , it follows that the response from the SOI to the output is given by  $G(\omega)$ . In this case, the spectral density of the error

$$\epsilon(\omega) = Y_d(\omega) - \mathbf{W}_d(\omega)^H \mathbf{Z}(\omega) \quad (13.123)$$

should be minimized with respect to  $\mathbf{W}_a(\omega)$ . The critical part in order to avoid SOI distortion is to have no SOI signal leaking into  $\mathbf{Z}(\omega)$ . In practice, the most difficult situation to use the GSC is for high SNR scenario, when small leakage of SOI signal is able to cancel parts of the SOI signal. To combat this, robust constraints are necessary which are either similar to those suggested in the time domain case or norm constraints on the weights. By applying norm constraints on the weights, there will be a cost on large weights and the optimum filter will be more robust.

### 3.13.5.6 Wiener filter

The Wiener filter can also be used as an optimum broadband beamformer. For a broadband FIR beamformer in time domain, the objective can be formulated as

$$\mathbf{w}_{\text{opt}} = \arg \left\{ \min_{\mathbf{w}} E[(s_0(k) - y(k))^2] \right\}, \quad (13.124)$$

where the output  $y(k)$  from the beamformer is given in (13.97), and the SOI observation  $s_0(k)$  is the source signal. Thus, the theory states that the source signal  $s_0(k)$  needs to be available. In practice it is desirable to use only sensor observations instead. Since the source signal is usually not accessible, an estimate is used instead. As an alternative, it is sometimes possible to use a training signal or integrate a model.

The optimal Wiener weights are given by

$$\mathbf{w}_{\text{opt}} = [\mathbf{R}_{\mathbf{xx}}]^{-1} \mathbf{r}_{\mathbf{x}s}, \quad (13.125)$$

where the cross correlation vector,  $\mathbf{r}_{\mathbf{x}s}$ , is defined as

$$\mathbf{r}_{\mathbf{x}s_0} = [\mathbf{r}_{\mathbf{x}_1 s_0} \quad \mathbf{r}_{\mathbf{x}_2 s_0} \quad \cdots \quad \mathbf{r}_{\mathbf{x}_P s_0}] \quad (13.126)$$

with

$$\mathbf{r}_{\mathbf{x}_p s} = [r_{x_p s_0}(0) \quad r_{x_p s_0}(1) \quad \cdots \quad r_{x_p s_0}(L-1)], \quad p = 0, 1, \dots, P-1, \quad (13.127)$$

where each element is defined as

$$r_{x_p s_0}(l) = E[x_p(k-l)s_0(k)], \quad p = 0, 1, \dots, P-1, \quad l = 0, 1, \dots, L-1. \quad (13.128)$$

The matrix  $\mathbf{R}_{\mathbf{xx}}$  is defined as in (13.103). The weights  $\mathbf{w}$  are arranged in the same way as in (13.97). It is thus straightforward to formulate the problem but the difficulty comes down to ensuring that the matrix  $\mathbf{R}_{\mathbf{xx}}$  is invertible and that the reference signal  $s_0(k)$  is available. In communication systems, the reference can usually be made available by training. For speech applications, it is common to use either a Voice Activity Detector (VAD) to detect when the SOI is active. As an alternative, a pre-calibration can be used. In radar and sonar applications, the SOI is not available but sometimes partial information can be available. Note that the number of filter coefficients becomes large if the frequency bandwidth of the beamformer is wide.

### 3.13.5.7 Frequency domain Wiener filter

In the frequency domain, the objective can be formulated similarly as for the time domain. For each frequency, the objective will be

$$\mathbf{W}_{\text{opt}}(\omega) = \arg \left( \min_{\mathbf{W}(\omega)} \mathbf{S}_{\epsilon\epsilon}(\omega) \right), \quad (13.129)$$

where the error signal  $\mathbf{S}_{\epsilon\epsilon}(\omega)$  is the Fourier transform of  $r_{\epsilon\epsilon}(0) = E[(s_0(k) - y(k))^2]$ . The optimal weights, which minimize the expectational square difference in (13.129) between the output and the reference signal for each frequency, is found by

$$\mathbf{W}_{\text{opt}}(\omega) = [\mathbf{S}_{\mathbf{XX}}(\omega)]^{-1} \mathbf{S}_{\mathbf{X}S_0}(\omega), \quad (13.130)$$

where the spectral density matrix  $\mathbf{S}_{\mathbf{XX}}(\omega)$  is defined in (13.108). The cross correlation vector  $\mathbf{r}_{\mathbf{X}s}(\omega)$  is defined as

$$\mathbf{S}_{\mathbf{X}S_0}(\omega) = [S_{X_0S_0}(\omega) \quad S_{X_1S_0}(\omega) \quad \dots \quad S_{X_{P-1}S_0}(\omega)]^T \quad (13.131)$$

with each element defined as

$$S_{X_pS_0}(\omega) = \mathcal{F}(E[x_p(k-l)s_0(k)]), \quad (13.132)$$

where  $\mathcal{F}$  means Fourier transform

$$p = 0, 1, \dots, P-1.$$

The weights  $\mathbf{W}_{\text{opt}}(\omega)$  are defined as complex valued vectors of dimension  $P$ . This solution gives the non-causal Wiener filter which is usually approximated by using discrete FFT. This type of filter has the same limitations as the time domain solution. However, it is much more efficient to implement since the size of the problem is determined by  $P$  and not  $P \cdot L$ . Also in practical scenarios, various ways have been suggested to estimate the SOI. For speech processing, it usually involves pre-calibration or VAD, whereas in communication systems, training sequences are included.

### 3.13.5.8 Optimal near-field signal-to-noise plus interference beamformer (SNIB)

The desired criteria for many applications such as radar and sonar is to maximize the output signal-to-noise plus interference power ratio (SNIR). This criteria is defined as

$$Q = \frac{\text{average signal output power}}{\text{average noise-plus-interference output power}} \quad (13.133)$$

and the beamformer which maximizes the ratio  $Q$  is the optimal signal-to-noise plus interference beamformer (SNIB). We need to express the mean signal output power as a function of the filter weights in the beamformer, and find the optimal weights maximizing  $Q$ .

#### 3.13.5.8.1 Time domain formulation

The power of the beamformer output, when only the SOI  $s_0(n)$  is active, is given by the zero lag of the autocorrelation function,  $r_{y_s y_s}(0)$ , as follows:

$$r_{y_s y_s}(0) = \mathbf{w}^H \mathbf{R}_{ss} \mathbf{w}, \quad (13.134)$$

where  $\mathbf{R}_{ss}$  is defined as

$$\mathbf{R}_{ss} = \begin{pmatrix} \mathbf{R}_{s_0,0s_0,0} & \mathbf{R}_{s_0,0s_0,1} & \cdots & \mathbf{R}_{s_0,0s_0,P-1} \\ \mathbf{R}_{s_0,1s_0,0} & \mathbf{R}_{s_0,1s_0,1} & \cdots & \mathbf{R}_{s_0,1s_0,P-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{s_0,P-1s_0,0} & \mathbf{R}_{s_0,P-1s_0,1} & \cdots & \mathbf{R}_{s_0,P-1s_0,P-1} \end{pmatrix} \quad (13.135)$$

with submatrices

$$\mathbf{R}_{s_0,ms_0,n} = \begin{pmatrix} r_{s_0,m s_0,n}(0) & r_{s_0,m s_0,n}(1) & \cdots & r_{s_0,m s_0,n}(L-1) \\ r_{s_0,m s_0,n}(1) & r_{s_0,m s_0,n}(0) & \cdots & r_{s_0,m s_0,n}(L-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_{s_0,m s_0,n}(L-1) & r_{s_0,m s_0,n}^*(L-2) & \cdots & r_{s_0,m s_0,n}(0) \end{pmatrix} \quad (13.136)$$

and

$$r_{s_0,m s_0,n}(l) = E[s_0,m(k-l)s_0,n(k)], \quad l = 0, 1, \dots, L-1, \quad (13.137)$$

where  $s_0,m(k)$  is the received signal at sensor element  $m$  from the SOI  $s_0(k)$ . The filters  $\mathbf{w}$  are arranged according to

$$\mathbf{w} = [\mathbf{w}_1^T \ \mathbf{w}_2^T \ \cdots \ \mathbf{w}_I^T]^T, \quad (13.138)$$

where

$$\mathbf{w}_p = [w_p(0) \ w_p(1) \ \cdots \ w_p(L-1)]^T, \quad p = 0, 1, \dots, P-1. \quad (13.139)$$

In the same way, one may write an expression for the noise-plus-interference power,  $r_{y_n y_n}(0)$ , when all the surrounding noise sources are active but the source signal of interest is inactive, as

$$r_{y_n y_n}(0) = \mathbf{w}^T \mathbf{R}_{nn} \mathbf{w}. \quad (13.140)$$

Now, the optimal weights are found by maximizing a ratio of two quadratic forms according to

$$\mathbf{w}_{opt} = \arg \left( \max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{R}_{ss} \mathbf{w}}{\mathbf{w}^T \mathbf{R}_{nn} \mathbf{w}} \right). \quad (13.141)$$

The solution to this problem is given by the eigenvector corresponding to the maximum eigenvalue of the generalized eigenvalue problem of  $\mathbf{R}_{ss}$  and  $\mathbf{R}_{nn}$ .

It is important that one can estimate the signal correlation matrix separately from the noise correlation matrix. This is not possible, so in practice only  $\mathbf{R}_{xx} = \mathbf{R}_{nn} + \mathbf{R}_{ss}$  is available. In many cases (for radar and sonar), the noise correlation matrix can be estimated. Then an estimate of  $\mathbf{R}_{ss}$  can be obtained as  $\hat{\mathbf{R}}_{ss} = \mathbf{R}_{xx} - \mathbf{R}_{nn}$ . This will work as long as the SNR is reasonably good and the noise can be estimated when there is no signal. This will require a VAD for speech applications or a communication protocol for communication applications. In active sonar and radar applications, it is possible to do this estimation in periods when the receiver acts in a passive mode. This time domain formulation is usually not well suited for broadband problems since optimization only consider one eigenvalue the corresponding eigenvector does not represent the signal space of the SOI as a result the signal output will be very distorted. The frequency domain formulation is more useful since  $\mathbf{R}_{ss}$  represented in frequency domain has low rank.

### 3.13.5.8.2 Frequency domain formulation

The time domain problem has high complexity for broadband scenarios when the number of sensors is high and the bandwidth covers a wide range, such that the length of the FIR filters is large. To lower the overall computational complexity, the optimal signal-to-noise plus interference beamformer can be formulated for each frequency individually. The two problems will not be equivalent in general but that fact is usually ignored to be able to provide an efficient solution. The weights that maximize the quadratic ratios for all frequencies represent the optimal beamformer that maximizes the total output power ratio. This is true provided that the different frequency bands are independent and the full-band signal can be created perfectly. The time domain and frequency domain criteria will not be equivalent since there is no direct relationship between the two solutions. The SNIB optimization is an eigenvalue problem; in one case it is solved frequency by frequency, and in the other case it is solved for the overall time domain problem.

Now consider the SNIB problem for frequency  $\omega$ . The quadratic ratio between the output SOI power spectral density and the output noise-plus-interference power density is given by

$$\mathbf{W}_{\text{opt}}(\omega) = \arg \left( \max_{\mathbf{W}(\omega)} \frac{\mathbf{W}(\omega)^H \mathbf{S}_{\mathbf{s}_0 \mathbf{s}_0}(\omega) \mathbf{W}(\omega)}{\mathbf{W}(\omega)^H \mathbf{S}_{\mathbf{n} \mathbf{n}}(\omega) \mathbf{W}(\omega)} \right) \quad (13.142)$$

and the spectral density matrix for the SOI is defined as

$$\mathbf{S}_{\mathbf{s}_0 \mathbf{s}_0}(\omega) = \begin{bmatrix} r_{s_0,0s_0,0}(\omega) & S_{s_0,0s_0,1}(\omega) & \cdots & S_{s_0,0s_0,P-2}(\omega) & S_{s_0,0s_0,P-1}(\omega) \\ S_{s_0,1s_0,0}(\omega) & S_{s_0,1s_0,1}(\omega) & \ddots & \cdots & S_{s_0,1s_0,P-1}(\omega) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ S_{s_0,P-2s_0,0}(\omega) & S_{s_0,P-2s_0,1}(\omega) & \ddots & S_{s_0,P-2s_0,P-2}(\omega) & S_{s_0,P-2s_0,P-1}(\omega) \\ S_{s_0,P-1s_0,0}(\omega) & S_{s_0,P-1s_0,1}(\omega) & \cdots & S_{s_0,P-1s_0,P-1}(\omega) & S_{s_0,P-1s_0,P-1}(\omega) \end{bmatrix}, \quad (13.143)$$

where

$$S_{s_0,i s_0,j}(\omega) = \mathcal{F}(E[s_{0,i}(k-l)s_{0,j}(k)]). \quad (13.144)$$

In a similar way, one may define the noise-plus-interference correlation spectral density matrix,  $\mathbf{S}_{\mathbf{n} \mathbf{n}}(\omega)$ , when only the disturbing sources are active.

The frequency domain weights  $\mathbf{W}(\omega)$  for frequency  $\omega$  are defined as complex valued vectors of dimension  $P$ . The problem is solved as a generalized eigenvalue problem for each frequency. However, the weights have been calculated for each frequency and thus an independent scaling may occur for each frequency. Therefore, when the beamformer weights are used to process data using STFT for instance, large SOI distortion may occur. To resolve this, the weights are converted to time domain and a continuity constraint is applied such that the equivalent time domain weights are smooth. But note that the frequency domain and time domain problem are not equivalent. In the time domain formulation the SOI is severely distorted even if one achieve a higher SNIR improvement.

### 3.13.5.9 Examples for optimal beamformers

As a comparison between the various methods and provide an idea on how the performance varies a few design examples are presented. The design examples are for the GSC, Wiener and SNR optimization

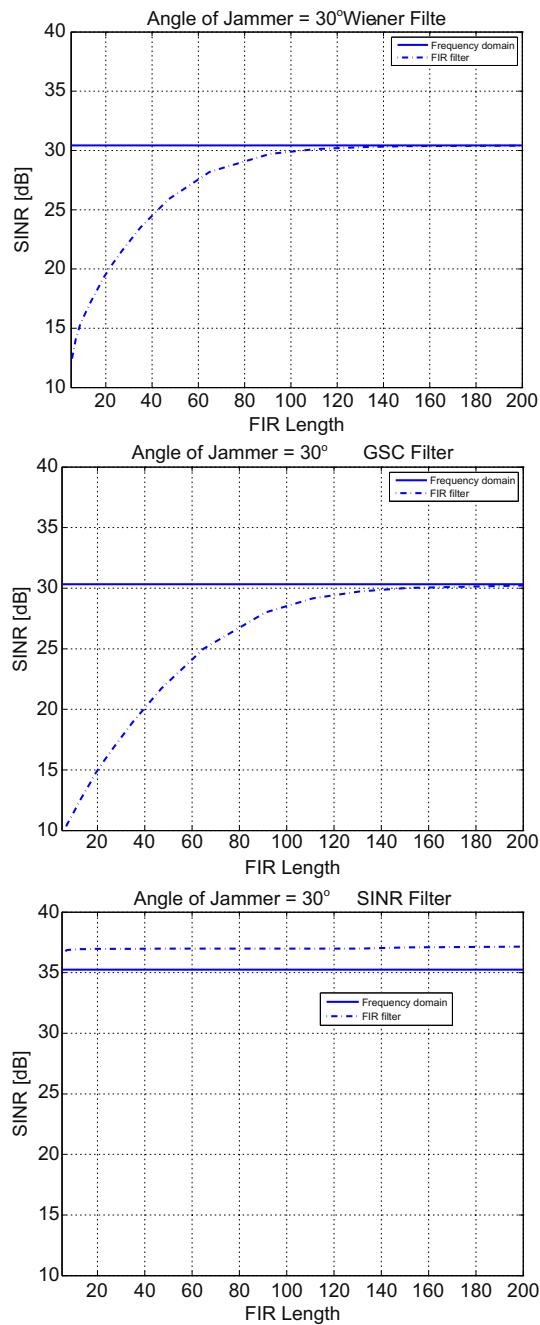
**Table 13.5** Common Design Parameters

Parameters	Values
Number of microphones, $K$	5
Spectral passband, $\Omega_p$	[300, 7000] Hz
Sampling frequency, $f_s$	16,000 Hz
Speed of sound, $c$	343 m s <sup>-1</sup>
Interfering jammer signal	$\theta = 30^\circ$

with the formulation in time and frequency domain. A broadside linear array is used with inter-element spacing of 5 cm. The design parameters are as specified in Table 13.5. The target signal and the jammer signal has the same spectral shape and the same power. The target is impinging from  $0^\circ$  and the jammer from  $30^\circ$ . The noise is additive diffuse noise independent between the elements with an SNR of 30 dB. As can be seen from the Figure 13.15 for this rather simple anechoic case the Wiener solution and GSC provide more or less the same Signal to Interference Noise Ratio, SINR. For this very broadband case a high number of FIR taps are needed to reach the frequency domain bound. SINR optimization gives better results in the SINR measure, in frequency domain the results does not give as high values. But this does not provide a full picture of the problem since the SINR optimization does not take into account the distortion of the source signal. Since, in the time domain design only the generalized eigenvector corresponding to largest generalized eigenvalue is considered means that much of the original source might have been suppressed as well. In the example the rank of the matrix  $\mathbf{R}_{ss}$  is much larger than one in the time domain problem but the spectral density matrix  $\mathbf{S}_{s_0s_0}(\omega)$  has rank one in frequency domain thus in the frequency domain more of the original source is maintained and also more of the diffuse noise giving less improvement in the overall SNIR. In time domain optimization less of the source is maintained and more of the noise is suppressed.

### 3.13.6 Conclusion

This study has provided insights into the design and processing of broadband beamformers, with the main applications being microphone arrays, sonar, radar and broadband communications. Currently there has been significant interest for broadband beamformer design from the radio physics community, particularly within the Square Kilometer Array (SKA) project. The two mainstream approaches for broadband beamforming are optimal beamforming and optimum beamforming. In the former, optimization tools are used to design weights which are data independent, and in the latter, the statistics of the input data are included in the optimization design. A natural extension of the optimum beamforming techniques is to apply adaptive algorithms to find the weights. In the optimization, it is important to include robustness in the design, and this is quintessential in broadband beamforming since the element spacing is chosen to avoid spatial aliasing. Spatial aliasing is connected to the upper frequency considered in the design. Since broadband beamforming covers one or several octaves, the resolution of the beam will be limited in the lowest frequency band. Thus for a design with constant beamwidth, the low frequency design

**FIGURE 13.15**

Signal to Interference and Noise Ratio for varying length and compared to frequency domain solution for (a) Wiener filter, (b) GSC, and (c) SINR optimization.

is highly demanding and the design without robustness constraints becomes very sensitive. Also for optimum designs to constrain the solution over a large range of frequencies is almost futile.

*Relevant Theory:* Signal Processing Theory and Array Signal Processing

See Vol. 1, Chapter 4 Random Signals and Stochastic Processes

See Vol. 1, Chapter 8 Modern Transform Design for Practical Audio/Image/Video Coding Applications

See Vol. 1, Chapter 12 Adaptive Filters

See this volume, Chapter 18 Source Localization and Tracking

### 3.13.6.1 Further reading

For further reading in the area of modeling and wave propagation we suggest the following books [4, 5, 9]. Those books provide a detailed theory and understanding of propagation models and acoustics. A very detailed modeling of underwater acoustics using so called matched field processing has also been suggested in [12, 13].

For broadband beamformer design, the literature is extensive; some suggestions are [1, 14–19]. The broadband beamformer problem has been studied in nearfield and farfield and using various optimization methods [7, 18]; in this chapter, we have discussed a general formulation.

Design methods that combine steering and beamformer design have been suggested for element space processing [20, 21] and of course for wave domain processing, see [8, 10, 11, 19] and many other references.

Optimum beamforming methods are discussed in some books [2] and overview papers [3, 22]. For some discussion on practical aspects of the optimum beamforming problem, see [23–26].

The important topic of robust and adaptive beamforming has been studied in many research works; some relevant ones are [6, 22, 27]. For a current reference we would like to refer to [6].

The pre-steering and beam steering is a vital part of the beamformer system, even if we have left out details in this chapter. That topic needs methods for tracking and localization algorithms. Here are some suggested adequate reading for these areas: Ref. [28] is the classic paper on localization using generalized coherence function; this work has been extended to the so-called phase transform (PHAT), which is more or less a *de facto* standard for microphone arrays [29]. For tracking, particle filter based methods are usually used [30–32].

For broadband beamformer applications with microphone arrays, we would suggest a number of books [29, 33] and some well known articles [34, 35].

Radar beamforming applications can be found in these articles [36–38] and in these books [39, 40]. Broadband beamforming for sonar application can be found in the following articles [41–43], and they are also discussed in these books [1, 4, 44, 45].

---

## References

- [1] D.H. Johnson, D.E. Dudgeon, *Array Signal Processing*, Prentice Hall, 1993, ISBN: 0130485136.
- [2] H.L. Van Trees, Knovel (firme), *Optimum Array Processing*, Wiley Online Library, 2002, ISBN: 0471463833.
- [3] B.D. Van Veen, K.M. Buckley, Beamforming: a versatile approach to spatial filtering, *IEEE Acoust. Speech Signal Process. Mag.* 5 (1988) 4–24.

- [4] J. Lawrence, Ziomek, *Acoustic Field Theory and Space-Time Signal Processing*, CRC Press, Boca Raton, Florida, 1995.
- [5] E.G. Williams, *Fourier Acoustics: Sound Radiation and Nearfield Acoustical Holography*, Academic Press, 1999.
- [6] S.A. Vorobyov, *Adaptive and Robust Beamforming*, Academic Press Library in Statistical and Array Signal Processing, Elsevier, in press.
- [7] Z. Feng, C. Yiu, S. Nordholm, A two-stages method for the design of near field broadband beamformer, *IEEE Trans. Signal Process.* 1 (2011) 99, doi:10.1109/TSP.2011.2133490, ISSN: 1053-587X.
- [8] S. Yan, H. Sun, U.P. Svensson, X. Ma, J.M. Hovem, Optimal modal beamforming for spherical microphone arrays, *IEEE Trans. Audio Speech Lang. Process.* 19 (2) (2011) 361–371.
- [9] P.M.C. Morse, K.U. Ingard, *Theoretical Acoustics*, Princeton University Press, 1968.
- [10] J. Meyer, G. Elko, A highly scalable spherical microphone array based on an orthonormal decomposition of the soundfield, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* 2002, vol. 2, IEEE, 2002, pp. 1781–1784.
- [11] B. Rafaely, Analysis and design of spherical microphone arrays, *IEEE Trans. Speech Audio Process.* 13 (1) (2005) 135–143.
- [12] A.B. Baggeroer, W.A. Kuperman, H. Schmidt, Matched field processing: source localization in correlated noise as an optimum parameter estimation problem, *J. Acoust. Soc. Am.* 83 (1988) 571.
- [13] A. Tolstoy, *Matched Field Processing for Underwater Acoustics*, vol. 52, World Scientific, Singapore, 1993.
- [14] Huawei Chen, Wee Ser, Design of robust broadband beamformers with passband shaping characteristics using tikhonov regularization, *IEEE Trans. Audio Speech Lang. Process.* 17 (4) (2009) 665–681, doi:10.1109/TASL.2008.2012318, ISSN: 1558-7916.
- [15] S. Doclo, M. Moonen, Design of broadband beamformers robust against gain and phase errors in the microphone array characteristics, *IEEE Trans. Signal Process.* 51 (10) (2003) 2511–2526, ISSN: 1053-587X.
- [16] R.A. Kennedy, T.D. Abhayapala, D.B. Ward, Broadband nearfield beamforming using a radial beampattern transformation, *IEEE Trans. Signal Process.* 46 (8) (1998) 2147–2156, ISSN: 1053-587X.
- [17] S. Nordebo, I. Claesson, S. Nordholm, Weighted Chebyshev approximation for the design of broadband beamformers using quadratic programming, *IEEE Signal Process. Lett.* 1 (7) (1994) 103–105.
- [18] SE Nordholm, V. Rehbock, KL Tee, S. Nordebo, Chebyshev optimization for the design of broadband beamformers in the near field, *IEEE Trans. Circ. Syst. II: Analog Digital Signal Process.* 45(1) (1998) 141–143, ISSN: 1057-7130.
- [19] D.B. Ward, R.A. Kennedy, RC Williamson, Theory and design of broadband sensor arrays with frequency invariant far-field beam patterns, *J. Acoust. Soc. Am.* 97 (2) (1995) 1023–1034.
- [20] C.C. Lai, S. Nordholm, Y.H. Leung, Design of robust steerable broadband beamformers with spiral arrays and the farrow filter structure, in: Proceeding of the International Workshop Acoustic Echo Noise, Control, August 2010.
- [21] L.C. Parra, Steerable frequency-invariant beamforming for arbitrary arrays, *J. Acoust. Soc. Am.* 119 (2006) 3839.
- [22] H. Krim, M. Viberg, Two decades of array signal processing research: the parametric approach, *IEEE Signal Process. Mag.* 13 (4) (1996) 67–94.
- [23] I. Claesson, S. Nordholm, A spatial filtering approach to robust adaptive beamforming, *IEEE Trans. Antennas Propag.* 40 (9) (1992) 1093–1096.
- [24] S. Doclo, M. Moonen, GSVD-based optimal filtering for single and multimicrophone speech enhancement, *IEEE Trans. Acoust. Speech Signal Process.* 50 (9) (2002) 2230–2244.
- [25] N. Grbić, S. Nordholm, Soft constrained subband beamforming for handsfree speech enhancement, *IEEE Int. Conf. Acoust. Speech Signal Process.* 1 (2002) 885–888.

- [26] S.A. Vorobyov, A.B. Gershman, Z.Q. Luo, Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem, *IEEE Trans. Signal Process.* 51 (2) (2003) 313–324, ISSN: 1053-587X.
- [27] J.E. Hudson, *Adaptive Array Principles*, vol. 11, Institute of Engineering and Technology, 1981.
- [28] G.C. Carter, Time delay estimation for passive sonar signal processing, *IEEE Trans. Acoust. Speech Signal Process.* 29 (3) (1981) 463–470.
- [29] M. Brandstein, D. Ward, *Microphone Arrays: Signal Processing Techniques and Applications*, Springer Verlag, 2001.
- [30] Z. Khan, T. Balch, F. Dellaert, Mcmc-based particle filtering for tracking a variable number of interacting targets, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (11) (2005) 1805–1819.
- [31] E. Lehmann, A. Johansson, Particle filter with integrated voice activity detection for acoustic source tracking, *EURASIP J. Adv. Signal Process.* (2007) Article ID 50870, 11 pp.
- [32] D.B. Ward, E.A. Lehmann, R.C. Williamson, Particle filtering algorithms for tracking an acoustic source in a reverberant environment, *IEEE Trans. Speech Audio Process.* 11 (6) (2003) 826–836.
- [33] J. Benesty, J. Chen, Y. Huang, *Microphone Array Signal Processing*, vol. 1, Springer Verlag, 2008.
- [34] J.L. Flanagan, J.D. Johnston, R. Zahn, G.W. Elko, Computer-steered microphone arrays for sound transduction in large rooms, *J. Acoust. Soc. Am.* 78 (1985) 1508.
- [35] Y. Kaneda, J. Ohga, Adaptive microphone-array system for noise reduction, *IEEE Trans. Acoust. Speech Signal Process.* 34 (6) (1986) 1391–1400.
- [36] J.R. Guerci, E.J. Baranowski, Knowledge-aided adaptive radar at DARPA: an overview, *IEEE Signal Process. Mag.* 23 (1) (2006) 41–50, ISSN: 1053-5888.
- [37] A. Haimovich, The eigencanceler: adaptive radar by eigenanalysis methods, *IEEE Trans. Aerosp. Electron. Syst.* 32 (2) (1996) 532–542.
- [38] J. Ward, Space-time adaptive processing for airborne radar, in: IEE Colloquium on Space-Time Adaptive Processing (Ref. No. 1998/241), IET, 1998, pp. 2/1–2/6.
- [39] J.C. Curlander, R.N. McDonough, *Synthetic Aperture Radar: Systems and Signal Processing*, vol. 199, Wiley, New York, 1991.
- [40] M.I. Skolnik et al., *Radar Handbook*, vol. 2, McGraw-Hill, New York, 1990.
- [41] H. Cox, R. Zeskind, M. Owen, Robust adaptive beamforming, *IEEE Trans. Acoust. Speech Signal Process.* 35 (10) (1987) 1365–1376.
- [42] A. Elfes, Sonar-based real-world mapping and navigation, *IEEE J. Robot. Autom.* 3 (3) (1987) 249–265, ISSN: 0882-4967.
- [43] W.C. Knight, R.G. Pridham, S.M. Kay, Digital signal processing for sonar, *Proc. IEEE* 69 (11) (1981) 1451–1506.
- [44] R.O. Nielsen, *Sonar Signal Processing*, Artech House, Inc., 1991.
- [45] N.L. Owsley, *Sonar Array Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.

# DOA Estimation Methods and Algorithms

# 14

Pei-Jung Chung<sup>\*</sup>, Mats Viberg<sup>†</sup>, and Jia Yu<sup>\*</sup>

<sup>\*</sup>The University of Edinburgh, UK

<sup>†</sup>Chalmers University of Technology, Sweden

## 3.14.1 Background

The problem of retrieving information conveyed in propagating waves occurs in a wide range of applications including radar, sonar, wireless communications, geophysics and biomedical engineering. Methods for processing data measured by sensor arrays have attracted lots of attention of many researchers over last three decades. Recent advances in computational technology have enabled implementation of sophisticated algorithms in practical systems.

Early space-time processing techniques view direction of arrival (DOA) as a spatial spectrum. The Fourier transform based conventional beamformer is subject to resolution limitation due to finite array aperture. Similar to its temporal counterpart, the spatial periodogram can not benefit from increasing signal to noise ratio (SNR) or number of samples. Better estimates can be achieved by applying windowing function to reduce spectral leakage effects. The minimum variance distortionless (MVDR) beamformer [1] overcomes the resolution limitation of Fourier based techniques by formulating the spectrum estimation as a constrained optimization problem. Also, its performance can be enhanced by high SNR. The multiple signal classification (MUSIC) algorithm [2] is representative of subspace methods based on eigenstructure of the spatial correlation matrix. In addition to high resolution, MUSIC takes advantage of SNR, number of sensors and number of samples. It improves estimation accuracy with respect to all dimensions and is statistically efficient. However, in the presence of correlated source signals, subspace methods degrade dramatically as the signal subspace suffers from rank deficiency. On the other hand, parametric methods such as the maximum likelihood (ML) approach exploit the data model fully, leading to statistically efficient estimators. More importantly, they remain robust in critical scenarios involving signal coherence, closely located signals and low SNRs. The optimal properties come along with increased computational complexity. Hence, efficient implementation is crucial for parametric methods.

The importance of array processing methods has been reflected by the huge amount of publications. Among these contributions, several review articles [3–5] have proven to be an excellent guide for first exposure to this research field, while the textbooks [6,7] have been valuable references for in-depth learning. Thanks to the advances in both theoretical methods and computational powers over the last decade, array processing methods have been re-examined from various aspects to address new challenges arising in these new application areas such as wireless communications. The purpose of this article is to

provide interested readers an overview of traditional array processing methods and recent development in the field.

The organization of this article is as follows. The data model based on plane wave propagation is introduced in Section 3.14.2. Important quantities including array response vector and second order statistics are derived therein. Standard direction finding algorithms are covered in Sections 3.14.3–3.14.5. Section 3.14.3 is devoted to spectral analysis based methods: conventional beamforming, MVDR beamforming techniques. The sparse data representation based approach is presented in the same section. The subspace methods and related issues are treated in Section 3.14.4. In the subsequent Section 3.14.5, the parametric approach is illustrated. Important algorithms based on the maximum likelihood principle and subspace fitting are illustrated. The implementation of the aforementioned estimators are also discussed. While the algorithms discussed in Sections 3.14.3–3.14.5 deal with narrow band data, we treat the broadband case separately in Section 3.14.6. The number of signals is a fundamental assumption in most array processing methods. The problem of signal detection is addressed in Section 3.14.7. In Section 3.14.8, we treat scenarios with non-standard assumptions by highlighting relevant techniques and references. A brief discussion is given in Section 3.14.9.

## 3.14.2 Data model

Propagating waves carry energy radiated by sources to sensors. To extract information conveyed in the propagating waves, such as source location or propagation direction, one needs a proper description of wavefields. In Section 3.14.2.1, we start with a brief discussion on the physics of wave propagation and formulate the transmission between signal sources and receiving sensors as a linear time invariant system. Then a frequency domain data model for far-field sources is developed in Section 3.14.2.2. The fundamental issue on identifiability is discussed in Section 3.14.2.3.

### 3.14.2.1 Wave propagation

The physics of propagating waves is governed by the wave equation for medium of interest. The homogeneous wave equation for a general scalar field  $E(t, \mathbf{r})$  at time instant  $t$  and location  $\mathbf{r} = [x \ y \ z]^T$  is given by

$$\frac{\partial^2 E(t, \mathbf{r})}{\partial x^2} + \frac{\partial^2 E(t, \mathbf{r})}{\partial y^2} + \frac{\partial^2 E(t, \mathbf{r})}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 E(t, \mathbf{r})}{\partial t^2}, \quad (14.1)$$

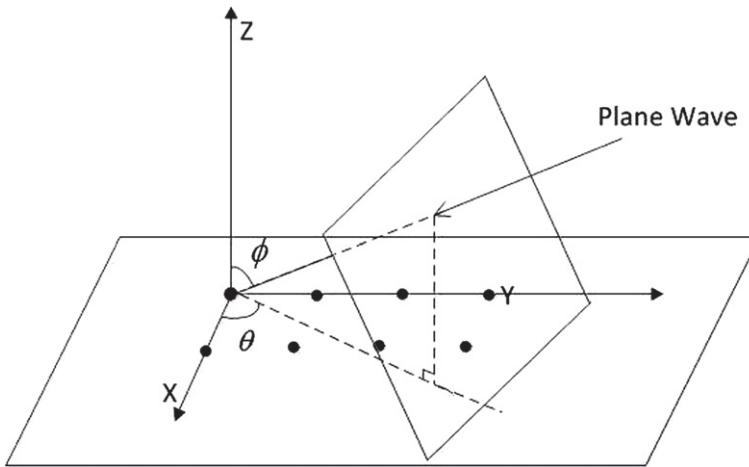
where the parameter  $c$  represents the propagation velocity.  $E(t, \mathbf{r})$  can be an electric density field in electromagnetics or acoustic pressure in acoustic waves.

A solution of special interest to (14.1) takes a complex exponential form:

$$E(t, \mathbf{r}) = A \exp[j(\omega t - \mathbf{k}^T \mathbf{r})], \quad (14.2)$$

where  $\omega = 2\pi f$  is the temporal frequency and  $\mathbf{k} = [k_x \ k_y \ k_z]^T$  is the wave number vector. Here  $\mathbf{k}$  is considered as the spatial frequency of this mono-chromatic wave. Inserting (14.2) into (14.1), one can readily see how temporal frequency  $\omega$  and the spatial frequency  $k$  are related:

$$\|\mathbf{k}\|^2 = k_x^2 + k_y^2 + k_z^2 = \frac{\omega^2}{c^2}. \quad (14.3)$$

**FIGURE 14.1**

Plane wave propagation and coordinate system.

Replacing the propagation velocity with  $c = f\lambda$  in (14.3) where  $\lambda$  is the wavelength, the magnitude of the wave number vector is given by  $|\mathbf{k}| = 2\pi/\lambda$ .

The elementary wave (14.2) also represents a propagating wave

$$E(t - \xi^T \mathbf{r}) = A \exp[j\omega(t - \xi^T \mathbf{r})], \quad (14.4)$$

where  $\xi = \mathbf{k}/\omega$  is termed *slowness vector*. It points in the same direction as  $\mathbf{k}$  and has the magnitude  $|\xi| = 1/c$ , which is the inverse of the propagation velocity. From (14.4), it can be readily seen that the direction of propagation is given by  $\mathbf{u}_r = \xi/|\xi|$ . Let the origin of the coordinate system be close to the sensor array. The slowness vector  $\xi$  can then be expressed as

$$\xi = -\frac{1}{c} \begin{bmatrix} \sin \phi \cos \theta \\ \sin \phi \sin \theta \\ \cos \phi \end{bmatrix}, \quad (14.5)$$

where  $\phi$  and  $\theta$  denote the elevation and azimuth angles, respectively. Both parameters characterize the direction of propagation and are referred to as *direction of arrival* (DOA) (see Figure 14.1).

*Far-field assumption:* For far-field sources, the propagation distance to a sensor array is much larger than the aperture of the array, the DOA parameter is approximately the same to all sensors. According to (14.2), the wave front of constant phase at time instant  $t$  is a plane perpendicular to the propagating direction given by  $\mathbf{k} \cdot \mathbf{r} = \text{constant}$ . The term, *plane wave assumption* is alternatively used in the literature.

### 3.14.2.2 Frequency domain description

In an ideal medium, the propagation between signal sources and a sensor array is considered as a linear time-invariant system. Due to the applicability of the superposition principle and the Fourier transform,

the analysis of wave propagation is greatly simplified. In the following, we will develop a general frequency domain model and discuss the narrow band case. The general model is useful for broadband data that may be measured in underwater acoustical or geophysical experiments. The narrow band model is preferred in applications where the signal bandwidth is much smaller than the carrier frequency, for example, wireless communications and radar.

### 3.14.2.2.1 General model

Consider an array of sensors located at  $\mathbf{r}_m$ ,  $m = 1, \dots, M$  receiving signals generated by  $P$  sources. Without loss of generality, the first sensor coincides with the origin of the coordinate system. Let  $s_p(t)$ ,  $p = 1, \dots, P$  denote the signals received by the first sensor. The signal observed by the  $m$ th sensor is the sum of a time-delayed version of original signals corrupted by noise  $n_m(t)$ :

$$x_m(t) = \sum_{p=1}^P s_p(t - \tau_{mp}) + n_m(t), \quad (14.6)$$

where  $\tau_{mp}$  denotes the propagation time difference between the origin and the  $m$ th sensors. The time difference is given by  $\tau_{mp} = \boldsymbol{\xi}_p \cdot \mathbf{r}_m$  where  $\boldsymbol{\xi}_p$  is associated with the  $p$ th wave. It is related to the DOA parameter through  $\mathbf{k} = \omega \boldsymbol{\xi}$  and (14.5).

Applying the Fourier transform to the array output  $\mathbf{x}(t) = [x_1(t), \dots, x_M(t)]^T$ , the time delay  $\tau_{mp}$  translates to a phase shift  $e^{-j\omega\tau_{mp}}$ . In the presence of noise, the array output vector  $\mathbf{x}(\omega)$  can be written as

$$\mathbf{x}(\omega) = \begin{bmatrix} x_1(\omega) \\ \vdots \\ x_M(\omega) \end{bmatrix} = \sum_{p=1}^P \mathbf{a}(\phi_p, \theta_p) s_p(\omega) + \mathbf{n}(\omega), \quad (14.7)$$

where the steering vector  $\mathbf{a}(\phi_p, \theta_p)$  is the spatial signature for the  $p$ th incoming wave:

$$\mathbf{a}(\phi_p, \theta_p) = \begin{bmatrix} e^{-j\mathbf{k}_p \cdot \mathbf{r}_1} \\ \vdots \\ e^{-j\mathbf{k}_p \cdot \mathbf{r}_M} \end{bmatrix}. \quad (14.8)$$

Define the steering matrix as

$$\mathbf{A}(\omega, \boldsymbol{\phi}, \boldsymbol{\theta}) = \begin{bmatrix} & \vdots & & \vdots \\ \mathbf{a}(\omega, \phi_1, \theta_1) & \cdots & \mathbf{a}(\omega, \phi_p, \theta_p) & \\ & \vdots & & \vdots \end{bmatrix}. \quad (14.9)$$

A compact expression for (14.7) is as follows:

$$\mathbf{x}(\omega) = \mathbf{A}(\omega, \boldsymbol{\phi}, \boldsymbol{\theta}) \mathbf{s}(\omega) + \mathbf{n}(\omega), \quad (14.10)$$

where the signal vector  $\mathbf{s}(\omega) = [s_1(\omega), \dots, s_P(\omega)]^T$ . According to the asymptotic theory of Fourier transform, the frequency domain data is complex normally distributed regardless of the distribution in the time domain. In addition, frequency bins are mutually independent. These properties form the basis for broadband DOA estimation.

### 3.14.2.2.2 Narrow band data

In many applications, the signal of interest is modulated by a carrier frequency  $\omega_c$  for transmission. At the receive side, the radio frequency signals are demodulated to baseband for further processing. Suppose the signal is band limited with the bandwidth  $B_s$ , and the maximal travel time between two sensors of the array for a plane wave is  $\Delta T$ . The *narrow band assumption* is valid if  $B_s \Delta T \ll 1$ . Then the complex baseband signal waveforms are approximately equal for all sensors. The general expression (14.10) can be simplified to the narrow band model

$$\mathbf{x}(t) = \mathbf{A}(\phi, \theta)s(t) + \mathbf{n}(t), \quad (14.11)$$

where the steering matrix  $\mathbf{A}(\phi, \theta)$  is computed at the carrier frequency  $\omega_c$  and  $s(t)$  represents the baseband signal waveform. The frequency dependence in (14.11) is omitted as relevant information is centered at  $\omega_c$ .

As DOA estimation is typically based on sampled values of  $\mathbf{x}(t)$  at time instants  $t_n, n = 1, \dots, N$ , we consider temporally discrete samples of (14.11) and replace  $t_n$  with the index  $n$ . Then the snapshot model is given by

$$\mathbf{x}(n) = \mathbf{A}(\phi, \theta)s(n) + \mathbf{n}(n), \quad n = 1, \dots, N. \quad (14.12)$$

Many array processing methods exploit second order statistics of the data. Assume the signal and noise are independent, stationary random processes with zero mean, correlation matrices  $\mathbf{R}_s = E[\mathbf{s}(n)\mathbf{s}(n)^H]$  and  $\mathbf{R}_n = E[\mathbf{n}(n)\mathbf{n}(n)^H]$ , respectively where  $(\cdot)^H$  denotes Hermitian transposition. Then the array correlation matrix can be expressed as

$$\mathbf{R}_x = E[\mathbf{x}(n)\mathbf{x}(n)^H] = \mathbf{A}(\phi, \theta)\mathbf{R}_s\mathbf{A}(\phi, \theta)^H + \mathbf{R}_n. \quad (14.13)$$

The noise process is often considered as spatially white, i.e.,  $\mathbf{R}_n = \sigma^2 \mathbf{I}$ , where  $\sigma^2$  denotes the noise level and  $\mathbf{I}$  is an  $M \times M$  matrix. This assumption is valid for sensors with sufficient spacing. In the presence of colored noise, the noise correlation matrix is no longer diagonal. In such case, the data can be *pre-whitened* by multiplying (14.12) with the inverse square root of the noise correlation matrix,  $\mathbf{R}_n^{-1/2}$ .

The array correlation matrix is estimated from array observations  $\mathbf{x}(n), n = 1, \dots, N$  by the sample correlation matrix

$$\widehat{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}(n)^H. \quad (14.14)$$

Under weak assumptions, the sample correlation matrix is a consistent estimator for the true correlation matrix, i.e.,  $\widehat{\mathbf{R}}_x$  converges with probability one to  $\mathbf{R}_x$  as the sample size increases. More details can be found in the book [8].

In many applications, sensors are located on a plane, for example, the  $yz$ -plane. Setting the elevation to be  $\phi = 90^\circ$  in (14.5), one can easily see that the slowness vector is characterized by only the azimuthal parameter,  $\theta$ . In the two dimensional scenario, the uniform linear array (ULA) is one of the most popular array configurations. Let the sensors of a ULA be placed along the  $y$ -axis,  $\mathbf{r}_m = [0, (m-1)\Delta, 0]^T$  where  $\Delta$  denotes or the inter-sensor distance. Then the phase shift term evaluated at  $\phi = \pi/2$  and  $\theta$  becomes  $\mathbf{k} \cdot \mathbf{r}_m = k(m-1)\Delta \sin \theta$ , leading to the steering vector

$$\mathbf{a}_{\text{ULA}}(\theta) = [1, e^{jk\Delta \sin \theta}, \dots, e^{jk(M-1)\Delta \sin \theta}]^T. \quad (14.15)$$

If  $\Delta$  is measured in wavelength,  $k\Delta = \frac{2\pi}{\lambda}\Delta = 2\pi d$  where  $d = \Delta/\lambda$ ,  $\mathbf{a}_{\text{ULA}}(\theta)$  can be expressed as

$$\mathbf{a}_{\text{ULA}}(\theta) = [1, e^{j2\pi d \sin \theta}, \dots, e^{j(M-1)2\pi d \sin \theta}]^T. \quad (14.16)$$

For a *standard ULA*, the inter-element spacing is half a wavelength,  $d = 1/2$ , the  $m$ th element in (14.16) becomes  $e^{j(m-1)\pi \sin \theta}$ .

For the 2D case, we shall use a shorter notation for the steering matrix,  $\mathbf{A}(\theta)$ , in (14.11)–(14.13). More specifically, the sampled array outputs and correlation matrix are expressed as

$$\mathbf{x}(n) = \mathbf{A}(\theta)\mathbf{s}(n) + \mathbf{n}(n) \quad (14.17)$$

and

$$\mathbf{R}_x = E[\mathbf{x}(n)\mathbf{x}(n)^H] = \mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}(\theta)^H + \mathbf{R}_n. \quad (14.18)$$

Given the observations  $\{\mathbf{x}(n), n = 1, \dots, N\}$ , the primary interest is to estimate the DOA parameter. In the following, we will present DOA estimation methods based on various ideas ranging from nonparametric spectral analysis, high resolution methods to the parametric approach. Our discussion will focus on the mostly investigated 2D case. The majority of the methods can be extended to multiple parameters per source in a straightforward manner. The broadband case will be discussed in detail in a separate chapter.

### 3.14.2.3 Uniqueness

A fundamental issue in the direction finding problem is whether the DOA parameters can be identified unambiguously. From the ideal data model (14.12) or (14.17), we know that the array output lies in a subspace spanned by the columns of the  $M \times P$  array steering matrix if the noise part  $\mathbf{n}(n)$  is ignored. For simplicity, we will present the results for the model (14.17). Assume any subset of vectors  $\mathbf{a}(\theta_p)$ ,  $p = 1, \dots, P$ , are linearly independent. The study in [9,10] specifies the conditions for the maximal number of sources that can be uniquely localized in terms of the number of sensors  $M$  and the rank of the signal correlation matrix  $R$ . The condition (1)  $P < (M+R)/2$  guarantees uniqueness for every batch of data, while the weaker condition (2)  $P < 2RM/(2R+1)$  guarantees uniqueness for almost every batch of data, with the exception of a set of batches of measure zero. When all sources are uncorrelated implying that  $R = P$ , condition (1) is always satisfied and uniqueness is always guaranteed. When the sources are fully correlated,  $R = 1$ , then uniqueness is ensured if  $P < (M+1)/2$  by condition (1). The weaker condition (2) leads to a less stringent condition  $P < 2M/3$ . Details on the proof are to be found in [10].

---

## 3.14.3 Beamforming methods

A homogenous wavefield  $E(t, \mathbf{r})$  has an interpretation as energy distribution in a frequency-wavenumber spectrum. The power spectrum of  $E(t, \mathbf{r})$  contains information about the source distribution over space [3]. From this perspective, estimation of DOA parameters is equivalent to finding the location in a spatial power spectrum where most power is concentrated. Beamforming techniques are spatial filters that combine weighted sensor outputs linearly

$$y(n) = \sum_{m=1}^M w_m^* x_m(n) = \mathbf{w}^H \mathbf{x}(n), \quad (14.19)$$

where  $\mathbf{w} = [w_1, w_2, \dots, w_M]^T$ . The weight vector is usually a function of the DOA parameter, i.e.,  $\mathbf{w} = \mathbf{w}(\theta)$ . The power of the beamformer output  $y(n)$  provides an estimate for the power spectrum at the direction  $\theta$ :

$$P(\theta) = \frac{1}{N} \sum_{n=1}^N |y(n)|^2 = \mathbf{w}^H \hat{\mathbf{R}}_x \mathbf{w}. \quad (14.20)$$

where  $\hat{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n) \mathbf{x}(n)^H$  is the sample covariance matrix. The maximum of  $P(\theta)$  is an indication for a signal source. Assuming  $P$  signals are present in the wavefield, we change the *look direction*  $\theta$  and evaluate  $P(\theta)$  over the range of interest. Then the  $P$  largest maxima are chosen as DOA estimates.

The expected beamformer output (14.20) is a smoothed version of the true power spectrum. Smoothing is carried out with the kernel  $|G(\theta - \theta_0)|^2$  centered at the look direction  $\theta = \theta_0$ , where  $|G(\theta)| = |\mathbf{w}^H \mathbf{a}(\theta)|$  is the array beam pattern [3]. An ideal beam pattern  $G(\theta) = \delta(\theta)$  would lead to an unbiased estimate for the power spectrum. However, in practice, due to finite array aperture, the beampattern consists of a main lobe and several sidelobes, leading to leakage from neighboring frequencies. Hence, the shape of the beampattern determines resolution capability and estimation accuracy of the spectral analysis based approach.

We will introduce the conventional beamformer in Section 3.14.3.1 and minimum variance distortionless (MVDR) beamformer in Section 3.14.3.2, respectively. A recently proposed sparse data approach that matches array measurements to a set of candidate directions will be discussed in Section 3.14.3.3.

### 3.14.3.1 Conventional beamformer

The conventional beamformer, also termed as delay-and-sum beamformer, combines the output of each sensor (14.6) coherently to obtain an enhanced signal from the noisy observation. For simplicity, assume  $P = 1$  for now. In its most general form, the conventional beamformer compensates the time delayed observation at the  $m$ th sensor  $s_m(t - \tau_m)$  by the amount of  $\tau_m$ . The sum of aligned sensor outputs leads to an estimate for the signal:

$$\hat{s}(t) = \frac{1}{M} \sum_{m=1}^M x_m(t + \tau_m) = s(t) + \frac{1}{M} \sum_{m=1}^M n_m(t + \tau_m). \quad (14.21)$$

From the above equation, it is clear that the signal to noise ratio is improved by a factor of  $M$ . For wideband data, the sum in (14.21) is often called *true time-delay beamforming*. It can be approximately implemented using various filtering techniques, see e.g., [6, 7]. For narrow band data, the shift in the time domain becomes phase shift in the frequency domain. Therefore, we have

$$y(n) = \frac{1}{M} \sum_{m=1}^M e^{j\omega\tau_m} x_m(n) = \frac{1}{M} \mathbf{a}(\theta)^H \mathbf{x}(n), \quad (14.22)$$

where  $\mathbf{a}(\theta)^H$  is the steering vector evaluated at the look direction  $\theta$ . Since for any steering vectors,  $\|\mathbf{a}(\theta)^H \mathbf{a}(\theta)\|^2 = M$ , the weight vector has unit length  $\|\mathbf{w}_{\text{conv}} = 1\|$  and

$$\mathbf{w}_{\text{conv}} = \frac{\mathbf{a}(\theta)}{\sqrt{\mathbf{a}(\theta)^H \mathbf{a}(\theta)}}. \quad (14.23)$$

An estimate for the power spectrum is then obtained as

$$P_{\text{conv}}(\theta) = \frac{\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x \mathbf{a}(\theta)}{\mathbf{a}(\theta)^H \mathbf{a}(\theta)}. \quad (14.24)$$

In the context of spectral analysis, the above expression corresponds to the periodogram in the spatial domain [3]. The expected value of  $P_{\text{conv}}(\theta)$  is the convolution between the beam pattern and the true power spectrum. A good beam pattern should be as close to the delta function  $\delta(\theta)$  as possible to minimize leakage from neighboring frequencies.

For a uniform linear array, the beam pattern  $G(\theta, \theta_0) = \frac{1}{M} \mathbf{a}_{\text{ULA}}(\theta_0)^H \mathbf{a}_{\text{ULA}}(\theta)$  has a main lobe around the look direction  $\theta_0$ . The Rayleigh beamwidth, the distance between the first two nulls of  $G(\theta, \theta_0)$ , is approximately given by

$$\theta_{\text{BW}} \approx \frac{2}{|Md \cos \theta_0|}. \quad (14.25)$$

The beamformer can only distinguish two sources with DOA separation larger than half of  $\theta_{\text{BW}}$ . Note that  $d = D/\lambda$  is the ratio between actual distance  $D$  and wavelength  $\lambda$ . As  $\theta_{\text{BW}}$  is inversely proportional to  $Md$ , the aperture of the array, the resolution capability improves with increasing number of sensors and sensor spacing. However,  $d = 1/2$  is the maximally allowed sensor distance. For  $d > 1/2$ , grating lobes appear in the beampattern and create ambiguity in the DOA estimation.

For a standard ULA with  $M = 12$  sensors,  $d = 1/2$ ,  $\theta_0 = 30^\circ$ , the beamwidth  $\theta_{\text{BW}} \approx 0.3849$ , meaning that the DOA separation between two sources must be larger than  $11^\circ$  to generate two peaks in the power spectrum  $P_{\text{conv}}(\theta)$ .

### 3.14.3.2 Minimum variance distortionless response (MVDR) beamformer

The minimum variance distortionless response (MVDR) beamformer [1] alleviates the resolution limit of the conventional beamformer by considering the following constrained optimization problem:

$$\min \mathbf{w}^H \widehat{\mathbf{R}}_x \mathbf{w} \quad (14.26)$$

$$\text{subject to } \mathbf{w}^H \mathbf{a}(\theta) = 1. \quad (14.27)$$

The objective function (14.26) represents the output power to be minimized. The constraint (14.27) ensures that signals from the desired direction  $\theta$  remain undistorted. In other words, while the power from all other directions is minimized, the beamformer concentrates only on the look direction. The behavior of the MVDR beamformer was also discussed by Lacoss [11]. The resulting beampattern has a sharp peak at the target DOA, leading to resolution capability beyond the Rayleigh beamwidth. Applying the method of Lagrange multipliers, we obtain the solution as

$$\mathbf{w}_{\text{MVDR}} = \frac{\widehat{\mathbf{R}}_x^{-1} \mathbf{a}(\theta)}{\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x^{-1} \mathbf{a}(\theta)}. \quad (14.28)$$

Replacing (14.28) into (14.20), one obtains the power function as

$$P_{\text{MVDR}}(\theta) = \frac{1}{\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x^{-1} \mathbf{a}(\theta)}. \quad (14.29)$$

A condition on the MVDR beamformer follows immediately from (14.28) where the inversion of the sample covariance matrix requires that  $\widehat{\mathbf{R}}_x$  is full rank, implying that the number of samples must be larger than the number of sensors, i.e.,  $N \geq M$ . When rank deficiency occurs or the sample number is small compared to the number of sensors, a popular technique known as *diagonal loading* is often employed to improve robustness. As the name implies, the sample covariance matrix (14.14) is modified by adding a small perturbation term to improve the conditioning:

$$\widetilde{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}(n)^H + \sigma_\epsilon \mathbf{I}, \quad (14.30)$$

where  $\sigma_\epsilon$  is a properly chosen small number and  $\mathbf{I}$  is an  $M \times M$  identity matrix. The choice of the coefficient  $\sigma_\epsilon$  is essential. Several criteria for optimal choice of diagonal loading have been reported in [12] and the references therein.

A variant of the MVDR beamformer, proposed in [13], replaces the constraint (14.27) by  $\mathbf{w}^H \mathbf{w} = 1$ . This formulation leads to the adapted angular response spectrum  $\frac{\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x^{-1} \mathbf{a}(\theta)}{\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x^{-2} \mathbf{a}(\theta)}$ . This Borgiotti-Kaplan beamformer is known to provide higher spatial resolution than that of the MVDR beamformer. In [14], the denominator of (14.29) is replaced by  $\mathbf{a}(\theta)^H \widehat{\mathbf{R}}_x^{-k} \mathbf{a}(\theta)$  where  $k > 1$ . Simulation results therein show that using higher order covariance matrix has superior resolution capability and robustness against signal correlation and low SNR.

The performance of the MVDR beamformer depends on the number of snapshots, array aperture, and SNR. Several interesting results are reported in [15]. In the presence of coherent or strongly correlated interferences, the performance of the MVDR beamformer degrades dramatically. Alternative methods addressing this issue are reported in [16–21]. Due to the distortionless response constraint, the MVDR beamformer is sensitive to errors in the target direction and array response imperfection. Robust methods to tackle this problem have been suggested in [22, 23].

### 3.14.3.3 Sparse data representation based approach

The beamforming methods localize the signal sources by estimating the power spectrum associated with various DOAs. Since the number of signals is usually small in array processing, the methods proposed in [24, 25] view DOA estimation as sparse data reconstruction and assign DOA estimates to signals with nonzero amplitude. In this approach, the first step is to find a sparse representation of the array output data. The beamforming output in the frequency domain [24] or the array observation (14.12) [25] can be used to construct a sparse data representation. Then the underlying optimization problem (typically convex) will be solved to find nonzero components. DOA estimates are obtained from angles associated with nonzero components. Recently this approach has attracted many researchers' attention thanks to advances in the theory and methodology of sparse data reconstruction [26].

For representational convenience, we follow the formulation in [25]. Let  $\tilde{\boldsymbol{\theta}} = [\tilde{\theta}_1, \tilde{\theta}_2, \dots, \tilde{\theta}_{N_\theta}]$  be a sampling grid of source locations of interest. An important assumption here is that the number of signals is much smaller than the number of sample grids, i.e.,  $P \ll N_\theta$ . The overcomplete array manifold matrix  $\widetilde{\mathbf{A}} = [\mathbf{a}(\tilde{\theta}_1), \mathbf{a}(\tilde{\theta}_2), \dots, \mathbf{a}(\tilde{\theta}_{N_\theta})]$  consists of  $N_\theta$  steering vectors. The  $N_\theta \times 1$  signal vector  $\tilde{\mathbf{s}}(n)$  has a nonzero component  $\tilde{s}_k$  if a signal source is present at  $\tilde{\theta}_k$ . For a single snapshot, the array output (14.12)

can be re-expressed in terms of the sparse vector  $\tilde{s}(n)$  as:

$$\mathbf{x}(n) = \tilde{\mathbf{A}}\tilde{s}(n) + \mathbf{n}(n). \quad (14.31)$$

Now the problem reduces to finding the nonzero component in  $\tilde{s}(n)$ . In the noiseless case, the ideal measure would be  $\|\tilde{s}(n)\|_0^0$  which counts the nonzero entries. This would in fact lead to the deterministic ML method over a grid search. But this metric will lead to a complicated combinatorial optimization problem. Therefore, one tries to approximate the solution by using  $l_1$  norm,  $\|\tilde{s}(n)\|_1$ . The significant advantage of  $l_1$  relaxation is that the convex optimization problem,

$$\min_{\tilde{s}(n)} \|\tilde{s}(n)\|_1 \text{ subject to } \mathbf{x}(n) = \tilde{\mathbf{A}}\tilde{s}(n) \quad (14.32)$$

has a unique global minimum and can be solved by computationally efficient numerical methods such as linear programming. For noisy measurements (14.31), the constraint  $\mathbf{x}(n) = \tilde{\mathbf{A}}\tilde{s}(n)$  can not hold and needs to be relaxed. An appropriate objective function is suggested in [24, 25]

$$\min_{\tilde{s}(n)} \|\mathbf{x}(n) - \tilde{\mathbf{A}}\tilde{s}(n)\| + \delta \|\tilde{s}(n)\|_1, \quad (14.33)$$

where  $\delta$  is a regularization parameter.

For multiple snapshots, we define the data matrix  $\mathbf{X} = [\mathbf{x}(1) \mathbf{x}(2) \dots \mathbf{x}(N)]$ , the signal matrix  $\tilde{\mathbf{S}} = [\tilde{s}(1) \tilde{s}(2) \dots \tilde{s}(N)]$  and the noise matrix  $\mathbf{N} = [\mathbf{n}(1) \mathbf{n}(2) \dots \mathbf{n}(N)]$ . They are related as follows:

$$\mathbf{X} = \tilde{\mathbf{A}}\tilde{\mathbf{S}} + \mathbf{N}. \quad (14.34)$$

To measure sparsity for multiple time samples, we define the  $i$ th row vector of  $\tilde{\mathbf{S}}$  corresponding to a particular DOA grid point  $\theta_i$  as  $\tilde{s}_i = [\tilde{s}_i(1), \tilde{s}_i(2), \dots, \tilde{s}_i(N)]$  and compute its  $l_2$  norm  $\|\tilde{s}_i\|_2$ . Then the spatial sparsity is imposed on the  $N_\theta \times 1$  vector  $\tilde{s}^{l_2} = [\|\tilde{s}_1\|_2, \|\tilde{s}_2\|_2, \dots, \|\tilde{s}_{N_\theta}\|_2]^T$ . The multiple sample version of (14.33) becomes

$$\min_{\tilde{\mathbf{S}}} \|\mathbf{X} - \tilde{\mathbf{A}}\tilde{\mathbf{S}}\|_F^2 + \delta \|\tilde{s}^{l_2}\|_1, \quad (14.35)$$

where  $\|\cdot\|_F$  is the Forbenius norm. The regularization parameter  $\delta$  is a tradeoff between the fit to data and the sparsity. In [24, 25], statistically motivated strategies for selecting  $\delta$  are discussed.

The computational cost for solving (14.35) increases significantly with the number of snapshots  $N$ . A coherent combination based on singular value decomposition (SVD) of the data matrix is suggested in [25]. A mixed norm approach for joint recovery is proposed in [27]. This problem can be avoided when other forms of sparsity are used, for example, the beamforming output in [24] and the covariance matrix in [28]. The resolution capability of this approach is investigated in [29].

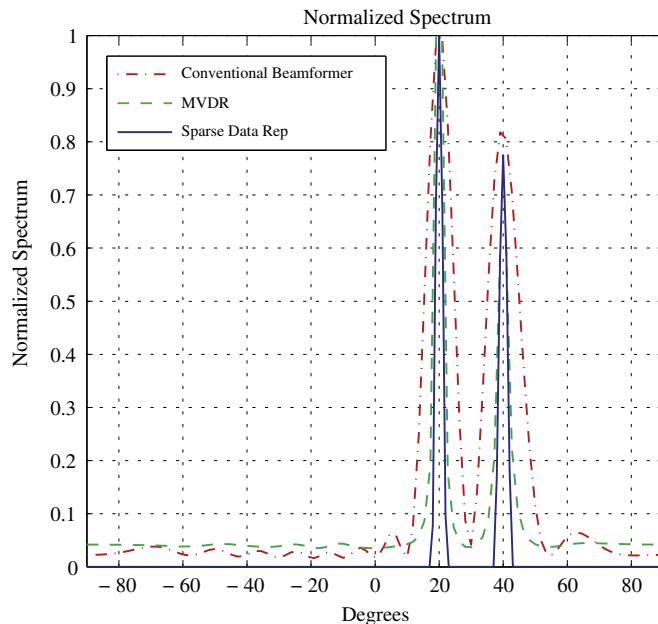
Simulation results in the above mentioned references show that the sparse data representation based approach has a much better resolution than the conventional and MVDR beamformers at the expense of increased computational cost. Another advantage over the subspace methods is the improved robustness against signal coherence. However, for low SNRs and closely located sources, the relatively high bias remains an challenging issue for this approach.

### 3.14.3.4 Numerical examples

In this section, the beamforming based methods discussed previously are tested by numerical experiments. In the simulation, a uniform linear array of 12 sensors with inter-element spacings of half a wavelength is employed. The narrow band signals are generated by  $P = 2$  uncorrelated signals of various strengths. The signal to noise ratio of the first and second signals are given by [5 0] dB, respectively. The number of snapshots is  $N = 200$  in each Monte Carlo trial.

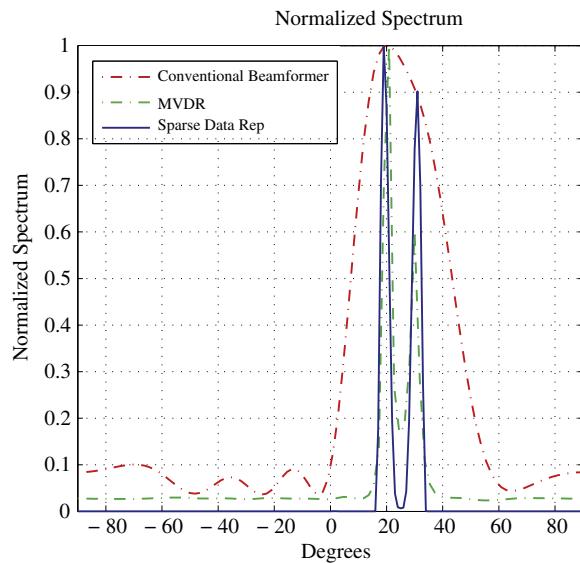
Figure 14.2 shows the normalized spectra obtained from conventional beamformer, MVDR beamformer and sparse data representation over  $-90^\circ$  to  $90^\circ$  for well separated sources located  $\theta = [20^\circ 40^\circ]$  relative to the broadside. All the three methods exhibit two peaks at the reference locations. For the second experiment, the reference DOA parameter is given by  $\theta = [20^\circ 30^\circ]$ , which corresponds to closely located signals. As shown in Figure 14.3, the conventional beamformer only leads to one maximum between the true DOAs and does not recognize the existence of two signals. On the other hand, the MVDR beamformer and the sparse data representation based approach perform well in resolving two signals.

In the third experiment, we compare the estimation accuracy of these methods. To avoid resolution problem, the reference DOA parameter is chosen as  $\theta = [20^\circ 40^\circ]$ . Both signals have equal power with SNR running from  $-5$  dB to  $20$  dB in a 1 dB step. Figure 14.4 depicts the root mean squared

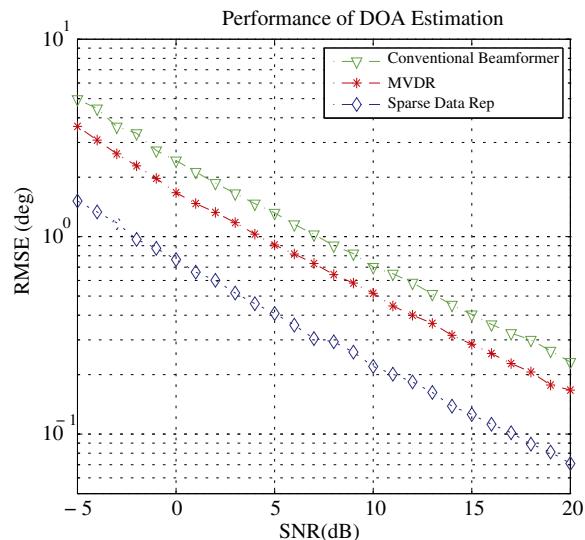


**FIGURE 14.2**

Normalized spectrum for well separated signals. Reference DOA parameter  $\theta = [20^\circ 40^\circ]$ , SNR = [5 0] dB,  $N = 200$ .

**FIGURE 14.3**

Normalized spectrum for closely located signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ , SNR = [5 0] dB,  $N = 200$ .

**FIGURE 14.4**

Estimation performance, RMSE vs. SNR. Reference DOA parameter  $\theta = [20^\circ \ 40^\circ]$ , equal power,  $N = 200$ .

error (RMSE) obtained from 1000 trials. For all three methods, RMSE decreases with increasing SNR. The sparse data representation based method has an overall best performance over the entire SNR range. The MVDR beamformer lies between the other two methods. The gap between these methods becomes most significant at low SNRs. For example, at  $\text{SNR} = -5 \text{ dB}$ , the RMSE of conventional beamformer is  $5^\circ$  which is more than three times that of the sparse data representation based method with  $\text{RMSE} = 1.5^\circ$ .

From the simulation results, we have observed the resolution limitation of the conventional beamformer. Improved resolution capability and estimation accuracy can be achieved by the MVDR beamformer and computationally involved sparse data representation based estimator.

### 3.14.4 Subspace methods

In an attempt to overcome the resolution limit of conventional beamforming, many spectral-like methods were introduced in seventies. They exploit the eigenstructure of the spatial correlation matrix (14.18) to form pseudo spectrum functions. These functions exhibit sharp peaks at the true parameters and lead to superior performance as compared to the Fourier based analysis. While the early work by Pisarenko [30] was devoted to harmonic retrieval, the MUSIC (Multiple SIgnal Classification) algorithm [2,31] was developed for array signal processing.

Recall that  $\mathbf{R}_x$  is a Hermitian symmetric matrix. Therefore, the eigenvectors are orthogonal. From (14.18), it is apparent that the eigenvalues induced by the signal part differs from the remaining ones by the noise level. More specifically, let  $(\lambda_i, \mathbf{u}_i)$ ,  $(i = 1, \dots, M)$  denote eigenvalue/eigenvector pairs of  $\mathbf{R}_x$ , the spectral decomposition of  $\mathbf{R}_x$  can be expressed as

$$\mathbf{R}_x = \sum_{i=1}^M \lambda_i \mathbf{u}_i \mathbf{u}_i^H = \mathbf{U}_s \Lambda_s \mathbf{U}_s^H + \mathbf{U}_n \Lambda_n \mathbf{U}_n^H, \quad (14.36)$$

where  $\Lambda_s = \text{diag}(\lambda_1, \dots, \lambda_P)$ ,  $\mathbf{U}_s = [\mathbf{u}_1, \dots, \mathbf{u}_P]$  and  $\Lambda_n = \text{diag}(\lambda_{P+1}, \dots, \lambda_M)$ ,  $\mathbf{U}_n = [\mathbf{u}_{P+1}, \dots, \mathbf{u}_M]$ . When the signal covariance matrix  $\mathbf{R}_s$  is full rank, i.e.,  $\text{rank}(\mathbf{R}_s) = P$ , the matrix  $\mathbf{A}(\boldsymbol{\theta}) \mathbf{R}_s \mathbf{A}(\boldsymbol{\theta})^H$  has the rank of  $P$ . The eigenvalues satisfy the property:  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_P > \lambda_{P+1} = \dots = \lambda_M = \sigma^2$ . The *signal eigenvectors* corresponding to the  $P$  largest eigenvalues span the same subspace as the steering matrix. The *noise eigenvectors* corresponding to the remaining  $(M - P)$  eigenvalues are orthogonal to the signal subspace. Mathematically, the signal and noise subspaces are related to the steering matrix as follows:

$$\text{sp}(\mathbf{U}_s) = \text{sp}(\mathbf{A}(\boldsymbol{\theta})), \quad \text{sp}(\mathbf{U}_n) \perp \text{sp}(\mathbf{A}(\boldsymbol{\theta})). \quad (14.37)$$

In practice, the analysis is based on the sample covariance matrix  $\widehat{\mathbf{R}}_x$ . The eigenvalues and eigenvectors in (14.36) are then replaced by their estimates  $\widehat{\lambda}_i, \widehat{\mathbf{u}}_i$ . Similarly, the matrices on the right hand side of (14.36) are substituted by corresponding estimates  $\widehat{\mathbf{U}}_s, \widehat{\mathbf{U}}_n, \widehat{\Lambda}_s$  and  $\widehat{\Lambda}_n$ , respectively. For finite samples,  $\widehat{\mathbf{U}}_s \neq \mathbf{U}_s$  and  $\widehat{\mathbf{U}}_n \neq \mathbf{U}_n$ , the property (14.37) is approximately valid. Many efforts have been made to find the best way of combining the estimated signal and noise eigenvectors to achieve high

resolution capability and estimation accuracy. In the following, we assume that the number of signals is known so that the signal and noise subspaces can be separated. Methods for determination of the number of sources will be discussed separately in Section 3.14.7. In the following, we will present the well known MUSIC and ESPRIT algorithms in Sections 3.14.4.1 and 3.14.4.2, respectively. The important issue of signal coherence will be discussed in Section 3.14.4.3.

### 3.14.4.1 MUSIC

The MUSIC algorithm suggested by Schmidt [2, 32], and Bienvenu and Kopp [31] exploits the orthogonality between signal and noise subspaces. From (14.37), we know that any vector  $\mathbf{a}(\theta) \in \text{sp}(\mathbf{A}(\theta))$  satisfies

$$\mathbf{a}^H(\theta)\mathbf{U}_n = \mathbf{0}. \quad (14.38)$$

Assume the array is unambiguous; that is, any collection of  $P$  distinct DOAs  $\{\theta_1, \dots, \theta_p\}$  forms a linearly independent set  $\{\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_p)\}$ . Then the above relation is valid for  $P$  distinct columns of  $\mathbf{A}(\theta)$ . Motivated by this observation, the MUSIC spectrum is defined in terms of the estimated noise eigenvectors as

$$P_{\text{MU}}(\theta) = \frac{1}{\mathbf{a}(\theta)^H \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}_n^H \mathbf{a}(\theta)}. \quad (14.39)$$

For high SNR (or large  $N$ ) and uncorrelated signal sources, the MUSIC spectrum exhibits high peaks near the true DOAs. To find the DOA estimates, we calculate  $P_{\text{MU}}(\theta)$  over a fine grid of  $\theta$  and locate the  $P$  largest maxima of  $P_{\text{MU}}(\theta)$ . In comparison with the MVDR beamformer (14.29), the MUSIC spectrum uses the projection matrix  $\widehat{\mathbf{U}}_n \widehat{\mathbf{U}}_n^H$  rather than  $\widehat{\mathbf{R}}_x^{-1}$ . To get more insight, assume a perfect spatial correlation matrix. Then, for noise eigenvalues  $\sigma^2 = 1$  and  $\mathbf{R}_s^{-1} \rightarrow 0$ ,  $P_{\text{MVDR}}(\theta)$  approaches  $P_{\text{MU}}(\theta)$ . Then MUSIC may be interpreted as a MVDR-like method with a correlation matrix calculated at infinite SNR. This explains the superior resolution of MUSIC than MVDR [3].

In [33, 34], an alternative implementation of MUSIC is suggested to improve estimation accuracy. The idea behind the sequential MUSIC is to find the strongest signal in each iteration and then remove the estimated signal from the observation for the next iteration. As theoretical analysis and numerical results in [34] show the advantage of sequential MUSIC over standard MUSIC is significant for correlated signals. The Toeplitz approximation method [35] provides another implementation of MUSIC specific to uncorrelated sources with a ULA.

For a uniform linear array, MUSIC has a simple implementation. Let  $z = e^{j\phi}$  where  $\phi = 2\pi d \sin \theta$ , the steering vector (14.16) has the form:

$$\mathbf{a}(z) = [1, z, \dots, z^{M-1}]^T. \quad (14.40)$$

The inverse of the MUSIC spectrum (14.39) becomes

$$P_{\text{MU}}(z) = \mathbf{a}(z^{-1})^T \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}_n^H \mathbf{a}(z). \quad (14.41)$$

The root-MUSIC algorithm [36] finds the roots of the complex polynomial function  $z^{M-1} P_{\text{MU}}(z)$  rather than searching for maxima of the MUSIC spectrum. Among the  $(2M - 1)$  possible candidates,  $P$  roots

$\hat{z}_i$ ,  $i = 1, \dots, P$  that are closest to the unit circle on the complex plane are selected to obtain DOA estimates. Since  $\phi = 2\pi d \sin \theta$ , the DOA parameters are given by  $\hat{\theta}_i = \sin^{-1}[\text{angle}(\hat{z}_i)/(2\pi d)]$ . It is known that root-MUSIC has the same asymptotic performance as standard MUSIC. In the finite sample case, root-MUSIC has a much better threshold behavior and improved resolution capability [37]. This is explained by the fact that the radial component of the errors in  $\hat{z}_i$  will not affect  $\hat{\theta}_i$ . Since the search procedure in standard MUSIC is replaced by solving the roots of a polynomial in root-MUSIC, the computational cost is significantly reduced. However, while standard MUSIC is applicable to arbitrary array geometry, root-MUSIC requires a ULA. When ULAs are not available, one can apply array interpolation techniques [38] to approximate the array response. For more details on array interpolation, the reader is referred to Chapter 16 of this book.

The extension of standard MUSIC to the two dimensional case is straightforward. The 2D steering vector (14.8) is used in the MUSIC spectrum, searching for the  $P$  largest maxima over a two dimensional space. For root-MUSIC, an additional ULA is required to be able to resolve both azimuth and elevation [39]. More algorithms and results regarding two dimensional DOA estimation are to be found in Chapter 15 of this book.

While the MUSIC algorithm utilizes all noise eigenvectors, the Minimum Norm (Min-Norm) algorithm suggested in [40,41] uses a single vector in the noise space. A comprehensive study on the resolution capability of MUSIC and the Min-Norm algorithm can be found in [42].

### 3.14.4.2 ESPRIT

The ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) algorithm proposed by Roy and Kailath [43] exploits the rotational invariance property of two identical subarrays and solves the eigenvalues of a matrix relating two signal subspaces. A simple way to construct identical subarrays is to select the first  $(M - 1)$  elements and the second to the  $M$ th elements of a ULA (see Figure 14.5). The array response matrices of the first and second subarrays are then expressed as

$$\mathbf{A}_1(\boldsymbol{\theta}) = \mathbf{J}_1 \mathbf{A}(\boldsymbol{\theta}), \quad \mathbf{A}_2(\boldsymbol{\theta}) = \mathbf{J}_2 \mathbf{A}(\boldsymbol{\theta}), \quad (14.42)$$

where  $\mathbf{J}_1$  and  $\mathbf{J}_2$  are selection matrices of the first and second subarrays. They consist of an  $(M - 1) \times (M - 1)$  identity matrix and an  $(M - 1) \times 1$  zero vector

$$\mathbf{J}_1 = [\mathbf{I}_{M-1} \quad \mathbf{0}] \quad \mathbf{J}_2 = [\mathbf{0} \quad \mathbf{I}_{M-1}]. \quad (14.43)$$

Define  $z_p = e^{j\phi_p}$ ,  $p = 1, \dots, P$ . From (14.42), we know that the two subarrays have the same array response up to a phase shift due to the distance between them. This observation leads to the *shift invariance property*

$$\mathbf{A}_2(\boldsymbol{\theta}) = \mathbf{A}_1(\boldsymbol{\theta})\Phi, \quad (14.44)$$

where  $\Phi = \text{diag}[z_1, \dots, z_P]$ . Note that ESPRIT is applicable for array geometries other than ULAs as long as the shift invariance property holds, see e.g., [43–45].

Recall that the signal subspace of the original array and the signal eigenvectors span the same subspace; therefore, they are related through a nonsingular linear transformation  $\mathbf{T}$ :

$$\mathbf{U}_s = \mathbf{A}(\boldsymbol{\theta})\mathbf{T}. \quad (14.45)$$

Multiplying both sides of (14.45) with  $\mathbf{J}_1$  and  $\mathbf{J}_2$ ,

$$\mathbf{U}_{s_1} = \mathbf{A}_1(\boldsymbol{\theta})\mathbf{T}, \quad \mathbf{U}_{s_2} = \mathbf{A}_2(\boldsymbol{\theta})\mathbf{T}. \quad (14.46)$$

Combining (14.44) and (14.46) yields the relation between  $\mathbf{U}_{s_1}$  and  $\mathbf{U}_{s_2}$ :

$$\mathbf{U}_{s_2} = \mathbf{U}_{s_1}\Psi, \quad \Psi = \mathbf{T}^{-1}\Phi\mathbf{T}. \quad (14.47)$$

Since the matrix  $\Psi$  is similar to the diagonal matrix  $\Phi$ , both matrices have the same eigenvalues,  $z_1, \dots, z_p$ .

Using estimates  $\widehat{\mathbf{U}}_{s_1}, \widehat{\mathbf{U}}_{s_2}$  in (14.47), one can apply LS (Least Squares) or TLS (Total Least Squares) to obtain  $\widehat{\Psi}$  [46]. Finally, DOA estimates are obtained from eigenvalues of  $\widehat{\Psi}$  by the formula  $\hat{\theta}_i = \sin^{-1}[\text{angle}(\hat{z}_i)/(2\pi d)]$ . The computational burden of the ESPRIT algorithm can be reduced by using real-valued operations as proposed in [44].

### 3.14.4.3 Signal coherence

In the development of subspace methods, we have made an important assumption that the rank of the signal covariance matrix  $\mathbf{R}_s$  equals the number of signals  $P$  so that the signal eigenvectors spans the same subspace as the column space of the array manifold matrix. However, this condition no longer holds when two signals are coherent, meaning that the magnitude of the correlation coefficient of the signals is one. Coherent signals are often encountered in wireless communications as a result of a multipath propagation effect or smart jamming in radar systems. In the presence of signal coherence,  $\mathbf{R}_s$  becomes rank deficient, leading to divergence of signal eigenvectors into the noise subspace. Since the property (14.38) is not satisfied, performance of subspace methods degrades significantly. To mitigate the effect of signal coherence, one could apply *forward-backward averaging* or *spatial smoothing* techniques. The former requires a ULA and can handle two coherent signals. The latter requires arrays with a translational invariance property and is able to deal with maximally  $P$  coherent signals.

Let  $\mathbf{E}$  denote the exchange matrix comprised of ones on the anti-diagonal and zeros elsewhere. For a ULA, the steering vector (14.16) has an interesting property:

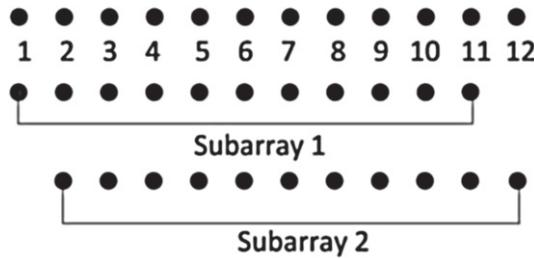
$$\mathbf{E}\mathbf{a}(\boldsymbol{\theta})^* = e^{-j(M-1)\phi}\mathbf{a}(\boldsymbol{\theta}), \quad (14.48)$$

which implies

$$\mathbf{E}\mathbf{A}(\boldsymbol{\theta})^* = \Phi^{-(M-1)}\mathbf{A}(\boldsymbol{\theta}), \quad (14.49)$$

where  $\Phi = \text{diag}[e^{j\phi_1}, \dots, e^{j\phi_P}]$ . The backward covariance is defined as

$$\mathbf{R}_B = \mathbf{E}\mathbf{R}_x^*\mathbf{E}. \quad (14.50)$$

**FIGURE 14.5**

Array and subarrays.

The forward-backward covariance matrix is obtained by averaging of  $\mathbf{R}_B$  and the standard array covariance matrix  $\mathbf{R}_x$ :

$$\mathbf{R}_{FB} = \frac{1}{2}(\mathbf{R}_x + \mathbf{E}\mathbf{R}_x^*\mathbf{E}) = \mathbf{A}(\boldsymbol{\theta})\tilde{\mathbf{R}}_s\mathbf{A}(\boldsymbol{\theta})^H + \sigma^2\mathbf{I}. \quad (14.51)$$

Applying the property (14.49), the modified signal covariance matrix is then given by

$$\tilde{\mathbf{R}}_s = \frac{1}{2}(\mathbf{R}_s + \Phi^{-(M-1)}\mathbf{R}_s^*\Phi^{(M-1)}). \quad (14.52)$$

The coherent signals are de-correlated through phase modulation by the diagonal elements of  $\Phi^{-(M-1)}$ . The forward-backward ESPRIT is equivalent to the unitary ESPRIT [44], because it only uses real valued components.

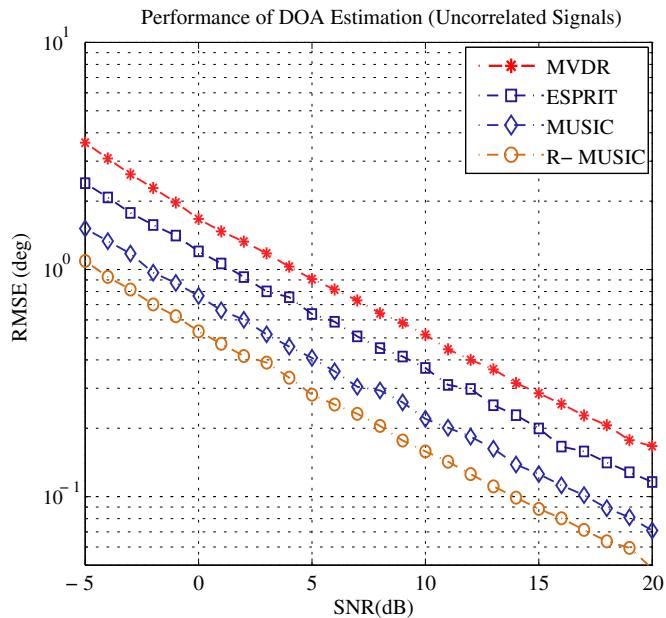
In a general case where more than two coherent signals are present, we need a more powerful solution to restore the rank of the signal covariance matrix. The *spatial smoothing* technique, first proposed in [47] and extended in [38,48,49], exploits the degrees of freedom of a regular array by splitting it into several identical subarrays. Let  $M_s$  denote the number of sensors of a subarray. For a ULA of  $M$  elements, the maximal number of subarrays is  $L = (M - M_s + 1)$ . The array response matrix of the  $l$ th subarray is related to  $\mathbf{A}(\boldsymbol{\theta})$  as

$$\mathbf{A}_l(\boldsymbol{\theta}) = \mathbf{E}_l\mathbf{A}(\boldsymbol{\theta}), \quad (14.53)$$

where the selection matrix is  $\mathbf{E}_l = [\mathbf{0}_{M_s \times (l-1)} \mathbf{I}_{M_s} \mathbf{0}_{M_s \times (M-M_s-l+1)}]$ . The spatially averaged covariance matrix is then given by

$$\mathbf{R}_{SS} = \frac{1}{L} \sum_{l=1}^L \mathbf{E}_l \mathbf{R}_x \mathbf{E}_l^T. \quad (14.54)$$

In the case of a ULA, one can combine the forward-backward averaging (14.51) with the spatial smoothing. In [48,50], it was shown that under mild conditions, the signal covariance matrix obtained from forward-backward averaging and spatial smoothing is nonsingular. Since the subarrays have a smaller aperture than the original array, the signal coherency is removed at the expense of resolution capability. The two dimensional extension of spatial smoothing is addressed in [51–53].

**FIGURE 14.6**

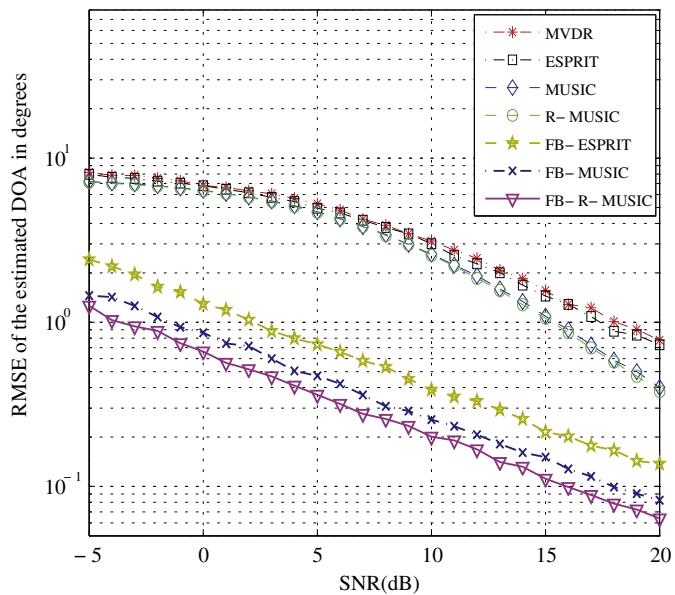
RMSE vs. SNR. Uncorrelated signals. Reference DOA parameter  $\theta = [20^\circ \ 40^\circ]$ ,  $N = 200$ .

### 3.14.4.4 Numerical examples

We demonstrate the performance of the subspace methods presented previously by numerical results. A uniform linear array of 12 sensors with inter-element spacings of half a wavelength is employed. In this case, the application of root-MUSIC is straightforward. Narrow band signals are generated by  $P = 2$  well separated sources of equal strengths located at  $\theta = [20^\circ \ 40^\circ]$ . The number of snapshots is  $N = 200$ . Both signals are equal power with SNR running from  $-5$  dB to  $20$  dB. The two subarrays used in ESPRIT are ULAs comprised of 11 elements.

In the first experiment, we consider uncorrelated signals. For comparison, the MVDR beamformer is applied to the same batch of data. The empirical RMSE is obtained from 1000 trials. From Figure 14.6, one can observe that root-MUSIC, denoted by R-MUSIC, outperforms standard MUSIC and ESPRIT. All the subspace methods have lower estimation errors than the MVDR beamformer. The performance difference is most significant at low SNRs. For SNR close to 20 dB, all methods behave similarly. The superior performance of root-MUSIC compared to standard MUSIC is expected as predicted by the theoretical analysis [37]. The estimation error of ESPRIT is higher than both MUSIC algorithms due to the reduced aperture.

In the second experiment, two correlated signals are considered with real valued correlation coefficient  $\rho = 0.92$ . One can observe from Figure 14.7 that the performance of all methods degrade rapidly. At  $\text{SNR} = -5$  dB, the RMSE is more than twice the RMSE in the uncorrelated case. Both MUSIC-based

**FIGURE 14.7**

RMSE vs. SNR. Correlated signals. Reference DOA parameter  $\theta = [20^\circ \ 40^\circ]$ ,  $N = 200$ .

algorithms have almost identical performance. They are slightly better than ESPRIT. The curve of ESPRIT almost coincides with that of MVDR. These observations indicate that subspace methods are very sensitive to signal correlation and can not provide accurate estimates when rank deficiency occurs. If we use the forward-backward averaged sample covariance matrix (14.51) in the subspace methods, denoted by FB-ESPRIT, FB-MUSIC, and FB-R-MUSIC, respectively, the estimation accuracy improves significantly as the three curves at the bottom show. In comparison with Figure 14.6, the RMSE increases only slightly when the FB technique is applied. For a detailed discussion on highly correlated signals, the reader is referred to Chapter 15 of this book.

### 3.14.5 Parametric methods

The spectral-like methods presented previously treat the direction finding problem as *spatial frequency* estimation. Although subspace methods overcome the resolution limitation of beamforming techniques, and yield good estimates at reasonable computational cost, the performance degrades dramatically in the presence of correlated or coherent signals. Parametric methods exploit the data model directly and are usually statistically well motivated. The well known maximum likelihood (ML) approach is representative of this class of estimators. Since parametric methods are not dependent on the eigenstructure of the sample covariance matrix, they provide reasonable results in scenarios involving signal correlation/coherence, low SNRs and small data samples. The price for the improved robustness

and accuracy is the increased computational complexity. We will introduce the maximum likelihood approach and the covariance matching estimation methods in Sections 3.14.5.1 and 3.14.5.4, respectively. Several numerical algorithms for efficient implementation of the ML estimator will be presented in Section 3.14.5.2. Analytical results on the performance of DOA estimators presented so far will be discussed in Section 3.14.5.5.

### 3.14.5.1 The maximum likelihood approach

The maximum likelihood method is a systematic tool for constructing estimators. Based on the statistical model for data samples, it maximizes the likelihood function over the parameters of interest to derive estimates. The well known properties of ML estimation include asymptotic normality and efficiency under proper conditions [54]. For DOA estimation, the application is straightforward. Recall the data model in (14.17)

$$\mathbf{x}(n) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n) + \mathbf{n}(n). \quad (14.55)$$

We assume the noise  $\mathbf{n}(n)$  is temporally independent and complex normally distributed with zero mean and covariance matrix  $\mathbf{R}_n = \sigma^2 \mathbf{I}$ , i.e.,  $\mathbf{n}(n) \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I})$ . In the array processing literature, two different interpretations of the source signals lead to the *deterministic* and the *stochastic* ML estimators.

#### 3.14.5.1.1 Deterministic maximum likelihood

In the deterministic ML approach, the signal  $\mathbf{s}(n)$  is viewed as a fixed realization of a stochastic process; the parameters  $\mathbf{s}(1), \mathbf{s}(2), \dots, \mathbf{s}(N)$  are considered to be deterministic and unknown. With the above noise assumption, the array output  $\mathbf{x}(n)$  is complex normally distributed with mean  $\mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n)$  and covariance matrix  $\sigma^2 \mathbf{I}$ , i.e.,  $\mathbf{x}(n) \sim \mathcal{CN}(\mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n), \sigma^2 \mathbf{I})$ . Because the array outputs  $\mathbf{x}(n)$ ,  $n = 1, \dots, N$  are independent, the joint likelihood function is the product of the likelihood associated with each snapshot, i.e.,

$$l_d(\boldsymbol{\theta}, \mathbf{s}(1), \dots, \mathbf{s}(N), \sigma^2) = \prod_{n=1}^N l_d(\boldsymbol{\theta}, \mathbf{s}(n), \sigma^2) = \prod_{n=1}^N \frac{1}{(\pi\sigma^2)^M} \exp\left(-\frac{1}{\sigma^2} \|\mathbf{x}(n) - \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n)\|^2\right), \quad (14.56)$$

where  $\|\cdot\|$  denotes the Euclidean norm. The log-likelihood function is then given by

$$\mathcal{L}_d(\boldsymbol{\theta}, \mathbf{s}(1), \dots, \mathbf{s}(N), \sigma^2) = - \sum_{n=1}^N M \log(\pi\sigma^2) + \frac{1}{\sigma^2} \|\mathbf{x}(n) - \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(n)\|^2. \quad (14.57)$$

The unknown parameters  $\boldsymbol{\vartheta}_d = \{\boldsymbol{\theta}, \mathbf{s}(1), \dots, \mathbf{s}(N), \sigma^2\}$  include the DOA parameter and nuisance parameters. As the number of the signal waveform parameters increases with increasing number of snapshots, the high dimension of parameter space will make direct optimization of (14.57) infeasible. It is well known that the likelihood function is separable [55, 3], and the likelihood can be concentrated with respect to the linear parameters. For a fixed, unknown  $\boldsymbol{\theta}$ , the ML estimate of the signal is given by

$$\hat{\mathbf{s}}(n) = \mathbf{A}^\ddagger(\boldsymbol{\theta})\mathbf{x}(n), \quad (14.58)$$

where  $A^\sharp(\theta)$  denotes the Moore-Penrose pseudo inverse. For a full column rank  $A(\theta)$ , it is given by  $A^\sharp(\theta) = (A(\theta)^H A(\theta))^{-1} A(\theta)^H$ . Replacing  $s(n)$  with  $\hat{s}(n)$  in (14.57), and maximizing the resulting likelihood over  $\sigma^2$ , we obtain the ML estimate

$$\hat{\sigma}^2 = \frac{1}{M} \text{tr}(\mathbf{P}^\perp(\theta) \widehat{\mathbf{R}}_x), \quad (14.59)$$

where  $\mathbf{P}^\perp(\theta) = \mathbf{I} - \mathbf{P}(\theta)$  is the orthogonal complement of the projection matrix  $\mathbf{P}(\theta) = A(\theta)A^\sharp(\theta)$ . Replacing  $\sigma^2$  with  $\hat{\sigma}^2$  in the likelihood function again, we obtain the concentrated likelihood function:

$$L_d(\theta) = -\log \text{tr}(\mathbf{P}^\perp(\theta) \widehat{\mathbf{R}}_x). \quad (14.60)$$

Since  $\log(\cdot)$  is a monotonically increasing function, the ML estimate can be obtained from maximizing the function  $L_d(\theta)$ , or equivalently,

$$\hat{\theta}_{DML} = \arg \min_{\theta} \text{tr}(\mathbf{P}^\perp(\theta) \widehat{\mathbf{R}}_x). \quad (14.61)$$

The signal waveform and noise parameters can be computed by replacing  $\theta$  with the estimate  $\hat{\theta}_{DML}$  into (14.58) and (14.59), respectively. Combining the criterion (14.60) and the noise estimate (14.59) shows that the deterministic ML estimate minimizes the distance between the observation and the model which is represented by the estimated noise power.

### 3.14.5.1.2 Stochastic maximum likelihood

In the stochastic ML, the signal  $s(n)$  is considered as a complex normal random process with zero mean and covariance matrix  $\mathbf{R}_s = E[s(n)s(n)^H]$ . Assuming independent, spatially white noise as in the deterministic case, the array observation  $\mathbf{x}(n)$  is normally distributed with zero mean and covariance matrix  $\mathbf{R}_x$ , i.e.,  $\mathcal{CN}(0, \mathbf{R}_x)$  where  $\mathbf{R}_x = A(\theta)\mathbf{R}_s A(\theta)^H + \sigma^2 \mathbf{I}$ . The joint log-likelihood function for the stochastic signal model is given by

$$l_s(\theta, \mathbf{S}, \sigma^2) = \prod_{n=1}^N \frac{1}{\pi^M \det \mathbf{R}_x} \exp \left( -\mathbf{x}(n)^H \mathbf{R}_x^{-1} \mathbf{x}(n) \right), \quad (14.62)$$

where the vector  $\mathbf{S}$  includes  $P^2$  unknown entries in the signal covariance matrix  $\mathbf{R}_s$ . Taking logarithm of  $l_s(\cdot)$  and omitting constants, we obtain

$$\mathcal{L}_s(\theta, \mathbf{S}, \sigma^2) = -\log \det \mathbf{R}_x - \text{tr} \left( \mathbf{R}_x^{-1} \widehat{\mathbf{R}}_x \right). \quad (14.63)$$

The parameter vector  $\vartheta_s = [\theta, \mathbf{S}, \sigma^2]$  remains the same over the observation interval, unlike the growing parameter vector in the deterministic case. It was shown in [56,57] that the linear parameters in (14.63) have a closed form expression for the ML estimates at an unknown fixed nonlinear parameter  $\theta$  as follows:

$$\hat{\sigma}^2 = \frac{1}{M - P} \text{tr} \left[ \mathbf{P}^\perp(\theta) \widehat{\mathbf{R}}_x \right], \quad (14.64)$$

$$\widehat{\mathbf{R}}_s = A^\sharp(\theta) \left( \widehat{\mathbf{R}}_x - \hat{\sigma}^2 \mathbf{I} \right) A^\sharp(\theta)^H. \quad (14.65)$$

Replacing  $\sigma^2$  and  $\mathbf{R}_s$  with  $\hat{\sigma}^2$  and  $\widehat{\mathbf{R}}_s$ , one obtains the concentrated stochastic likelihood function as

$$L_s(\boldsymbol{\theta}) = -\log \det \left( \mathbf{A}(\boldsymbol{\theta}) \widehat{\mathbf{R}}_s \mathbf{A}(\boldsymbol{\theta})^H + \hat{\sigma}^2 \mathbf{I} \right) \quad (14.66)$$

$$= -\log \det \left[ \mathbf{P}(\boldsymbol{\theta}) \widehat{\mathbf{R}}_x \mathbf{P}(\boldsymbol{\theta}) + \frac{1}{M-P} \text{tr} \left( \mathbf{P}^\perp(\boldsymbol{\theta}) \widehat{\mathbf{R}}_x \right) \mathbf{P}^\perp(\boldsymbol{\theta}) \right]. \quad (14.67)$$

The ML estimate for the DOA parameter is derived by minimizing the negative likelihood function over  $\boldsymbol{\theta}$

$$\hat{\boldsymbol{\theta}}_{\text{SML}} = \arg \min_{\boldsymbol{\theta}} -L_s(\boldsymbol{\theta}). \quad (14.68)$$

Once  $\hat{\boldsymbol{\theta}}_{\text{SML}}$  is available, the signal and noise parameters can be computed from (14.64) and (14.65) by replacing the DOA parameter with its estimate. The criterion (14.66) has a nice interpretation as the generalized variance minimized by the estimated model parameter [3]. If there is only one signal source, the projection matrix is given by  $\mathbf{P}(\boldsymbol{\theta}) = \mathbf{a}(\boldsymbol{\theta}) \mathbf{a}(\boldsymbol{\theta})^H / M$ , and the criterion  $L_s(\boldsymbol{\theta})$  is a monotonically increasing function of  $\mathbf{a}(\boldsymbol{\theta}) \widehat{\mathbf{R}}_x \mathbf{a}(\boldsymbol{\theta})^H$ . The optimum wave parameter maximizes the conventional beamforming output and therefore results in the same estimate as the conventional beamformer.

In the above discussion, the spectral covariance  $\widehat{\mathbf{R}}_s$  is not necessarily positive definite as the optimization was over Hermitian matrices. The positive definiteness of  $\widehat{\mathbf{R}}_s$  is taken into account by imposing a constraint in the optimization process in [58, 59]. Fortunately,  $\widehat{\mathbf{R}}_s$  has full rank with probability 1 for sufficiently large  $N$ , provided the true  $\mathbf{R}_s$  has full rank. The ML estimator for known signal waveforms was developed in [60–62] under various assumptions. The problem of unknown noise structures was discussed in [63].

### 3.14.5.2 Implementation

In the derivation of ML estimators, we have reduced the size of the problem by concentrating the signal and noise parameters. The resulting objective functions: (14.60) and (14.66) depend only on the nonlinear parameters. Maximization of both criteria still requires multi-dimensional searches. Hence, efficient implementation of the ML estimators becomes an important issue. The alternating projection algorithm [64] is an iterative technique for finding the maximum of the concentrated likelihood function (14.60). It performs maximization with respect to a single parameter, while all other parameters are held fixed. In [65], Newton-type methods are suggested for the large sample case.

In the following, we will present the statistically motivated expectation and maximization (EM) and space alternating generalized EM (SAGE) algorithm. The multidimensional search in the original problem can be replaced by several one dimensional maximizations. Both algorithms assume the deterministic signal model so that the suggested augmentation scheme is valid. It is possible to derive EM and SAGE for stochastic ML for uncorrelated signals, see e.g., [66]. When a ULA is available, the iterative quadratic maximum likelihood (IQML) algorithm can be applied to minimize the deterministic likelihood function. A common feature of these methods is that a good initial estimate is required for the convergence to the global maximum. One way to obtain the initial estimate is via other simpler methods such beamforming techniques or subspace methods. Another approach is to optimize the ML criteria directly by stochastic optimization procedures such as the genetic algorithm (GA) [67], simulated annealing [68] and particle swarm method [69] prior to the local maximization algorithms.

### 3.14.5.2.1 EM algorithm

The expectation and maximization (EM) algorithm [70] is a well known iterative algorithm in statistics for locating modes of likelihood functions. Because of its simplicity and stability, it has been applied to many problems since its first appearance. The idea behind EM is quite simple: rather than performing a complicated maximization of the observed data log-likelihood, one augments the observations with imputed values that simplify the maximization and applies the augmented data to estimate the unknown parameters. Because the imputed data are unknown, they are estimated from the observed data. This procedure continues to iterate between the E- and M-steps until no changes occur in the parameter estimates.

Let  $\mathbf{X}$  and  $\mathbf{Y}$  denote the observed and augmented data, respectively. The corresponding density functions are denoted by  $f_X(\mathbf{x}|\boldsymbol{\vartheta})$  and  $f_Y(\mathbf{y}|\boldsymbol{\vartheta})$ . The augmented data  $\mathbf{Y}$  is specified so that  $\mathcal{M}(\mathbf{Y}) = \mathbf{X}$  is a many-to-one mapping. Starting from an initial guess  $\boldsymbol{\vartheta}^{[0]}$ , each iteration of the EM algorithm consists of an expectation (E) step and a maximization (M) step. At the  $(i+1)$ st iteration, ( $i = 0, 1, \dots$ ), the E-step evaluates the conditional expectation of the augmented data log-likelihood  $\log f_Y(\mathbf{y}|\boldsymbol{\vartheta})$  given the observed data  $\mathbf{x}$  and the  $i$ th iterate  $\boldsymbol{\vartheta} = \boldsymbol{\vartheta}^{[i]}$ :

$$Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{[i]}) = E \left[ \log f_Y(\mathbf{y}|\boldsymbol{\vartheta}) | \mathbf{x}, \boldsymbol{\vartheta}^{[i]} \right]. \quad (14.69)$$

For notational simplicity,  $\mathbf{y}$  is also used to denote a random vector in expressions like (14.69). The M-step determines  $\boldsymbol{\vartheta}^{[i+1]}$  by maximizing the expected augmented data log-likelihood

$$\boldsymbol{\vartheta}^{[i+1]} = \arg \max_{\boldsymbol{\vartheta}} Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{[i]}). \quad (14.70)$$

A simple proof based on Jensen's inequality [71] shows that the observed data likelihood increases monotonically or never decreases with iterations [70]. As with most optimization techniques, EM is not guaranteed to always converge to a unique global maximum. In well-behaved problems  $\log f_X(\mathbf{x}|\boldsymbol{\vartheta})$  is unimodal and concave over the entire parameter space, then EM converges to the maximum likelihood estimate from any starting value [70, 72].

For DOA estimation, the observed data consists of the array outputs  $\mathbf{x}(n)$ ,  $n = 1, \dots, N$ . For the deterministic signal model, the augmented data  $\mathbf{y}(n)$  is constructed by decomposing  $\mathbf{x}(n)$  virtually into its signal and noise parts [73]:

$$\mathbf{y}(n) = \begin{bmatrix} \mathbf{y}_1(n) \\ \vdots \\ \mathbf{y}_P(n) \end{bmatrix} = \begin{bmatrix} \mathbf{a}(\theta_1)s_1(n) \\ \vdots \\ \mathbf{a}(\theta_P)s_P(n) \end{bmatrix} + \begin{bmatrix} \mathbf{n}_1(n) \\ \vdots \\ \mathbf{n}_P(n) \end{bmatrix}, \quad (14.71)$$

where the  $M \times 1$  vectors  $\mathbf{y}_p(n)$ ,  $p = 1, \dots, P$  are independent and complex normally distributed as  $\mathcal{CN}(\mathbf{a}(\theta_p)s_p(n), \sigma_p^2 \mathbf{I})$ . The noise parameters are positive and must satisfy the constraint  $\sum_{p=1}^P \sigma_p^2 = \sigma^2$ . The total unknown parameter vector is given by  $\boldsymbol{\vartheta} = [\boldsymbol{\vartheta}_1, \boldsymbol{\vartheta}_2, \dots, \boldsymbol{\vartheta}_P]$ , where  $\boldsymbol{\vartheta}_p = [\theta_p, s_p(1), \dots, s_p(N), \sigma_p^2]$ . Through data augmentation, maximization of  $\log f_Y(\mathbf{y}|\boldsymbol{\vartheta})$  can be performed over distinct parameter sets  $\boldsymbol{\vartheta}_p$ ,  $p = 1, \dots, P$  in parallel. For the unknown noise case, the  $(i+1)$ st iteration proceeds as follows [74].

*E-step:* Calculate the conditional mean  $\hat{\mathbf{y}}_p(n)$  and correlation matrix  $\hat{\mathbf{C}}_{\mathbf{y}_p}$  of  $\mathbf{y}_p(n)$ :

$$\begin{aligned}\hat{\mathbf{y}}_p(n) &= \mathbf{a}(\theta_p^{[i]}) s_p(n)^{[i]} + \frac{\sigma_p^{2[i]}}{\sigma^{2[i]}} (\mathbf{x}(n) - \mathbf{A}(\theta^{[i]}) \mathbf{s}(n)^{[i]}), \\ \hat{\mathbf{C}}_{\mathbf{y}_p} &= \frac{1}{N} \sum_{n=1}^N \hat{\mathbf{y}}_p(n) \hat{\mathbf{y}}_p(n)^H + \frac{(\sigma_p^{2[i]})^2}{\sigma_p^{2[i]}} \mathbf{I}.\end{aligned}$$

*M-step:* Update  $\theta_p$  for  $p = 1, \dots, P$

$$\theta_p^{[i+1]} = \arg \max_{\theta_p} \mathbf{a}(\theta_p)^H \hat{\mathbf{C}}_{\mathbf{y}_p} \mathbf{a}(\theta_p), \quad (14.72)$$

Given the estimate (14.72), the signal parameter  $s_p(n)^{[i+1]}$  can be obtained from (14.58) by replacing  $\mathbf{x}(n)$  with  $\hat{\mathbf{y}}_p(n)$ , using  $\mathbf{A}(\theta) = \mathbf{a}(\theta_p^{[i+1]})$ . Similarly, the noise parameter  $\sigma_p^{2[i+1]}$  is obtained from (14.59) by replacing  $\hat{\mathbf{R}}_x$  with  $\hat{\mathbf{C}}_{\mathbf{y}_p}$ .

The M-step (14.72) requires only a one dimensional search. The multi-dimensional nonlinear optimization in the original problem is greatly simplified by data augmentation. The EM algorithms for the known noise case [73, 75] and the stochastic signal model [76] also demonstrate this computational advantage. A major shortcoming of EM is that it may converge slowly. To address this issue, we will discuss an implementation based on a more flexible augmentation scheme in the following.

To improve the convergence rate, the space alternating generalized EM (SAGE) algorithm [77] is derived in [78, 79].

### 3.14.5.2.2 SAGE algorithm

The space alternating generalized EM (SAGE) algorithm [77] generalizes the idea of data augmentation to simplify computations of the EM algorithm. Instead of estimating all parameters at once, SAGE breaks up the problem into several smaller problems by conditioning sequentially on a subset of the parameters and then applies EM to each reduced problem. Because each of the reduced problems considers the likelihood as a function of a different subset of parameters, it is natural to use a different augmentation scheme for each of the corresponding EM algorithms [77, 80]. In some settings, this attempt turns out to be very useful for speeding up the algorithm.

Unlike the EM algorithm, each iteration of SAGE consists of several cycles. The parameter subset associated with the  $c$ th cycle  $\boldsymbol{\eta}_c$  is updated by maximizing the conditional expectation of log-likelihood  $\log f_{\mathbf{Z}_c}(\mathbf{z}_c | \boldsymbol{\eta}_c)$  of the augmented data  $\mathbf{Z}_c$ . The data augmentation schemes are allowed to vary between cycles. Within one iteration, every element of the parameter vector  $\boldsymbol{\eta}$  must be updated at least once. Let  $\tilde{\boldsymbol{\eta}}_c$  be the vector containing all parameters of  $\boldsymbol{\eta}$  except the elements of  $\boldsymbol{\eta}_c$ . Then  $\boldsymbol{\eta} = (\boldsymbol{\eta}_c, \tilde{\boldsymbol{\eta}}_c)$  is a partition of the parameter set at the  $c$ th cycle. The estimate at the  $c$ th cycle,  $i$ th iteration is represented by  $(\cdot)^{[i,c]}$ . The output of the last cycle of the  $i$ th iteration is used as the input of the  $(i+1)$ st iteration:  $\boldsymbol{\eta}^{[i+1,0]} = \boldsymbol{\eta}^{[i,C]}$ . Starting from the initial estimate  $\boldsymbol{\eta}^{[0,0]}$ , the  $(i+1)$ st iteration of the SAGE algorithm proceeds as follows.

For  $c = 1, \dots, C$

*E-step:* Compute

$$Q^{[c]}(\boldsymbol{\eta}_c, \boldsymbol{\eta}^{[i+1,c-1]}) = E \left[ \log f_{\mathbf{Z}_c}(\mathbf{z}_c | \boldsymbol{\eta}_c) | \mathbf{x}, \boldsymbol{\eta}^{[i+1,c-1]} \right]. \quad (14.73)$$

*M-step:* Update  $\boldsymbol{\eta}_c$  by maximizing  $Q^{[c]}(\boldsymbol{\eta}_c, \boldsymbol{\eta}^{[i+1,c-1]})$  with respect to  $\boldsymbol{\eta}_c$

$$\boldsymbol{\eta}_c^{[i+1,c]} = \arg \max_{\boldsymbol{\eta}_c} Q^{[c]}(\boldsymbol{\eta}_c, \boldsymbol{\eta}^{[i+1,c-1]}), \quad (14.74)$$

$$\boldsymbol{\eta}^{[i+1,c]} = (\boldsymbol{\eta}_c^{[i+1,c]}, \tilde{\boldsymbol{\eta}}_c^{[i+1,c-1]}). \quad (14.75)$$

Similarly to EM, it can be shown that any sequence generated by the above procedure increases (or maintains)  $\log f_X(\mathbf{x} | \boldsymbol{\theta})$  at every cycle [77].

A natural augmentation scheme for DOA estimation is to consider one source at each cycle:

$$\mathbf{z}_c(n) = \mathbf{a}(\theta_c) s_c(n) + \mathbf{n}(n). \quad (14.76)$$

Compared to the augmentation scheme specified in (14.71),  $\mathbf{z}_c(n)$  is more noisy since the whole noise component is fully incorporated in every cycle. The parameter vector associated with the  $c$ th cycle is given by  $\boldsymbol{\eta}_c = (\theta_c, s_c(1), \dots, s_c(N), \sigma^2)$ . The  $c$ th cycle in the  $(i+1)$ st iteration is as follows:

*E-step:* Calculate the conditional mean  $\hat{\mathbf{z}}_c(n)$  and correlation matrix  $\hat{\mathbf{C}}_{\mathbf{z}_c}$ .

$$\hat{\mathbf{z}}_c(n) = \mathbf{a}(\boldsymbol{\theta}_c^{[i+1,c-1]}) s_c(n)^{[i+1,c-1]} + \mathbf{x}(n) - \mathbf{A}(\boldsymbol{\theta}^{[i+1,c-1]}) \mathbf{s}(n)^{[i+1,c-1]}, \quad (14.77)$$

$$\hat{\mathbf{C}}_{\mathbf{z}_c} = \frac{1}{N} \sum_{n=1}^N \hat{\mathbf{z}}_c(n) \hat{\mathbf{z}}_c(n)^H. \quad (14.78)$$

*M-step:* Update  $\theta_c$

$$\boldsymbol{\theta}_c^{[i+1,c]} = \arg \max_{\boldsymbol{\theta}_c} \mathbf{a}(\boldsymbol{\theta}_c)^H \hat{\mathbf{C}}_{\mathbf{z}_c} \mathbf{a}(\boldsymbol{\theta}_c). \quad (14.79)$$

The computational complexity for each iteration of EM and SAGE is almost the same. The total computational cost is determined by the convergence rate. It has been shown in [74] that the SAGE algorithm converges faster than the EM algorithm when certain conditions on observed and augmented information matrices are satisfied.

### 3.14.5.2.3 Iterative quadratic maximum likelihood

The iterative quadratic maximum likelihood (IQML) algorithm [81], also proposed independently in [82], can trace its root back to system identification methods [83]. Unlike EM and SAGE which are applicable to arbitrary arrays, the IQML algorithm requires a ULA so that the array response matrix  $\mathbf{A}(\boldsymbol{\theta})$  has a Vandermonde structure:

$$\mathbf{A}(\boldsymbol{\theta}) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j\phi_1} & e^{j\phi_2} & \cdots & e^{j\phi_P} \\ \vdots & \vdots & \cdots & \vdots \\ e^{j(M-1)\phi_1} & e^{j(M-1)\phi_2} & \cdots & e^{j(M-1)\phi_P} \end{pmatrix}, \quad (14.80)$$

where  $\phi_p = 2\pi d \sin \theta_p$ ,  $p = 1, \dots, P$  according to (14.15). To re-parameterize the likelihood function (14.60), one defines a polynomial with roots  $z_p = e^{j\phi_p}$ ,  $p = 1, \dots, P$  as

$$b(z) = z^P + b_1 z^{P-1} + \dots + b_P = \prod_{p=1}^P (z - e^{j\phi_p}). \quad (14.81)$$

By construction, the Toeplitz matrix  $\mathbf{B}^H$

$$\mathbf{B}^H = \begin{pmatrix} b_P & b_{P-1} & \cdots & 1 & \cdots & 0 \\ \ddots & \ddots & & & \ddots & 0 \\ 0 & & b_P & b_{P-1} & \cdots & 1 \end{pmatrix} \quad (14.82)$$

is full rank and satisfies the following relation:

$$\mathbf{B}^H \mathbf{A}(\boldsymbol{\theta}) = 0. \quad (14.83)$$

In other words, the column space of  $\mathbf{B}$  is orthogonal to that of  $\mathbf{A}(\boldsymbol{\theta})$ . Therefore, the projection matrix in (14.60) can be reformulated as

$$\mathbf{P}^\perp(\boldsymbol{\theta}) = \mathbf{B}(\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H. \quad (14.84)$$

Then the likelihood criterion (14.60) can be re-parameterized in terms of the polynomial coefficients  $\mathbf{b} = [b_1, \dots, b_P]$

$$L_{\text{IQML}}(\mathbf{b}) = -\text{tr}(\mathbf{B}(\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H \hat{\mathbf{R}}_x). \quad (14.85)$$

Maximizing  $L_{\text{IQML}}(\mathbf{b})$  leads to an estimate for the coefficient vector  $\hat{\mathbf{b}}$ . With  $\mathbf{b}$  replaced by  $\hat{\mathbf{b}}$ , the DOA estimate is computed by finding the roots of (14.81).

Minimization of the criterion (14.85) is still a complicated optimization problem. However, if the matrix  $\mathbf{Q} = (\mathbf{B}^H \mathbf{B})^{-1}$  is replaced by a known matrix, the criterion becomes quadratic in  $\mathbf{b}$  and has a closed form solution. This observation leads to the iterative algorithm suggested in [81]. Setting the initial value  $\mathbf{Q}^{(0)} = \mathbf{I}$  and denoting the  $i$ th iterate as  $\hat{\mathbf{b}}^{(i)}$ , the  $(i+1)$ st iteration of the IQML algorithm proceeds as follows:

1. Compute  $\mathbf{Q}^{(i)} = (\mathbf{B}^{(i)H} \mathbf{B}^{(i)})^{-1}$ .

2. Find  $\hat{\mathbf{b}}^{(i)}$  by solving

$$\hat{\mathbf{b}}^{(i+1)} = \arg \max_{\mathbf{b}} -\text{tr}(\mathbf{B} \mathbf{Q}^{(i)} \mathbf{B}^H \hat{\mathbf{R}}_x). \quad (14.86)$$

The procedure continues until the distance between two consecutive iterates is less than a pre-specified small number  $\delta$ , i.e.,  $\|\mathbf{b}^{(i+1)} - \mathbf{b}^{(i)}\| \leq \delta$ . To ensure the roots of the polynomial with estimated coefficients lie on the unit circle, constraints on  $\mathbf{b}$  have been suggested in [81, 84]. For example, since  $b(z)$  has all its roots on the unit circle, its coefficients satisfy the conjugate symmetry constraint:  $b_k = b_{P-k}^*$ ,  $k = 0, 1, \dots, P$ . Similar to EM and SAGE, the IQML algorithm converges to local maxima which may or may not coincide with global optimal solution. The complexity of IQML algorithm is discussed in [85]. The asymptotic performance is investigated in [86]. In [87] an efficient implementation is suggested for the IQML algorithm.

### 3.14.5.3 Subspace fitting methods

The maximum likelihood approach exploits the parametric model and statistical distribution of array observations fully and exhibits excellent performance. The subspace fitting method [88–90] provides a unified framework for the deterministic ML estimator and subspace based methods. More specifically, it uses a nonlinear least square formulation

$$\{\hat{\boldsymbol{\theta}}, \hat{\mathbf{T}}\} = \arg \min_{\boldsymbol{\theta}, \mathbf{T}} \|\mathbf{M} - \mathbf{A}(\boldsymbol{\theta})\mathbf{T}\|^2, \quad (14.87)$$

where  $\mathbf{M}$  is a data matrix and  $\mathbf{T}$  is any matrix of conformable dimension. For fixed  $\mathbf{A}(\boldsymbol{\theta})$ , the minimization with respect with  $\mathbf{T}$  measures how well the column spaces of  $\mathbf{M}$  and  $\mathbf{A}(\boldsymbol{\theta})$  match. Replacing the closed form solution  $\hat{\mathbf{T}} = \mathbf{A}^\dagger(\boldsymbol{\theta})\mathbf{M}$  back into (14.87) results in a concentrated criterion:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \text{tr}(\mathbf{P}^\perp(\boldsymbol{\theta})\mathbf{M}\mathbf{M}^H). \quad (14.88)$$

Clearly, the deterministic ML estimator (14.61) can be obtained by using array observations as the data matrix  $\mathbf{M} = [\mathbf{x}(1), \dots, \mathbf{x}(N)]$ .

The subspace fitting criterion is derived when the estimated signal eigenvectors are inserted as data  $\mathbf{M} = \hat{\mathbf{U}}_s$ . The weighted least square fitting solution to (14.87) has the expression:

$$\hat{\boldsymbol{\theta}}_{\text{SSF}} = \arg \min_{\boldsymbol{\theta}} \text{tr}\left(\mathbf{P}^\perp(\boldsymbol{\theta})\hat{\mathbf{U}}_s \mathbf{W} \hat{\mathbf{U}}_s^H\right), \quad (14.89)$$

where the weighting matrix  $\mathbf{W}$  is Hermitian and positive definite. The analysis in [89, 90] shows that the estimator  $\hat{\boldsymbol{\theta}}_{\text{SSF}}$  is strongly consistent and asymptotically normally distributed. Minimization of the error covariance matrix of  $\hat{\boldsymbol{\theta}}_{\text{SSF}}$  leads to the optimal weighting matrix  $\mathbf{W}_{\text{opt}} = \tilde{\boldsymbol{\Lambda}}^2 \boldsymbol{\Lambda}_s^{-1}$  where  $\tilde{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda}_s - \sigma^2 \mathbf{I}$ . Recall that the diagonal matrix  $\boldsymbol{\Lambda}_s$  contains  $P$  signal eigenvalues and  $\sigma^2$  denotes the noise eigenvalue. In practice, both  $\boldsymbol{\Lambda}_s$  and  $\sigma^2$  are estimated from data. The weighted subspace fitting (WSF) algorithm (or the method of direction estimation (MODE) [91]) is obtained when a consistent estimator  $\hat{\mathbf{W}}_{\text{opt}} = \tilde{\boldsymbol{\Lambda}}^2 \hat{\boldsymbol{\Lambda}}_s^{-1}$  is inserted into (14.89):

$$\hat{\boldsymbol{\theta}}_{\text{WSF}} = \arg \min_{\boldsymbol{\theta}} \text{tr}(\mathbf{P}^\perp(\boldsymbol{\theta})\hat{\mathbf{U}}_s \tilde{\boldsymbol{\Lambda}}^2 \hat{\boldsymbol{\Lambda}}_s^{-1} \hat{\mathbf{U}}_s^H). \quad (14.90)$$

The signal subspace fitting formulation (14.89) has certain advantages over the data-domain nonlinear least square (14.87), in particular when  $P \ll M$ . It is then significantly cheaper to compute (14.89) than (14.87).

In addition to the criterion (14.89), a noise subspace fitting formulation is developed in [91]. Although the resulting criterion is quadratic in the steering matrix  $\mathbf{A}(\boldsymbol{\theta})$ , the noise subspace fitting criterion can not produce reliable estimates in the presence of signal coherence. The covariance matching estimator [92] that will be presented shortly can be formulated as (14.87).

The implementation of nonlinear least square type criteria (14.87) has been addressed in several papers [64, 77, 93]. A common feature of these methods is similar to the SAGE algorithm in the sense that instead of maximizing all parameters simultaneously, a subset of parameters, or the parameters associated with one signal source is computed in one step while keeping other parameters fixed. The RELAX procedure [93] is similar to the SAGE algorithm (14.77) and (14.79), although it has a simpler

motivation and interpretation. The WSF (or MODE) criterion (14.90) can be formulated in terms of polynomial coefficients for ULAs by the expression (14.84) [91]. An iterative implementation, iterative MODE, similar to IQML is suggested in [84]. Theoretical and numerical results in [84] show that iterative MODE provides more accurate estimates and is computationally more efficient than IQML.

### 3.14.5.4 Covariance matching estimation methods

The covariance matching estimation methods are referred to as generalized least squares in the statistical literature. The application to array processing has led to several interesting algorithms [92, 94–97]. Covariance matching can treat temporally correlated data and provides the same large sample properties as maximum likelihood estimation at often a lower computational cost [92].

Recall that the array output covariance matrix is give by  $\mathbf{R}_x = \mathbf{A}(\boldsymbol{\theta})\mathbf{R}_s\mathbf{A}^H(\boldsymbol{\theta}) + \sigma^2\mathbf{I}$ . By stacking the columns of  $\mathbf{R}_x$ , one obtains the following expression:

$$\mathbf{r} = \text{vec}(\mathbf{R}_x) = \Psi(\boldsymbol{\theta})\boldsymbol{\mu} + \Sigma\sigma^2 = [\Psi(\boldsymbol{\theta})\Sigma] \begin{bmatrix} \boldsymbol{\mu} \\ \sigma^2 \end{bmatrix} = \Phi(\boldsymbol{\theta})\boldsymbol{\alpha}, \quad (14.91)$$

where the elements of  $\boldsymbol{\mu}$  are source signal covariance parameters,  $\boldsymbol{\alpha}^T = [\boldsymbol{\mu}^T \sigma^2]$ ,  $\Sigma = \text{vec}(\mathbf{I})$  is a known matrix, and  $\Phi(\boldsymbol{\theta})$  is a given function of the unknown parameter vector  $\boldsymbol{\theta}$ . In general, the DOA parameter vector  $\boldsymbol{\theta}$  enters  $\Psi(\boldsymbol{\theta})$  in a nonlinear fashion. An estimate for  $\mathbf{r}$  can be obtained from the sample covariance matrix by  $\hat{\mathbf{r}} = \text{vec}(\hat{\mathbf{R}}_x)$ . Fitting the data  $\hat{\mathbf{r}}$  to the model (14.91) in the weighted least squares sense leads to the following criterion

$$(\hat{\mathbf{r}} - \mathbf{r})^H \hat{\mathbf{W}}^{-1} (\hat{\mathbf{r}} - \mathbf{r}) = \left\| \hat{\mathbf{W}}^{-\frac{1}{2}} \hat{\mathbf{r}} - \hat{\mathbf{W}}^{-\frac{1}{2}} \Phi(\boldsymbol{\theta}) \boldsymbol{\alpha} \right\|^2, \quad (14.92)$$

where the weighting matrix  $\hat{\mathbf{W}} = \hat{\mathbf{R}}^* \otimes \hat{\mathbf{R}}$  is a consistent estimate of the covariance matrix  $E(\hat{\mathbf{r}} - \mathbf{r})(\hat{\mathbf{r}} - \mathbf{r})^H$ . The symbol  $\otimes$  denotes the Kronecker matrix product.

The least square criterion is separable in the linear and nonlinear parameters. Minimizing (14.92) over the linear parameter vector  $\boldsymbol{\alpha}$  results in a closed form expression:

$$\hat{\boldsymbol{\alpha}} = [\Phi(\boldsymbol{\theta})^H \hat{\mathbf{W}}^{-1} \Phi(\boldsymbol{\theta})]^{-1} \Phi(\boldsymbol{\theta})^H \hat{\mathbf{W}}^{-1} \hat{\mathbf{r}}. \quad (14.93)$$

Substituting (14.93) into (14.92) leads to a concentrated criterion:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \hat{\mathbf{r}}^H \hat{\mathbf{W}}^{-\frac{1}{2}} (\mathbf{I} - \mathbf{P}_{\Phi}(\boldsymbol{\theta})) \hat{\mathbf{W}}^{-\frac{1}{2}} \hat{\mathbf{r}}, \quad (14.94)$$

where  $\mathbf{P}_{\Phi}(\boldsymbol{\theta}) = \hat{\mathbf{W}}^{-\frac{1}{2}} \Phi(\boldsymbol{\theta}) [\Phi(\boldsymbol{\theta})^H \hat{\mathbf{W}}^{-1} \Phi(\boldsymbol{\theta})]^{-1} \Phi(\boldsymbol{\theta})^H \hat{\mathbf{W}}^{-\frac{1}{2}}$  denotes a projection matrix onto the column space of  $\hat{\mathbf{W}}^{-\frac{1}{2}} \Phi(\boldsymbol{\theta})$ . To apply (14.94), one needs to find the matrix  $\Phi(\boldsymbol{\theta})$  corresponding to the covariance matrix of the array observations. Based on the extended invariance principle, it was shown that the covariance matching estimator is a large sample realization of the ML method and asymptotically efficient [92]. A drawback of the covariance matrix matching based estimation methods is that they inherently assume a large sample size, and are less suitable to scenarios involving a small number of observations and high SNRs.

### 3.14.5.5 Performance bound

The performance of an estimator is measured by its average distance to the true parameters. In many cases, one is interested in the following questions: (1) Whether it converges to the true parameter as the number of data samples approaches infinity. (2) Whether the asymptotic error covariance matrix attains the Cramér-Rao bound (CRB). These two properties, consistency and efficiency, and the error covariance matrix are major concerns in a performance study. In this section, we will outline several important results. A comprehensive coverage on performance analysis is given in [Chapter 14](#) in this book.

It is well known that the estimation error covariance of any unbiased estimator is lower bounded by the Cramér-Rao bound [54]. The Crámer-Rao bounds for the conditional and unconditional model are derived in [98–100], respectively. The conditional CRB, denoted by  $\mathbf{B}_c(\boldsymbol{\theta})$  is given by

$$\mathbf{B}_c(\boldsymbol{\theta}) = \frac{\sigma^2}{2N} \left\{ \sum_{n=1}^N \operatorname{Re} \left[ \mathbf{S}^H(n) \mathbf{H}(\boldsymbol{\theta}) \mathbf{S}(n) \right] \right\}^{-1}, \quad (14.95)$$

where  $\mathbf{S}(n) = \operatorname{diag}(s_1(n), \dots, s_p(n))$ ,  $\mathbf{D}(\boldsymbol{\theta}) = [\mathbf{d}(\boldsymbol{\theta}_1), \dots, \mathbf{d}(\boldsymbol{\theta}_M)]$  contains the first derivative of steering vectors,  $\mathbf{d}(\boldsymbol{\theta}) = d\mathbf{a}(\boldsymbol{\theta})/d\boldsymbol{\theta}$ , and  $\mathbf{H}(\boldsymbol{\theta}) = \mathbf{D}^H(\boldsymbol{\theta}) (\mathbf{I} - \mathbf{A}(\boldsymbol{\theta})(\mathbf{A}^H(\boldsymbol{\theta})\mathbf{A}(\boldsymbol{\theta}))^{-1}\mathbf{A}^H(\boldsymbol{\theta})) \mathbf{D}(\boldsymbol{\theta})$ . For  $N \rightarrow \infty$ , the conditional CRB tends to the limit

$$\mathbf{B}_c^{as}(\boldsymbol{\theta}) = \frac{\sigma^2}{2N} \left\{ \operatorname{Re} \left[ \mathbf{H}(\boldsymbol{\theta}) \odot \mathbf{R}_s^T \right] \right\}^{-1}. \quad (14.96)$$

The unconditional CRB, denoted by  $\mathbf{B}_u(\boldsymbol{\theta})$ , is given by

$$\mathbf{B}_u(\boldsymbol{\theta}) = \frac{\sigma^2}{2N} \left\{ \operatorname{Re} \left[ \mathbf{H}(\boldsymbol{\theta}) \odot (\mathbf{R}_s \mathbf{A}(\boldsymbol{\theta})^H \mathbf{R}_x^{-1} \mathbf{A}(\boldsymbol{\theta}) \mathbf{R}_s)^T \right] \right\}^{-1}. \quad (14.97)$$

From (14.95) and (14.97), we can observe that the CRBs decrease with increasing number of snapshots. Theoretical analysis and simulation results also show that the CRBs decrease as the number of sensors or SNRs grow.

#### 3.14.5.5.1 ML methods

The CRBs (14.95) and (14.97) are related as  $\mathbf{B}_u(\boldsymbol{\theta}) \geq \mathbf{B}_c(\boldsymbol{\theta})$  in a positive definite sense. Because the number of signal parameters  $s(n)$  increases with the number of snapshots, the conditional CRB can not be attained by the conditional ML estimator. Under the unconditional data model, the parameter vector remains finite dimensional when  $N \rightarrow \infty$ , the unconditional ML estimator is consistent and achieves the unconditional CRB asymptotically. Let  $\mathbf{C}_c$  and  $\mathbf{C}_u$  denote the covariance matrix of the conditional and unconditional ML estimators, respectively. The following inequality is proved in [100]:  $\mathbf{C}_c \geq \mathbf{C}_u \geq \mathbf{B}_u \geq \mathbf{B}_c$ . In summary, the conditional ML estimator is consistent, but not efficient; whereas the unconditional ML estimator is both consistent and efficient.

#### 3.14.5.5.2 Subspace methods

The subspace methods are derived from signal—and noise eigenvector/eigenvalue estimates. Therefore, statistical properties of the eigen-analysis of  $\widehat{\mathbf{R}}_x$  [8] play an important role in the performance study.

The asymptotic distribution derived in [98] shows that the MUSIC algorithm is a consistent estimator. Its covariance matrix may grow rapidly when some of the signal eigenvalues are close to the noise eigenvalue. This scenario occurs when two signals are closely located or correlated, leading to an almost rank deficient  $\mathbf{A}(\theta)$ . For uncorrelated signals, the MUSIC estimator exhibits good performance comparable with the conditional ML estimator. The asymptotic performance of MUSIC is investigated in [101]. A performance study of root-MUSIC can be found in [36,37]. Asymptotic analysis of ESPRIT is carried out in [90,102].

### 3.14.5.6 Numerical examples

In this section, we compare the performance of conditional ML estimator and root-MUSIC algorithm in a simulated environment. A ULA of 12 elements with half wavelength spacing is employed to receive two far-field narrow band signal sources of equal strengths located at  $\theta = [20^\circ \ 30^\circ]$ . The sample covariance matrix is estimated from  $N = 200$  snapshots. Each experiment performs 500 Monte Carlo trials.

The RMSEs for DOA estimates and the conditional CRB (14.95) for uncorrelated and correlated signals are depicted in Figures 14.8 and 14.9, respectively. For the uncorrelated case, ML performs slightly better than root-MUSIC; in particular, at low SNR between  $-5$  and  $5$  dB. In the correlated case

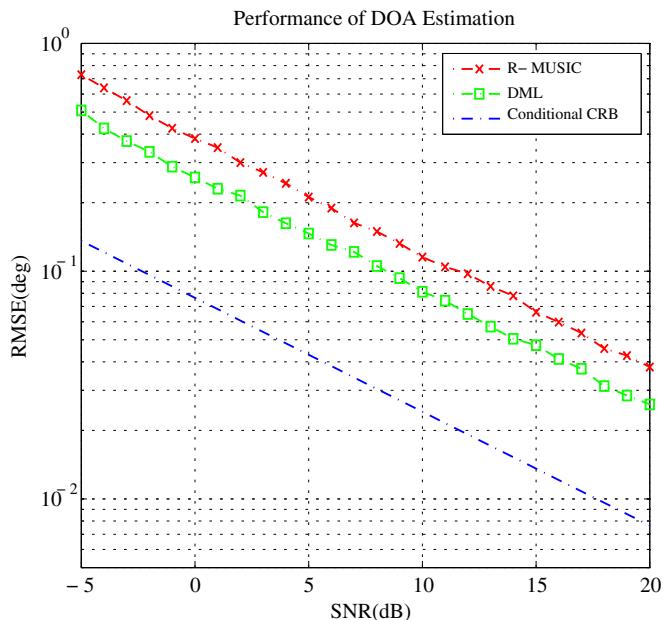
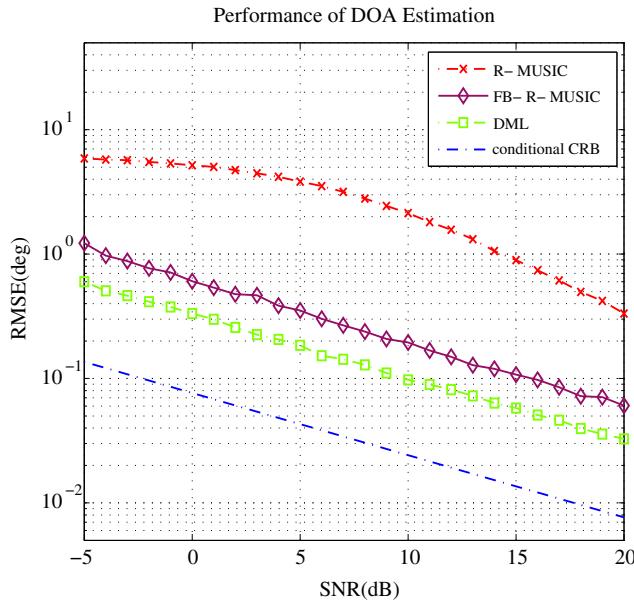


FIGURE 14.8

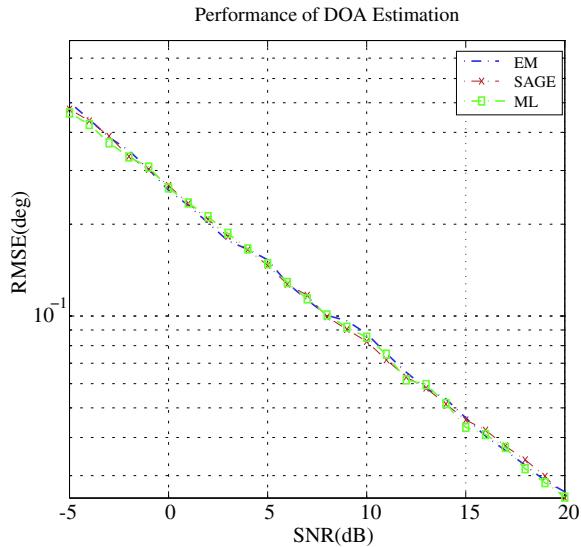
RMSE vs. SNR. Uncorrelated signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ ,  $N = 200$ .

**FIGURE 14.9**

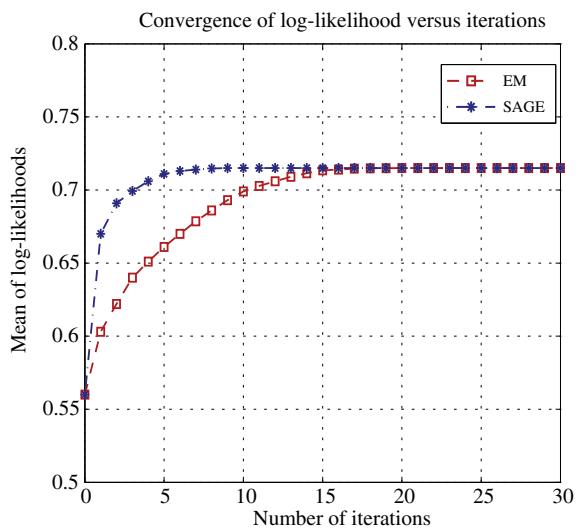
RMSE vs. SNR. Correlated signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ ,  $N = 200$ .

with the correlation coefficient  $\rho = 0.92$ , while the deterministic ML estimator performs as well as in the uncorrelated case and close to the CRB, root-MUSIC no longer provides reliable estimates. As can be observed from Figure 14.9, even for SNR as high as 10 dB, root-MUSIC has RMSE larger than 2°. The difference between ML and root-MUSIC is most significant at low SNRs. In other words, the ML estimator is more robust than root-MUSIC against signal correlation and low SNRs. Although the performance of root-MUSIC is improved by forward-backward averaging, the RMSE is still larger than the ML approach. For SNR below 0 dB, it is twice as much as that of ML.

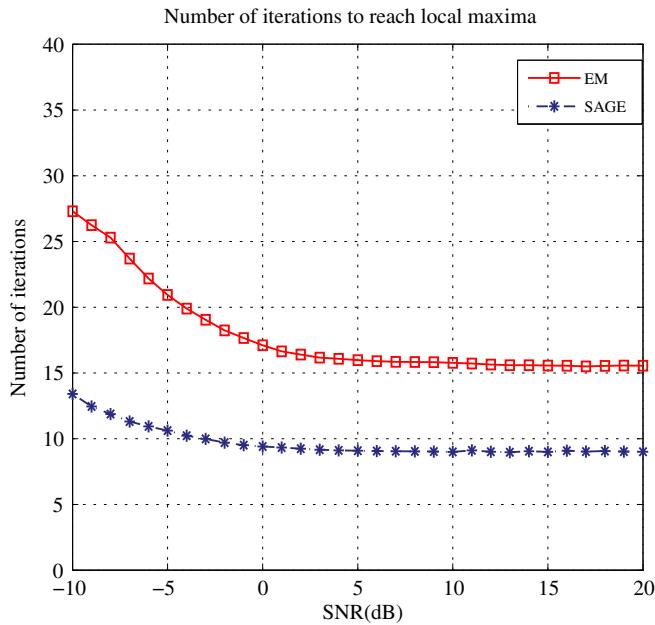
As mentioned previously, the ML estimator requires multi-dimensional nonlinear optimization. We compare three different implementations: (1) matlab function `fmincon`, (2) EM algorithm, (3) SAGE algorithm. The initial estimates for (1) are found by the genetic algorithm. The initial estimates for EM and SAGE are fixed at  $\theta^{(0)} = [16^\circ \ 34^\circ]$  to simplify the investigation of their convergence behavior. Figure 14.10 shows that all three methods find ML estimates and achieve the same accuracy. To compare the convergence behavior of EM and SAGE, we plot the average log-likelihood value vs. iterations in Figure 14.11. As the convergence analysis in [74] predicted, SAGE converges faster than EM. It requires only 6 iterations to reach the final likelihood value, while EM requires 15 iterations. This is further confirmed by the average total number of iterations shown in Figure 14.12. As we can observe, at SNR from -10 dB to 5 dB, EM needs more than twice iterations than SAGE. For moderate to high SNR, the number of iterations of EM always exceeds that of SAGE by at least six iterations. This suggests

**FIGURE 14.10**

RMSE vs. SNR. EM, SAGE, ML with Newton methods. Uncorrelated signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ , SNR = [00] dB,  $N = 200$ .

**FIGURE 14.11**

Log-likelihood vs. iterations. Uncorrelated signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ , SNR = [0 0] dB,  $N = 200$ .

**FIGURE 14.12**

Number of iterations vs. SNR. Uncorrelated signals. Reference DOA parameter  $\theta = [20^\circ \ 30^\circ]$ ,  $N = 200$ .

that EM requires 60% more computational time than SAGE. It should be mentioned that the genetic algorithm [67] requires more than 10 times as long computational time compared to SAGE.

### 3.14.6 Wideband DOA estimation

The DOA estimation methods presented previously are suitable for narrow band signals that often occur in communications and radar systems. In applications like sonar or seismic monitoring, the signals are often broadband. The important issue for the wideband case is how to combine multiple frequencies in an optimal way. For the maximum likelihood approach, the extension is straightforward because of the asymptotic properties of the Fourier transform [3, 76, 103]. Since the signal subspace is different for various frequencies, subspace methods require a pre-processing procedure to form a coherent averaging of the signal subspace as suggested in [104, 105]. Another approach is to evaluate the spectral matrix and derive estimates from each frequency and then combine these estimates in some appropriate way [106]. Numerical results show that the former coherent approach provides better estimates than the latter one. In the following, we will describe the wideband version of the ML estimators and then discuss the coherent signal subspace method.

### 3.14.6.1 Wideband maximum likelihood estimation

In Section 3.14.2.2.1, the data model (14.10) was developed for a continuous temporal array output  $\mathbf{x}(t)$ . In practice, the array outputs are temporally sampled at a properly chosen frequency. The data within an observation interval  $\mathbf{x}(1), \dots, \mathbf{x}(N)$  is divided into  $K$  non-overlapping snapshots of length  $N' = N/K$  and Fourier-transformed. For large number of samples, the frequency domain data  $X^k(\omega)$  can be modeled by (14.10). Under some regularity conditions including stationarity of array outputs, the following asymptotic properties hold [107]:

1.  $X^1(\omega), \dots, X^K(\omega)$  are independent, identically complex normally distributed with zero mean and covariance matrix  $\mathbf{R}_X(\omega) = \mathbf{A}(\omega, \theta)\mathbf{R}_s(\omega)\mathbf{A}(\omega, \theta)^H + \mathbf{R}_n(\omega)$ .
2. For  $0 < \omega_1, \dots, < \omega_J < \pi$ ,  $X^k(\omega_1), \dots, X^k(\omega_J)$  are stochastically independent.
3. Given the signal vector  $\mathbf{S}^k(\omega) = [S_1^k(\omega), \dots, S_p^k(\omega)]^T$ ,  $X^k(\omega)$  is complex normally distributed with mean  $\mathbf{A}(\omega, \theta)\mathbf{S}^k(\omega)$  and covariance matrix  $\mathbf{R}_n(\omega)$ .

In the following, we assume that  $\mathbf{R}_n(\omega) = \sigma^2(\omega)\mathbf{I}$ . Because of the independency between different frequency bins, the likelihood function is a product of those associated with various frequencies. For the deterministic data model, the log-likelihood function is given by

$$\begin{aligned} \mathcal{L}_{w,d}(\theta, \mathbf{S}_{w,d}, \sigma_{w,d}^2) = -\sum_{j=1}^J \sum_{k=1}^K & \left[ N \log \pi + N \log \sigma^2(\omega_j) \right. \\ & \left. + \frac{1}{\sigma^2(\omega_j)} \|X^k(\omega_j) - \mathbf{A}(\omega_j, \theta)\mathbf{S}^k(\omega_j)\|^2 \right], \end{aligned} \quad (14.98)$$

where the signal vector  $\mathbf{S}_{w,d}$  and noise vector  $\sigma_{w,d}^2$  contain signal and noise parameters of  $J$  frequencies. Similar to the narrow band case, the likelihood function can be concentrated with respect to the signal and noise parameters, leading to the concentrated likelihood function

$$L_{w,d}(\theta) = -\sum_{j=1}^J \log \text{tr} \left[ \mathbf{P}^\perp(\omega_j, \theta) \widehat{\mathbf{R}}_x(\omega_j) \right], \quad (14.99)$$

where the sample covariance matrix  $\widehat{\mathbf{R}}_x(\omega_j) = \frac{1}{K} \sum_{k=1}^K X^k(\omega_j) X^k(\omega_k)^H$  and  $\mathbf{P}^\perp(\omega_j, \theta)$  is the orthogonal complement of the projection matrix  $\mathbf{P}(\omega_j, \theta) = \mathbf{A}(\omega_j, \theta)\mathbf{A}^\#(\omega_j, \theta)$ . Note that the summand in (14.99) has the same form as the narrow band likelihood function (14.61). The broadband criterion can be considered as an arithmetic average of narrow band likelihood functions over frequencies. The deterministic ML estimate is computed by maximizing this criterion or minimizing the negative log-likelihood:

$$\hat{\theta}_{W,\text{DML}} = \arg \min_{\theta} -L_{w,d}(\theta). \quad (14.100)$$

For the stochastic signal model, the log-likelihood function is given by

$$\mathcal{L}_{w,s}(\theta, \mathbf{S}_{w,s}, \sigma_{w,d}) = -K \sum_{j=1}^J \left[ M \log \pi + \log \det \mathbf{R}_x(\omega_j) + \text{tr}(\mathbf{R}_x(\omega_j)^{-1} \widehat{\mathbf{R}}_x(\omega_j)) \right], \quad (14.101)$$

where  $\mathbf{S}_{w,s}$  contains signal spectral parameters of all frequencies and  $\sigma_{w,d}$  includes noise power parameters. Similar to the narrow band case, the stochastic likelihood function can be simplified by substituting ML estimates of signal and noise parameters into (14.101), leading to the concentrated criterion:

$$L_{w,s}(\boldsymbol{\theta}) = - \sum_{j=1}^J \log \det \left[ \mathbf{P}(\omega_j, \boldsymbol{\theta}) \widehat{\mathbf{R}}_x(\omega_j) \mathbf{P}(\omega_j, \boldsymbol{\theta}) + \hat{\sigma}^2(\omega_j) \mathbf{P}^\perp(\omega_j, \boldsymbol{\theta}) \right]. \quad (14.102)$$

where  $\hat{\sigma}^2(\omega_j) = \frac{1}{N-M} \text{tr}(\mathbf{P}^\perp(\omega_j, \boldsymbol{\theta}) \widehat{\mathbf{R}}_x(\omega_j))$  is an estimate for the noise parameter. Similar to the deterministic case, (14.102) is an average of the narrow band likelihood criterion (14.61) over frequencies. The stochastic ML estimator is then obtained by maximizing this criterion or minimizing the negative likelihood:

$$\hat{\boldsymbol{\theta}}_{W,SML} = \arg \min_{\boldsymbol{\theta}} -L_{w,s}(\boldsymbol{\theta}). \quad (14.103)$$

The performance analysis in [66] shows that under regularity conditions,  $\hat{\boldsymbol{\theta}}_{W,SML}$  is an asymptotically consistent, efficient estimator for  $\boldsymbol{\theta}$ . The deterministic ML estimator  $\hat{\boldsymbol{\theta}}_{W,DML}$  is asymptotically consistent, but not efficient. The computation complexity can be also simplified by the EM or EM-like algorithms [74, 76].

### 3.14.6.2 Coherent signal subspace methods

The coherent signal subspace methods proposed in [105] combine the broadband data by multiplying each frequency with the *focusing matrix*  $\mathbf{T}(\omega_j)$  satisfying the following property

$$\mathbf{T}(\omega_j) \mathbf{A}(\omega_j, \boldsymbol{\theta}) = \mathbf{A}(\omega_0, \boldsymbol{\theta}), \quad (14.104)$$

where  $\omega_0$  is a selected reference frequency. The coherently averaged covariance matrix  $\mathbf{R}_y$  is given by

$$\mathbf{R}_y = \sum_{j=1}^J \mathbf{T}(\omega_j) \mathbf{R}_x(\omega_j) \mathbf{T}(\omega_j)^H = \sum_{j=1}^J \mathbf{T}(\omega_j) \mathbf{R}_s(\omega_j) \mathbf{T}(\omega_j)^H + \tilde{\sigma}^2 \widetilde{\mathbf{R}}_n, \quad (14.105)$$

where  $\tilde{\sigma}^2$  is the sum of noise level over frequencies and  $\widetilde{\mathbf{R}}_n = \sum_{j=1}^J \frac{\sigma^2(\omega_j)}{\tilde{\sigma}^2} \mathbf{T}(\omega_j) \mathbf{T}(\omega_j)^H$ . Given the known noise structure  $\widetilde{\mathbf{R}}_n$ , the eigenvalue/eigenvector pairs of  $\mathbf{R}_y$  satisfy the same properties (14.36), (14.37) as in the narrow band case. Hence, the subspace methods introduced previously can be applied to the new matrix (14.105).

The design of (14.104) requires a rough estimate of the DOA parameter, which can be obtained from the narrow band MVDR or conventional beamformer. In [104], the rotational signal subspace focusing matrix is developed by solving the constrained optimization problem:

$$\min_{\mathbf{T}(\omega_j)} \| \mathbf{A}(\omega_0, \boldsymbol{\theta}) - \mathbf{T}(\omega_j) \mathbf{A}(\omega_j, \boldsymbol{\theta}) \|_F, \quad j = 1, \dots, J \quad (14.106)$$

$$\text{subject to } \mathbf{T}(\omega_j)^H \mathbf{T}(\omega_j) = \mathbf{I}. \quad (14.107)$$

The solution to (14.106) is given by  $\mathbf{T}(\omega_j) = \mathbf{V}(\omega_j)\mathbf{U}(\omega_j)^H$ , where the columns of  $\mathbf{V}(\omega_j)$  and  $\mathbf{U}(\omega_j)$  are left and right singular vectors of  $\mathbf{A}(\omega_0, \theta)\mathbf{A}(\omega_j, \theta)^H$ . The selection of reference frequency and accuracy improvement are discussed in detail in [104, 105]. Within the class of signal subspace transformation matrices, the rotational subspace transformation matrix is well known for their optimality in preserving SNR after focusing [108]. In practice, the spatial correlation matrix  $\mathbf{R}_x$  is replaced by the sample covariance matrix  $\widehat{\mathbf{R}}_x$ . By (14.105), we compute the coherently averaged sample covariance matrix  $\widehat{\mathbf{R}}_y$  and construct coherent signal/noise subspaces from  $\widehat{\mathbf{R}}_y$ . Then standard subspace methods are applied with  $\omega_0$  as the reference frequency for DOA estimation.

In [109], coherent subspace averaging is achieved by using weighted signal subspaces  $\mathbf{U}_s(\omega_i)\mathbf{P}_i\mathbf{P}_i^H\mathbf{U}_s(\omega_i)^H$  instead of the array correlation matrix  $\mathbf{R}_x(\omega_j)$  in (14.105), where  $\mathbf{U}_s(\omega_i)$  contains signal eigenvectors at frequency  $\omega_i$  and  $\mathbf{P}_i$  is a weighting matrix. While the aforementioned methods concentrates on the design of the *best* coherently averaged signal subspaces and finding the DOA estimates by standard narrow band eigenstructure based algorithms such as MUSIC, the test of orthogonality of projected subspaces (TOPS) algorithm proposed in [110] utilizes the property of transformed signal subspaces and test whether the hypothesized subspaces and the noise subspaces are orthogonal. A significant advantage of this approach is that it does not require initial DOA estimates. Simulation results in [110] show that TOPS performs better than the aforementioned methods in mid SNR ranges, while the coherent methods work best at low SNR and incoherent methods work best at high SNR.

### 3.14.7 Signal detection

Estimation of the number of signals is fundamental to array processing algorithms. It is usually the first step in the application of direction finding algorithms. In the previous discussion, we assumed that the number of signals,  $P$ , is known *a priori*. In practice, the number of signals needs to be estimated from measurements as well. Popular approaches for determining the number of signals can be classified as nonparametric or parametric methods. The former utilizes the eigenstructure of the array correlation matrix (14.13) and estimates the dimension of the signal subspace by employing the information theoretic criteria [10, 111–113] or hypothesis tests [63, 114, 115]. Parametric methods exploit the array output model (14.12) and jointly estimate the parameter and number of signals [65, 116–118]. The nonparametric approach is computationally simple but sensitive to signal coherence and small data samples. The parametric approach requires more time for parameter estimation, but performs significantly better than the nonparametric one in critical scenarios. In the following, we will describe the ideas behind the aforementioned methods briefly and give more related references.

#### 3.14.7.1 Nonparametric methods

We have learned in subspace methods that the array correlation matrix (14.13) has the eigen-decomposition  $\mathbf{R}_x = \mathbf{U}_s\Lambda_s\mathbf{U}_s^H + \mathbf{U}_n\Lambda_n\mathbf{U}_n^H$  where the diagonal matrix  $\Lambda_s$  consists of the  $P$  largest eigenvalues and  $\mathbf{U}_s$  contains corresponding eigenvectors. The remaining  $M - P$  eigenvalues/vectors are included in  $\Lambda_n$  and  $\mathbf{U}_n$  in a similar way. When the signal covariance matrix  $\mathbf{R}_s$  is full rank, the

eigenvalues satisfy the property

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_P > \lambda_{P+1} = \cdots = \lambda_M = \sigma^2. \quad (14.108)$$

The smallest  $(M - P)$  noise eigenvalues are equal to  $\sigma^2$ . This observation suggests that the number of signals can be determined from the multiplicity of the smallest eigenvalue. In practice, the correlation matrix  $\mathbf{R}_x$  is unknown and the eigenvalues are estimated from the sample covariance matrix  $\widehat{\mathbf{R}}_x$ . The ordered sample eigenvalues  $\widehat{\lambda}_i$ ,  $i = 1, \dots, M$  are distinct with probability one [8].

The sphericity test, originating from the statistical literature [119], is modified for detection purpose in [115]. Therein, a series of nested hypothesis tests is formulated to test the equality of  $(M - i)$ ,  $i = 0, \dots, M - 1$  eigenvalues, i.e.,

$$\begin{aligned} H_i : \lambda_1 &\geq \lambda_2 \geq \cdots \geq \lambda_i > \lambda_{i+1} = \cdots = \lambda_M, \\ A_i : \lambda_1 &\geq \lambda_2 \geq \cdots \geq \lambda_i > \lambda_{i+1} > \lambda_M, \end{aligned} \quad (14.109)$$

where  $H_i$  and  $A_i$  denote the null hypothesis and alternative, respectively. Starting from  $i = 0$ , the test proceeds to the next hypothesis if  $H_i$  is rejected. Upon acceptance of  $H_i$ , the test stops, implying all remaining tests are true and leading to the estimate  $\widehat{P} = i$ . For non-Gaussian distribution and small samples case, the test statistic does not have a closed form expression for null distribution. To overcome this problem, a procedure using bootstrap techniques are developed in [114]. The test for unknown noise fields is suggested in [120]. Recently, due to the development of random matrix theory, the eigenvalue based multiple test is re-visited for the small sample case in [121, 122] and the references therein.

The information theoretic criteria based approach views signal detection as model order selection. The Akaike's information criterion (AIC) and Rissanen's minimum description length (MDL) was derived for signal detection in [111]. In the derivation, the data set  $\{\mathbf{x}(1), \dots, \mathbf{x}(N)\}$  is parameterized by the eigenvalues and eigenvectors of correlation matrix  $\mathbf{R}_x$ , rather than the DOA parameter. Maximization of the log-likelihood function leads to the AIC criterion

$$\text{AIC}(i) = -N \log \left[ \frac{\prod_{l=i+1}^M \widehat{\lambda}_l}{\left( \frac{1}{M-i} \sum_{l=i+1}^M \widehat{\lambda}_l \right)^{M-i}} \right] + i(2M - i), \quad (14.110)$$

and the MDL criterion

$$\text{MDL}(i) = -N \log \left[ \frac{\prod_{l=i+1}^M \widehat{\lambda}_l}{\left( \frac{1}{M-i} \sum_{l=i+1}^M \widehat{\lambda}_l \right)^{M-i}} \right] + \frac{1}{2} i(2M - i) \log N. \quad (14.111)$$

The number of signals  $\widehat{P}$  is determined as the minimizing value  $i \in \{0, 1, \dots, P_{\max}\}$  of AIC or MDL, where  $P_{\max}$  ( $\leq M - 1$ ) denotes the maximal number of signals. Eqs. (14.110) and (14.111) show that both criteria has the first term in common, which is a ratio between geometric mean and arithmetic mean of the smallest  $(M - i)$  eigenvalues. The penalty term of AIC depends only on the number of free adjustable parameters, while the penalty term of MDL depends also on the data length  $N$ . In general, AIC tends to overestimate the number of signals while MDL is consistent as the number of

samples approaches infinity [111,123]. Various improvement strategies for MDL for situations involving fully correlated signals, correlated noise and small samples have been suggested in [112,124–127] and references therein.

### 3.14.7.2 Parametric methods

In parametric methods, the DOA parameter in the model (14.12) enters the algorithms directly. Determination of the number of signals can be formulated as a multiple hypothesis test. In the multiple hypothesis test approach, we consider a series of nested hypotheses.

$$\begin{aligned} H_i : \mathbf{x}(n) &= \mathbf{A}_{i-1}(\boldsymbol{\theta}_{i-1})\mathbf{s}_{i-1}(t) + \mathbf{n}(n) && (\text{data contains at most } (i-1) \text{ signals}), \\ A_i : \mathbf{x}(n) &= \mathbf{A}_i(\boldsymbol{\theta}_i)\mathbf{s}_i(n) + \mathbf{n}(n) && (\text{data contains at least } i \text{ signals}). \end{aligned} \quad (14.112)$$

The subscripts  $(i - 1)$  and  $i$  are used to emphasize the dimension of the steering matrix and the signal vector under the null hypothesis  $H_i$  and the alternative  $A_i$ , respectively. Here, the signal vectors are considered as unknown and deterministic.

In [116,118,128], a sequential test procedure is developed to detect the signal one after another. Starting from noise only case,  $i = 0$ ,  $H_i$  is tested against  $A_i$ . If  $H_i$  is rejected, a new signal is declared as detected and the test procedure proceeds to the next hypothesis  $H_{i+1}$ . It stops when the null hypothesis is accepted, implying that no further signal can be detected. The number of signals is then estimated by the dimension of the signal vector under the accepted null hypothesis. The test statistics are constructed by the generalized likelihood ratio principle. In the narrow band case, it leads to an  $F$ -test similar to that suggested in [129]. For broadband data, a closed form expression for the null distribution is not available. In this case, bootstrap techniques or Edgeworth expansions can be applied to approximate the significance level or test threshold. The global significance level in the sequential test procedure is usually controlled by Bonferroni-type procedures. As each test is conducted at a much lower level, results may be conservative when the size of the problem increases. A more powerful test procedure based on the false discovery rate criterion is developed in [117]. A joint estimation and detection procedure was also investigated in [65].

Simulation shows that the multiple hypothesis approach has superior performance than information theoretic approach [117]. In particular, signal coherence has little impact on the performance. Due to ML estimates required in the test, parametric methods are computationally more expensive than eigenvalue based methods.

### 3.14.7.3 Additional issues

When the estimated number of signals,  $\hat{P}$ , provided by detection algorithms is accurate, the performance of DOA estimation is well studied in the literature. In the low SNR region and small sample case, the number of signals may be over- or underestimated. In the presence of overestimated signals, the true DOA parameters are included in the oversized parameter vector with increased variance for the true parameters. In the case of under estimation, the inaccuracy in signal numbers will lead to bias and increased mean squared errors [130]. Robust algorithms have been suggested in [131,132] to retrieve information when the number of signals is unknown.

### 3.14.8 Special topics

We have presented DOA estimation algorithms assuming far-field propagation and static sources. In practice, these conditions may change and require modification to standard algorithms. In the following, we will discuss several interesting issues and provide related references.

#### 3.14.8.1 Tracking

Localization of moving sources is essential to many applications. In the classical tracking problem, the DOA estimates obtained from array data are considered as input data for track estimation. The main task is to match DOA estimates and to contacts, and many solutions have been developed to solve the data association problem [133, 134]. An alternative approach incorporates target motion into the likelihood function and estimates the DOA parameter, and velocity directly from data [135–137]. In [138, 139], the EM algorithm is suggested to reduce the computational cost for maximizing the likelihood function. To further simplify the implementation, the recursive EM algorithm is developed in [128, 140]. The recursive EM algorithm is a stochastic approximation procedure for finding ML estimates [141]. With a specialized gain matrix derived from the EM algorithm, it has a simple implementation and leads to asymptotic normality and consistency. With a proper formulation, it can also be used for estimating time varying DOA parameters.

For subspace methods, how to compute the time-varying signal or noise subspaces efficiently becomes the most important step. In early works, classical batch Eigenvalue Decomposition/SVD techniques have been modified for the use in adaptive processing. Fast computation methods based on subspace averaging were proposed in [142–144]. Another class of algorithms considers ED/SVD as a constrained or unconstrained optimization problem [145–149]. For example, in [149], it is shown that the signal subspace can be computed by solving the following unconstrained optimization problem:

$$\min E \left[ \| \mathbf{x}(n) - \mathbf{W} \mathbf{W}^H \mathbf{x}(n) \|^2 \right], \quad (14.113)$$

where  $\mathbf{W} \in \mathbb{C}^{M \times r}$  denotes the matrix argument and  $r$  is the number of signal eigenvectors. The aim of subspace tracking is to compute  $\mathbf{W}$  efficiently at the time instant  $n$  from the subspace estimate at time instant  $(n - 1)$ . For time-varying subspaces, the expectation in (14.113) is replaced by an exponentially weighted sum of snapshots to ensure the samples in the distant past is down weighted. In addition to low computational complexity, fast convergence and numerical stability are also desired properties in the implementation of subspace tracking techniques. Once the subspace estimates are updated, the DOA estimates are computed by subspace methods presented previously.

#### 3.14.8.2 Signals with known structures

In some applications, the source signals exhibit specific structures and can be exploited for DOA estimation. For example, in communication systems, modulated signals are characterized by *cyclostationarity*, which is referred to as being periodically correlated. This property allows estimation of DOAs of only those signals having specified cycle frequency. Also, the noise can have unknown spatial characteristics as long as it is cyclically independent from signals of interest. Several direction finding algorithms were suggested and analyzed in [150–154].

In standard array processing methods, array data are assumed to be Gaussian and completely characterized by second order statistics. In the presence of non-Gaussian signals, *higher order statistics* can be exploited to the advantage of Gaussian noise suppression and increased aperture [155–158]. A common feature of both types of methods is the requirement of large amount of data samples to achieve comparable results as standard algorithms.

### 3.14.8.3 Spatially correlated noise fields

Most existing array processing methods assume that the background noise is spatially white, i.e., the covariance matrix is proportional to the identity matrix. This assumption is often violated in practical situations [159]. If the noise covariance matrix is known or estimated from signal-free measurements, the data can be pre-whitened. In the absence of this knowledge, the quality of DOA estimates degrade dramatically at low SNR [160,161].

Methods that take small errors in the assumed noise covariance into account are proposed in [63, 162, 163]. These algorithms are not applicable when the noise covariance is completely unknown, unless SNR is very high. Another approach considers parametric noise models and estimates DOA and noise parameters simultaneously [56, 164, 165]. In this approach, the additional noise parameters require extra computational time and increase variances of DOA estimates. In [166–168], the effect of unknown correlated noise is alleviated by the covariance differencing technique. The instrumental variables based approach is proposed in [169, 170]. This technique relies on the assumption that the emitter signals are temporally correlated with correlation time significantly longer than that of the noise. Other solutions include exploitation of prior knowledge of signals [171] or specific array configurations [172]. In [171], the signal waveform is expressed as a linear combination of known basis functions. This assumption is reasonable in applications like radar and active sonar. The algorithm developed therein is a good example showing that knowledge of the spatially colored noise can be traded against alternative *a priori* information about signals.

### 3.14.8.4 Beamspace processing

The computational complexity for DOA estimation grows rapidly with data dimension, i.e., the number of sensors. In many applications like radar, arrays may have thousands of elements [4, 173]. Methods for reducing the data dimension without loss of information are important to these scenarios. Motivated by the idea of beamforming, the *beamspace processing* employs a linear transformation to the outputs of the full sensors of the array

$$\mathbf{z}(n) = \mathbf{T}^H \mathbf{x}(n), \quad (14.114)$$

where  $\mathbf{T}$  is an orthonormal  $M \times R$ , ( $R < M$ ) transformation matrix. The columns of  $\mathbf{T}$  correspond to beamformers within a narrow DOA sector. The design of the transformation matrix is achieved by maximizing the average signal-to-noise ratio [174], selection of spatial sector [175] or minimizing error variances [176]. The beamspace processing may improve estimation performance by filtering out interferences outside the sector of interest and relax the assumption of white noise to local whiteness. As indicated in [177, 178], the resolution threshold of the MUSIC algorithm can be lowered in beamspace. With prior knowledge on the true DOA parameter, it is possible to theoretically attain the Crámer-Rao bound by proper choice of the transformation matrix [176].

The adoption of beamspace processing into MUSIC and ESPRIT algorithms is addressed in [179] and references therein. It is interesting to observe that the array response vector becomes  $\mathbf{T}^H \mathbf{a}(\theta)$  after the transformation (14.114). This allows a simpler expression of array manifold vector and facilitates the application of root-MUSIC and ESPRIT in the 2D case [180]. As pointed out in [180], beamspace transformation has a close link to array interpolation. The interpolated array scheme proposed by Friedlander and Weiss [38] employs a linear transformation to map the manifold vectors for an arbitrary array onto ULA-type response vectors. The field of view of the array is divided into  $L$  sectors, defined by  $[\theta_l^{(1)}, \theta_l^{(2)}](l = 1, \dots, L)$ . Then a set of angles is selected for each sector,

$$\Theta_l = [\theta_l^{(1)}, \theta_l^{(1)} + \Delta\theta, \theta_l^{(1)} + 2\Delta\theta, \dots, \theta_l^{(2)}]. \quad (14.115)$$

Let  $\mathbf{A}(\Theta_l)$  and  $\overline{\mathbf{A}(\Theta_l)}$  denote the array response matrix of the real array and the virtual array with desired response, respectively. An interpolation matrix is computed for each sector as the least square solution such that  $\|\mathbf{B}_l \mathbf{A}_l - \overline{\mathbf{A}_l}\|_F$  is minimized. One may use a weighted least square formulation to improve interpolation accuracy. In [181], an MSE design criterion is suggested to reduce DOA estimation bias caused by array interpolation. An interesting observation is the duality between array interpolation and coherent signal space averaging introduced in Section 3.14.6.2. The former designs the mapping matrix based on the spatially sampled frequencies, while the latter based on samples in the temporal frequency domain. Both techniques are useful for increasing applicability of computationally efficient subspace methods.

### 3.14.8.5 Distributed sources

In several array processing applications, such as radio communications, underwater acoustics and radar, physical measurements show that the effects of angle spread should be taken into account in the modeling. This appears in wireless communications, for example, an elevated base station experiences the received signal as distributed in space due to local scattering around the mobile [182, 183]. The array output in distributed source modeling can be expressed as  $\mathbf{x}(n) = \sum_{i=1}^P s_i(n) + \mathbf{n}(t)$ , where  $s_i(n)$  describes the contribution of the  $i$ th signal to the array output. Unlike in the point source modeling where  $s_i(n) = s_i(n)\mathbf{a}(\theta_i)$ , the source energy is spread over some angular volume and is written as [184–186]

$$s_i(n) = \int_{\theta \in \Theta} \tilde{s}_i(\theta, \psi_i, n) \mathbf{a}(\theta) d\theta, \quad (14.116)$$

where  $\tilde{s}_i(\theta, \psi_i, n)$  is the angular signal density of the  $i$ th source,  $\psi_i$  contains location parameters of the  $i$ th source and  $\Theta$  is the angular field of view. Examples for  $\psi_i$  are the two bounds of a uniformly distributed source or mean and standard deviation of source with Gaussian angular distribution. The problem of interest here is to estimate the unknown parameter vector  $\psi_i$ .

For small angular spread, the distributed source modeling (14.116) usually leads to a signal covariance matrix of the form

$$\widetilde{\mathbf{R}}_{s_i} = \mathbf{a}(\theta_0) \mathbf{a}(\theta_0)^H \odot \mathbf{B}, \quad (14.117)$$

where the matrix  $\mathbf{B}$  is a fully occupied matrix with elements depending on the array shape and signal distribution. As a result, the rank of the signal covariance matrix  $\widetilde{\mathbf{R}}_s = \sum_{i=1}^P \widetilde{\mathbf{R}}_{s_i}$  is equal to the

number of sensors  $M$ . This implies that a separation between signal subspace and noise subspace is not possible. To overcome this difficulty, a number of approaches based on subspace methods have been suggested. In [187], the signal subspace is approximated by eigenvectors associated with dominant eigenvalues. In [185, 186], a generalized subspace in Hilbert space is defined to preserve the eigenstructure as in the point source modeling. Based on an approximation of the signal covariance matrix, a root-MUSIC algorithm is derived in [188]. In [189], the property of the inverse of the covariance matrix is exploited to establish the orthogonality between signal and noise subspace. Performance bounds and analysis of subspace methods in the context of distributed sources are considered in [190, 191].

Note that the distributed source modeling also affects the design of the adaptive beamformer as the distortionless constraint (14.27) no longer holds. In [185], a generalized minimum variance beamformer is proposed by considering total distributed energy of the signal. The resolution of [185] is found superior to [186–188] in simulation. While the above mentioned methods are mostly semiparametric, parametric methods based on ML approach and covariance matching techniques are derived in [97] and [94], respectively. Through the application of the extended invariance principle, the high computational complexity often encountered by the parametric approach is significantly lowered in [94].

### 3.14.8.6 Polarization sensitivity

Most direction finding algorithms presented before consider sensor arrays in which the output of each sensor is a scalar response to, for example, acoustic pressure or one component of the electric or magnetic fields. As the array manifold depends only on the direction of arrival, one is able to retrieve the spatial signature of the emitting signals without estimating polarization parameters. Polarization is an important property of electromagnetic waves. In wireless communications, polarization diversity has played a key in antenna design [192]. The transverse components of the electric or magnetic fields are related through polarization parameters. Due to this additional information, the DOA estimation performance can be improved by polarization sensitive antenna arrays. In [193], an extension of MUSIC is suggested for polarization sensitive arrays. The subspace fitting method, ML based approach and ESPRIT algorithm were developed for diversely polarized signals [61, 194–199], respectively. A performance study can be found in [200, 201].

A complete data model for vector sensors that characterizes all six components of the electromagnetic fields is suggested in [202]. Therein, it is shown that in contrast to scalar sensor arrays, DOA estimation is possible using only one single vector sensor. Identifiability and uniqueness issues related to vector sensors have been investigated in [203, 204]. Other interesting applications including seismic localization, acoustics, and biomedical engineering have been discussed in [205–208].

---

### 3.14.9 Discussion

The problem of estimating the direction of arrival using an array of sensors has been discussed in detail in this contribution. Starting with the beamforming approach, we have presented eigenstructure based subspace methods and parametric methods. The spectral analysis based beamforming techniques are essentially spatial filters and requires least computational effort. Subspace methods achieve high resolution and estimation accuracy at an affordable computational cost. Parametric methods exploit the data model fully, and are characterized by excellent statistical properties and robustness in critical scenarios.

at the expense of increased computations. Selection of suitable algorithms depends on the underlying propagation environment, required accuracy and processing speed, available software and hardware. For simplicity, the essential DOA estimators are presented in the context of narrow band data. Techniques for processing broadband data are treated separately. Methods for signal detection are included in a separate section. Despite the richness of theoretical and experimental results in array processing, technological innovation and theoretical advances have shifted the research focus to application specific methods. To reflect this trend, we selected several topics including tracking, structured signals, correlated noise field, beamspace processing, distributed sources, and vector sensors for discussion.

The materials covered in this article are presented in a tutorial style, trying to serve the first exposure and as a tour guide into this exciting area. References listed here are by no means complete, but in the hope to assist interested readers for further study. More specialized aspects of array processing will be treated in detail by other contributing authors of this series. Their works are particularly valuable to fill the gap between this rather introductory review and in-depth knowledge. Finally, we feel extremely grateful to all researchers that have enriched the area of array processing and made this article possible.

## Acknowledgments

The authors would like to thank Prof. Johann F. Böhme and Prof. Jean-Pierre Delmas for their valuable comments and suggestions that significantly improve this paper.

*Relevant Theory:* Statistical Signal Processing and Signal Processing

See [Vol. 1, Chapter 4](#) Random Signals and Stochastic Processes

See [Vol. 1, Chapter 11](#) Parametric Estimation

See this Volume, [Chapter 2](#) Model Order Selection

See this Volume, [Chapter 8](#) Performance Analysis and Bounds

## References

- [1] J. Capon, High resolution frequency wave number spectrum analysis, Proc. IEEE 57 (1969) 1408–1418.
- [2] R.O. Schmidt, Multiple emitter location and signal parameter estimation, IEEE Trans. Antennas Propag. 34 (3) (1986) 276–280.
- [3] J.F. Böhme, Array processing, in: S. Haykin (Ed.), *Advances in Spectrum Analysis and Array Processing*, Prentice Hall, Englewood Cliffs, NJ, 1991, pp. 1–63.
- [4] H. Krim, M. Viberg, Two decades of array signal processing research: the parametric approach, IEEE Signal Process. Mag. 13 (4) (1996) 67–94.
- [5] B.D. Van Veen, K.M. Buckley, Beamforming: a versatile approach to spatial filtering, IEEE Acoust. Speech Signal Process. Mag. (1988) 4–24.
- [6] D.H. Johnson, D.E. Dudgeon, *Array Signal Processing: Concepts and Techniques*, Prentice Hall, 1993.
- [7] H.L. Van Trees, *Optimum Array Processing (Detection, Estimation, and Modulation Theory, Part IV)*, Wiley, New York, 2002.
- [8] T.W. Anderson, *An Introduction to Multivariate Statistical Analysis*, third ed., Wiley, 2003.
- [9] Y. Bresler, A. Macovski, On the number of signals resolvable by a uniform linear array, IEEE Trans. Acoust. Speech Signal Process. ASSP-34 (1986) 1361–1375.

- [10] M. Wax, I. Ziskind, On unique localization of multiple sources by passive sensor arrays, *IEEE Trans. Acoust. Speech Signal Process.* 37 (7) (1989) 996–1000.
- [11] R.T. Lacoss, Data adaptive spectral analysis methods, *Geophysics* 36 (71) 661–675.
- [12] J. Li, P. Stoica, Z. Wang, On robust Capon beamforming and diagonal loading, *IEEE Trans. Signal Process.* 51 (7) (2003) 1702–1715.
- [13] G. Borgiotti, L. Kaplan, Superresolution of uncorrelated interreference sources by using adaptive array techniques, *IEEE Trans. Antennas Propag.* 27 (3) (1979) 842–845.
- [14] M-X Huang, J.J. Shih, R.R. Lee, D.L. Harrington, R.J. Thoma M.P. Weisend, F. Hanlon, K.M. Paulson, T. Li, K. Martin, G.A. Miller, J.M. Canive, Commonalities and differences among vectorized beamformers in electromagnetic source imaging, *Brain Topogr.* 16 (2004) 139–158.
- [15] C. Vaidyanathan, K.M. Buckley, Performance analysis of the MVDR spatial spectrum estimator, *IEEE Trans. Signal Process.* 43 (6) (1995) 1427–1437.
- [16] A. Luthra, A solution to the adaptive nulling problem with a look-direction constraint in the presence of coherent jammers, *IEEE Trans. Antennas Propag.* 34 (5) (1986) 702–710.
- [17] N.L. Owsley, An overview of optimum adaptive control in sonar array processing, in: K.S. Narendra, R.V. Monopoli (Eds.), *Applications of Adaptive Control*, Academic Press, New York, 1980, pp. 131–164.
- [18] V. Reddy, A. Paulraj, T. Kailath, Performance analysis of the optimum beamformer in the presence of correlated sources and its behavior under spatial smoothing, *IEEE Trans. Acoust. Speech Signal Process.* 35 (7) (1987) 927–936.
- [19] T.J. Shan, T. Kailath, Adaptive beamforming for coherent signals and interference, *IEEE Trans. Acoust. Speech Signal Process.* 33 (3) (1985) 527–536.
- [20] C.-J. Tsai, J.-F. Yang, T.-H. Shiu, Performance analyses of beamformers using effective SINR on array parameters, *IEEE Trans. Signal Process.* 43 (1) (1995) 300–303.
- [21] M.D. Zoltowski, On the performance analysis of the MVDR beamformer in the presence of correlated interference, *IEEE Trans. Acoust. Speech Signal Process.* 36 (6) (1988) 945–947.
- [22] C.D. Richmond, Response of sample covariance based MVDR beamformer to imperfect look and inhomogeneities, *IEEE Signal Process. Lett.* 5 (12) (1998) 325–327.
- [23] S.A. Vorobyov, A.B. Gershman, Z.-Q Luo, Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem, *IEEE Trans. Signal Process.* 51 (2) (2003) 313–324.
- [24] J.-J. Fuchs, On the application of the global matched filter to DOA estimation with uniform circular arrays, *IEEE Trans. Signal Process.* 49 (4) (2001) 702–709.
- [25] D. Malioutov, M. Cetin, A.S. Willsky, A sparse signal reconstruction perspective for source localization with sensor arrays, *IEEE Trans. Signal Process.* 53 (8) (2005) 3010–3022.
- [26] J.A. Tropp, Just relax: convex programming methods for identifying sparse signals in noise, *IEEE Trans. Info. Theory* 52 (3) (2006) 1030–1051.
- [27] M.M. Hyder, K. Mahata, Direction-of-arrival estimation using a mixed  $l_2$  norm approximation, *IEEE Trans. Signal Process.* 58 (9) (2010) 4646–4655.
- [28] J. Yin, T. Chen, Direction-of-arrival estimation using a sparse representation of array covariance vectors, *IEEE Trans. Signal Process.* 59 (9) (2011) 4489–4493.
- [29] A. Panahi, M. Viberg, On the resolution of the Lasso-based Doa estimation method, in: International ITG Workshop on Smart Antennas (WSA 2011), Aachen, Germany, IEEE, 2011.
- [30] V.F. Pisarenko, The retrieval of harmonics from a covariance function, *Geophys. J. Roy. Astron. Soc.* 33 (1973) 347–366.
- [31] G. Bienvenu, L. Kopp, Principle de la goniometrie passive adaptive, in: Proceedings of the 7'eme Colloque GRESTIT, Nice, France, 1979, pp. 106/1–106/10.

- [32] R.O. Schmidt, Multiple emitter location and signal parameter estimation, in: Proceedings of the RADC Spectrum Estimation Workshop, Rome, NY, 1979, pp. 243–258.
- [33] S.K. Oh, C.K. Un, A sequential estimation approach for performance improvement of eigen-structure based methods in array processing, *IEEE Trans. Signal Process.* 1 (41) (1993) 457–463.
- [34] P. Stoica, P. Handel, A. Nehorai, Improved sequential MUSIC, *IEEE Trans. Aerosp. Electron. Syst.* 31 (4) (1995) 1230–1239.
- [35] S.Y. Kung, K.S. Arun, D.V. Bhaskar Rao, State-space and singular-value decomposition-based approximation methods for the harmonic retrieval problem, *J. Opt. Soc. Am.* 73 (12) (1983) 1799–1811.
- [36] A.J. Barabell, Improving the resolution performance of eigenstructure-based direction-finding algorithms, in: Proceedings of the ICASSP 83, Boston, MA, 1983, pp. 336–339.
- [37] B.D. Rao, K.V.S. Hari, Performance analysis of root-MUSIC, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-37 (12) (1989) 1939–1949.
- [38] B. Friedlander, A.J. Weiss, Direction finding using spatial smoothing with interpolated arrays, *IEEE Trans. Aerosp. Electron. Syst.* 28 (2) (1992) 574–587.
- [39] D.V. Sidorovich, A.B. Gershman, Two-dimensional wideband interpolated root-MUSIC applied to measured seismic data, *IEEE Trans. Signal Process.* 46 (8) (1998) 2263–2267.
- [40] R. Kumaresan, D.W. Tufts, Estimating the angles of arrival of multiple plane waves, *IEEE Trans. Aerosp. Electron. Syst.* AES-19 (1983) 134–139.
- [41] S.S. Reddi, Multiple source location—a digital approach, *IEEE Trans. Aerospace Electron. Syst.* 15 (1979) 95–105.
- [42] M. Kaveh, A.J. Barabell, The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-34 (1986) 331–341.
- [43] R. Roy, T. Kailath, ESPRIT—Estimation of signal parameters via rotational invariance techniques, *IEEE Trans. Acoust. Speech Signal Process.* 37 (7) (1989) 984–995.
- [44] M. Haardt, J. Nossek, Unitary ESPRIT: how to obtain increased estimation accuracy with a reduced computational burden, *IEEE Trans. Signal Process.* 43 (5) (1995) 1232–1242.
- [45] B.D. Rao, K.V.S. Hari, Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-37 (12) (1989) 1990–1995.
- [46] G.H. Golub, C.F. Van Loan, *Matrix Computations*, third ed., John Hopkins University Press, Baltimore, 1996.
- [47] J.E. Evans, J.R. Johnson, D.F. Sun, Application of advanced signal processing techniques to angle of arrival estimation in ATC navigation and surveillance systems, Technical Report, MIT Lincoln Laboratory, June 1982.
- [48] S.U. Pillai, B.H. Kwon, Forward/backward spatial smoothing techniques for coherent signal identification, *IEEE Trans. Acoust. Speech Signal Process.* 37 (1) (1989) 8–15.
- [49] T.J. Shan, M. Wax, T. Kailath, On spatial smoothing for direction-of-arrival estimation of coherent signals, *IEEE Trans. Acoust. Speech Signal Process.* 33 (1985) 806–811.
- [50] R.T. Willimas, S. Prasad, A.K. Mahalanabis, L.H. Sibul, An improved spatial smoothing technique for bearing estimation in a multipath environment, *IEEE Trans. Acoust. Speech Signal Process.* 36 (4) (1988) 425–431.
- [51] J. Fuhl, J.P. Rossi, E. Bonek, High resolution 3-D direction of arrival determination for urban mobile radio, *IEEE Trans. Antennas Propag.* 45 (4) (1997) 672.
- [52] M. Haardt, M.D. Zoltowski, C.P. Mathews, J.A. Nossek 2-D unitary ESPRIT for efficient 2-D parameter estimation, in: Proceedings of the ICASSP, Detroit, vol. 4, IEEE, May 1995, pp. 2096–2099.
- [53] M.D. Zoltowski, C.P. Mathews, M. Haardt, Closed-form 2D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT, *IEEE Trans. Signal Process.* 44 (2) (1996) 316–328.
- [54] E.L. Lehmann, G. Casella, *Theory of Point Estimation*, second ed., Springer, New York, 1998.

- [55] J.F. Böhme, Estimation of source parameters by maximum likelihood and nonlinear regression, in: Proceedings of the ICASSP 84, vol. 9, 1984, pp. 271–274.
- [56] J.F. Böhme, Estimation of spectral parameters of correlated signals in wavefields, *Signal Process.* 11 (1986) 329–337.
- [57] J.F. Böhme, Separated estimation of wave parameters and spectral parameters by maximum likelihood, in: Proceedings of the ICASSP 86, Tokyo, Japan, 1986, pp. 2818–2822.
- [58] Y. Bresler, Maximum likelihood estimation of linearly structured covariance with application to antenna array processing, in: Proceedings of the 4th ASSP Workshop on Spectrum Estimation and Modeling, Minneapolis, MN, August 1988, pp. 172–175.
- [59] Y. Bresler, V.U. Reddy, T. Kailath, Optimum beamforming for coherent signal and interferences, *IEEE Trans. Acoust. Speech Signal Process.* 36 (6) (1988) 833–843.
- [60] M. Cedervall, R.L. Moses, Efficient maximum likelihood doa estimation for signals with known waveforms in the presence of multipath, *IEEE Trans. Signal Process.* 45 (3) (1997) 808–811.
- [61] J. Li, R.T. Compton, Maximum likelihood angle estimation for signals with known waveforms, *IEEE Trans. Signal Process.* 41 (1993) 2850–2862.
- [62] J. Li, B. Halder, P. Stoica, M. Viberg, Computationally efficient angle estimation for signals with known waveforms, *IEEE Trans. Signal Process.* 43 (1995) 2154–2163.
- [63] K.M. Wong, J.P. Reilly, Q. Wu, S. Qiao, Estimation of the directions of arrival of signals in unknown correlated noise, Parts i and ii, *IEEE Trans. Signal Process.* 40 (1992) 2007–2028.
- [64] I. Ziskind, M. Wax, Maximum likelihood localization of multiple sources by alternating projection, *IEEE Trans. Acoust. Speech Signal Process.* 36 (10) (1988) 1553–1560.
- [65] B. Ottersten, M. Viberg, P. Stoica, A. Nehorai, Exact and large sample ML techniques for parameter estimation and detection in array processing, in: S. Haykin, J. Litva, T.J. Shepherd (Eds.), *Radar Array Processing*, Springer Verlag, Berlin, 1993, pp. 99–151.
- [66] D. Kraus, Approximative Maximum-Likelihood-Schätzung und verwandte Verfahren zur Ortung und Signalschätzung mit Sensorgruppen, Dr.-Ing. Dissertation, Faculty of Electrical Engineering, Ruhr-Universität Bochum, Shaker Verlag, Aachen, 1993.
- [67] D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1988.
- [68] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, *Science* 220 (4598) (1983) 671–680.
- [69] R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan, 1995, pp. 39–43.
- [70] A.P. Dempster, N. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. Roy. Stat. Soc. Ser. B* 39 (1977) 1–38.
- [71] C.R. Rao, *Linear Statistical Inference and its Application*, Wiley, New York, 1973.
- [72] C.F.J. Wu, On the convergence properties of the EM algorithm, *Ann. Stat.* 11 (1983) 95–103.
- [73] M. Feder, E. Weinstein, Parameter estimation of superimposed signals using the EM algorithm, *IEEE Trans. Acoust. Speech Signal Process.* 36 (4) (1988) 477–489.
- [74] P.-J. Chung, J.F. Böhme, Comparative convergence analysis of EM and SAGE algorithms in DOA estimation, *IEEE Trans. Signal Process.* 49 (12) (2001) 2940–2949.
- [75] Michael I. Miller, Daniel R. Fuhrmann, Maximum-Likelihood narrow-band direction finding and the EM algorithm, *IEEE Trans. Acoust. Speech Signal Process.* 38 (9) (1990) 1560–1577.
- [76] D. Kraus, J.F. Böhme, Maximum likelihood location estimation of wideband sources using the em-algorithm, in: Proceedings of the IFAC/ACASP Symposium on Adaptive Systems in Control and Signal Processing, Grenoble, 1992.

- [77] Jeffrey A. Fessler, Alfred O. Hero, Space-alternating generalized expectation-maximization algorithm, *IEEE Trans. Signal Process.* 42 (10) (1994) 2664–2677.
- [78] Nail Cadalli, Orhan Arikan, Wideband maximum likelihood direction finding and signal parameter estimation by using tree-structured EM algorithm, *IEEE Trans. Signal Process.* 47 (1) (1999) 201–206.
- [79] P.-J. Chung, J.F. Böhme, DOA estimation using fast EM and SAGE algorithms, *Signal Process.* 82 (11) (2002) 1753–1762.
- [80] X.L. Meng, D. van Dyk, The EM algorithm—an old folk song sung to the fast tune, *J. Roy. Stat. Soc. Ser. B* 59 (1997) 511–567.
- [81] Y. Bresler, A. Macovski, Exact maximum likelihood parameter estimation of superimposed exponential signals in noise, *IEEE Trans. Acoust. Speech Signal Process.* 34 (5) (1986) 1081–1089.
- [82] R. Kumaresan, L. Scharf, A. Shaw, An algorithm for pole-zero modeling and spectral analysis, *IEEE Trans. Acoust. Speech Signal Process.* 34 (3) (1986) 637–640.
- [83] K. Steiglitz, L. McBride, A technique for identification of linear systems, *IEEE Trans. Autom. Control.* 10 (1965) 461–464.
- [84] J. Li, P. Stoica, Z.-S. Liu, Comparative study of IQML and MODE direction-of-arrival estimators, *IEEE Trans. Signal Process.* 46 (1) (1998) 149–160.
- [85] M.P. Clark, L.L. Scharf, On the complexity of IQML algorithms, *IEEE Trans. Acoust. Speech Signal Process.* 40 (7) (1992) 1811–1813.
- [86] P. Stoica, J. Li, T. Soderstrom, On the inconsistency of IQML, *Signal Process.* 56 (1997) 185–190.
- [87] Y. Hua, The most efficient implementation of IQML algorithm, *IEEE Trans. Signal Process.* 42 (8) (1994) 2203–2204.
- [88] B. Ottersten, M. Viberg, T. Kailath, Analysis of subspace fitting and ML techniques for parameter estimation from sensor array data, *IEEE Trans. Signal Process.* 40 (1992) 590–600.
- [89] M. Viberg, B. Ottersten, Sensor array processing based on subspace fitting, *IEEE Trans. Signal Process.* 39 (5) (1991) 1110–1121.
- [90] M. Viberg, B. Ottersten, T. Kailath, Detection and estimation in sensor arrays using weighted subspace fitting, *IEEE Trans. Signal Process.* 39 (11) (1991) 2436–2449.
- [91] P. Stoica, K. Sharman, Maximum likelihood methods for direction-of-arrival estimation, *IEEE Trans. Acoust. Speech Signal Process. ASSP-38* (1990) 1132–1143.
- [92] B. Ottersten, P. Stoica, R. Roy, Covariance matching estimation techniques for array signal processing applications, *Digital Signal Process.* 8 (3) (1998) 185–210.
- [93] J. Li, D. Zheng, P. Stoica, Angle and waveform estimation via RELAX, *IEEE Trans. Aerospace Electron. Syst.* 33 (3) (1997) 1077–1087.
- [94] O. Besson, P. Stoica, Decoupled estimation of doa and angular spread for a spatially distributed source, *IEEE Trans. Signal Process.* 48 (7) (2000) 1872–1882.
- [95] A.B. Gershman, F.F. Mecklenbräuker, J.F. Böhme, Matrix fitting approach to direction of arrival estimation with imperfect spatial coherence of wavefronts, *IEEE Trans. Signal Process.* 45 (7) (1997) 1894–1899.
- [96] D. Kraus, J.F. Böhme, Asymptotic and empirical results on approximate maximum likelihood and least squares methods for array processing, in: *Proceedings of the ICASSP, Albuquerque, NM, USA, IEEE*, 1990, pp. 2795–2798.
- [97] T. Trump, B. Ottersten, Estimation of nominal direction of arrival and angular spread using an array of sensors, *Signal Process.* 50 (1–2) (1996) 57–70.
- [98] P. Stoica, A. Nehorai, Music, maximum likelihood and Cramér-Rao bound, *IEEE Trans. Acoust. Speech Signal Process. ASSP-37* (1989) 720–741.
- [99] P. Stoica, A. Nehorai, Music, maximum likelihood and Cramér-Rao bound: Further results and comparisons, *IEEE Trans. Acoust. Speech Signal Process. ASSP-38* (1990) 2140–2150.

- [100] P. Stoica, A. Nehorai, Performance study of conditional and unconditional direction-of-arrival estimation, *IEEE Trans. Acoust. Speech Signal Process.* 38 (1990) 1783–1795.
- [101] X.L. Xu, K.M. Buckley, Bias analysis of the MUSIC location estimator, *IEEE Trans. Acoust. Speech Signal Process.* 40 (10) (1992) 2559–2569.
- [102] B. Ottersten, M. Viberg, T. Kailath, Performance analysis of the total least squares ESPRIT algorithm, *IEEE Trans. Signal Process.* SP-39 (1991) 1122–1135.
- [103] D. Kraus, J.F. Böhme, EM dual maximum likelihood estimation for wideband source location, in: Proceedings of the IEEE ICASSP, Minneapolis, 1993.
- [104] H. Hung, M. Kaveh, Focussing matrices for coherent signal-subspace processing, *IEEE Trans. Acoust. Speech Signal Process.* 36 (8) (1988) 1272–1281.
- [105] H. Wang, M. Kaveh, Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wide-band sources, *IEEE Trans. Acoust. Speech Signal Process.* 33 (4) (1985) 823–831.
- [106] G. Su, M. Morf, The signal subspace approach for multiple wideband emitter location, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-31 (6) (1983) 1502–1522.
- [107] D.R. Brillinger, *Time Series: Data Analysis and Theory*, Holden-Day, San Francisco, 1981.
- [108] M.A. Doron, A.J. Weiss, On focusing matrices for wide-band array processing, *IEEE Trans. Signal Process.* 40 (6) (1992) 1295–1302.
- [109] E.D. di Claudio, R. Parisi, WAVES: weighted average of signal subspaces for robust wideband direction finding, *IEEE Trans. Signal Process.* 49 (10) (2001) 2179–2191.
- [110] Y.-S. Yoon, L.M. Kaplan, J.H. McClellan, TOPS: new DOA estimator for wideband signals, *IEEE Trans. Signal Process.* 54 (6) (2006) 1977–1989.
- [111] M. Wax, T. Kailath, Detection of signals by information theoretic criteria, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-33 (2) (1985) 387–392.
- [112] M. Wax, I. Ziskind, Detection of the number of coherent signals by the MDL principle, *IEEE Trans. Acoust. Speech Signal Process.* 37 (8) (1989) 1190–1196.
- [113] L.C. Zhao, P.R. Krishnaiah, Z.D. Bai, On detection of the number of signals in presence of white noise, *J. Multivariate Anal.* 20 (1) (1986) 1–25.
- [114] Ramon F. Brich, Abdelhak M. Zoubir, Per Pelin, Detection of sources using bootstrap techniques, *IEEE Trans. Signal Process.* 50 (2) (2002) 206–215.
- [115] D. Williams, D. Johnson, Using the sphericity test for narrow-band passive arrays, *IEEE Trans. Acoust. Speech Signal Process.* 38 (1990) 2008–2014.
- [116] J.F. Böhme, Statistical array signal processing of measured sonar and seismic data, in: Proceedings of the SPIE 2563 Advanced Signal Processing Algorithms, San Diego, July 1995, pp. 2–20.
- [117] P.-J. Chung, J.F. Böhme, C.F. Mecklenbräuker, A.O. Hero, Detection of the number of signals using the Benjamini-Hochberg procedure, *IEEE Trans. Signal Process.* 55 (6) (2007) 2497–2508.
- [118] D. Maiwald, J.F. Böhme, Multiple testing for seismic data using bootstrap, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Adelaide, vol. VI, 1994, pp. 89–92.
- [119] D.N. Lawley, Tests of significance of the latent roots of the covariance and correlation matrices, *Biometrika* 43 (1956) 128–136.
- [120] Q. Wu, K.M. Wong, Determination of the number of signals in unknown noise environments-PARADE, *IEEE Trans. Signal Process.* 43 (1) (1995) 362–365.
- [121] S. Kritchman, B. Nadler, Non-parametric detection of the number of signals: hypothesis testing and random matrix theory, *IEEE Trans. Signal Process.* 57 (10) (2009) 3930–3941.
- [122] P.O. Perry, P.J. Wolfe, Minimax rank estimation for subspace tracking, *IEEE J. Sel. Top. Signal Process.* 4 (3) (2010) 504–513.
- [123] L.C. Zhao, P.R. Krishnaiah, Z.D. Bai, Remarks on certain criteria for detection of number of signals, *IEEE Trans. Acoust. Speech Signal Process.* 35 (2) (1987) 129–132.

- [124] E. Fishler, H. Messer, Order statistics approach for determining the number of sources using an array of sensors, *IEEE Signal Process. Lett.* 6 (7) (1999) 179–182.
- [125] R.R. Nadakuditi, A. Edelman, Sample eigenvalue based detection of high-dimensional signals in white noise using relatively few samples, *IEEE Trans. Signal Process.* 56 (7) (2008) 2625–2638.
- [126] M. Wong, Q.T. Zou, J.P. Reilly, On information theoretic criterion for determining the number of signals in high resolution array processing, *IEEE Trans. Acoust. Speech Signal Process.* 38 (11) (1990) 1959–1971.
- [127] C. Xu, S. Kay, Source enumeration via the EEF criterion, *IEEE Signal Process. Lett.* 15 (2008) 569–572.
- [128] D. Maiwald, Breitbandverfahren zur Signalentdeckung und -ortung mit Sensorgruppen in Seismik- und Sonaranwendungen, Dr.-Ing. Dissertation, Department of Electrical Engineering, Ruhr-Universität Bochum, Shaker Verlag, Aachen, 1995.
- [129] R.H. Shumway, Replicated time-series regression: an approach to signal estimation and detection, in: D.R. Brillinger, P.R. Krishnaiah (Eds.), *Handbook of Statistics*, vol. 3, Elsevier Science Publishers B.V., 1983, pp. 383–408.
- [130] P.-J. Chung, Stochastic maximum likelihood estimation under misspecified numbers of signals, *IEEE Trans. Signal Process.* 55 (9) (2007) 4726–4731.
- [131] R. Badeau, B. David, G. Richard, A new perturbation analysis for signal enumeration in rotational invariance techniques, *IEEE Trans. Signal Process.* 54 (2) (2006) 450–458.
- [132] P.-J. Chung, M. Viberg, C.F. Mecklenbräuker, Broadband ML estimation under model order uncertainty, *Signal Process.* 90 (5) (2010) 1350–1356.
- [133] Y. Bar-Shalom, X.R. Li, T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, first ed., Wiley, New York, 2001.
- [134] M. Orton, W. Fitzgerald, A bayesian approach to tracking multiple targets using sensor arrays and particle filters, *IEEE Trans. Signal Process.* 50 (2) (2002) 216–223.
- [135] V. Katkovnik, A.B. Gershman, A local polynomial approximation based beamforming for source localization and tracking in nonstationary environments, *IEEE Signal Process. Lett.* 7 (1) (2000) 3–5.
- [136] C.R. Rao, C.R. Sastry, B. Zhou, Tracking the direction of arrival of multiple moving targets, *IEEE Trans. Signal Process.* 42 (5) (1994) 1133–1144.
- [137] Y. Zhou, P.C. Yip, H. Leung, Tracking the direction-of-arrival of multiple moving targets by passive arrays: algorithm, *IEEE Trans. Signal Process.* 47 (10) (1999) 2655–2666.
- [138] L. Frenkel, M. Feder, Recursive expectation and maximization (em) algorithms for time-varying parameters with application to multiple target tracking, *IEEE Trans. Signal Process.* 47 (2) (1999) 306–320.
- [139] R.E. Zarnich, K.L. Bell, H.L. Van Trees, A unified method for measurement and tracking of contacts from an array of sensors, *IEEE Trans. Signal Process.* 49 (12) (2001) 2950–2961.
- [140] P.-J. Chung, J.F. Böhme, A.O. Hero, Tracking of multiple moving sources using recursive EM algorithm, *EURASIP J. Appl. Signal Process.* 2005 (2005) 50–60.
- [141] D.M. Titterington, Recursive parameter estimation using incomplete data, *J. Roy. Stat. Soc. Ser. B* 46 (2) (1984) 257–267.
- [142] R. DeGroat, Noniterative subspace tracking, *IEEE Trans. Signal Process.* 40 (3) (1992) 571–577.
- [143] I. Karasalo, Estimating the covariance matrix by signal subspace averaging, *IEEE Trans. Acoust. Speech Signal Process. ASSP-34* (1) (1986) 8–12.
- [144] S. Ouyang, Y. Hua, Bi-iterative least-square method for subspace tracking, *IEEE Trans. Signal Process.* 53 (8) (2005) 2984–2996.
- [145] K. Abed-Meraim, A. Chkeif, Y. Hua, Fast orthonormal PAST algorithm, *IEEE Signal Process. Lett.* 7 (3) (2000) 60–62.
- [146] R. Badeau, G. Richard, B. David, Fast and stable yast algorithm for principal and minor subspace tracking, *IEEE Trans. Signal Process.* 56 (8) (2008) 3437–3446.

- [147] C.E. Davila, Efficient, high performance, subspace tracking for time-domain data, *IEEE Trans. Signal Process.* 48 (12) (2000) 3307–3315.
- [148] J. Xin, A. Sano, Efficient subspace-based algorithm for adaptive bearing estimation and tracking, *IEEE Trans. Signal Process.* 53 (12) (2005) 4485–4505
- [149] B. Yang, Projection approximation subspace tracking, *IEEE Trans. Signal Process.* 43 (1) (1995) 95–107.
- [150] W.A. Gardner, Simplification of MUSIC and ESPRIT by exploitation of cyclostationarity, *Proc. IEEE* 76 (1988) 845–847.
- [151] S.V. Schell, Performance analysis of the cyclic MUSIC method of direction estimation for cyclostationary signals, *IEEE Trans. Signal Process.* 42 (11) (1994) 3043–3050.
- [152] Q. Wu, K.M. Wong, Blind adaptive beam forming for cyclostationary signals, *IEEE Trans. Signal Process.* 44 (11) (1996) 2757–2767.
- [153] G. Xu, T. Kailath, Direction-of-arrival estimation via exploitation of cyclostationary—a combination of temporal and spatial processing, *IEEE Trans. Signal Process.* 40 (7) (1992) 1775–1786.
- [154] H. Yan, H.H. Fan, Doa estimation for wideband cyclostationary signals under multipath environment, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004, Proceedings (ICASSP '04)*, vol. 2, May 2004, pp. ii-77–ii-80.
- [155] P. Chevalier, L. Albera, A. Ferreol, P. Comon, On the virtual array concept for higher order array processing, *IEEE Trans. Signal Process.* 53 (4) (2005) 1254–1271.
- [156] M.C. Dogan, J.M. Mendel, Cumulant-based blind optimum beamforming, *IEEE Trans. Aerosp. Electron. Syst.* 30 (3) (1994) 722–741.
- [157] M.C. Dogan, J.M. Mendel, Applications of cumulants to array processing. I. Aperture extension and array calibration, *IEEE Trans. Signal Process.* 43 (5) (1995) 1200–1216.
- [158] B. Porat, B. Friedlander, Direction finding algorithms based on high-order statistics, *IEEE Trans. Signal Process.* 39 (9) (1991) 2016–2024.
- [159] B.F. Cron, C.H. Sherman, Spatial correlation functions for various noise models, *J. Acoust. Soc. Am.* 34 (1962) 1732–1736.
- [160] F. Li, R.J. Vaccaro, Performance degradation of DOA estimators due to unknown noise fields, *IEEE Trans. Signal Process.* 40 (3) (1992) 686–690.
- [161] M. Viberg, Sensitivity of parametric direction finding to colored noise fields and undermodeling, *Signal Process.* 34 (2) (1993) 207–222.
- [162] M. Viberg, A.L. Swindlehurst, Analysis of the combined effects of finite samples and model errors on array processing performance, *IEEE Trans. Signal Process.* 42 (1994) 3073–3083.
- [163] M. Wax, Detection and localization of multiple sources in noise with unknown covariance, *IEEE Trans. Signal Process.* 40 (1) (1992) 245–249.
- [164] J.F. Böhme, D. Kraus, On least squares methods for direction of arrival estimation in the presence of unknown noise fields, in: *Proceedings of the ICASSP 88*, New York, NY, 1988, pp. 2833–2836.
- [165] V. Nagesha, S. Kay, Maximum likelihood estimation for array processing in colored noise, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-93*, 1993, vol. 4, April 1993, pp. 240–243.
- [166] A. Paulraj, T. Kailath, Direction-of-arrival estimation by eigenstructure methods with unknown sensor gain and phase, in: *Proceedings of the IEEE ICASSP*, Tampa, FL, March 1985, pp. 17.7.1–17.7.4.
- [167] S. Prasad, R.T. Williams, A.K. Mahalanabis, L.H. Sibul, A transform-based covariance differencing approach for some classes of parameter estimation problems, *IEEE Trans. Acoust. Speech Signal Process.* 36 (5) (1988) 631–641.

- [168] F. Tuteur, Y. Rockah, A new method for detection and estimation using the eigenstructure of the covariance difference, in: Proceedings of the ICASSP 86 Conference, Tokyo, Japan, 1986, pp. 2811–2814.
- [169] R.L. Moses, A.A. Beex, Instrumental variable adaptive array processing, *IEEE Trans. Aerosp. Electron. Syst.* 24 (2) (1988) 192–202.
- [170] M. Viberg, P. Stoica, B. Ottersten, Array processing in correlated noise fields based on instrumental variables and subspace fitting, *IEEE Trans. Signal Process.* 43 (5) (1995) 1187–1199.
- [171] M. Viberg, P. Stoica, B. Ottersten, Maximum likelihood array processing in spatially correlated noise fields using parameterized signals, *IEEE Trans. Signal Process.* 45 (4) (1997) 996–1004.
- [172] S.A. Vorobyov, A.B. Gershman, K.M. Wong, Maximum likelihood direction-of-arrival estimation in unknown noise fields using sparse sensor arrays, *IEEE Trans. Signal Process.* 53 (1) (2005) 34–43.
- [173] X.L. Xu, K. Buckley, An analysis of beam-space source localization, *IEEE Trans. Signal Process.* 41 (1) (1993) 501.
- [174] B.D. Van Veen, B.G. Williams, Dimensionality reduction in high resolution direction of arrival estimation, in: Twenty-Second Asilomar Conference on Signals, Systems and Computers, 1988, vol. 2, 1988, pp. 588–592.
- [175] P. Forster, G. Vezzosi, Application of spheroidal sequences to array processing, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '87, vol. 12, April 1987, pp. 2268–2271.
- [176] S. Anderson, On optimal dimension reduction for sensor, array signal processing, *Signal Process.* 30 (2) (1993) 245–256.
- [177] H.B. Lee, M.S. Wengrovitz, Resolution threshold of beamspace MUSIC for two closely spaced emitters, *IEEE Trans. Acoust. Speech Signal Process.* 38 (9) (1990) 1545–1559.
- [178] X.L. Xu, K. Buckley, A comparison of element and beam space spatial-spectrum estimation for multiple source clusters, in: Proceedings of the ICASSP 90, Albuquerque, NM, April 1990.
- [179] M.D. Zoltowski, G.M. Kautz, S.D. Silverstein, Beamspace root-mUSIC, *IEEE Trans. Signal Process.* 41 (1) (1993) 344.
- [180] C.P. Mathews, M.D. Zoltowski, Eigenstructure techniques for 2-D angle estimation with uniform circular arrays, *IEEE Trans. Signal Process.* 42 (9) (1994) 2395–2407.
- [181] P. Hyberg, M. Jansson, B. Ottersten, Array interpolation and DOA MSE reduction, *IEEE Trans. Signal Process.* 53 (12) (2005) 4464–4471.
- [182] K.I. Pedersen, P.E. Mogensen, B.H. Fleury, A stochastic model of the temporal and azimuthal dispersion seen at the base station in outdoor propagation environments, *IEEE Trans. Veh. Technol.* 49 (2) (2000) 437–447.
- [183] M. Tapio, On the use of beamforming for estimation of spatially distributed signals, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003 (ICASSP '03), vol. 5, April 2003, pp. V-369–V-372.
- [184] A. Hassanien, S. Shahbazpanahi, A.B. Gershman, A generalized capon estimator for localization of multiple spread sources, *IEEE Trans. Signal Process.* 52 (1) (2004) 280–283.
- [185] S. Shahbazpanahi, S. Valaee, M.H. Bastani, Distributed source localization using ESPRIT algorithm, *IEEE Trans. Signal Process.* 49 (10) (2001) 2169–2178.
- [186] S. Valaee, B. Champagne, P. Kabal, Parametric localization of distributed sources, *IEEE Trans. Signal Process.* 43 (9) (1995) 2144–2153.
- [187] Y. Meng, P. Stoica, K.M. Wong, Estimation of the directions of arrival of spatially dispersed signals in array processing, *IEE Proc.—Radar Sonar Navig.* 143 (1) (1996) 1–9.
- [188] M. Bengtsson, B. Ottersten, Low-complexity estimators for distributed sources, *IEEE Trans. Signal Process.* 48 (8) (2000) 2185–2194.
- [189] A. Zoubir, Y. Wang, P. Charge, Efficient subspace-based estimator for localization of multiple incoherently distributed sources, *IEEE Trans. Signal Process.* 56 (2) (2008) 532–542.

- [190] D. Astely, B. Ottersten, The effects of local scattering on direction of arrival estimation with MUSIC, *IEEE Trans. Signal Process.* 47 (12) (1999) 3220–3234.
- [191] R. Raich, J. Goldberg, H. Messer, Bearing estimation for a distributed source: modeling, inherent accuracy limitations and algorithms, *IEEE Trans. Signal Process.* 48 (2) (2000) 429–441.
- [192] C.B. Dietrich, K. Dietze, J.R. Nealy, W.L. Stutzman, Spatial, polarization, and pattern diversity for wireless handheld terminals, *IEEE Trans. Antennas Propag.* 49 (9) (2001) 1271–1281.
- [193] E. Ferrara Jr., T. Parks, Direction finding with an array of antennas having diverse polarizations, *IEEE Trans. Antennas Propag.* 31 (2) (1983) 231–236.
- [194] J. Li, R.T. Compton, Angle and polarization estimation using ESPRIT with a polarization sensitive array, *IEEE Trans. Antennas Propag.* 39 (9) (1991) 1376–1383.
- [195] J. Li, P. Stoica, Efficient parameter estimation of partially polarized electromagnetic waves, *IEEE Trans. Signal Process.* 42 (11) (1994) 3114–3125.
- [196] D. Rahamim, J. Tabrikian, R. Shavit, Source localization using vector sensor array in a multipath environment, *IEEE Trans. Signal Process.* 52 (11) (2004) 3096–3103.
- [197] A. Swindlehurst, M. Viberg, Subspace fitting with diversely polarized antenna arrays, *IEEE Trans. Antennas Propag.* 41 (12) (1993) 1687–1694.
- [198] I. Ziskind, M. Wax, Maximum likelihood localization of diversely polarized sources by simulated annealing, *IEEE Trans. Antennas Propag.* 38 (7) (1990) 1111–1114.
- [199] M.D. Zoltowski, K.T. Wong, ESPRIT-based 2-D direction finding with a sparse uniform array of electromagnetic vector sensors, *IEEE Trans. Signal Process.* 48 (8) (2000) 2195–2204.
- [200] Q. Cheng, Y. Hua, Performance analysis of the MUSIC and Pencil-MUSIC algorithms for diversely polarized array, *IEEE Trans. Signal Process.* 42 (11) (1994) 3150–3165.
- [201] A.J. Weiss, B. Friedlander, Performance analysis of diversely polarized antenna arrays, *IEEE Trans. Signal Process.* 39 (7) (1991) 1589–1603.
- [202] A. Nehorai, E. Paldi, Vector-sensor array processing for electromagnetic source localization, *IEEE Trans. Signal Process.* 42 (2) (1994) 376–398.
- [203] K.-C. Ho, K.-C. Tan, W. Ser, Investigation on number of signals whose direction of arrival are uniquely determinable with an electromagnetic sensor, *Signal Process.* 47 (1995) 41–54.
- [204] B. Hochwald, A. Nehorai, Identifiability in array processing models with vector-sensor applications, *IEEE Trans. Signal Process.* 44 (1) (1996) 83–95.
- [205] M. Akcakaya, C.H. Muravchik, A. Nehorai, Biologically inspired coupled antenna array for direction-of-arrival estimation, *IEEE Trans. Signal Process.* 59 (10) (2011) 4795–4808.
- [206] D. Donno, A. Nehorai, U. Spagnolini, Seismic velocity and polarization estimation for wavefield separation, *IEEE Trans. Signal Process.* 56 (10) (2008) 4794–4809.
- [207] M. Hawkes, A. Nehorai, Wideband source localization using a distributed acoustic vector-sensor array, *IEEE Trans. Signal Process.* 51 (6) (2003) 1479–1491.
- [208] B. Hochwald, A. Nehorai, Magnetoencephalography with diversely oriented and multicomponent sensors, *IEEE Trans. Biomed. Eng.* 44 (1) (1997) 40–50.

# Subspace Methods and Exploitation of Special Array Structures

# 15

Martin Haardt<sup>\*</sup>, Marius Pesavento<sup>†</sup>, Florian Roemer<sup>\*</sup>, and Mohammed Nabil El Korso<sup>‡</sup>

<sup>\*</sup>Communications Research Laboratory, Ilmenau University of Technology, Ilmenau, Germany

<sup>†</sup>Communication Systems Group, Darmstadt University of Technology, Darmstadt, Germany

<sup>‡</sup>Waves Material and Systems Group, Energetic Mechanic Electromagnetic Lab (LEME, EA-4416), University Paris-Ouest Nanterre-La Defense, Ville d'Avray, France

## 3.15.1 Introduction

Several important applications (including radar, wireless channel sounding, sonar, and seismology) require the estimation of the directions of arrival of several propagating waves from noise-corrupted measurements taken by an array of sensors. Modern subspace based high-resolution frequency or direction of arrival (DOA) estimation schemes [1] provide a resolution that exceeds the traditional Rayleigh resolution limit.<sup>1</sup> They can be classified according to their numerical procedure [2] into

- *extrema-searching techniques*, e.g., spectral MUSIC [3], spectral RARE,
- *polynomial-rooting techniques*, e.g., Pisarenko's harmonic decomposition [4], Min-Norm [5], root-MUSIC [6, 7], MODE, or root-RARE, and
- *matrix-shifting techniques*, e.g., Standard ESPRIT [8], state space methods (direct data approach or Toeplitz approximation method) [9, 10], matrix pencil methods [11–13], optimally weighted ESPRIT [14], or Unitary ESPRIT [15].

Notice that matrix-shifting techniques utilize estimates of the *signal subspace* whereas extrema-searching techniques and most polynomial-rooting techniques use estimates of its orthogonal complement, often referred to as *noise subspace*.

Due to its simplicity and high-resolution capability, *ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques)* [8] has become one of the most popular signal subspace based DOA or spatial frequency estimation schemes. Here, the spatial frequency estimates are obtained without nonlinear optimization and without the computation or search of any spectral measure. ESPRIT is explicitly premised on a point source model for the sources and is restricted to use with array geometries that exhibit so-called invariances [8]. This requirement, however, is not very restrictive as many of the common array geometries used in practice exhibit these invariances, or their output can be transformed to effect these invariances. ESPRIT may be viewed as a complement to the *MUSIC (MUltiple SIgnal*

<sup>1</sup>For a uniform linear array of  $M$  identical sensors, the Rayleigh criterion for resolution states that two incoherent plane waves propagating into two slightly different directions can only be resolved if the difference of their spatial frequencies is at least  $2\pi/M$  [16]. This resolution is, for instance, provided by the DFT-based periodogram.

*Classification*) algorithm [3], the forerunner of all subspace based DOA methods, in that it is based on properties of the signal eigenvectors whereas MUSIC is based on properties of the noise eigenvectors. It should be noted that ESPRIT may also be used in the dual problem of estimating the frequencies of multiple sinusoids embedded in additive noise (harmonic retrieval) [8]. In the latter application, ESPRIT is more generally applicable than MUSIC as it can handle damped sinusoids and provides estimates of the damping factors as well as the constituent frequencies. There are three primary steps in any ESPRIT-type algorithm:

1. *signal subspace estimation*: computation of a basis for the estimated signal subspace,
2. *solution of the invariance equation*: solution of an (in general) overdetermined system of equations, the so-called invariance equation, derived from the basis matrix estimated in step 1, and
3. *spatial frequency estimation*: computation of the eigenvalues of the solution of the invariance equation formed in step 2.

Extensions of these subspace-based high-resolution parameter estimation to the  $R$ -dimensional ( $R$ -D) case are required for a variety of applications, such as estimating the multi-dimensional parameters of the dominant multipath components from MIMO channel measurements [17], which may be used for geometry-based channel modeling. In this case, the dominant multipath components may be parametrized in terms of their azimuth and elevation angles at the transmitter (directions of departure), their azimuth and elevation angles at the receiver (directions of arrival), as well as the corresponding propagation delays and Doppler shifts, leading to an  $R = 6$  dimensional harmonic retrieval problem [17]. Other applications include radar, wireless communications [18], sonar, seismology, and medical imaging. Numerous multi-dimensional harmonic retrieval techniques have been developed, ranging from Fourier-based methods to parametric high resolution techniques, cf. [19] for an overview. Efficient solutions to this problem are given by subspace-based algorithms like ESPRIT- or MUSIC-based techniques [8] and their multi-dimensional extensions such as 2-D Unitary ESPRIT [20],  $R$ -D Unitary ESPRIT [21],  $R$ -D MUSIC [22],  $R$ -D MDF (multi-dimensional folding) [23], or  $R$ -D RARE (rank reduction estimator) [24].

In the traditional approaches to subspace-based parameter estimation, the  $R$ -D signals are stored in matrices by means of a stacking operation. Obviously, this representation does not account for the  $R$ -D grid structure inherent in the data. A more natural approach to store and manipulate multi-dimensional data is given by tensors. Tensors have already been used in parallel factor (PARAFAC) analysis techniques to obtain important identifiability results for the multi-dimensional harmonic retrieval problem [25, 26]. Parameter estimates based on the PARAFAC model are often obtained via iterative techniques such as alternating Least Squares (ALS) [27] that might require many iterations and do not guarantee convergence to the global optimum [28]. Therefore, it has been proposed to use ESPRIT-type methods to initialize these iterative techniques [29]. In contrast to existing tensor approaches using PARAFAC [29], we focus on a direct analogy to the matrix case by using higher-order extensions of the SVD, i.e., the higher-order SVD (HOSVD) [30], and their low-rank approximations [31]. The HOSVD can be viewed as a Tucker3 model [32], which has a long history in tensor analysis [27, 33, 34]. Note that the COMFAC algorithm [35, 36], which is a fast implementation of trilinear ALS, also uses a low-rank approximation based on the Tucker3 model as a preprocessing step. This “Tucker3 compression” is used to speed up the iterative Least Squares fitting procedure of the PARAFAC model (without the Vandermonde structure that is specific to the harmonic retrieval problem), and thereby avoids a brute force implementation of ALS in the raw data space.

In this chapter, we show that the tensor representation allows us to exploit the structure inherent in the data further. We demonstrate how existing concepts like forward-backward averaging [37] and the mapping of complex centro-Hermitian covariance matrices to real-valued matrices of the same size [37,38] can be generalized to tensors. We also discuss how an HOSVD-based low-rank approximation leads to an improved estimate of the signal subspace which can be used to improve any multi-dimensional subspace-based parameter estimation scheme, e.g., *R*-D Unitary ESPRIT, *R*-D MUSIC, or *R*-D RARE. As examples, we derive the *R*-D standard Tensor-ESPRIT and the *R*-D Unitary Tensor-ESPRIT algorithms explicitly.

Table 15.1 summarizes the ESPRIT-type algorithms that are discussed in this chapter together with a reference where they have first been proposed and a reference where a performance analysis for them has been derived. In this case (open) means that analytical performance results are not available yet. Tensor-ESPRIT-type algorithms are written in *italic* letters. Since all ESPRIT-type algorithms can be combined with different methods to solve the shift invariance equations, Table 15.2 summarizes the different Least Squares solutions that are available together with the corresponding references.

**Table 15.1** Overview of ESPRIT-Type Algorithms and Their Performance Analysis. Tensor-ESPRIT-Type Algorithms are Written in *Italic* Letters

Algorithm	Proposed	Performance Analysis
1-D Standard ESPRIT	[39]	[7, 40], ...
1-D Unitary ESPRIT	[15]	[41]
<i>R</i> -D Standard ESPRIT	(Implicit in [42])	(Implicit in [40])
<i>R</i> -D Unitary ESPRIT	[20] (2-D), [21] ( <i>R</i> -D)	[41] (2-D), [43] ( <i>R</i> -D)
<i>R</i> -D Standard <i>Tensor-ESPRIT</i>	[42]	[43, 44]
<i>R</i> -D Unitary <i>Tensor-ESPRIT</i>	[42]	[43, 44]
1-D NC Standard ESPRIT	[45]	( = 1-D NC Unitary ESPRIT)
1-D NC Unitary ESPRIT	[46]	(open)
<i>R</i> -D NC Standard ESPRIT	(Implicit in [46])	(= <i>R</i> -D NC Unitary ESPRIT)
<i>R</i> -D NC Unitary ESPRIT	[46]	(open)
<i>R</i> -D NC Standard <i>Tensor-ESPRIT</i>	[47]	(open)
<i>R</i> -D NC Unitary <i>Tensor-ESPRIT</i>	[47]	(= <i>R</i> -D NC Unitary <i>Tensor-ESPRIT</i> )

**Table 15.2** Overview of Least-Squares Algorithms to Solve the Invariance Equations of ESPRIT-Type Algorithms and Their Performance Analysis

Algorithm	Proposed	Performance Analysis
Least Squares (LS)	[39]	[7, 40]
Total Least Squares (TLS)	[48]	[49]
Structured Least Squares (SLS)	[50]	[51] (1-D)
Tensor-Structure SLS	[52]	(open)

This chapter is organized as follows. After the introduction of the data model in Section 3.15.2, we discuss matrix- and tensor-based subspace estimation techniques in Section 3.15.3. Based on these subspace estimation techniques, we provide an overview of important subspace-based parameter estimation techniques in Section 3.15.4, before the main conclusions are summarized in Section 3.15.5.

## 3.15.2 Data model

### 3.15.2.1 Notation

In order to facilitate the distinction between scalars, matrices, and tensors, the following notation is used: Scalars are denoted as italic letters ( $a, b, \dots, A, B, \dots, \alpha, \beta, \dots$ ), column vectors as lower-case bold-face letters ( $\mathbf{a}, \mathbf{b}, \dots$ ), matrices as bold-face capitals ( $\mathbf{A}, \mathbf{B}, \dots$ ), and tensors are written as bold-face calligraphic letters ( $\mathcal{A}, \mathcal{B}, \dots$ ). Lower-order parts are consistently named: the  $(i, j)$ -element of the matrix  $\mathbf{A}$ , is denoted as  $a_{i,j}$  and the  $(i, j, k)$ -element of a third order tensor  $\mathcal{B}$  as  $b_{i,j,k}$ .

We use the superscripts  $T, H, *, -1, +$  for transposition, Hermitian transposition, complex conjugation, matrix inversion, and the Moore-Penrose pseudo inverse of a matrix, respectively. Moreover, the Kronecker product of two matrices  $\mathbf{A}$  and  $\mathbf{B}$  is denoted as  $\mathbf{A} \otimes \mathbf{B}$  and the Khatri-Rao product (column-wise Kronecker product) as  $\mathbf{A} \diamond \mathbf{B}$ . An  $n$ -mode vector of an  $(I_1 \times I_2 \times \dots \times I_N)$ -dimensional tensor  $\mathcal{A}$  is an  $I_n$ -dimensional vector obtained from  $\mathcal{A}$  by varying the index  $i_n$  and keeping the other indices fixed. A subtensor of the tensor  $\mathcal{A}$ , denoted by  $\mathcal{A}_{i_n=k}$ , is obtained by fixing the  $n$ th index to some value  $k$ . Moreover, a matrix unfolding of the tensor  $\mathcal{A}$  along the  $n$ th mode is denoted by  $[\mathcal{A}]_{(n)}$  and can be understood as a matrix containing all the  $n$ -mode vectors of the tensor  $\mathcal{A}$ . The order of the columns is chosen in accordance with [53].

The outer product of the tensors  $\mathcal{A} \in \mathcal{C}^{I_1 \times I_2 \times \dots \times I_N}$  and  $\mathcal{B} \in \mathcal{C}^{J_1 \times J_2 \times \dots \times J_M}$  is given by

$$\begin{aligned} \mathcal{C} &= \mathcal{A} \circ \mathcal{B} \in \mathcal{C}^{I_1 \times \dots \times I_N \times J_1 \times \dots \times J_M}, \quad \text{where} \\ c_{i_1, i_2, \dots, i_N, j_1, j_2, \dots, j_M} &= a_{i_1, i_2, \dots, i_N} \cdot b_{j_1, j_2, \dots, j_M}. \end{aligned} \tag{15.1}$$

In other words, the tensor  $\mathcal{C}$  contains all possible combinations of pairwise products between the elements of  $\mathcal{A}$  and  $\mathcal{B}$ . This operator is very closely related to the Kronecker product defined for matrices.

The  $n$ -mode product of a tensor  $\mathcal{A} \in \mathcal{C}^{I_1 \times I_2 \times \dots \times I_N}$  and a matrix  $\mathbf{U} \in \mathcal{C}^{J_n \times I_n}$  along the  $n$ th mode is denoted as  $\mathcal{B} = \mathcal{A} \times_n \mathbf{U}$  and defined via

$$\mathcal{B} = \mathcal{A} \times_n \mathbf{U} \Leftrightarrow [\mathcal{B}]_{(n)} = \mathbf{U} \cdot [\mathcal{A}]_{(n)}, \tag{15.2}$$

i.e., it may be visualized by multiplying all  $n$ -mode vectors of  $\mathcal{A}$  from the left-hand side by the matrix  $\mathbf{U}$ .

The higher-order SVD (HOSVD) of a tensor  $\mathcal{A} \in \mathcal{C}^{I_1 \times I_2 \times \dots \times I_N}$  is given by

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N, \tag{15.3}$$

where  $\mathcal{S} \in \mathcal{C}^{I_1 \times I_2 \times \dots \times I_N}$  is the core tensor which satisfies the all-orthogonality conditions [53] and  $\mathbf{U}_n \in \mathcal{C}^{I_n \times I_n}$ ,  $n = 1, 2, \dots, N$ , are the unitary matrices of  $n$ -mode singular vectors.

We also define the concatenation of two tensors along the  $n$ th mode via the operator  $[\mathcal{A} \sqcup_n \mathcal{B}]$ .

The operations we have defined so far satisfy the following properties that can easily be verified

$$\mathcal{A} \times_1 X_1 \times_2 X_2 = \mathcal{A} \times_2 X_2 \times_1 X_1, \quad (15.4)$$

$$(\mathcal{A} \times_1 X_1) \times_1 Y_1 = \mathcal{A} \times_1 (Y_1 \cdot X_1), \quad (15.5)$$

$$[\mathcal{A} \times_1 X_1 \times_2 X_2 \cdots \times_R X_R]_{(n)} = X_n \cdot [\mathcal{A}]_{(n)} \cdot (X_{n+1} \otimes X_{n+2} \cdots \otimes X_R \\ \otimes X_1 \cdots \otimes X_{n-1})^T, \quad (15.6)$$

$$[\mathcal{I}_{R,N} \times_1 F_1 \cdots \times_R F_R]_{(p)} = F_p \cdot (F_{p+1} \diamond \cdots \diamond F_R \diamond F_1 \diamond \cdots \diamond F_{p-1})^T, \quad (15.7)$$

$$[\mathcal{A} \sqcup_r \mathcal{B}] \times_p U_p = [\mathcal{A} \times_p U_p \sqcup_r \mathcal{B} \times_p U_p], \quad \text{where } r \neq p, \quad (15.8)$$

$$[\mathcal{A} \sqcup_r \mathcal{B}] \times_r [U_r, W_r] = \mathcal{A} \times_r U_r + \mathcal{B} \times_r W_r, \quad (15.9)$$

$$\mathcal{A} \times_r \begin{bmatrix} X_r \\ Y_r \end{bmatrix} = [(\mathcal{A} \times_r X_r) \sqcup_r (\mathcal{A} \times_r Y_r)], \quad (15.10)$$

where  $r, p \in \{1, 2, \dots, R\}$  and the dimensions of the tensors and matrices are  $\mathcal{A}, \mathcal{B} \in \mathcal{C}^{M_1 \times \cdots \times M_R}$ ,  $U_r, W_r \in \mathcal{C}^{N_r \times M_r}$ ,  $V_r \in \mathcal{C}^{P_r \times N_r}$ ,  $X_r \in \mathcal{C}^{N_r \times M_r}$ , and  $Y_r \in \mathcal{C}^{Q_r \times M_r}$ . The Euclidean (vector) norm, the Frobenius (matrix) norm, and the Higher-Order Frobenius (tensor) norm are denoted by  $\|a\|_2$ ,  $\|A\|_F$ , and  $\|\mathcal{A}\|_H$ , respectively. All three norms are computed by taking the square-root of the sum of the squared magnitude of all the elements in their arguments. It is easily verified that

$$\|\mathcal{A}\|_H = \|[\mathcal{A}]_{(n)}\|_F, \quad n = 1, 2, \dots, N. \quad (15.11)$$

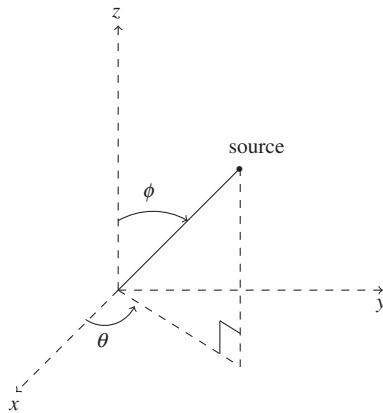
A  $p \times p$  matrix  $Q_p$  is called left- $\Pi$ -real if  $\Pi_p \cdot Q_p^* = Q_p$ , where  $\Pi_p$  is the  $p \times p$  exchange matrix with ones on its antidiagonal and zeros elsewhere. The special set of unitary sparse left- $\Pi$ -real matrices introduced in [15] is denoted as  $Q_p^{(s)}$ . They are given by

$$Q_{2n}^{(s)} = \frac{1}{\sqrt{2}} \begin{bmatrix} I_n & jI_n \\ \Pi_n & -j\Pi_n \end{bmatrix} \quad \text{and} \quad Q_{2n+1}^{(s)} = \frac{1}{\sqrt{2}} \begin{bmatrix} I_n & \theta_{n \times 1} & jI_n \\ \theta_{n \times 1}^T & \sqrt{2} & \theta_{n \times 1}^T \\ \Pi_n & \theta_{n \times 1} & -j\Pi_n \end{bmatrix}, \quad (15.12)$$

for odd and even order, respectively. Furthermore, a matrix  $X \in \mathcal{C}^{M \times N}$  is called centro-Hermitian if  $\Pi_M \cdot X^* \cdot \Pi_N = X$ . The vector  $e_k$  denotes the  $k$ th column of an identity matrix.

### 3.15.2.2 General data model

Consider an array with  $M$  identical omni-directional sensors distributed in the  $R$ -dimensional ( $R$ -D) space with complex gain equal to one (see the introductory chapter for a discussion on beam patterns). The location of each sensor  $m$  (for  $m = 1, \dots, M$ ) is denoted by  $\varrho_m$  in the  $R$ -D space. Depending on the scenario the array geometry could be fully known or could be divided into many smaller known subarrays in the case of partially calibrated arrays. Assume that there are  $d$  ( $d < M$ ) far-field point sources emitting narrow-band signals whose baseband model at time  $t$  is denoted by  $s(t)$ . These sources are assumed to be located at azimuth angles  $\theta_1, \theta_2, \dots, \theta_d$  and at elevation angles  $\phi_1, \phi_2, \dots, \phi_d$  (see Figure 15.1). Estimating these angles, called direction-of-arrivals (DOAs), are the objective of various estimation techniques. We also assume that  $N$  observations or snapshots are available at times  $t_n$ ,

**FIGURE 15.1**

Coordinate system for one source.

i.e.,  $n = 1, 2, \dots, N$ . Throughout the text, the number of the sources is assumed to be known or can be estimated using the well-known methods presented in [54–56]. Moreover, the sources are considered to be uncorrelated. The  $m$ th sensor noise for  $m = 1, 2, \dots, M$  is modeled as independently identically distributed (i.i.d.) zero-mean complex white Gaussian additive noise, i.e.,

$$n_m(t) \sim \mathcal{CN}(0, \sigma^2). \quad (15.13)$$

Furthermore, the noise vector  $\mathbf{n}(t)$  can be written as

$$\mathbf{n}(t) = [n_1(t) \ n_2(t) \ \cdots \ n_M(t)]^T. \quad (15.14)$$

The noise is assumed to be both spatially and temporally white, hence

$$\mathbb{E}\{\mathbf{n}(t_1)\mathbf{n}^H(t_2)\} = \begin{cases} \sigma^2 \mathbf{I}_M, & t_1 = t_2, \\ 0, & t_1 \neq t_2. \end{cases} \quad (15.15)$$

The full observation matrix (also referred to as array output signal) is given by

$$\mathbf{X} = [\mathbf{x}(t_1) \ \mathbf{x}(t_2) \ \cdots \ \mathbf{x}(t_N)] \quad (15.16)$$

in which the  $t$ th snapshot of the array observation vector in the presence of the sensor noise  $\mathbf{n}(t)$  is given by

$$\mathbf{x}(t) = \sum_{l=1}^d \mathbf{a}(\theta_l, \phi_l) s_l(t) + \mathbf{n}(t), \quad (15.17)$$

$$= \mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi}) \mathbf{s}(t) + \mathbf{n}(t), \quad (15.18)$$

where  $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \dots \ \theta_d]^T$  and  $\boldsymbol{\phi} = [\phi_1 \ \phi_2 \ \dots \ \phi_d]^T$  are, respectively, the azimuth and the elevation angles of the source DOAs, the vector  $\mathbf{a}(\theta_l, \phi_l)$  indicates the array response (commonly referred to as array steering vector) to the  $l$ th source,<sup>2</sup> and

$$\mathbf{s}(t) = [s_1(t) \ s_2(t) \ \dots \ s_d(t)]^T \quad (15.19)$$

is the signal waveform vector which is assumed to be stochastic. The matrix  $\mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi})$  can be represented as containing  $d$  column-vectors each corresponding to a source such that

$$\mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi}) = [\mathbf{a}(\theta_1, \phi_1) \ \mathbf{a}(\theta_2, \phi_2) \ \dots \ \mathbf{a}(\theta_d, \phi_d)]. \quad (15.20)$$

Thus, the full observation matrix can be written as

$$\mathbf{X} = \mathbf{X}_0 + \mathbf{N}, \quad (15.21)$$

where  $\mathbf{N} = [\mathbf{n}(t_1) \ \mathbf{n}(t_2) \ \dots \ \mathbf{n}(t_N)]$  and the noiseless observation matrix is given by  $\mathbf{X}_0 = \mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi})\mathbf{S}$  in which the source signal matrix  $\mathbf{S} = [s(t_1) \ s(t_2) \ \dots \ s(t_N)]$ .

Generally, the  $m$ th element of the array steering vector  $[\mathbf{a}(\theta_l, \phi_l)]_{(m)}$ , i.e., the response of sensor  $m$  to the  $l$ th source, can be shown to be [22]

$$[\mathbf{a}_m(\theta_l, \phi_l)]_{(m)} = g_m(\theta_l, \phi_l) \cdot e^{j\mathbf{k}_l^T \mathbf{e}_m}, \quad (15.22)$$

where  $\mathbf{k}_l$  is defined as the wavenumber corresponding to plane wave impinging on the array from the direction of the  $l$ th source such that

$$\mathbf{k}_l = -\frac{2\pi}{\lambda} \begin{bmatrix} \cos \theta_l \sin \phi_l \\ \sin \theta_l \sin \phi_l \\ \cos \phi_l \end{bmatrix}, \quad (15.23)$$

where the azimuth and the elevation angles are defined similar to those in the spherical coordinate system, i.e.,  $\theta$  is the azimuth angle in the  $xy$ -plane from the  $x$ -axis and  $\phi$  is the elevation angle from positive  $z$ -axis (see Figure 15.1). It is customary to assume, without loss of generality, that the first sensor is placed in the origin of the coordinate system. Moreover, the term  $g_m(\theta_l, \phi_l) \in \mathcal{C}$  in (15.22) is the complex beam pattern of the antenna array at the azimuth angle  $\theta_l$  and the elevation angle  $\phi_l$ . In the special case where the elements are assumed to be isotropic we have  $g_m(\theta, \phi) = 1, \forall \theta, \phi$ .

### 3.15.2.3 Special array structures

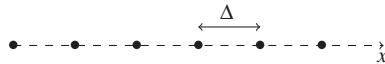
In this part, arrays with special structures are taken into account. The popularity of many of such arrays is due to their array steering matrix structural feature which can be exploited to develop search-free low computational complexity DOA estimation algorithms.

#### 3.15.2.3.1 Uniform linear arrays (ULAs)

In a uniform linear array (ULA), all the  $M$  sensors lie on a line and the distance between the adjacent sensors is identical  $\Delta$  for any two adjacent sensor, see Figure 15.2. Hence, if we assume the first sensor

---

<sup>2</sup>Depending on the situation and for sake of simplicity, the array steering vectors and/or matrices will be indexed by the azimuth  $\theta$  and the elevation  $\phi$ , or by the spatial frequency  $\mu$  given, for example, by (15.28) and (15.29) in the 2-D context using a uniform rectangular array.

**FIGURE 15.2**

Uniform linear array geometry with  $M = 6$  sensors.

to be the reference sensor, for a ULA of size  $M$  it can be said that

$$\varrho_m = (m - 1)\Delta, \quad m = 1, \dots, M. \quad (15.24)$$

The ULAs are unable to distinguish between sources with different elevation angles. Hence, ULAs are incapable of estimating the elevation angles of the source DOA [57], i.e.,  $\phi_1, \dots, \phi_d$ . The  $l$ th element of the array steering vector for ULA lying on the  $x$ -axis can be written as

$$[\mathbf{a}(\theta_l)]_{(m)} = e^{j \frac{2\pi}{\lambda} (m-1)\Delta \cos \theta_l}. \quad (15.25)$$

Later we will see that sometimes it is more useful to write the array steering vector of a ULA as a function of spatial frequency  $\mu$  rather than as a function of DOAs. The spatial frequency associated with the  $l$ th source is defined as

$$\mu_l = \frac{2\pi}{\lambda} \Delta \cos \theta_l. \quad (15.26)$$

Then, the ULA steering matrix can be written as follows:

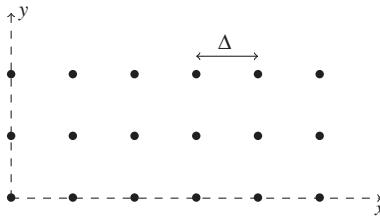
$$\begin{aligned} \mathbf{A}(\boldsymbol{\mu}) &= [\mathbf{a}(\mu_1) \quad \mathbf{a}(\mu_2) \quad \cdots \quad \mathbf{a}(\mu_d)] \\ &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{j\mu_1} & e^{j\mu_2} & \cdots & e^{j\mu_d} \\ e^{j2\mu_1} & e^{j2\mu_2} & \cdots & e^{j2\mu_d} \\ \vdots & \vdots & \vdots & \vdots \\ e^{j(M-1)\mu_1} & e^{j(M-1)\mu_2} & \cdots & e^{j(M-1)\mu_d} \end{bmatrix}, \end{aligned} \quad (15.27)$$

where  $\boldsymbol{\mu} = [\mu_1 \cdots \mu_d]^T$ . As it can be observed the obtained array steering matrix for a ULA is a Vandermonde matrix [7].

### 3.15.2.3.2 Minimum redundancy linear arrays

In [58] Moffet introduced a class of non-uniform spaced linear arrays in order to achieve best DOA estimation performance for a given number of sensors. This class is the so-called minimum redundancy arrays. Let us consider the one dimensional case where the first sensor denotes the reference sensor such that  $\varrho_1 = 0$ . Let  $\Delta_g$  denotes the greatest common divisor of all existing inter-element. The minimum redundancy array are defined such that:

- All intermediate distances are present, i.e., one has  $m\Delta_g \in \mathcal{D}, \forall m \in \{1, 2, \dots, M - 1\}$  in which  $\mathcal{D}$  contains all the existing sensors inter-element.
- Minimize the number of the redundant lags, i.e., pairs of sensors separated by the same distance.

**FIGURE 15.3**

Uniform rectangular array geometry with  $M_1 = 6$  and  $M_2 = 3$  sensors.

### 3.15.2.3.3 Uniform rectangular arrays (URAs)

In a uniform rectangular array (URA), the sensors lie on uniform grid of a rectangular shape where the sensor spacing is equal to  $\Delta$  (see Figure 15.3). We assume that the URA lies on the  $xy$ -plane and it consists of sensors in a grid of size  $M_1 \times M_2$ , hence the total number of sensors is  $M = M_1 M_2$ . It should be remarked that unlike ULAs, URAs are capable of estimating both the azimuth and the elevation angles of the source DOAs. Defining the spatial frequencies associated with azimuth and elevation angles of the  $l$ th source, respectively,

$$\mu_l^{(1)} = \frac{2\pi}{\lambda} \Delta \cos \theta_l \sin \phi_l, \quad (15.28)$$

$$\mu_l^{(2)} = \frac{2\pi}{\lambda} \Delta \sin \theta_l \sin \phi_l, \quad (15.29)$$

it can be shown that the URA steering vector corresponding to the  $l$ th source can be formed as

$$\mathbf{a}(\mu_l^{(1)}, \mu_l^{(2)}) = \mathbf{a}(\mu_l^{(1)}) \otimes \mathbf{a}(\mu_l^{(2)}), \quad (15.30)$$

where  $\mathbf{a}(\mu_l^{(1)})$  and  $\mathbf{a}(\mu_l^{(2)})$  are equivalent to the ULA steering vectors composed of  $M_1$  and  $M_2$  sensors, respectively.

### 3.15.2.3.4 Uniform circular arrays (UCAs)

The uniform circular array (UCA) is a specific class of planar arrays where all sensors lie in a unique circle such that the angular separation,  $\zeta$ , between two successive sensors is the same, see Figure 15.4. The UCA is generally preferred to the ULA due to the ambiguities introduced by the linear arrays [59].

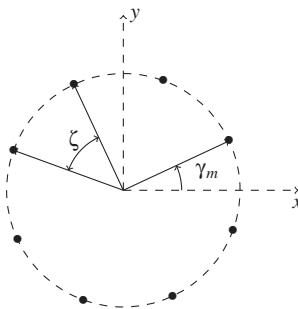
The steering vectors are given as

$$[\mathbf{a}(\theta_l, \phi_l)]_{(m)} = e^{j \frac{2\pi r}{\lambda} \cos(\theta_l - \gamma_m) \sin(\phi_l)}, \quad (15.31)$$

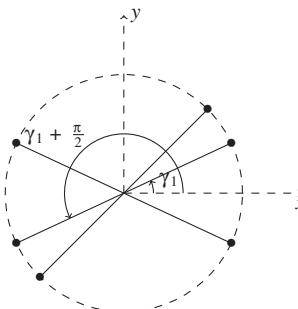
where  $r, \gamma_m$  denote the radius of the array and the angle of the  $m$ th sensor.

### 3.15.2.3.5 Centro-symmetric arrays

An array is called centro-symmetric if it can be mirrored around its centroid without changing its geometry, see, for example, Figure 15.5. Mathematically speaking, its array steering matrix must satisfy

**FIGURE 15.4**

Uniform circular array geometry with  $M = 8$ .

**FIGURE 15.5**

Centro-symmetric circular array geometry with  $M = 6$ .

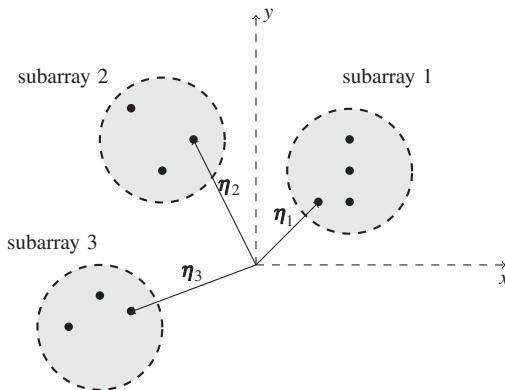
the property

$$\Pi_M \cdot \mathbf{A}^* = \mathbf{A} \cdot \Delta. \quad (15.32)$$

Here,  $\Delta$  is a diagonal matrix containing phase terms which account for the fact that the phase reference does not necessarily coincide with the array's centroid, e.g., ULAs or UCAs. Note that for condition (15.32) to be valid, the array elements do not need to be omnidirectional. We can have an arbitrary complex beam pattern  $g_m(\theta, \phi)$  for the antennas, as long as all elements have identical beam patterns, i.e.,  $g_m(\theta, \phi) = g(\theta, \phi)$  for  $m = 1, 2, \dots, M$ .

### 3.15.2.3.6 Partially calibrated arrays

Although sensor arrays with large aperture size are favorable but these arrays are costly. Moreover, they are more susceptible to modeling errors (such as sensor mutual coupling, channel mismatches between subarrays, sensor gain and phase uncertainties, array geometry uncertainties, and time synchronization issue) and classic subspace-based DOA estimation methods are known to be sensitive to these errors. To avoid the modeling errors in the process of DOA estimation, the idea of large-aperture sparse sensor

**FIGURE 15.6**

3 Arbitrary known subarrays with arbitrary unknown displacements.

arrays is presented. Sparse sensor arrays is divided into  $K$  smaller subarrays that are much easier to calibrate. Hence, these arrays are referred to as partially calibrated arrays (PCAs), see Figure 15.6.

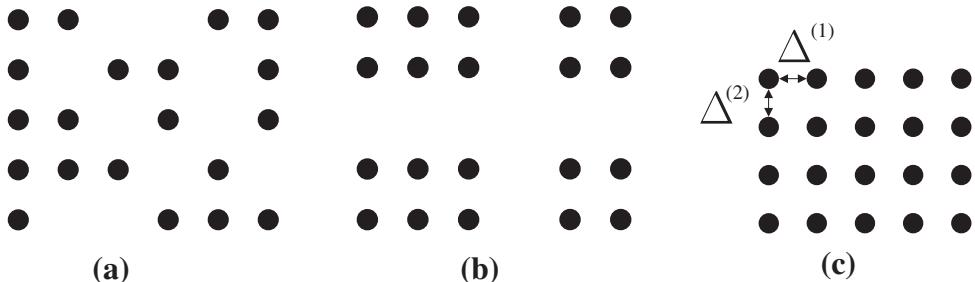
For this class of arrays, the array steering matrix, which is as always a function of DOAs, is also a function of new unknown signal-independent parameters which are the modeling errors. This set of parameters is denoted by  $\eta = [\eta_2 \ \eta_3 \ \dots \ \eta_K]^T$ . For instance,  $\eta$  can indicate the unknown (or uncertain) displacement vectors  $\eta_2, \eta_3, \dots, \eta_K$  between the first (or reference) subarray and the other subarrays, c.f. Figure 15.6. To estimate the source DOAs, in this case, the array steering matrix is normally partitioned such that the part which solely depends on the DOAs is separated from other part(s) that depends on the unknown modeling errors and the DOAs. Hence, the structure of the array steering matrix (or vector) can be exploited to estimate the DOAs by circumventing the modeling errors in the steering matrix. The partitioning of the array steering matrix can be done in various ways for different estimation schemes which will be discussed when those schemes are presented.

### 3.15.2.3.7 Multidimensional arrays

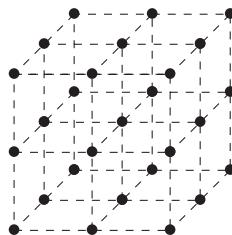
The ULA and URA configurations can be generalized to the  $R$ -D case by considering  $R$ -dimensional sampling grids. In order to be able to apply  $R$ -D algorithms such as  $R$ -D matrix-based or tensor-based ESPRIT, this grid needs to satisfy certain properties.

In particular, an  $R$ -dimensional sampling grid is called *separable* if it can be constructed from an outer product of  $R$  one-dimensional sampling grids. In other words, for each dimension we can design the sampling freely but then all combinations of sampling points must be present in the sampled data. Examples of not separable and separable 2-D sampling grids are shown in Figure 15.7a and b, respectively. Figure 15.7c shows the special case of a 2-D uniform sampling grid which is assumed for URAs.<sup>3</sup>

<sup>3</sup>Hexagonal arrays are another example of non-separable 2-D arrays. Therefore, we cannot apply tensor calculus there, even though the application of 3-D Unitary ESPRIT is possible, as we have shown in [60, 61].

**FIGURE 15.7**

Examples of 2-D sampling grids: (a) not a separable 2-D sampling grid; (b) separable 2-D sampling grid composed of the outer product of two (non-uniform) linear arrays; (c) uniform separable 2-D grid.

**FIGURE 15.8**

3-D array geometry with  $M_1 = M_2 = M_3 = 3$ .

Separable  $R$ -D sampling grids lead to array manifolds that are separable with respect to the  $R$  dimensions. To this end, let  $\mathbf{a}(\mu^{(r)}) \in \mathcal{C}^{M_r \times 1}$  be the array manifold in the  $r$ th dimension, comprising  $M_r$  sampling points, where  $\mu^{(r)}$  is the spatial frequency in the  $r$ th dimension. Then the array manifold of the  $R$ -D array satisfies

$$\mathbf{a}(\mu^{(1)}, \dots, \mu^{(R)}) = \mathbf{a}(\mu^{(1)}) \otimes \dots \otimes \mathbf{a}(\mu^{(R)}) \in \mathcal{C}^{M \times 1}, \quad (15.33)$$

where  $\otimes$  represents the Kronecker product and

$$M = \prod_{r=1}^R M_r. \quad (15.34)$$

In the special case where  $\mathbf{a}(\mu^{(r)})$  is chosen uniformly in all dimensions  $r = 1, 2, \dots, R$ , the  $R$ -D array is referred to as a uniform  $R$ -D sampling grid. Uniform  $R$ -D sampling can be seen as the generalization of ULAs and URAs to  $R$  dimensions. Figure 15.8 exemplifies such a grid for  $R = 3$ . Note that dimensions are not restricted to spatial dimensions. We can also consider time and frequency dimensions, for instance.

Note that for 2-D antenna arrays, the following conditions must be fulfilled so that the array manifold becomes separable:

1. The array elements are placed in a 2-D grid that is separable, i.e., it can be constructed as the outer product of two 1-D grids.
2. The complex beam patterns  $g_m(\theta, \phi)$  are *either*
  - a. expressed via a separable function over the direction cosines  $\mu^{(r)}$  corresponding to the two array dimensions, cf. (15.28) and (15.29), or
  - b. equal for all antenna elements, i.e.,  $g_m(\theta, \phi) = g(\theta, \phi)$  for  $m = 1, 2, \dots, M$  (including their spatial orientation in the array). In the latter case, separability is not needed.

If the 2-D array does not obey the two assumptions, its two dimensions cannot be separated and we have to stack the elements in one mode of our measurement tensor.

The Kronecker-structured array manifold shown in (15.33) can be represented via tensors in a very natural way. Instead of stacking the  $R$  dimensions along of the rows of a long array steering vector  $\mathbf{a}$ , we can preserve the natural  $R$ -D structure by considering an array steering tensor [42]. For the  $i$ th wavefront we can write

$$\mathcal{A}_i = \mathbf{a}(\mu_i^{(1)}) \circ \mathbf{a}(\mu_i^{(2)}) \circ \cdots \circ \mathbf{a}(\mu_i^{(R)}) \in \mathcal{C}^{M_1 \times M_2 \times \cdots \times M_R}, \quad (15.35)$$

where  $\circ$  represents the outer product and  $i = 1, 2, \dots, d$ . Similar to the array steering matrix  $\mathbf{A} \in \mathcal{C}^{M \times d}$  we can define an array steering tensor by concatenating the  $\mathcal{A}_i$  from (15.35) by virtue of the concatenation operator  $\sqcup_h$  (cf. Section 3.15.2.1)

$$\mathcal{A} = [\mathcal{A}_1 \sqcup_{R+1} \mathcal{A}_2 \sqcup_{R+1} \cdots \sqcup_{R+1} \mathcal{A}_d]. \quad (15.36)$$

Likewise, the multidimensional observations can be arranged into a measurement tensor  $\mathcal{X} \in \mathcal{C}^{M_1 \times \cdots \times M_R \times N}$ . Together with (15.36) we may write

$$\mathcal{X} = \mathcal{A} \times_{R+1} \mathbf{S}^T + \mathcal{N} = \mathcal{X}_0 + \mathcal{N}, \quad (15.37)$$

which is the tensor-based equivalent of (15.21). The relations to the matrix-based model are given by

$$\mathbf{X} = [\mathcal{X}]_{(R+1)}^T, \quad \mathbf{A} = [\mathcal{A}]_{(R+1)}^T, \quad \mathbf{N} = [\mathcal{N}]_{(R+1)}^T. \quad (15.38)$$

### 3.15.2.3.8 R-D shift invariance structure

For  $R$ -D matrix shifting-based algorithms we additional require an  $R$ -D shift invariance in the array. This means that the array can be divided into two subarrays that are identical except for a displacement in all  $R$  dimensions. Since for an  $R$ -D harmonic wave, a displacement in the  $r$ th mode incurs a phase offset proportional to the frequency of the wave in the  $r$ th mode, frequency estimates are obtained by estimating the phase offsets for all waves in all  $R$  modes.

To estimate the frequencies of an  $R$ -D harmonic wave in all dimensions<sup>4</sup> in this manner, we require a shift invariance of the sampling grid in all  $R$  dimensions. This can be expressed via tensor calculus in

---

<sup>4</sup>If the signal is harmonic only in  $R'$  of the  $R$  dimensions ( $R' < R$ ), we can apply  $R'$ -D ESPRIT in these modes and leave the others modes untouched. In this case the  $R - R'$  “non-harmonic” dimensions simply provide additional snapshots (which we collect in mode  $R + 1$  in our data model for simplicity).

a natural way. Let  $\mathcal{A}_i \in \mathcal{C}^{M_1 \times M_2 \cdots \times M_R}$  be the “array steering tensor” of the  $i$ th wavefront as defined in (15.35). Then, the shift invariance of  $\mathcal{A}_i$  in the  $r$ th mode can be expressed as

$$(\mathcal{A}_i \times_r \mathbf{J}_1^{(r)}) \cdot e^{j\mu_i^{(r)}} = \mathcal{A}_i \times_r \mathbf{J}_2^{(r)}, \quad r = 1, 2, \dots, R, \quad (15.39)$$

where  $\mathbf{J}_1^{(r)} \text{ and } \mathbf{J}_2^{(r)} \in \mathcal{R}^{M_r^{(\text{sel})} \times M_r}$  are the selection matrices which select the  $M_r^{(\text{sel})}$  out of  $M_r$  indices belonging to the first and the second subarray in the  $r$ th mode, respectively. Moreover,  $\mu_i^{(r)}$  is the spatial frequency of the  $i$ th wavefront in the  $r$ th mode for  $i = 1, 2, \dots, d$  and  $r = 1, 2, \dots, R$ .

For the special case of an  $R$ -D uniform sampling grid introduced above we choose  $\mathbf{J}_1^{(r)}$  and  $\mathbf{J}_2^{(r)}$  to

$$\mathbf{J}_1^{(r)} = [\mathbf{I}_{M_r-1} \ \ \mathbf{0}_{(M_r-1) \times 1}] \quad \mathbf{J}_2^{(r)} = [\mathbf{0}_{(M_r-1) \times 1} \ \ \mathbf{I}_{M_r-1}], \quad (15.40)$$

such that  $M_r^{(\text{sel})} = M_r - 1$ , which corresponds to maximally overlapping subarrays.

The shift invariance relation for a single source from (15.39) can be extended to consider all  $d$  sources jointly. We obtain [42]

$$\mathcal{A} \times_r \mathbf{J}_1^{(r)} \times_{R+1} \Phi^{(r)} = \mathcal{A} \times_r \mathbf{J}_2^{(r)}, \quad (15.41)$$

where  $\mathcal{A} = [\mathcal{A}_1 \sqcup_{R+1} \mathcal{A}_2 \sqcup_{R+1} \cdots \sqcup_{R+1} \mathcal{A}_d] \in \mathcal{C}^{M_1 \times M_2 \cdots \times M_R \times d}$  is the array steering tensor (cf. (15.36)) and

$$\Phi^{(r)} = \text{diag} \left( [e^{j\mu_1^{(r)}}, \dots, e^{j\mu_d^{(r)}}] \right) \in \mathcal{C}^{d \times d}. \quad (15.42)$$

Note that a matrix-based equivalent of (15.41) in terms of the array steering matrix  $\mathbf{A} = [\mathcal{A}]_{(R+1)}^T$  is found by considering the transpose of the  $(R + 1)$ -mode unfolding of (15.41). Using (15.6) we obtain

$$\tilde{\mathbf{J}}_1^{(r)} \cdot \mathbf{A} \cdot \Phi^{(r)} = \tilde{\mathbf{J}}_2^{(r)} \cdot \mathbf{A}, \quad \text{where} \quad (15.43)$$

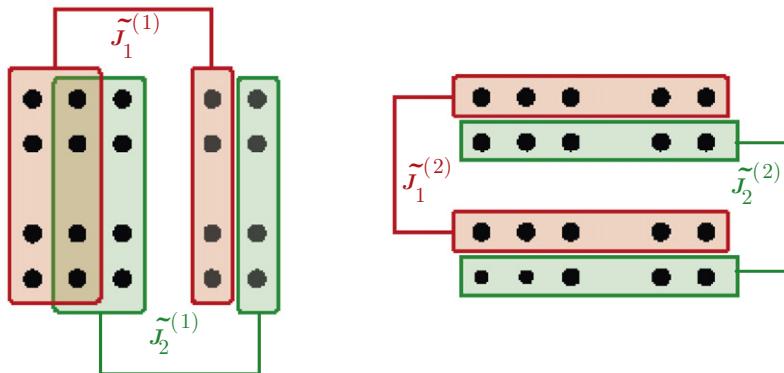
$$\tilde{\mathbf{J}}_n^{(r)} = (\mathbf{I}_{M_1} \otimes \cdots \otimes \mathbf{I}_{M_{r-1}}) \otimes \mathbf{J}_n^{(r)} \otimes (\mathbf{I}_{M_{r+1}} \otimes \cdots \otimes \mathbf{I}_{M_R}), \quad n = 1, 2, \quad (15.44)$$

which coincides with the matrix-based shift invariance equations derived in [21].

Note that in the 1-D case, the shift invariance equation simplifies into

$$\mathbf{J}_1 \cdot \mathbf{A} \cdot \Phi = \mathbf{J}_2 \cdot \mathbf{A}. \quad (15.45)$$

Figure 15.9 shows the 2-D shift invariance of a  $5 \times 4$  separable 2-D sampling grid. The left-hand side shows how to choose the selection matrices  $\mathbf{J}_1^{(1)}$  and  $\mathbf{J}_2^{(1)}$  for the first dimension (horizontal) and the right-hand side shows how to choose the selection matrices  $\mathbf{J}_1^{(2)}$  and  $\mathbf{J}_2^{(2)}$  for the second dimension (vertical), i.e.,

**FIGURE 15.9**

2-D shift invariance for a  $5 \times 4$  separable 2-D sampling grid. Left: subarrays for the first (horizontal) dimension, right: subarrays for the second (vertical) dimension.

$$\begin{aligned} J_1^{(1)} &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, & J_2^{(1)} &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \\ J_1^{(2)} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, & J_2^{(2)} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned}$$

Moreover,  $\tilde{J}_n^{(1)} = J_n^{(1)} \otimes I_4$  and  $\tilde{J}_n^{(2)} = I_5 \otimes J_n^{(2)}$  for  $n = 1, 2$ .

### 3.15.2.4 Non-circular data

Up to here we have not made any further assumptions about the amplitudes of the multidimensional signals which we collect in the matrix  $S$  (cf. (15.21)), except for the fact that the rank of  $S$  should be equal to  $d$ . However, as we discuss in Sections 3.15.4.2 and 3.15.4.3.3, via simple modifications of ESPRIT-type algorithms<sup>5</sup> [46, 47] we can take advantage of a particular structure in these amplitudes referred to as second-order non-circularity. This occurs for instance in communication-type scenarios where the transmitters employ specific modulation schemes, such as binary phase shift keying (BPSK), amplitude shift keying (ASK), minimum shift keying (MSK), or Offset Quadrature Phase Shift Keying (OQPSK).

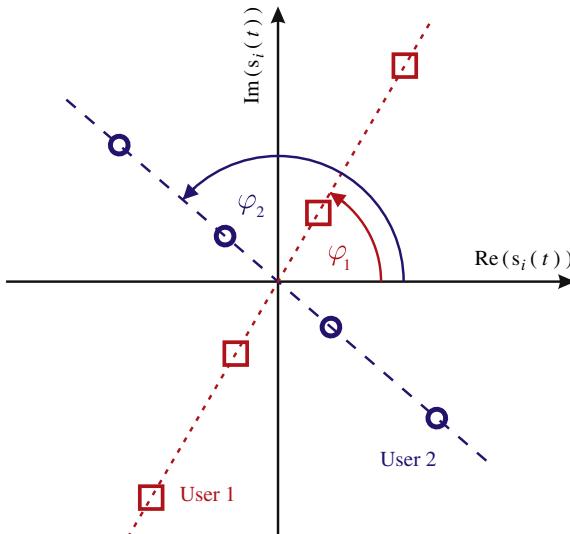
A full statistical description of complex random variables includes not only the individual distribution of their real and imaginary parts but also the joint distribution since they might be correlated. A simplifying assumption that is often made is to consider second-order *circularly symmetric* complex random variables. A zero mean complex random variable  $Z = X + jY$  is said to be second-order circularly symmetric if it satisfies  $E\{Z^2\} = 0$ , which implies that real part and imaginary part are uncorrelated

<sup>5</sup>Many other subspace-based parameter estimation schemes have been enhanced to benefit from non-circular sources as well, e.g., [45, 62–64].

and have the same variance. Consequently, if  $E\{Z^2\} \neq 0$ , the random variable  $Z$  is (second-order) non-circular. We can measure the degree of non-circularity via a scalar parameter  $\zeta$  referred to as the “non-circularity rate” [65], “circularity coefficient” [66], or “circularity quotient” [67],

$$\zeta = \frac{E\{Z^2\}}{E\{|Z|^2\}}. \quad (15.46)$$

It can be shown that  $|\zeta| \leq 1$ . A random variable with  $0 < |\zeta| < 1$  is called (second-order) *weak-sense* non-circular, for  $|\zeta| = 1$  we speak of (second-order) *strict-sense* non-circularity. Strict-sense non-circular random variables are sometimes also referred to as rectilinear [68]. Note that strict-sense non-circularity implies a linear dependence between real and imaginary part of  $Z$ . We can think of  $Z$  as a real-valued random variable which is rotated by a complex phase term, i.e.,  $Z = W \cdot e^{j\varphi}$ , where  $W \in \mathcal{R}$  is a random variable and  $\varphi$  is deterministic (fixed). In a communication system, the amplitudes  $s_i(t)$  are non-circular random variables if the symbols are drawn from constellations which are not circularly symmetric. We obtain strict-sense non-circular amplitudes if the transmitters use real-valued constellations (such as BPSK or ASK), which appear rotated by complex phase terms at the receiver since each transmitter may have a different transmission delay. Note that OQPSK and MSK symbols can be transformed into rectilinear amplitudes by applying an appropriate derotation at the receiver [68]. Figure 15.10 shows an example of an I/Q diagram displaying Inphase vs. Quadrature (I/Q) components



**FIGURE 15.10**

Example for strict-sense non-circular amplitudes: two users (red, blue) transmit symbols drawn from real-valued constellations. Since they undergo different phase rotations, the I/Q diagram at the receiver consists of differently rotated real-valued random variables, i.e., the complex symbols  $s_i(t)$  can be described as strict-sense non-circular random variables. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this book.)

of the received symbols for two users that transmit using a real-valued constellation. Since each user's transmission undergoes a different phase rotation ( $\varphi_i$ ), the receiver observes rotated real-valued random variables that satisfy the strict-sense non-circularity property. For the source symbol matrix  $S \in \mathcal{C}^{d \times N}$  this implies the structure [45]

$$S = \Psi \cdot S_0, \quad (15.47)$$

where  $S_0 \in \mathcal{R}^{d \times N}$  and  $\Psi = \text{diag}([\mathrm{e}^{\jmath\varphi_1}, \dots, \mathrm{e}^{\jmath\varphi_d}])$ .

Non-circular random variables can be exploited in signal processing applications since they carry a specific structure. If  $s[n]$  is a non-circular random variable in addition to the covariance matrix  $\Sigma_s = \mathrm{E}\{s[n] \cdot s[n]^H\}$ , the pseudo-covariance matrix  $\tilde{\Sigma}_s = \mathrm{E}\{s[n] \cdot s[n]^T\}$  contains statistical information about  $s[n]$  we can take advantage of. For circular random variables, the pseudo-covariance matrix is equal to the zero matrix.

### 3.15.3 Subspace estimation

#### 3.15.3.1 Matrix-based subspace estimation

The covariance matrix of the array output signal is defined as

$$\begin{aligned} \Sigma_x &\triangleq \mathrm{E}\{\mathbf{x}(t)\mathbf{x}^H(t)\} \\ &= \mathbf{A}\mathrm{E}\{s(t)s^H(t)\}\mathbf{A}^H + \mathrm{E}\{\mathbf{n}(t)\mathbf{n}^H(t)\} \\ &= \mathbf{A}\Sigma_s\mathbf{A}^H + \sigma^2\mathbf{I}_M \end{aligned} \quad (15.48)$$

in which it is assumed that the noise and the signals are independent and have zero-mean. Moreover, in (15.48), we define the signal covariance matrix as

$$\Sigma_s \triangleq \mathrm{E}\{s(t)s^H(t)\}. \quad (15.49)$$

Assuming that the signals are uncorrelated, the  $d \times d$  matrix  $\Sigma_s$  becomes diagonal and non-singular such that

$$\Sigma_s = \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_d\}, \quad (15.50)$$

where  $\sigma_l \neq 0$  for  $l = 1, \dots, d$  is the signal power of the  $l$ th source. Therefore, since the  $M \times d$  array steering matrix  $\mathbf{A}$  is of full-column rank, the  $M \times M$  matrix  $\mathbf{A}\Sigma_s\mathbf{A}^H$  is rank-deficient and of rank  $d$ . This low-rank property can be exploited in the presence of the sensor noise to construct two subspaces which are at the foundation of the subspace-based estimation methods. From (15.48) it can be observed that  $\Sigma_x$  has  $M - d$  eigenvalues equal to the noise power  $\sigma^2$  and  $d$  eigenvalues greater than  $\sigma^2$ . In other words, if we define  $\lambda_m$  as the  $m$ th largest eigenvalue of  $\Sigma_x$  then

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d > \lambda_{d+1} = \dots = \lambda_M = \sigma^2. \quad (15.51)$$

Hence the  $L$  largest eigenvalues of  $\Sigma_x$  are called signal eigenvalues and the rest of the  $M - d$  eigenvalues are called noise eigenvalues. After performing the eigen-decomposition on the array output covariance

matrix  $\Sigma_x$  we write

$$\begin{aligned}\Sigma_x &= \sum_{m=1}^M \lambda_m \mathbf{u}_m \mathbf{u}_m^H \\ &= \mathbf{U}_s \Lambda_s \mathbf{U}_s^H + \mathbf{U}_n \Lambda_n \mathbf{U}_n^H \\ &= \mathbf{U}_s \Lambda_s \mathbf{U}_s^H + \sigma^2 \mathbf{U}_n \mathbf{U}_n^H,\end{aligned}\quad (15.52)$$

where  $\Lambda_s$  and  $\Lambda_n$  denote, respectively, the  $d \times d$  and the  $(M-d) \times (M-d)$  diagonal matrices containing the signal and the noise eigenvalues

$$\Lambda_s = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_d\}, \quad (15.53)$$

$$\Lambda_n = \sigma^2 \mathbf{I}_{M-d}. \quad (15.54)$$

Moreover, the  $M \times d$  signal- and the  $M \times (M-d)$  noise-eigenvector matrices  $\mathbf{U}_s$  and  $\mathbf{U}_n$ , respectively, contain the eigenvectors corresponding to the signal and to the noise eigenvalues. We will refer to the matrices  $\mathbf{U}_s$  and  $\mathbf{U}_n$  as signal subspace matrix and noise subspace matrix, respectively.

It is well-known that assuming both  $\mathbf{A}$  and  $\Sigma_s$  to be full column-rank matrices, both the array steering matrix and the signal-eigenvectors subspace span the same subspace, i.e.,

$$\text{range}\{\mathbf{U}_s\} = \text{range}\{\mathbf{A}\}, \quad (15.55)$$

whereas, the columns of  $\mathbf{U}_n$  span its orthogonal complement, i.e., the null space of  $\mathbf{A}^H$ . Consequently,

$$\mathbf{U}_s \mathbf{U}_s^H = \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H, \quad (15.56)$$

$$\mathbf{U}_n \mathbf{U}_n^H = \mathbf{I}_M - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H. \quad (15.57)$$

In other words, for some full-rank  $d \times d$  matrix  $\mathbf{P}$ , called mixing matrix, the following important relationship can be formed which will be referred to in the text many times

$$\mathbf{A} = \mathbf{U}_s \mathbf{P}. \quad (15.58)$$

Similarly, since  $\mathbf{P}$  is nonsingular, hence invertible, we can also write

$$\begin{aligned}\mathbf{U}_s &= \mathbf{A} \mathbf{P}^{-1} \\ &= \mathbf{A} \mathbf{P}',\end{aligned}\quad (15.59)$$

where for the simplicity in the notations in the later sections we define

$$\mathbf{P}' \triangleq \mathbf{P}^{-1}. \quad (15.60)$$

The true array covariance matrix is generally unknown in practice, therefore its finite sample estimate

$$\widehat{\Sigma}_x = \frac{1}{N} \sum_{t=1}^N \mathbf{x}(t) \mathbf{x}^H(t), \quad (15.61)$$

which is the maximum likelihood estimate of the  $\Sigma_x$  in (15.48) in the case of Gaussian noise is used. It is assumed that the number of snapshots is larger than the number of sensors, i.e.,  $N \geq M$ . This assumption is required so that the rank of the obtained sample covariance matrix  $\widehat{\Sigma}_x$  (in the presence of the noise) becomes equal to  $M$ ; a necessary condition for the subsequent construction of the signal and noise subspaces for the subspace-based methods which are discussed in next chapter.

Let  $\hat{\lambda}_m$ , for  $m = 1, \dots, M$ , denote the  $m$ th largest eigenvalue of the sample covariance matrix  $\widehat{\Sigma}_x$  in (15.61) such that

$$\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_d \geq \hat{\lambda}_{d+1} \geq \dots \geq \hat{\lambda}_M. \quad (15.62)$$

Similar to the true covariance matrix  $\widehat{\Sigma}_x$  the eigenvalues can be divided into the signal eigenvalues containing the  $d$  largest eigenvalues, i.e.,  $\hat{\lambda}_1, \dots, \hat{\lambda}_d$ , and the noise eigenvalues consisting of  $M - d$  smallest eigenvalues, i.e.,  $\hat{\lambda}_{d+1}, \dots, \hat{\lambda}_M$ . Similarly, the eigen-decomposition of the sample covariance matrix  $\widehat{\Sigma}_x$  can be written as

$$\begin{aligned} \widehat{\Sigma}_x &= \sum_{m=1}^M \hat{\lambda}_m \hat{\mathbf{u}}_m \hat{\mathbf{u}}_m^H \\ &= \widehat{\mathbf{U}}_s \widehat{\Lambda}_s \widehat{\mathbf{U}}_s^H + \widehat{\mathbf{U}}_n \widehat{\Lambda}_n \widehat{\mathbf{U}}_n^H, \end{aligned} \quad (15.63)$$

where  $\widehat{\Lambda}_s$  and  $\widehat{\Lambda}_n$  denote, respectively, the  $d \times d$  and the  $(M - d) \times (M - d)$  diagonal matrices containing the signal and the noise eigenvalues

$$\widehat{\Lambda}_s = \text{diag}\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_d\}, \quad (15.64)$$

$$\widehat{\Lambda}_n = \text{diag}\{\hat{\lambda}_{d+1}, \hat{\lambda}_{d+2}, \dots, \hat{\lambda}_M\}. \quad (15.65)$$

The matrices  $\widehat{\mathbf{U}}_s$  and  $\widehat{\mathbf{U}}_n$  are, respectively, the estimates of the  $M \times d$  signal- and the  $M \times (M - d)$  noise-eigenvector matrices consist of the eigenvectors corresponding to the signal and to the noise eigenvalues.

Note that, instead of performing an eigendecomposition of (15.63) and finding the  $d$  dominant eigenvectors, one can also compute an SVD of the measurement matrix  $X$  directly and obtain  $\widehat{\mathbf{U}}_s$  from the  $d$  dominant left singular vectors. The truncated SVD of  $X$  is given by

$$X \approx \widehat{\mathbf{U}}_s \cdot \widehat{\Sigma}_s \cdot \widehat{\mathbf{V}}_s^H, \quad (15.66)$$

where  $\widehat{\Sigma}_s \in \mathcal{R}^{d \times d}$  and  $\widehat{\mathbf{V}}_s \in \mathcal{C}^{N \times d}$ . We refer to this process as the “direct data approach” to obtain the subspace.

### 3.15.3.2 Subspace estimation with a small number of snapshots

In particular applications, the number  $N$  of available data snapshots in (15.18) may be insufficient to span the entire signal subspace. The rank of the sample covariance matrix  $\widehat{\Sigma}_x$  in (15.61) is however lower bounded by the number of linearly independent snapshots. Hence, in case that  $N < d$  the sample covariance matrix exhibits a rank smaller than  $d$ . In other words, there exist at least  $M - N > N - d$  zero eigenvalues and the signal subspace can no longer be extracted from the  $d$  principal eigenvectors as in the large snapshot case. To overcome these difficulties the redundancy in the data snapshots resulting

from the uniform linear array structure encountered in uniform  $R$ -D arrays can be exploited to create additionally, so-called virtually, snapshots. Consider for simplicity the ULA geometry composed of  $M$  omni-directional sensor as defined in (15.25). A generalization of the spatial smoothing technique to general uniform linear  $R$ -D array structures is then straight forward. The ULA can be decomposed in  $K$  overlapping identical uniform linear subarray of length  $M_{\text{sub}} = M - K + 1 > d$  such that

$$\mathbf{a}_{\text{sub}}(\mu) \triangleq \left[ 1, e^{j\mu}, \dots, e^{j(M_{\text{sub}}-1)\mu} \right]^T = e^{j(k-1)\mu} \mathbf{J}_k \mathbf{a}(\mu), \quad (15.67)$$

where  $\mathbf{a}(\mu)$  is defined in (15.27) and where

$$\mathbf{J}_k \triangleq \left[ \mathbf{0}_{M_{\text{sub}} \times (k-1)} \quad \mathbf{I}_{M_{\text{sub}}} \quad \mathbf{0}_{M_{\text{sub}} \times (M-k+1)} \right]^T \quad (15.68)$$

is a  $M_{\text{sub}} \times M$  subarray selection matrix. With the above definition it can readily be verified that

$$\begin{aligned} \mathbf{x}_{\text{sub}}(t) &\triangleq \mathbf{J}_k \mathbf{x}(t) = \mathbf{A}_{\text{sub}} \mathbf{s}(t) + \mathbf{J}_k \mathbf{n}(t) \\ &= \mathbf{A}_{\text{sub}} \Phi^{k-1} \mathbf{s}(t) + \mathbf{J}_k \mathbf{n}(t) \\ &= \mathbf{A}_{\text{sub}} \mathbf{s}_k(t) + \mathbf{n}_k(t) \end{aligned} \quad (15.69)$$

for  $k = 1, \dots, K$  where  $\mathbf{s}_k(t) \triangleq \Phi^{k-1} \mathbf{s}(t)$ ,  $\mathbf{n}_k(t) \triangleq \mathbf{J}_k \mathbf{n}(t)$ ,  $\mathbf{A}_{\text{sub}} = \mathbf{J}_K \mathbf{A}$  and

$$\Phi \triangleq \text{diag}\{e^{j\mu_1}, \dots, e^{j\mu_d}\} \quad (15.70)$$

is defined in accordance to (15.42). We observe from (15.69) that  $K$  generally linearly independent snapshots  $\mathbf{x}_{\text{sub}}$  of size  $M_{\text{sub}} \times 1$  can be obtained from a single measurement  $\mathbf{x}(t)$  of size  $M \times 1$ . In this case, even from a single snapshot a sample covariance matrix of rank  $M_{\text{sub}}$  can generally be computed as

$$\begin{aligned} \widehat{\Sigma}_{\mathbf{x}}(t) &\triangleq \frac{1}{K} \sum_{k=1}^K \mathbf{x}_k(t) \mathbf{x}_k^H(t) \\ &\simeq \mathbf{A}_{\text{sub}} \left( \frac{1}{K} \sum_{k=1}^K \mathbf{s}_k(t) \mathbf{s}_k^H(t) \right) \mathbf{A}_{\text{sub}}^H + \frac{1}{K} \sum_{k=1}^K \mathbf{n}_k(t) \mathbf{n}_k^H(t) \\ &= \mathbf{A}_{M_{\text{sub}}} \widehat{\Sigma}_{\mathbf{s}, K}(t) \mathbf{A}_{M_{\text{sub}}}^H + \widehat{\Sigma}_{\mathbf{n}}(t), \end{aligned} \quad (15.71)$$

where

$$\begin{aligned} \widehat{\Sigma}_{\mathbf{s}, K}(t) &\triangleq \frac{1}{K} \sum_{k=1}^K \mathbf{s}_k(t) \mathbf{s}_k^H(t) \\ &= \frac{1}{K} \sum_{k=1}^K \Phi^{k-1} \mathbf{s}(t) \mathbf{s}^H(t) (\Phi^*)^{k-1} \end{aligned} \quad (15.72)$$

and  $\widehat{\Sigma}_{\mathbf{n}}(t) \triangleq \frac{1}{K} \sum_{k=1}^K \mathbf{n}_k(t) \mathbf{n}_k^H(t)$  denote signal and noise sample covariance matrices, respectively, corresponding to the  $t$ th snapshot. Note that the latter approximation is based on the consideration

that due to the independence of the source and noise signals for sufficiently large  $K$ , the cross terms can be neglected. The process of averaging overlapping subarray snapshots is generally referred to as spatial smoothing. Taking statistical expectation on both sides of (15.72) we further observe that spatial smoothing can also be applied in the case of correlated or coherent sources, i.e., when the signal covariance matrix  $\Sigma_s \triangleq E\{s(t)s^H(t)\}$  in (15.49) is rank deficient. Unlike the conventional (estimated) signal covariance matrix, the spatially smoothed estimate

$$\begin{aligned}\Sigma_{s,K} &\triangleq E\{\widehat{\Sigma}_{s,K}(t)\} \\ &= \frac{1}{K} \sum_{k=1}^K \Phi^{k-1} E\{s(t)s^H(t)\} (\Phi^*)^{k-1} \\ &= \frac{1}{K} \sum_{k=1}^K \Phi^{k-1} \Sigma_s (\Phi^*)^{k-1}\end{aligned}\quad (15.73)$$

generally exhibits up to  $K$  times the rank of  $\Sigma_s$ , obviously however, without exceeding a rank equal to the dimension  $d$ . Despite the benefits of spatial smoothing in the case of small snapshot numbers and correlated sources we remark that there is a performance penalty associated with non-coherent averaging over subarray snapshots, resulting in reduced resolution capability of the DOA estimation methods due to the reduction of the aperture size. Further, we note that due to the reduction of the effective array size from  $M$  to  $M - K + 1$ , the total number of resolvable sources is reduced in spatial smoothing. In practice, a compromise between the subspace separation capability of the sample covariance matrix, the resolution performance of the DOA estimation methods, and the computational complexity needs to be found [26, 69, 70].

### 3.15.3.3 Forward-backward averaging and real-valued subspace estimation

If the array is centro-symmetric, i.e.,  $\Pi_M \cdot A^* = A \cdot \Delta$  (cf. (15.32)), we can apply forward-backward averaging (FBA) to the data matrix  $X$ . FBA uses a symmetry in the data to create an additional set of  $N$  “virtual” snapshots. Moreover, via FBA, two coherent source can be decorrelated. The augmented measurement matrix  $X$  can be written as

$$X^{(fba)} = [X \quad \Pi_M \cdot X^* \cdot \Pi_N] \in \mathcal{C}^{M \times 2N}. \quad (15.74)$$

Note that  $X^{(fba)}$  has  $2N$  columns, i.e., the number of snapshots has been virtually doubled. Since  $X^{(fba)}$  is a centro-symmetric matrix, we can apply the one-to-one mapping between the set of centro-symmetric matrices and the set of real-valued matrices from [38]. In other words, the matrix

$$\varphi(X^{(fba)}) = Q_M^H \cdot X^{(fba)} \cdot Q_{2N} = T(X) \quad (15.75)$$

is real-valued for unitary matrices  $Q_M$  that are left- $\Pi$ -real, i.e., they satisfy  $Q_M^* \cdot \Pi_M = Q_M$ . The notation  $T(X)$  is introduced to simplify the application of both, FBA and the real-valued transformation. Note that the transformation (15.75) can be efficiently implemented by considering sparse unitary left- $\Pi$ -real matrices [15] shown in (15.12). The advantage of (15.75) is that since the matrix is real-valued,

a subspace estimate is obtained by a real-valued SVD, which has a lower computational complexity compared to the complex-valued counterpart. A real-valued signal subspace estimate  $\widehat{\mathbf{E}}_s \in \mathcal{R}^{M \times d}$  is then obtained by collecting the  $d$  dominant left singular vectors of  $\varphi(X^{(\text{fba})})$  into a matrix. Based on  $\widehat{\mathbf{E}}_s$ , “unitary” versions of many DOA estimation algorithms can be defined, e.g., the Unitary ESPRIT algorithm discussed in Section 3.15.4.1.

### 3.15.3.4 Tensor-based subspace estimation

To find a subspace estimate that takes the natural tensor structure into account we employ a multi-dimensional extension of the SVD in form of a suitable tensor decomposition. We choose the higher-order SVD (HOSVD) since it is easily computed via SVDs of the unfoldings of the tensor. Moreover, the truncated HOSVD<sup>6</sup> allows for multilinear low-rank approximation in a manner similar to the truncated SVD.

Let  $\mathcal{X}_0$  be the noise-free observation, such that  $\mathcal{X} = \mathcal{X}_0 + \mathcal{N}$ . Then, the SVD of  $r$ th unfolding of  $\mathcal{X}_0$  and  $\mathcal{X}$  can be expressed as

$$[\mathcal{X}_0]_{(r)} = \begin{bmatrix} \mathbf{U}_r^{[s]} & \mathbf{U}_r^{[n]} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_r^{[s]} & \mathbf{0}_{d \times (N-d)} \\ \mathbf{0}_{(M-d) \times d} & \mathbf{0}_{(M-d) \times (N-d)} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{V}_r^{[s]} & \mathbf{V}_r^{[n]} \end{bmatrix}^H, \quad (15.76)$$

$$[\mathcal{X}]_{(r)} = \begin{bmatrix} \widehat{\mathbf{U}}_r^{[s]} & \widehat{\mathbf{U}}_r^{[n]} \end{bmatrix} \cdot \begin{bmatrix} \widehat{\boldsymbol{\Sigma}}_r^{[s]} & \mathbf{0}_{d \times (N-d)} \\ \mathbf{0}_{(M-d) \times d} & \widehat{\boldsymbol{\Sigma}}_r^{[n]} \end{bmatrix} \cdot \begin{bmatrix} \widehat{\mathbf{V}}_r^{[s]} & \widehat{\mathbf{V}}_r^{[n]} \end{bmatrix}^H, \quad (15.77)$$

where  $\mathbf{U}_r^{[s]} \in \mathcal{C}^{M_r \times p_r}$  and  $\mathbf{U}_r^{[n]} \in \mathcal{C}^{M_r \times (M_r - p_r)}$  denote the basis for the  $r$ -space and its orthogonal complement, respectively. Here,  $p_r$  denotes the  $r$ -rank<sup>7</sup> of  $\mathcal{X}_0$ . Based on the  $r$ -spaces  $\mathbf{U}_r^{[s]}$  we can estimate the core tensor  $\mathcal{S}^{[s]} \in \mathcal{C}^{p_1 \times \dots \times p_R \times p_{R+1}}$  via

$$\mathcal{S}^{[s]} = \mathcal{X}_0 \times_1 \mathbf{U}_1^{[s]H} \cdots \times_R \mathbf{U}_R^{[s]H} \times_{R+1} \mathbf{U}_{R+1}^{[s]H}, \quad (15.78)$$

$$\widehat{\mathcal{S}}^{[s]} = \mathcal{X} \times_1 \widehat{\mathbf{U}}_1^{[s]H} \cdots \times_R \widehat{\mathbf{U}}_R^{[s]H} \times_{R+1} \widehat{\mathbf{U}}_{R+1}^{[s]H}. \quad (15.79)$$

The truncated HOSVD then reads as

$$\mathcal{X}_0 = \mathcal{S}^{[s]} \times_1 \mathbf{U}_1^{[s]} \cdots \times_R \mathbf{U}_R^{[s]} \times_{R+1} \mathbf{U}_{R+1}^{[s]}, \quad (15.80)$$

$$\mathcal{X} \approx \widehat{\mathcal{S}}^{[s]} \times_1 \widehat{\mathbf{U}}_1^{[s]} \cdots \times_R \widehat{\mathbf{U}}_R^{[s]} \times_{R+1} \widehat{\mathbf{U}}_{R+1}^{[s]} = \widehat{\mathcal{X}}. \quad (15.81)$$

If we compare the truncated HOSVD of  $\mathcal{X}$  in (15.81) with the truncated SVD in (15.66), we observe that a unique feature of the HOSVD is that it performs low-rank approximations in all  $R + 1$  modes. Hence, the multilinear structure is exploited to perform more efficient denoising.

<sup>6</sup>Note that, unlike the truncated SVD, the truncated HOSVD does not provide the Least-Squares optimal low-rank approximation of the tensor. In [31], an iterative Higher Order Orthogonal Iterations (HOOI) algorithm is proposed for this task. However, since the improvement of the HOOI solution compared to the truncated HOSVD is only marginal, we propose to use the truncated HOSVD for signal subspace estimation.

<sup>7</sup>In practice, we can estimate the  $r$ -ranks individually via a model order selection scheme operating on all unfoldings. Alternatively, we can apply tensor-based model order selection schemes [71] to estimate  $d$  and then use  $p_r = \min(M_r, d)$ .

As a multilinear extension of the subspace estimate  $\widehat{\mathbf{U}}_s$  we introduce the following *signal subspace tensor*<sup>8</sup>  $\widehat{\mathcal{U}}^{[s]} \in \mathcal{C}^{M_1 \times \dots \times M_R \times d}$

$$\widehat{\mathcal{U}}^{[s]} = \widehat{\mathcal{S}}^{[s]} \times_1 \widehat{\mathbf{U}}_1^{[s]} \cdots \times_R \widehat{\mathbf{U}}_R^{[s]} \times_{R+1} \widehat{\boldsymbol{\Sigma}}_{R+1}^{[s]^{-1}}. \quad (15.82)$$

A more formal link between  $\widehat{\mathcal{U}}^{[s]}$  and  $\widehat{\mathbf{U}}_s^{[s]}$  is given by the following important identity (shown for  $R = 2$  in [44])

$$\left[ \widehat{\mathcal{U}}^{[s]} \right]_{(R+1)}^T = (\widehat{\mathbf{T}}_1 \otimes \widehat{\mathbf{T}}_2 \otimes \cdots \otimes \widehat{\mathbf{T}}_R) \cdot \widehat{\mathbf{U}}_s, \quad (15.83)$$

where  $\widehat{\mathbf{T}}_r \in \mathcal{C}^{M_r \times M_r}$  represent estimates of the projection matrices onto the  $r$ -spaces of  $\mathcal{X}_0$ , which are computed via  $\widehat{\mathbf{T}}_r = \widehat{\mathbf{U}}_r^{[s]} \widehat{\mathbf{U}}_r^{[s]H}$ .

It is worth pointing out that (15.83) provides some rather interesting insights. Firstly, it shows that the matrix  $\left[ \widehat{\mathcal{U}}^{[s]} \right]_{(R+1)}^T \in \mathcal{C}^{M \times d}$  yields an estimate for the signal subspace, which can be used to replace the matrix  $\widehat{\mathbf{U}}_s$ . Secondly, it demonstrates that an explicit computation of the core tensor of  $\mathcal{X}$  is actually not necessary if only the HOSVD-based subspace estimate is needed. Thirdly, it shows that the HOSVD-based subspace estimate can be seen as the projection of the (unstructured) matrix-based subspace estimate onto the Kronecker structure inherent in the data and that this projection is achieved by virtue of the Kronecker product of  $r$ -space projection matrices. Since this projection leaves the desired signal unaltered it affects only the noise, filtering out the part of the noise which does not obey the required Kronecker structure. This observation provides a different way of understanding the denoising obtained via multilinear rank reduction. The relation (15.83) also shows that for any mode  $r$  where  $d \geq M_r$  we have  $\widehat{\mathbf{T}}_r = \mathbf{I}_r$  and hence no performance improvement can be obtained in this particular mode  $r$ . As a corollary from this we have  $\left[ \widehat{\mathcal{U}}^{[s]} \right]_{(R+1)}^T = \widehat{\mathbf{U}}_s$  if  $d \geq \max_{r=1,2,\dots,R} (M_r)$ , i.e., there is no improvement in terms of the subspace estimation accuracy from the HOSVD-based subspace estimate if the number of wavefront  $d$  is greater than or equal to the number of sensors in all  $R$  modes.

Forward-backward averaging (FBA) and the real-valued transformation that are introduced in Section 3.15.3.3 for the matrix case can also be formulated in terms of tensors. For forward-backward averaging we can write [42]

$$\mathcal{X}^{(\text{fba})} = [\mathcal{X} \sqcup_{R+1} \mathcal{X}^* \times_1 \boldsymbol{\Pi}_{M_1} \cdots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N]. \quad (15.84)$$

The subsequent real-valued transformation can be expressed as

$$\mathcal{T}(\mathcal{X}) = \mathcal{X}^{(\text{fba})} \times_1 \mathcal{Q}_{M_1}^H \cdots \times_R \mathcal{Q}_{M_R}^H \times_{R+1} \mathcal{Q}_{2N}^H. \quad (15.85)$$

Note that the spatial smoothing technique described in Section 3.15.3.2 can also be formulated in tensor notation, as shown in [42]. Moreover, a tensor-based spatial smoothing technique for 1-D damped and undamped harmonic retrieval with a single snapshot is discussed in [70]. The extension to multiple snapshots is introduced in [72] and an  $R$ -D extension is shown in [73]. A major advantage of [72, 73] is that the performance of the ESPRIT-type parameter estimates is almost independent of the choice of the subarray size for the spatial smoothing.

---

<sup>8</sup>Note that in [42],  $\widehat{\mathcal{U}}^{[s]}$  was defined without the multiplication by  $\widehat{\boldsymbol{\Sigma}}_{R+1}^{[s]^{-1}}$  in the  $(R+1)$ th mode. While this has no impact on the subspace of interest, we include it here since this definition simplifies the notation at this point.

### 3.15.4 Subspace-based algorithms

Passive source localization by an array of sensors arises in several applications, such as radio astronomy, sonar, seismology, radar, geophysics and oceanography. Several algorithms have been proposed to solve the estimation problem of far-field narrowband source localization. Among them, the maximum likelihood (ML) approaches have been shown to reach the best accuracy. Nevertheless, the main drawback of ML schemes is their computationally cost which make them impractical in reality. To overcome this drawback, several subspace-based algorithms have been developed in the literature. More precisely, the key idea is to take into account the intrinsic properties of the eigen-structure of the observed covariance matrix. Based on this, a number of computationally simpler high-resolution algorithm have been proposed. These subspace-based algorithms can be classified, into a *spectral searching techniques* and *search free techniques*, as follows:

- *Spectral searching techniques*, e.g., MUSIC [74] and its variant (weighted MUSIC [75], Min-Norm [76, 77], sequential MUSIC [78], recursively applied and projected MUSIC [79]), RARE [80], weighted subspace fitting [81].
- *Search free techniques*
  - *Polynomial-rooting techniques*, e.g., root-MUSIC [6] and its variant (unitary root-MUSIC [15], interpolated root-MUSIC [82]), root-RARE [80], MODE [83], manifold separation [84, 85], Fourier domain root-MUSIC [86].
  - *Matrix-shifting techniques*, e.g., ESPRIT [8] and its variant (Unitary ESPRIT [15], weighted ESPRIT [14]) and matrix pencil methods [11–13].

#### 3.15.4.1 One-dimensional algorithms using matrix-based subspace estimates

This subsection is dedicated to the one-dimensional algorithms. For sake of simplicity, in the following, steering vectors will be indexed by  $\theta$ ,  $z$  or  $\check{z}$  depending on the situation.

##### 3.15.4.1.1 MUSIC

As it has been noticed in (15.56), the manifold matrix  $\mathbf{A}(\theta)$  spans the same subspace as the signal eigenvector matrix  $\mathbf{U}_s$ . Therefore, each column of the array steering matrix is orthogonal to the noise subspace  $\mathbf{U}_n$ , hence

$$\mathbf{U}_n^H \mathbf{a}(\theta_l) = \mathbf{0} \quad (15.86)$$

for  $\theta_l = \theta_1, \dots, \theta_d$  or equivalently

$$\mathbf{a}^H(\theta_l) \mathbf{U}_n \mathbf{U}_n^H \mathbf{a}(\theta_l) = 0. \quad (15.87)$$

This is the core idea of the MUSIC estimation method. In practice, in order to estimate the DOAs, the estimate of the noise subspace matrix  $\widehat{\mathbf{U}}_n$  obtained from the sample covariance matrix  $\widehat{\Sigma}_x$  in (15.63) must be used. Therefore, the following “spectral” function is proposed in [74]

$$\begin{aligned} f_{\text{MUSIC}}(\theta) &= \frac{1}{\|\widehat{\mathbf{U}}_n^H \mathbf{a}(\theta)\|^2} \\ &= \frac{1}{\mathbf{a}^H(\theta) \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}_n^H \mathbf{a}(\theta)}. \end{aligned} \quad (15.88)$$

The estimated DOAs  $\hat{\theta}_1, \dots, \hat{\theta}_d$  are, then, obtained by the angles  $\theta$  corresponding to the  $d$  maxima of  $f_{\text{MUSIC}}(\theta)$ . The so-called MUSIC pseudo null-spectrum function  $f_{\text{N-MUSIC}}$ , given as the denominator of the MUSIC function in (15.88), can be interpreted as the measure of the projection of the array manifold vector onto the noise subspace  $\widehat{\mathbf{U}}_n$  which ideally for the true DOAs is zero. Then, the estimated DOAs are the ones that minimize this projection. To find the DOAs, a scan over the entire field-of-view (FOV) is required and the function  $f_{\text{MUSIC}}(\theta)$  in (15.88) needs to be evaluated for each  $\theta$ .

The popularity of the MUSIC DOA estimation method is due to its relative computational simplicity (compared to maximum-likelihood method which requires multidimensional search [22]), its high resolution capability (compared to traditional beamforming technique and Capon method [87]), and its asymptotic efficiency [88].

---

**Algorithm 1.** Summary of the MUSIC algorithm

---

1. Compute the eigendecomposition of  $\widehat{\Sigma}_x$  and obtain the matrix  $\widehat{\mathbf{U}}_n$ .
  2. Find the  $d$  maxima of  $f_{\text{MUSIC}}(\theta)$  in (15.88) by scanning the entire FOV.
- 

### 3.15.4.1.2 Weighted MUSIC

As it can be observed, in the MUSIC spectral function of (15.88), all the noise eigenvectors are treated equally. The MUSIC method can be extended to include a specific weighting matrix for controlling the effect of each noise eigenvector on the estimates. A proper choice of the weighting matrix will be particularly useful to improve the performance of the estimators in difficult situations such as low number of snapshots and low SNR to overcome some of the shortcomings of the MUSIC method [75]. Toward this end, the following spectrum function is defined to take into account the different effects of the noise eigenvectors

$$f_{\text{WMUSIC}}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \widehat{\mathbf{U}}_n \mathbf{W}_{\text{WMUSIC}} \widehat{\mathbf{U}}_n^H \mathbf{a}(\theta)}. \quad (15.89)$$

It is clear that the conventional MUSIC function in (15.88) is a special case of the weighted-MUSIC function in (15.89) with  $\mathbf{W}_{\text{WMUSIC}} = \mathbf{I}_{M-d}$ . This choice of weight matrix is proved to be the optimal weight matrix in the sense that it yields the best asymptotic performance [83].

A useful choice of the weighting matrix is

$$\mathbf{W}_{\text{WMUSIC}} = \widehat{\mathbf{U}}_n^H \mathbf{e}_1 \mathbf{e}_1^T \widehat{\mathbf{U}}_n, \quad (15.90)$$

where  $\mathbf{e}_1$  is the first column of the  $M \times M$  identity matrix. The choice of  $\mathbf{W}_{\text{WMUSIC}}$  in (15.89) coincides with the well-known Min-Norm method [76, 77]. In the Min-Norm method, a non-zero vector with minimum norm in the noise subspace, i.e., a linear combination of the noise eigenvectors, is obtained. Then, the orthogonality of this minimum length vector and the array manifold vector is measured similar to the one used for the MUSIC method in (15.88) for the angles in the FOV. The Min-Norm method is known to yield an improved resolution capability of distinguishing two close sources, as compared to the MUSIC method in the ULAs [22].

**Algorithm 2.** Summary of the weighted MUSIC algorithm

1. Compute the eigendecomposition of  $\widehat{\Sigma}_x$  and obtain the matrix  $\widehat{U}_n$ .
2. Choose the weighting matrix  $W_{\text{WMUSIC}}$  (e.g.,  $W_{\text{WMUSIC}} = I_{M-d}$  for the MUSIC algorithm,  $W_{\text{WMUSIC}} = \widehat{U}_n^H e_1 e_1^T \widehat{U}_n$  for the Min-Norm method).
3. Find the  $d$  maxima of  $f_{\text{WMUSIC}}(\theta)$  in (15.89) by scanning the entire FOV.

**3.15.4.1.3 Root-MUSIC**

The root-MUSIC DOA estimation method [6] exploits the Vandermonde structure of the array manifold vector in the ULAs in (15.24) to estimate the DOAs through a search-free algorithm based on polynomial rooting. Defining

$$z = e^{\frac{2\pi}{\lambda} \Delta \cos \theta} \quad (15.91)$$

the parametric array manifold vector  $\mathbf{a}(z)$  becomes

$$\mathbf{a}(z) = \begin{bmatrix} 1 & z & z^2 & \dots & z^{M-1} \end{bmatrix}^T. \quad (15.92)$$

Furthermore, it is simple to show that

$$\mathbf{a}^H(z) = \mathbf{a}^T \left( \frac{1}{z} \right). \quad (15.93)$$

Then, the MUSIC criteria in (15.87) for the true DOAs transforms into

$$\mathbf{a}^T \left( \frac{1}{z} \right) \mathbf{U}_n \mathbf{U}_n^H \mathbf{a}(z) = 0. \quad (15.94)$$

From (15.93), it can be seen that if  $z$  is a root of the polynomial in (15.94), then its conjugate reciprocal  $1/z^*$  is also a root. Therefore, the polynomial in (15.94), which is of degree  $2M - 2$ , has  $2M - 2$  roots with  $M - 1$  roots on/inside the unit-circle and their  $M - 1$  conjugate reciprocal pairs on/outside the unit-circle. In practice, the estimate of the noise subspace matrix, i.e.,  $\widehat{U}_n$  in (15.63), from the sample covariance matrix  $\widehat{\Sigma}_x$  in (15.61) is used and the following polynomial is obtained

$$f_{\text{root-MUSIC}}(z) = \mathbf{a}^T \left( \frac{1}{z} \right) \widehat{U}_n \widehat{U}_n^H \mathbf{a}(z). \quad (15.95)$$

The true spatial frequencies, i.e., the ones associated with the true DOAs, are on the unit-circle. Therefore, to estimate the DOAs from  $f_{\text{root-MUSIC}}(z)$ , the  $d$  complex roots of  $f_{\text{root-MUSIC}}(z)$ , namely  $\hat{z}_1, \dots, \hat{z}_d$ , closest to the unit-circle and inside it should be selected and the estimated DOAs can be computed for  $l = 1, \dots, d$  from

$$\hat{\theta}_l = \cos^{-1} \left\{ \frac{\lambda}{2\pi\Delta} \angle(\hat{Z}_l) \right\}, \quad (15.96)$$

where  $\angle(\cdot)$  denotes the phase of a complex variable. It has been demonstrated [88, 89] that both MUSIC and root-MUSIC have the same asymptotic performances. From (15.96), one can observe that the

estimated DOA  $\hat{\theta}_l$  (for  $l = 1, \dots, d$ ) depends only on the phase of the root  $\hat{z}_l$  of the root-MUSIC polynomial in (15.95) and not on the magnitude of  $\hat{z}_l$ . Hence, any changes in the magnitude has no effect on the estimated DOAs and the root-MUSIC method is robust to the radial errors of the estimated roots [90]. Because of this property, the root-MUSIC method enjoys superior performance in comparison to the MUSIC method in low SNR and low number of snapshots. One can notice that, the root-MUSIC method is only applicable to the ULAs and also to the uniform circular arrays (UCAs) [59], and not to any arbitrary array geometry (unlike the MUSIC method). However, there are methods, such as array interpolation [91] and beamspace methods [92], in which the array manifold of an arbitrary array geometry can be approximately transformed into the array manifold of a virtual ULA so that the root-MUSIC method can be implemented.

**Algorithm 3.** Summary of the root-MUSIC algorithm

1. Compute the eigendecomposition of  $\widehat{\Sigma}_x$  and obtain the matrix  $\widehat{U}_n$ .
2. Root the polynomial  $\mathbf{a}^T \left(\frac{1}{z}\right) \widehat{U}_n \widehat{U}_n^H \mathbf{a}(z)$ .
3. Find the DOA estimates using the largest magnitude roots which lie inside the unit circle.

#### 3.15.4.1.4 Unitary root-MUSIC

The computational complexity of the root-MUSIC estimation technique can be further reduced by using a unitary transformation to reformulate the complex-valued algorithm to the real-valued one which makes its implementation simpler.

Let  $\mathbf{Q}_M^{(s)}$  be any unitary, column conjugate symmetric, i.e.,  $\mathbf{\Pi}_M \left(\mathbf{Q}_M^{(s)}\right)^* = \mathbf{Q}_M^{(s)}$  see (15.12).

Define the real-valued sample covariance matrix  $\widehat{\mathbf{C}} \triangleq \left(\mathbf{Q}_M^{(s)}\right)^H \widehat{\Sigma}_{FB} \mathbf{Q}_M^{(s)}$  where  $\widehat{\Sigma}_{FB}$  is the sample covariance matrix obtained from forward-backward averaging, i.e.,  $\widehat{\Sigma}_{FB} \triangleq \widehat{\Sigma}_x + \mathbf{\Pi}_P \widehat{\Sigma}_x^* \mathbf{\Pi}_P$ . Thus, it can be easily shown that

$$\widehat{\mathbf{C}} = \text{Re} \left\{ \left(\mathbf{Q}_M^{(s)}\right)^H \widehat{\Sigma}_x \mathbf{Q}_M^{(s)} \right\}. \quad (15.97)$$

The eigenvalues and the eigenvectors of the real-valued sample covariance matrix  $\widehat{\mathbf{C}}$  and the sample forward-backward matrix  $\widehat{\Sigma}_x$  are related through the unitary matrix  $\mathbf{Q}_M^{(s)}$  such that [15]

$$\hat{\mathbf{u}}_{m,\mathbf{Q}_M^{(s)}} = \left(\mathbf{Q}_M^{(s)}\right)^H \hat{\mathbf{u}}_{m,FB}, \quad (15.98)$$

$$\hat{\lambda}_{m,\mathbf{Q}_M^{(s)}} = \hat{\lambda}_{m,FB}, \quad (15.99)$$

where  $\hat{\mathbf{u}}_{m,FB}$  and  $\hat{\mathbf{u}}_{m,\mathbf{Q}_M^{(s)}}$  are the eigenvectors corresponding to the  $m$ th largest eigenvalue of  $\widehat{\Sigma}_x$  and  $\widehat{\mathbf{C}}$ , respectively, and  $\lambda_{m,FB}$  and  $\lambda_{m,\mathbf{Q}_M^{(s)}}$  are the  $m$ th largest eigenvalues of  $\widehat{\Sigma}_x$  and  $\widehat{\mathbf{C}}$ , respectively.

Writing the root-MUSIC polynomial for the forward-backward averaging and using the property of  $\mathbf{Q}_M^{(s)}$ , the Unitary root-MUSIC polynomial is obtained

$$\begin{aligned}
f_{\text{FB-RMUSIC}}(z) &= \mathbf{a}^T \left( \frac{1}{z} \right) \widehat{\mathbf{U}}_{n,\text{FB}} \widehat{\mathbf{U}}_{n,\text{FB}}^H \mathbf{a}(z) \\
&= \mathbf{a}^T \left( \frac{1}{z} \right) \mathbf{Q}_M^{(s)} \left( \mathbf{Q}_M^{(s)} \right)^H \widehat{\mathbf{U}}_{n,\text{FB}} \widehat{\mathbf{U}}_{n,\text{FB}}^H \mathbf{Q}_M^{(s)} \left( \mathbf{Q}_M^{(s)} \right)^H \mathbf{a}(z) \\
&= \mathbf{a}^T \left( \frac{1}{z} \right) \mathbf{Q}_M^{(s)} \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}} \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}}^H \left( \mathbf{Q}_M^{(s)} \right)^H \mathbf{a}(z) \\
&= \mathbf{a}_{\mathbf{Q}_M^{(s)}}^T \left( \frac{1}{z} \right) \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}} \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}}^H \mathbf{a}_{\mathbf{Q}_M^{(s)}}(z) \\
&\triangleq f_{\text{Q-RMUSIC}}(z),
\end{aligned} \tag{15.100}$$

where the transformed array steering vector is defined as

$$\mathbf{a}_{\mathbf{Q}_M^{(s)}}(z) \triangleq \left( \mathbf{Q}_M^{(s)} \right)^H \mathbf{a}(z). \tag{15.101}$$

The  $d$  roots inside and closest to the unit-circle can be used as the estimate of  $z_1, \dots, z_d$  and subsequently to obtain the DOAs in the same way explained for the root-MUSIC method.

The Unitary root-MUSIC enjoys from reduced computational complexity compared to the forward-backward root-MUSIC. In comparison with the root-MUSIC, its unitary version has the advantage of better asymptotic performance in the case of correlated sources due to the effect of the forward-backward matrix on the decorrelation of source pairs.

---

#### Algorithm 4. Summary of the Unitary root-MUSIC algorithm

---

1. Compute the eigendecomposition of  $\widehat{\mathbf{C}}$  and obtain the matrix  $\widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}}$ .
  2. Root the polynomial  $f_{\text{Q-RMUSIC}}(z) = \mathbf{a}_{\mathbf{Q}_M^{(s)}}^T \left( \frac{1}{z} \right) \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}} \widehat{\mathbf{U}}_{n,\mathbf{Q}_M^{(s)}}^H \mathbf{a}_{\mathbf{Q}_M^{(s)}}(z)$ .
  3. Find the DOA estimates using the  $d$  largest magnitude roots which lie inside the unit circle.
- 

#### 3.15.4.1.5 Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT)

The ESPRIT technique [39] is a search-free DOA estimation algorithms applicable to arrays composed of two identical (possibly unknown) subarrays with known intersubarray displacement. Such arrays satisfy a shift invariance relation as shown in Section 3.15.2.3, Eq. (15.45), which we restate here for convenience

$$\mathbf{J}_1 \cdot \mathbf{A} \cdot \Phi = \mathbf{J}_2 \cdot \mathbf{A}, \tag{15.102}$$

where  $\mathbf{J}_1, \mathbf{J}_2 \in \mathcal{R}^{M^{(\text{sel})} \times M}$  are the selection matrices for the first and the second subarray and  $\Phi = \text{diag}([\text{e}^{j\mu_1}, \dots, \text{e}^{j\mu_d}])$  contains the unknown parameters  $\mu_i, i = 1, 2, \dots, d$ . To eliminate the unknown array steering matrix we use (15.58) to replace  $\mathbf{A}$  by the estimated signal subspace  $\widehat{\mathbf{U}}_s$ . The transformed

shift invariance equation then becomes

$$\mathbf{J}_1 \cdot \widehat{\mathbf{U}}_s \cdot \boldsymbol{\Psi} \approx \mathbf{J}_2 \cdot \widehat{\mathbf{U}}_s, \quad (15.103)$$

where  $\boldsymbol{\Psi} = \mathbf{P} \cdot \boldsymbol{\Phi} \cdot \mathbf{P}^{-1}$ , i.e., the eigenvalues of  $\boldsymbol{\Psi}$  are given by  $e^{j\mu_i}$ . The system of equations in (15.103) is overdetermined since we have  $M^{(\text{sel})} \cdot d$  equations for  $d^2$  unknowns. Consequently, we need an appropriate Least Squares technique to solve it. The simplest choice is given by the method of Least Squares which selects the matrix  $\boldsymbol{\Psi}$  that minimizes the Frobenius norm of the difference of the left-hand side and the right-hand side in (15.103). The resulting closed-form solution is given by

$$\widehat{\boldsymbol{\Psi}}_{\text{LS}} = (\mathbf{J}_1 \cdot \widehat{\mathbf{U}}_s)^+ \cdot \mathbf{J}_2 \cdot \widehat{\mathbf{U}}_s, \quad (15.104)$$

where  $^+$  denotes the Moore-Penrose pseudo inverse. The 1-D Standard ESPRIT algorithm is summarized in Algorithm 5.

**Algorithm 5** [39]. Summary of 1-D Standard ESPRIT using Least Squares

1. Estimate the signal subspace  $\widehat{\mathbf{U}}_s$  via the truncated SVD of the observation matrix  $\mathbf{X} \in \mathcal{C}^{M \times N}$ .
2. Solve the overdetermined shift invariance equation

$$\mathbf{J}_1 \cdot \widehat{\mathbf{U}}_s \cdot \boldsymbol{\Psi} \approx \mathbf{J}_2 \cdot \widehat{\mathbf{U}}_s \quad (15.105)$$

for the matrix  $\boldsymbol{\Psi}$  via the method of Least Squares (LS).

3. Compute the eigenvalues  $\hat{\lambda}_i$  for  $i = 1, 2, \dots, d$  of  $\widehat{\boldsymbol{\Psi}}$ . Recover the frequencies  $\hat{\mu}_i$  via  

$$\hat{\mu}_i = \arg(\hat{\lambda}_i).$$

If the array is centro-symmetric, forward-backward averaging can be applied to the data. Since the resulting data matrix is then centro-Hermitian, it can be mapped into the real-valued domain, as explained in Section 3.15.3.3. This allows to save computational complexity because all preceding calculations can be carried out in the real domain.

Based on this idea, a “unitary” version of ESPRIT has been proposed in [15]. As shown there, a real-valued version of the invariance Eqs. (15.103) is given by

$$\mathbf{K}_1 \cdot \widehat{\mathbf{E}}_s \cdot \boldsymbol{\Upsilon} \approx \mathbf{K}_2 \cdot \widehat{\mathbf{E}}_s, \quad (15.106)$$

where  $\widehat{\mathbf{E}}_s \in \mathcal{R}^{M \times d}$  is the estimated real-valued signal subspace (cf. Section 3.15.3.3), the eigenvalues of  $\boldsymbol{\Upsilon}$  are given by  $\tan(\mu_i/2)$ , and  $\mathbf{K}_n$  are the transformed selection matrices given by

$$\mathbf{K}_1 = 2 \cdot \text{Re} \left( \mathbf{Q}_{M^{(\text{sel})}}^H \cdot \mathbf{J}_2 \cdot \mathbf{Q}_M \right), \quad (15.107)$$

$$\mathbf{K}_2 = 2 \cdot \text{Im} \left( \mathbf{Q}_{M^{(\text{sel})}}^H \cdot \mathbf{J}_2 \cdot \mathbf{Q}_M \right). \quad (15.108)$$

Since (15.103) and (15.106) follow the same algebraic form, the LS solution is given by

$$\widehat{\boldsymbol{\Upsilon}}_{\text{LS}} = (\mathbf{K}_1 \cdot \widehat{\mathbf{E}}_s)^+ \cdot \mathbf{K}_2 \cdot \widehat{\mathbf{E}}_s. \quad (15.109)$$

The 1-D Unitary ESPRIT algorithm is summarized in Algorithm 6.

---

**Algorithm 6 [15].** Summary of 1-D Unitary ESPRIT using Least Squares

1. Estimate the real-valued signal subspace  $\hat{\mathbf{E}}_s$  via the truncated SVD of the transformed real-valued observation matrix  $\mathcal{T}(X) = \mathbf{Q}_M^H \cdot [X \quad \mathbf{\Pi}_M X^* \mathbf{\Pi}_N] \cdot \mathbf{Q}_{2N} \in \mathbb{R}^{M \times 2N}$ , where  $\mathbf{Q}_p$  is a unitary  $p \times p$  left- $\mathbf{\Pi}$ -real matrix (i.e.,  $\mathbf{\Pi}_p \cdot \mathbf{Q}_p^* = \mathbf{Q}_p$ ).
2. Solve the overdetermined shift invariance equations

$$\mathbf{K}_1 \cdot \hat{\mathbf{E}}_s \cdot \mathbf{\Upsilon} \approx \mathbf{K}_2 \cdot \hat{\mathbf{E}}_s \quad (15.110)$$

for the matrix  $\mathbf{\Upsilon}$  via the method of Least Squares (LS).

3. Compute the eigenvalues  $\hat{\omega}_i$  for  $i = 1, 2, \dots, d$  of  $\hat{\mathbf{\Upsilon}}$ . Recover the frequencies  $\hat{\mu}_i$  via  $\hat{\mu}_i = 2 \cdot \arctan(\hat{\omega}_i)$ .
- 

While the LS solution is closed-form and simple to implement, it is in general suboptimal. The reason for this is that an LS solution to an overdetermined set of equations, say,  $\mathbf{A} \cdot \mathbf{x} \approx \mathbf{b}$  can always be interpreted as finding a projection of the vector  $\mathbf{b}$  onto the subspace spanned by the columns of  $\mathbf{A}$ . In that respect, one inherently assumes that  $\mathbf{A}$  is perfectly known and the only error lies on the right-hand side of the equation, i.e., we find an error term  $\Delta\mathbf{b}$  such that  $\mathbf{A} \cdot \mathbf{x} = \mathbf{b} + \Delta\mathbf{b}$  and  $\|\Delta\mathbf{b}\|_2$  is minimized. However, in the case of a shift invariance equation we have  $\mathbf{J}_1 \hat{\mathbf{U}}_s \mathbf{\Psi} \approx \mathbf{J}_2 \hat{\mathbf{U}}_s$ , which we solve for  $\mathbf{\Psi}$ . Consequently, the error clearly lies on both sides of the equation as neither “ $\mathbf{A}$ ” ( $\mathbf{J}_1 \hat{\mathbf{U}}_s$ ) nor “ $\mathbf{b}$ ” ( $\mathbf{J}_2 \hat{\mathbf{U}}_s$ ) are perfectly known.

This observation has inspired the use of the TLS procedure [48] for solving the invariance equation. TLS allows for errors in all variables, hence one error term for  $\mathbf{J}_1 \hat{\mathbf{U}}_s$  and another error term for  $\mathbf{J}_2 \hat{\mathbf{U}}_s$  is explicitly computed with the goal to align their subspaces until an exact solution for  $\mathbf{\Psi}$  exists.

The drawback of TLS is that the error terms for  $\mathbf{J}_1 \hat{\mathbf{U}}_s$  and  $\mathbf{J}_2 \hat{\mathbf{U}}_s$  are found independently of each other. However, as long as the two subarrays used for ESPRIT overlap, they have common elements. This is additional information coming from the particular structure of the array which is ignored by TLS. In order to take this structure into account, SLS was proposed in [50]. In SLS we model an explicit error term for  $\hat{\mathbf{U}}_s$ , accounting for the fact that the true source of error in the shift invariance equation is the subspace estimation error. Since the resulting cost function represents a quadratic Least Squares problem, an exact closed-form solution does not exist anymore. However, it is shown in [50] that the cost function can be solved iteratively by local linearization and that one iteration is typically sufficient.

To this end, the SLS cost function for a 1-D shift invariance equation<sup>9</sup>  $\mathbf{J}_1 \cdot \hat{\mathbf{U}}_s \cdot \mathbf{\Psi} \approx \mathbf{J}_2 \cdot \hat{\mathbf{U}}_s$  can be expressed as

$$\begin{aligned} \hat{\mathbf{\Psi}}_{SLS} &= \hat{\mathbf{\Psi}}_{LS} + \Delta\mathbf{\Psi}_{SLS}, \quad \text{where} \\ \Delta\mathbf{\Psi}_{SLS} &= \arg \min_{\Delta\mathbf{\Psi}, \Delta\mathbf{U}_s} \left\| \mathbf{J}_1 \cdot \left( \hat{\mathbf{U}}_s + \Delta\mathbf{U}_s \right) \cdot \left( \hat{\mathbf{\Psi}}_{LS} + \Delta\mathbf{\Psi} \right) - \mathbf{J}_2 \cdot \left( \hat{\mathbf{U}}_s + \Delta\mathbf{U}_s \right) \right\|_F^2 + \kappa^2 \|\Delta\mathbf{U}_s\|_F^2. \end{aligned} \quad (15.111)$$

---

<sup>9</sup>The same algorithm applies to  $R$ -D shift invariance equations (where  $\mathbf{J}_n$  is replaced by  $\mathbf{J}_n^{(r)}$  for  $n = 1, 2$  and  $r = 1, 2, \dots, R$ ) and to the transformed real-valued invariance equations (where  $\mathbf{J}_n^{(r)}$ ,  $\mathbf{U}_s$ , and  $\mathbf{\Psi}^{(r)}$  are replaced by  $\mathbf{K}_n^{(r)}$ ,  $\mathbf{E}_s$ , and  $\mathbf{\Upsilon}^{(r)}$ , respectively).

Here,  $\widehat{\Psi}_{\text{LS}}$  refers to the LS solution given by (15.104). Moreover,  $\kappa$  is a regularization constant controlling the influence of the regularization term that penalizes too large updates in  $\Delta \mathbf{U}_s$ . It is given by  $\kappa^2 = \frac{M^{(\text{sel})}}{M \cdot \alpha}$ , where  $\alpha \in (0, \infty)$  controls the amount of regularization: large values of  $\alpha$  refer to using less regularization. Since (15.111) is a quadratic Least Squares problem, it is solved iteratively by local linearization. In the  $k$ th iteration, the updates to  $\Delta \mathbf{U}_s$  and  $\Delta \Psi$  are calculated via [50]

$$\begin{aligned} \Delta \mathbf{U}_{s,k+1} &= \Delta \mathbf{U}_{s,k} + \Delta \Delta \mathbf{U}_{s,k} \quad \text{and} \quad \Delta \Psi_{k+1} = \Delta \Psi_k + \Delta \Delta \Psi_k, \quad \text{where} \\ \begin{bmatrix} \text{vec}(\Delta \Delta \Psi_k) \\ \text{vec}(\Delta \Delta \mathbf{U}_{s,k}) \end{bmatrix} &= -\mathbf{F}^+ \cdot \begin{bmatrix} \text{vec}(\mathbf{R}_k) \\ \kappa \cdot \text{vec}(\Delta \mathbf{U}_{s,k}) \end{bmatrix} \quad \text{with} \\ \mathbf{R}_k &= \mathbf{J}_1 \cdot (\widehat{\mathbf{U}}_s + \Delta \mathbf{U}_{s,k}) \cdot (\widehat{\Psi}_{\text{LS}} + \Delta \Psi_k) - \mathbf{J}_2 \cdot (\widehat{\mathbf{U}}_s + \Delta \mathbf{U}_{s,k}) \quad \text{and} \quad (15.112) \\ \mathbf{F} &= \begin{bmatrix} \mathbf{I}_d \otimes (\mathbf{J}_1 (\widehat{\mathbf{U}}_s + \Delta \mathbf{U}_{s,k})) & \left[ (\widehat{\Psi}_{\text{LS}} + \Delta \Psi_k) \otimes \mathbf{J}_1 \right] - [\mathbf{I}_d \otimes \mathbf{J}_2] \\ \mathbf{0} & \kappa \cdot \mathbf{I}_{M \cdot d} \end{bmatrix}, \end{aligned}$$

where the initial values are given by  $\Delta \mathbf{U}_{s,0} = \mathbf{0}_{M \times d}$  and  $\Delta \Psi_0 = \mathbf{0}_{d \times d}$ . Even though SLS is derived as an iterative procedure, Haardt [50] argues that only one iteration is required to achieve a considerable improvement in estimation accuracy and therefore only a single iteration is needed.

It is important to note that TLS and SLS can be used to replace LS for the solution of the shift invariance equations in all the LS-based ESPRIT algorithms that are shown in this chapter. Since they all follow the same three steps (signal subspace estimation, solution of the invariance equations, extraction of the spatial frequencies), we simply exchange the second step, using SLS instead of LS to solve the invariance equations.

A tensor-based extension of SLS was introduced in [52] under the name Tensor-Structure SLS (TS-SLS). TS-SLS is based on the underlying idea in SLS to model an explicit perturbation for the signal subspace. However, the structure of the subspace tensor is exploited explicitly by modeling individual perturbation terms for the components it is constructed from. This leads to an improved parameter estimation accuracy. Note that TS-SLS provides a tensor gain even in scenarios where the HOSVD-based subspace estimate does not provide a tensor gain, i.e., if  $d \geq \max\{M_1, \dots, M_R\}$ .

### 3.15.4.1.6 Generalized ESPRIT (GESPRIT)

The generalized ESPRIT approach of [93] has been originally formulated for the array model composed of two  $\frac{M}{2}$ -sensor non-overlapping subarrays with pairwise sensor calibration such that the displacement vectors  $\tilde{\eta}_m = \left[ x_m - x_{m+\frac{M}{2}}, y_m - y_{m+\frac{M}{2}} \right]^T$  for  $m = 1, \dots, \frac{M}{2}$  between the  $m$ th sensor in the first subarray and its corresponding sensor, i.e., the  $(m + \frac{M}{2})$ th sensor in the second subarray, is known.

It is shown in [93] that if  $d \leq \frac{M}{2}$ , then for any  $\frac{M}{2} \times d$  full-rank matrix  $\mathbf{W}_{\text{GESPRIT}}$ , the matrix  $\mathbf{W}_{\text{GESPRIT}}^H (\mathbf{U}_{s,2} - \Phi_p(\theta) \mathbf{U}_{s,1})$  drops rank where  $\mathbf{U}_{s,1}$  and  $\mathbf{U}_{s,2}$  are the  $\frac{M}{2} \times d$  submatrices from  $\mathbf{U}_s$  such that

$$\mathbf{U}_s = \begin{bmatrix} \mathbf{U}_{s,1} \\ \mathbf{U}_{s,2} \end{bmatrix} \quad (15.113)$$

and the  $\frac{M}{2} \times \frac{M}{2}$  diagonal matrix  $\Phi_p(\theta)$  contains the displacement-phase information between the sensor pairs and the  $\frac{M}{2}$  diagonal entries are defined as

$$[\Phi_p(\theta)]_{(m,m)} \triangleq e^{j(x_m - x_{m+\frac{M}{2}}) \sin \theta + (y_m - y_{m+\frac{M}{2}}) \cos \theta} \quad (15.114)$$

for  $m = 1, \dots, \frac{M}{2}$ . It can be noted that the generalized ESPRIT scheme makes also use of rank dropping criterion which was used for the RARE algorithm in [80]. The difference is that the generalized ESPRIT is a ESPRIT-like algorithm, whereas the RARE method is a MUSIC-like technique.

In the finite sample case, we usually replace  $\mathbf{U}_{s,1}$  and  $\mathbf{U}_{s,2}$  by their estimates given by  $\widehat{\mathbf{U}}_{s,1}$  and  $\widehat{\mathbf{U}}_{s,2}$ , respectively. This leads to the following generalized ESPRIT spectrum, for  $\mathbf{W}_{\text{GESPRIT}} = \widehat{\mathbf{U}}_{s,1}$  [93]

$$f_{\text{GES1}}(\theta) = \frac{1}{|\det\{\widehat{\mathbf{U}}_{s,1}^H \widehat{\mathbf{U}}_{s,2} - \widehat{\mathbf{U}}_{s,1}^H \Phi(\theta) \widehat{\mathbf{U}}_{s,1}\}|}, \quad (15.115)$$

where the signal DOAs are estimated from the  $d$  highest peaks of (15.115).

Another meaningful choice of  $\mathbf{W}_{\text{GESPRIT}}$  is  $\mathbf{W}_{\text{GESPRIT}} = \mathbf{U}_{s,2} - \Phi_p(\theta) \mathbf{U}_{s,1}$  [94]. With the latter choice, the generalized ESPRIT spectral function becomes

$$f_{\text{GES2}}(\theta) = \frac{1}{|\det\{(\widehat{\mathbf{U}}_{s,2} - \Phi(\theta) \widehat{\mathbf{U}}_{s,1})^H (\widehat{\mathbf{U}}_{s,2} - \Phi(\theta) \widehat{\mathbf{U}}_{s,1})\}|}. \quad (15.116)$$

### 3.15.4.1.7 Method of direction of arrival estimation (MODE)

The MODE technique is in fact the rooting version of the weighted subspace fitting (WSF) method for the ULAs. The cost function of the WSF technique which has to be minimized can be shown to be [83]

$$f_{\text{MODE}}(\theta) = \text{Tr} \left( \mathbf{P}_{A(\theta)}^\perp \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \right), \quad (15.117)$$

where

$$\mathbf{P}_{A(\theta)}^\perp = \mathbf{I}_M - A(\theta)(A^H(\theta)A(\theta))^{-1}A^H(\theta) \quad (15.118)$$

indicates the orthogonal projection matrix of the array steering matrix,

$$\mathbf{W}_{\text{MODE}} = (\widehat{\mathbf{\Lambda}}_s - \hat{\sigma}^2 \mathbf{I}_d) \widehat{\mathbf{\Lambda}}_s^{-1} \quad (15.119)$$

is the asymptotic-optimum weight matrix, and

$$\hat{\sigma}^2 = \frac{1}{M-d} \text{Tr}(\widehat{\mathbf{\Lambda}}_n) \quad (15.120)$$

---

#### Algorithm 7 [93]. Summary of the generalized ESPRIT scheme

1. Compute the eigendecomposition of  $\widehat{\mathbf{\Sigma}}_x$  and obtain the matrix  $\widehat{\mathbf{U}}_{s,1}$  and  $\widehat{\mathbf{U}}_{s,2}$ .
  2. Depending on your choice of  $\mathbf{W}_{\text{GESPRIT}}$ , find the  $d$  maxima of  $f_{\text{GESPRIT}}(\theta)$  in (15.115) or  $f_{\text{GES2}}(\theta)$  in (15.116) by scanning the entire FOV.
-

denotes the estimated power of the noise. The objective in MODE is to reformulate the function  $f_{\text{MODE}}$  in (15.117) so that the multidimensional minimization becomes less computationally costly. Let us define a polynomial of degree  $d$  which has the spatial frequencies  $z_1, \dots, z_d$  as its roots such that

$$b(z) = b_0 z^d + b_1 z^{d-1} + \dots + b_d = b_0 \prod_{l=1}^d (z - z_l). \quad (15.121)$$

Defining the highly-structured matrix  $\mathbf{B}$

$$\mathbf{B}^H = \begin{bmatrix} b_d & \cdots & b_1 & b_0 & \cdots & 0 \\ \ddots & \ddots & & & \ddots & \\ 0 & & b_d & \cdots & b_1 & b_0 \end{bmatrix} \quad (15.122)$$

it can be clearly seen that

$$\mathbf{B}^H \mathbf{A}(\boldsymbol{\theta}) = \mathbf{0}. \quad (15.123)$$

Then,  $f_{\text{MODE}}(\boldsymbol{\theta})$  in (15.117) can be reformulated as

$$f_{\text{MODE}}(\mathbf{b}) = \text{Tr} \left( \mathbf{P}_B \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \mathbf{B} \right), \quad (15.124)$$

where

$$\mathbf{b} = [b_0 \ b_1 \ \cdots \ b_d]^T \quad (15.125)$$

and

$$\mathbf{P}_B = \mathbf{B}(\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H. \quad (15.126)$$

#### **Algorithm 8.** Solving the MODE function

1. Obtain the initial estimate  $\hat{\mathbf{b}}^{(0)}$  of  $\mathbf{b}$  from the quadratic function

$$\begin{aligned} f_{\text{MODE}}^{(0)}(\mathbf{b}) &= \text{Tr} \left( \mathbf{B}^H \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \mathbf{B} \right) \\ \text{s.t. } \text{Re}(b_0) &= 1, \quad b_k = b_{d-k}^*, \quad k = 0, 1, \dots, d \end{aligned} \quad (15.127)$$

and form  $\widehat{\mathbf{B}}$  from  $\hat{\mathbf{b}}^{(0)}$ .

2. Solve the following quadratic function

$$\begin{aligned} f_{\text{MODE}}^{(1)}(\mathbf{b}) &= \text{Tr} \left( (\widehat{\mathbf{B}} \widehat{\mathbf{B}})^{-1} \mathbf{B}^H \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \mathbf{B} \right) \\ \text{s.t. } \text{Re}(b_0) &= 1, \quad b_k = b_{d-k}^*, \quad k = 0, 1, \dots, d. \end{aligned} \quad (15.128)$$

to obtain  $\hat{\mathbf{b}}$  and root the polynomial with coefficients  $\hat{\mathbf{b}}$  to estimate DOAs.

Consequently, the MODE algorithm is performed by computing the coefficient vector  $\mathbf{b}$  which minimizes  $f_{\text{MODE}}(\mathbf{b})$  in (15.124). The DOAs can then be estimated from the roots of the polynomial

containing  $\mathbf{b}$  as its coefficients. In order to guarantee that the roots of the MODE polynomial exhibit the unit norm property the process of minimizing the function  $f_{\text{MODE}}(\mathbf{b})$ , has to take into account the conjugate-symmetric constraint w.r.t. the polynomial coefficients. Further, a norm constraint on  $\mathbf{b}$ , e.g.,  $\|\mathbf{b}\|^2 = 1$ , needs to be taken into account to remove the trivial solution [95,96]. Another possible normalization of  $\mathbf{b}$  is described in Algorithm 8 where the real part (or the imaginary part) of  $b_0$  is fixed to 1.

In contrast to other subspace-based methods like MUSIC, MODE achieves statistical efficiency for both uncorrelated and highly correlated sources through a search-free algorithm based on closed-form quadratic solutions and polynomial rooting.

### 3.15.4.1.8 Rank-reduction (RARE) DOA estimation method

The RARE technique has been developed in [80,97,98] for the case of sensor arrays consisting of multiple fully-calibrated subarrays without any calibration information in-between subarrays. Let us assume that the array composed of  $K$  subarrays and the unknown array geometry-dependent parameter  $\boldsymbol{\eta}$  consists of the displacement vectors between the subarrays. For this class of arrays, the columns of the array manifold matrix can be described as

$$\mathbf{a}(\theta_l, \boldsymbol{\eta}) = \mathbf{L}_R(\theta_l) \mathbf{h}_R(\theta_l, \boldsymbol{\eta}), \quad (15.129)$$

where the  $M \times K$  matrix  $\mathbf{L}_R(\theta)$  is defined as

$$\mathbf{L}_R(\theta_l) \triangleq \begin{bmatrix} \mathbf{a}_1(\theta_l) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{a}_2(\theta_l) & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{a}_K(\theta_l) \end{bmatrix} \quad (15.130)$$

for  $\theta_l = \theta_1, \dots, \theta_d, \mathbf{a}_k(\theta_l)$  for  $k = 1, \dots, K$  is the  $l$ th column of the manifold matrix for the  $k$ th subarray such that the first sensor of that subarray is considered as the origin (or reference) sensor, c.f. Figure 15.11, and the  $K \times 1$  vector  $\mathbf{h}(\theta_l, \boldsymbol{\eta})$  contains the phase information resulting from the uncalibrated or unknown part of the array such as intersubarray displacement vectors  $\boldsymbol{\eta}_k = [\alpha_k \beta_k]^T$  for  $k = 2, \dots, K$  such that

$$\mathbf{h}_R(\theta_l, \boldsymbol{\eta}) = [1 \ \phi_{2,l} \ \cdots \ \phi_{K,l}]^T, \quad (15.131)$$

where

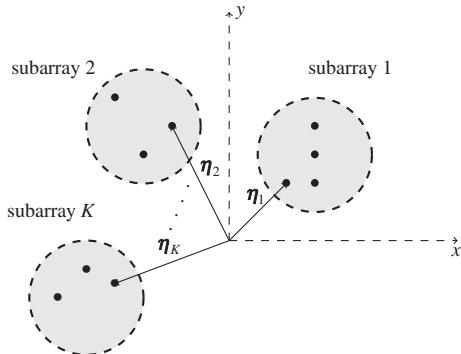
$$\phi_k(\theta_l) \triangleq e^{j(2\pi/\lambda)(\alpha_k \sin \theta_l + \beta_k \cos \theta_l)} \quad (15.132)$$

for  $k = 2, \dots, K$  and  $l = 1, \dots, d$ . Note, that  $\mathbf{L}_R(\theta)$  is solely dependent on the DOAs and the known or calibrated part of the array. The MUSIC criterion in Section 3.15.4.1.1, i.e., the orthogonality of the eigenvector matrix of the noise subspace and the array manifold matrix (15.87), can then be used

$$\begin{aligned} \mathbf{a}^H(\theta_l, \boldsymbol{\eta}) \mathbf{U}_n \mathbf{U}_n^H \mathbf{a}(\theta_l, \boldsymbol{\eta}) &= \\ \mathbf{h}_R^H(\theta_l, \boldsymbol{\eta}) \mathbf{L}_R(\theta_l) \mathbf{U}_n \mathbf{U}_n^H \mathbf{L}_R^H(\theta_l) \mathbf{h}_R(\theta_l, \boldsymbol{\eta}) &= \\ \mathbf{h}_R^H(\theta_l, \boldsymbol{\eta}) \mathbf{F}_{\text{RARE}}(\theta_l) \mathbf{h}_R(\theta_l, \boldsymbol{\eta}) &= 0, \end{aligned} \quad (15.133)$$

where

$$\mathbf{F}_{\text{RARE}}(\theta) \triangleq \mathbf{L}_R(\theta) \mathbf{U}_n \mathbf{U}_n^H \mathbf{L}_R^H(\theta). \quad (15.134)$$

**FIGURE 15.11**

$K$  arbitrary known subarrays with arbitrary unknown displacements.

The idea in the RARE algorithm is based on the observation that if  $K \leq M - d$  and taking into the account that  $\text{rank}\{\mathbf{U}_n\} \geq K$ , then Eq. (15.133) is true only when the  $K \times K$  matrix  $\mathbf{F}_{\text{RARE}}(\theta_l)$  drops rank, i.e., when  $\text{rank}\{\mathbf{F}_{\text{RARE}}(\theta_l)\} < K$ . In the finite sample case, however, the  $K \times K$  matrix  $\widehat{\mathbf{F}}_{\text{RARE}}(\theta)$  is given by

$$\widehat{\mathbf{F}}_{\text{RARE}}(\theta) \triangleq \mathbf{L}_R(\theta) \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}_n^H \mathbf{L}_R^H(\theta). \quad (15.135)$$

Then, in order to estimate the DOAs, the  $d$  maxima of the following function in the entire FOV must be found

$$f_{\text{RARE}}(\theta) = \frac{1}{|\det\{\widehat{\mathbf{F}}_{\text{RARE}}(\theta)\}|}. \quad (15.136)$$

It should be remarked that the spectral-RARE function can be expressed in other ways as well which yields approximately the same DOA estimation performance (see [80]). It should be remarked that defining the constant diagonal matrix

$$\boldsymbol{\Omega} = \mathbf{L}_R(\theta)^H \mathbf{L}_R(\theta) \quad (15.137)$$

and the scalar  $\Omega_r = \det\{\boldsymbol{\Omega}\}$  and applying Schur's complement, the alternative RARE matrix

$$\mathbf{F}_{\text{RARE}}(\theta) \triangleq \Omega_r \left( \mathbf{I}_d - \mathbf{U}_s^T \mathbf{L}_R^H(\theta) \boldsymbol{\Omega}^{-1} \mathbf{L}_R(\theta) \mathbf{U}_s \right) \quad (15.138)$$

and for the finite sample case

$$\widehat{\mathbf{F}}_{\text{RARE}}(\theta) \triangleq \Omega_r \left( \mathbf{I}_d - \widehat{\mathbf{U}}_s^T \mathbf{L}_R^H(\theta) \boldsymbol{\Omega}^{-1} \mathbf{L}_R(\theta) \widehat{\mathbf{U}}_s \right) \quad (15.139)$$

is obtained that exhibits the same rank properties as the RARE matrix in (15.134) in the sense, that it yields the same determinant function for all values of  $\theta$ . The same statement also holds true if the true signal subspace eigenvector matrix  $\mathbf{U}_s$  in (15.138) is replaced by the corresponding finite sample estimates  $\widehat{\mathbf{U}}_s$ . The RARE matrix  $\mathbf{F}_{\text{RARE}}(\theta)$  in (15.138) is of dimension  $d \times d$ , whereas the dimension of  $\mathbf{F}_{\text{RARE}}(\theta)$  in (15.134) is of size  $K \times K$ . Hence, in the case where the number of subarrays  $K$  exceeds

the number of sources  $d$  the use of alternative RARE matrix in the evaluation of the determinant is preferable from a computational point of view.

For some special geometry of the array where the PCA is composed of identically oriented uniform subarrays, a search-free RARE algorithm, known as root-RARE, is also developed [80] which is described in the next subsection.

---

**Algorithm 9.** Summary of the RARE technique
 

---

1. Compute the eigendecomposition of  $\widehat{\Sigma}_x$  and obtain the matrix  $\widehat{U}_n$  if  $d \leq K$ , and  $\widehat{U}_s$  if  $K \leq d$ .
  2. If  $d \leq K$ , then find the  $d$  maxima of  $f_{\text{RARE}}(\theta)$  in (15.136) by scanning the entire FOV.
  3. Else, use the alternative definition of the RARE function given in (15.138) and find the  $d$  maxima of  $f_{\text{RARE}}(\theta)$  by scanning the entire FOV.
- 

### 3.15.4.1.9 Root-RARE

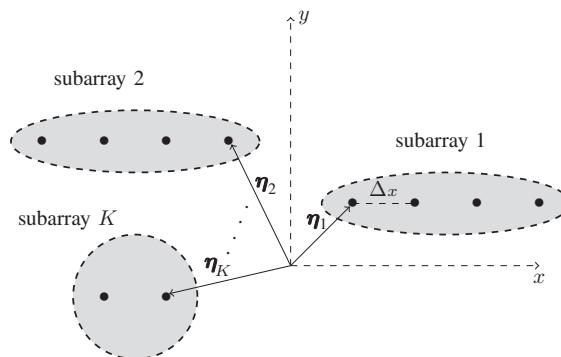
Let us consider a PCA composed of  $K$  identically oriented uniform subarrays, in which, the  $k$ th subarray consists of  $m_k$  sensors. Without loss of generality, we assume that all these subarrays are parallel to the  $x$ -axis. Consequently, the cartesian coordinates of the  $n$ th sensor belonging to the  $k$ th subarray is given by  $(\alpha_k + \kappa_n \Delta_x, \beta_k)$ , where  $(\alpha_k, \beta_k)$  denotes the unknown coordinate of the first sensor belonging to the  $k$ th subarray and  $\kappa_n$  is the integer multiple of the common baseline  $\Delta_x$ , which determines the location of the  $n$ th sensor, c.f. Figure 15.12.

Then, the  $m_k \times 1$  steering vector of the  $k$ th subarray is expressed as

$$\mathbf{a}_k(\theta, \boldsymbol{\eta}_k) = \mathbf{b}_k(z) e^{j \frac{2\pi}{\lambda} (\alpha_k \sin(\theta) + \beta_k \cos(\theta))} \quad (15.140)$$

in which  $\boldsymbol{\eta}_k = [\alpha_k \ \beta_k]^T$  and

$$\mathbf{b}_k(z) = [1 \ z^{\kappa_2} \ \dots \ z^{\kappa_{m_k}}]^T, \quad (15.141)$$



**FIGURE 15.12**

$K$  arbitrary known uniform and linear subarrays with arbitrary unknown displacements.

where  $z = e^{j \frac{2\pi}{\lambda} \Delta_x \cos(\theta)}$ . From (15.140), one can deduce the steering vector expression of the whole array:

$$\mathbf{a}(\theta, \boldsymbol{\eta}) = \begin{bmatrix} \mathbf{a}_1(\theta)^T & \cdots & \mathbf{a}_K(\theta)^T \end{bmatrix}^T = \mathbf{L}_{\text{rR}}(z) \mathbf{h}_{\text{rR}}(\theta, \boldsymbol{\eta}) \quad (15.142)$$

in which  $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^T \cdots \boldsymbol{\eta}_K^T]^T$  and

$$\mathbf{h}(\theta, \boldsymbol{\eta}) = \begin{bmatrix} e^{j \frac{2\pi}{\lambda} (\alpha_1 \sin(\theta) + \beta_1 \cos(\theta))} & \cdots & e^{j \frac{2\pi}{\lambda} (\alpha_K \sin(\theta) + \beta_K \cos(\theta))} \end{bmatrix}^T \quad (15.143)$$

and

$$\mathbf{L}_{\text{rR}}(z) = \begin{bmatrix} \mathbf{b}_1(z) & \mathbf{0}_{m_1 \times 1} & \cdots & \mathbf{0}_{m_1 \times 1} \\ \mathbf{0}_{m_2 \times 1} & \mathbf{b}_2(z) & \cdots & \mathbf{0}_{m_2 \times 1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{m_K \times 1} & \mathbf{0}_{m_K \times 1} & & \mathbf{b}_K(z) \end{bmatrix}. \quad (15.144)$$

Since the noise subspace is orthogonal to the steering vectors of the true DOAs, one obtains

$$\mathbf{h}_{\text{rR}}^H(\theta_l, \boldsymbol{\eta}) \mathbf{L}_{\text{rR}}^H(z_l) \mathbf{U}_n \mathbf{U}_n^H \mathbf{B}_{\text{rR}}(z_l) \mathbf{h}_{\text{rR}}(\theta_l, \boldsymbol{\eta}) = 0 \quad \text{for } l = 1, \dots, d. \quad (15.145)$$

From (15.145), one can notice that  $\mathbf{B}_{\text{rR}}^H(z) \mathbf{U}_n \mathbf{U}_n^H \mathbf{L}_{\text{rR}}(z)$  is rank deficient for  $z = z_l$  (i.e.,  $\theta = \theta_l$ ),  $l = 1, \dots, d$ . Consequently, the rooting polynomial of the so-called root-RARE scheme, in the finite sample case, is defined as follows:

$$f_{\text{root-RARE}} = \det \left( \mathbf{L}_{\text{rR}}^H(z) \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}^H \mathbf{L}_{\text{rR}}(z) \right) = \det \left( \mathbf{L}_{\text{rR}}^T(1/z) \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}^H \mathbf{L}_{\text{rR}}(z) \right). \quad (15.146)$$

Similar to the root-MUSIC, the roots of  $f_{\text{root-RARE}}$  form conjugate reciprocal pairs and the DOAs can be estimated using the largest magnitude roots which are inside the unit circle.

#### **Algorithm 10.** Summary of the root-RARE scheme

1. Compute the eigendecomposition of  $\widehat{\Sigma}_x$  and obtain the matrix  $\widehat{\mathbf{U}}_n$ .
2. Root the polynomial  $\det \left( \mathbf{L}_{\text{rR}}^T(1/z) \widehat{\mathbf{U}}_n \widehat{\mathbf{U}}^H \mathbf{L}_{\text{rR}}(z) \right)$ .
3. Find the  $d$  DOA estimates using the largest magnitude roots which lie inside the unit circle.

#### **3.15.4.1.10 Interpolated root-MUSIC**

Interpolated scheme uses the principal of a virtual array in order to obtain a search-free algorithm based on polynomial rooting. More precisely, the main idea is to approximate the true non uniform array by a virtual ULA using an interpolation matrix  $\mathbf{V}$ . The true steering vector is then given by the following approximation [82]:

$$\mathbf{a}(\theta) \simeq \mathbf{V} \mathbf{a}_{\text{v}}(z) \quad (15.147)$$

in which the  $M_V \times 1$  vector  $\mathbf{a}_V(z)$  denotes the steering vector of the new virtual ULA. The interpolation matrix  $V$  of dimension  $M \times M_V$  is designed to reduce the interpolation error. Plugging (15.147) into the MUSIC pseudo null-spectrum function, one obtains the polynomial  $f_V(z)$  of degree  $2M_V - 2$ :

$$f_{\text{N-MUSIC}}(\theta) \simeq f_V(z) = \mathbf{a}_V^T \left( \frac{1}{z} \right) V^H \hat{U}_n^H \hat{U}_n V \mathbf{a}_V(z). \quad (15.148)$$

Finally, as in the root-MUSIC scheme, the DOAs are found from the largest-magnitude roots of the polynomial  $f_V(z)$  that are located inside the unit circle. For more details on the interpolated root-MUSIC scheme, the reader can refer to the chapter entitled *Array Processing in the face of Nonidealities* by Costa et al.

#### Algorithm 11. Designing the interpolated array: an off line procedure

1. Divide the field of view into  $S$  regions and define a set of angles for each  $s$ th sector:

$$\Theta = [\theta_{s,\text{begin}} \quad \theta_{s,\text{begin}} + \Delta_\theta \quad \dots \quad \theta_{s,\text{end}}], \quad (15.149)$$

where the  $s$ th sector is delimited by  $\theta_{s,\text{begin}}$  and  $\theta_{s,\text{end}}$ .

2. Decide where to place the virtual elements of the new interpolated array corresponding to the  $s$ th sector. The steering matrix of the real array and the virtual array associated with the  $s$ th section are given respectively, by  $A_s = [\mathbf{a}(\theta_{s,\text{begin}}) \dots \mathbf{a}(\theta_{s,\text{end}})]$  and  $A_{V,s} = [\mathbf{a}_V(\theta_{s,\text{begin}}) \dots \mathbf{a}_V(\theta_{s,\text{end}})]$ .
3. Compute the matrix  $V_s$  as the Least Square solution of

$$\tilde{V}_s A_s = A_{V,s}, \quad (15.150)$$

which can be given by

$$A_s = (\tilde{V}_s^H \tilde{V}_s)^{-1} \tilde{V}_s^H A_{V,s} \quad (15.151)$$

thus, one obtains a set of an interpolation matrices. If the Frobenius norm of  $A_{V,s} - V_s A_s$  is not sufficiently small compared to the norm of  $A_{V,s}$ , then, come back to the first step and reduce the  $s$ th sector's size.

#### 3.15.4.1.11 Manifold separation

In [85] Doron and Doron proposed the so-called manifold separation scheme which can be considered as an other root-MUSIC like approach for arbitrary array geometry. The manifold separation technique approximates the true steering vector by

$$\mathbf{a}(\theta) \simeq V_{\text{MS}} \mathbf{a}_{\text{MS}}(\check{z}) \quad (15.152)$$

in which  $\check{z} = e^{j\theta}$  and where the  $M_{\text{MS}} \times 1$  Vandermonde  $\mathbf{a}_{\text{MS}}(\check{z})$  depends only on  $\check{z}$  such that

$$\mathbf{a}_{\text{MS}}(\check{z}) = [\check{z}^0 \quad \dots \quad \check{z}^{(M_{\text{MS}}-1)}]^T \quad (15.153)$$

in which the  $M \times M_{\text{MS}}$  matrix  $\mathbf{V}_{\text{MS}}$ , depends only on the array parameters. Consequently, the MUSIC pseudo null-spectrum can be approximated by

$$f_{\text{N-MUSIC}}(\theta) \simeq f_{\text{MS}}(\check{z}) = \mathbf{a}_{\text{MS}}^T \left( \frac{1}{\check{z}} \right) \mathbf{V}_{\text{MS}}^H \hat{\mathbf{U}}_n \hat{\mathbf{U}}_n^H \mathbf{V}_{\text{MS}} \mathbf{a}_{\text{MS}}(\check{z}). \quad (15.154)$$

Finally, as in the root-MUSIC scheme, the DOAs are found from the largest-magnitude roots of the polynomial  $f_{\text{MS}}(\check{z})$  that are located inside the unit circle.

One can find a plethora of methods to design  $\mathbf{V}_{\text{MS}}$  based on the Least Square scheme or on the inverse discrete Fourier transform of  $\mathbf{a}(\theta)$  taken at different angles [84, 85]. It can be noted that the parameter  $M_{\text{MS}}$  represents the accuracy of the approximation given in (15.152). Increasing  $M_{\text{MS}}$  will improve the latter approximation which leads to a more accurate estimate. In [85], it has been suggested that the minimal value of  $M_{\text{MS}}$  leading to an acceptable DOA estimation is equal or greater than  $8\pi \frac{q}{\lambda}$  where  $q$  is given as the largest distance between the origin of coordinate system and the array sensors. For more details on the manifold separation scheme, the reader can refer to the chapter entitled *Array Processing in the face of Nonidealities* by Costa et al.

### 3.15.4.1.12 Fourier domain root-MUSIC

Noting that the MUSIC null-spectrum function,  $f_{\text{N-MUSIC}}(\theta)$ , is periodic w.r.t. the DOA with the period  $2\pi$ , in [86] the authors introduced the so-called Fourier domain root-MUSIC. More precisely,  $f_{\text{N-MUSIC}}(\theta)$  can be written using the Fourier series expansion

$$f_{\text{N-MUSIC}}(\theta) = \sum_{m=-\infty}^{+\infty} \mathcal{F}_m e^{jm\theta} \quad (15.155)$$

in which the Fourier coefficients are given by

$$\mathcal{F}_m = \frac{1}{2\pi} \int_{-\pi}^{\pi} f_{\text{N-MUSIC}}(\theta) e^{-jm\theta} d\theta. \quad (15.156)$$

Consequently, using only  $\mathcal{M}_{\text{FD}} = 2M_{\text{FD}} - 1$  points in (15.155), one obtains  $\hat{f}_{\text{N-MUSIC}}(\theta)$ , the approximation of  $f_{\text{N-MUSIC}}(\theta)$ , given by

$$\hat{f}_{\text{N-MUSIC}}(\theta) = \sum_{m=-M_{\text{FD}}+1}^{M_{\text{FD}}-1} \mathcal{F}_m \check{z}^m \quad (15.157)$$

in which  $\check{z} = e^{jm\theta}$ . In practice, the Fourier coefficients  $\mathcal{F}_m$  are approximated by  $\hat{\mathcal{F}}_m$  using the discrete Fourier transform

$$\hat{\mathcal{F}}_m = \frac{1}{\mathcal{M}_{\text{FD}}} \sum_{m'=-M_{\text{FD}}+1}^{M_{\text{FD}}-1} f_{\text{N-MUSIC}} \left( \frac{2\pi m'}{\mathcal{M}_{\text{FD}}} \right) e^{-j2\pi \frac{m'm}{\mathcal{M}_{\text{FD}}}}. \quad (15.158)$$

Thus, similar to the way of root-MUSIC, the estimated DOAs can be selected from the largest magnitude roots of

$$\hat{f}_{FD}(\check{z}) = \sum_{m=-M_{FD}+1}^{M_{FD}-1} \hat{\mathcal{F}}_m \check{z}^m. \quad (15.159)$$

One can observe that  $\hat{f}_{FD}(\check{z})$  can be negative. Thus, sign changes of  $\hat{f}_{FD}(\check{z})$  means that there are two roots that lie on the unit circle closely to each other. Furthermore, it has been proved in [86] that the roots of  $\hat{f}_{FD}(\check{z})$  satisfy the conjugate reciprocity property, meaning that the corresponding roots for which  $\hat{f}_{FD}(\check{z})$  is negative, do not form a conjugate reciprocal pair. Consequently, one can obtain two distinct groups. The first one contains pairs of roots lying exactly at the unit circle. The second one contains the roots which form conjugate reciprocal and do not belong to the unit circle. Based on this discuss, the proposed algorithm of [86] is explained in the algorithm summary box 12.

---

**Algorithm 12** [86]. Fourier domain root-MUSIC

---

1. Select the closest root from the unit circle.
  2. Identify the class group of the computed root by checking whether its conjugate reciprocal value is another root.
  3. Use the latter root to estimate the DOA if it belongs to the second group, then drop both this root and its conjugate reciprocal pair. Then, go to step 5.
  4. Else, estimate the source DOA from the average of this root and its closest neighbor, and drop both these roots.
  5. Repeat steps 1–4 until  $d$  DOAs will be estimated.
- 

### 3.15.4.2 Multi-dimensional algorithms using matrix-based subspace estimates

#### 3.15.4.2.1 R-D Standard ESPRIT

To solve the  $R$ - $D$  shift invariance Eq. (15.43) for the matrices  $\Phi^{(r)}$ , we need to eliminate the unknown array steering matrix  $A$ . This is achieved by observing that the column space of  $A$  and  $U_s$  agree and hence we can write  $A = U_s \cdot P$  for a non-singular square matrix  $P$  (cf. (15.58)). In practice, we estimate  $U_s$  via an SVD of the noisy measurements  $X$ . The estimate  $\widehat{U}_s$  satisfies  $A \approx \widehat{U}_s \cdot P$ . Inserting this relation into (15.43), we have

$$\begin{aligned} \widetilde{J}_1^{(r)} \cdot \widehat{U}_s \cdot P \cdot \Phi^{(r)} &\approx \widetilde{J}_2^{(r)} \cdot \widehat{U}_s \cdot P, \\ \widetilde{J}_1^{(r)} \cdot \widehat{U}_s \cdot \underbrace{P \cdot \Phi^{(r)} \cdot P^{-1}}_{\Psi^{(r)}} &\approx \widetilde{J}_2^{(r)} \cdot \widehat{U}_s, \end{aligned} \quad (15.160)$$

which is an overdetermined set of equations for  $\Psi^{(r)}$ . An unstructured “Least Squares” solution of (15.160) for  $\Psi^{(r)}$  is obtained by

$$\widehat{\Psi}_{LS}^{(r)} = \arg \min_{\Psi} \left\| \widetilde{J}_1^{(r)} \cdot \widehat{U}_s \cdot \Psi - \widetilde{J}_2^{(r)} \cdot \widehat{U}_s \right\|_F^2 = \left( \widetilde{J}_1^{(r)} \cdot \widehat{U}_s \right)^+ \cdot \widetilde{J}_2^{(r)} \cdot \widehat{U}_s. \quad (15.161)$$

Since  $\Psi^{(r)} = P \cdot \Phi^{(r)} \cdot P^{-1}$  represents an EVD, we obtain an estimate of  $\Phi^{(r)}$  via an EVD of  $\widehat{\Psi}_{\text{LS}}^{(r)}$ . To ensure the correct pairing across the dimensions, the matrices  $\widehat{\Phi}^{(r)}$  should be estimated via a *joint* EVD of  $\widehat{\Psi}_{\text{LS}}^{(r)}$  (e.g., via [99]). Algorithm 13 summarized the *R-D* Standard ESPRIT procedure.

---

**Algorithm 13.** Summary of *R-D* Standard ESPRIT using Least Squares
 

---

1. Estimate the signal subspace  $\widehat{U}_s$  via the truncated SVD of the observation matrix  $X \in \mathcal{C}^{M \times N}$ .
  2. Solve the overdetermined shift invariance equations  $\widetilde{J}_1^{(r)} \cdot \widehat{U}_s \cdot \Psi^{(r)} \approx \widetilde{J}_2^{(r)} \cdot \widehat{U}_s$  for the matrices  $\Psi^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS).
  3. Compute the eigenvalues  $\widehat{\lambda}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\widehat{\Psi}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99]. Recover the correctly paired frequencies  $\widehat{\mu}_i^{(r)}$  via  $\widehat{\mu}_i^{(r)} = \arg(\widehat{\lambda}_i^{(r)})$ .
- 

### 3.15.4.2.2 *R-D Unitary ESPRIT*

*R-D* Unitary ESPRIT can be applied if the array is centro-symmetric, i.e., its structure is invariant under mirroring around one centroid (cf. Section 3.15.2.3.5). If the array is centro-symmetric we can exploit the fact that  $A$  and  $\Pi_M \cdot A^*$  span the same column space. Therefore, the measurements  $X \in \mathcal{C}^{M \times N}$  can be augmented by  $\Pi_M \cdot X^*$  along the columns without changing the column space. This creates another set of  $N$  “virtual snapshots” (cf. Section 3.15.3.3). Moreover, under certain conditions, this step allows to decorrelate two coherent sources.<sup>10</sup> Finally, the redundancies in the resulting augmented measurement matrix can be used to transform the complex-valued measurement in the real-valued domain and perform the entire processing using real-valued additions and multiplications only. The details of the derivation are found in [15, 21]. Here we only provide a summary in Algorithm 14.

### 3.15.4.2.3 *R-D RARE*

In this section we extend the RARE algorithm of Section 3.15.4.1.9 that has originally been proposed for 1-D DOA estimation in partly calibrated arrays to DOA estimation in *R-D* array structures [80]. The basic idea of the multidimensional extension of the RARE algorithm is to estimate the parameters along the different baselines separately using the 1-D RARE algorithm. Therefore, the *R-D* array is considered as subarray system composed of multiple shifted and identically oriented ULAs for which the RARE algorithm can be applied to estimate the spatial frequencies along a ULA baseline. The described estimation procedure can then be applied to estimate the DOA parameters along all remaining baselines of the *R-D* structure. Such a separate estimation procedure is attractive from a computational viewpoint, as it allows to decompose a  $R$  dimensional estimation problem into  $R$  one dimensional rooting problems. However, if the number of sources is large, the overhead required for associating the various DOA parameter estimates obtained separately along the different baselines of the array to the

---

<sup>10</sup>The decorrelation relies on phase offsets between the sources and hence there are pathological cases where it fails, e.g., sources arriving in-phase (which means that their complex correlation coefficient is equal to 1 or  $-1$ ) at an array where the phase reference is chosen in the center.

---

**Algorithm 14** [21]. Summary of R-D Unitary ESPRIT using Least Squares

1. Estimate the real-valued signal subspace  $\widehat{\mathbf{E}}_s$  via the truncated SVD of the transformed real-valued observation matrix  $\mathcal{T}(X) = \mathbf{Q}_M^H \cdot [X \quad \mathbf{\Pi}_M X^* \mathbf{\Pi}_N] \cdot \mathbf{Q}_{2N} \in \mathbb{R}^{M \times 2N}$ , where  $\mathbf{Q}_p$  is a unitary  $p \times p$  left- $\mathbf{\Pi}$ -real matrix (i.e.,  $\mathbf{\Pi}_p \cdot \mathbf{Q}_p^* = \mathbf{Q}_p$ ).
2. Solve the overdetermined shift invariance equations

$$\widetilde{\mathbf{K}}_1^{(r)} \cdot \widehat{\mathbf{E}}_s \cdot \mathbf{\Upsilon}^{(r)} \approx \widetilde{\mathbf{K}}_2^{(r)} \cdot \widehat{\mathbf{E}}_s \quad (15.162)$$

for the matrices  $\mathbf{\Upsilon}^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS), where  $\widetilde{\mathbf{K}}_1^{(r)}$  and  $\widetilde{\mathbf{K}}_2^{(r)}$  are the transformed selection matrices given by

$$\widetilde{\mathbf{K}}_1^{(r)} = 2 \cdot \text{Re} \left( \mathbf{Q}_{M_r^{(\text{sel})} \cdot M/M_r}^H \cdot \widetilde{\mathbf{J}}_2^{(r)} \cdot \mathbf{Q}_M \right), \quad (15.163)$$

$$\widetilde{\mathbf{K}}_2^{(r)} = 2 \cdot \text{Im} \left( \mathbf{Q}_{M_r^{(\text{sel})} \cdot M/M_r}^H \cdot \widetilde{\mathbf{J}}_2^{(r)} \cdot \mathbf{Q}_M \right). \quad (15.164)$$

3. Compute the eigenvalues  $\hat{\omega}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\widehat{\mathbf{\Upsilon}}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99] or via the Simultaneous Schur Decomposition proposed in [21]. Recover the correctly paired frequencies  $\hat{\mu}_i^{(r)}$  via  $\hat{\mu}_i^{(r)} = 2 \cdot \arctan(\hat{\omega}_i^{(r)})$ .
- 

individual sources can be significant. Exploiting the rich nullspace structure of the RARE polynomial matrices the R-D RARE algorithm has been developed that entirely avoids the computationally complex combinatorial parameter association procedure and further enhances the resolution performance of the 1-D RARE estimates [24, 97].

In order to derive the multidimensional extension of the root-RARE algorithm, consider the R-D Kronecker steering vector model (15.33), which for  $k = 1, \dots, R$  and in accordance to (15.129) can also be expressed as

$$\mathbf{a} \left( \mu_l^{(1)}, \dots, \mu_l^{(R)} \right) = \mathbf{a} \left( \mu_l^{(1)} \right) \otimes \dots \otimes \mathbf{a} \left( \mu_l^{(R)} \right) = \mathbf{K}_{R,r} \left( \mu_l^{(r)} \right) \mathbf{h}_R \left( \left\{ \mu_l^{(k)} \right\}_{k \neq r}^R \right), \quad (15.165)$$

where  $\mathbf{a} \left( \mu^{(r)} \right)$  is the  $M_r \times 1$  steering vector along the baseline  $d$ ,

$$\mathbf{L}_{R,r} \left( \mu_l^{(r)} \right) \triangleq \mathbf{I}_{M/M_r} \otimes \mathbf{a} \left( \mu_l^{(r)} \right) \otimes \mathbf{I}_{M/M_r} \quad (15.166)$$

and

$$\mathbf{h}_R \left( \left\{ \mu_l^{(k)} \right\}_{k \neq r}^R \right) = \mathbf{a} \left( \mu_l^{(1)} \right) \otimes \dots \otimes \mathbf{a} \left( \mu_l^{(r-1)} \right) \otimes \mathbf{a} \left( \mu_l^{(r+1)} \right) \otimes \dots \otimes \mathbf{a} \left( \mu_l^{(R)} \right). \quad (15.167)$$

Defining  $z^{(r)} = e^{j\mu^{(r)}}$  and inserting (15.165) in (15.134) we obtain the RARE matrix

$$\mathbf{F}_{\text{RARE}}^{(r)} \left( z^{(r)} \right) \triangleq \mathbf{L}_{R,r}^T \left( 1/z^{(r)} \right) \mathbf{U}_n \mathbf{U}_n^H \mathbf{L}_{R,r} \left( z^{(r)} \right) \quad (15.168)$$

which represents a  $(M/M_r) \times (M/M_r)$  matrix polynomial of degree  $2M_r - 1$ . Alternatively we can also consider the RARE matrix as defined in (15.138), which yields the matrix polynomial of dimension  $d \times d$  as

$$\mathbf{F}_{\text{RARE}}^{(r)}(z^{(r)}) \triangleq M \left( \mathbf{I}_d - 1/M_r \mathbf{U}_s^T \mathbf{L}_{R,r}^T \left( 1/z^{(r)} \right) \mathbf{L}_{R,r} \left( z^{(r)} \right) \mathbf{U}_s \right), \quad (15.169)$$

where  $M$  is defined in (15.34). In the finite sample case, replacing the true signal and noise eigenvectors  $\mathbf{U}_s$  and  $\mathbf{U}_n$  by their respective estimates  $\widehat{\mathbf{U}}_s$  and  $\widehat{\mathbf{U}}_n$  in the matrix polynomials in (15.168) and (15.169) the parameter estimates of  $\mu_1^{(r)}, \dots, \mu_d^{(r)}$  along the  $r$ th array baseline can then be obtained from the  $d$  largest roots of the matrix polynomials inside the unit circle.

For the alternative matrix polynomial formulation in (15.169) an interesting property can be derived that interrelates the matrix polynomials composed along different baselines  $r = 1, \dots, R$ . In fact it was shown in [97], that with  $z_l^{(r)} = e^{j\mu_l^{(r)}}$  the matrix polynomials  $\mathbf{F}_{\text{RARE}}^{(r)}(z^{(r)})$  in (15.169) evaluated at  $z^{(r)} = z_l^{(r)}$  for  $r = 1, \dots, R$  yield matrices with intersecting subspaces. Hence a vector  $\mathbf{p}_l = \mathbf{p}_l^{(1)} = \dots = \mathbf{p}_l^{(R)}$  can be found such that

$$\mathbf{F}_{\text{RARE}}^{(1)}(z_l^{(1)}) \mathbf{p}_l^{(r)} = \dots = \mathbf{F}_{\text{RARE}}^{(R)}(z_l^{(R)}) \mathbf{p}_l^{(r)} = \mathbf{0}_{d \times 1}. \quad (15.170)$$

Furthermore, it can be proven that for unique signal roots  $z_l^{(r)}$  the nullspace vector  $\mathbf{p}_l^{(r)}$  is equivalent to the  $k$ th column of the mixing matrix in (15.58). The property (15.170) suggests a sophisticated parameter association procedure described in Algorithm 15.

---

**Algorithm 15.** Summary of the  $R$ -D RARE technique

---

1. Compute the  $d$  largest roots  $\hat{z}_l^{(r)}$  inside the unit circle from the RARE matrix polynomials in (15.169) with  $\mathbf{U}_s$  replaced by  $\widehat{\mathbf{U}}_s$  for  $r = 1, \dots, R$  to form the sets  $\mathcal{Z}^{(r)} = \{\hat{z}_l^{(r)}\}_{l=1}^d$ .
2. For each root  $\hat{z}_l^{(r)} \in \mathcal{Z}^{(r)}$  obtained in the previous step compute the unit norm nullspace vector  $\hat{\mathbf{p}}_l^{(r)}$  and form the unsorted set  $\mathcal{P}^{(r)} \triangleq \{\hat{\mathbf{p}}_1^{(r)}, \dots, \hat{\mathbf{p}}_d^{(r)}\}$ . Initialize the iteration counter with  $i = 1$ .
3. While  $i \leq d$ : Select a root  $\hat{z}_{s,i}^{(r')}$  and the baseline  $r'$  among the roots in all sets  $\mathcal{Z}^{(r)}$  that is located close to the unit-circle and that is well separated from the remaining roots along that baseline. Remove the root from the set  $\mathcal{Z}^{(r)}$  and select  $\hat{\mathbf{p}}_{s,i}$  as the corresponding normalized nullspace vector.
4. Determine the root  $\hat{z}_i^{(r)} \in \mathcal{Z}^{(r)}$  along all remaining baselines  $r \neq r'$  as

$$\hat{z}_{s,i}^{(r')} = \max_{\hat{z}_l^{(d)} \in \mathcal{Z}^{(d)}} \left( \hat{\mathbf{p}}_{s,i}^{(r')H} \mathbf{F}_{\text{RARE}}^{(d)} \left( \hat{z}_l^{(d)} \right) \hat{\mathbf{p}}_{s,i}^{(r')H} \right) / \left\| \mathbf{F}_{\text{RARE}}^{(d)} \left( \hat{z}_l^{(d)} \right) \right\|_F^2 \quad (15.171)$$

and remove the corresponding roots from the sets  $\mathcal{Z}^{(r)}$ . Set  $i = i + 1$ .

5. Obtain the  $R$ -D parameter estimates  $(\hat{\mu}_l^{(1)}, \dots, \hat{\mu}_l^{(R)}) = (\arg \{\hat{z}_{s,l}^{(1)}\}, \dots, \arg \{\hat{z}_{s,l}^{(R)}\})$  for  $l = 1, \dots, d$ .
-

The intersecting subspace property (15.170) and the estimated mixing matrix  $\hat{\mathbf{P}}_s \triangleq [\hat{\mathbf{p}}_{s,1}, \dots, \hat{\mathbf{p}}_{s,d}]$  computed in Algorithm 15 can further be exploited to enhance DOA estimation performance. Making use of property (15.58) we can obtain a simple estimate of the *R-D* steering matrix as

$$\hat{\mathbf{A}} = \hat{\mathbf{U}}_s \hat{\mathbf{P}}_s \quad (15.172)$$

from which refined source root and spatial frequency estimates can be computed in a straightforward manner.

#### 3.15.4.2.4 R-D MODE

The Multiple Invariance (MI) MODE algorithm proposed in [100] has been originally developed for DOA estimation in sensor arrays composed of multiple subarrays. In this section we introduce a simple extension of this algorithm to the case of DOA estimation in uniform *R-D* array structures. Consider the general *R-D* Kronecker steering vector model in (15.33). The steering matrix  $\mathbf{A}$  partitions as

$$\mathbf{A} \triangleq \left[ \mathbf{a}\left(\mu_1^{(1)}, \dots, \mu_1^{(R)}\right), \dots, \mathbf{a}\left(\mu_d^{(1)}, \dots, \mu_d^{(R)}\right) \right] = \mathbf{A}\left(\boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(R-1)}\right) \diamond \mathbf{A}\left(\boldsymbol{\mu}^{(R)}\right), \quad (15.173)$$

$$\mathbf{A}\left(\boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(R-1)}\right) \triangleq \mathbf{A}\left(\boldsymbol{\mu}^{(1)}\right) \diamond \dots \diamond \mathbf{A}\left(\boldsymbol{\mu}^{(R-1)}\right). \quad (15.174)$$

The general idea of the *R-D* MODE algorithm is similar to that of the conventional 1-D MODE algorithm. Hence, the weighted subspace fitting criteria (15.117) is minimized. Similarly to the 1-D case of Section 3.15.4.1.7 the noise subspace is modeled by a highly-structured sparse matrix  $\mathbf{G}$  such that

$$\mathbf{G}^H \mathbf{A} = \mathbf{0}_{(M-d) \times d}. \quad (15.175)$$

From the Khatri-Rao structure of the steering matrix in (15.173) it can readily be verified that  $\mathbf{G}$  can, e.g., be chosen as

$$\mathbf{G}^H \triangleq \begin{bmatrix} \mathbf{B}_{M_R}^H & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C}_1 & \mathbf{J}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_R^H & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C}_2 & \mathbf{0} & \mathbf{J}_1 & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{M_R}^H & \cdots & \mathbf{0} \\ \vdots & & & & \vdots \\ \mathbf{C}_{M/M_R} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{J}_1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{B}_{M_R}^H \end{bmatrix}, \quad (15.176)$$

$$\mathbf{C}_j \triangleq \begin{bmatrix} c_{d,j} & \cdots & c_{2,j} & c_{1,j} & \cdots & 0 \\ \ddots & \ddots & \ddots & & \ddots & \\ 0 & & c_{d,j} & \cdots & c_{2,j} & c_{1,j} \end{bmatrix} \quad (15.177)$$

with

$$\mathbf{c}_j \triangleq [c_{1,j}, c_{2,j}, \dots, c_{d,j}]^T, \quad (15.178)$$

$\mathbf{J}_1^{(r)} \triangleq [\mathbf{I}_d, \mathbf{0}_{d \times M/M_R}]$  and the  $M_R \times (M_R - d)$  full rank matrix  $\mathbf{B}_{M_R}$  defined in (15.122). We observe, that with (15.175) and (15.176) also  $\mathbf{B}_{M_R}^H \mathbf{A}(\boldsymbol{\mu}_R) = \mathbf{0}_{(M_R-d) \times d}$  such that (15.121) is satisfied. If we further choose

$$\mathbf{e}_1^T \mathbf{C}_j \mathbf{A}(\boldsymbol{\mu}_R) = \mathbf{1}_{1 \times d}$$

then

$$\mathbf{C}_j \mathbf{A}(\boldsymbol{\mu}_R) = -(\mathbf{J}_1 \mathbf{A}(\boldsymbol{\mu}_R)) \diamond (\mathbf{e}_{j+1}^T \mathbf{A}(\boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(R-1)})) \quad (15.179)$$

for  $j = 1, \dots, \prod_{r=1}^{R-1} M_r$ , where  $\mathbf{e}_j$  denotes the  $j$ th column of the identity matrix of conformable dimensions and  $\mathbf{1}_{1 \times d}$  is the  $1 \times d$  vector containing ones in all entries.

Replacing the orthogonal projector  $\mathbf{P}_A^\perp = \mathbf{I}_{M \times M} - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  in (15.117) with

$$\mathbf{P}_G \triangleq \mathbf{G}(\mathbf{G}^H \mathbf{G})^{-1} \mathbf{G}^H \quad (15.180)$$

yields

$$f_{\text{WSF}}(\mathbf{b}, \mathbf{c}_1, \dots, \mathbf{c}_{M/M_R}) = \text{Tr}(\mathbf{P}_G \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H), \quad (15.181)$$

where  $\mathbf{b}$  and  $\mathbf{W}_{\text{MODE}}$  defined in (15.125) and (15.119), respectively. The R-D MODE algorithm can then be carried out as summarized in Algorithm 16. It should be noted that for simplicity of notation we considered the special case in which the signal are separated according to that the spatial frequencies along  $R$ th baseline, which are estimated first. However, by exchanging the indices and rearranging the received data correspondingly it is straightforward to also exchange the order of the parameter estimation along the various baselines. We remark that due to the sequential estimation procedure the performance the R-D MODE technique critically depends on the estimation order. The spatial frequencies along which the sources are well-separated should therefore be estimated first.

#### Algorithm 16. Solving the R-D MODE function

- Obtain the initial estimate  $\hat{\mathbf{b}}^{(0)}, \hat{\mathbf{c}}_1^{(0)}, \dots, \hat{\mathbf{c}}_{M/M_R}^{(0)}$  of  $\mathbf{b}, \mathbf{c}_1, \dots, \mathbf{c}_{M/M_R}$ , respectively, from the quadratic function

$$f_{\text{WSF}}^{(0)}(\mathbf{b}, \mathbf{c}_1, \dots, \mathbf{c}_{M/M_R}) = \text{Tr}(\mathbf{G}^H \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \mathbf{G}) \quad (15.182)$$

and form  $\widehat{\mathbf{G}}$  from  $\hat{\mathbf{b}}^{(0)}, \hat{\mathbf{c}}_1^{(0)}, \dots, \hat{\mathbf{c}}_{M/M_R}^{(0)}$ .

- Solve the following quadratic function

$$f_{\text{WSF}}^{(1)}(\mathbf{b}, \mathbf{c}_1, \dots, \mathbf{c}_{M/M_R}) = \text{Tr}((\widehat{\mathbf{G}} \widehat{\mathbf{G}})^{-1} \mathbf{G}^H \widehat{\mathbf{U}}_s \mathbf{W}_{\text{MODE}} \widehat{\mathbf{U}}_s^H \mathbf{G}). \quad (15.183)$$

to obtain the vectors  $\hat{\mathbf{b}}^{(1)}, \hat{\mathbf{c}}_1^{(1)}, \dots, \hat{\mathbf{c}}_{M/M_R}^{(1)}$ .

- Root the polynomial with coefficients  $\hat{\mathbf{b}}^{(1)}$  to obtain the estimate  $\hat{\boldsymbol{\mu}}_R$  of the spatial frequencies along the  $R$ th baseline. Form the estimated steering matrix  $\mathbf{A}(\hat{\boldsymbol{\mu}}_R)$  and insert it in (15.179) and (15.179) along with  $\widehat{\mathbf{B}}^{(1)}$  and  $\widehat{\mathbf{C}}_j^{(1)}, j = 1, \dots, M/M_R$ . Compute the spatial frequencies along the remaining baselines from the solution of the system of equations.

### 3.15.4.2.5 R-D NC Standard ESPRIT

R-D NC Standard ESPRIT is applicable if the source signals  $s_i[n]$  for  $i = 1, 2, \dots, d, n = 1, 2, \dots, N$  represent samples from a strict-sense non-circular distribution, as described in Section 3.15.2.4. This implies that they can be expressed as  $s_i(t) = e^{\varphi_i} \cdot s_{0,i}(t)$ , where  $s_{0,i}(t) \in \mathcal{R}$  and  $\varphi_i$  does not change with time ( $n$ ). For the matrix of amplitudes  $\mathbf{S} \in \mathcal{C}^{d \times N}$  we can then write  $\mathbf{S} = \Psi \cdot \mathbf{S}_0$ , where  $\Psi = \text{diag}([e^{\varphi_1}, \dots, e^{\varphi_d}])$  and  $\mathbf{S}_0 \in \mathcal{R}^{d \times N}$ .

Based on this assumption we can define an augmented measurement matrix  $\mathbf{X}^{(\text{nc})}$  as [46, 63]<sup>11</sup>

$$\mathbf{X}^{(\text{nc})} = \begin{bmatrix} \mathbf{X} \\ \mathbf{\Pi}_M \cdot \mathbf{X}^* \end{bmatrix}. \quad (15.184)$$

Inserting  $\mathbf{X} = \mathbf{A} \cdot \mathbf{S} + \mathbf{N}$  and  $\mathbf{S} = \Psi \cdot \mathbf{S}_0$ , we can rewrite (15.184) into

$$\begin{aligned} \mathbf{X}^{(\text{nc})} &= \begin{bmatrix} \mathbf{A} \cdot \mathbf{S} \\ \mathbf{\Pi}_M \cdot \mathbf{A}^* \cdot \mathbf{S}^* \end{bmatrix} + \begin{bmatrix} \mathbf{N} \\ \mathbf{\Pi}_M \cdot \mathbf{N}^* \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A} \\ \mathbf{\Pi}_M \cdot \mathbf{A}^* \cdot \Psi^* \cdot \Psi^* \end{bmatrix} \cdot \mathbf{S} + \begin{bmatrix} \mathbf{N} \\ \mathbf{\Pi}_M \cdot \mathbf{N}^* \end{bmatrix} \\ &= \mathbf{A}^{(\text{nc})} \cdot \mathbf{S} + \mathbf{N}^{(\text{nc})}, \end{aligned} \quad (15.185)$$

since  $\mathbf{S}^* = \Psi^* \cdot \mathbf{S}_0$  and  $\Psi^* \cdot \mathbf{S} = \mathbf{S}_0$ . Equation (15.185) shows that the desired signal component of the augmented  $\mathbf{X}^{(\text{nc})}$  can be factorized into an extended array steering matrix  $\mathbf{A}^{(\text{nc})} \in \mathcal{C}^{2M \times d}$  and the original matrix of amplitudes  $\mathbf{S} \in \mathcal{C}^{d \times N}$ . A remarkable property of  $\mathbf{A}^{(\text{nc})}$  is the following: If the array steering matrix  $\mathbf{A}$  is shift-invariant, i.e.,  $\mathbf{J}_1 \cdot \mathbf{A} \cdot \Phi = \mathbf{J}_2 \cdot \mathbf{A}$ , where  $\mathbf{J}_1$  and  $\mathbf{J}_2 \in \mathcal{R}^{M^{(\text{sel})} \times M}$  are the selection matrices for the first and the second subarray, then  $\mathbf{A}^{(\text{nc})}$  satisfies

$$\mathbf{J}_1^{(\text{nc})} \cdot \mathbf{A}^{(\text{nc})} \cdot \Phi = \mathbf{J}_2^{(\text{nc})} \cdot \mathbf{A}^{(\text{nc})}, \quad \text{where} \quad (15.186)$$

$$\mathbf{J}_1^{(\text{nc})} = \begin{bmatrix} \mathbf{J}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi}_{M^{(\text{sel})}} \cdot \mathbf{J}_2 \cdot \mathbf{\Pi}_M \end{bmatrix} \quad \text{and} \quad \mathbf{J}_2^{(\text{nc})} = \begin{bmatrix} \mathbf{J}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi}_{M^{(\text{sel})}} \cdot \mathbf{J}_1 \cdot \mathbf{\Pi}_M \end{bmatrix} \in \mathcal{R}^{2M^{(\text{sel})} \times 2M}. \quad (15.187)$$

The shift invariance in (15.186) was already used in [45] and by us in [46] for the special case of a ULA and the special case of a centro-symmetric array, respectively. Equation (15.186) is more general since it does not need further assumptions about the array except for the shift invariance. Note that (15.187) implies that via the augmentation we have created a virtual array of  $2M$  sensors with two shift invariant subarrays containing  $2M^{(\text{sel})}$  sensors. Consequently, this step doubles the number of sources that can be resolved simultaneously as well. Based on the shift invariance equation shown in (15.186) we can define an R-D Standard ESPRIT-type algorithm following the same steps as before. The resulting R-D NC Standard ESPRIT algorithm is summarized in Algorithm 17.

<sup>11</sup>Charg et al. [63] defines  $\mathbf{X}^{(\text{nc})}$  for root-MUSIC without the matrix  $\mathbf{\Pi}_M$ . The formulation in (15.184) we use here was first proposed by us in [46] to facilitate the real-valued implementation for Unitary ESPRIT.

**Algorithm 17.** Summary of *R-D NC Standard ESPRIT using Least Squares*

1. Estimate the augmented signal subspace  $\widehat{\mathbf{U}}_s^{(nc)} \in \mathcal{C}^{2M \times d}$  via the truncated SVD of the augmented observation matrix  $\mathbf{X}^{(nc)} \in \mathcal{C}^{2M \times N}$ .
2. Solve the overdetermined shift invariance equations

$$\widetilde{\mathbf{J}}_1^{(nc)(r)} \cdot \widehat{\mathbf{U}}_s^{(nc)} \cdot \Psi^{(r)} \approx \widetilde{\mathbf{J}}_2^{(nc)(r)} \cdot \widehat{\mathbf{U}}_s^{(nc)} \quad (15.188)$$

for the matrices  $\Psi^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS), where  $\widetilde{\mathbf{J}}_1^{(nc)(r)}$  and  $\widetilde{\mathbf{J}}_2^{(nc)(r)}$  are defined as (cf. (15.187))

$$\widetilde{\mathbf{J}}_n^{(nc)(r)} = \mathbf{I}_{M_1 \dots M_{r-1}} \otimes \mathbf{J}_n^{(nc)(r)} \otimes \mathbf{I}_{M_{r+1} \dots M_R}, \quad (15.189)$$

$$\mathbf{J}_1^{(nc)(r)} = \text{blkdiag} \left\{ \mathbf{J}_1^{(r)}, \Pi \cdot \mathbf{J}_2^{(r)} \cdot \Pi \right\}, \quad (15.190)$$

$$\mathbf{J}_2^{(nc)(r)} = \text{blkdiag} \left\{ \mathbf{J}_2^{(r)}, \Pi \cdot \mathbf{J}_1^{(r)} \cdot \Pi \right\}. \quad (15.191)$$

3. Compute the eigenvalues  $\hat{\lambda}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\widehat{\Psi}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99]. Recover the correctly paired frequencies  $\hat{\mu}_i^{(r)}$  via  $\hat{\mu}_i^{(r)} = \arg(\hat{\lambda}_i^{(r)})$ .

**3.15.4.2.6 R-D NC Unitary ESPRIT**

The extension of *R-D NC Standard ESPRIT* to *R-D NC Unitary ESPRIT* is again quite straightforward. There are two remarkable things to note here though. Firstly, while *R-D Unitary ESPRIT* requires the original array to be centro-symmetric, this is not required for *R-D NC Unitary ESPRIT*. The reason is that even if  $\mathbf{A}$  is not centro-symmetric, the augmented array steering matrix  $\mathbf{A}^{(nc)}$  is always centro-symmetric.<sup>12</sup>

The second surprising result is that forward-backward averaging has no effect on the performance. That means if we apply FBA to  $\mathbf{X}^{(nc)}$  the subspace estimate  $\widehat{\mathbf{U}}_s$  remains unaltered since

$$\mathbf{X}^{(nc)(fba)} \cdot \left( \mathbf{X}^{(nc)(fba)} \right)^H = 2 \cdot \mathbf{X}^{(nc)} \cdot \left( \mathbf{X}^{(nc)} \right)^H, \quad (15.192)$$

where  $\mathbf{X}^{(nc)(fba)} = \left[ \mathbf{X}^{(nc)} \ \boldsymbol{\Pi}_{2M} \cdot \mathbf{X}^{(nc)*} \cdot \boldsymbol{\Pi}_N \right]$ . Note that (15.192) has two important consequences. Firstly, it shows that the performance of *R-D NC Standard ESPRIT* and *R-D NC Unitary ESPRIT* is identical.<sup>13</sup> Secondly, it shows that unlike Unitary ESPRIT, NC Unitary ESPRIT cannot handle two coherent sources: FBA has no decorrelation effect as shown in (15.192) and the row-wise augmentation applied for NC ESPRIT has no decorrelation effect either (as evident from (15.185)).

<sup>12</sup>In the special case where the array is centro-symmetric, we have  $\mathbf{J}_2 = \boldsymbol{\Pi}_{M^{(\text{sel})}} \cdot \mathbf{J}_1 \cdot \boldsymbol{\Pi}_M$  and hence the augmented selection matrices simplify into  $\mathbf{J}_n^{(nc)} = \mathbf{I}_2 \otimes \mathbf{J}_n$ ,  $n = 1, 2$ .

<sup>13</sup>Combining the first two observations, it becomes clear that there is actually no need for a “Standard” version of *R-D NC Unitary ESPRIT*. It is included in this chapter for the sake of completeness only.

The third surprising result is that applying forward-backward averaging and the real-valued transformation, the resulting transformed measurement matrix takes the following simple form:

$$\mathcal{T}(X^{(nc)}) = 2 \cdot \begin{bmatrix} \operatorname{Re}(X) & \boldsymbol{\theta}_{M \times N} \\ \operatorname{Im}(X) & \boldsymbol{\theta}_{M \times N} \end{bmatrix}, \quad (15.193)$$

if the sparse left- $\Pi$ -real matrices  $\mathbf{Q}_p^{(s)}$  from (15.12) are used for the real-valued transformation. Since the zero block matrices and the factor 2 in front can be skipped, we conclude that the signal subspace can be estimated directly from the matrix where the real part of  $X$  and the imaginary part of  $X$  are stacked on top of each other. Based on this observation, an  $R$ -D NC Unitary ESPRIT algorithm can be derived, which is summarized in Algorithm 18.

---

**Algorithm 18** [46]. Summary of  $R$ -D NC Unitary ESPRIT using Least Squares

---

1. Estimate the augmented real-valued signal subspace  $\widehat{\mathbf{E}}_s^{(nc)} \in \mathcal{R}^{2M \times d}$  via the truncated SVD of the stacked observation  $[\operatorname{Re}(X)^T, \operatorname{Im}(X)^T]^T \in \mathcal{R}^{2M \times N}$ .
2. Solve the overdetermined shift invariance equations

$$\widetilde{\mathbf{K}}_1^{(nc)(r)} \cdot \widehat{\mathbf{E}}_s^{(nc)} \cdot \boldsymbol{\Upsilon}^{(r)} \approx \widetilde{\mathbf{K}}_2^{(nc)(r)} \cdot \widehat{\mathbf{E}}_s^{(nc)} \quad (15.194)$$

for the matrices  $\boldsymbol{\Upsilon}^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS), where

$$\widetilde{\mathbf{K}}_1^{(nc)(r)} = 2 \cdot \operatorname{Re} \left( \mathbf{Q}_{M_r^{(\text{sel})} \cdot M / M_r}^H \cdot \widetilde{\mathbf{J}}_2^{(nc)(r)} \cdot \mathbf{Q}_M \right), \quad (15.195)$$

$$\widetilde{\mathbf{K}}_2^{(nc)(r)} = 2 \cdot \operatorname{Im} \left( \mathbf{Q}_{M_r^{(\text{sel})} \cdot M / M_r}^H \cdot \widetilde{\mathbf{J}}_2^{(nc)(r)} \cdot \mathbf{Q}_M \right), \quad (15.196)$$

and  $\widetilde{\mathbf{J}}_n^{(nc)(r)}$  are defined in Algorithm 17.

3. Compute the eigenvalues  $\hat{\omega}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\widehat{\boldsymbol{\Upsilon}}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99] or via the Simultaneous Schur Decomposition proposed in [21]. Recover the correctly paired frequencies  $\hat{\mu}_i^{(r)}$  via  $\hat{\mu}_i^{(r)} = 2 \cdot \arctan(\hat{\omega}_i^{(r)})$ .
- 

### 3.15.4.3 Algorithms using tensor-based subspace estimates

#### 3.15.4.3.1 $R$ -D Standard Tensor-ESPRIT

As we have demonstrated in Section 3.15.2.3.8, the use of tensor algebra leads to a simplified and more natural formulation of the  $R$ -D shift invariance equations, since the artificial stacking operation and its consequences (such as the introduction of many Kronecker products) are avoided. Based on this idea, an  $R$ -D Standard ESPRIT algorithm can be formulated entirely in terms of tensors [42]. As we show in the sequel, this enhances the estimation accuracy due to the improved tensor-based subspace estimate shown in Section 3.15.3.4. Moreover, it enables us to find tensor-based solutions to the overdetermined shift invariance equations [52].

We first eliminate the unknown array steering tensor from the shift invariance equations (15.41) by virtue of the signal subspace tensor (15.82). This step is facilitated by the following relation between  $\mathcal{A}$  and  $\mathcal{U}^{[s]}$ :

$$\mathcal{A} = \mathcal{U}^{[s]} \times_{R+1} \bar{\mathbf{T}}, \quad (15.197)$$

where  $\bar{\mathbf{T}} \in \mathcal{C}^{d \times d}$  is a non-singular transform matrix. Essentially, (15.197) shows that the row spaces of the  $(R+1)$ -mode unfoldings of  $\mathcal{A}$  and  $\mathcal{U}^{[s]}$  agree. At the same time, if we compute any  $r$ -mode unfolding of (15.197) it becomes apparent that all the  $r$ -spaces of  $\mathcal{A}$  and  $\mathcal{U}^{[s]}$  coincide for  $r = 1, 2, \dots, R$ . It allows to eliminate the unknown array steering tensor from the shift invariance equations, replacing it by the estimated signal subspace tensor  $\hat{\mathcal{U}}^{[s]}$  via  $\mathcal{A} \approx \hat{\mathcal{U}}^{[s]} \times_{R+1} \bar{\mathbf{T}}$ . We then obtain

$$\hat{\mathcal{U}}^{[s]} \times_r \mathbf{J}_1^{(r)} \times_{R+1} \Psi^{(r)} \approx \hat{\mathcal{U}}^{[s]} \times_r \mathbf{J}_2^{(r)}, \quad (15.198)$$

where<sup>14</sup>  $\Psi^{(r)} = \bar{\mathbf{T}}^{-1} \cdot \Phi^{(r)} \cdot \bar{\mathbf{T}}$ ,  $r = 1, 2, \dots, R$  follows by applying identity (15.5) for repeated  $n$ -mode products. Note that due to (15.5), the order of the matrices in the definition of  $\Psi^{(r)}$  is reversed compared to the matrix case shown in (15.160).

The next step is the solution of the overdetermined sets of Eqs. (15.198) to yield the estimates  $\hat{\Psi}^{(r)}$ . It is easy to show that the Least Squares solution of the tensor-valued shift invariance equation (15.198) has the following closed-form solution:

$$\hat{\Psi}_{\text{LS}}^{(r)} = \arg \min_{\Psi} \left\| \hat{\mathcal{U}}^{[s]} \times_r \mathbf{J}_1^{(r)} \times_{R+1} \Psi - \hat{\mathcal{U}}^{[s]} \times_r \mathbf{J}_2^{(r)} \right\|_H^2 \quad (15.199)$$

$$\Rightarrow \hat{\Psi}_{\text{LS}}^{(r)T} = \left( \tilde{\mathbf{J}}_1^{(r)} \cdot [\hat{\mathcal{U}}^{[s]}]_{(R+1)}^T \right)^+ \cdot \tilde{\mathbf{J}}_2^{(r)} \cdot [\hat{\mathcal{U}}^{[s]}]_{(R+1)}^T. \quad (15.200)$$

Comparing (15.200) with (15.161) we see that the Least Squares solution of the matrix-based shift invariance equations for  $R$ -D Standard ESPRIT and the tensor-based shift invariance equations for  $R$ -D Standard Tensor-ESPRIT differ only in the choice of the subspace. Contemplating that the remaining steps (joint eigendecomposition of  $\hat{\Psi}_{\text{LS}}^{(r)}$  to recover the frequencies  $\mu_i^{(r)}$ ) are also the same, we can conclude that  $R$ -D Standard Tensor-ESPRIT is algebraically equivalent to  $R$ -D Standard ESPRIT if we replace the SVD-based subspace estimate  $\hat{\mathcal{U}}_s$  by the HOSVD-based subspace estimate  $[\hat{\mathcal{U}}^{[s]}]_{(R+1)}^T$ .

### 3.15.4.3.2 R-D unitary Tensor-ESPRIT

In the previous section we have seen that tensor calculus allows to derive a tensor-valued version of  $R$ -D Standard ESPRIT and that it is algebraically equivalent to matrix-based  $R$ -D Standard ESPRIT except for using the enhanced HOSVD-based subspace estimate.

We can proceed in a similar manner for  $R$ -D Unitary Tensor-ESPRIT. If the array is centro-symmetric, we can apply forward-backward averaging to the measurement tensor  $\mathcal{X}$  and then transform the resulting tensor onto the real-valued domain to lower the computational complexity. We can then estimate the

---

<sup>14</sup>Note that this is not exactly the same as the  $\Psi^{(r)}$  defined in Section 3.15.4.2.1, since the matrix of eigenvectors is different. However, since we are only interested in the eigenvalues, this difference is irrelevant, and hence we use the same variable for brevity.

signal subspace tensor via a truncated HOSVD of the transformed tensor  $\mathcal{T}(\mathcal{X}) \in \mathcal{R}^{M_1 \times \dots \times M_R \times 2N}$  shown in (15.85), i.e.,

$$\begin{aligned}\mathcal{T}(\mathcal{X}) &\approx \widehat{\mathcal{S}}_{\mathcal{T}}^{[s]} \times_1 \widehat{\mathbf{E}}_1^{[s]} \cdots \times_R \widehat{\mathbf{E}}_R^{[s]} \times_{R+1} \widehat{\mathbf{E}}_{R+1}^{[s]} \\ \Rightarrow \widehat{\mathcal{E}}^{[s]} &= \widehat{\mathcal{S}}_{\mathcal{T}}^{[s]} \times_1 \widehat{\mathbf{E}}_1^{[s]} \cdots \times_R \widehat{\mathbf{E}}_R^{[s]} \times_{R+1} \Sigma_{R+1}^{[s]^{-1}},\end{aligned}\quad (15.201)$$

where  $\widehat{\mathcal{E}}^{[s]} \in \mathcal{R}^{M_1 \times \dots \times M_R \times d}$ . Applying the real-valued transformation in (15.85) to the shift invariance Eqs. (15.198) yields the following transformed equations

$$\widehat{\mathcal{E}}^{[s]} \times_r \mathbf{K}_1^{(r)} \times_{R+1} \boldsymbol{\Upsilon}^{(r)} \approx \widehat{\mathcal{E}}^{[s]} \times_r \mathbf{K}_2^{(r)}, \quad (15.202)$$

where  $\mathbf{K}_1^{(r)} = 2 \cdot \text{Re} \left( \mathbf{Q}_{M_r^{(\text{sel})}}^H \cdot \mathbf{J}_2^{(r)} \cdot \mathbf{Q}_{M_r} \right)$  and  $\mathbf{K}_2^{(r)} = 2 \cdot \text{Im} \left( \mathbf{Q}_{M_r^{(\text{sel})}}^H \cdot \mathbf{J}_2^{(r)} \cdot \mathbf{Q}_{M_r} \right)$ . The real-valued shift invariance equations in (15.202) have the same algebraic form as the shift invariance equations for R-D Standard Tensor-ESPRIT shown in (15.198). Consequently, using similar arguments as in (15.200) we find the closed-form Least Squares solution to (15.202) via

$$\boldsymbol{\Upsilon}_{\text{LS}}^{(r)T} = \left( \widetilde{\mathbf{K}}_1^{(r)} \cdot \left[ \widehat{\mathcal{E}}^{[s]} \right]_{(R+1)}^T \right)^+ \cdot \widetilde{\mathbf{K}}_2^{(r)} \cdot \left[ \widehat{\mathcal{E}}^{[s]} \right]_{(R+1)}^T. \quad (15.203)$$

Note that as for R-D Standard Tensor-ESPRIT, for R-D Unitary Tensor-ESPRIT we again obtain a solution which is algebraically equivalent to the matrix-based R-D Unitary ESPRIT algorithm. A slight difference is that the transformed selection matrices  $\widetilde{\mathbf{K}}_1^{(r)}$  and  $\widetilde{\mathbf{K}}_2^{(r)}$  used in (15.203) are given by

$$\widetilde{\mathbf{K}}_n^{(r)} = (\mathbf{I}_{M_1} \otimes \dots \otimes \mathbf{I}_{M_{r-1}}) \otimes \mathbf{K}_n^{(r)} \otimes (\mathbf{I}_{M_{r+1}} \otimes \dots \otimes \mathbf{I}_{M_R}), \quad n = 1, 2, \quad (15.204)$$

and hence, they coincide with the matrices  $\widetilde{\mathbf{K}}_n^{(r)}$  defined in Algorithm 14 only if we choose the left- $\boldsymbol{\Pi}$ -real matrices  $\mathbf{Q}_M$  and  $\mathbf{Q}_{M_r^{(\text{sel})}, M/M_r}$  according to

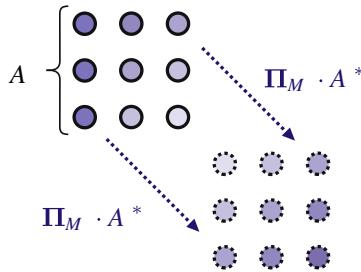
$$\mathbf{Q}_M = \mathbf{Q}_{M_1} \otimes \mathbf{Q}_{M_2} \otimes \dots \otimes \mathbf{Q}_{M_r} \otimes \dots \otimes \mathbf{Q}_{M_R}, \quad (15.205)$$

$$\mathbf{Q}_{M_r^{(\text{sel})}, M/M_r} = \mathbf{Q}_{M_1} \otimes \mathbf{Q}_{M_2} \otimes \dots \otimes \mathbf{Q}_{M_r^{(\text{sel})}} \otimes \dots \otimes \mathbf{Q}_{M_R}, \quad (15.206)$$

where the smaller  $\mathbf{Q}_{M_r}$  are arbitrary unitary left- $\boldsymbol{\Pi}$ -real matrices. Moreover, the matrix  $\mathbf{Q}_M$  used in the transformation  $\mathcal{T}(\mathcal{X})$  from (15.85) should also be chosen as in (15.205). However, since the particular choice of the left- $\boldsymbol{\Pi}$ -real matrices is irrelevant for the performance of R-D Unitary ESPRIT we again conclude that R-D Unitary ESPRIT and R-D Unitary Tensor-ESPRIT are algebraically equivalent except for the fact that the SVD-based subspace estimate  $\mathbf{E}_s$  is replaced by the HOSVD-based subspace estimate  $\widehat{\mathcal{E}}^{[s]}$ .

### 3.15.4.3.3 R-D NC standard Tensor-ESPRIT

In Section 3.15.4.3.1 we have shown how we can exploit the multidimensional structure of the R-D harmonic retrieval problem by virtue of tensor algebra, giving rise to the R-D Tensor-ESPRIT-type

**FIGURE 15.13**

Virtually doubled 2-D array after matrix-based augmentation of the measurements. The virtually doubled  $3 \times 3$  URA is augmented by a second URA flipped in both dimensions. The resulting array is not a separable 2-D sampling grid.

algorithms. On the other hand, in Section 3.15.4.2 we have shown how strict-sense non-circularity of the amplitudes (source symbols) can be exploited by virtue of widely linear signal processing, giving rise to NC-ESPRIT-type algorithms. This sparks the question whether both approaches can be combined for the case of  $R$ -D harmonic retrieval with strict-sense non-circular amplitudes.

However, combining the two approaches is not a trivial task. In fact, the augmentation that was applied for  $R$ -D NC-ESPRIT-type algorithms destroys the  $R$ -D separable sampling grid structure required for  $R$ -D Tensor-ESPRIT-type algorithms. This is exemplified in Figure 15.13, where we show the virtual 18-sensor array which results from performing the augmentation for matrix-based NC-ESPRIT-type algorithms to a  $3 \times 3$  URA. The additional virtual URA is flipped in both dimensions but neither augmented vertically nor horizontally. Hence, the resulting array is not a separable 2-D sampling grid, since we cannot express it as the outer product of 1-D sampling grids.

Consequently, in order to exploit both, the  $R$ -D structure and the strict-sense non-circularity at the same time, a tensor-compliant way of exploiting non-circularity is required. As shown in [47], this is accomplished by performing the augmentation along the individual modes separately (in the 2-D example along the rows and along the columns) and exploiting all these augmentations jointly.

To this end, let the  $r$ -mode augmented measurement tensor be given by

$$\mathcal{X}^{(nc,r)} = [\mathcal{X} \sqcup_r \mathcal{X}^* \times_1 \mathbf{\Pi}_{M_1} \cdots \times_R \mathbf{\Pi}_{M_R}] \in \mathcal{C}^{M_1 \times \cdots \times M_{r-1} \times 2M_r \times M_{r+1} \times \cdots \times M_R \times N}. \quad (15.207)$$

This tensor admits a factorization similar to (15.185), i.e.,

$$\mathcal{X}^{(nc,r)} = \mathcal{A}^{(nc,r)} \times_{R+1} S^T + \mathcal{N}^{(nc,r)}, \quad (15.208)$$

where the  $r$ -mode augmented array steering tensor  $\mathcal{A}^{(nc,r)}$  is given by

$$\begin{aligned} \mathcal{A}^{(nc,r)} &= [\mathcal{A} \sqcup_r \mathcal{A}^* \times_1 \mathbf{\Pi}_{M_1} \cdots \times_R \mathbf{\Pi}_{M_R} \times_{R+1} (\Psi^* \cdot \Psi^*)] \\ &\in \mathcal{C}^{M_1 \times \cdots \times M_{r-1} \times 2M_r \times M_{r+1} \times \cdots \times M_R \times d}. \end{aligned} \quad (15.209)$$

The  $R$ -D NC Tensor-ESPRIT-type algorithms are based on the shift invariance of  $\mathcal{A}^{(nc,r)}$ . It can be shown that the  $r$ -mode augmented array steering tensor  $\mathcal{A}^{(nc,r)}$  defined in (15.209) obeys the following

shift invariance equation

$$\mathcal{A}^{(nc,r)} \times_r J_1^{(nc,r)} \times_{R+1} \Phi^{(r)} = \mathcal{A}^{(nc,r)} \times_r J_2^{(nc,r)} \quad (15.210)$$

for  $r = 1, 2, \dots, R$ , where  $J_1^{(nc,r)}$  and  $J_2^{(nc,r)}$  are defined in (15.190) and (15.191), respectively.

In other words, (15.210) shows that the  $r$ -mode augmented array steering tensor is shift invariant with a “doubled” number of elements in the  $r$ th mode.<sup>15</sup> Therefore, the idea to exploit non-circularity and the  $R$ -D tensor structure jointly is to use all  $r$ -mode augmentations jointly, i.e., to extract estimates for  $\Phi^{(r)}$  from the  $\mathcal{X}^{(nc,r)}$  for  $r = 1, 2, \dots, R$ .

In order accomplish this goal, the unknown array steering tensors need to be replaced by estimates of appropriate signal subspace tensors. To this end, let the truncated HOSVD of the noise-free  $r$ -mode augmented measurement tensor  $\mathcal{X}_0^{(nc,r)}$  be given by  $\mathcal{X}_0^{(nc,r)} = \mathcal{S}^{[s](r)} \times_1 U_1^{[s](r)} \cdots \times_R U_R^{[s](r)} \times_{R+1} U_{R+1}^{[s](r)}$ . Define the  $r$ -mode augmented signal subspace tensor  $\mathcal{U}^{[s](r)}$  via

$$\mathcal{U}^{[s](r)} = \mathcal{S}^{[s](r)} \times_1 U_1^{[s](r)} \cdots \times_R U_R^{[s](r)} \times_{R+1} \Sigma_{R+1}^{[s](r)-1} \quad (15.211)$$

be the signal subspace tensor originating from the  $r$ -mode augmented measurement tensor  $\mathcal{X}^{(nc,r)}$ . Then, the following set of shift invariance equations is satisfied:

$$\mathcal{U}^{[s](r)} \times_r J_1^{(nc,r)} \times_{R+1} \Psi^{(r)} = \mathcal{U}^{[s](r)} \times_r J_2^{(nc,r)}, \quad r = 1, 2, \dots, R, \quad (15.212)$$

where  $\Psi^{(r)} = T \cdot \Phi^{(r)} \cdot T^{-1}$ , i.e.,  $T$  is not a function of  $r$ .

It is important to note that if the array is not centro-symmetric, the  $n$ -ranks of  $\mathcal{A}^{(nc,r)}$  can exceed  $d$ , which must be taken into account when computing the truncated HOSVD for  $\mathcal{U}^{[s](r)}$ . Since they are equal to  $2d$  in the worst case, it is safe to truncate the HOSVD to  $2d$  in the first  $R$  modes (of course, we still truncate to  $d$  in mode  $R+1$ ).

A very important aspect of (15.212) is that  $T$  is not a function of  $r$ , i.e., all  $\Psi^{(r)}$  still have a common set eigenvectors. This is crucial since the automatic pairing in  $R$ -D ESPRIT-type algorithms is based on this fact.

The  $R$ -D NC Standard Tensor-ESPRIT follows naturally from (15.212). It is summarized in Algorithm 19.

### 3.15.4.3.4 R-D NC unitary Tensor-ESPRIT

The extension of  $R$ -D NC Standard Tensor-ESPRIT to  $R$ -D NC Unitary Tensor-ESPRIT is again quite straightforward. In fact, many of the results from the matrix case (cf. Section 3.15.4.2.6) carry over to the tensor case. Firstly, the augmented array steering tensor  $\mathcal{A}^{(nc,r)}$  is centro-symmetric even if the original array steering tensor  $\mathcal{A}$  is not centro-symmetric.<sup>16</sup> Secondly, forward-backward averaging has no effect on the augmented tensor  $\mathcal{X}^{(nc,r)}$ , i.e.,

$$\left[ \mathcal{X}^{(nc,r)(fba)} \right]_{(R+1)}^T \cdot \left( \left[ \mathcal{X}^{(nc,r)(fba)} \right]_{(R+1)}^T \right)^H = 2 \cdot \left[ \mathcal{X}^{(nc,r)} \right]_{(R+1)}^T \left( \left[ \mathcal{X}^{(nc,r)} \right]_{(R+1)}^T \right)^H, \quad (15.214)$$

<sup>15</sup>Note that  $\mathcal{A}^{(nc,r)}$  is shift invariant in the other modes  $q = 1, 2, \dots, R, q \neq r$  only if the array is centro-symmetric in the  $q$ th mode, i.e.,  $\Pi_{M_q} \cdot A^{(q)*}$  and  $A^{(q)}$  span the same column space. However, this additional shift invariance is not needed for  $R$ -D NC Tensor-ESPRIT type algorithms.

<sup>16</sup>The condition on centro-symmetry  $\Pi_M \cdot A^* = A \cdot \Delta$  for the matrix case is expressed in tensor notation as  $\mathcal{A}^* \times_1 \Pi_1 \cdots \times_R \Pi_{M_R} = \mathcal{A} \times_{R+1} \Delta$ , where  $\Delta$  is a unitary diagonal matrix.

**Algorithm 19.** Summary of  $R$ -D NC Standard Tensor-ESPRIT using Least Squares

1. Estimate the augmented signal subspace tensors  $\hat{\mathcal{U}}^{[s](r)} \in \mathcal{C}^{M_1 \times \dots \times 2M_r \times \dots \times M_R \times d}$  via the truncated HOSVD of the  $r$ -mode augmented observation tensors  $\mathcal{X}^{(nc,r)}$  following (15.211) for  $r = 1, 2, \dots, R$ .
2. Solve the overdetermined shift invariance equations

$$\hat{\mathcal{U}}^{[s](r)} \times_r J_1^{(nc)(r)} \times_{R+1} \hat{\Psi}^{(r)} \approx \hat{\mathcal{U}}^{[s](r)} \times_r J_2^{(nc)(r)}. \quad (15.213)$$

for the matrices  $\hat{\Psi}^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS).

3. Compute the eigenvalues  $\hat{\lambda}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\hat{\Psi}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99]. Recover the correctly paired frequencies  $\hat{\mu}_i^{(r)}$  via  $\hat{\mu}_i^{(r)} = \arg(\hat{\lambda}_i^{(r)})$ .

where  $\mathcal{X}^{(nc,r)^{(fba)}} = [\mathcal{X}^{(nc,r)} \sqcup_{R+1} \mathcal{X}^{(nc,r)*} \times_1 \Pi_{M_1} \dots \times_R \Pi_{M_R} \times_{R+1} \Pi_N]$ . As in the matrix case, this shows that the performance of  $R$ -D NC Standard Tensor-ESPRIT is identical to  $R$ -D NC Unitary Tensor-ESPRIT and hence the latter is clearly preferable due to the lower computational complexity. Thirdly, in the matrix case, we had the result that the transformed real-valued measurement matrix has a very simple form (cf. (15.193)). Applying the tensor-based forward-backward averaging and the corresponding real-valued transformation which was introduced in (15.85) to  $\mathcal{X}^{(nc,r)}$ , we arrive at a simple direct form of the transformed measurement tensor as well. It is given by

$$\begin{aligned} \mathcal{T}(\mathcal{X}^{(nc,r)}) &= \left[ \mathcal{X}^{(nc,r)} \sqcup_{R+1} (\mathcal{X}^{(nc,r)*} \times_1 \Pi_{M_1} \dots \times_R \Pi_{M_R} \times_{R+1} \Pi_N) \right] \\ &\quad \times_1 \mathcal{Q}_{M_1}^H \dots \times_R \mathcal{Q}_{M_R}^H \times_{R+1} \mathcal{Q}_{2N}^H = \left[ \left[ 2 \cdot \operatorname{Re}(\bar{\mathcal{X}}^{(r)}) \sqcup_r 2 \cdot \operatorname{Im}(\bar{\mathcal{X}}^{(r)}) \right] \right. \\ &\quad \left. \times \sqcup_{R+1} [\mathcal{O}_{M_1 \times \dots \times M_R \times N} \sqcup_r \mathcal{O}_{M_1 \times \dots \times M_R \times N}] \right], \end{aligned}$$

where  $\bar{\mathcal{X}}^{(r)} = \mathcal{X} \times_1 \mathcal{Q}_{M_1}^H \dots \times_{r-1} \mathcal{Q}_{M_{r-1}}^H \times_{r+1} \mathcal{Q}_{M_{r+1}}^H \dots \times_R \mathcal{Q}_{M_R}^H$ .

Note that the zero entries in  $\mathcal{T}(\mathcal{X}^{(nc,r)})$  and the factor 2 can be skipped as they have no influence on the signal subspace estimate. Therefore, we can replace  $\mathcal{T}(\mathcal{X}^{(nc,r)})$  by the following simplified version

$$\bar{\mathcal{T}}(\mathcal{X}^{(nc,r)}) = \left[ \operatorname{Re}(\bar{\mathcal{X}}^{(r)}) \sqcup_r \operatorname{Im}(\bar{\mathcal{X}}^{(r)}) \right] \in \mathcal{R}^{M_1 \times \dots \times M_{r-1} \times 2M_r \times M_{r+1} \times \dots \times M_R \times N}. \quad (15.215)$$

Based on this result, the  $R$ -D NC Unitary Tensor-ESPRIT algorithm follows straightforwardly. It is summarized in Algorithm 20.

### 3.15.4.4 Simulation results

#### 3.15.4.4.1 1-D algorithms using matrix-based subspace estimates

In the first scenario we consider the 1-D algorithms using matrix-based subspace estimates adapted for a ULA. More precisely, we compare the performance of the following algorithms : root-MUSIC, MODE

---

**Algorithm 20** [47]. Summary of R-D NC Unitary Tensor-ESPRIT using Least Squares

---

1. Estimate the real-valued augmented signal subspace tensors  $\widehat{\mathcal{E}}^{[s](r)} \in \mathcal{R}^{M_1 \times \dots \times 2M_r \times \dots \times M_R \times d}$  via the truncated HOSVD of the transformed  $r$ -mode augmented observation tensors  $\tilde{T}(\mathcal{X}^{(nc,r)}) \in \mathcal{R}^{M_1 \times \dots \times M_{r-1} \times 2M_r \times M_{r+1} \times \dots \times M_R \times N}$  shown in (15.215) for  $r = 1, 2, \dots, R$ .
2. Solve the overdetermined shift invariance equations

$$\widehat{\mathcal{E}}^{[s](r)} \times_r \mathbf{K}_1^{(nc)(r)} \times_{R+1} \widehat{\mathbf{\Upsilon}}^{(r)} \approx \widehat{\mathcal{E}}^{[s](r)} \times_r \mathbf{K}_2^{(nc)(r)} \quad (15.216)$$

for the matrices  $\widehat{\mathbf{\Upsilon}}^{(r)}$  for  $r = 1, 2, \dots, R$  via the method of Least Squares (LS), where

$$\mathbf{K}_1^{(nc)(r)} = 2 \cdot \text{Re} \left( \mathbf{Q}_{M_r^{(\text{sel})}}^H \cdot \mathbf{J}_2^{(nc)(r)} \cdot \mathbf{Q}_{M_r} \right), \quad (15.217)$$

$$\mathbf{K}_2^{(nc)(r)} = 2 \cdot \text{Im} \left( \mathbf{Q}_{M_r^{(\text{sel})}}^H \cdot \mathbf{J}_2^{(nc)(r)} \cdot \mathbf{Q}_{M_r} \right), \quad (15.218)$$

and  $\mathbf{J}_n^{(nc)(r)}$  are defined in (15.190) and (15.191).

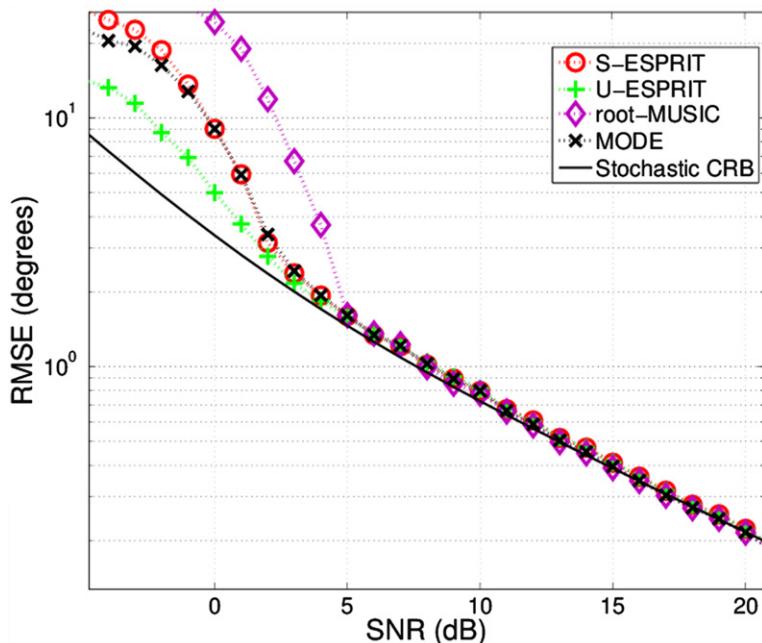
3. Compute the eigenvalues  $\hat{\omega}_i^{(r)}$  for  $i = 1, 2, \dots, d$  of  $\widehat{\mathbf{\Upsilon}}^{(r)}$  jointly for all  $r = 1, 2, \dots, R$ , e.g., via the joint diagonalization scheme proposed in [99] or via the Simultaneous Schur Decomposition proposed in [21]. Recover the correctly paired frequencies  $\hat{\mu}_i^{(r)}$  via  $\hat{\mu}_i^{(r)} = 2 \cdot \arctan(\hat{\omega}_i^{(r)})$ .
- 

(as presented in the summary box 8), Standard ESPRIT (S-ESPRIT), and Unitary ESPRIT (U-ESPRIT) with the stochastic Cramér-Rao lower bound (CRB) [57]. The array is assumed to be uniform and linear with  $M = 8$  sensors spaced by half wavelength. The two sources, assumed to be far-field narrowband complex circular Gaussian sequences with zero mean and variance equal to one, impinge on the array from  $\theta_1 = 10^\circ$  and  $\theta_2 = 15^\circ$ . Finally, the simulation results are averaged over 1000 simulation runs and the ESPRIT-based schemes consider a maximum overlap.

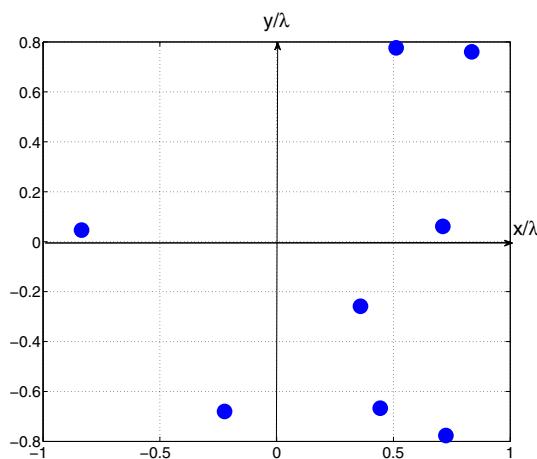
From Figure 15.14 one can notice that the threshold of the root-MUSIC algorithm occurs at a higher SNR than the MODE and ESPRIT-based algorithms. Furthermore, Figure 15.14 suggests that the MODE and S-ESPRIT algorithms exhibit the same threshold phenomena, whereas the U-ESPRIT shows the best breakdown point. Finally, all the listed algorithms above are in a good agreement with the stochastic CRB in the asymptotic region.

In the second scenario we focus on the 1-D search free schemes using matrix-based subspace estimates adapted for a NULA. More precisely, we compare the performance of the following algorithms: interpolated root-MUSIC, manifold separation, and Fourier domain root-MUSIC. The stochastic CRB is plotted as benchmark, the spectral MUSIC and the Min-Norm schemes are also added.

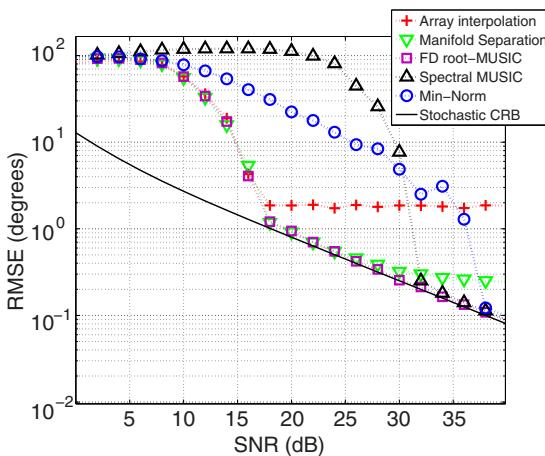
The array is assumed to be non-uniform with  $M = 8$  sensors as represented by Figure 15.15. The two sources, assumed to be far-field narrowband complex circular Gaussian sequences with zero mean and variance equal to one, impinge on the array from  $\theta_1 = 10^\circ$  and  $\theta_2 = 15^\circ$  with  $N = 15$  snapshots. The simulation results are averaged over 1000 simulation runs and the matrices  $V_v$  and  $V_{MS}$  are derived using the LS in order to obtain an optimum approximation of the array manifold. The interpolated root-MUSIC scheme is applied to sectors of width  $45^\circ$  where  $M_V = M_{MS} = M_{FD} = 21$ .

**FIGURE 15.14**

1-D algorithms using matrix-based subspace estimates for 2 uncorrelated sources using a ULA with  $M = 8$ .

**FIGURE 15.15**

Non-uniform array geometry used for the second example.

**FIGURE 15.16**

1-D algorithms using matrix-based subspace estimates for 2 uncorrelated sources using the non-uniform array represented in Figure 15.15 with  $M = 8$ .

Figure 15.16 shows that the interpolated root-MUSIC, the manifold separation and the Fourier domain root-MUSIC schemes exhibit a saturation in their performance in the asymptotic region. Nevertheless, it is shown that the Fourier domain root-MUSIC algorithm's saturation happens for a higher SNR than for the other methods. This saturation can be attenuated by increasing  $M_V$ ,  $M_{MS}$ , and  $M_{FD}$  for the interpolated root-MUSIC, the manifold separation and the Fourier domain root-MUSIC algorithms, respectively, but in return, this increases the computational cost of these search free methods. Finally, in comparison with the spectral form of the MUSIC and the Min-Norm techniques, it can be noticed from Figure 15.16 that the manifold separation and the Fourier domain root-MUSIC show a breakdown point which occurs in a lower SNR.

### 3.15.4.4.2 R-D algorithms using matrix-based subspace estimates

In order to compare the multidimensional (matrix-based and tensor-based) algorithms we investigate multidimensional harmonics sampled on separable R-D sampling grids in the sequel. For simplicity we assume totally uniform sampling, i.e., the sampling grid is chosen uniformly in all  $R$  modes. For  $R = 2$ , this assumption coincides with a uniform rectangular array (URA) of  $M_1 \times M_2$  sensors. The source amplitudes  $s_i(t)$  are modeled as circularly symmetric complex Gaussian distributed random variables with zero mean, variance one, and a correlation coefficient given by  $|E\{s_i(t) \cdot s_j(t)^*\}| = \rho \in [0, 1] \forall i \neq j = 1, 2, \dots, d$ . Moreover, the noise samples are assumed to be mutually uncorrelated zero mean circularly symmetric complex Gaussian distributed with common variance  $\sigma_n^2$ . The SNR is defined as  $SNR = \sigma_n^{-2}$ . The estimation accuracy of the algorithms is measured with respect to a root mean square estimation error (RMSE) defined as

$$RMSE = \sqrt{E \left\{ \frac{1}{d} \frac{1}{R} \sum_{i=1}^d \sum_{r=1}^R (\mu_i^{(r)} - \hat{\mu}_i^{(r)})^2 \right\}}. \quad (15.219)$$

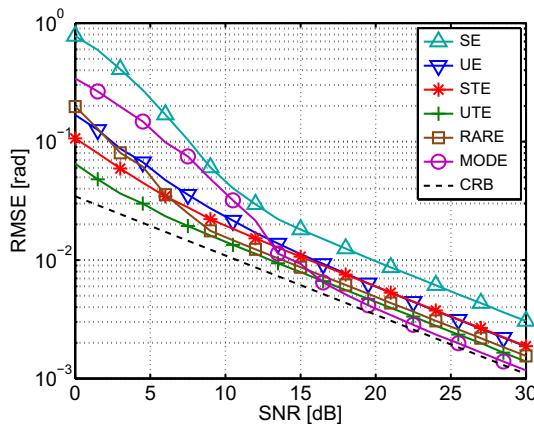
**Table 15.3** Abbreviations Used for *R-D* the Simulation Results

Abbreviation	Meaning
RMSE	Root mean square error
SNR	Signal to noise ratio
SE	<i>R-D</i> Standard ESPRIT
UE	<i>R-D</i> Unitary ESPRIT
STE	<i>R-D</i> Standard Tensor-ESPRIT
UTE	<i>R-D</i> Unitary Tensor-ESPRIT
NC UE	<i>R-D</i> NC Unitary ESPRIT
NC UTE	<i>R-D</i> NC Unitary Tensor-ESPRIT
CRB	Deterministic Cramér-Rao bound
CRBnc	Deterministic Cramér-Rao bound for strict-sense non-circular sources [101]
LS	Least Squares
SLS	Structured Least Squares
TS-SLS	Tensor-Structure Structured Least Squares
TS-RD-SLS	Tensor-Structure <i>R-D</i> Structured Least Squares

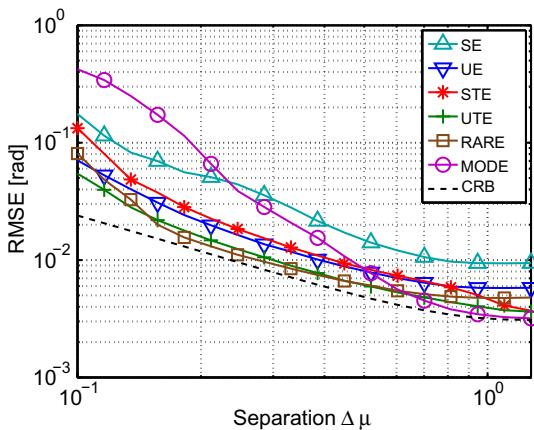
We compare the *R-D* versions of Standard ESPRIT (SE), Unitary ESPRIT (UE), Standard Tensor-ESPRIT (STE), Unitary Tensor-ESPRIT, MODE, as well as RARE. Note that for *R-D* MODE and *R-D* RARE, forward-backward averaging is used as a preprocessing step. As a reference, we also display the corresponding deterministic Cramér-Rao bound (CRB). Table 15.3 summarizes the abbreviations used in the subsequent figure captions.

Figures 15.17 and 15.18 compare the performance of the 2-D MODE and the 2-D RARE algorithm with the 2-D ESPRIT-type algorithms SE, STE, UE, and UTE, all based on LS. We consider a  $6 \times 6$  URA,  $N = 6$  snapshots, and  $d = 2$  correlated sources with  $\rho = 0.9$ . For Figure 15.17 we set the true spatial frequencies to  $\mu_1^{(1)} = 1, \mu_2^{(1)} = 0.2, \mu_1^{(2)} = 0.2$ , and  $\mu_2^{(2)} = 1$  and vary the SNR. For Figure 15.18 we fix the SNR to 20 dB and vary the source positions as a function of the spatial separation  $\Delta\mu$  according to  $\mu_1^{(1)} = \mu_1^{(2)} = 1$  and  $\mu_2^{(1)} = \mu_2^{(2)} = 1 - \Delta\mu$ . We can clearly see that the tensor-based algorithms STE and UTE outperform the matrix-based algorithms SE and UE as a result of the enhanced HOSVD-based subspace estimate. For high SNR and a larger spatial separation, 2-D MODE outperforms UTE. Note that we do not show UTE combined with SLS in Figures 15.17 and 15.18, which performs better than the UTE with LS shown here.

In Figures 15.19 and 15.20 we investigate the performance of SLS and its tensor-based extension TS-SLS. For the first result shown in Figure 15.19 we consider  $d = 3$  highly correlated sources ( $\rho = 0.999$ ) captured by a  $3 \times 3$  URA and use  $N = 10$  snapshots. The true spatial frequencies are  $\mu_1^{(1)} = 1, \mu_1^{(2)} = -1, \mu_2^{(1)} = 0, \mu_2^{(2)} = 1, \mu_3^{(1)} = -1, \mu_3^{(2)} = 0$ . Note that since  $d = M_1 = M_2$ , this is a scenario where the HOSVD-based subspace estimate coincides with the SVD-based subspace estimate (cf. Section 3.15.3.4). Consequently, the LS-based Tensor-ESPRIT algorithms coincide with their matrix based counterparts, as evident from the overlapping curves for SE LS and STE LS in Figure 15.19. Likewise, UE SLS and UTE SLS coincide. However, using the tensor-based extensions of

**FIGURE 15.17**

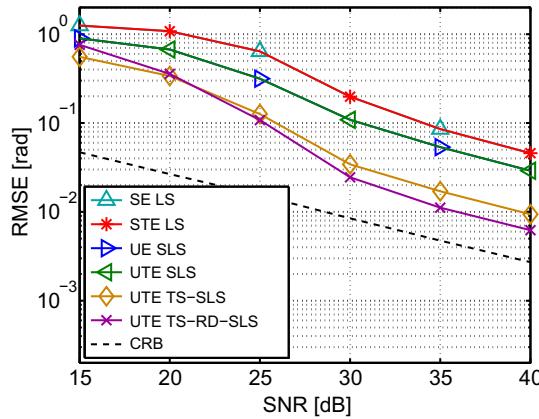
RMSE vs. SNR for a  $6 \times 6$  URA,  $N = 6$  snapshots, two correlated sources ( $\rho = 0.9$ ) positioned at  $\mu_1^{(1)} = 1$ ,  $\mu_2^{(1)} = 0.2$ ,  $\mu_1^{(2)} = 0.2$ , and  $\mu_2^{(2)} = 1$ .

**FIGURE 15.18**

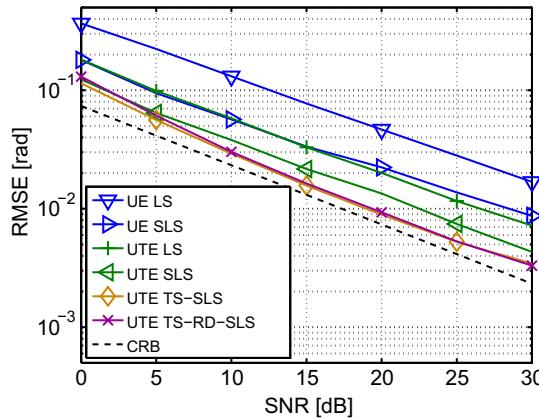
RMSE vs. the spatial separation  $\Delta\mu$  for a  $6 \times 6$  URA,  $N = 6$  snapshots, an SNR of 20 dB and two correlated sources ( $\rho = 0.9$ ) positioned at  $\mu_1^{(1)} = \mu_1^{(2)} = 1$  and  $\mu_2^{(1)} = \mu_2^{(2)} = 1 - \Delta\mu$ .

SLS we can still benefit from the tensor structure in this scenario. UTE TS-SLS and UTE TS-2D-SLS clearly outperform UTE SLS.

For Figure 15.20 we switch to a  $5 \times 7$  URA and consider a single snapshot ( $N = 1$ ) only. The  $d = 2$  sources' spatial frequencies are given by  $\mu_1^{(1)} = 1$ ,  $\mu_1^{(2)} = -1$ ,  $\mu_2^{(1)} = 0$ ,  $\mu_2^{(2)} = 1$ . The figure shows that SLS outperforms LS, UTE outperforms UE, and TS-SLS outperforms SLS, as expected.

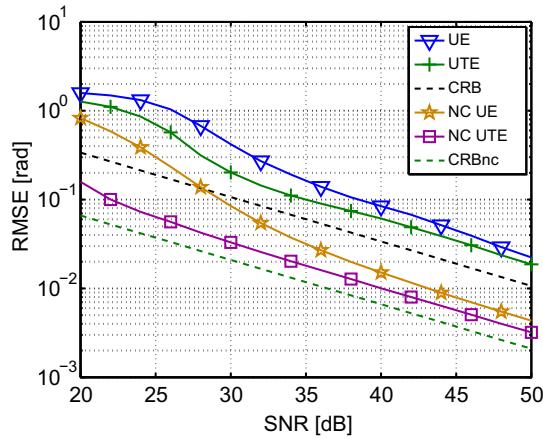
**FIGURE 15.19**

RMSE vs. SNR for  $d = 3$  correlated sources ( $\rho = 0.999$ ) on a  $3 \times 3$  URA,  $N = 10$ ,  $\mu_1^{(1)} = 1$ ,  $\mu_1^{(2)} = -1$ ,  $\mu_2^{(1)} = 0$ ,  $\mu_2^{(2)} = 1$ ,  $\mu_3^{(1)} = -1$ ,  $\mu_3^{(2)} = 0$ .

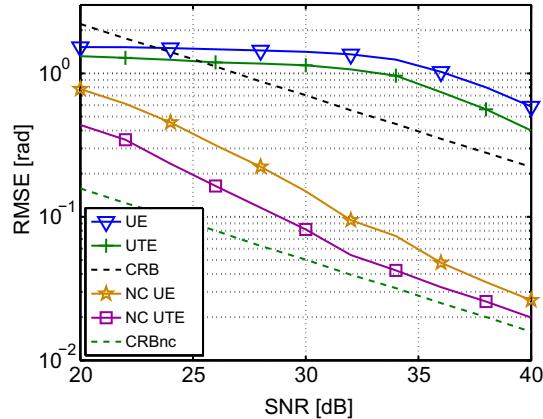
**FIGURE 15.20**

RMSE vs. SNR for  $d = 2$  sources on a  $5 \times 7$  URA, single snapshot ( $N = 1$ ),  $\mu_1^{(1)} = 1$ ,  $\mu_1^{(2)} = -1$ ,  $\mu_2^{(1)} = 0$ ,  $\mu_2^{(2)} = 1$ .

The final set of simulations demonstrates the performance of ESPRIT-type algorithms for non-circular sources. For simplicity we only consider Unitary ESPRIT-type algorithms and compare UE and NC UE with their tensor versions UTE and NC UTE. For comparison we also display the deterministic Cramér-Rao bound for strict-sense non-circular sources (CRB<sub>nc</sub>) from [101]. The strict-sense non-circular source amplitudes are generated according to (15.47), generating  $S_0$  from a standard normal distribution with source correlation  $|\mathbb{E} \{s_{0,i}(t) \cdot s_{0,j}(t)\}| = \rho$ .

**FIGURE 15.21**

RMSE vs. SNR for  $d = 3$  correlated sources ( $\rho = 0.99$ ) at fixed positions  $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.85, \mu_3^{(1)} = \mu_3^{(2)} = 1.15$  with phase angles  $\varphi_1 = 0, \varphi_2 = \pi/2, \varphi_3 = \pi/4$ . A  $5 \times 7$  URA and  $N = 10$  snapshots.

**FIGURE 15.22**

RMSE vs. SNR for a  $6 \times 6$  URA,  $N = 10$  snapshots,  $d = 4$  uncorrelated sources at fixed positions  $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.9, \mu_3^{(1)} = \mu_3^{(2)} = 0.8, \mu_4^{(1)} = \mu_4^{(2)} = 0.7$  with phase angles  $\varphi_1 = 0, \varphi_2 = \pi/6, \varphi_3 = \pi/3, \varphi_4 = \pi/2$ .

For the simulation result shown in Figure 15.21 we consider a  $5 \times 7$  URA,  $N = 10$  snapshots and  $d = 3$  correlated sources ( $\rho = 0.99$ ). The sources' phase angles are fixed to  $\varphi_1 = 0, \varphi_2 = \pi/2, \varphi_3 = \pi/4$  and the true spatial frequencies are given by  $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.85, \mu_3^{(1)} = \mu_3^{(2)} = 1.15$ . On the other hand, for Figure 15.22 we use a  $6 \times 6$  URA and  $d = 4$  uncorrelated sources with phase angles

given by  $\varphi_1 = 0, \varphi_2 = \pi/6, \varphi_3 = \pi/3, \varphi_4 = \pi/2$ . Moreover, the true spatial frequencies in this scenario are  $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.9, \mu_3^{(1)} = \mu_3^{(2)} = 0.8, \mu_4^{(1)} = \mu_4^{(2)} = 0.7$ . Both simulation results show that NC UE outperforms UE (due to exploiting the noncircularity), UTE outperforms UE (due to exploiting the *R-D* structure), and NC UTE outperforms both (by combining both benefits).

### 3.15.5 Conclusions

In this chapter we have presented a detailed overview of subspace methods adapted for uniform arrays, non-uniform arrays, and other specific array structures. The popularity of many of these special array structures is due to the availability of search-free low computational complexity DOA or spatial frequency estimation algorithms particularly to exploit the structure of the array. Different subspace based algorithms have been compared using numerical simulations.

These subspace-based algorithms can be classified, with respect to their numerical procedure into spectral searching techniques and search-free techniques. The spectral searching techniques include the well known MUSIC, RARE, weighted subspace fitting schemes, and their variants. Whereas, the search-free techniques can be partitioned into two subclasses:

- Polynomial-rooting techniques, e.g., root-MUSIC and its variants for the ULA context: in the non-uniform array context, the main idea is to use the approximation of the true steering vector by a virtual steering vector (using, e.g., manifold separation or Fourier transform) in order to obtain a search-free algorithm based on polynomial rooting. In this chapter, we have presented and compared several polynomial-rooting techniques with application to non-uniform arrays. It has been noticed that the interpolated root-MUSIC, the manifold separation and the Fourier domain root-MUSIC schemes exhibit a saturation in their performance in the asymptotic region. This saturation can be attenuated by increasing the number of the virtual arrays, but in return, this increases the computational cost.
- The matrix-shifting techniques, e.g., matrix pencil methods, ESPRIT, and its variants: due to its simplicity and universal applicability ESPRIT has becomes one of the most popular signal subspace based spatial frequency estimation methods. ESPRIT is premised on array geometries that exhibit a shift invariance structure that enables the estimation of the source DOA parameters from the eigenvalues of an estimated matrix. Numerical simulations have shown that the threshold of the root-MUSIC algorithm occurs at a higher SNR than ESPRIT-based algorithms, in which the Unitary ESPRIT scheme performs best among all ESPRIT-based schemes.

For the case of multidimensional parameter estimation, we have introduced *R-D* matrix-based and tensor-based algorithms. We have demonstrated that multidimensional signals can be represented by tensors which provide a natural formulation of the *R*-dimensional signals and their properties (such as the *R-D* shift invariances needed for matrix shifting techniques). Based on this representation, an improved HOSVD-based signal subspace estimate was defined. We have shown that this subspace estimate performs a more efficient denoising of the data which leads to a tensor gain in terms of an enhanced estimation accuracy. This subspace estimate can be combined with arbitrary existing multidimensional subspace-based parameter estimation schemes.

Then we have discussed the tensor-based schemes *R-D* Standard Tensor-ESPRIT and *R-D* Unitary Tensor-ESPRIT. They outperform the matrix based *R-D* ESPRIT-type algorithms due to the enhanced

subspace estimate obtained from the HOSVD. We have also shown that strict-sense non-circular sources can be exploited to virtually double the number of available sensors by an augmentation of the measurement matrix. Based on this idea, the *R*-D NC Standard ESPRIT and the *R*-D NC Unitary ESPRIT algorithm are derived. As a result, the number of resolvable wavefronts is doubled and the achievable estimation accuracy is improved. Finally, the family of NC Tensor-ESPRIT-type algorithms has been introduced to combine both benefits, the strict-sense non-circular source symbols and the multidimensional structure of the signals. This is a non-trivial task, since the augmentation of the measurement matrix performed for *R*-D NC Unitary ESPRIT destroys the structure needed for the Tensor-ESPRIT-type algorithms. It has been solved by defining a mode-wise augmentation of the measurement tensor. The resulting gain from using tensors and using non-circular sources has been demonstrated in numerical simulations.

---

## Acknowledgment

This work was partially supported by the European Research Council (ERC) Advanced Investigator Grants program under Grant 227477-ROSE.

---

## References

- [1] A.J. van der Veen, E.F. Deprettere, A.L. Swindlehurst, Subspace-based signal analysis using singular value decomposition, Proc. IEEE 81 (1993) 1277–1308.
- [2] F. Li, R.J. Vaccaro, Unified analysis for DOA estimation algorithms in array signal processing, Signal Process. 25 (1991) 147–169.
- [3] R.O. Schmidt, Multiple emitter location and signal parameter estimation, in: Proceeding of the RADCOM Spectrum Estimation Workshop, Griffiths AFB, NY, 1979, pp. 243–258 (Reprinted in IEEE Trans. Antennas Propag. 34 (1986) 276–280).
- [4] V.F. Pisarenko, The retrieval of harmonics from a covariance function, Geophys. J. Roy. Astron. Soc. 33 (1973) 347–366.
- [5] R. Kumaresan, D.W. Tufts, Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise, IEEE Trans. Acoust. Speech Signal Process. ASSP-30 (1982) 833–840.
- [6] A.J. Barabell, Improving the resolution performance of eigenstructure-based direction-finding algorithms, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing, Boston, MA, 1983, pp. 336–339.
- [7] B.D. Rao, K.V.S. Hari, Performance analysis of root-MUSIC, IEEE Trans. Acoust. Speech Signal Process. 37 (1989) 1939–1948.
- [8] R. Roy, T. Kailath, ESPRIT-Estimation of signal parameters via rotational invariance techniques, IEEE Trans. Acoust. Speech Signal Process. ASSP-37 (1989) 984–995.
- [9] S.Y. Kung, K.S. Arun, D.V. Bhaskar Rao, State space and SVD based approximation methods for the harmonic retrieval problem, J. Opt. Soc. Am. 73 (1983) 1799–1811.
- [10] B.D. Rao, K.S. Arun, Model based processing of signals: a state space approach, Proc. IEEE 80 (1992) 283–309.
- [11] Y. Hua, Estimating two-dimensional frequencies by matrix enhancement and matrix pencil, IEEE Trans. Signal Process. 40 (1992) 2267–2280.

- [12] Y. Hua, A pencil-MUSIC algorithm for finding two-dimensional angles and polarizations using crossed dipoles, *IEEE Trans. Antennas Propag.* 41 (1993) 370–376.
- [13] Y. Hua, T.K. Sarkar, On SVD for estimating generalized eigenvalues of singular matrix pencil in noise, *IEEE Trans. Signal Process.* 39 (1991) 892–900.
- [14] A. Eriksson, P. Stoica, Optimally weighted ESPRIT for direction estimation, *Signal Process.* 38 (1994) 223–229.
- [15] M. Haardt, J.A. Nossek, Unitary ESPRIT: how to obtain increased estimation accuracy with a reduced computational burden, *IEEE Trans. Signal Process.* 43 (1995) 1232–1242.
- [16] L.L. Scharf, *Statistical Signal Processing*, Addison-Wesley Publishing Comp., Reading, MA, 1991.
- [17] M. Haardt, R.S. Thomä, A. Richter, Multidimensional high-resolution parameter estimation with applications to channel sounding, in: Y. Hua, A. Gershman, Q. Chen (Eds.), *High-Resolution and Robust Signal Processing*, Marcel Dekker, New York, NY, 2004, pp. 255–338 (Chapter 5).
- [18] X. Liu, N.D. Sidiropoulos, A. Swami, Blind high-resolution localization and tracking of multiple frequency hopped signals, *IEEE Trans. Signal Process.* 50 (2002) 889–901.
- [19] X. Liu, N.D. Sidiropoulos, T. Jiang, Multidimensional harmonic retrieval with applications in MIMO wireless channel sounding, in: A. Gershman, N. Sidiropoulos (Eds.), *Space-Time Processing for MIMO Communications*, John Wiley & Sons, Ltd., 2005, pp. 41–75 (Chapter 2).
- [20] M.D. Zoltowski, M. Haardt, C.P. Mathews, Closed-form 2D angle estimation with rectangular arrays in element space or beamspace via Unitary ESPRIT, *IEEE Trans. Signal Process.* 44 (1996) 316–328.
- [21] M. Haardt, J.A. Nossek, Simultaneous schur decomposition of several non-symmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems, *IEEE Trans. Signal Process.* 46 (1998) 161–169.
- [22] H.L. Van Trees, *Optimum Array Processing: Detection, Estimation, and Modulation Theory, Part IV*, Wiley, New York, 2002.
- [23] K.N. Mokios, N.D. Sidiropoulos, M. Pesavento, C.E. Mecklenbräuker, On 3-D harmonic retrieval for wireless channel sounding, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2004), 2004, pp. 89–92.
- [24] M. Pesavento, C.F. Mecklenbräuker, J.F. Böhme, Multidimensional rank reduction estimator for parametric MIMO channel models, *EURASIP J. Appl. Signal Process.* (2004) 1354–1363.
- [25] T. Jiang, N.D. Sidiropoulos, J.M.F. ten Berge, Almost sure identifiability of multidimensional harmonic retrieval, *IEEE Trans. Signal Process.* 49 (2002) 1849–1859.
- [26] X. Liu, N.D. Sidiropoulos, Almost sure identifiability of constant modulus multidimensional harmonic retrieval, *IEEE Trans. Signal Process.* 50 (2002) 2366–2368.
- [27] P.M. Kroonenberg, J. de Leeuw, Principal component analysis of three-mode data by means of alternating least squares algorithms, *Psychometrika* 45 (1980) 69–97.
- [28] M. Rajih, Blind identification of underdetermined mixtures based on the characteristic function, Ph.D. Thesis, Université de Nice à Sophia Antipolis, 2006.
- [29] N.D. Sidiropoulos, R. Bro, G.B. Giannakis, Parallel factor analysis in sensor array processing, *IEEE Trans. Signal Process.* 48 (2000) 2377–2388.
- [30] L. deLathauwer, B. deMoor, J. Vanderwalle, A multilinear singular value decomposition, *SIAM J. Matrix Anal. Appl.* (2000) 21.
- [31] L. deLathauwer, B. deMoor, J. Vanderwalle, On the best rank-1 and rank- $(r_1, r_2, \dots, r_n)$  approximation of higher-order tensors, *SIAM J. Matrix Anal. Appl.* (2000) 21.
- [32] L.R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika* 31 (1966) 279–311.
- [33] H.A.L. Kiers, P.M. Kroonenberg, J.M.F. ten Berge, An efficient algorithm for TUCKALS3 on data with large numbers of observation units, *Psychometrika* 57 (1992) 415–422.

- [34] P.M. Kroonenberg, Three-Mode Principle Component Analysis: Theory and Applications, DSWO Press, Leiden, 1983.
- [35] R. Bro, N. Sidiropoulos, G.B. Giannakis, A fast least squares algorithm for separating trilinear mixtures, in: Proceedings of the International Workshop on Independent Component Analysis for Blind Signal Separation (ICA 99), 1999, pp. 289–294.
- [36] N.D. Sidiropoulos, G.B. Giannakis, R. Bro, Blind PARAFAC receivers for DS-CDMA systems, IEEE Trans. Signal Process. 48 (2000) 810–823.
- [37] D.A. Linebarger, R.D. DeGroat, E.M. Dowling, Efficient direction finding methods employing forward/backward averaging, IEEE Trans. Signal Process. 42 (1994) 2136–2145.
- [38] A. Lee, Centrohermitian and skew-centrohermitian matrices, Linear Algebra Appl. 29 (1980) 205–210.
- [39] R. Roy, A. Paulraj, T. Kailath, ESPRIT - a subspace rotation approach to estimation of parameters of cisoids in noise, IEEE Trans. Acoust. Speech Signal Process. AASP-34 (1986) 1340–1342.
- [40] F. Li, H. Liu, R.J. Vaccaro, Performance analysis for DOA estimation algorithms: unification, simplifications, and observations, IEEE Trans. Aerosp. Electron. Syst. 29 (1993) 1170–1184.
- [41] C.P. Mathews, M. Haardt, M.D. Zoltowski, Performance analysis of closed-form, ESPRIT based 2-D angle estimator for rectangular arrays, IEEE Signal Process. Lett. 3 (1996) 124–126.
- [42] M. Haardt, F. Roemer, G. Del Galdo, Higher-order SVD based subspace estimation to improve the parameter estimation accuracy in multi-dimensional harmonic retrieval problems, IEEE Trans. Signal Process. 56 (2008) 3198–3213.
- [43] F. Roemer, H. Becker, M. Haardt, Analytical performance analysis for multi-dimensional Tensor-ESPRIT-type parameter estimation algorithms, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2010), Dallas, TX, 2010.
- [44] F. Roemer, H. Becker, M. Haardt, M. Weis, Analytical performance evaluation for HOSVD-based parameter estimation schemes, in: Proceeding of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2009), Aruba, Dutch Antilles, 2009.
- [45] A. Zoubir, P. Chargé, Y. Wang, Non circular sources localization with ESPRIT, in: Proceeding of the European Conference on Wireless Technology (ECWT 2003), Munich, Germany, 2003.
- [46] M. Haardt, F. Roemer, Enhancements of Unitary ESPRIT for non-circular sources, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2004), Montreal, Canada, 2004, pp. 101–104.
- [47] F. Roemer, M. Haardt, Multidimensional unitary Tensor-ESPRIT for non-circular sources, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2009), Taipei, Taiwan, 2009.
- [48] R. Roy, T. Kailath, Total least-squares ESPRIT, in: Proceedings of the 21st Asilomar Conference Circuits System Computer, Pacific Grove, CA, 1987.
- [49] B. Ottersten, M. Viberg, T. Kailath, Performance analysis of the total least squares ESPRIT algorithm, IEEE Trans. Signal Process. 39 (1991) 1122–1135.
- [50] M. Haardt, Structured least squares to improve the performance of ESPRIT-type algorithms, IEEE Trans. Signal Process. 45 (1997) 792–799.
- [51] F. Roemer, M. Haardt, Analytical performance assessment of 1-D structured least squares, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2011), Prague, Czech Republic, 2011.
- [52] F. Roemer, M. Haardt, Tensor-structure structured least squares (TS-SLS) to improve the performance of multi-dimensional ESPRIT-type algorithms, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2007), Honolulu, HI, 2007, pp. 893–896.
- [53] L. De Lathauwer, B. De Moor, J. Vandewalle, A multilinear singular value decomposition, SIAM J. Matrix Anal. Appl. 21 (2000) 1253–1278.

- [54] L. Huang, T. Long, S. Wu, Source enumeration for high-resolution array processing using improved Gershgorin radii without eigendecomposition, *IEEE Trans. Signal Process.* 56 (2008) 5916–5925.
- [55] M. Wax, T. Kailath, Detection of signals by information theoretic criteria, in: Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1985), Florida, USA, 1985.
- [56] M. Wax, I. Ziskind, Detection of the number of coherent signals by the MDL principle, *IEEE Trans. Acoust. Speech Signal Process.* 37 (1989) 1190–1196.
- [57] P. Stoica, R. Moses, *Spectral Analysis of Signals*, Prentice Hall, NJ, 2005.
- [58] A.T. Moffet, Minimum redundancy linear arrays, *IEEE Trans. Antennas Propag.* 16 (1968) 172–175.
- [59] C.P. Mathews, M.D. Zoltowski, Performance analysis of the UCA-ESPRIT algorithm for circular ring arrays, *IEEE Trans. Signal Process.* 42 (1994) 2535–2539.
- [60] F. Roemer, M. Haardt, Using 3-D Unitary ESPRIT on a hexagonal shaped ESPAR antenna for 1-D and 2-D direction of arrival estimation, in: Proceeding of the ITG/IEEE Workshop on Smart Antennas (WSA'05), Duisburg, Germany, 2005.
- [61] F. Roemer, M. Haardt, Efficient 1-D and 2-D DOA estimation for non-circular sources with hexagonal shaped ESPAR arrays, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2006), Toulouse, France, 2006, pp. 881–884.
- [62] H. Abeida, J.-P. Delmas, MUSIC-like estimation of direction of arrival for noncircular sources, *IEEE Trans. Signal Process.* 54 (2006) 2678–2690.
- [63] P. Charg, Y. Wang, J. Saillard, A non circular sources direction finding method using polynomial rooting, *Signal Process.* (2001) 1765–1770.
- [64] A. Liu, G. Liao, Q. Xu, C. Zeng, A circularity-based DOA estimation method under coexistence of noncircular and circular signals, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing, Kyoto, Japan, 2012.
- [65] J.P. Delmas, H. Abeida, Stochastic cramer-rao bound for noncircular signals with application to DOA estimation, *IEEE Trans. Signal Process.* 52 (2004) 3192–3199.
- [66] J. Eriksson, V. Koivunen, Complex random vectors and ICA models: identifiability, uniqueness, and separability, *IEEE Trans. Inform. Theory* 52 (2006) 1017–1029.
- [67] E. Ollila, On the circularity of a complex random variable, *IEEE Signal Process. Lett.* 15 (2008) 841–844.
- [68] P. Chevalier, F. Pipon, New insights into optimal widely linear array receivers for the demodulation of BPSK, MSK and GMSK signals corrupted by non circular interferences—application to SAIC, *IEEE Trans. Signal Process.* 54 (2006) 870–883.
- [69] T.-J. Shan, M. Wax, T. Kailath, On spatial smoothing for direction-of-arrival estimation of coherent signals, *IEEE Trans. Acoust. Speech Signal Process.* 33 (1985) 806–811.
- [70] A. Thakre, M. Haardt, K. Giridhar, Single snapshot spatial smoothing with improved effective array aperture, *IEEE Signal Process. Lett.* 16 (2009) 505–509.
- [71] J.P.C.L. Da Costa, F. Roemer, M. Haardt, R.T. de Sousa Jr., Multi-dimensional model order selection, *EURASIP J. Adv. Signal Process.* 26 (2011) (review article).
- [72] A. Thakre, M. Haardt, F. Roemer, K. Giridhar, Tensor-Based spatial smoothing (TB-SS) using multiple snapshots, *IEEE Trans. Signal Process.* (2010).
- [73] A. Thakre, M. Haardt, K. Giridhar, Single snapshot  $r$ -d Unitary ESPRIT using an augmentation of the tensor order, in: Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2009), 2009.
- [74] R.O. Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Trans. Antennas Propag. AP-34* (1986) 243–258.
- [75] S.S. Reddi, Multiple source location—a digital approach, *IEEE Trans. Aerosp. Electron. Syst.* 15 (1979) 95–105.

- [76] M. Kaveh, A.J. Barabell, The statistical performance of the MUSIC and minimum-norm algorithms in resolving plane waves in noise, *IEEE Trans. Acoust. Speech Signal Process.* 34 (1986) 331–341.
- [77] H. Krim, P. Forster, J.G. Proakis, Operator approach to performance analysis of root-MUSIC and root Min-Norm, *IEEE Trans. Acoust. Speech Signal Process.* 40 (1992) 1687–1688.
- [78] P. Stoica, P. Handel, A. Nehorai, Improved sequential MUSIC, *IEEE Trans. Aerosp. Electron. Syst.* 31 (1995) 1230–1239.
- [79] J.C. Mosher, R.M. Leahy, Source localization using recursively applied and projected (RAP) music, *IEEE Trans. Signal Process.* 39 (1999) 332–340.
- [80] M. Pesavento, A.B. Gershman, K.M. Wong, Direction finding in partly calibrated sensor arrays composed of multiple subarrays, *IEEE Trans. Signal Process.* 50 (2002) 2103–2115.
- [81] M. Viberg, B. Ottersten, T. Kailath, Detection and estimation in sensor arrays using weighted subspace fitting, *IEEE Trans. Signal Process.* 39 (1991) 2436–2449.
- [82] B. Friedlander, A.J. Weiss, Direction finding using spatial smoothing with interpolated arrays, *IEEE Trans. Aerosp. Electron. Syst.* 28 (1992a) 574–587.
- [83] H.L. VanTrees, *Detection, Estimation and Modulation Theory: Optimum Array Processing*, vol. 4, Wiley, New York, 2002.
- [84] F. Belloni, A. Richter, V. Koivunen, DOA estimation via manifold separation for arbitrary array structures, *IEEE Trans. Signal Process.* 55 (2007) 4800–4810.
- [85] M. Doron, E. Doron, Wavefield modeling and array processing, Part II. Algorithm, *IEEE Trans. Signal Process.* 42 (1994) 2571–2580.
- [86] M. Rubsam, A. Gershman, Direction-of-arrival estimation for non-uniform sensor arrays: from manifold separation to Fourier domain MUSIC methods, *IEEE Trans. Signal Process.* 57 (2007) 588–599.
- [87] H. Krim, M. Viberg, Two decades of array signal processing research: the parametric approach, *IEEE Signal Process. Mag.* 13 (1996) 67–94.
- [88] P. Stoica, A. Nehorai, MUSIC, maximum likelihood, and Cramer-Rao bound, *IEEE Trans. Acoust. Speech Signal Process.* 37 (1989) 720–741.
- [89] P. Stoica, A. Nehorai, MUSIC, maximum likelihood and Cramer-Rao bound: further results and comparisons, *IEEE Trans. Acoust. Speech Signal Process.* 38 (1990) 2140–2150.
- [90] B.D. Rao, K.V.S. Hari, Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise, *IEEE Trans. Acoust. Speech Signal Process. AASP-37* (1989) 1990–1995.
- [91] B. Friedlander, A.J. Weiss, Direction finding using spatial smoothing with interpolated arrays, *IEEE Trans. Aerosp. Electron. Syst.* 28 (1992b) 574–587.
- [92] M.D. Zoltowski, J.M. Kautz, S.D. Silverstein, Beamspace root-music, *IEEE Trans. Signal Process.* 41 (1996) 344–364.
- [93] F. Gao, A.B. Gershman, A generalized ESPRIT approach to direction-of-arrival estimation, *IEEE Signal Process. Lett.* 10 (2005) 254–257.
- [94] E. Tuncer, B. Friedlander, *Classical and Modern Direction-of-Arrival Estimation*, Academic Press, Elsevier Inc., USA, 2009.
- [95] J. Li, P. Stoica, Z.-S. Liu, Comparative study of IQML and MODE direction-of-arrival estimators, *IEEE Trans. Signal Process.* 46 (1998) 149–160.
- [96] P. Stoica, K. Sharman, Maximum likelihood methods for direction of arrival estimation, *IEEE Trans. Acoust. Speech Signal Process.* 38 (1990) 1132–1143.
- [97] M. Pesavento, Fast algorithms for multidimensional harmonic retrieval, Ph.D. Thesis, Ruhr-University Bochum, 2005.
- [98] C.M.S. See, A.B. Gershman, Direction-of-arrival estimation in partly calibrated subarray-based sensor arrays, *IEEE Trans. Signal Process.* 52 (2004) 329–338.

- [99] T. Fu, X. Gao, Simultaneous diagonalization with similarity transformation for non-defective matrices, in: Proceeding of the IEEE International Conference Acoustics, Speech Signal Processing (ICASSP 2006), Toulouse, France, 2006.
- [100] A. Swindlehurst, P. Stoica, M. Jansson, Exploiting arrays with multiple invariances using music and mode, *IEEE Trans. Signal Process.* 49 (2001) 2511–2521.
- [101] F. Roemer, M. Haardt, Deterministic Cramér-Rao bounds for strict sense non-circular sources, in: Proceeding of the ITG/IEEE Workshop on Smart Antennas (WSA'07), Vienna, Austria, 2007.

# Performance Bounds and Statistical Analysis of DOA Estimation

# 16

Jean Pierre Delmas

TELECOM SudParis, Département CITI, CNRS UMR 5157, Evry Cedex, France

## 3.16.1 Introduction

Over the last three decades, many direction of arrival (DOA) estimation and source number detection methods have been proposed in the literature. Early studies on statistical performance were only based on extensive Monte Carlo experiments. Analytical performance evaluations, that allow one to evaluate the expected performance, as pioneering by Kaveh and Barabell [1], have since attracted much excellent research.

The earlier works were devoted to the statistical performance analysis of subspace-based algorithms. In particular the celebrated MUSIC algorithm has been extensively investigated (see, e.g., [2–5] among many others). But curiously, these works were based on first-order perturbations of the eigenvectors and eigenvalues of the sample covariance matrix, and thus involved very complicated derivations. Subsequently, Krim et al. [6] carried out a performance analysis of two eigenstructure-based DOA estimation algorithms, using a series expansion of the orthogonal projectors on the signal and noise subspaces, allowing considerable simplification of the previous approaches. Motivated by this point of view, several unified analyses of subspace-based algorithms have been presented (see, e.g., [7–9]). In parallel to these works, a particular attention has been paid to the statistical performance of the exact and approximative maximum likelihood algorithms (ML), in relation to the celebrated Cramer-Rao bound (see, e.g., [10,11], and the tutorial [12] with the references therein).

The statistical performance analysis of the difficult and critical problem of the detection of the number of sources impinging on an array, has been based on principally standard techniques of the statistical detection literature. In particular, the information theoretical criteria and especially the minimum description length (MDL), as popularized in the signal processing literature by [13], have been analyzed (see, e.g., [14–16]). Related to the DOA estimation accuracy and to the detection of the number of sources, the resolvability of closely spaced signals in terms of their parameters of interest have been also extensively studied (see, e.g., [17,18]).

The aim of this chapter is not to give a survey of all performance analysis of DOA estimation and source detection methods that have appeared in the literature, but rather, to provide a unified methodology introduced in [19] and then specialized to second-order in [20] to study the theoretical statistical performance of arbitrary DOA estimation and source number detection methods and to tackle the resolvability of closely space sources. To illustrate this framework, several examples are detailed

such as the conventional MUSIC algorithm, the MDL criterion and the angular resolution limit based on the detection theory.

This chapter is organized as follows. Section 3.16.2 presents the mathematical model of the array output and introduce the basic assumptions. General statistical tools for performance bounds and statistical analysis of DOA estimation algorithms are given in Section 3.16.3 based on a functional approach providing a common unifying framework. Then, Section 3.16.4 embarks on statistical performance analysis of beamforming-based, maximum likelihood and second-order algorithms with a particular attention paid to the subspace-based algorithms. In particular the robustness w.r.t. the Gaussian distribution, the independence and narrowband assumptions, and array modeling errors are considered. Finally some elements of statistical performance analysis of high-order algorithms complete this section. A glimpse into the detection of the number of sources is given in Section 3.16.5 where a performance analysis of the minimum description length (MDL) criterion is derived. Finally, Section 3.16.6 is devoted to criteria for resolving two closely spaced sources.

The following notations are used throughout this chapter:  $o(\epsilon)$  and  $O(\epsilon)$  denote quantities such that  $\lim_{\epsilon \rightarrow 0} o(\epsilon)/\epsilon = 0$  and  $|O(\epsilon)/\epsilon|$  is bounded in the neighborhood of  $\epsilon = 0$ , respectively.

## 3.16.2 Models and basic assumption

### 3.16.2.1 Parametric array model

Consider an array of  $M$  sensors arranged in an arbitrary geometry that receives the waveforms generated by  $P$  point sources (electromagnetic or acoustic). The output of each sensor is modeled as the response of a linear time-invariant bandpass system of bandwidth  $B$ . The impulse response of each sensor to a signal impinging on the array depends on the physical antenna structure, the receiver electronics and other antennas in the array through mutual coupling. The complex amplitudes  $s_p(t)$  of these sources w.r.t. a carrier frequency  $f_0$  are assumed to vary very slowly relative to the propagation time across the array (more precisely, the array aperture measured in wavelength, is much less than the inverse relative bandwidth  $f_0/B$ ). This so-called narrowband assumption allows the time delays  $\tau_{m,p}$  of the  $p$ th source at the  $m$ th sensor, relative to some fixed reference point, to be modeled as a simple phase-shift of the carrier frequency. If  $\mathbf{n}(t)$  is the complex envelope of the additive noise, the complex envelope of the signals collected at the output of the sensors is given by applying the superposition principle for linear sensors by:

$$\mathbf{x}(t) = \sum_{p=1}^P \mathbf{a}(\boldsymbol{\theta}_p) s_p(t) + \mathbf{n}(t) = \mathbf{A}(\boldsymbol{\theta}) \mathbf{s}(t) + \mathbf{n}(t), \quad (16.1)$$

where  $\mathbf{s}(t) \stackrel{\text{def}}{=} [s_1(t), \dots, s_P(t)]^T$  and  $\boldsymbol{\theta}_p$  may include generally azimuth, elevation, range and polarization of the  $p$ th source. However, we will here assume that there is only one parameter per source, referred as the direction of arrival (DOA)  $\boldsymbol{\theta}$ .  $\mathbf{a}(\boldsymbol{\theta}_p)$  is the steering vector associated with the  $p$ th source. The array manifold, defined as the set  $\{\mathbf{a}(\boldsymbol{\theta}), \boldsymbol{\theta} \in \Theta\}$  for some region  $\Theta$  in DOA space, is perfectly known, either analytically or by measuring it in the field. It is further required for performance analysis that  $\mathbf{a}(\boldsymbol{\theta})$  be continuously twice differentiable w.r.t.  $\boldsymbol{\theta}$ .  $\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\boldsymbol{\theta}_1), \dots, \mathbf{a}(\boldsymbol{\theta}_P)]$  is the  $M \times P$  steering matrix with  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_P]^T$ .

To illustrate the parameterization of the steering vector  $\mathbf{a}(\theta)$ , assume that the sources are in the far field of the array, and that the medium is non-dispersive, so that the waveforms can be approximated as planar. In this case, the  $m$ th component of  $\mathbf{a}(\theta)$  is simply  $g_m(\theta)e^{-i\mathbf{k}^T \mathbf{r}_m}$  where  $g_m(\theta)$  is the directivity gain of the  $m$ th sensor,  $\mathbf{k} \stackrel{\text{def}}{=} \frac{2\pi f_0}{c} \mathbf{u}$ ,  $c$  represents the speed of propagation,  $\mathbf{u}$  is a unit vector pointing in the direction of propagation and  $\mathbf{r}_m$  is the position of the  $m$ th sensor relative the origin of the different delays.

The by far most studied sensor geometry is that of uniform linear array (ULA), where the  $M$  sensors are assumed to be identical and omnidirectional over the DOA range of interest. Referenced w.r.t. the first sensor that is used as the origin,  $g_m(\theta) = 1$  and  $\mathbf{k}^T \mathbf{r}_m = (m-1) \frac{2\pi f_0}{c} d \sin(\theta) = (m-1) \frac{2\pi d}{\lambda_0} \sin(\theta)$ , where  $\lambda_0$  is the wavelength. To avoid any ambiguity,  $d$  must be less than or equal to  $\frac{\lambda_0}{2}$ . The standard ULA has  $d = \frac{\lambda_0}{2}$  that ensures a maximum accuracy on the estimation of  $\theta$ . In this case

$$\mathbf{a}(\theta) = [1, e^{i\pi \sin(\theta)}, \dots, e^{i(M-1)\pi \sin(\theta)}]^T. \quad (16.2)$$

### 3.16.2.2 Signal assumptions and problem formulation

Each vector observation  $\mathbf{x}(t)$  is called a snapshot of the array output. Let the process  $\mathbf{x}(t)$  be observed at  $N$  time instants  $\{t_1, \dots, t_N\}$ .  $\mathbf{x}(t)$  is often sampled at a slow sampling frequency  $1/T_s$  compared to the bandwidth of  $\mathbf{x}(t)$  for which  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  are independent. Temporal correlation between successive snapshots is generally not a problem, but implies that a larger number  $N$  of snapshots is needed for the same performance. We will prove in Section 3.16.4.3 that the parameter that fixes the performance is not  $N$ , but the observation interval  $T = NT_s$ . The signals  $\{s_p(t)\}_{p=1, \dots, P}$  and  $\mathbf{n}(t)$  are assumed independent.<sup>1</sup> For well calibrated arrays,  $\mathbf{n}(t)$  is often assumed to be dominated by thermal noise in the receivers, which can be well modeled as zero-mean temporally and spatially white circular Gaussian random process. In this case,  $E[\mathbf{n}(t_i)\mathbf{n}^H(t_j)] = \sigma_n^2 \delta_{i,j} \mathbf{I}$  and  $E[\mathbf{n}(t_i)\mathbf{n}^T(t_j)] = \mathbf{0}$ , for which the spatial covariance and spatial complementary covariance matrices are given by  $\mathbf{R}_n \stackrel{\text{def}}{=} E[\mathbf{n}(t)\mathbf{n}^H(t)] = \sigma_n^2 \mathbf{I}$  and  $\mathbf{C}_n \stackrel{\text{def}}{=} E[\mathbf{n}(t)\mathbf{n}^T(t)] = \mathbf{0}$ , respectively. A common, alternative model assumes that  $\mathbf{n}(t)$  is spatially correlated where  $\mathbf{R}_n$  is known up to a scalar multiplicative term  $\sigma_n^2$ , i.e.,  $\mathbf{R}_n = \sigma_n^2 \Sigma_n$  where  $\Sigma_n$  is a known definite positive matrix. In this case,  $\mathbf{x}(t)$  can be pre-multiplied by an inverse square-root factor  $\Sigma_n^{-1/2}$  of  $\Sigma_n$ , which renders the resulting noise spatially white and preserves model (16.1) by replacing the steering vectors  $\mathbf{a}(\theta)$  by  $\Sigma_n^{-1/2} \mathbf{a}(\theta)$ .

Two kind of assumptions are used for  $\{s_p(t)\}_{p=1, \dots, P}$ . In the first one, called stochastic or unconditional model (see, e.g., [10, 11]),  $\{s_p(t)\}_{p=1, \dots, P}$  are assumed to be zero-mean random variables for which the most commonly used distribution is the circular Gaussian one with spatial covariance  $\mathbf{R}_s \stackrel{\text{def}}{=} E[\mathbf{s}(t)\mathbf{s}^H(t)]$  and spatial complementary covariance  $\mathbf{C}_s \stackrel{\text{def}}{=} E[\mathbf{s}(t)\mathbf{s}^T(t)] = \mathbf{0}$ .  $\mathbf{R}_s$  is nonsingular for not fully correlated sources (called also noncoherent) or near-singular for highly correlated sources. In the case of coherent sources (specular multipath or smart jamming, where some signals impinging on the array of sensors can be sums of scaled and delayed versions of the others),  $\mathbf{R}_s$  is singular. In this chapter  $\mathbf{R}_s$  is usually assumed nonsingular. For these assumptions, the snapshots  $\mathbf{x}(t)$  are zero-mean complex circular Gaussian distributed with covariance matrix

---

<sup>1</sup>Note that only the uncorrelation assumption is required for second-order based algorithms, in contrast to fourth-order based algorithms, that require the independent assumption. However, this latter one simplifies the statistical performance analysis.

$$\mathbf{R}_x = \mathbf{A}(\boldsymbol{\theta})\mathbf{R}_s\mathbf{A}^H(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I}. \quad (16.3)$$

This circular Gaussian assumption lies not only in the fact that circular Gaussian data are rather frequently encountered in applications, but also because optimal detection and estimation algorithms are much easier to deduce under this assumption. Furthermore, as will be discussed in Section 3.16.4, under rather general conditions and in large samples [21], the Gaussian CRB is the largest of all CRB matrices corresponding to different distributions of the sources of identical covariance matrix  $\mathbf{R}_s$ . This stochastic model can be extended by assuming that  $\mathbf{s}(t)$  is arbitrarily distributed with finite fourth-order moments [20] including the case where  $\mathbf{C}_s \neq \mathbf{0}$  associated with the second-order noncircular distributions.

A common alternative assumption, called deterministic or conditional model (see, e.g., [10, 11]) is used when the distribution of  $\mathbf{s}(t)$  is unknown or/and clearly non-Gaussian, for example in radar and radio communications. Here  $\mathbf{s}(t)$  is nonrandom, i.e., the sequence  $\{\mathbf{s}(t)\}_{t_1, \dots, t_N}$  is frozen in all realizations of the random snapshots  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$ . Consequently,  $\{\mathbf{s}(t)\}_{t_1, \dots, t_N}$  is considered as a complex unknown parameter in  $\mathbb{C}^{NP}$ . For this assumption, the snapshots  $\mathbf{x}(t)$  are complex circular Gaussian distributed with mean  $\mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t)$  and covariance matrix  $\sigma_n^2\mathbf{I}$ .

With these preliminaries, the main DOA problem can now be formulated as follows: Given the observations,  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  and the described model (16.1), detect the number  $P$  of incoming sources and estimate their DOAs  $\{\theta_p\}_{p=1, \dots, P}$ .

### 3.16.2.3 Parameter identifiability

Once the distribution of the observations  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  has been fixed, the question of the identifiability of the parameters (including the DOA  $\{\theta_p\}_{p=1, \dots, P}$ ) must be raised. For example, under the assumption of independent, zero-mean circular Gaussian distributed observations, all information in the measured data is contained in the covariance matrix  $\mathbf{R}_x$  (16.3). The question of parameter identifiability is thus reduced to investigating under which conditions  $\mathbf{R}_x$  determines the unknown parameters. Thus, if no a priori information on  $\mathbf{R}_s$  is available, the unknown parameter  $\boldsymbol{\alpha}$  of  $\mathbf{R}_x$  contains the following  $P + P^2 + 1$  real-valued parameters:

$$\boldsymbol{\alpha} = \left[ \theta_1, \dots, \theta_P, [\mathbf{R}_s]_{1,1}, \dots, [\mathbf{R}_s]_{P,P}, \operatorname{Re}([\mathbf{R}_s]_{2,1}), \right. \\ \left. \operatorname{Im}([\mathbf{R}_s]_{2,1}), \dots, \operatorname{Re}([\mathbf{R}_s]_{P,P-1}), \operatorname{Im}([\mathbf{R}_s]_{P,P-1}), \sigma_n^2 \right]^T \quad (16.4)$$

and the parameter  $\boldsymbol{\alpha}$  is identifiable if and only if  $\mathbf{R}_x(\boldsymbol{\alpha}^{(1)}) = \mathbf{R}_x(\boldsymbol{\alpha}^{(2)}) \Rightarrow \boldsymbol{\alpha}^{(1)} = \boldsymbol{\alpha}^{(2)}$ . To ensure this identifiability, it is necessary that  $\mathbf{A}(\boldsymbol{\theta})$  be full column rank for any collection of  $P$ , distinct  $\theta_p \in \Theta$ . An array satisfying this assumption is said to be unambiguous. Notice that this requirement is problem-dependent and, therefore, has to be established for the specific array under study. For example, due to the Vandermonde structure of  $\mathbf{a}(\boldsymbol{\theta})$  in the ULA case (16.2), it is straightforward to prove that the ULA is unambiguous if  $\Theta = (-\pi/2, +\pi/2)$ . In the case where the rank of  $\mathbf{R}_s$ , that is the dimension of the linear space spanned by  $\mathbf{s}(t)$  is known and equal to  $r$ , different conditions of identifiability has been given in the literature. In particular, the condition

$$P < \frac{M+r}{2} \quad (\text{which reduces to } P < M \text{ when } \mathbf{R}_s \text{ is nonsingular}) \quad (16.5)$$

has been proved to be sufficient [22] and practically necessary [23].

When  $\mathbf{s}(t)$  are not circularly Gaussian distributed, the identifiability condition is generally much more involved. For example, when  $\mathbf{s}(t)$  is noncircularly Gaussian distributed,  $\mathbf{x}(t)$  is noncircularly Gaussian distributed as well with complementary covariance

$$\mathbf{C}_x = \mathbf{A}(\boldsymbol{\theta})\mathbf{C}_s\mathbf{A}^T(\boldsymbol{\theta}) \neq \mathbf{0} \quad (16.6)$$

and the distribution of the observations are now characterized by both  $\mathbf{R}_x$  and  $\mathbf{C}_x$ . Consequently, the condition of identifiability will be modified w.r.t. the circular case given in (16.5). This condition has not been presented in the literature, except for the particular case of uncorrelated and rectilinear (called also maximally improper) sources impinging on a ULA for which, the augmented covariance matrix  $\mathbf{R}_{\tilde{x}} \stackrel{\text{def}}{=} \mathbb{E}[\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^H(t)]$  with  $\tilde{\mathbf{x}}(t) \stackrel{\text{def}}{=} [\mathbf{x}^T(t), \mathbf{x}^H(t)]^T$  is given by

$$\mathbf{R}_{\tilde{x}} = \sum_{p=1}^P \sigma_p^2 \mathbf{a}(\theta_p, \phi_p) \mathbf{a}^H(\theta_p, \phi_p) + \sigma_n^2 \mathbf{I}, \quad (16.7)$$

where  $\mathbf{a}(\theta_p, \phi_p) \stackrel{\text{def}}{=} [\mathbf{a}^T(\theta_p), e^{-2i\phi_p} \mathbf{a}^H(\theta_p)]^T$  with  $\phi_p$  is the second-order phase of noncircularity defined by

$$\mathbb{E}[s_p^2(t)] = e^{2i\phi_p} \mathbb{E}[s_p^2(t)] = e^{2i\phi_p} \sigma_p^2. \quad (16.8)$$

Due to the Vandermonde-like structure of the extended steering matrix  $\mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi}) \stackrel{\text{def}}{=} [\mathbf{a}(\theta_1, \phi_1), \dots, \mathbf{a}(\theta_P, \phi_P)]$ , the condition of identifiability is now here  $P < 2M - 1$ .

Note that when  $\mathbf{s}(t)$  is discrete distributed (for example when  $s_p(t)$  are symbols  $s_{p,k(p)}$  of a digital modulation taking  $q$  different values), the condition of identifiability is nontrivial despite the distribution of  $\mathbf{x}(t)$  is a mixture of  $q^P$  circular Gaussian distributions of mean  $\sum_{p=1}^P s_{p,k(p)} \mathbf{a}(\theta_p)$  and covariance  $\sigma_n^2 \mathbf{I}$ .

### 3.16.3 General statistical tools for performance analysis of DOA estimation

#### 3.16.3.1 Performance analysis of a specific algorithm

##### 3.16.3.1.1 Functional analysis

To study the statistical performance of any DOA's estimator (often called an algorithm as a succession of different steps), it is fruitful to adopt a functional analysis that consists in recognizing that the whole process of constructing the estimate  $\hat{\boldsymbol{\theta}}_N$  is equivalent to defining a functional relation linking this estimate to the measurements from which it is inferred. As generally  $\hat{\boldsymbol{\theta}}_N$  are functions of some statistics  $\mathbf{g}_N$  (assumed complex-valued vector in  $\mathbb{C}^L$ ) deduced from  $(\mathbf{x}(t))_{t_1, \dots, t_N}$ , we have the following mapping:

$$\{\mathbf{x}(t)\}_{t_1, \dots, t_N} \longmapsto \mathbf{g}_N \xrightarrow{\text{alg}} \hat{\boldsymbol{\theta}}_N. \quad (16.9)$$

Many often, the statistics  $\mathbf{g}_N$  are sample moments or cumulants of  $\mathbf{x}(t)$ . The most common ones are second-order sample moments of  $\mathbf{x}(t)$  deduced from the sample covariance and complementary covariance matrices  $\mathbf{R}_{x,N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \mathbf{x}(t_n) \mathbf{x}^H(t_n)$  and  $\mathbf{C}_{x,N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \mathbf{x}(t_n) \mathbf{x}^T(t_n)$ , respectively. For non-Gaussian symmetric sources distributions, even sample high-order cumulants of  $\mathbf{x}(t)$  are also used,

in particular the fourth-order sample cumulants deduced from the sample quadrivariance matrices  $\mathbf{Q}_{x,N}$ ,  $\mathbf{Q}'_{x,N}$  and  $\mathbf{Q}''_{x,N}$  where  $[\mathbf{Q}_x]_{i+(j-1)M, k+(l-1)M} \stackrel{\text{def}}{=} \text{Cum}(x_i(t), x_j^*(t), x_k^*(t), x_l(t))$ ,  $[\mathbf{Q}'_x]_{i+(j-1)M, k+(l-1)M} \stackrel{\text{def}}{=} \text{Cum}(x_i(t), x_j^*(t), x_k(t), x_l(t))$  and  $[\mathbf{Q}''_x]_{i+(j-1)M, k+(l-1)M} \stackrel{\text{def}}{=} \text{Cum}(x_i(t), x_j(t), x_k(t), x_l(t))$ , estimated through the associated fourth and second-order sample moments. In these cases, the algorithms are called second-order, high-order and fourth-order algorithms, respectively.

The statistic  $\mathbf{g}_N$  generally satisfies two conditions:

- i.  $\mathbf{g}_N$  converges almost surely (from the strong law of large numbers) to  $E(\mathbf{g}_N)$  when  $N$  tends to infinity, that is a function of the DOAs and other parameters denoted  $\mathbf{g}(\boldsymbol{\theta})$ .
- ii. The DOAs  $\boldsymbol{\theta}$  are identifiable from  $\mathbf{g}(\boldsymbol{\theta})$ , i.e., there exists a mapping  $\mathbf{g}(\boldsymbol{\theta}) \mapsto \boldsymbol{\theta}$ .

Furthermore, we assume that the algorithm **alg** satisfies  $\mathbf{alg}[(\mathbf{g}(\boldsymbol{\theta}))] = \boldsymbol{\theta}$  for all  $\boldsymbol{\theta} \in \Theta$ . Consequently the functional dependence  $\hat{\boldsymbol{\theta}}_N = \mathbf{alg}(\mathbf{g}_N)$  constitutes a particular extension of the mapping  $\mathbf{g}(\boldsymbol{\theta}) \mapsto \boldsymbol{\theta}$  in the neighborhood of  $\mathbf{g}(\boldsymbol{\theta})$  that characterizes all algorithm based on the statistic  $\mathbf{g}_N$ .

Note that for circular Gaussian stochastic and deterministic models of the sources, the likelihood functions of the measurements depend on  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  through only the sample covariance  $\mathbf{R}_{x,N}$ , and therefore the algorithms called respectively stochastic maximum likelihood (SML) and deterministic maximum likelihood (DML) algorithms are second-order algorithms [12]. The SML algorithm has been extended to noncircular Gaussian sources, for which the ML algorithm is built from both  $\mathbf{R}_{x,N}$  and  $\mathbf{C}_{x,N}$  [24].

However, due to their complexity, many suboptimal algorithms with much lower computational requirements have been proposed in the literature. Among them, many algorithms are based on the noise (or signal) orthogonal projector  $\mathbf{\Pi}_{x,N}$  onto the noise (or signal) subspace associated with the sample covariance  $\mathbf{R}_{x,N}$ . These algorithms are called subspace-based algorithms. The most celebrated is the MUSIC algorithm that offers a good trade-off between performance and computational costs. Its statistical performance has been thoroughly studied in the literature (see, e.g., [1, 3, 25, 26]). In these cases, the mapping (16.9) becomes

$$\{\mathbf{x}(t)\}_{t_1, \dots, t_N} \mapsto \mathbf{R}_{x,N} \mapsto \mathbf{\Pi}_{x,N} \xrightarrow{\mathbf{alg}} \hat{\boldsymbol{\theta}}_N, \quad (16.10)$$

where the mapping **alg** characterizes the specific subspace-based algorithm. Some of these algorithms have been extended for noncircular sources through subspace-based algorithms based on  $(\mathbf{\Pi}_{x,N}, \mathbf{\Pi}'_{x,N})$  or  $\mathbf{\Pi}_{\tilde{x},N}$  where  $\mathbf{\Pi}'_{x,N}$  and  $\mathbf{\Pi}_{\tilde{x},N}$  are the orthogonal projectors onto the noise subspace associated with the sample complementary covariance  $\mathbf{C}_{x,N}$  and the sample augmented covariance  $\mathbf{R}_{\tilde{x},N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \tilde{\mathbf{x}}(t_n) \tilde{\mathbf{x}}^H(t_n)$  with  $\tilde{\mathbf{x}}(t_n) \stackrel{\text{def}}{=} (\mathbf{x}^T(t_n), \mathbf{x}^H(t_n))^T$ , respectively [27].

### 3.16.3.1.2 Asymptotic distribution of statistics

Due to the nonlinearity of model (16.1) w.r.t. the DOA's parameter, the performance analysis of detectors for the number of sources and the DOA's estimation procedures are not possible for a finite number  $N$  of snapshots. But in many cases, asymptotic performance analyses are available when the number  $N$  of measurements, the signal-to-noise ratio (SNR) (see, e.g., [28]) or the number of sensors  $M$  converges to infinity (see, e.g., [29]). In practice  $N$ , SNR and  $M$  are naturally finite and thus available results in

the asymptotic regime are approximations, whose domain of validity are specified through Monte Carlo simulations. We will consider in this chapter, only asymptotic properties w.r.t.  $N$  and thus, the presented results will be only valid in practice when  $N \gg M$ . When  $N$  is of the same order of magnitude than  $M$ , although very large, the approximations given by the asymptotic regime w.r.t.  $N$  are generally very bad.

To derive the asymptotic distribution, covariance and bias of estimated DOAs w.r.t. the number  $N$  of measurements, we first need to specify the asymptotic distribution of some statistics  $\mathbf{g}_N$ .

For the second-order statistics

$$\mathbf{g}_N = \text{vec}(\mathbf{R}_{x,N}, \mathbf{C}_{x,N}) = \frac{1}{N} \sum_{n=1}^N \begin{bmatrix} \mathbf{x}^*(t_n) \otimes \mathbf{x}(t_n) \\ \mathbf{x}(t_n) \otimes \mathbf{x}(t_n) \end{bmatrix},$$

where  $\text{vec}(\cdot)$  and  $\otimes$  denote, respectively, the vectorization operator that turns a matrix into a vector by stacking the columns of the matrix one below another and the standard Kronecker product of matrices, closed-form expressions of the covariance  $E[(\mathbf{g}_N - \mathbf{g})(\mathbf{g}_N - \mathbf{g})^H]$  and complementary covariance  $E[(\mathbf{g}_N - \mathbf{g})(\mathbf{g}_N - \mathbf{g})^T]$  matrices (where  $\mathbf{g} \stackrel{\text{def}}{=} \mathbf{g}(\boldsymbol{\theta})$  for short), and their asymptotic distributions<sup>2</sup> have been given [30] for independent measurements, fourth-order arbitrary distributed sources and Gaussian distributed noise:

$$\begin{aligned} E[(\mathbf{g}_N - \mathbf{g})(\mathbf{g}_N - \mathbf{g})^H] &= \frac{1}{N} \begin{pmatrix} \mathbf{R}_{R_x} & \mathbf{R}_{R_x, C_x} \\ \mathbf{R}_{R_x, C_x}^H & \mathbf{R}_{C_x} \end{pmatrix}, \\ E[(\mathbf{g}_N - \mathbf{g})(\mathbf{g}_N - \mathbf{g})^T] &= \frac{1}{N} \begin{pmatrix} \mathbf{C}_{R_x} & \mathbf{C}_{R_x, C_x} \\ \mathbf{C}_{R_x, C_x}^T & \mathbf{C}_{C_x} \end{pmatrix}, \\ \sqrt{N} (\text{vec}(\mathbf{R}_{x,N}, \mathbf{C}_{x,N}) - \text{vec}(\mathbf{R}_x, \mathbf{C}_x)) &\xrightarrow{\mathcal{L}} \mathcal{N}_C \left( \mathbf{0}; \left( \begin{pmatrix} \mathbf{R}_{R_x} & \mathbf{R}_{R_x, C_x} \\ \mathbf{R}_{R_x, C_x}^H & \mathbf{R}_{C_x} \end{pmatrix}, \begin{pmatrix} \mathbf{C}_{R_x} & \mathbf{C}_{R_x, C_x} \\ \mathbf{C}_{R_x, C_x}^T & \mathbf{C}_{C_x} \end{pmatrix} \right) \right), \end{aligned} \quad (16.11)$$

with

$$\begin{aligned} \mathbf{R}_{R_x} &= \mathbf{R}_x^* \otimes \mathbf{R}_x + \mathbf{K}(\mathbf{C}_x \otimes \mathbf{C}_x^*) + (\mathbf{A}^* \otimes \mathbf{A}) \mathbf{Q}_s (\mathbf{A}^T \otimes \mathbf{A}^H), \\ \mathbf{R}_{C_x} &= \mathbf{R}_x \otimes \mathbf{R}_x + \mathbf{K}(\mathbf{R}_x \otimes \mathbf{R}_x) + (\mathbf{A} \otimes \mathbf{A}) \mathbf{Q}_s''' (\mathbf{A}^H \otimes \mathbf{A}^H), \\ \mathbf{C}_{R_x} &= \mathbf{R}_{R_x} \mathbf{K}, \\ \mathbf{C}_{C_x} &= \mathbf{C}_x \otimes \mathbf{C}_x + \mathbf{K}(\mathbf{C}_x \otimes \mathbf{C}_x) + (\mathbf{A} \otimes \mathbf{A}) \mathbf{Q}_s'' (\mathbf{A}^T \otimes \mathbf{A}^T), \\ \mathbf{R}_{R_x, C_x} &= \mathbf{C}_x^* \otimes \mathbf{R}_x + \mathbf{K}(\mathbf{R}_x \otimes \mathbf{C}_x^*) + (\mathbf{A}^* \otimes \mathbf{A}) \mathbf{Q}_s'''' (\mathbf{A}^H \otimes \mathbf{A}^H), \\ \mathbf{C}_{R_x, C_x} &= \mathbf{R}_x^* \otimes \mathbf{C}_x + \mathbf{K}(\mathbf{C}_x \otimes \mathbf{R}_x^*) + (\mathbf{A}^* \otimes \mathbf{A}) \mathbf{Q}_s' (\mathbf{A}^T \otimes \mathbf{A}^T), \end{aligned} \quad (16.12)$$

where  $\mathbf{A} \stackrel{\text{def}}{=} \mathbf{A}(\boldsymbol{\theta})$  for short and  $\mathbf{K}$  denotes the vec-permutation matrix which transforms  $\text{vec}(\mathbf{C})$  to  $\text{vec}(\mathbf{C}^T)$  for any square matrix  $\mathbf{C}$ .  $\mathbf{Q}_s$ ,  $\mathbf{Q}_s'$ , and  $\mathbf{Q}_s''$  are defined as for  $\mathbf{x}(t)$  defined previously and

---

<sup>2</sup>Throughout this chapter  $\mathcal{N}_R(\mathbf{m}; \mathbf{R})$ ,  $\mathcal{N}_C(\mathbf{m}; \mathbf{R})$  and  $\mathcal{N}_C(\mathbf{m}; \mathbf{R}, \mathbf{C})$  denote the real, circular complex, arbitrary complex Gaussian distribution, respectively, with mean  $\mathbf{m}$ , covariance  $\mathbf{R}$  and complementary covariance  $\mathbf{C}$ .

$[\mathbf{Q}_s''']_{i+(j-1)P, k+(l-1)P} \stackrel{\text{def}}{=} \text{Cum}(s_i(t), s_j(t), s_k^*(t), s_l^*(t))$ ,  $[\mathbf{Q}_s''']_{i+(j-1)P, k+(l-1)P} \stackrel{\text{def}}{=} \text{Cum}(s_i(t), s_j^*(t), s_k^*(t), s_l^*(t))$ . Note that the asymptotic distribution of  $\mathbf{R}_{x,N}$  has been extended to non independent measurements with arbitrary distributed sources and noise of finite fourth-order moments with  $\mathbf{R}_n$  arbitrarily structured in [20, 31].

Consider now the noise orthogonal projector  $\mathbf{g}_N = \text{vec}(\boldsymbol{\Pi}_{x,N})$ . Its asymptotic distribution is deduced from the standard first-order perturbation for orthogonal projectors [32] (see also [6]):

$$\delta(\boldsymbol{\Pi}_{x,N}) = -\boldsymbol{\Pi}_x \delta(\mathbf{R}_{x,N}) \mathbf{S}^\# - \mathbf{S}^\# \delta(\mathbf{R}_{x,N}) \boldsymbol{\Pi}_x + o(\delta(\mathbf{R}_{x,N})), \quad (16.13)$$

where  $\delta(\boldsymbol{\Pi}_{x,N}) \stackrel{\text{def}}{=} \boldsymbol{\Pi}_{x,N} - \boldsymbol{\Pi}_x$ ,  $\delta(\mathbf{R}_{x,N}) \stackrel{\text{def}}{=} \mathbf{R}_{x,N} - \mathbf{R}_x$  and  $\mathbf{S}^\#$  is the Moore-Penrose inverse of  $\mathbf{S} = \mathbf{A}(\boldsymbol{\theta}) \mathbf{R}_s \mathbf{A}^H(\boldsymbol{\theta})$ . The remainder in (16.13) is a standard  $o(\delta(\mathbf{R}_{x,N}))$  for a realization of the random matrix  $\mathbf{R}_{x,N}$ , but an  $o_p(\delta(\mathbf{R}_{x,N}))$  if  $\mathbf{R}_{x,N}$  is considered as random. The relation (16.13) proves that  $\mathbf{g}_N$  is differentiable w.r.t.  $\text{vec}(\mathbf{R}_{x,N})$  in the neighborhood of  $\text{vec}(\mathbf{R}_x)$  and its differential matrix (called also Jacobian matrix) evaluated at  $\text{vec}(\mathbf{R}_x)$  is

$$\mathbf{D}_{R_x, \boldsymbol{\Pi}_x} = -\left( \mathbf{S}^{*\#} \otimes \boldsymbol{\Pi}_x + \boldsymbol{\Pi}_x^* \otimes \mathbf{S}^\# \right). \quad (16.14)$$

Then using the standard theorem of continuity (see, e.g., [33, Theorem B, p. 124]) on regular functions of asymptotically Gaussian statistics, the asymptotic behaviors of  $\boldsymbol{\Pi}_{x,N}$  and  $\mathbf{R}_{x,N}$  are directly related:

$$\sqrt{N} (\text{vec}(\boldsymbol{\Pi}_{x,N}) - \text{vec}(\boldsymbol{\Pi}_x)) \xrightarrow{\mathcal{L}} \mathcal{N}_C(\mathbf{0}; \mathbf{R}_{\boldsymbol{\Pi}_x}, \mathbf{R}_{\boldsymbol{\Pi}_x} \mathbf{K}), \quad (16.15)$$

where  $\mathbf{R}_{\boldsymbol{\Pi}_x}$  is given for independent measurements, fourth-order arbitrary distributed sources and Gaussian distributed noise, using (16.12) by

$$\mathbf{R}_{\boldsymbol{\Pi}_x} = \mathbf{D}_{R_x, \boldsymbol{\Pi}_x} \mathbf{R}_{R_x} \mathbf{D}_{R_x, \boldsymbol{\Pi}_x}^H = \boldsymbol{\Pi}_x^* \otimes \mathbf{U} + \mathbf{U}^* \otimes \boldsymbol{\Pi}_x, \quad (16.16)$$

with  $\mathbf{U} = \sigma_n^2 \mathbf{S}^\# \mathbf{R}_x \mathbf{S}^\#$ . We see that  $\mathbf{R}_{\boldsymbol{\Pi}_x}$  does not depend on  $\mathbf{C}_s$  and the quadravariances of the sources. Consequently, all subspace-based algorithms are robust to the distribution and to the noncircularity of the sources; i.e., the asymptotic performances are those of the standard complex circular Gaussian case. Note that the asymptotic distribution of  $(\boldsymbol{\Pi}_{x,N}, \boldsymbol{\Pi}'_{x,N})$  and  $\boldsymbol{\Pi}_{\tilde{x},N}$  have also been derived under the same assumptions in [27], where it is proved that they do not depend on the quadravariances of the sources, as well. The asymptotic distributions of  $\boldsymbol{\Pi}_{x,N}$ ,  $(\boldsymbol{\Pi}_{x,N}, \boldsymbol{\Pi}'_{x,N})$  and  $\boldsymbol{\Pi}_{\tilde{x},N}$  will allow us to derive the statistical performance of arbitrary subspace-based algorithms based on these orthogonal projectors in Section 3.16.4.4.

Note that the second-order expansion of  $\boldsymbol{\Pi}_{x,N}$  w.r.t.  $\mathbf{R}_{x,N}$  has been used in [6] to analyze the behavior of the root-MUSIC and root-min-norm algorithms dedicated to ULA, but is useless as far as we are concerned by the asymptotic distribution of the DOAs alone, as it has been specified in [27], where an extension of the root-MUSIC algorithm to noncircular sources has been proposed.

Finally, consider now the asymptotic distribution of the signal eigenvalues of  $\mathbf{R}_{x,N}$  that is useful for the statistical performance analysis of information theoretic criteria (whose MDL criterion popularized by Wax and Kailath [13] is one of the most successful), for the detection of the number  $P$  of sources. Let  $\lambda_1, \dots, \lambda_P, \lambda_{P+1} = \sigma_n^2, \dots, \lambda_M = \sigma_n^2$  denote the eigenvalues of  $\mathbf{R}_x$ , ordered in decreasing order and

$\mathbf{v}_1, \dots, \mathbf{v}_P$  the associated eigenvectors (defined up to a multiplicative unit modulus complex number) of the signal subspace. Then, suppose that for a “small enough” perturbation  $\mathbf{R}_{x,N} - \mathbf{R}_x$ , the largest  $P$  associated eigenvalues of the sample covariance  $\mathbf{R}_{x,N}$  are  $\lambda_{1,N} > \dots > \lambda_{P,N}$ . It is proved in [14], extending the work by Kaveh and Barabell [1] to arbitrary distributed independent measurements (16.1) with finite fourth-order moment, not necessarily circular and Gaussian, the following convergence in distribution:

$$\sqrt{N}(\boldsymbol{\lambda}_N - \boldsymbol{\lambda}) \xrightarrow{\mathcal{L}} \mathcal{N}_R(\mathbf{0}; \mathbf{R}_\lambda), \quad (16.17)$$

with  $\boldsymbol{\lambda}_N = [\lambda_{1,N}, \dots, \lambda_{P,N}]^T$ ,  $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_P]^T$ , and  $[\mathbf{R}_\lambda]_{i,j} = \lambda_i^2 \delta_{i,j} + |\lambda_{i,j}|^2 + \lambda_{i,i,j,j}$  for  $i, j = 1, \dots, P$ ,  $\delta_{i,j}$  is the Kronecker delta,  $\lambda_{i,j} \stackrel{\text{def}}{=} \mathbf{v}_i^H \mathbf{C}_x \mathbf{v}_j^*$  and  $\lambda_{i,j,k,l} \stackrel{\text{def}}{=} (\mathbf{v}_i^T \otimes \mathbf{v}_j^H) \mathbf{Q}_x (\mathbf{v}_k^* \otimes \mathbf{v}_l)$ . In contrast to the circular Gaussian distribution [1], we see that the estimated eigenvalues  $\{\lambda_{i,N}\}_{i=1,\dots,P}$  are no longer asymptotically mutually independent. Furthermore, it is proved in [14] that for  $i, j = 1, \dots, P$ :

$$E[\lambda_{i,N}] = \lambda_i + \frac{1}{N} \sum_{1 \leq k \neq i \leq M} \frac{\lambda_i \lambda_k + |\lambda_{i,k}|^2 + \lambda_{i,k,i,k}}{\lambda_i - \lambda_k} + o\left(\frac{1}{N}\right), \quad (16.18)$$

$$\text{Cov}[\lambda_{i,N}, \lambda_{j,N}] = \frac{1}{N} (\lambda_i^2 \delta_{i,j} + |\lambda_{i,j}|^2 + \lambda_{i,i,j,j}) + o\left(\frac{1}{N}\right). \quad (16.19)$$

We note that these results are also valid for the augmented covariance matrix  $\mathbf{R}_{\tilde{x},N}$  where  $M$  and  $P$  are replaced by  $2M$  and the rank of  $\mathbf{R}_{\tilde{x},N} - \sigma_n^2 \mathbf{I}$ , respectively.

### 3.16.3.1.3 Asymptotic distribution of estimated DOA

In the following, we consider arbitrary DOA algorithms that are in practice “regular” enough.<sup>3</sup> More specifically, we assume that the mapping **alg** is  $\mathbb{R}$ -differentiable w.r.t.  $\mathbf{g}_N \in \mathbb{C}^L$  in the neighborhood of  $\mathbf{g}(\theta)$ , i.e.,

$$\hat{\boldsymbol{\theta}}_N = \mathbf{alg}(\mathbf{g}_N) = \mathbf{alg}(\mathbf{g}) + \mathbf{D}_{g,\theta}^{\text{alg}}(\mathbf{g}_N - \mathbf{g}) + \mathbf{D}_{g,\theta}^{\text{alg}*}(\mathbf{g}_N - \mathbf{g})^* + o\|\mathbf{g}_N - \mathbf{g}\|, \quad (16.20)$$

with  $\mathbf{alg}(\mathbf{g}) = \boldsymbol{\theta}$  and  $P \times L$  matrix  $\mathbf{D}_{g,\theta}^{\text{alg}}$  is the  $\mathbb{R}$ -differential matrix (Jacobian) of the mapping  $\mathbf{g}_N \xrightarrow{\text{alg}} \hat{\boldsymbol{\theta}}_N$  evaluated at  $\mathbf{g}(\theta)$ . In practice, this matrix is derived from the chain rule by decomposing the algorithm as successive simpler mappings, and in each of these mapping, this matrix is simply deduced from first-order expansions. Then, applying a simple extension of the standard theorem of continuity [33, Theorem B, p. 124] (also called  $\Delta$ -method), it is straightforwardly proved the following convergence in distribution:

$$\sqrt{N}(\hat{\boldsymbol{\theta}}_N - \boldsymbol{\theta}) \xrightarrow{\mathcal{L}} \mathcal{N}_R(\mathbf{0}; \mathbf{R}_\theta) \quad \text{with} \quad \mathbf{R}_\theta = 2 \left[ \mathbf{D}_{g,\theta}^{\text{alg}} \mathbf{R}_g \left( \mathbf{D}_{g,\theta}^{\text{alg}} \right)^H + \text{Re} \left( \mathbf{D}_{g,\theta}^{\text{alg}} \mathbf{C}_g \left( \mathbf{D}_{g,\theta}^{\text{alg}} \right)^T \right) \right], \quad (16.21)$$

where  $\mathbf{R}_g$  and  $\mathbf{C}_g$  are the covariance and the complementary covariance matrices of the asymptotic distribution of the statistics  $\mathbf{g}_N$ . We note that for subspace-based algorithms and second-order algorithms

---

<sup>3</sup>This is the case, for example when  $\hat{\boldsymbol{\theta}}_N$  maximizes w.r.t.  $\boldsymbol{\alpha}$ , a real-valued function  $f(\boldsymbol{\alpha}, \mathbf{g}_N)$  that is twice- $\mathbb{R}$  differentiable w.r.t.  $\boldsymbol{\alpha}$  and  $\mathbf{g}_N$ .

based on  $\mathbf{R}_{x,N}$  or  $\mathbf{R}_{\tilde{x},N}$ ,  $\mathbf{g}_N^* = \mathbf{K}\mathbf{g}_N$  (because the orthogonal projector matrices and the covariance matrices are Hermitian structured), and generally for statistics  $\mathbf{g}_N$  that contain all conjugate of its components, the mapping  $\mathbf{alg}$  is  $\mathbb{C}$ -differentiable w.r.t.  $\mathbf{g}_N$  in the neighborhood of  $\mathbf{g}(\theta)$  and (16.20) and (16.21) become respectively:

$$\hat{\boldsymbol{\theta}}_N = \mathbf{alg}(\mathbf{g}_N) = \mathbf{alg}(\mathbf{g}) + \mathbf{D}_{g,\theta}^{\text{alg}}(\mathbf{g}_N - \mathbf{g}) + o\|\mathbf{g}_N - \mathbf{g}\|, \quad (16.22)$$

where now,  $\mathbf{D}_{g,\theta}^{\text{alg}}$  is the  $\mathbb{C}$ -differential matrix of the mapping  $\mathbf{g}_N \xrightarrow{\text{alg}} \hat{\boldsymbol{\theta}}_N$  evaluated at  $\mathbf{g}(\theta)$  and

$$\sqrt{N}(\hat{\boldsymbol{\theta}}_N - \boldsymbol{\theta}) \xrightarrow{\mathcal{L}} \mathcal{N}_R(\mathbf{0}; \mathbf{R}_\theta) \quad \text{with } \mathbf{R}_\theta = \mathbf{D}_{g,\theta}^{\text{alg}} \mathbf{R}_g \left( \mathbf{D}_{g,\theta}^{\text{alg}} \right)^H. \quad (16.23)$$

### 3.16.3.1.4 Asymptotic covariance and bias

Under additional regularities of the algorithm  $\mathbf{alg}$ , that are generally satisfied, the covariance of  $\hat{\boldsymbol{\theta}}_N$  is given by

$$\text{Cov}(\hat{\boldsymbol{\theta}}_N) = \frac{1}{N} \mathbf{R}_\theta + o\left(\frac{1}{N}\right). \quad (16.24)$$

Using a second-order expansion of  $\mathbf{alg}(\mathbf{g}_N)$  and  $\mathbb{CR}$ -calculus, where  $\mathbf{alg}$  is assumed to be twice- $\mathbb{R}$ -differentiable, the bias is given by

$$E(\hat{\boldsymbol{\theta}}_N) - \boldsymbol{\theta} = \frac{1}{2N} \begin{bmatrix} \text{Tr}(\mathbf{R}_{\tilde{g}} \mathbf{H}_{\tilde{g},\theta,1}^{\text{alg}}) \\ \vdots \\ \text{Tr}(\mathbf{R}_{\tilde{g}} \mathbf{H}_{\tilde{g},\theta,P}^{\text{alg}}) \end{bmatrix} + o\left(\frac{1}{N}\right), \quad (16.25)$$

where  $\mathbf{H}_{\tilde{g},\theta,k}^{\text{alg}} = \frac{\partial}{\partial \tilde{g}} \left( \frac{\partial \mathbf{alg}}{\partial \tilde{g}} \right)^H = \begin{bmatrix} \mathbf{H}_{g,\theta,k}^{(1)} & \mathbf{H}_{g,\theta,k}^{(2)} \\ \mathbf{H}_{g,\theta,k}^{(2)*} & \mathbf{H}_{g,\theta,k}^{(1)*} \end{bmatrix}$  is the complex augmented Hessian matrix [34,

A2.3] of the  $k$ th component of the function  $\mathbf{alg}$  at point  $\mathbf{g}(\theta)$  and  $\mathbf{R}_{\tilde{g}} = \begin{bmatrix} \mathbf{R}_g & \mathbf{C}_g \\ \mathbf{C}_g^* & \mathbf{R}_g^* \end{bmatrix}$  is the augmented covariance of the asymptotic distribution of  $\mathbf{g}_N$ . In the particular case where  $\mathbf{alg}$  is twice- $\mathbb{C}$ -differentiable (see, e.g., the examples given for  $\mathbb{C}$ -differentiable algorithms (16.22)), i.e.,

$$\hat{\boldsymbol{\theta}}_N = \mathbf{alg}(\mathbf{g}_N) = \mathbf{alg}(\mathbf{g}) + \mathbf{D}_{g,\theta}^{\text{alg}}(\mathbf{g}_N - \mathbf{g}) + \frac{1}{2} [\mathbf{I}_P \otimes (\mathbf{g}_N - \mathbf{g})^H] \begin{bmatrix} \mathbf{H}_{g,\theta,1}^{\text{alg}} \\ \vdots \\ \mathbf{H}_{g,\theta,P}^{\text{alg}} \end{bmatrix} [\mathbf{g}_N - \mathbf{g}] + o\|\mathbf{g}_N - \mathbf{g}\|^2, \quad (16.26)$$

(16.25) reduces to

$$E(\hat{\boldsymbol{\theta}}_N) - \boldsymbol{\theta} = \frac{1}{2N} \begin{bmatrix} \text{Tr}(\mathbf{R}_g \mathbf{H}_{g,\theta,1}^{\text{alg}}) \\ \vdots \\ \text{Tr}(\mathbf{R}_g \mathbf{H}_{g,\theta,P}^{\text{alg}}) \end{bmatrix} + o\left(\frac{1}{N}\right). \quad (16.27)$$

We note that relations (16.24), (16.25) and (16.27) are implicitly used in the signal processing literature by simple first and second-order expansions of the estimate  $\hat{\theta}_N$  w.r.t. the involved statistics without checking any necessary mathematical conditions concerning the remainder terms of the first and second-order expansions. In fact these conditions are very difficult to prove for the involved mappings  $\mathbf{g}_N \xrightarrow{\text{alg}} \hat{\theta}_N$ . For example, the following necessary conditions are given in [35, Theorem 4.2.2] for second-order algorithms: (i) the measurements  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  are independent with finite eighth moments, (ii) the mapping  $\mathbf{g}_N \xrightarrow{\text{alg}} \hat{\theta}_N$  is four times  $\mathbb{R}$ -differentiable, (iii) the fourth derivative of this mapping and those of its square are bounded. These assumptions that do not depend on the distribution of the measurements are very strong, but fortunately (16.24), (16.25) and (16.27) continue to hold in many cases in which these assumptions are not satisfied, in particular for Gaussian distributed data (see, e.g., [35, Example 4.2.2]).

In practice, (16.24), (16.25), and (16.27) show that the mean square error (MSE)

$$\mathbb{E}\|\hat{\theta}_N - \theta\|^2 = \|\mathbb{E}(\hat{\theta}_N) - \theta\|^2 + \text{Tr}[\text{Cov}(\hat{\theta}_N)] \quad (16.28)$$

is then also of order  $1/N$ . Its main contribution comes from the variance term, since the square of the bias is of order  $1/N^2$ . But as empirically observed, this bias contribution may be significant when SNR or  $N$  is not sufficiently large. However, there are very few contributions in the literature, that have derived closed-form bias expressions. Among them, Xu and Cave [36] has considered the bias of the MUSIC algorithm, whose derivation ought to be simplified by using the asymptotic distribution of the orthogonal projector  $\Pi_{x,N}$ , rather than those of the sample signal eigenvectors  $(\mathbf{e}_{1,N}, \dots, \mathbf{e}_{P,N})$ .

### 3.16.3.2 Cramer-Rao bounds (CRB)

The accuracy measures of performance in terms of covariance and bias of any algorithm, described in the previous section may be of limited interest, unless one has an idea of what the best possible performance is. An important measure of how well a particular DOA finding algorithm performs is the mean square error (MSE) matrix  $\mathbb{E}[(\hat{\theta} - \theta)(\hat{\theta} - \theta)^T]$  of the estimation error  $\hat{\theta}_N - \theta$ . Among the lower bounds on this matrix, the celebrated Cramer-Rao bound (CRB) is by far the most commonly used. We note that this CRB is indeed deduced from the CRB on the complete unknown parameter  $\alpha$  of the parametrized DOA model, for example, given by (16.4) for the circular Gaussian stochastic model. Furthermore, rigorously speaking, this CRB ought to be only used for unbiased estimators and under sufficiently regular distributions of the measurements. Fortunately, these technical conditions are satisfied in practice and due to the property that the bias contribution is often weak w.r.t. the variance term in the mean square error (16.28) for  $N \gg 1$ , the CRB that lower bounds the covariance matrix of any unbiased estimators is used to lower bound the MSE matrix of any asymptotically unbiased estimator<sup>4</sup>

$$\mathbb{E}[(\hat{\alpha} - \alpha)(\hat{\alpha} - \alpha)^T] \geq \text{CRB}(\alpha) \quad (16.29)$$

with  $\text{CRB}(\alpha)$  is given under weak regularity conditions by:

$$\text{CRB}(\alpha) = \mathbf{FIM}^{-1}(\alpha), \quad (16.30)$$

---

<sup>4</sup>Note that for for finite  $N$ , the estimator  $\hat{\alpha}$  is always biased and (16.29) does not apply. Additionally, biased estimators may exist whose MSE matrices are smaller than the CRB (see, e.g., [37]).

where  $\mathbf{FIM}(\boldsymbol{\alpha})$  is the Fisher information matrix (FIM) given elementwise by

$$[\mathbf{FIM}(\boldsymbol{\alpha})]_{k,l} = -\mathbb{E} \left[ \left( \frac{\partial^2 \log p(\mathbf{x}; \boldsymbol{\alpha})}{\partial \alpha_k \partial \alpha_l} \right) \right] \quad (16.31)$$

associated with the probability density function  $p(\mathbf{x}; \boldsymbol{\alpha})$  of the measurements  $\mathbf{x} = [\mathbf{x}^T(t_1), \dots, \mathbf{x}^T(t_N)]^T$ .

The main reason for the interest of this CRB is that it is often asymptotically (when the amount  $N$  of data is large) tight, i.e., there exist algorithms, such that the stochastic maximum likelihood (ML) estimator (see 3.16.4.2), whose covariance matrices asymptotically achieve this bound. Such estimators are said to be asymptotically efficient. However, at low SNR and/or at low number  $N$  of snapshots, the CRB is not achieved and is overly optimistic. This is due to the fact that estimators are generally biased in such non-asymptotic cases. For these reasons, other lower bounds are available in the literature, that are more relevant to lower bound the MSE matrices. But unfortunately, their closed-form expressions are much more complex to derive and are generally non interpretable (see, e.g., the Weiss-Weinstein bound in [38]).

In practice, closed-form expressions of the FIM (16.31) are difficult to obtain for arbitrary distributions of the sources and noise. In general, the involved integrations of (16.31) are solved numerically by replacing the expectations by arithmetical averages over a large number of computer generated measurements. But for Gaussian distributions, there are a plethora of closed-form expressions of  $\text{CRB}(\boldsymbol{\theta})$  in the literature. And the reason of the popularity of this CRB is the simplicity of the FIM for Gaussian distributions of  $\mathbf{x}$ .

### 3.16.3.2.1 Gaussian stochastic case

One way to derive closed-form expressions of  $\text{CRB}(\boldsymbol{\theta})$  is to use the extended Slepian-Bangs [39, 40] formula, where the FIM (16.31) is given elementwise by

$$[\mathbf{FIM}(\boldsymbol{\alpha})]_{k,l} = 2\text{Re} \left[ \left( \frac{\partial \mathbf{m}_x}{\partial \alpha_k} \right)^H \mathbf{R}_x^{-1} \frac{\partial \mathbf{m}_x}{\partial \alpha_l} \right] + \text{Tr} \left[ \frac{\partial \mathbf{R}_x}{\partial \alpha_k} \mathbf{R}_x^{-1} \frac{\partial \mathbf{R}_x}{\partial \alpha_l} \mathbf{R}_x^{-1} \right] \quad (16.32)$$

for a circular<sup>5</sup> Gaussian  $\mathcal{N}_C(\mathbf{m}_x; \mathbf{R}_x)$  distribution of  $\mathbf{x}$ . But there are generally difficulties to derive compact matrix expressions of the CRB for DOA parameters alone given by

$$\text{CRB}(\boldsymbol{\theta}) = [\mathbf{FIM}^{-1}(\boldsymbol{\alpha})]_{(1:P, 1:P)}$$

with  $\boldsymbol{\alpha} = (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T)^T$  where  $\boldsymbol{\beta}$  gathers all the nuisance parameters (in many applications, only the DOAs are of interest). Another way, based on the asymptotic efficiency of the ML estimator (under certain regularity conditions) has been used to indirectly derive the CRB on the DOA parameter alone (see 3.16.4.2).

For the circular Gaussian stochastic model of the sources introduced in Section 3.16.2.2, compact matrix expressions of  $\text{CRB}(\boldsymbol{\theta})$  have been given in the literature, when no a priori information is available on the structure of the spatial covariance  $\mathbf{R}_s$  of the sources. For example, Stoica et al. [41] have derived

---

<sup>5</sup>Note that this Slepian-Bangs formula has been extended to noncircular Gaussian  $\mathcal{N}_C(\mathbf{m}_x; \mathbf{R}_x, \mathbf{C}_x)$  distribution in [42] where (16.32) becomes  $[\mathbf{FIM}(\boldsymbol{\alpha})]_{k,l} = \left( \frac{\partial \mathbf{m}_{\tilde{x}}}{\partial \alpha_k} \right)^H \mathbf{R}_{\tilde{x}}^{-1} \frac{\partial \mathbf{m}_{\tilde{x}}}{\partial \alpha_l} + \frac{1}{2} \text{Tr} \left[ \frac{\partial \mathbf{R}_{\tilde{x}}}{\partial \alpha_k} \mathbf{R}_{\tilde{x}}^{-1} \frac{\partial \mathbf{R}_{\tilde{x}}}{\partial \alpha_l} \mathbf{R}_{\tilde{x}}^{-1} \right]$  with  $\mathbf{m}_{\tilde{x}} \stackrel{\text{def}}{=} (\mathbf{m}_x^T, \mathbf{m}_x^H)^T$  and  $\mathbf{R}_{\tilde{x}} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{R}_x & \mathbf{C}_x \\ \mathbf{C}_x^* & \mathbf{R}_x^* \end{bmatrix}$ .

the following expression for one parameter per source and uniform white noise (i.e.,  $\mathbf{R}_n = \sigma_n^2 \mathbf{I}$ )

$$\text{CRB}_{\text{CG}}(\boldsymbol{\theta}) = \frac{\sigma_n^2}{2N} \left\{ \text{Re} \left[ (\mathbf{D}^H \boldsymbol{\Pi}_x \mathbf{D}) \odot \left( \mathbf{R}_s \mathbf{A}^H \mathbf{R}_s^{-1} \mathbf{A} \mathbf{R}_s \right)^T \right] \right\}^{-1}, \quad (16.33)$$

where  $\odot$  denotes the Hadamard product (i.e., element-wise multiplication),  $\boldsymbol{\Pi}_x$  is the orthogonal projector on the noise subspace, i.e.,  $\boldsymbol{\Pi}_x = \boldsymbol{\Pi}_A^\perp \stackrel{\text{def}}{=} \mathbf{I} - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  and  $\mathbf{D} \stackrel{\text{def}}{=} \left[ \frac{d\mathbf{a}(\theta_1)}{d\theta_1}, \dots, \frac{d\mathbf{a}(\theta_P)}{d\theta_P} \right]$ . We note the surprising fact that when the sources are known to be coherent (i.e.,  $\mathbf{R}_s$  singular), the associated Gaussian CRB  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  that includes this prior, keeps the same expression (16.33) [43].

As is well known, the importance of this Gaussian CRB formula lies in the fact that circular Gaussian data are rather frequently encountered in applications. Another important point is that under rather general conditions that will be specified in Section 3.16.4.2, the circular complex Gaussian CRB matrix (16.33) is the largest of all CRB matrices among the class of arbitrary complex distributions of the sources with given covariance matrix  $\mathbf{R}_x$  (see, e.g., [21, p. 293]). Note that many extensions of (16.33) have been given. For example this formula has been extended to several parameters per source (see, e.g., [44, Appendix D]), to nonuniform white noise (i.e.,  $\mathbf{R}_n = \text{Diag}[\sigma_1^2, \dots, \sigma_M^2]$ ) and unknown parameterized noise field (i.e.,  $\mathbf{R}_n = \boldsymbol{\Sigma}(\boldsymbol{\sigma})$ ) in [45–47], respectively. Due to the domination of the Gaussian distribution, these bounds have often been denoted in the literature as stochastic CRB (e.g., in [10]) or unconditional CRB (e.g., in [11]), without specifying the involved distribution.

Furthermore, all these closed-form expressions of the CRB have been extended to the noncircular Gaussian stochastic model of the sources in [42, 44, 48, Appendix D], given associated  $\text{CRB}_{\text{NCG}}(\boldsymbol{\theta})$  expressions satisfying

$$\text{CRB}_{\text{NCG}}(\boldsymbol{\theta}) \leq \text{CRB}_{\text{CG}}(\boldsymbol{\theta})$$

corresponding to the same covariance matrix  $\mathbf{R}_s$ . For example, for a single source, with one parameter  $\theta_1$ ,  $\text{CRB}_{\text{NCG}}(\theta_1)$  decreases monotonically as the second-order noncircularity rate  $\gamma_1$  (defined by  $E|s_1^2(t)| = \gamma_1 e^{2i\phi_1} E[s_1^2(t)]$  and satisfying  $0 \leq \gamma_1 \leq 1$ ) increases from 0 to 1, for which we have, respectively,

$$\begin{aligned} \text{CRB}_{\text{CG}}(\theta_1) &= \frac{1}{N} \left( \frac{1}{h_1} \left[ \frac{\sigma_n^2}{\sigma_1^2} + \frac{1}{\|\mathbf{a}(\theta_1)\|^2} \frac{\sigma_n^4}{\sigma_1^4} \right] \right), \\ \text{CRB}_{\text{NCG}}(\theta_1) &= \frac{1}{N} \left( \frac{1}{h_1} \left[ \frac{\sigma_n^2}{\sigma_1^2} + \frac{1}{2\|\mathbf{a}(\theta_1)\|^2} \frac{\sigma_n^4}{\sigma_1^4} \right] \right), \end{aligned} \quad (16.34)$$

where  $h_1$  is the purely geometrical factor  $2 \frac{d\mathbf{a}^H(\theta_1)}{d\theta_1} \boldsymbol{\Pi}_{\mathbf{a}_1}^\perp \frac{d\mathbf{a}(\theta_1)}{d\theta_1}$  with  $\boldsymbol{\Pi}_{\mathbf{a}_1}^\perp \stackrel{\text{def}}{=} \mathbf{I}_M - \frac{\mathbf{a}(\theta_1)\mathbf{a}^H(\theta_1)}{\|\mathbf{a}(\theta_1)\|^2}$ .

If the source covariance  $\mathbf{R}_s$  is constrained to have a specific structure, (i.e., if a prior on  $\mathbf{R}_s$  is taken into account), a specific expression of  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$ , which integrates this prior ought to be derived, to assess the performance of an algorithm that uses this prior. But unfortunately, the derivation of  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  is very involved and lacks any engineering insight. For example, when it is known that the sources are uncorrelated, the expression given in [49, Theorem 1] of  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  includes a matrix  $\mathbf{B}$ , defined as any matrix, whose columns span the null space of  $[\mathbf{a}^*(\theta_1) \otimes \mathbf{a}(\theta_1), \dots, \mathbf{a}^*(\theta_P) \otimes \mathbf{a}(\theta_P)]^H$ . And to the best of our knowledge no closed-form expression of  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  has been published in the important case of coherent sources, when the rank of  $\mathbf{R}_s$  is fixed strictly smaller than  $P$ .

Finally, note that the scalar field modeling one component of electromagnetic field or acoustic pressure (16.1) has been extended to vector fields with vector sensors, where associated stochastic CRBs for the DOA (azimuth and elevation) alone have been derived and analyzed for a single source. In particular, the electromagnetic (six electric and magnetic field components) and acoustic (three velocity components and pressure) fields have been considered in [50,51], respectively.

### 3.16.3.2.2 Gaussian deterministic case

For the deterministic model of the sources introduced in Section 3.16.2.2, the unknown parameter  $\boldsymbol{\alpha}$  of  $\mathbf{R}_x$  is now

$$\boldsymbol{\alpha} = \left[ \theta_1, \dots, \theta_P, \left\{ \text{Re}[\mathbf{s}^T(t_n)], \text{Im}[\mathbf{s}^T(t_n)] \right\}_{n=1,\dots,N}, \sigma_n^2 \right]^T. \quad (16.35)$$

Applying the extended Slepian-Bangs formula (16.32) to the circular Gaussian  $\mathcal{N}_C \left( \begin{bmatrix} \mathbf{As}(t_1) \\ \vdots \\ \mathbf{As}(t_N) \end{bmatrix}; \sigma_n^2 \mathbf{I}_{NM} \right)$

distribution of  $\mathbf{x}$ , Stoica and Nehorai [11] have obtained the following CRB for the DOA alone:  $\text{CRB}_{\text{Det}}(\boldsymbol{\theta}) = \frac{\sigma_n^2}{2N} \left\{ \text{Re} [\mathbf{D}^H \boldsymbol{\Pi}_x \mathbf{D} \odot \mathbf{R}_{s,N}] \right\}^{-1}$ , where  $\mathbf{R}_{s,N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \mathbf{s}(t_n) \mathbf{s}^H(t_n)$ . Furthermore, it was proved in [3] that  $\text{CRB}_{\text{Det}}(\boldsymbol{\theta})$  decreases monotonically with increasing  $N$  (and  $M$ ). This implies, that if the sources  $\mathbf{s}(t_n)$  are second-order ergodic sequences,  $\mathbf{R}_{s,N}$  has a limit  $\mathbf{R}_s$  when  $N$  tends to infinity, and we obtain for large  $N$ , the following expression denoted in the literature as deterministic CRB or conditional CRB (e.g., in [11])

$$\text{CRB}_{\text{Det}}(\boldsymbol{\theta}) \approx \frac{\sigma_n^2}{2N} \left\{ \text{Re} [(\mathbf{D}^H \boldsymbol{\Pi}_x \mathbf{D}) \odot \mathbf{R}_s] \right\}^{-1}. \quad (16.36)$$

Finally, we remark that the CRB for near-field DOA localization has been much less studied than the far-field one. To the best of our knowledge, only papers [52–54] have given and analyzed closed-form expressions of the stochastic and deterministic CRB, and furthermore in the particular case of a single source for specific arrays. For a ULA where the DOA parameters are the azimuth  $\theta$  and the range  $r$ , based on the DOA algorithms, the steering vector (16.2) has been approximated in [53] by

$$[\mathbf{a}(\theta, r)]_{m=1,\dots,M} = e^{i(\omega(m-1) + \phi(m-1)^2)},$$

where  $\omega$  and  $\phi$  are the so-called electric angles connected to the physical parameters  $\theta$  and  $r$  by  $\omega = 2\pi \frac{d}{\lambda_0} \sin(\theta)$  and  $\phi = \pi \frac{d^2}{\lambda_0 r} \cos^2(\theta)$ . Then in [52], the exact propagation model

$$[\mathbf{a}(\theta, r)]_{m=1,\dots,M} = e^{i \frac{2\pi r}{\lambda_0} \left( \sqrt{1 + \frac{2(m-1)d \sin(\theta)}{r} + \frac{(m-1)^2 d^2}{r^2}} - 1 \right)},$$

has been used, that has revealed interesting features and interpretations not shown in [53]. Very recently, the uniform circular array (UCA) has been investigated in [54] in which the exact propagation model is now:

$$[\mathbf{a}(\theta, \phi, r)]_{m=1, \dots, M} = e^{i \frac{2\pi r}{\lambda_0} \left( 1 - \sqrt{1 - 2 \frac{r_0}{r} \cos\left(\theta - \frac{2\pi(m-1)}{M}\right) \sin(\phi) + \frac{r_0^2}{r^2}} \right)},$$

where  $r_0$ ,  $\theta$  and  $\phi$  denote the radius of the UCA, the azimuth and the elevation of the source. Note that in contrast to the closed-form expressions given in [53] and [52], the ones given in [54] relate the near and far-field CRB on the azimuth and elevation by very simple expressions.

### 3.16.3.2.3 Non Gaussian case

The stochastic CRB for the DOA appears to be prohibitive to compute for non-Gaussian sources. To cope with this difficulty, the deterministic model for the sources has been proposed for its simplicity. But in contrast to the stochastic ML estimator, the corresponding deterministic (or conditional) ML method does not asymptotically achieve this deterministic CRB, because the deterministic likelihood function does not meet the required regularity conditions (see Section 3.16.4.2). Consequently, this deterministic CRB is only a nonattainable lower bound on the covariance of any unbiased DOA estimator for arbitrary non-Gaussian distributions of the sources. So, it is useful to have explicit expressions of the stochastic CRB under non-Gaussian distributions.

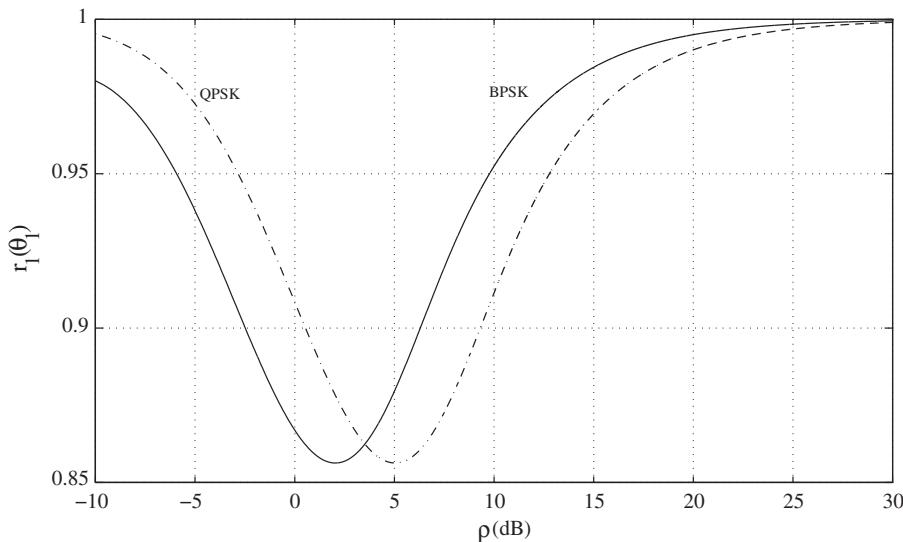
To the best of our knowledge, such stochastic CRBs have only been given in the case of binary phase-shift keying (BPSK), quaternary phase-shift keying (QPSK) signal waveforms [55] and then, to arbitrary  $L$ -ary square QAM constellation [56], and for a single source only. In these works, it is assumed Nyquist shaping and ideal sample timing apply so that the intersymbol interference at each symbol spaced sampling instance can be ignored. In the absence of frequency offset but with possible phase offset, the signals at the output of the matched filter can be represented as  $s_1(t) = \sigma_1^2 e^{i\phi_1} \epsilon_1(t)$ , where  $\{\epsilon_1(t)\}_{t_1, \dots, t_N}$  are independent identically distributed random symbols taking values  $\pm 1$  for BPSK symbols and  $\{\pm(2k-1)a \pm i(2l-1)a\}_{l,k=1, \dots, 2^q-1}$  with  $L = 2^{2q}$  for  $L$ -ary square QAM symbols, where  $2a$  is the intersymbol distance in the I/Q plane, which is adjusted such that  $E|\epsilon_1(t)|^2 = 1$ . For these discrete sources, the unknown parameter of this stochastic model is

$$\boldsymbol{\alpha} = [\theta_1, \phi_1, \sigma_1^2, \sigma_n^2]^T$$

and it has been proved in [55, 56] that the parameters  $(\theta_1, \phi_1)$  and  $(\sigma_1^2, \sigma_n^2)$  are decoupled in the associated FIM. This allows one to derive closed-form expressions of the so called non-data-aided (NDA) CRBs on the parameter  $\theta_1$  alone. In particular, it has been proved [55] that for a BPSK and QPSK source, that is respectively rectilinear and second-order circular, we have

$$\frac{\text{CRB}_{\text{BPSK}}(\theta_1)}{\text{CRB}_{\text{NCG}}(\theta_1)} = \frac{1}{(1 - g(\rho)) \left( 1 + \frac{1}{2\rho} \right)} \quad \text{and} \quad \frac{\text{CRB}_{\text{QPSK}}(\theta_1)}{\text{CRB}_{\text{CG}}(\theta_1)} = \frac{1}{(1 - g(\frac{\rho}{2})) \left( 1 + \frac{1}{\rho} \right)}, \quad (16.37)$$

where  $\text{CRB}_{\text{NCG}}(\theta_1)$  and  $\text{CRB}_{\text{CG}}(\theta_1)$  are given by (16.34) and with  $\rho \stackrel{\text{def}}{=} \frac{M\sigma_1^2}{\sigma_n^2}$  and  $g$  is the following decreasing function of  $\rho$ :  $g(\rho) \stackrel{\text{def}}{=} \frac{e^{-\rho}}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \frac{e^{-\frac{u^2}{2}}}{\cosh(u\sqrt{2\rho})} du$ . Equation (16.37) is illustrated in Figure 16.1 for a ULA of  $M$  sensors spaced a half-wavelength apart. We see from this figure that the CRBs under the non-circular [resp. circular] complex Gaussian distribution are tight upper bounds on the

**FIGURE 16.1**

Ratios  $r_1(\theta_1) \stackrel{\text{def}}{=} \frac{\text{CRB}_{\text{BPSK}}(\theta_1)}{\text{CRB}_{\text{NCG}}(\theta_1)}$  and  $r_1(\theta_1) \stackrel{\text{def}}{=} \frac{\text{CRB}_{\text{QPSK}}(\theta_1)}{\text{CRB}_{\text{CG}}(\theta_1)}$  as a function of  $\rho \stackrel{\text{def}}{=} \frac{M\sigma_1^2}{\sigma_n^2}$ .

CRBs under the BPSK [resp. QPSK] distribution at very low and very high SNRs only. Finally, note that among the numerous results of [55, 56], these stochastic NDA CRBs have been compared with those obtained with different a priori knowledge. In particular, it has been proved that in the presence of any unknown phase offset (i.e., non-coherent estimation), the ultimate achievable performance on the NDA DOA estimates holds almost the same irrespectively of the modulation order  $L$ . However, the NDA CRBs obtained in the absence of phase offset (i.e., coherent estimation) vary, in the high SNR region, from one modulation order to another.

Finally note that the ML estimation of the DOAs of these discrete sources has been proposed [57], where the maximization of the ML criterion (which is rather involved) is iteratively carried out by the expectation maximization (EM) algorithm. Adapted to the distribution of these sources, this approach allows one to account for any arbitrary noise covariance  $\mathbf{R}_n$  as soon as  $\mathbf{n}(t)$  is Gaussian distributed.

### 3.16.3.3 Asymptotically minimum variance bounds (AMVB)

To assess the performance of an algorithm based on a specific statistic  $\mathbf{g}_N$  built on  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$ , it is interesting to compare the asymptotic covariance  $\mathbf{R}_\theta$  (16.21) or (16.23) to an attainable lower bound that depends on the statistic  $\mathbf{g}_N$  only. The asymptotically minimum variance bound (AMVB) is such a bound. Furthermore, we note that the CRB appears to be prohibitive to compute for non-Gaussian sources and noise, except in simple cases and consequently this AMVB can be used as an useful benchmark against which potential estimates  $\hat{\theta}_N$  are tested. To extend the derivations of Porat and Friedlander [58] concerning this AMVB to complex-valued measurements, two additional conditions to those introduced in Section 3.16.3.1.1 must be satisfied:

- iii. The involved function **alg** that defines the considered algorithm must be  $\mathbb{C}$ -differentiable, i.e., must satisfy (16.22). In practice, it is sufficient to add conjugate components to all complex-valued components of  $\mathbf{g}$ , as in example (16.41);
- iv. The covariance  $\mathbf{R}_g$  of the asymptotic distribution of  $\mathbf{g}_N$  must be nonsingular. To satisfy this latter condition, the components of  $\mathbf{g}_N$  that are random variables, must be asymptotically linearly independent. Consequently the redundancies in  $\mathbf{g}_N$  must be withdrawn.

Under these four conditions, the covariance matrix  $\mathbf{R}_\theta$  of the asymptotic distribution of any estimator  $\hat{\boldsymbol{\theta}}_N$  built on the statistics  $\mathbf{g}_N$  is bounded below by  $(\mathbf{G}^H(\boldsymbol{\theta})\mathbf{R}_g^{-1}\mathbf{G}(\boldsymbol{\theta}))^{-1}$ :

$$\mathbf{R}_\theta = \mathbf{D}_{g,\theta}^{\text{alg}} \mathbf{R}_g \left( \mathbf{D}_{g,\theta}^{\text{alg}} \right)^H \geq \left( \mathbf{G}^H(\boldsymbol{\theta})\mathbf{R}_g^{-1}\mathbf{G}(\boldsymbol{\theta}) \right)^{-1}, \quad (16.38)$$

where  $\mathbf{G}(\boldsymbol{\theta})$  is the  $L \times P$  matrix  $\frac{d\mathbf{g}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}$ .

Furthermore, this lowest bound  $\text{AMVB}_{\mathbf{g}_N}(\boldsymbol{\theta}) \stackrel{\text{def}}{=} (\mathbf{G}^H(\boldsymbol{\theta})\mathbf{R}_g^{-1}\mathbf{G}(\boldsymbol{\theta}))^{-1}$  is asymptotically tight, i.e., there exists an algorithm **alg** whose covariance of its asymptotic distribution satisfies (16.38) with equality. The following nonlinear least square algorithm is an AMV second-order algorithm:

$$\hat{\boldsymbol{\theta}}_N = \arg \min_{\boldsymbol{\alpha} \in \Theta^P} [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})]^H \mathbf{R}_g^{-1}(\boldsymbol{\alpha}) [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})], \quad (16.39)$$

where we have emphasized here the dependence of  $\mathbf{R}_g$  on the unknown DOA  $\boldsymbol{\alpha}$ . In practice, it is difficult to optimize the nonlinear function (16.39), where it involves the computation of  $\mathbf{R}_g^{-1}(\boldsymbol{\alpha})$ . Porat and Friedlander proved for the real case in [59] that the lowest bound (16.38) is also obtained if an arbitrary weakly consistent estimate  $\mathbf{R}_{g,N}$  of  $\mathbf{R}_g(\boldsymbol{\alpha})$  is used in (16.39), giving the simplest algorithm:

$$\hat{\boldsymbol{\theta}}_N = \arg \min_{\boldsymbol{\alpha} \in \Theta^P} [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})]^H \mathbf{R}_{g,N} [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})]. \quad (16.40)$$

This property has been extended to the complex case in [60].

This AMVB and AMV algorithm have been applied to second-order algorithms that exploit both  $\mathbf{R}_{x,N}$  and  $\mathbf{C}_{x,N}$  in [24]. In this case, to fulfill the previously mentioned conditions (i)–(iv), the second-order statistics  $\mathbf{g}_N$  are given by

$$\mathbf{g}_N = \begin{bmatrix} \text{vec}(\mathbf{R}_{x,N}) \\ \text{v}(\mathbf{C}_{x,N}) \\ \text{v}(\mathbf{C}_{x,N}^*) \end{bmatrix}, \quad (16.41)$$

where  $\text{v}(\cdot)$  denotes the operator obtained from  $\text{vec}(\cdot)$  by eliminating all supradiagonal elements of a matrix. Finally, note that these AMVB and AMV DOA finding algorithm have been also derived for fourth-order statistics by splitting the measurements and statistics  $\mathbf{g}_N$  into its real and imaginary parts in [60].

### 3.16.3.4 Relations between AMVB and CRB: projector statistics

The AMVB based on any statistics is generally lower bounded by the CRB because this later bound concerns arbitrary functions of the measurements  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$ . But it has been proved in [44], that the AMVB associated with the different estimated projectors  $\mathbf{\Pi}_{x,N}$ ,  $(\mathbf{\Pi}_{x,N}, \mathbf{\Pi}'_{x,N})$  and  $\mathbf{\Pi}_{\tilde{x},N}$  introduced

in Section 3.16.3.1.2, which are functions of the second-order statistics of the measurements, attains the stochastic CRB in the case of circular or noncircular Gaussian signals. Consequently, there always exist asymptotically efficient subspace-based DOA algorithms in the Gaussian context.

To prove this asymptotic efficiency, i.e.,

$$\text{AMVB}_{\text{vec}(\boldsymbol{\Pi}_{x,N})}(\boldsymbol{\theta}) = \text{CRB}_{\text{CG}}(\boldsymbol{\theta}) \quad (16.42)$$

and

$$\text{AMVB}_{\text{vec}(\boldsymbol{\Pi}_{x,N}, \boldsymbol{\Pi}'_{x,N})}(\boldsymbol{\theta}) = \text{AMVB}_{\text{vec}(\boldsymbol{\Pi}_{\tilde{x},N})}(\boldsymbol{\theta}) = \text{CRB}_{\text{NCG}}(\boldsymbol{\theta}), \quad (16.43)$$

the condition (iv) of Section 3.16.3.3 that is not satisfied [61] for these statistics ought to be extended and consequently the results (16.38) and (16.39) must be modified as well, because here  $\mathbf{R}_g$  is singular.

In this singular case, it has been proved [61] that if the condition (iv) in the necessary conditions (i)–(iv) is replaced by the new condition  $\text{Span}(\mathbf{G}(\boldsymbol{\theta})) \subset \text{Span}(\mathbf{R}_g(\boldsymbol{\theta}))$ , (16.38) and (16.39) becomes respectively

$$\mathbf{R}_{\boldsymbol{\theta}} = \mathbf{D}_{g,\boldsymbol{\theta}}^{\text{alg}} \mathbf{R}_g \left( \mathbf{D}_{g,\boldsymbol{\theta}}^{\text{alg}} \right)^H \geq \left( \mathbf{G}^H(\boldsymbol{\theta}) \mathbf{R}_g^\# \mathbf{G}(\boldsymbol{\theta}) \right)^{-1} \quad (16.44)$$

and

$$\hat{\boldsymbol{\theta}}_N = \arg \min_{\boldsymbol{\alpha} \in \Theta^P} [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})]^H \mathbf{R}_g^\# (\boldsymbol{\alpha}) [\mathbf{g}_N - \mathbf{g}(\boldsymbol{\alpha})]. \quad (16.45)$$

And it is proved that the three statistics  $\text{vec}(\boldsymbol{\Pi}_{x,N})$ ,  $\text{vec}(\boldsymbol{\Pi}_{x,N}, \boldsymbol{\Pi}'_{x,N})$ , and  $\text{vec}(\boldsymbol{\Pi}_{\tilde{x},N})$  satisfy the conditions (i)–(iii) and (v) and thus satisfy results (16.44) and (16.45).

Finally, note that this efficiency property of the orthogonal projectors extends to the model of spatially correlated noise, for which  $\mathbf{R}_n = \sigma_n^2 \boldsymbol{\Sigma}_n$  where  $\boldsymbol{\Sigma}_n$  is a known positive definite matrix. In this case, for example, the orthogonal projector  $\boldsymbol{\Pi}_{x_w,N}$  defined after whitening

$$\begin{aligned} \{\mathbf{x}(t)\}_{t_1, \dots, t_N} &\longmapsto \{\mathbf{x}_w(t)\}_{t_1, \dots, t_N} \stackrel{\text{def}}{=} \{\boldsymbol{\Sigma}_n^{-1/2} \mathbf{x}(t)\}_{t_1, \dots, t_N} \longmapsto \mathbf{R}_{x_w, N} \\ &= \frac{1}{N} \sum_{n=1}^N \mathbf{x}_w(t_n) \mathbf{x}_w^H(t_n) \longmapsto \boldsymbol{\Pi}_{x_w, N} \end{aligned}$$

satisfies

$$\text{AMVB}_{\text{vec}(\boldsymbol{\Pi}_{x_w,N})}(\boldsymbol{\theta}) = \text{CRB}_{\text{CG}}^w(\boldsymbol{\theta}) = \frac{\sigma_n^2}{2N} \left\{ \text{Re} \left[ (\mathbf{D}^H \boldsymbol{\Pi}_{x_w} \mathbf{D}) \odot \left( \mathbf{R}_s \mathbf{A}^H \mathbf{R}_x^{-1} \mathbf{A} \mathbf{R}_s \right)^T \right] \right\}^{-1},$$

where  $\boldsymbol{\Pi}_{x_w} \stackrel{\text{def}}{=} \boldsymbol{\Sigma}_n^{-1} - \boldsymbol{\Sigma}_n^{-1} \mathbf{A} (\mathbf{A}^H \boldsymbol{\Sigma}_n^{-1} \mathbf{A})^{-1} \boldsymbol{\Sigma}_n^{-H} \mathbf{A}^H$  is insensitive to the choice of the square root  $\boldsymbol{\Sigma}_n^{1/2}$  of  $\boldsymbol{\Sigma}_n$ , and is no longer a projection matrix.

### 3.16.4 Asymptotic distribution of estimated DOA

We are now specifying in this section the asymptotic statistical performances of the main DOA algorithms that may be classified into three main categories, namely beamforming-based, maximum likelihood and moments-based algorithms.

### 3.16.4.1 Beamforming-based algorithms

Among the so-called beamforming-based algorithms, also referred to as low-resolution, compared to the parametric algorithms, the conventional (Bartlett) beamforming and Capon beamforming are the most referenced representatives of this family. These algorithms do not make any assumption on the covariance structure of the data, but the functional form of the steering vector  $\mathbf{a}(\theta)$  is assumed perfectly known. These estimators  $\hat{\theta}_N$  are given by the  $P$  highest (supposed isolated) maximizer and minimizer in  $\alpha$  of the respective following criteria:

$$\mathbf{a}^H(\alpha)\widehat{\mathbf{R}}_x\mathbf{a}(\alpha) \quad \text{and} \quad \mathbf{a}^H(\alpha)\widehat{\mathbf{R}}_x^{-1}\mathbf{a}(\alpha), \quad (16.46)$$

where  $\widehat{\mathbf{R}}_x$  is the unbiased sample estimate  $\mathbf{R}_{x,N}$  and  $\widehat{\mathbf{R}}_x^{-1}$  is either the biased estimate  $\mathbf{R}_{x,N}^{-1}$  or the unbiased estimate  $[(N-M)/N]\mathbf{R}_{x,N}^{-1}$  (that both give the same estimate  $\hat{\theta}_N$ ). Note that these algorithms extend to  $d$  parameters per source, where  $\alpha$  is replaced by  $\alpha = (\alpha_1, \dots, \alpha_d)$  in (16.46).

For arbitrary noise field (i.e., arbitrary noise covariance  $\mathbf{R}_n$ ) and/or an arbitrary number  $P$  of sources, the estimate  $\hat{\theta}_N$  given by these two algorithms are non-consistent, i.e.,

$$\lim_{N \rightarrow \infty} \hat{\theta}_N \neq \theta$$

and asymptotically biased. The asymptotic bias  $\text{AsBias}(\theta)$  can be straightforwardly derived by a second-order expansion of the criterion  $\mathbf{a}^H(\alpha)\mathbf{R}_x^\epsilon\mathbf{a}(\alpha)$  around each true values  $(\theta_p)_{p=1,\dots,P}$  (with  $\epsilon = +1$  [resp.,  $\epsilon = -1$ ] for the conventional [resp. Capon] algorithm), but noting that  $\lim_{N \rightarrow \infty} E(\hat{\theta}_{p,N})$  is a maximizer or minimizer  $\bar{\theta}_p$  of  $\mathbf{a}^H(\alpha)\mathbf{R}_x\mathbf{a}(\alpha)$  or  $\mathbf{a}^H(\alpha)\mathbf{R}_x^{-1}\mathbf{a}(\alpha)$ , respectively. The following value is obtained [62]

$$\text{AsBias}(\theta_p) \stackrel{\text{def}}{=} \lim_{N \rightarrow \infty} E(\hat{\theta}_{p,N}) - \theta_p = -\frac{\text{Re}[\mathbf{a}'^H(\theta_p)\mathbf{R}_x^\epsilon\mathbf{a}(\theta_p)]}{\mathbf{a}'^H(\theta_p)\mathbf{R}_x^\epsilon\mathbf{a}'(\theta_p) + \text{Re}[\mathbf{a}^H(\theta_p)\mathbf{R}_x^\epsilon\mathbf{a}''(\theta_p)]}, \quad (16.47)$$

$$\text{with } \mathbf{a}'(\theta_p) \stackrel{\text{def}}{=} \frac{d\mathbf{a}^H(\theta_p)}{d\theta_p} \text{ and } \mathbf{a}''(\theta_p) \stackrel{\text{def}}{=} \frac{d^2\mathbf{a}^H(\theta_p)}{d\theta_p^2}.$$

Following the methodology of Section 3.16.3.1.2, the additional bias for finite value of  $N$ , that is of order  $1/N$  can be derived, which gives

$$E(\hat{\theta}_{p,N}) - \theta_p = \text{AsBias}(\theta_p) + \frac{b_p}{N} + o\left(\frac{1}{N}\right),$$

see, e.g., the involved expression of  $b_p$  for the Capon algorithm [62, rel. (35)].

In the same way, the covariance  $E[(\hat{\theta}_N - E(\hat{\theta}_N))(\hat{\theta}_N - E(\hat{\theta}_N))^T]$  which is of order  $1/N$  can be derived. It is obtained with  $\bar{\theta} \stackrel{\text{def}}{=} [\bar{\theta}_1, \dots, \bar{\theta}_P]^T$

$$E[(\hat{\theta}_N - E(\hat{\theta}_N))(\hat{\theta}_N - E(\hat{\theta}_N))^T] = E[(\hat{\theta}_N - \bar{\theta})(\hat{\theta}_N - \bar{\theta})^T] + o\left(\frac{1}{N}\right) = \frac{\mathbf{R}_\theta}{N} + o\left(\frac{1}{N}\right),$$

see, e.g., the involved expression [50, rel. (24)] of  $\mathbf{R}_\theta$  associated with a source for several parameters per source. The relative values of the asymptotic bias, additional bias and standard deviation depend

on the SNR,  $M$  and  $N$ , but in practice the standard deviation is typically dominant over the asymptotic bias and additional bias (see examples given in [62]).

Finally, note that in the particular case of a single source, uniform white noise ( $\mathbf{R}_n = \sigma_n^2 \mathbf{I}$ ) and an arbitrary number  $d$  of parameters of the source (here  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_d)^T$ ), it has been proved [63], that  $\hat{\boldsymbol{\theta}}_N$  given by these two beamforming-based algorithms is asymptotically unbiased ( $\text{AsBias}(\theta_p)$  given by (16.47) is zero), if and only if  $\|\mathbf{a}(\boldsymbol{\theta})\|$  is constant. Furthermore, based on the general expressions (16.48) of the FIM<sup>6</sup>

$$\mathbf{FIM}(\boldsymbol{\theta}) = \frac{2N\sigma_s^4}{\sigma_n^2(\sigma_n^2 + \|\mathbf{a}(\boldsymbol{\theta})\|^2\sigma_n^2)} \operatorname{Re} \left[ \|\mathbf{a}(\boldsymbol{\theta})\|^2 \mathbf{D}(\boldsymbol{\theta})^H \mathbf{D}(\boldsymbol{\theta}) - \mathbf{D}(\boldsymbol{\theta})^H \mathbf{a}(\boldsymbol{\theta}) \mathbf{a}^H(\boldsymbol{\theta}) \mathbf{D}(\boldsymbol{\theta}) \right], \quad (16.48)$$

where  $\mathbf{D}(\boldsymbol{\theta})$  is defined here by  $[\partial \mathbf{a}(\boldsymbol{\theta}) / \partial \theta_1, \dots, \partial \mathbf{a}(\boldsymbol{\theta}) / \partial \theta_d]$ , for  $d$  parameters associated with a single source, and expression [50, rel. (24)] of  $\mathbf{R}_{\theta}$  specialized to  $\mathbf{R}_n = \sigma_n^2 \mathbf{I}$ , it has been proved that  $\frac{1}{N} \mathbf{R}_{\theta} = \mathbf{FIM}^{-1}(\boldsymbol{\theta})$ , i.e., the conventional and Capon algorithms are asymptotically efficient, if and only if  $\|\mathbf{a}(\boldsymbol{\theta})\|$  is constant.

### 3.16.4.2 Maximum likelihood algorithms

#### 3.16.4.2.1 Stochastic and deterministic ML algorithms

As discussed in Section 3.16.2.2, the two main models for the sensor array problem in Gaussian noise, corresponding to stochastic and deterministic modeling of the source signals lead to two different Gaussian distributions of the measurements  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$ , and consequently to two different log-likelihoods  $l(\boldsymbol{\alpha}) = \ln p(\mathbf{x}; \boldsymbol{\alpha})$ , where the unknown parameter  $\boldsymbol{\alpha}$  is respectively given by (16.4) and (16.35).

With some algebraic effort, the stochastic ML criterion  $l(\boldsymbol{\alpha})$  can be concentrated w.r.t.  $\mathbf{R}_s$  and  $\sigma_n^2$  (see, e.g., [64, 65]), thus reducing the dimension of the required numerical maximization to the required  $P$  DOAs ( $\theta_1, \dots, \theta_P$ ) and giving the following optimization problem:

$$\hat{\boldsymbol{\theta}}_N^{\text{SML}} = \arg \min_{\boldsymbol{\theta} \in \Theta^P} J_{\text{SML}}[\boldsymbol{\theta}, \mathbf{R}_{x,N}], \quad (16.49)$$

with

$$J_{\text{SML}}[\boldsymbol{\theta}, \mathbf{R}_{x,N}] = \ln[\det(\mathbf{A}(\boldsymbol{\theta}) \mathbf{R}_{s,N}(\boldsymbol{\theta}) \mathbf{A}^H(\boldsymbol{\theta}) + \sigma_{n,N}^2(\boldsymbol{\theta}) \mathbf{I})], \quad (16.50)$$

where

$$\mathbf{R}_{s,N}(\boldsymbol{\theta}) = \mathbf{A}^{\#}(\boldsymbol{\theta}) [\mathbf{R}_{x,N} - \sigma_{n,N}^2(\boldsymbol{\theta}) \mathbf{I}] \mathbf{A}^{\#H}(\boldsymbol{\theta}) \quad \text{and} \quad \sigma_{n,N}^2(\boldsymbol{\theta}) = \frac{1}{M-P} \operatorname{Tr}[\Pi_{\mathbf{A}}^{\perp}(\boldsymbol{\theta}) \mathbf{R}_{x,N}], \quad (16.51)$$

where  $\Pi_{\mathbf{A}}^{\perp}(\boldsymbol{\theta}) = \mathbf{I} - \mathbf{A}(\boldsymbol{\theta}) \mathbf{A}^{\#}(\boldsymbol{\theta})$  is the orthogonal projector onto the null space of  $\mathbf{A}^H$ . Despite its reduction of the parameter space,  $J_{\text{SML}}[\boldsymbol{\theta}, \mathbf{R}_{x,N}]$  is a complicated nonlinear expression in  $\boldsymbol{\theta}$ , that cannot be analytically minimized. Consequently, numerical optimization procedures are required.

Remark that in this modeling, the obvious a priori information that  $\mathbf{R}_s$  is positive semi-definite has not been taken into account. This knowledge, and more generally, the prior that  $\mathbf{R}_s$  is positive semi-definite of rank  $r$  smaller or equal than  $P$  can be included in the modeling by the parametrization

<sup>6</sup>For one parameter ( $d = 1$ ) or  $\|\mathbf{a}(\boldsymbol{\theta})\|$  constant, (16.48) can be simplified by withdrawing the real operator [2, rel. (49)].

$\mathbf{R}_s = \mathbf{L}\mathbf{L}^H$ , where  $\mathbf{L}$  is a  $P \times r$  lower triangular matrix. But this modification will have no effect for “large enough  $N$ ” since  $\widehat{\mathbf{R}}_s$  given by (16.51) is a weakly consistent estimate of  $\mathbf{R}_s$  [12]. And since this new parametrization leads to significantly more involved optimization, the unrestricted parametrization of  $\mathbf{R}_s$  used in (16.50) appears to be preferable.

Due to the quadratic dependence of the deterministic ML criterion  $l(\boldsymbol{\alpha})$  in the parameters  $\{\mathbf{s}(t)\}_{t_1, \dots, t_N}$ , its concentration w.r.t.  $\{\mathbf{s}(t)\}_{t_1, \dots, t_N}$  and  $\sigma_N^2$  is much more simpler than for the stochastic ML criterion. It gives the following new ML estimator:

$$\widehat{\boldsymbol{\theta}}_N^{\text{DML}} = \arg \min_{\boldsymbol{\theta} \in \Theta^P} J_{\text{DML}}[\boldsymbol{\theta}, \mathbf{R}_{x,N}], \quad (16.52)$$

with

$$J_{\text{DML}}[\boldsymbol{\theta}, \mathbf{R}_{x,N}] = \text{Tr}[\Pi_A^\perp(\boldsymbol{\theta}) \mathbf{R}_{x,N}]. \quad (16.53)$$

Comparing (16.53) and (16.50), we see that the dependence in  $\boldsymbol{\theta}$  of the DML criterion is simpler than for the SML criterion. But both criteria require nonlinear  $P$ th-dimensional minimizations with a large number of local minima that give two different estimates  $\boldsymbol{\theta}$ , except for a single source for which the minimization of (16.53) and (16.50) reduce to the maximization of the common criteria

$$\frac{\mathbf{a}^H(\boldsymbol{\theta}) \mathbf{R}_{x,N} \mathbf{a}(\boldsymbol{\theta})}{\|\mathbf{a}(\boldsymbol{\theta})\|^2}.$$

This implies that when the norm of the steering vector  $\mathbf{a}(\boldsymbol{\theta})$  is constant (which is generally assumed), the conventional and Capon beamforming, SML and DML algorithms coincide and thus conventional and Capon beamforming and DML algorithms inherit the asymptotical efficiency of the SML algorithm. Note that this property extends to several parameters per source.

### 3.16.4.2.2 Asymptotic properties of ML algorithms

We consider in this Subsection, the asymptotic properties of DML or SML algorithms used under the respectively, deterministic and circular Gaussian stochastic modeling of the sources. In the field of asymptotic performance characterization of DML or SML algorithms, asymptotic generally refers to either the number  $N$  of snapshots or the SNR value.

First, consider the asymptotic properties w.r.t.  $N$ , that are the most known. Under regularity conditions that are satisfied by the SML algorithm, the general properties of ML estimation states that  $\widehat{\boldsymbol{\theta}}_N^{\text{SML}}$  is consistent and asymptotically efficient and Gaussian distributed, more precisely

$$\sqrt{N} (\widehat{\boldsymbol{\theta}}_N^{\text{SML}} - \boldsymbol{\theta}) \xrightarrow{\mathcal{L}} \mathcal{N}_R(\mathbf{0}; \mathbf{R}_\theta^{\text{SML}}) \quad \text{with} \quad \mathbf{R}_\theta^{\text{SML}} = N \text{CRB}_{\text{CG}}(\boldsymbol{\theta}), \quad (16.54)$$

where  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  is given by (16.33). This property of the SML algorithm extends to nonuniform white and unknown parameterized noise field in [45,46], respectively, and to general noncircular Gaussian stochastic modeling of the sources with the associated  $\text{CRB}_{\text{NCG}}(\boldsymbol{\theta})$  [42,48]. Note that to circumvent the difficulty to extract the “ $\boldsymbol{\theta}$  corner” from the inverse of  $\mathbf{FIM}(\boldsymbol{\alpha})$ , a matrix closed-form expression of  $\text{CRB}_{\text{CG}}(\boldsymbol{\theta})$  has been first obtained in an indirect manner by an asymptotic analysis of the SML estimator [10,11]. Then, only 10 years later, this CRB has been obtained directly from the extended Slepian-Bangs formula [41,45].

As for the DML algorithm, since the signal waveforms themselves are regarded as unknown parameters, it follows that the number of unknown parameters  $\alpha$  (16.35) in the modeling, grows without limit with increasing  $N$ , the general asymptotic properties of the ML no longer apply. More precisely, the DML estimate of  $\theta$  is weakly consistent, whereas the DML estimate of  $\{\mathbf{s}(t_n)\}_{n=1,\dots,N}$  is inconsistent. The asymptotic distribution of  $\hat{\theta}_N^{\text{DML}}$  has been derived in [4,66]

$$\sqrt{N} \left( \hat{\theta}_N^{\text{DML}} - \theta \right) \xrightarrow{\mathcal{L}} \mathcal{N}_R \left( \mathbf{0}; \mathbf{R}_\theta^{\text{DML}} \right) \quad (16.55)$$

with

$$\mathbf{R}_\theta^{\text{DML}} = N \text{CRB}_{\text{Det}}(\theta) + 2N^2 \text{CRB}_{\text{Det}}(\theta) \text{Re} \left[ (\mathbf{D}^H \boldsymbol{\Pi}_x \mathbf{D}) \odot (\mathbf{A}^H \mathbf{A})^{-T} \right] \text{CRB}_{\text{Det}}(\theta), \quad (16.56)$$

where  $\text{CRB}_{\text{Det}}(\theta)$  is given by (16.36). Note that the inequality  $\frac{1}{N} \mathbf{R}_\theta^{\text{DML}} \leq \text{CRB}_{\text{Det}}(\theta)$  in (16.56) does not follow from the Cramer-Rao inequality theory directly, because the Cramer-Rao inequality requires that the number of unknown parameters be finite. As the number of real-valued parameters in  $\alpha$  (16.35) is  $P + 2NP + 1$ , it increases with  $N$  and the Cramer-Rao inequality does not apply here. Note that the DML estimates of  $\{\mathbf{s}(t_n)\}_{n=1,\dots,N}$  are indeed asymptotically unbiased, despite being non-consistent.

Furthermore, it has been proved in [4], that if the DML algorithm is used under the circular Gaussian stochastic modeling of the sources, the asymptotic distribution (16.54) of  $\hat{\theta}_N^{\text{DML}}$  is preserved. But under this assumption on the sources, the DML algorithm is suboptimal, and thus  $\frac{1}{N} \mathbf{R}_\theta^{\text{DML}} \geq \text{CRB}_{\text{CG}}(\theta)$ . Finally comparing directly the expressions (16.33) and (16.36) of the Cramer-Rao bound by applying the matrix inversion lemma, it is straightforward to prove that  $\text{CRB}_{\text{CG}}(\theta) \geq \text{CRB}_{\text{Det}}(\theta)$ . This allows one to relate  $\mathbf{R}_\theta^{\text{DML}}$ ,  $\mathbf{R}_\theta^{\text{SML}}$ ,  $\text{CRB}_{\text{CG}}(\theta)$ , and  $\text{CRB}_{\text{Det}}(\theta)$  by the following relation:

$$\frac{1}{N} \mathbf{R}_\theta^{\text{DML}} \geq \frac{1}{N} \mathbf{R}_\theta^{\text{SML}} = \text{CRB}_{\text{CG}}(\theta) \geq \text{CRB}_{\text{Det}}(\theta). \quad (16.57)$$

In particular, for a single source with  $q$  parameters, we have

$$\text{CRB}_{\text{CG}}(\theta) = \left( 1 + \frac{\sigma_n^2}{\|\mathbf{a}(\theta)\|^2 \sigma_s^2} \right) \text{CRB}_{\text{Det}}(\theta), \quad (16.58)$$

with  $\text{CRB}_{\text{CG}}(\theta) = \mathbf{FIM}^{-1}(\theta)$ , where  $\mathbf{FIM}(\theta)$  is given by (16.48).

Finally, note an asymptotic robustness property [10,11] of the SML and DML algorithms that states that the asymptotic distribution of  $\hat{\theta}_N^{\text{SML}}$  and  $\hat{\theta}_N^{\text{DML}}$  is preserved whatever the modeling of the source: circular Gaussian distributed with  $E[\mathbf{s}(t)\mathbf{s}^H(t)] = \mathbf{R}_s$  or modeled by arbitrary second-order ergodic signals with  $\mathbf{R}_s = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \mathbf{s}(t_n)\mathbf{s}^H(t_n)$ . We will present a more general asymptotic robustness property that applies to a large category of second-order algorithms in Section 3.16.4.3. The fact that the SML algorithm always outperforms (for  $P > 1$ ) the DML algorithm, provides strong justifications for the appropriateness of the stochastic modeling of sources for the DOA estimation problem.

Consider now, the asymptotic properties of the SML and DML algorithms w.r.t. SNR, used under their respective source model assumptions. It has been proved in [28], that under the circular Gaussian assumption of the sources, the SML estimates  $\hat{\theta}_N^{\text{SML}}$  is asymptotically (w.r.t. SNR) non-Gaussian

distributed and non-efficient, i.e.,  $\tilde{\boldsymbol{\theta}}_{\sigma_n} \stackrel{\text{def}}{=} \frac{1}{\sigma_n}(\hat{\boldsymbol{\theta}}_N^{\text{SML}} - \boldsymbol{\theta})$  converges in distribution to a non-Gaussian distribution, when  $\sigma_n$  tends to zero, with  $N$  fixed, with  $\lim_{\sigma_n \rightarrow 0} E[\tilde{\boldsymbol{\theta}}_{\sigma_n} \tilde{\boldsymbol{\theta}}_{\sigma_n}^T] \geq \lim_{\sigma_n \rightarrow 0} \frac{1}{\sigma_n^2} \text{CRB}_{\text{CG}}(\boldsymbol{\theta})$ . In practice,  $\hat{\boldsymbol{\theta}}_N^{\text{SML}}$  is non-Gaussian distributed and non-efficient at high SNR, only for a very small number  $N$  of snapshots.<sup>7</sup> For example, for a single source, using (16.37), it is proved in [28] that

$$\lim_{\sigma_n \rightarrow 0} E[\tilde{\boldsymbol{\theta}}_{\sigma_n} \tilde{\boldsymbol{\theta}}_{\sigma_n}^T] = \frac{N}{N-1} \lim_{\sigma_n \rightarrow 0} \frac{1}{\sigma_n^2} \text{CRB}_{\text{CG}}(\boldsymbol{\theta}) = \frac{N}{N-1} \left( \frac{1}{Nh_1 \sigma_1^2} \right),$$

(see (16.34) for the second equality), where  $h_1$  is defined just after (16.34). These properties contrast with the DML algorithm used under the deterministic modeling of the sources, which is proved [67] to be asymptotically (w.r.t. SNR) Gaussian distributed and efficient, i.e.,  $\frac{1}{\sigma_n}(\hat{\boldsymbol{\theta}}_N^{\text{DML}} - \boldsymbol{\theta}) \xrightarrow{\mathcal{L}} \mathcal{N}_R(\mathbf{0}; \frac{1}{2N}\{\text{Re}[(\mathbf{D}^H \boldsymbol{\Pi}_x \mathbf{D}) \odot \mathbf{R}_s]\}^{-1})$  when  $\sigma_n$  tends to zero, with  $N$  arbitrary fixed. These results are consistent with those of [11]. In practice for very high SNR and “not too small”  $N$ , (16.57) becomes

$$\frac{1}{N} \mathbf{R}_{\boldsymbol{\theta}}^{\text{DML}} \approx \frac{1}{N} \mathbf{R}_{\boldsymbol{\theta}}^{\text{SML}} = \text{CRB}_{\text{CG}}(\boldsymbol{\theta}) \approx \text{CRB}_{\text{Det}}(\boldsymbol{\theta}). \quad (16.59)$$

Furthermore, it has been proved in [11], that (16.59) is also valid for  $M \gg 1$ . The asymptotic distribution of the DOA estimate w.r.t.  $M$  (for finite data) of the SML and DML algorithms has been studied in [68]. The strong consistency has been proved for both ML algorithms. Furthermore, unlike the previously studied large sample case, the asymptotic covariance matrices of the DOA estimates coincide with the deterministic CRB (16.36) for the SML and DML algorithms. The asymptotic distribution of the DOA estimates given by subspace-based algorithms has been studied in [29], when  $M, N \rightarrow \infty$ , whereas  $M/N$  converges to a strictly positive constant. In this asymptotic regime, it is proved, in particular, that these traditional DOA estimates are not consistent. The threshold and the so-called subspace swap of the SML and MUSIC algorithms have been studied w.r.t.  $N, M$  and SNR (see, e.g., [69]). Furthermore, a new consistent subspace-based estimate has been proposed, which outperforms the standard subspace-based methods for values of  $M$  and  $N$  of the same order of magnitude [29].

### 3.16.4.2.3 Large sample ML approximations

Since the SML and DML algorithms are often deemed exceedingly complex, suboptimal algorithms are of interest. Many such algorithms have been proposed in the literature and surprisingly, some of them are asymptotically as accurate as the ML algorithms, but with a reduced computational cost. These algorithms have been derived, either by approximations of the ML criteria by neglecting terms that do not affect the asymptotic properties of the estimates, or by using a purely geometrical point of view. We present this latter approach that allows one to unify a large number of algorithms [12]. These algorithms rely on the geometrical properties of the spectral decomposition of the covariance matrix  $\mathbf{R}_x$ :

$$\mathbf{R}_x = \mathbf{E}_s \boldsymbol{\Lambda}_s \mathbf{E}_s^H + \sigma_n^2 \mathbf{E}_n \mathbf{E}_n^H$$

<sup>7</sup>In practice the approximate covariances deduced from the asymptotic analysis w.r.t. the number of snapshots are also valid for high SNR with fixed “not too small number” of snapshots for the second-order DOA algorithms. But note that there is no theoretical result on the asymptotic distribution of the sample projector w.r.t. the SNR.

with  $\mathbf{E}_s = [\mathbf{e}_1, \dots, \mathbf{e}_r]$ ,  $\Lambda_s = \text{Diag}(\lambda_1, \dots, \lambda_r)$ , and  $\mathbf{E}_n = [\mathbf{e}_{r+1}, \dots, \mathbf{e}_M]$  where  $r$  is the rank of  $\mathbf{R}_s$ , associated with the consistent estimates

$$\mathbf{R}_{x,N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \mathbf{x}(t_n) \mathbf{x}^H(t_n) = \mathbf{E}_{s,N} \Lambda_{s,N} \mathbf{E}_{s,N}^H + \sigma_{n,N}^2 \mathbf{E}_{n,N} \mathbf{E}_{n,N}^H. \quad (16.60)$$

These algorithms can be classified as signal subspace-based and noise subspace-based fitting algorithms. The former algorithms based on  $\text{Span}(\mathbf{E}_s) \subseteq \text{Span}(\mathbf{A}(\boldsymbol{\theta}))$  are given by the following optimization:

$$\hat{\boldsymbol{\theta}}_N^{\text{SSF}} = \arg \min_{\boldsymbol{\theta} \in \Theta^P} \text{Tr} \left[ \boldsymbol{\Pi}_A^{\perp}(\boldsymbol{\theta}) \mathbf{E}_{s,N} \mathbf{W} \mathbf{E}_{s,N}^H \right], \quad (16.61)$$

where  $\mathbf{W}$  is a weighting  $r \times r$  positive definite matrix to be specified. And the latter algorithms based on  $\mathbf{E}_n^H \mathbf{A}(\boldsymbol{\theta}) = \mathbf{0}$ , that is valid only if the source covariance matrix is nonsingular ( $r = P$ ), are given by

$$\hat{\boldsymbol{\theta}}_N^{\text{NSF}} = \arg \min_{\boldsymbol{\theta} \in \Theta^P} \text{Tr} [\mathbf{U} \mathbf{A}^H(\boldsymbol{\theta}) \mathbf{E}_{n,N} \mathbf{E}_{n,N}^H \mathbf{A}(\boldsymbol{\theta})], \quad (16.62)$$

where  $\mathbf{U}$  is a weighting  $P \times P$  positive definite matrix to be specified.

Introduced from a purely geometrical point of view, these two classes of algorithms present unexpected relations with the previously described ML algorithms. First, for arbitrary positive definite weighting matrices  $\mathbf{W}$  and  $\mathbf{U}$ , the estimates  $\hat{\boldsymbol{\theta}}_N^{\text{SSF}}$  and  $\hat{\boldsymbol{\theta}}_N^{\text{NSF}}$  given respectively by (16.61) and (16.62), are weakly consistent. Second, for the weighting matrices that give the lowest covariance matrix of the asymptotic distribution of  $\hat{\boldsymbol{\theta}}_N^{\text{SSF}}$  and  $\hat{\boldsymbol{\theta}}_N^{\text{NSF}}$ , that are respectively given [12] by

$$\mathbf{W}_{\text{opt}} = (\Lambda_s - \sigma_n^2 \mathbf{I})^2 \Lambda_s^{-1} \quad \text{and} \quad \mathbf{U}_{\text{opt}} = \mathbf{A}^{\#}(\boldsymbol{\theta}_0) \mathbf{E}_s \mathbf{W}_{\text{opt}} \mathbf{E}_s^H \mathbf{A}^{\#H}(\boldsymbol{\theta}_0),$$

where  $\boldsymbol{\theta}_0$  denotes here the true value of the DOAs, the associated estimates  $\hat{\boldsymbol{\theta}}_N^{\text{SSF}}$  and  $\hat{\boldsymbol{\theta}}_N^{\text{NSF}}$  are asymptotically equivalent to  $\hat{\boldsymbol{\theta}}_N^{\text{SML}}$  (i.e.,  $\sqrt{N}(\hat{\boldsymbol{\theta}}_N^{\text{SSF}} - \hat{\boldsymbol{\theta}}_N^{\text{SML}}) \rightarrow \mathbf{0}$  and  $\sqrt{N}(\hat{\boldsymbol{\theta}}_N^{\text{NSF}} - \hat{\boldsymbol{\theta}}_N^{\text{SML}}) \rightarrow \mathbf{0}$  in probability as  $N \rightarrow \infty$ ) and thus have the same asymptotic distribution that the SML algorithm. Furthermore and fortunately, this property extends for any weakly consistent estimates  $\mathbf{W}_N$  and  $\mathbf{U}_N$  of respectively  $\mathbf{W}_{\text{opt}}$  and  $\mathbf{U}_{\text{opt}}$ , e.g., derived from the spectral decomposition of the sample covariance matrix  $\mathbf{R}_{x,N}$  (16.60) with  $\sigma_{n,N}^2$  is the average of  $M - r$  smallest eigenvalues of  $\mathbf{R}_{x,N}$  and with  $\boldsymbol{\theta}_0$  is replaced by a weakly consistent estimates of  $\boldsymbol{\theta}$ . This implies a two steps procedure to run the optimal noise subspace-based fitting algorithm. Due to this drawback, the signal subspace-based fitting algorithm with the weighting  $\mathbf{W}_N = (\Lambda_{s,N} - \sigma_{n,N}^2 \mathbf{I})^2 \Lambda_{s,N}^{-1}$ , denoted weighted subspace fitting (WSF) algorithm, is preferred to the noise subspace-based fitting algorithms.

Finally, note that this algorithm is based on eigenvalues and eigenvectors of the sample covariance matrix  $\mathbf{R}_{x,N}$ . This contrasts with the subspace-based algorithms whose asymptotic statistical properties will be studied in Section 3.16.4.4 that are based on the noise or signal orthogonal projector  $\boldsymbol{\Pi}_{x,N}$  associated with  $\mathbf{R}_{x,N}$  only. Note that general properties of subspace-based estimators focused on asymptotic invariance of these estimators have been given in [70].

### 3.16.4.3 Second-order algorithms

Most of the narrowband DOA algorithms presented in the literature are second-order algorithms, i.e., are based on the sample covariance  $\mathbf{R}_{x,N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \mathbf{x}(t_n) \mathbf{x}^H(t_n)$  or more generally on  $\mathbf{R}_{\tilde{x},N} \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^N \tilde{\mathbf{x}}(t_n) \tilde{\mathbf{x}}^H(t_n)$ . To prove common properties of this class of algorithm, it is useful to use the functional analysis presented in Section 3.16.3.1.1

$$\{\mathbf{x}(t)\}_{t_1, \dots, t_N} \longmapsto \mathbf{R}_{x,N} \xrightarrow{\text{alg}} \hat{\boldsymbol{\theta}}_N, \quad (16.63)$$

in which any second-order algorithm is a mapping **alg** that generally satisfies

$$\text{alg}(\mathbf{A}(\boldsymbol{\theta}) \mathbf{R}_s \mathbf{A}^H(\boldsymbol{\theta}) + \sigma_n^2 \mathbf{I}) = \boldsymbol{\theta} \quad \text{for any } \boldsymbol{\theta} \in \Theta^P, \quad (16.64)$$

but not necessarily for all  $P \times P$  Hermitian positive semi-definite matrix  $\mathbf{R}_s$ . Depending on the a priori knowledge about  $\mathbf{R}_s$ , that is required by the second-order algorithms **alg**, different constraints are satisfied by the  $\mathbb{C}$ -differential matrix  $\mathbf{D}_{R_x, \theta}^{\text{alg}}$  of the algorithm at the point  $\mathbf{R}_x$  (16.22). In particular, it has been proved the following main two constraints [20]:

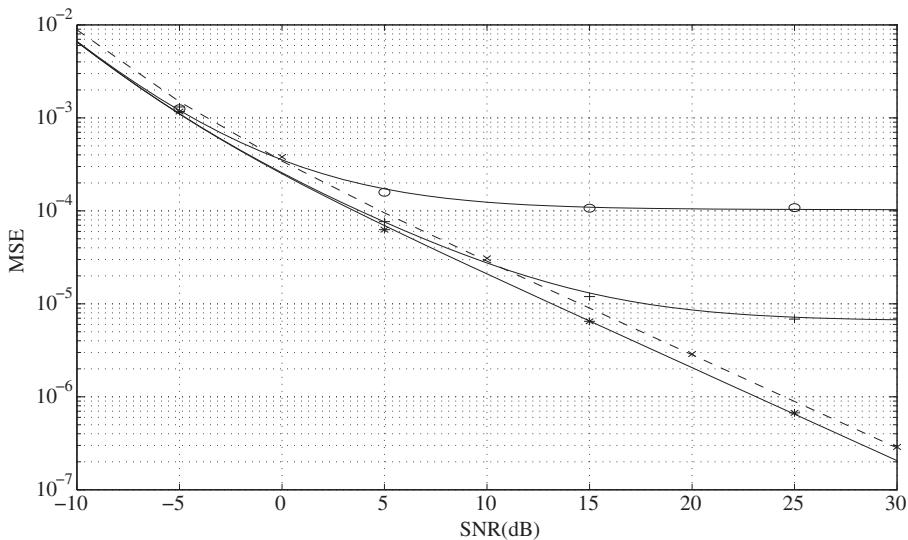
$$\mathbf{D}_{R_x, \theta}^{\text{alg}} (\mathbf{A}(\boldsymbol{\theta}) \otimes \mathbf{A}(\boldsymbol{\theta})) = \mathbf{0} \quad \text{for } \mathbf{R}_s \text{ unstructured} \quad (16.65)$$

$$\mathbf{D}_{R_x, \theta}^{\text{alg}} (\mathbf{a}(\theta_p) \otimes \mathbf{a}(\theta_p)) = \mathbf{0}, \quad p = 1, \dots, P \quad \text{for } \mathbf{R}_s \text{ structured diagonal.} \quad (16.66)$$

Using these constraints, the general expression  $\mathbf{R}_{R_x}$  of the covariance of the asymptotic distribution of the sample covariance  $\mathbf{R}_{x,N}$  [31] obtained under mild conditions for non independent measurements with arbitrary distributed sources and noise of finite fourth-order moments, and the general relation (16.23), that links  $\mathbf{R}_{R_x}$  and  $\mathbf{D}_{R_x, \theta}^{\text{alg}}$  to the covariance  $\mathbf{R}_\theta$  of the asymptotic distribution of  $\hat{\boldsymbol{\theta}}_N$ , allows one to prove the following two results, that extend a robustness property presented in [71]:

- For any second-order algorithms based on  $\mathbf{R}_{x,N}$ , that do not require the sources spatially uncorrelated and when the noise signals  $\{\mathbf{n}(t)\}_{t_1, \dots, t_N}$  are temporally uncorrelated,  $\mathbf{R}_\theta$  is invariant to the distribution, the second-order noncircularity and the temporal distribution of the sources, but depends on the distribution of the noise through its second-order and fourth-order moments. In particular for circular Gaussian noise, the asymptotic distribution of  $\hat{\boldsymbol{\theta}}_N$  are those of the standard complex circular Gaussian case.
- For any second-order algorithms based on  $\mathbf{R}_{x,N}$  that require the sources spatially uncorrelated and/or when the noise signals  $\{\mathbf{n}(t)\}_{t_1, \dots, t_N}$  are temporally correlated,  $\mathbf{R}_\theta$  is sensitive to the distribution, the second-order noncircularity and the temporal distribution of the sources.

Note that the majority of the second-order algorithms (e.g., the beamforming, ML, MUSIC, Min Norm, ESPRIT algorithms) does not require spatially uncorrelated sources. In contrast, second-order techniques based on state-space realizations (e.g., the Toeplitz approximation method (TAM), see [8]) and Toeplitzization or augmentation with ULA or uniform rectangular arrays, require this uncorrelation, and thus the asymptotic distribution of  $\hat{\boldsymbol{\theta}}_N$  will be generally (except for a single source, for which the constraint (16.66) reduces to (16.65)) sensitive to the distribution, the second-order noncircularity or the temporal distribution of the sources, even when the noise is temporally uncorrelated.

**FIGURE 16.2**

Theoretical and estimated MSE (with 500 Monte Carlo runs) of  $\theta_1$  versus the SNR, for respectively white ( $\circ$ ), colored (+) and harmonic (\*) signals for  $N = 100$  after Toeplitzization (—) and without Toeplitzization (- - -).

To illustrate this sensitivity to the source distribution when the noise is temporally uncorrelated, we consider in Figure 16.2, the case of two equipowered and spatially uncorrelated sources impinging on a ULA of 10 sensors,  $\theta_1 = 20^\circ$  and  $\theta_2 = 30^\circ$ , where the DOAs are estimated by the standard MUSIC algorithm after Toeplitzization. The sources are either white Gaussian, ARMA Gaussian (generated by a (10, 10) Butterworth filter driven by a white circular Gaussian noise, where the bandwidth is fixed to 0.5) or harmonic. The centered frequencies of the ARMA and the frequencies of the harmonics are  $-0.25$  and  $0.25$ . Figure 16.2 shows that the Toeplitzization improves the performance for very weak SNR only, whereas is very sensitive to the distribution of the sources for high SNR.

Usually, performance analyses are evaluated as a function of the number  $N$  of observed snapshots without taking the sampling rate into account. In fact, depending on the value of this sampling rate, the collected samples  $\mathbf{x}(t_n)$  are more or less temporally correlated and performance is affected. Thus, the interesting question arises as to how the asymptotic covariance of the DOA estimators (denoted here  $\hat{\theta}_T$ ) varies with this sampling rate  $\frac{1}{T_s}$  for a fixed observation interval  $T = NT_s$ . This question has been investigated in [20], in which the continuous-time noise envelope  $\mathbf{n}(t)$  is spatially white and temporally white in the bandwidth  $[-\frac{B}{2}, +\frac{B}{2}]$ . It has been proved:

- If the signals  $\mathbf{x}(t)$  are oversampled ( $\frac{1}{T_s} > B$ )

$$\mathbb{E}[(\hat{\theta}_T - \theta)(\hat{\theta}_T - \theta)^T] \approx \frac{1}{BT} \mathbf{R}_\theta > \frac{1}{N} \mathbf{R}_\theta \quad \text{for } N \gg 1,$$

irrespective of the sample rate  $1/T_s$ .

- If the signals  $\mathbf{x}(t)$  are subsampled ( $\frac{1}{T_s} < B$ )

$$\mathbb{E}[(\hat{\boldsymbol{\theta}}_T - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_T - \boldsymbol{\theta})^T] \approx \frac{T_s}{T} \mathbf{R}_\theta = \frac{1}{N} \mathbf{R}_\theta > \frac{1}{BT} \mathbf{R}_\theta \quad \text{for } N \gg 1 \text{ and } BT_s \gg 1.$$

Consequently the array must be temporally oversampled, and the parameter of interest that characterizes performance ought not to be the number  $N$  of snapshots, but rather the observation interval  $T$ .

### 3.16.4.4 Subspace-based algorithms

We concentrate now on the family of second-order algorithms based on the orthogonal noise<sup>8</sup> projector  $\Pi_{x,N}$  (16.10). These algorithms estimate  $\boldsymbol{\theta}$ , either by extrema-searching approaches (MUSIC, Min-Norm, etc.), by polynomial rooting approaches (Pisarenko, root MUSIC, and root Min-Norm for ULA), or by matrix shifting approaches (ESPRIT, TAM, Matrix pencil method). The most celebrated of these algorithms is the MUSIC algorithm, where  $\boldsymbol{\theta}$  is estimated as the  $P$  deepest minima in a  $d$ -dimensional (for  $d$  parameters per source) of the following localization function  $J_{\text{MUSIC}}[\boldsymbol{\theta}, \Pi_{x,N}]$ :

$$J_{\text{MUSIC}}[\boldsymbol{\theta}, \Pi_{x,N}] = \mathbf{a}^H(\boldsymbol{\theta}) \Pi_{x,N} \mathbf{a}(\boldsymbol{\theta}), \quad (16.67)$$

of the so-called spatial null spectrum (or equivalently as the  $P$  highest peaks (maxima) of its inverse). This algorithm has given a plethora of variants. For example, in the particular case of the ULA, this standard MUSIC algorithm have been favorably replaced by the root MUSIC algorithm. Using the general methodology presented in Section 3.16.3.1.2, the asymptotic distribution of  $\hat{\boldsymbol{\theta}}_N$  given by any subspace-based algorithms **alg** is simply derived from the expression of the  $\mathbb{C}$ -differential matrix  $\mathbf{D}_{\Pi_x, \theta}^{\text{alg}}$  of the mapping  $\Pi_{x,N} \xrightarrow{\text{alg}} \hat{\boldsymbol{\theta}}_N$  evaluated at  $\Pi_x(\boldsymbol{\theta})$ . For example, for the standard MUSIC algorithm,  $\mathbf{D}_{\Pi_x, \theta}^{\text{MUSIC}}$  is straightforwardly obtained from the first-order expansion of  $\left( \frac{\partial J_{\text{MUSIC}}(\boldsymbol{\theta}, \Pi_{x,N})}{\partial \boldsymbol{\theta}} \right)_{\boldsymbol{\theta}=\boldsymbol{\theta}_p+\delta\boldsymbol{\theta}_{p,N}} = 0$  that gives for one parameter per source

$$\mathbf{D}_{\Pi_x, \theta}^{\text{MUSIC}} = \begin{bmatrix} \mathbf{d}_1^T \\ \vdots \\ \mathbf{d}_P^T \end{bmatrix} \quad \text{with} \quad \mathbf{d}_p^T = -\frac{1}{h_p} \left( (\mathbf{a}'^T(\boldsymbol{\theta}_p) \otimes \mathbf{a}^H(\boldsymbol{\theta}_p)) + (\mathbf{a}^T(\boldsymbol{\theta}_p) \otimes \mathbf{a}'^H(\boldsymbol{\theta}_p)) \right), \quad p = 1, \dots, P, \quad (16.68)$$

with  $\mathbf{a}'(\boldsymbol{\theta}_p) \stackrel{\text{def}}{=} \frac{d\mathbf{a}(\boldsymbol{\theta}_p)}{d\theta_p}$  and  $h_p \stackrel{\text{def}}{=} 2\mathbf{a}'^H(\boldsymbol{\theta}_p) \Pi_x \mathbf{a}'(\boldsymbol{\theta}_p)$ . Using (16.68) with (16.16) and (16.23) allow one to directly prove that the sequences  $\sqrt{N}(\hat{\boldsymbol{\theta}}_N - \boldsymbol{\theta})$  converges in distribution to the zero-mean Gaussian distribution of covariance matrix given elementwise by  $(\mathbf{R}_\theta^{\text{MUSIC}})_{k,l} = \frac{2}{h_k h_l} \text{Re}((\mathbf{a}^H(\boldsymbol{\theta}_l) \mathbf{U} \mathbf{a}(\boldsymbol{\theta}_k))(\mathbf{a}'^H(\boldsymbol{\theta}_k) \mathbf{U} \mathbf{a}'(\boldsymbol{\theta}_l)))$  and compactly by

$$\mathbf{R}_\theta^{\text{MUSIC}} = 2(\mathbf{H} \odot \mathbf{I})^{-1} \text{Re} \left( \mathbf{H} \odot (\mathbf{A}^H \mathbf{U} \mathbf{A})^T \right) (\mathbf{H} \odot \mathbf{I})^{-1}, \quad (16.69)$$

---

<sup>8</sup>Note that since  $\Pi_x + \Pi_x^\perp = \mathbf{I}$  and  $\Pi_{x,N} + \Pi_{x,N}^\perp = \mathbf{I}$ , all algorithm based on the orthogonal signal projector comes down to an algorithm based on the orthogonal noise projector.

where  $(\mathbf{H})_{p,p} \stackrel{\text{def}}{=} h_p$  and  $\mathbf{U}$  has been defined in Section 3.16.3.1.2. Note that these expressions have been derived in [3] by much more involved derivations based on the asymptotic distribution of the eigenvectors of the sample covariance matrix  $\mathbf{R}_{x,N}$ . Finally, note that if the sample orthogonal noise projector  $\mathbf{\Pi}_{x,N}$  is replaced by an adaptive estimator  $\mathbf{\Pi}_{x,\gamma}$  of  $\mathbf{\Pi}_x$ , where  $\gamma$  is the step-size of an arbitrary constant step-size recursive stochastic algorithm (see e.g., [72, 73]), it has been proved in [72] that  $\sqrt{\gamma}(\hat{\theta}_\gamma - \theta)$  converges in distribution to the zero-mean Gaussian distribution of covariance matrix given also by  $\mathbf{R}_\theta^{\text{MUSIC}}$ , where  $\hat{\theta}_\gamma$  is an adaptive estimate of  $\theta$  given by the MUSIC algorithm based on the specific adaptive estimate  $\mathbf{\Pi}_{x,\gamma}$  of  $\mathbf{\Pi}_x$  studied in [72]. Using a similar approach [26], it has been proved that the Root MUSIC algorithm associated with the ULA, presents the same asymptotic distribution, but slightly outperforms the standard MUSIC algorithm outside the asymptotic regime. This analysis has been extended to MUSIC-like algorithms applied to the orthogonal noise projectors  $\mathbf{\Pi}'_{x,N}$  [resp.  $\mathbf{\Pi}_{\tilde{x},N}$ ] associated with the complementary sample covariance  $\mathbf{C}_{x,N}$  [the augmented sample covariance  $\mathbf{R}_{\tilde{x},N}$ ] matrices for the DOA estimation of arbitrary noncircular [resp. rectilinear] sources [27]. Finally, note that with our general methodology, all the expressions of the covariance  $\mathbf{R}_\theta^{\text{MUSIC}}$  can be straightforwardly extended for several parameter per source.

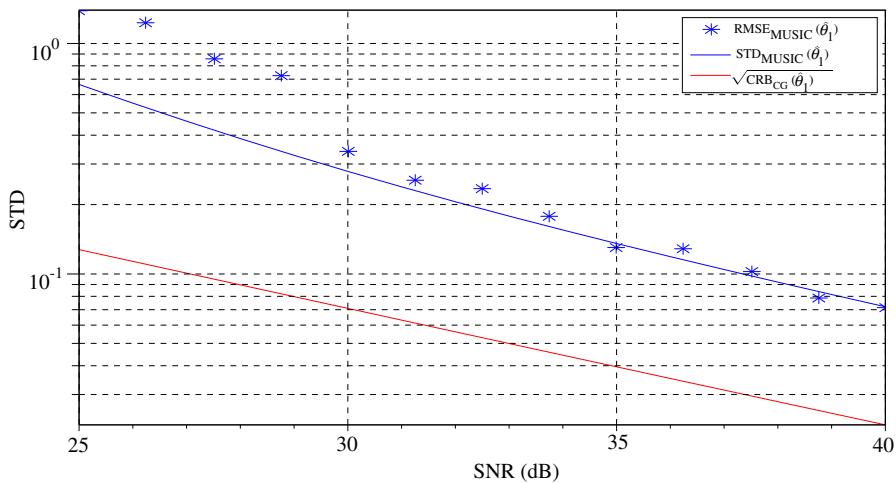
The expression of the covariance (16.69) of the asymptotic distribution of  $\hat{\theta}_N$  given the standard MUSIC algorithm has been analyzed in detail (see, e.g., [2, 3]). In particular it has been proved that the MUSIC algorithm is asymptotically efficient for a single source, an arbitrary number of parameters per source and  $\|\mathbf{a}(\theta_1)\|$  depending on  $\theta_1$ , e.g., for one parameter per source

$$\frac{1}{N} \mathbf{R}_{\theta_1}^{\text{MUSIC}} = \text{CRB}_{\text{CG}}(\theta_1) = \frac{1}{N} \left( \frac{1}{h_1} \left[ \frac{\sigma_n^2}{\sigma_1^2} + \frac{1}{\|\mathbf{a}(\theta_1)\|^2} \frac{\sigma_n^4}{\sigma_1^4} \right] \right).$$

For several sources, the MUSIC algorithm is in general asymptotically inefficient, in particular for correlated sources for which the efficiency degrades when the correlation between the sources increases. The degradation of performances are considerable for highly correlated sources for any value of the SNRs. In contrast, for uncorrelated sources, the MUSIC algorithm is asymptotically efficient when  $\sigma_n^2$  tends to zero, in the following sense  $\lim_{\sigma_n^2 \rightarrow 0} [\frac{1}{N} \mathbf{R}_\theta^{\text{MUSIC}}] [\text{CRB}_{\text{CG}}(\theta)]^{-1} = \mathbf{I}$ . So, in practice, for uncorrelated sources, the MUSIC algorithm is asymptotically efficient for high SNRs of all the sources.

It is of utmost importance to investigate in what region of  $N$  and SNR, the asymptotic theoretical results can predict actual performance. But unfortunately, only Monte Carlo simulations can specify this region. We illustrate in the following the SNR threshold region for the SML, DML, and MUSIC algorithm.

Consider two zero-mean circular Gaussian sources impinging on an ULA (16.2) with  $M = 6$  (for which the 3 dB bandwidth is about  $8^\circ$ ) and a spatially uniform white noise (16.3). The source  $s_1(t)$  consists of a strong direct path at  $\theta_1 = 0^\circ$  relative to array broadside and a weaker (multipath at  $\theta_2 = 4^\circ$  at  $-3$  dB w.r.t.  $s_1(t)$ ). The correlation between  $s_1(t)$  and  $s_2(t)$  is 0.99 giving thus the source covariance matrix  $\mathbf{R}_s = \begin{bmatrix} 1 & 0.7 \\ 0.7 & 0.5 \end{bmatrix}$ . Figure 16.3 shows the root mean square error (RMSE) of the estimated DOA  $\hat{\theta}_1$  by the MUSIC algorithm w.r.t. the SNR defined by  $\sigma_1^2/\sigma_n^2$ , compared with the theoretical standard deviation (TSD)  $\sqrt{\frac{1}{N} (\mathbf{R}_\theta^{\text{MUSIC}})_{1,1}}$  and the square root of the stochastic CRB  $\sqrt{\text{CRB}_{\text{CG}}(\theta_1)}$ . We see from this figure

**FIGURE 16.3**

RMSE of  $\hat{\theta}_1$  estimated by the MUSIC algorithm (averaged on 1000 runs) compared with the theoretical standard deviation and the square root of the stochastic CRB, as a function of the SNR for  $N = 1000$ .

that the MUSIC algorithm is not efficient at all for highly correlated sources. Furthermore, the domain of validity of the asymptotic regime is here very limited, i.e., for  $N = 1000$ ,  $\text{SNR} > 30 \text{ dB}$  is required.

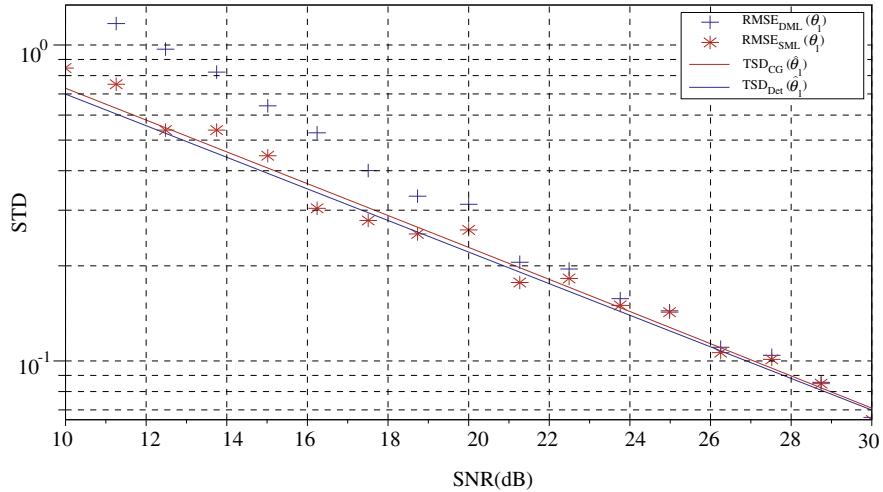
With the same parameters, Figure 16.4 shows the RMSE of the estimated DOA  $\hat{\theta}_1$  by the SML and DML algorithms which are compared with the TSD  $\sqrt{\frac{1}{N}(\mathbf{R}_{\theta}^{\text{SML}})_{1,1}}$  and  $\sqrt{\frac{1}{N}(\mathbf{R}_{\theta}^{\text{DML}})_{1,1}}$  and the square roots of the CRBs  $\sqrt{CRB_{CG}(\hat{\theta}_1)}$  and  $\sqrt{CRB_{DET}(\hat{\theta}_1)}$ . We see from this figure that the numerical values of the four expressions of (16.57) are very close and the performance of the two ML algorithms are very similar except for the SNR threshold region for which the SML algorithm is efficient for  $\text{SNR} > 0 \text{ dB}$  with  $N = 1000$ . Finally, comparing Figures 16.3 and 16.4, we see that both ML algorithms largely outperform the MUSIC algorithm for highly correlated sources.

### 3.16.4.5 Robustness of algorithms

We distinguish in this subsection, the robustness of the DOA estimation algorithms w.r.t. the narrowband assumption and to array modeling errors, because for the array modeling errors, the model (16.1) remains valid with a modified steering matrix, in contrast to the violation of narrowband assumption, for which (16.1) must be modified.

#### 3.16.4.5.1 Robustness w.r.t. the narrowband assumption

As the wideband assumption generally requires an increased computational complexity compared to the narrowband ones, it is of interest to examine if the narrowband methods can be used for a sufficiently wide bandwidth without sacrificing performance. Some responses to this question have been given in [74] for symmetric spectra w.r.t. the demodulation frequency and in [75] for non-symmetric spectra and/or offset of the centered value of the spectra w.r.t. the demodulation frequency  $f_0$ . In these assumptions,

**FIGURE 16.4**

RMSE of  $\hat{\theta}_1$  estimated by the SML and DML algorithms (averaged on 1000 runs) compared with the theoretical standard deviations and the square root of the stochastic and deterministic CRBs, as a function of the SNR for  $N = 1000$ .

the model (16.1) of the complex envelope of the measurements becomes

$$\mathbf{x}(t) = \sum_{p=1}^P \int_{-B/2}^{+B/2} \mathbf{a}(\theta_p, f_0 + f) e^{i2\pi f t} d\mu_p(f) + \mathbf{n}(t), \quad (16.70)$$

where  $\mathbf{a}(\theta_p, v) \stackrel{\text{def}}{=} [e^{i2\pi v \tau_{1,p}}, \dots, e^{i2\pi v \tau_{M,p}}]^T$  (with  $\mathbf{a}(\theta_p, f_0) = \mathbf{a}(\theta_p)$ ) and  $\mu_p(f)$  is the spectral measure of the  $p$ th source. Using the general methodology explained in Section 3.16.3.1, based on a first-order expansion of the DOA estimate  $\hat{\theta}_N = \text{alg}(\boldsymbol{\Pi}_{x,N})$  in the neighborhood of  $\boldsymbol{\Pi}_x$  (where  $\boldsymbol{\Pi}_{x,N}$  and  $\boldsymbol{\Pi}_x$  are the orthogonal projectors onto the noise subspace associated with the covariance of (16.70) and (16.1), respectively), general closed-form expressions of the asymptotic (w.r.t. the number of snapshots and source bandwidth) for arbitrary subspace-based algorithm have been derived in [75]. It is found that the behavior of these DOA estimators strongly depends on the symmetry of the source spectra w.r.t. their centered value and on the offset of this centered value w.r.t.  $f_0$ . It is showed that the narrowband SOS-based algorithms are much more sensitive to the frequency offset than to the bandwidth.

In particular for source spectra  $S_s(f)$  symmetric w.r.t. the demodulation frequency  $f_0$ , it is proved that the estimated DOAs given by any narrowband subspace-based algorithm are asymptotically unbiased w.r.t. the number of snapshots and signal bandwidth. More precisely

$$E(\hat{\theta}_N) - \boldsymbol{\theta} = \left( \frac{f_\sigma^2}{f_0^2} \right) \mathbf{b}^{\text{alg}} + O\left( \frac{f_\sigma^4}{f_0^4} \right) + O\left( \frac{1}{N} \right),$$

where  $f_\sigma \stackrel{\text{def}}{=} \left[ \int_{-B}^B S_s(f) f^2 df / \int_{-B}^B S_s(f) df \right]^{1/2}$  is the definition used for the bandwidth. Furthermore, for a single source,  $\mathbf{R}_x = \mathbf{R}_{s_1} \odot \mathbf{a}(\theta_1)\mathbf{a}^H(\theta_1) + \sigma_n^2 \mathbf{I}$ , where the nuisance parameters are now the terms of the Hermitian matrix  $\mathbf{R}_{s_1}$  and  $\sigma_n^2$ . This new parameterization allows to derive the circular Gaussian stochastic CRB issued from a non-zero bandwidth  $\text{CRB}_{\text{CG}}^{\text{NZB}}(\theta_1)$ . It is related to the standard  $\text{CRB}_{\text{CG}}(\theta_1)$  by the relation

$$\text{CRB}_{\text{CG}}^{\text{NZB}}(\theta_1) = \text{CRB}_{\text{CG}}(\theta_1) \left( 1 + c \left( \frac{f_\sigma^2}{f_0^2} \right) + O \left( \frac{f_\sigma^4}{f_0^4} \right) \right),$$

where the expression of  $c$  is given in [75].

#### 3.16.4.5.2 Robustness to array modeling errors

Imprecise knowledge of the gain and phase characteristics of the array sensors, and of the sensor locations and possible mutual coupling, can seriously degrade the theoretical performance of the DOA estimation algorithms. Experimental systems attempt to eliminate or minimize these errors by careful calibrations. But even when initial calibration is possible, system parameters may change over time and thus the array modeling errors cannot be completely eliminated. Consequently, it is useful to qualify the sensitivity of the DOA estimator algorithms to these modeling errors, i.e., to study the effect of difference between the true and assumed array manifold  $\{\mathbf{a}(\theta), \theta \in \Theta\}$  caused by modeling errors, on DOA estimator algorithms. This analysis has received relatively little attention in the literature.

In these studies, to simplify the analysis, the covariance matrix  $\mathbf{R}_x$  is assumed perfectly known, i.e., the effects of a finite number of samples is assumed negligible. Let  $\gamma$  gather the array parameters which are the subject of the sensitivity analysis. For example,  $\gamma$  may contain the sensors gain, phases or location, or other parameters such as the mutual coupling coefficients of the array sensors. A DOA estimation algorithm uses the steering matrix  $\mathbf{A}(\theta, \gamma_0) = [\mathbf{a}(\theta_1, \gamma_0), \dots, \mathbf{a}(\theta_P, \gamma_0)]$ , corresponding to a nominal value  $\gamma_0$  of the array parameters that differs from the true steering matrix  $\mathbf{A}(\theta, \gamma)$ , where  $\gamma$  is slightly different from  $\gamma_0$  (see particular parameterizations studied in [76, 77]). We refer to the difference between the true and assumed array parameters as a modeling error. The sensitivity study of a particular DOA estimation algorithm consists to provide a relation between  $\delta\theta = \theta_\gamma - \theta$  and the modeling error  $\delta\gamma = \gamma - \gamma_0$  in the mapping

$$\mathbf{R}_x(\gamma) = \mathbf{A}(\theta, \gamma) \mathbf{R}_s \mathbf{A}^H(\theta, \gamma) + \sigma_n^2 \mathbf{I} \xrightarrow{\text{alg}(\gamma_0)} \theta_\gamma, \quad (16.71)$$

where naturally  $\mathbf{R}_x(\gamma_0) \xrightarrow{\text{alg}(\gamma_0)} \theta$ , if  $\text{alg}(\gamma_0)$  denotes an arbitrary second-order algorithm based on the nominal array. Using a first order perturbation of (16.71) in the neighborhood of  $\gamma_0$ , through those of the orthogonal projector on the noise subspace  $\Pi_x(\gamma)$ , a relation  $\delta\theta = h(\delta\gamma) + o(\delta\gamma)$  where  $h$  is linear has been given for the MUSIC and DML algorithms in [77–79], respectively. These works model the errors  $\delta\gamma$  by zero-mean independent random variables ( $\delta\gamma = \sigma_\gamma \mathbf{u}$  where  $\mathbf{u}$  is a random vector whose elements are zero-mean unit variance random variables). They lead to estimates that are approximatively unbiased (i.e.,  $E(\theta_\gamma) - \theta = o(\sigma_\gamma)$ ) and where their approximative variances depend only on the second-order statistics of the modeling errors (more precisely  $\text{Var}(\theta_{p,\gamma}) = c_p \sigma_\gamma^2 + o(\sigma_\gamma^2)$ ,  $p = 1, \dots, P$ ). However, by confronting these theoretical results with numerical experiments, one notices that the MUSIC and DML algorithms are biased in the presence of multiple sources and these theoretic and experimental

variances do not agree with larger modeling errors. More precisely, these theoretical results are valid only up to the point where the probability of resolution is close to one (see [25]).

To take into account these larger modeling errors, a more accurate relation between  $\delta\theta$  and  $\delta\gamma$ , based on a second-order expansion of  $\Pi_x(\gamma)$  around  $\gamma_0$  (provided by a recursive  $n$ th order expansion of  $\delta\Pi_x$  w.r.t.  $\delta\mathbf{R}_x$  [6]) as been given in [25, 80] for analyzing the sensitivity of the MUSIC and DML algorithms to larger modeling errors. Modeling the errors  $\delta\gamma$  as previously, an approximation of the bias  $E(\theta_\gamma) - \theta$  that depends on the second-order statistics of the modeling errors, and of the variance that now depends on the fourth-order statistics of the modeling errors, are given. These refined closed-form expressions can predict the actual performance observed by numerical experiments for larger modeling errors, in particular in the threshold regions of the MUSIC and DML algorithms.

Note that the sensitivity of DOA estimators to modeling errors of the noise covariance matrix, that includes the presence of undetected weak signals, has also been studied in the literature (see, e.g., [81]). Finally, note that the combined effects of random array modeling errors and finite samples have been analyzed for the class of so-called signal subspace fitting (SSF) algorithms in [82]. In addition to deriving the first-order asymptotic expressions for the covariance of the estimation error, an additional weighting matrix has been introduced in (16.61) that has been optimized for any particular random array modeling errors.

### 3.16.4.6 High-order algorithms

When the sources are non Gaussian distributed, they convey valuable statistical information in their moments of order greater than two (this is in particular true when considering communications signals). In these circumstances, it makes sense to consider DOA estimation techniques using this higher order information. Of particular interest are the algorithms based on higher order cumulants of the measurements  $\{\mathbf{x}(t)\}_{t_1, \dots, t_N}$  due to their additivity property in the sums of independent components. Furthermore, these cumulants show the distinctive property of being in a certain sense, insensitive to additive Gaussian noise, making it possible to devise consistent DOA estimates without it being necessary to know, to model or to estimate the noise covariance  $\mathbf{R}_n$ . As generally, the distributions of the sources are even, their odd order moments are zero and thus to cope with these signals, only the even high-order cumulants of the measurements are used.

Computational considerations dictate using mainly fourth-order cumulants. To use these approaches, we consider the assumptions of Section 3.16.2.2, in which we add that the sources  $\{\mathbf{s}_p(t)\}_{p=1, \dots, P}$  have nonvanishing fourth-order cumulants. Furthermore, we assume that their moments are finite up to the eighth-order, to study the statistical performance of these algorithms.

Of course, there are many more quadruples than pairs of indices, and consequently a very large number of cumulants  $\text{Cum}(x_i(t), x_j^*(t), x_k^*(t), x_l(t))$ ,  $i, j, k, l = 1, \dots, M$  for circular sources (and more,  $\text{Cum}(x_i(t), x_j^*(t), x_k(t), x_l(t))$  and  $\text{Cum}(x_i(t), x_j(t), x_k(t), x_l(t))$ ,  $i, j, k, l = 1, \dots, M$  for noncircular sources) can be exploited despite their redundancies, to identify the DOA parameters with unknown noise covariance. For example, for circular signals, the maximum set of nonredundant cumulants is

$$\text{Cum}\left(x_i(t), x_j^*(t), x_k^*(t), x_l(t)\right) \quad \text{with } 1 \leq i \leq M, 1 \leq l \leq i, 1 \leq j \leq i \quad \text{and } 1 \leq k \leq j.$$

The asymptotically minimum variance (AMV) algorithm (see Section 3.16.3.3) based on a subset of fourth-order cumulants that can identify the DOA parameters, is the nonlinear least square algorithm

(16.40) in which  $\mathbf{g}_N$  gathers the involved cumulants. To implement this AMV algorithm, one has to decide which cumulants should be included in  $\mathbf{g}_N$ . The best estimate would be obtained when all nonredundant cumulants are selected. This, however, may require excessive computations if  $M$  is large. However it is sufficient to deal with a reduced set of cumulants, although there do not seem to be any simple guidelines in this matter [60]. In practice, a good tradeoff between computational complexity and accuracy is to devise suboptimal algorithms that require an overall computational effort similar to the second-order algorithms, while retaining a fourth-order cumulants subset, sufficient for DOA identification. Such algorithms have been proposed in the literature such as the diagonal slice (DS), the contracted quadricovariance (CQ) and the so called 4-MUSIC [60] algorithms. The first two algorithms are fourth-order subspace-based algorithms built on the following rank defective  $M \times M$  matrices:

$$\begin{aligned} (\mathbf{Q}_x^{\text{DS}})_{i,j} &= \text{Cum}\left(x_i(t), x_j^*(t), x_j^*(t), x_j(t)\right), \\ (\mathbf{Q}_x^{\text{CQ}})_{i,j} &= \sum_{m=1}^M \text{Cum}\left(x_i(t), x_j^*(t), x_m^*(t), x_m(t)\right). \end{aligned}$$

They require  $P < M$  sources and their statistical performance has been analyzed in [19] with the general framework explained in Section 3.16.3.1. In particular, it is has been proved that for a single source and a ULA in spatially uniform white noise, these two fourth-order algorithms have similar performance to the MUSIC algorithm, except for low SNR, for which the MUSIC algorithm outperforms both fourth-order algorithms. The 4-MUSIC algorithm is built from the rank defective  $M^2 \times M^2$  matrix

$$(\mathbf{Q}_x^{\text{4-MUSIC}})_{i+(j-1)M, k+(l-1)M} = \text{Cum}\left(x_i(t), x_j^*(t), x_k^*(t), x_l(t)\right).$$

It is proved in [60] that

$$\mathbf{Q}_x^{\text{4-MUSIC}} = [\mathbf{A}^*(\boldsymbol{\theta}) \otimes \mathbf{A}(\boldsymbol{\theta})] \mathbf{Q}_s [\mathbf{A}^*(\boldsymbol{\theta}) \otimes \mathbf{A}(\boldsymbol{\theta})]^H,$$

where  $(\mathbf{Q}_s)_{i+(j-1)P, k+(l-1)P} = \text{Cum}(s_i(t), s_j^*(t), s_k^*(t), s_l(t))$ ,  $i, j, k, l = 1, \dots, P$ .  $\mathbf{Q}_x^{\text{4-MUSIC}}$  is indefinite in general and its rank is  $\sum_{g=1}^G r_g^2$  where the  $P$  sources are divided in  $G$  groups, with  $r_g$  in the  $g$ th group. The sources in each group are assumed to be dependent, while sources belonging to different groups are assumed independent. Because the vectors  $\mathbf{a}^*(\boldsymbol{\theta}_p) \otimes \mathbf{a}(\boldsymbol{\theta}_p)$ ,  $p = 1, \dots, P$  are  $P$  columns of  $\mathbf{A}^*(\boldsymbol{\theta}) \otimes \mathbf{A}(\boldsymbol{\theta})$ , the 4-MUSIC algorithm is obtained by searching the  $P$  deepest minima of the following localization function  $J_{\text{4-MUSIC}}[\boldsymbol{\theta}, \mathbf{\Pi}_{x,N}]$ :

$$J_{\text{4-MUSIC}}[\boldsymbol{\theta}, \mathbf{\Pi}_{x,N}] = [\mathbf{a}^*(\boldsymbol{\theta}) \otimes \mathbf{a}(\boldsymbol{\theta})]^H \mathbf{\Pi}_{x,N} [\mathbf{a}^*(\boldsymbol{\theta}) \otimes \mathbf{a}(\boldsymbol{\theta})], \quad (16.72)$$

where  $\mathbf{\Pi}_{x,N}$  is now, the orthogonal projector onto the noise subspace of the sample estimate  $\mathbf{Q}_{x,N}^{\text{4-MUSIC}}$  of  $\mathbf{Q}_x^{\text{4-MUSIC}}$ . In practice the statistical dependence of the sources are unknown. Porat and Fiedlander [60] has proposed to retain only  $M^2 - P^2$ , rather  $M^2 - \sum_{g=1}^G r_g^2$  eigenvectors corresponding to the smallest singular values of  $\mathbf{Q}_{x,N}^{\text{4-MUSIC}}$ . We note that, to the best of our knowledge, no complete statistical performance analysis of this algorithm has yet appeared in the literature. Despite its higher variance (w.r.t. the MUSIC algorithm under the assumption of spatially uniform white noise), this fourth-order

algorithm presents some advantages, aside from its capacity to deal with unknown Gaussian noise fields. Using the concept of virtual array, it is proved in [83] that this algorithm can identify up to  $M^2 - M$  sources when the sensors are identical and up to  $M^2 - 1$  sources for different sensors. Furthermore, it is shown that its resolution for closely spaced sources and robustness to modeling errors is improved with respect to the MUSIC algorithm. To increase even more its number of sources to be processed, resolution and robustness to modeling errors, extensions of this 4-MUSIC algorithms, giving rise to the  $2q$ -MUSIC (with  $q > 2$ ) has been proposed [84].

### 3.16.5 Detection of number of sources

One of the more difficult and critical problems facing passive sensor arrays systems is the detection of the number  $P$  of sources impinging on the array. This is a key step in most of the parametric estimation techniques that were briefly described in Section 3.16.4. The eigendecomposition based techniques require in addition, information on the dimension  $r$  of the signal subspace. If the source covariance  $\mathbf{R}_s$  has full rank, i.e., there are no coherent sources present,  $P$  and  $r$  are identical. Moreover, the solution of the detection problem has, in many cases, value of its own, regardless of the DOA estimation problem.

A natural scheme for detecting the number  $P$  of sources is to formulate a likelihood ratio test based on the SML estimator (16.49). Such a test is often referred to as a generalized likelihood ratio test (GLRT). This test can be implemented by a sequential test procedure (see, e.g., [12, Sec. 4.7.1]). For each hypothesis, the likelihood ratio statistic is computed and compared to a threshold. The accepted hypothesis is the first one for which the threshold is crossed. The problem with this method is the subjective judgment required for deciding on the threshold levels or the associated probabilities of false alarm related by the asymptotic distribution of the normalized likelihood ratio.

Another important approach to the detection problem is the application of the information theoretic criteria for model selection. Unlike the conventional hypothesis testing based approaches, these criteria do not require any subjective threshold setting. Among them, the minimum description length (MDL) criterion introduced by Rissanen [85] is the most widely used because of its consistency. This technique has been used for detecting the signal subspace dimension  $r$  [13], and also for detecting the number of sources  $P$  [86]. We concentrate now on the detection of  $r$ .

#### 3.16.5.1 MDL criterion

The information theoretic criteria approach is a general method for detecting the order  $r$  of a statistical model. That is, given a parameterized probability density function  $p(\mathbf{x}; \boldsymbol{\alpha}^{(r)})$  for various order  $r$ , detect  $\hat{r}$  such that  $\hat{r} = \arg \min_r \{-\ln[p(\mathbf{x}; \hat{\boldsymbol{\alpha}}_{\text{ML}}^{(r)})] + g(r)\}$ , where  $\hat{\boldsymbol{\alpha}}_{\text{ML}}^{(r)}$  is the ML estimate of  $\boldsymbol{\alpha}^{(r)}$  and  $g(r)$  is a penalty function. For the MDL criterion which is based on a particular penalty function,  $\hat{r}$  is given for  $N$  independent identically distributed measurements  $\mathbf{x}(t_n)$ , by

$$\hat{r} = \arg \min_r \left[ -\ln \left( p \left( \mathbf{x}; \hat{\boldsymbol{\alpha}}_{\text{ML}}^{(r)} \right) \right) + \frac{1}{2} \text{card}(\boldsymbol{\alpha}^{(r)}) \ln(N) \right], \quad (16.73)$$

where  $\text{card}(\boldsymbol{\alpha}^{(r)})$  denotes the number of free real-valued parameters in  $\boldsymbol{\alpha}^{(r)}$ . Depending on the distribution of the measurements  $\mathbf{x}$  and its parametrization  $\boldsymbol{\alpha}$ , different implementations of the MDL criterion have been proposed.

The most often used assumption, is the zero-mean circular Gaussian distribution associated with the parametrization (16.1) in which all the elements of the steering matrix  $\mathbf{A}$  are assumed unknown with the only restriction that  $\mathbf{A}$  has full column rank with  $M > P$ . For this modeling, the measurements can be parameterized by the parameter

$$\boldsymbol{\alpha}^{(r)} = \left[ \mathbf{v}_1^T, \dots, \mathbf{v}_r^T, \lambda_1, \dots, \lambda_r, \sigma_n^2 \right]^T,$$

where  $\lambda_1 \geq \dots \geq \lambda_r > \sigma_n^2 = \dots, \sigma_n^2$  are the eigenvalues of  $\mathbf{R}_x$  and  $\mathbf{v}_1, \dots, \mathbf{v}_r$ , the eigenvectors associated with the largest  $r$  eigenvalues, and the general MDL criterion (16.73), which is referred to as the Gaussian MDL (GMDL), becomes [13]

$$\hat{r} = \underset{r}{\operatorname{Arg\,min}} \Lambda_r \quad \text{with} \quad \Lambda_r \stackrel{\text{def}}{=} N(M-r) \ln \left( \frac{\hat{a}_r}{\hat{g}_r} \right) + \frac{1}{2} r(2M-r) \ln N, \quad (16.74)$$

with  $\hat{a}_r \stackrel{\text{def}}{=} \frac{1}{M-r} \sum_{i=r+1}^M \hat{\lambda}_i$  and  $\hat{g}_r \stackrel{\text{def}}{=} \prod_{i=r+1}^M \hat{\lambda}_i^{1/(M-r)}$ , where  $\hat{\lambda}_1 > \hat{\lambda}_2 > \dots > \hat{\lambda}_M$  are the eigenvalues of the sample covariance matrix  $\frac{1}{N} \sum_{n=1}^N \mathbf{x}(t_n) \mathbf{x}^H(t_n)$ , denoted here by  $\widehat{\mathbf{R}}_x$ .

### 3.16.5.2 Performance analysis of MDL criterion

This GMDL criterion has been analyzed in [16], and it has been shown to be a consistent estimator of the rank  $r$ , i.e., the probability of error decreases to zero as the number  $N$  of measurements increases to infinity. Moreover, under mild regularity conditions, like finite second moments, it is a consistent estimator of the rank  $r$ , even if the measurements are non-Gaussian. This property contrasts with the Akaike information criterion (AIC) that yields an inconsistent estimate of that tends, asymptotically, to overestimate  $r$  [13].

The GMDL criterion has been further analyzed by considering the events  $\hat{r} < r$  and  $\hat{r} > r$ , called underestimation and overestimation, respectively. Since  $(\Lambda_r)_{r=0,\dots,M-1}$  are functions of the eigenvalues  $(\hat{\lambda}_i)_{i=1,\dots,M}$  of  $\widehat{\mathbf{R}}_x$ , the derivation of the probabilities  $P(\hat{r} > r)$  and  $P(\hat{r} < r)$  needs the joint exact or asymptotic distribution of  $(\hat{\lambda}_i)_{i=1,\dots,M}$ . This asymptotic distribution is available for circular complex Gaussian distribution [87] and more generally for arbitrary distributions with finite fourth-order moments [14], but unfortunately, the functional  $(\Lambda_r)_{r=0,\dots,M-1}$  (16.74) is too complicated to infer its asymptotic distribution. Therefore, for simplifying the derivation of these probabilities, it has been argued [36, 88, 89] by extended Monte Carlo experiments (essentially for  $r = 1$  and  $r = 2$ ) that

$$\begin{aligned} P(\hat{r} > r) &\approx P(\hat{r} = r + 1) \approx P(\Lambda_{r+1} < \Lambda_r) \quad \text{and} \\ P(\hat{r} < r) &\approx P(\hat{r} = r - 1) \approx P(\Lambda_{r-1} < \Lambda_r). \end{aligned} \quad (16.75)$$

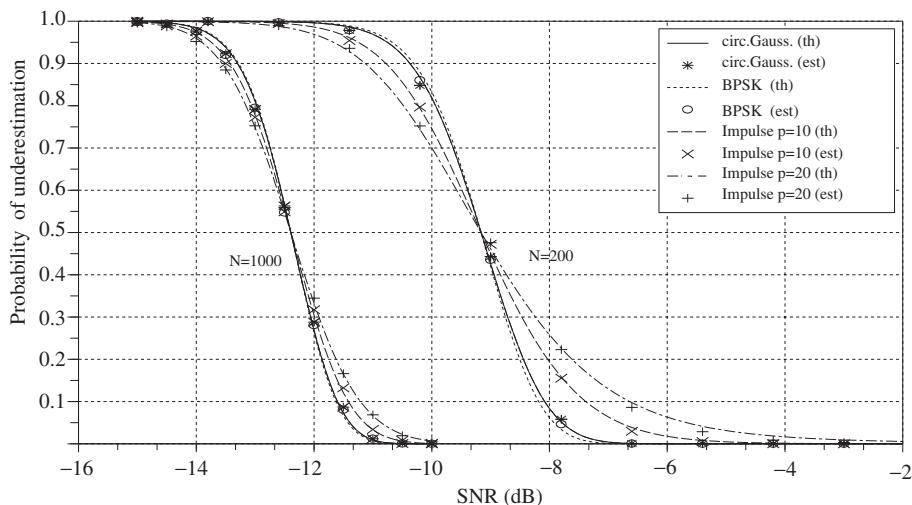
As the probability of overestimation is concerned, exact and approximate asymptotic upper bound of this probability have been derived in [36] showing that generally  $P(\hat{r} > r) \ll 1$ . Therefore, only the

probability of underestimation has been analyzed by many authors. In particular, using the refinement introduced by [90]

$$E(\hat{a}_r) = \frac{1}{M-r} \left( \text{Tr}(\mathbf{R}_x) - \sum_{i=1}^r E(\hat{\lambda}_i) \right) = \sigma_n^2 + \frac{1}{M-r} \sum_{i=1}^r (\lambda_i - E(\hat{\lambda}_i))$$

of the classical approximation  $E(\hat{a}_r) \approx \sigma_n^2$  and the asymptotic bias (16.18) and covariance (16.19), a closed-form expression of the probability of underestimation given by the GMDL criterion, used under arbitrary distributions with finite fourth-order moments, has been given in [14]. This expression has been analyzed for  $P = r = 1$  and  $P = r = 2$  for different distributions of the sources in [14]. Figure 16.5 illustrates the robustness of the MDL criterion to the distribution of the sources. We see from this figure that the probability of underestimation is sensitive to the distribution of the source, particularly for sources of large kurtosis and for weak values of the number  $N$  of snapshots.

The general MDL criterion has been studied in [15], where using the approximation (16.75), a general analytical expression of  $P(\hat{r} < r)$  has been given. This expression allows one to prove the consistency of the general MDL criterion when the number of snapshots tends to infinite and has been specialized to particular parameterized distributions. Among them, the Gaussian assumption associated with a parameterized steering matrix  $\mathbf{A}(\theta)$  has been studied and some numerical illustrations show that the use of this prior information about the array geometry enables an improvement in performance of about 2 dB. Finally, note that the MDL criterion generally fails when the sample size is smaller than the number of sensors. In this situation a sample eigenvalue based detector has been proposed in [91].



**FIGURE 16.5**

$P(\hat{r} = 0/r = 1)$  as a function of the SNR for four distributions of the source (the impulsive takes the values  $\{-1, 0, +1\}$  with  $P(s(t_n) = -1) = P(s(t_n) = +1) = \frac{1}{2\rho}$ ) and two values of the number  $N$  of snapshots, for an ULA with five sensors.

---

### 3.16.6 Resolution of two closely spaced sources

An important measure to quantify the statistical performance for the DOA estimation problem is the resolvability of closely spaced signals in terms of their parameters of interest. The principal question to characterize this resolvability is to find the minimum SNR (denoted threshold array SNR (ASNR)) required for a sensor array to correctly resolve two closely spaced signals for a given DOA distance  $\Delta\theta \stackrel{\text{def}}{=} |\theta_2 - \theta_1|$  (called angular resolution limit (ARL) or statistical resolution limit) between them. Generally in the literature there are three different ways to describe this resolution limit. The first one is based on the mean null spectrum concerning a specific algorithm. The second one is based on the estimation accuracy, more precisely on the Cramer-Rao Bound. The last one is based on the detection theory using the hypothesis test formulation.

#### 3.16.6.1 Angular resolution limit based on mean null spectra

Based on the array beam-pattern  $G(\theta_0, \theta) = |\mathbf{a}^H(\theta_0)\mathbf{a}(\theta)|$ , different resolution criteria have been defined from its main lobe w.r.t. a look direction  $\theta_0$ , as the celebrated Rayleigh resolutions such as the half power beamwidth or the null to null beamwidth that depends solely on the antenna geometry, and consequently have the serious shortcoming of being independent of the SNR.

For specific so-called high resolution algorithms, such as different MUSIC-like algorithms, based on the search for two local minima of sample null spectra  $J_{\text{Alg}}(\theta, \boldsymbol{\Pi}_{x,N})$ , two main criteria based on the mean null spectrum  $E[J_{\text{Alg}}(\theta, \boldsymbol{\Pi}_{x,N})]$  have been defined. These criteria are justified by the property that the standard deviation  $\sqrt{\text{Var}(J_{\text{Alg}}(\theta, \boldsymbol{\Pi}_{x,N}))}$  of the sample null spectrum associated with the conventional MUSIC and Min-Norm algorithms is small compared to its mean value  $E[J_{\text{Alg}}(\theta, \boldsymbol{\Pi}_{x,N})]$  in the vicinity of the true DOAs for  $N \gg M$  for arbitrary SNR [1].

For the first criterion, introduced by Cox [92], two sources are resolved if the midpoint mean null spectrum is greater than the mean null spectrum in the two true source DOAs:

$$E[J_{\text{Alg}}(\theta_m, \boldsymbol{\Pi}_{x,N})] \geq \frac{1}{2} (E[J_{\text{Alg}}(\theta_1, \boldsymbol{\Pi}_{x,N})] + E[J_{\text{Alg}}(\theta_2, \boldsymbol{\Pi}_{x,N})]) \quad \text{with } \theta_m \stackrel{\text{def}}{=} \frac{1}{2}(\theta_1 + \theta_2).$$

This criterion was first studied by Kaveh and Barabell [1] and Kaveh and Wang [93] in the resolution analysis of the conventional MUSIC and Min-Norm algorithms for two uncorrelated equal-powered sources and a ULA. This analysis has been extended to more general classes of situations, e.g., for two correlated or coherent equal-powered sources with the smoothed MUSIC algorithm [94], then for two unequal-powered sources impinging on an arbitrary array with the conventional and beamspace MUSIC algorithm [95]. A subsequent paper by Zhou et al. [96] developed a resolution measure based on the mean null spectrum and compared their results to Kaveh and Barabell's work.

For the second criterion, introduced by Sharman and Durrani [97] and then studied by Forster and Villier [26] in the context of the conventional MUSIC and Min-Norm algorithms for two uncorrelated equal-powered sources and a ULA, two sources are resolved if the second derivative of the mean null spectrum at the midpoint is negative

$$\frac{d^2 E[J_{\text{Alg}}(\theta, \boldsymbol{\Pi}_{x,N})]}{d\theta^2} \Big|_{\theta=\theta_m} \leq 0.$$

Resorting to an analysis based on perturbations of the noise projector  $\Pi_{x,N}$  [6], instead of those of the eigenvectors (e.g., [1, 95]), these two criteria have been studied for arbitrary distributions of the sources, for the conventional MUSIC algorithm. The following closed-form expressions of the approximation of the threshold ASNR given by these two criteria have been obtained in [27]:

$$\begin{aligned} \text{ASNR}_1 &\approx \frac{2}{N(\Delta\theta)^4} \left( 1 + \sqrt{1 + \frac{N(\Delta\theta)^2}{2\beta_M}} \right) \text{ and} \\ \text{ASNR}_2 &\approx \frac{1}{N(\Delta\theta)^4} \left( 1 + \sqrt{1 + \frac{N(\Delta\theta)^2}{\beta_M}} \right), \end{aligned} \quad (16.76)$$

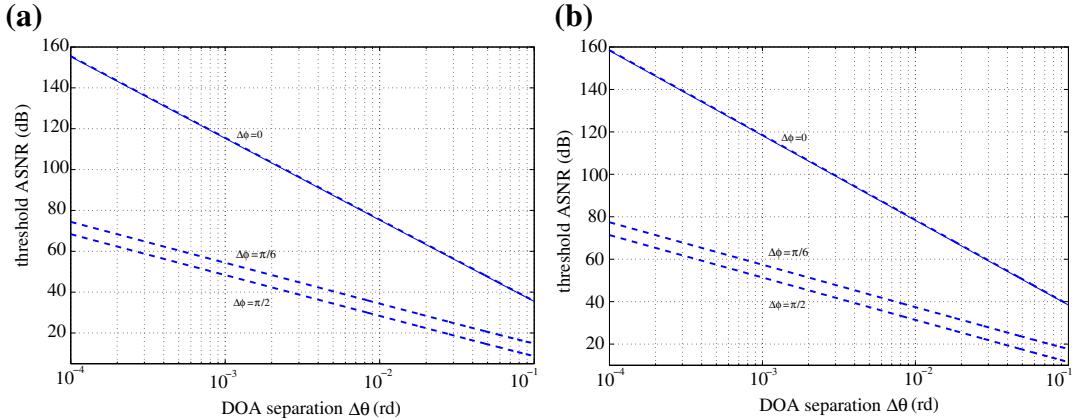
where  $\alpha_M$  and  $\beta_M$  are fractional expressions in  $M$  specified in [27] for ULAs. These expressions (16.76) have been extended in [27] to a noncircular MUSIC algorithm adapted to rectilinear signals, introduced and analyzed in [98], for which (16.76) becomes

$$\begin{aligned} \text{ASNR}_1 &\approx \frac{2}{N} \alpha_M^{\Delta\theta, \Delta\phi} \left( 1 + \sqrt{1 + \frac{N}{2\beta_M^{\Delta\theta, \Delta\phi}}} \right) \text{ and} \\ \text{ASNR}_2 &\approx \frac{1}{N} \alpha_M^{\Delta\theta, \Delta\phi} \left( 1 + \sqrt{1 + \frac{N}{\beta_M^{\Delta\theta, \Delta\phi}}} \right), \end{aligned} \quad (16.77)$$

where  $\Delta\phi \stackrel{\text{def}}{=} \phi_2 - \phi_1$  is the second-order noncircularity phase separation (16.8) and where now  $\alpha_M^{\Delta\theta, \Delta\phi}$  and  $\beta_M^{\Delta\theta, \Delta\phi}$  are expansions of  $1/(\Delta\theta)^2$  without constant term, whose coefficients depend on  $M$ ,  $\Delta\phi$  and the array configuration. Closed-form expressions of  $\alpha_M^{\Delta\theta, \Delta\phi}$  and  $\beta_M^{\Delta\theta, \Delta\phi}$  are given in [27] for weak and large second-order noncircularity phase separations and ULAs, where it is proved that  $\text{ASNR}_1$  and  $\text{ASNR}_2$  are decreasing functions of  $\Delta\phi$  and thus are minimum for  $\Delta\phi = \pi/2$ .

Figure 16.6 illustrates these two threshold ASNRs for two independent equal-powered BPSK modulated signals impinging on a ULA with  $M = 10$  and  $T = 500$ . We clearly see in this figure that the noncircular MUSIC algorithm outperforms the conventional MUSIC algorithm except for very weak second-order noncircularity phase separations or which the ASNR thresholds of these two algorithms are very similar. Furthermore, we note that the behaviors of the ASNR threshold given by the two criteria are very similar although the ASNR thresholds are slightly weaker for the Sharman and Durrani criterion than for the Cox criterion.

Moreover, several authors have considered (e.g., [99–101]) the probability of resolution or an approximation of it, based on the Cox criterion applied to the null sample spectrum to circumvent the possible misleading results given by these two criteria. Finally note that the resolution capability of the conventional and Capon beamforming algorithms have been thoroughly analyzed (see, e.g., [102]). Thanks to the simple expression of their spatial null spectra (16.46), it is possible to derive an approximation of the probability of resolution defined as the probability that the dip in midway between the two sources is at least 3-dB less than the peak of either source as a function of the SNR and DOA separation. Thus, fixing a specific high confidence level, this allows one to predict the SNR required to resolve two closely spaced sources. The superiority of the Capon algorithm is proved in [102], as the resolving

**FIGURE 16.6**

Comparison of the threshold ASNRs given by the Cox (a) and Sharman and Durrani (b) criteria as a function of the DOA separation  $\Delta\theta$  associated with the conventional MUSIC (—) and noncircular MUSIC algorithms (- -) for three values of the second-order noncircularity phase separation  $\Delta\phi$ .

power increases with SNR; in contrast, the Bartlett algorithm cannot exceed the Fourier/Rayleigh limit no matter how strong the signals.

### 3.16.6.2 Angular resolution limit based on the CRB

Array resolution has been studied independently of any algorithm by using the CRB. Based on the observation that the standard MUSIC algorithm is unlikely to resolve closely spaced signals if the standard deviation of the DOA estimates exceed  $\Delta\theta/8$  [3], Lee [103] has proposed to define the resolution limit as the DOA separation  $\Delta\theta$  for which

$$\max \left( \sqrt{\text{CRB}(\theta_1)}, \sqrt{\text{CRB}(\theta_2)} \right) = c \Delta\theta, \quad (16.78)$$

for the two closely spaced sources, where  $c$  is somewhat arbitrarily chosen. This criterion ignores the coupling between the estimates  $\hat{\theta}_1$  and  $\hat{\theta}_2$ . To overcome these drawbacks, Smith has proposed [18] to define the resolution limit as the source separation that equals the square root of its own CRB, i.e.,

$$\sqrt{\text{CRB}(\Delta\theta)} = c \Delta\theta, \quad (16.79)$$

with  $c = 1$ .<sup>9</sup> This means that the angular resolution limit or the threshold ASNR are obtained by resolving the implicit equations (16.78) and (16.79). This latter criterion has been applied to the deterministic modeling of the sources in [18] and then extended to multiple parameters per source in [104]. For the stochastic modeling of the sources, the circular Gaussian distribution has been compared to the discrete

<sup>9</sup>Note that this translation factor  $c$  is somewhat arbitrarily chosen (see different values cited in [17]).

one in [105]. In particular it has been proved that the threshold ASNR is inversely proportional to the number  $N$  of snapshots and to the square of  $\Delta\theta$  for the Gaussian case, in contrast to BPSK, MSK and QPSK case, for which it is inversely proportional to the fourth power of  $\Delta\theta$ .

### 3.16.6.3 Angular resolution limit based on the detection theory

The previous two approaches to characterize the angular resolution have in fact two different purposes. The first one studies the capability of a specific algorithm to estimate the DOAs of two closely spaced sources when the number of sources is known. In contrast, the second one is aiming to define an absolute limit on resolution that depends only of the array configuration and parameters of interest as the number  $M$  of sensors and SNR. But this latter approach based on the ad hoc relationships (16.78) and (16.79), essentially makes sense because the CRB indicates the parameter estimation accuracy and intuitively should be related to the resolution limit. But it suffers from two drawbacks. First, the resolution limit defined by this approach is not rigorously grounded in a statistical setting. Second, if the resolution limit is expressible by (16.78) or (16.79), can the translation factor  $c$ , be analytically determined?

To solve these two problems, Liu and Nehorai have proposed to use a hypothesis test formulation [17]. This approach has been introduced in a 3D reference frame, but to be consistent with the notations of this section, it is briefly summarized in the following in the 2D framework, where the DOA of a source is the parameter  $\theta$ . As the source localization accuracy may vary at different DOAs, consider the resolution limit at a specific DOA of interest. More precisely, assume there exists a source at a known DOA  $\theta_1$  and we are interested in the minimum angular separation  $\Delta\theta$  that the array can resolve between this source at  $\theta_1$  and another source at a direction  $\theta_2$  close to  $\theta_1$ . Quite naturally, the resolution of the two sources can be achieved through the binary composite hypothesis test

$$\begin{aligned} H_0: \Delta\theta = 0, & \quad \text{one source is present,} \\ H_1: \Delta\theta > 0, & \quad \text{two sources are present.} \end{aligned}$$

To rigorously define the resolution limit  $\Delta\theta$ , we fix the values of  $P_{\text{FA}}$  and  $P_{\text{D}}$  for this test. Otherwise,  $\Delta\theta$  could be arbitrary low, while the result of the test may be meaningless. Let  $\boldsymbol{\alpha} = [\Delta\theta, \boldsymbol{\beta}^T]^T$  be the unknown parameter of our statistical model, where  $\Delta\theta$  is the parameter of interest and  $\boldsymbol{\beta}$  gathers all the unknown nuisance parameters. To conduct this test, the GLRT is considered due to the unknown nuisance parameters:

$$L_G(\mathbf{x}, N) = \frac{p(\mathbf{x}; \hat{\Delta}\theta, \hat{\boldsymbol{\beta}}_1, H_1)}{p(\mathbf{x}; \hat{\boldsymbol{\beta}}_0, H_0)} H_1 > \gamma', \quad (16.80)$$

where  $p(\mathbf{x}; \Delta\theta, \boldsymbol{\beta}, H_1)$  and  $p(\mathbf{x}; \boldsymbol{\beta}, H_0)$  denote the probability density function of the measurement  $\mathbf{x} = [\mathbf{x}^T(t_1), \dots, \mathbf{x}^T(t_N)]^T$  under the hypothesis  $H_1$  and  $H_0$ , respectively.  $\hat{\Delta}\theta$  and  $\hat{\boldsymbol{\beta}}_1$  are respectively the ML estimate of  $\Delta\theta$  and  $\boldsymbol{\beta}$  under  $H_1$ , and  $\hat{\boldsymbol{\beta}}_0$  is the ML estimate of  $\boldsymbol{\beta}$  under  $H_0$ . The distribution of this GLRT  $L_G(\mathbf{x}, N)$  is generally very involved to derive, but hopefully, approximations of the distribution of  $2 \ln L_G(\mathbf{x}, N)$  for large values of  $N$  are available under  $H_0$  and  $H_1$ . First, under  $H_0$ , Wilk's theorem with nuisance parameters (see, e.g., [106, p. 132]) can be applied without having to know the exact form of  $L_G(\mathbf{x}, N)$ . This theorem states the following convergence in distribution when  $N$  tends to  $\infty$ :

$$2 \ln L_G(\mathbf{x}, N) \xrightarrow{\mathcal{L}} \chi^2(1) \quad \text{under } H_0, \quad (16.81)$$

where  $\chi^2(1)$  denotes the central chi-square distribution with one degree of freedom (associated with the single parameter  $\Delta\theta$ ). Under  $H_1$ , the derivation of the asymptotic distribution of  $2 \ln L_G(\mathbf{x}, N)$  is much more involved. Using a theoretical result by Stroud [107], Stuart et al. [108, Chapter 14.7] have stated that when  $\Delta\theta$  can take values<sup>10</sup> near  $\mathbf{0}$ ,  $2 \ln L_G(\mathbf{x}, N)$  is approximately distributed<sup>11</sup> as

$$2 \ln L_G(\mathbf{x}, N) \xrightarrow{a} \chi^2(1, \lambda_N) \quad \text{under } H_1, \quad (16.82)$$

where  $\chi^2(1, \lambda_N)$  denotes the noncentral chi-squared distribution with one degree of freedom and noncentrality parameter  $\lambda_N$  given by (see [109, Section 6.5])

$$\lambda_N = (\Delta\theta - 0)([\mathbf{FIM}^{-1}(\boldsymbol{\alpha})]_{1,1})^{-1}(\Delta\theta - 0), \quad (16.83)$$

whose dependence on  $N$  in the FIM of  $\boldsymbol{\alpha}$  is emphasized, and where  $[\mathbf{FIM}^{-1}(\boldsymbol{\alpha})]_{1,1}$  denotes the (1,1) th entry of  $\mathbf{FIM}^{-1}(\boldsymbol{\alpha})$ . It is further shown [109, Appendix 6C] that as  $N$  is large, (16.83) is approximated by

$$\lambda_N \approx (\Delta\theta)^2 ([\mathbf{FIM}^{-1}(\boldsymbol{\alpha})]_{1,1}|_{\Delta\theta=0})^{-1} = \text{CRB}^{-1}(\Delta\theta)|_{\Delta\theta=0}. \quad (16.84)$$

Based on these limit and approximate distributions of  $2 \ln L_G(\mathbf{x}, N)$  under  $H_0$  and  $H_1$  for which the GLRT in (16.81) can be rewritten as

$$2 \ln L_G(\mathbf{x}, N) \stackrel{H_1}{>} \gamma \stackrel{\text{def}}{=} 2 \ln \gamma', \quad (16.85)$$

the angular resolution limit (ARL) has been computed in [17] by using the two constraints

$$P_{\text{FA}} = Q_{\chi^2(1)}(\gamma) \quad \text{and} \quad P_{\text{D}} = Q_{\chi^2(1, \lambda_N)}(\gamma),$$

where the values of  $P_{\text{FA}}$  and  $P_{\text{D}}$  are fixed and where  $Q_{\chi^2(1)}$  and  $Q_{\chi^2(1, \lambda_N)}$  denote the right tail probability of the  $\chi^2(1)$  and  $\chi^2(1, \lambda_N)$  distributions, respectively. It assumes the form

$$\Delta\theta = \sqrt{\lambda_K} \sqrt{\text{CRB}(\Delta\theta)|_{\Delta\theta=0}},$$

where the factor  $\sqrt{\lambda_K}$  is analytically determined by the preassigned values of  $P_{\text{FA}}$  and  $P_{\text{D}}$ . Note that the SNR is embedded in the expression of  $\text{CRB}(\Delta\theta)$  that is proportional to  $K$ . The dependence on the SNR of the CRB may vary according to the distribution of the sources. For example, Delmas and Abeida [105] proves that the CRB of the DOA separation of discrete sources is very different from those of Gaussian sources.

---

<sup>10</sup>The following more formal condition is given in [107],  $\Delta\theta$  is embedded in an adequate sequence indexed by  $N$  that converges to zero at the rate  $N^{-1/2}$  or faster, i.e.,  $\|\Delta\theta\| = O(1/N^{1/2})$ . Note the simplified condition given by Kay [109, A. 6A]:  $\|\Delta\theta\| = c/\sqrt{N}$  for some constant  $c$ , that is reduced to the rough assumption of weak SNR [109, Section 6.5].

<sup>11</sup>The accurate formulation is  $\lim_{N \rightarrow \infty} \{P(2 \ln L_G(\mathbf{x}, N)) < t\} - P(V_N < t)\} = 0 \forall t$ , where  $V_N$  has a noncentral chi-squared distribution with one degree of freedom and noncentrality parameter  $\mu_N$  that depends on the data length  $N$ .

*Relevant Theory:* Statistical Signal Processing

See this Volume, [Chapter 1](#) Introduction: Statistical Signal Processing

See this Volume, [Chapter 2](#) Model Selection

See this Volume, [Chapter 8](#) Performance Analysis and Bounds

## References

- [1] M. Kaveh, A.J. Barabell, The statistical performance of the MUSIC and the Minimum-Norm algorithms in resolving plane waves in noise, *IEEE Trans. ASSP* 34 (2) (1986) 331–341.
- [2] B. Porat, B. Friedlander, Analysis of the asymptotic relative efficiency of the MUSIC algorithm, *IEEE Trans. ASSP* 36 (4) (1988) 532–544.
- [3] P. Stoica, A. Nehorai, MUSIC, maximum likelihood, and Cramer-Rao bound, *IEEE Trans. ASSP* 37 (5) (1989) 720–741.
- [4] P. Stoica, A. Nehorai, MUSIC, maximum likelihood, and Cramer-Rao bound: further results and comparisons, *IEEE Trans. ASSP* 38 (12) (1990) 2140–2150.
- [5] W. Xu, K.M. Buckley, Bias analysis of the MUSIC location estimator, *IEEE Trans. Signal Process.* 40 (10) (1992) 2559–2569.
- [6] H. Krim, P. Forster, G. Proakis, Operator approach to performance analysis of root-MUSIC and root-min-norm, *IEEE Trans. Signal Process.* 40 (7) (1992) 1687–1696.
- [7] A. Gorokhov, Y. Abramovich, J.F. Böhme, Unified analysis of DOA estimation algorithms for covariance matrix transforms, *Signal Process.* 55 (1996) 107–115.
- [8] F. Li, R.J. Vaccaro, Unified analysis for DOA estimation algorithms in array signal processing, *Signal Process.* 25 (2) (1991) 147–169.
- [9] F. Li, H. Liu, R.J. Vaccaro, Performance analysis for DOA estimation algorithms: unification, simplification, and observations, *IEEE Trans. Aerosp. Electron. Syst.* 29 (4) (1993) 1170–1184.
- [10] B. Ottersten, M. Viberg, T. Kailath, Analysis of subspace fitting and ML techniques for parameter estimation from sensor array data, *IEEE Trans. Signal Process.* 40 (3) (1992) 590–599.
- [11] P. Stoica, A. Nehorai, Performance study of conditional and unconditional direction of arrival estimation, *IEEE Trans. ASSP* 38 (10) (1990) 1783–1795.
- [12] B. Ottersten, M. Viberg, P. Stoica, A. Nehorai, Exact and large sample maximum likelihood techniques for parameter estimation and detection in array processing, in: S. Haykin, J. Litva, T.J. Shepherd (Eds.), *Radar Array Processing*, Springer-Verlag, Berlin, 1993, pp. 99–151.
- [13] M. Wax, T. Kailath, Detection of signals by information theoretic criteria, *IEEE Trans. ASSP* 33 (2) (1985) 387–392.
- [14] J.P. Delmas, Y. Meurisse, On the second-order statistics of the EVD of sample covariance matrices—application to the detection of noncircular or/and nonGaussian components, *IEEE Trans. Signal Process.* 59 (8) (2011) 4017–4023.
- [15] E. Fishler, M. Grosmann, H. Messer, Detection of signals by information theoretic criteria: general asymptotic performance analysis, *IEEE Trans. Signal Process.* 50 (5) (2002) 1027–1036.
- [16] L.C. Zhao, P.R. Krishnaiah, Z.D. Bai, On detection of the number of signals in the presence of white noise, *J. Multivariate Anal.* 20 (1) (1986) 1–20.
- [17] Z. Liu, A. Nehorai, Statistical angular resolution limit for point sources, *IEEE Trans. Signal Process.* 55 (11) (2007) 5521–5527.
- [18] S.T. Smith, Statistical resolution limits and the complexified Cramer-Rao bound, *IEEE Trans. Signal Process.* 53 (5) (2005) 1597–1609.

- [19] J.F. Cardoso, E. Moulines, Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants, *IEEE Trans. Signal Process.* 43 (1) (1995) 214–224.
- [20] J.P. Delmas, Asymptotic performance of second-order algorithms, *IEEE Trans. Signal Process.* 50 (1) (2002) 49–57.
- [21] P. Stoica, R. Moses, *Introduction to Spectral Analysis*, Prentice Hall, Inc., 1997.
- [22] M. Wax, I. Ziskind, On unique localization of multiple sources by passive sensor arrays, *IEEE Trans. ASSP* 37 (7) (1989) 996–1000.
- [23] A. Nehorai, D. Starer, P. Stoica, Direction of arrival estimation with multipath and few snapshots, *Circ. Syst. Signal Process.* 10 (3) (1991) 327–342.
- [24] J.P. Delmas, Asymptotically minimum variance second-order estimation for non-circular signals with application to DOA estimation, *IEEE Trans. Signal Process.* 52 (5) (2004) 1235–1241.
- [25] A. Ferreol, P. Larzabal, M. Viberg, On the resolution probability of MUSIC in presence of modeling errors, *IEEE Trans. Signal Process.* 56 (5) (2008) 1945–1953.
- [26] P. Forster, E. Villier, Simplified formulas for performance analysis of MUSIC and Min Norm, in: *Proceedings of Ocean Conference*, September 1998.
- [27] H. Abeida, J.P. Delmas, MUSIC-like estimation of direction of arrival for non-circular sources, *IEEE Trans. Signal Process.* 54 (7) (2006) 2678–2690.
- [28] A. Renaux, P. Forster, E. Boyer, P. Larzabal, Unconditional maximum likelihood performance at finite number of samples and high signal to noise ratio, *IEEE Trans. Signal Process.* 55 (5) (2007) 2358–2364.
- [29] P. Vallet, P. Loubaton, X. Mestre, Improved subspace estimation for multivariate observations of high dimension: the deterministic signal case, *IEEE Trans. Inform. Theory* 58 (2) (2012) 1002–3234.
- [30] J.P. Delmas, H. Abeida, Asymptotic distribution of circularity coefficients estimate of complex random variables, *Signal Process.* 89 (2009) 2670–2675.
- [31] J.P. Delmas, Y. Meurisse, Asymptotic performance analysis of DOA algorithms with temporally correlated narrow-band signals, *IEEE Trans. Signal Process.* 48 (9) (2000) 2669–2674.
- [32] T. Kato, *Perturbation Theory for Linear Operators*, Springer-Verlag, Berlin, 1995.
- [33] R.J. Serfling, *Approximation Theorems of Mathematical Statistics*, John Wiley and Sons, 1980.
- [34] P.J. Schreier, L.L. Scharf, *Statistical Signal Processing of Complex-Valued Data—The Theory of Improper and Noncircular Signals*, Cambridge University Press, 2010.
- [35] E.L. Lehmann, *Elements of Large-Sample Theory*, Springer-Verlag, New-York, 1999.
- [36] W. Xu, M. Kaveh, Analysis of the performance and sensitivity of eigendecomposition-based detectors, *IEEE Trans. Signal Process.* 43 (6) (1995) 1413–1426.
- [37] P. Stoica, R.L. Moses, On biased estimators and the unbiased Cramer-Rao lower bound, *Signal Process.* 21 (1990) 349–350.
- [38] D.T. Vu, A. Renaux, R. Boyer, S. Marcos, Closed-form expression of the Weiss-Weinstein bound for 3D source localization: the conditional case, in: *Proceedings of SAM*, Jerusalem, Israel, October 2010.
- [39] W.J. Bangs, Array processing with generalized beamformers, Ph.D. Thesis, Yale University, New Haven, CT, 1971.
- [40] D. Slepian, Estimation of signal parameters in the presence of noise, *Trans. IRE Prof. Group Inform. Theory PG IT-3*, 1954, pp. 68–89.
- [41] P. Stoica, A.G. Larsson, A.B. Gershman, The stochastic CRB for array processing: a textbook derivation, *IEEE Signal Process. Lett.* 8 (5) (2001) 148–150.
- [42] J.P. Delmas, H. Abeida, Stochastic Cramer-Rao bound for non-circular signals with application to DOA estimation, *IEEE Trans. Signal Process.* 52 (11) (2004) 3192–3199.
- [43] P. Stoica, B. Ottersten, M. Viberg, R.L. Moses, Maximum likelihood array processing for stochastic coherent sources, *IEEE Trans. Signal Process.* 44 (1) (1996) 96–105.

- [44] H. Abeida, J.P. Delmas, Efficiency of subspace-based DOA estimators, *Signal Process.* 87 (9) (2007) 2075–2084.
- [45] A.B. Gershman, P. Stoica, M. Pesavento, E.G. Larsson, Stochastic Cramer-Rao bound for direction estimation in unknown noise fields, *IEE Proc—Radar Sonar Navig.* 149 (1) (2002) 2–8.
- [46] M. Pesavento, A.B. Gershman, Maximum-likelihood direction of arrival estimation in the presence of unknown nonuniform noise, *IEEE Trans. Signal Process.* 49 (7) (2001) 1310–1324.
- [47] H. Ye, R.D. Degroat, Maximum likelihood DOA estimation and asymptotic Cramer-Rao bounds for additive unknown colored noise, *IEEE Trans. Signal Process.* 43 (4) (1995) 938–949.
- [48] H. Abeida, J.P. Delmas, Cramer-Rao bound for direction estimation of non-circular signals in unknown noise fields, *IEEE Trans. Signal Process.* 53 (12) (2005) 4610–4618.
- [49] M. Jansson, B. Göransson, B. Ottersten, Subspace method for direction of arrival estimation of uncorrelated emitter signals, *IEEE Trans. Signal Process.* 47 (4) (1999) 945–956.
- [50] M. Hawkes, A. Nehorai, Acoustic vector-sensor beamforming and Capon direction estimation, *IEEE Trans. Signal Process.* 46 (9) (1998) 2291–2304.
- [51] A. Nehorai, E. Paldi, Vector-sensor array processing for electromagnetic source localization, *IEEE Trans. Signal Process.* 42 (2) (1994) 376–398.
- [52] Y. Begriche, M. Thameri, K. Abed-Meraim, Exact Cramer-Rao bound for near field source localization, in: International Conference on ISSPA, Montreal, July 2012.
- [53] M.N. El Korso, R. Boyer, A. Renaux, S. Marcos, Conditional and unconditional Cramer-Rao bounds for near-field source localization, *IEEE Trans. Signal Process.* 58 (5) (2010) 2901–2907.
- [54] J.P. Delmas, H. Gazzah, Analysis of near-field source localization using uniform circular arrays, in: International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013), Vancouver, Canada, May 2013.
- [55] J.P. Delmas, H. Abeida, Cramer-Rao bounds of DOA estimates for BPSK and QPSK modulated signals, *IEEE Trans. Signal Process.* 54 (1) (2006) 117–126.
- [56] F. Bellili, S.B. Hassen, S. Affes, A. Stephenne, Cramer-Rao lower bounds of DOA estimates from square QAM-modulated signals, *IEEE Trans. Commun.* 59 (6) (2011) 1675–1685.
- [57] M. Lavielle, E. Moulines, J.F. Cardoso, A maximum likelihood solution to DOA estimation for discrete sources, in: Proceedings of Seventh IEEE Workshop on SP, 1994, pp. 349–353.
- [58] B. Porat, B. Friedlander, Performance analysis of parameter estimation algorithms based on high-order moments, *Int. J. Adapt. Control Signal Process.* 3 (1989) 191–229.
- [59] B. Friedlander, B. Porat, Asymptotically optimal estimation of MA and ARMA parameters of non-Gaussian processes from high-order moments, *IEEE Trans. Automat. Control* 35 (1990) 27–35.
- [60] B. Porat, B. Friedlander, Direction finding algorithms based on higher order statistics, *IEEE Trans. Signal Process.* 39 (9) (1991) 2016–2024.
- [61] H. Abeida, J.P. Delmas, Asymptotically minimum variance estimator in the singular case, in: Proceedings of EUSIPCO, Antalya, September 2005.
- [62] C. Vaidyanathan, K.M. Buckley, Performance analysis of the MVDR spatial spectrum estimator, *IEEE Trans. Signal Process.* 43 (6) (1995) 1427–1437.
- [63] H. Gazzah, J.P. Delmas, Spectral efficiency of beamforming-based parameter estimation in the single source case, in: SSP 2011, Nice, June 2011.
- [64] A.G. Jaffer, Maximum likelihood direction finding of stochastic sources: a separable solution, in: Proceedings of ICASSP, New York, April 11–14 1988, pp. 2893–2896.
- [65] P. Stoica, N. Nehorai, On the concentrated stochastic likelihood function in array processing, *Circ. Syst. Signal Process.* 14 (1995) 669–674.
- [66] M. Viberg, B. Ottersten, Sensor array signal processing based on subspace fitting, *IEEE Trans. ASSP* 39 (5) (1991) 1110–1121.

- [67] A. Renaux, P. Forster, E. Chaumette, P. Larzabal, On the high SNR conditional maximum likelihood estimator full statistical characterization, *IEEE Trans. Signal Process.* 54 (12) (2006) 4840–4843.
- [68] M. Viberg, B. Ottersten, A. Nehorai, Performance analysis of direction finding with large arrays and finite data, *IEEE Trans. Signal Process.* 43 (2) (1995) 469–477.
- [69] B.A. Johnson, Y.I. Abramovich, X. Mestre, MUSIC, G-MUSIC, and maximum-likelihood performance breakdown, *IEEE Trans. Signal Process.* 56 (8) (2008) 3944–3958.
- [70] J.F. Cardoso, E. Moulines, Invariance of subspace based estimator, *IEEE Trans. Signal Process.* 48 (9) (2000) 2495–2505.
- [71] J.F. Cardoso, E. Moulines, A robustness property of DOA estimators based on covariance, *IEEE Trans. Signal Process.* 42 (11) (1994) 3285–3287.
- [72] J.P. Delmas, J.F. Cardoso, Performance analysis of an adaptive algorithm for tracking dominant subspace, *IEEE Trans. Signal Process.* 46 (11) (1998) 3045–3057.
- [73] J.P. Delmas, J.F. Cardoso, Asymptotic distributions associated to Oja's learning equation for Neural Networks, *IEEE Trans. Neural Networks* 9 (6) (1998) 1246–1257.
- [74] J. Sorelius, R.L. Moses, T. Söderström, A.L. Swindlehurst, Effects of nonzero bandwidth on direction of arrival estimators in array processing, *IEE Proc.—Radar Sonar Navig.* 145 (6) (1998) 317–324.
- [75] J.P. Delmas, Y. Meurisse, Robustness of narrowband DOA algorithms with respect to signal bandwidth, *Signal Process.* 83 (3) (2003) 493–510.
- [76] A. Ferreol, P. Larzabal, M. Viberg, On the asymptotic performance analysis of subspace DOA estimation in the presence of modeling errors: case of MUSIC, *IEEE Trans. Signal Process.* 54 (3) (2006) 907–920.
- [77] B. Friedlander, A sensitivity analysis of the MUSIC algorithm, *IEEE Trans. ASSP* 38 (10) (1990) 1740–1751.
- [78] B. Friedlander, A sensitivity analysis of the maximum likelihood direction-finding algorithm, *IEEE Trans. Aerosp. Electron. Syst.* 26 (11) (1990) 953–958.
- [79] A.L. Swindlehurst, T. Kailath, A performance analysis of subspace-based methods in the presence of model errors. Part I: The MUSIC algorithm, *IEEE Trans. Signal Process.* 40 (7) (1992) 1758–1773.
- [80] A. Ferreol, P. Larzabal, M. Viberg, Performance prediction of maximum likelihood direction of arrival estimation in the presence of modeling error, *IEEE Trans. Signal Process.* 56 (10) (2008) 4785–4793.
- [81] M. Viberg, Sensitivity of parametric direction finding to colored noise fields and undermodeling, *Signal Process.* 34 (2) (1993) 207–222.
- [82] M. Viberg, A.L. Swindlehurst, Analysis of the combined effects of finite samples and model errors on array processing performance, *IEEE Trans. Signal Process.* 42 (11) (1994) 3073–3083.
- [83] P. Chevalier, A. Ferreol, On the virtual array concept for the fourth-order direction finding problem, *IEEE Trans. Signal Process.* 47 (9) (1999) 2592–2595.
- [84] P. Chevalier, A. Ferreol, L. Albera, High resolution direction finding from higher order statistics; the 2q-MUSIC algorithm, *IEEE Trans. Signal Process.* 54 (8) (2006) 2986–2997.
- [85] J. Rissanen, Modeling by shortest data description, *Automatica* 14 (1978) 465–471.
- [86] M. Wax, I. Ziskind, Detection of the number of coherent signals by the MDL principle, *IEEE Trans. ASSP* 37 (8) (1989) 1190–1196.
- [87] T.W. Anderson, Asymptotic theory for principal component analysis, *Ann. Math. Stat.* 34 (1963) 122–148.
- [88] M. Kaveh, H. Wang, H. Hung, On the theoretic performance of a class of estimators of the number of narrow-band sources, *IEEE Trans. ASSP* 35 (9) (1987) 1350–1352.
- [89] H. Wang, M. Kaveh, On the performance of signal subspace processing—Part I: Narrow-band systems, *IEEE Trans. ASSP* 34 (5) (1986) 1201–1209.
- [90] F. Haddadi, M.M. Mohammadi, M.M. Nayebi, M.R. Aref, Statistical performance analysis of MDL source enumeration in array processing, *IEEE Trans. Signal Process.* 58 (1) (2010) 452–457.

- [91] R.R. Nadakuditi, A. Edelman, Sample eigenvalue based detection of high-dimensional signals in white noise using relatively few samples, *IEEE Trans. Signal Process.* 56 (17) (2008) 2625–2638.
- [92] H. Cox, Resolving power and sensitivity to mismatch of optimum array processors, *J. Acoust. Soc. Am.* 54 (3) (1973) 771–785.
- [93] M. Kaveh, H. Wang, Threshold properties of narrowband signal subspace array processing methods, in: S. Haykin (Ed.), *Advances in Spectrum Analysis and Array Processing*, vol. 2, Prentice-Hall, pp. 173–220.
- [94] S.U. Pillai, G.H. Kwon, Performance analysis of MUSIC-type high resolution estimators for direction finding in correlated and coherent scenes, *IEEE Trans. ASSP* 37 (8) (1989) 1176–1189.
- [95] H.B. Lee, M.S. Wengrovitz, Resolution threshold of beamspace MUSIC for two closely spaced emitters, *IEEE Trans. ASSP* 38 (9) (1990) 1445–1559.
- [96] C. Zhou, F. Haber, D.L. Jaggard, A resolution measure for the MUSIC algorithm and its application to plane wave arrivals contaminated by coherent interference, *IEEE Trans. Signal Process.* 39 (2) (1991) 454–463.
- [97] K.C. Sharman, S.T. Durrani, Resolving power of signal subspace methods for finite data lengths, in: *Proceedings of ICASSP*, Tampa, Florida, April 1985.
- [98] H. Abeida, J.P. Delmas, Statistical performance of MUSIC-like algorithms in resolving noncircular sources, *IEEE Trans. Signal Process.* 56 (9) (2008) 4317–4329.
- [99] H.B. Lee, M.S. Wengrovitz, Statistical characterization of the MUSIC algorithm null spectrum, *IEEE Trans. Signal Process.* 39 (6) (1991) 1333–1347.
- [100] Q.T. Zhang, Probability of resolution of the MUSIC algorithm, *IEEE Trans. Signal Process.* 43 (4) (1995) 978–987.
- [101] Q.T. Zhang, A statistical resolution theory of the beamformer-based spatial spectrum for determining the directions of signals in white noise, *IEEE Trans. ASSP* 43 (8) (1995) 1867–1873.
- [102] C.D. Richmond, Capon algorithm mean-squared error threshold SNR prediction and probability of resolution, *IEEE Trans. Signal Process.* 53 (8) (2005) 2748–2764.
- [103] H.B. Lee, The Cramer-Rao bound on frequency estimates of signals closely spaced in frequency, *IEEE Trans. Signal Process.* 40 (6) (1992) 1508–1517.
- [104] M.N. El Korso, R. Boyer, A. Renaux, S. Marcos, Statistical resolution limit for multiple parameters of interest and for multiple signals, in: *Proceedings of ICASSP*, Dallas, May 2010.
- [105] J.P. Delmas, H. Abeida, Statistical resolution limits of DOA for discrete sources, in: *Proceedings of ICASSP*, Toulouse, May 2006.
- [106] G.A. Young, R.L. Smith, *Essentials of Statistical Inference*, Cambridge Series in Statistical and Probabilistic Mathematics, 2005.
- [107] T.W.F. Stroud, Fixed alternatives and Wald's formulation of the noncentral asymptotic behavior of the likelihood ratio statistics, *Ann. Math. Stat.* 43 (2) (1972) 447–454.
- [108] A. Stuart, J.K. Ord, *Advanced Theory of Statistics*, fifth ed., vol. 2, Edward Arnold, 1991.
- [109] S.M. Kay, *Fundamentals of Statistical Signal Processing, Detection Theory*, vol. II, Prentice-Hall, 1998.

# DOA Estimation of Nonstationary Signals

# 17

**Moeness G. Amin and Yimin D. Zhang**

*Center for Advanced Communications, Villanova University, Villanova, PA, USA*

## 3.17.1 Introduction

For many decades, time-frequency (t-f) signal representations, such as the Wigner-Ville distribution (WVD) and spectrograms, were only applied to analyze nonstationary signals incident on single-sensor receivers. The objective is to characterize the data observations in the t-f domain, leading to proper signal detection and characterization, separation, classification, and cancelation. These offerings were subsequently enhanced by introducing reduced interference quadratic time-frequency distributions (TFDs) which have led to improved multi-component signal power localizations in the t-f domain. With sensor arrays being ubiquitous in many application areas of signal processing, such as communications, radar, acoustic, and biomedical, it has become important to consider TFDs in the context of array processing. This is successfully achieved within the framework of spatial time-frequency distribution (STFD) [1–6].

This chapter discusses the direction-of-arrival (DOA) estimation of far-field sources producing nonstationary signals, particularly those in the form of frequency modulated (FM) waveforms. We use DOA estimation methods based on the quadratic (bilinear) STFD framework. Nonstationary signals are encountered in various passive and active arrays using different sensing modalities. For example, many modern radar systems use linear FM (LFM) signals to achieve pulse compression (LFM signals are also referred to as chirp signals). Alternatively, accelerating, rotating, or maneuvering targets generate time-varying Doppler frequencies, with well defined Doppler-frequency signatures [7,8]. Further, FM signals can be easily generated and, as such, they are considered the preferred waveforms for smart jammers, and have proven effective in hindering and compromising communications and radar receivers [9].

The STFD framework was first developed by Belouchrani and Amin for the blind separation of narrowband nonstationary signals [4]. It was shown that the STFD matrix is related to the source TFD matrix by the spatial mixing matrix in a manner similar to the commonly used expression, in narrowband array processing problems, relating the sensor spatial covariance matrix to the source covariance matrix. The same STFD framework was then used for direction finding of nonstationary signals using t-f based DOA estimation approaches, such as t-f MUSIC, t-f root-MUSIC, t-f maximum likelihood (t-f ML), and t-f ESPRIT [5,10–15]. These techniques, when applied to the source t-f signatures, have shown improved performance, compared to conventional DOA estimation approaches. The latter, which directly operate on the data or its covariance matrix, do not account for,

or properly utilize, the instantaneous frequency (IF) characterizations of the source signals. Comprehensive analyses, supporting STFD-based DOA estimations and demonstrating the robustness of the signal and noise subspaces associated with the STFD matrices, were provided by Zhang et al. [6]. It was shown that, signal-to-noise ratio (SNR) enhancement can be obtained from constructing the STFD matrix incorporating signal power concentration regions in the time-frequency domain. This, in turn, leads to robustness of the signal and noise subspace estimates compared to their counterparts, which are obtained using the data covariance matrix. In addition, masking and filtering of the source t-f signatures allow separation and, subsequently, the consideration of individual, or a subgroup of, sources in the field of view. With such source discriminatory capability, the receiver can handle and process more signals than sensors. Reduction of sources further increases SNR and lowers the mutual interference between the signals, improving signal and noise subspace estimations. In some applications, particularly when acoustic signals are involved, the TFD of multiple signals can be approximately disjoint or orthogonal, i.e., only one signal is present at a given t-f point. With a single source considered at a time, the source DOA can be estimated using only two receivers [16]. In essence, DOA estimation for nonstationary signals, which are well separated in the t-f domain, should be performed using t-f methods.

It is recognized that the advantages of t-f based DOA estimation can only be materialized if appropriate t-f points are selected in the formulation of the STFD matrices. While in some scenarios, the selection of t-f points of peak power may be relatively simple, the problem may become challenging in other situations, e.g., when the signals are highly contaminated by noise. Different approaches for t-f point selection are summarized in [17] and references therein. Utilization of the spatial degrees-of-freedom, or spatial diversity, embedded in the STFD matrix, can reduce noise and enhance the t-f signatures of the signals of interest without the need of using robust t-f kernels [18]. A simple example for achieving this task is through an average of the TFDs over all receive sensors [19]. Insights into the characteristics of TFDs corresponding to different sensors can also assist in the identification of auto-term and cross-term points [20,21]. It is worthnoting that the separation of the auto-term t-f points from cross-term t-f points in general is less of an issue in DOA estimation than in blind source separation applications. This is because, in performing DOA estimation, STFD matrices only need to meet the full rank requirement [10]. This requirement can be satisfied with the inclusion of cross-terms. Nevertheless, the capability of separating auto-term points from cross-term points allows the selection of t-f regions corresponding to a subset of sources for proper source discrimination.

This chapter focuses on the DOA estimation approaches of nonstationary signals based on the STFD framework. An analogous framework can be provided using linear transforms, such as the short-time Fourier transform (STFT) and wavelet transform, both can achieve power localization and SNR enhancement. However, multi-resolution analysis is not most effective when dealing with signals characterized by their IF laws, which is the assumption made throughout this chapter. The STFT, on the other hand, has a well known shortcoming, trading off temporal and spectral resolutions, and its magnitude square is already considered within the STFD framework. Nevertheless, recent advances on the exploitation of fractional Fourier transform (FrFT) and signal stationarization for DOA estimations are considered in this chapter.

In addition to STFDs, we also review relevant recent approaches for nonstationary signal DOA estimations. One of these approaches is the exploitation of spatial joint-variable distributions (SJVD), such as spatial ambiguity function (SAF). The latter employs the Doppler and time-lag variables, in lieu of the time and frequency variables [22]. Another important extension of the STFD and SJVD based DOA

estimation methods is the consideration of wideband signals where the narrowband assumption does not hold, i.e., the steering vector varies with the frequency within the signal bandwidth [11,23,24]. The latest application of STFD to multiple-input multiple-output (MIMO) radar systems for joint direction-of-departure (DOD) and DOA estimations [25] is also introduced.

The following notations are used in this chapter. A lower (upper) case bold letter denotes a vector (matrix).  $E[\cdot]$  represents statistical mean operation.  $(\cdot)^*$ ,  $(\cdot)^T$  and  $(\cdot)^H$  respectively denote complex conjugation, transpose, and conjugate transpose (Hermitian) operations.  $\text{Re}(\cdot)$  represents the real part operation of a complex variable, vector or matrix.  $\odot$  denotes the Hadamard product,  $\otimes$  is the Kronecker product, and  $\diamond$  denotes the Khatri-Rao product.  $\mathbf{I}_n$  expresses the  $n \times n$  identity matrix.  $\text{Diag}(\mathbf{x})$  denotes a diagonal matrix using the elements of  $\mathbf{x}$  as its diagonal elements,  $\text{diag}(\mathbf{X})$  a vector consisting of the diagonal elements of matrix  $\mathbf{X}$ , and  $\text{vec}(\mathbf{X})$  a vectorized result of matrix  $\mathbf{X}$ .  $\det(\cdot)$  and  $\text{tr}(\cdot)$  respectively denote the determinant and trace of a matrix. In addition,  $\mathbb{C}^{N \times M}$  denotes the complete set of  $N \times M$  complex entries,  $[\mathbf{a}]_n$  denotes the  $n$ th element of vector  $\mathbf{a}$ , and  $[\mathbf{A}]_{m,n}$  denotes the  $(m,n)$ th element of matrix  $\mathbf{A}$ .  $\delta_{i,l}$  is the Kronecker delta function which equals to 1 when  $i = l$  and 0 otherwise.

## 3.17.2 Nonstationary signals and time-frequency representations

### 3.17.2.1 Nonstationary signals

A large class of signal processing techniques deal with deterministic or stochastic time-domain signals that are stationary. A deterministic signal is said to be stationary if it can be written as a discrete sum of sinusoids, whereas in the random case, a signal  $x(t)$  is said to be wide-sense stationary (or stationary up to the second order) if its expectation is independent of time, and its autocorrelation function  $E[x(t_1)x^*(t_2)]$  depends only on the time difference  $t_2 - t_1$  [26]. For stationary signals, parameters such as the mean and variance, if they exist, also do not change over time or space. For this type of signals, the Fourier transform is widely used to extract the frequency-domain information from the time-domain signals and also as a pre-processing step for various temporal, spatial, and spatio-temporal processing methods.

Many real-world signals, however, are nonstationary. A signal is referred to as nonstationary if one of the fundamental assumptions of stationary signals is no longer valid. For example, a finite duration signal, in particular a transient signal (for which the length is short compared to the observation duration), is nonstationary. Also, many nonstationary signals have their frequency contents and properties change with time. Signals with time-varying spectra include: the impulse response of a wireless communication channel, radar and sonar acoustic waves, seismic acoustic waves, biomedical signals, such as electrocardiogram (ECG) or neonatal seizures, biological signals, such as bat or dolphin echolocation sounds, vocals in speech, notes in music, engine noise, shock waves in fault structures and jamming signals [27].

### 3.17.2.2 Cohen's class of time-frequency representations

There are a number of ways one can perform t-f analysis for nonstationary signals, most of which fall into the following two classes: linear t-f analysis and bilinear t-f analysis. Short-time Fourier transform (STFT), fractional Fourier transform (FrFT) and wavelet transform are commonly used techniques to perform linear t-f analysis [26,28]. In contrast with linear t-f representations, which decompose the signal to basis functions, or elementary components (the atoms), the bilinear t-f representations,

introduced in the following subsection, distribute the signal power over two description variables: time and frequency.

The Cohen's class of TFDs is the foundation of the STFD framework for direction finding as shown in Sections 3.17.3 and 3.17.4. The Cohen's class of auto-term TFDs of a narrowband signal  $x(t)$  is defined as [29, 30],

$$D_{xx}(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(t-u, \tau) x\left(u + \frac{\tau}{2}\right) x^*\left(u - \frac{\tau}{2}\right) e^{-j2\pi f\tau} du d\tau, \quad (17.1)$$

where  $t$  and  $f$  represent the time and frequency indexes, respectively,  $\phi(t, \tau)$  is the t-f kernel, and  $\tau$  is the time-lag variable.

The cross-term TFD of two signals  $x_i(t)$  and  $x_k(t)$  is defined as

$$D_{x_i x_k}(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(t-u, \tau) x_i\left(u + \frac{\tau}{2}\right) x_k^*\left(u - \frac{\tau}{2}\right) e^{-j2\pi f\tau} du d\tau. \quad (17.2)$$

In practice, TFDs are often evaluated using their discrete-time forms [31]. To use integer time delay  $\tau$ , we rewrite (17.1) as

$$D_{xx}(t, f) = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(t-u, 2\tau) x(u+\tau) x^*(u-\tau) e^{-j4\pi f\tau} du d\tau. \quad (17.3)$$

The discrete form of the auto-term TFD corresponding to (17.3) is typically expressed as

$$D_{xx}(t, f) = \sum_{u=-\infty}^{\infty} \sum_{\tau=-\infty}^{\infty} \phi(t-u, \tau) x(u+\tau) x^*(u-\tau) e^{-j4\pi f\tau}, \quad (17.4)$$

which excludes the constant of two and a scaling factor in  $\tau$  for expressional convenience. Similarly, the discrete-form of the cross-term TFD corresponding to (17.2) is given by

$$D_{x_i x_k}(t, f) = \sum_{u=-\infty}^{\infty} \sum_{\tau=-\infty}^{\infty} \phi(t-u, \tau) x_i(u+\tau) x_k^*(u-\tau) e^{-j4\pi f\tau}. \quad (17.5)$$

It is clear from the above equations that the TFD maps one-dimensional (1-D) signals in the time domain into two-dimensional (2-D) signal representations in the t-f domain. The fundamental TFD property of concentrating the input signal energy around its IF, while spreading the noise energy over the entire t-f domain, is crucial in DOA estimation, as it increases the effective SNR. For a single-component LFM signal, pseudo Wigner-Ville distribution (PWVD) can achieve SNR improvement up to the window length [6]. The SNR enhancement is dominantly determined by the window size, but is less sensitive to the type of t-f kernels [32]. When all the t-f points are selected within a 3-dB bandwidth from the peaks, the SNR improvement remains proportional to the window length [33]. Such observations are valid for a general class of signals, provided that the third-order derivative of the waveform phase is negligible or, equivalently, the waveforms can be approximated by an LFM within each sliding window interval.

The properties of a TFD can be characterized by simple constraints on the kernel. Different kernels can be designed and used to generate TFDs with prescribed, desirable properties. WVD is often regarded as the basic or prototype quadratic TFDs, since the other quadratic TFDs can be described as filtered version of the WVD. WVD is known to provide the best t-f resolution for single-component LFM

**Table 17.1** Example of Time-Frequency Kernels

Distribution	Kernel $\phi(t, \tau)$
Wigner-Ville	$\delta(t)$
Pseudo Wigner-Ville	$\delta(t)w(\tau)$
Choi-Williams [34]	$\frac{\sqrt{\pi\sigma}}{ \tau } \exp\left(-\frac{\pi^2\sigma t^2}{\tau^2}\right)$
Zhao-Atlas-Marks [35]	$w(\tau)\text{rect}\left(\frac{t}{2\tau/a}\right)$

signals, but it yields high cross-terms when the frequency law is nonlinear or when a multi-component signal is considered. Various reduced interference kernels have been developed to reduce the cross-term interference. Table 17.1 shows some commonly used kernel functions [7], where  $\delta(t)$  is a Dirac delta function,  $\text{rect}(t)$  is a rectangular window function,  $w(t)$  is an arbitrary window function, and  $\sigma$  and  $a$  are scalars. TFD examples that use these kernels are illustrated in the following two examples. In addition to these kernels that assume fixed parameters, some kernels, such as the adaptive optimal kernel, provide signal-adaptive filtering capability [36].

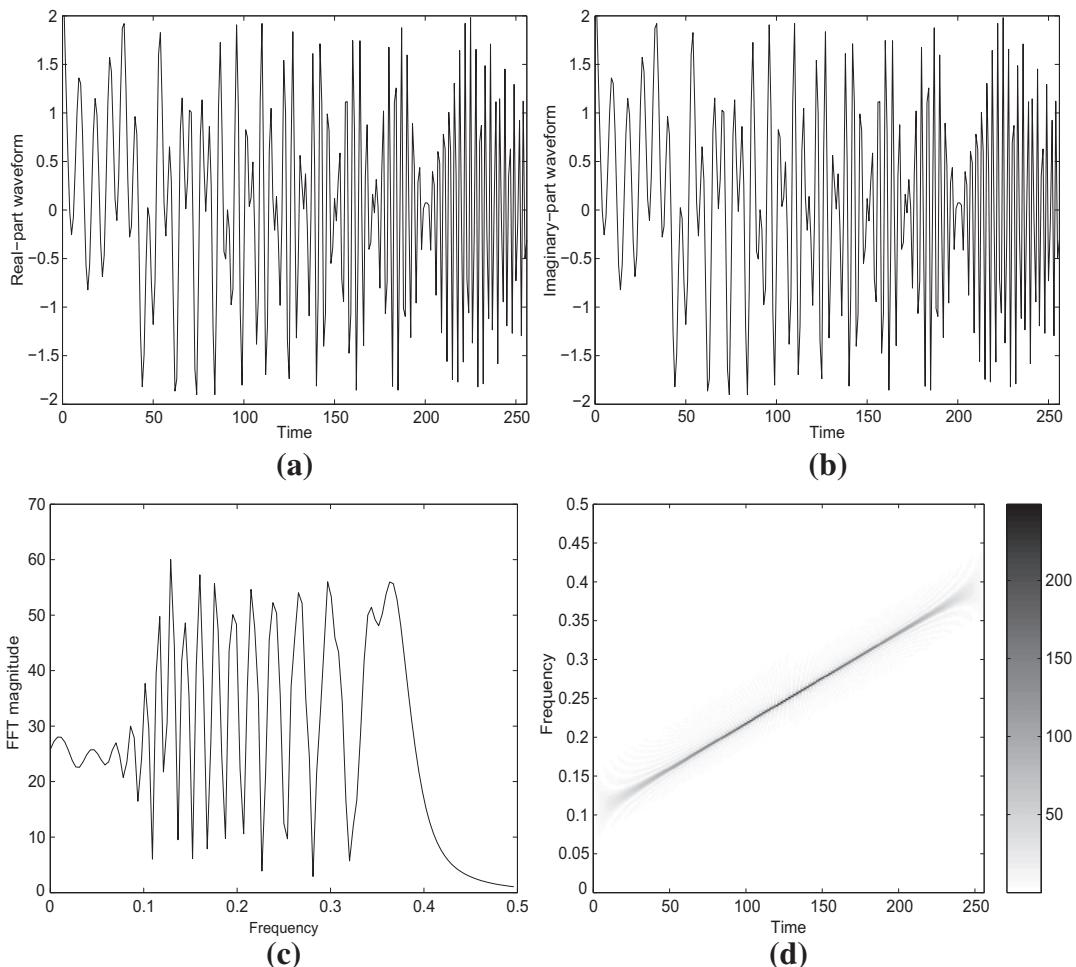
### 3.17.2.2.1 Examples

Figures 17.1a and 17.1b show the real- and imaginary-parts of an analytic LFM signal, expressed as  $x(t) = \exp[j2\pi(0.1t + 0.001176t^2)]$ ,  $t = 0, \dots, 255$ . The start and end frequencies of the LFM signals are, respectively, 0.1 and 0.4. Performing Fourier transform of the signal yields a spectrum spreading over the normalized frequency band [0.1, 0.4], as shown in Figure 17.1c. The WVD of the waveform, shown in Figure 17.1d, depicts high energy concentration of the instantaneous narrowband signal with a linearly time-varying IF signature.

Figure 17.2 shows the t-f representations of two time-limited parallel LFM signals using different kernels. The WVD provides sharp auto-term signatures, whereas the cross-terms are evidently present in the middle of the two auto-term signatures. By using PWVD, the cross-terms in the time-domain are mitigated, whereas the frequency-domain cross-terms remain. Both the Choi-Williams distribution and Zhao-Atlas-Marks distribution provide much reduced cross-term presence. The three non-WVD distributions yield much wider auto-term signatures in the t-f domain.

### 3.17.2.3 Wigner-Radon transform and fractional Fourier transform for LFM signals

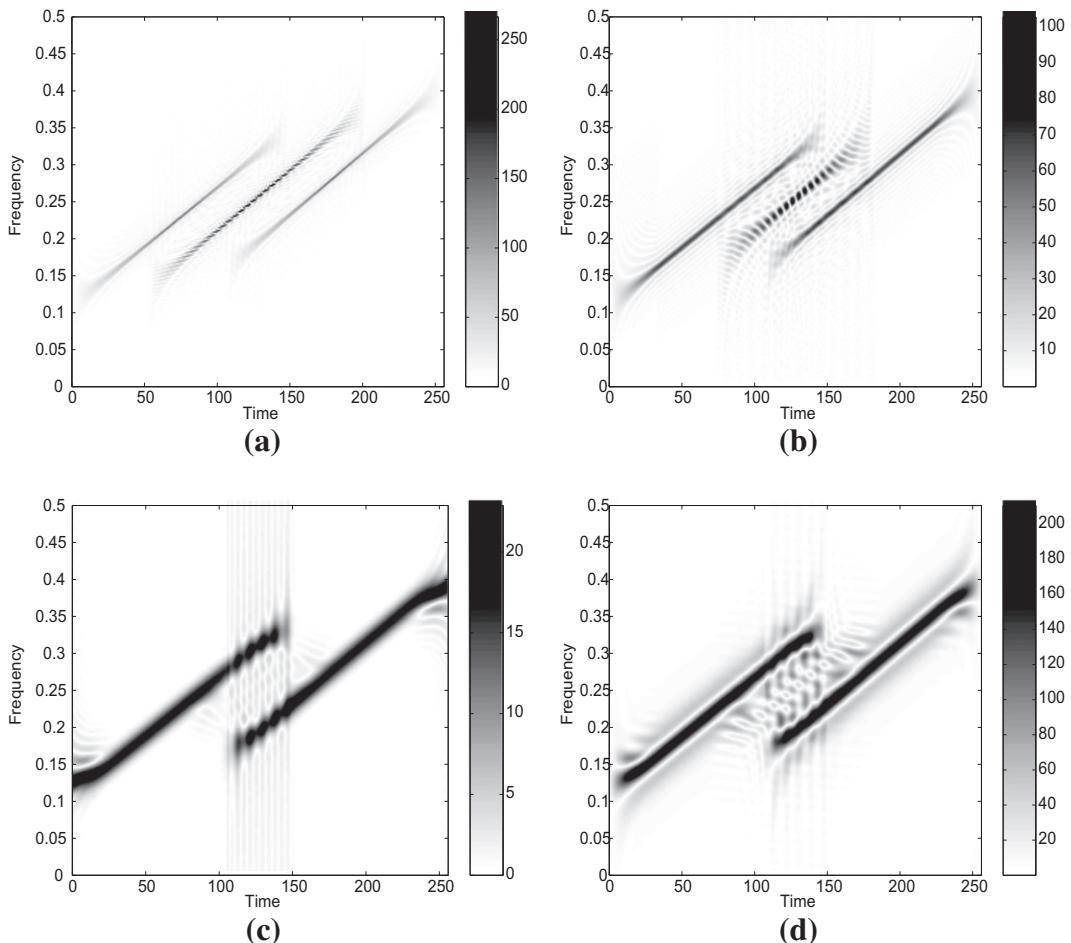
The IFs of LFM (chirp) signals vary linearly with time. The WVD auto-terms of a chirp signal represent themselves in the t-f plane as a straight line with positive values. The time-varying frequency behavior of each LFM component can be described in the t-f domain by the slope (chirp rate) and the initial frequency. This property enables signal parameter estimation and characterizations. The Wigner-Radon transform and fractional Fourier transform (FrFT) are well known techniques that utilize these LFM signal properties. These techniques can be used for DOA estimation of LFM signals. FrFT-based DOA estimation is discussed in Section 3.17.4.4.

**FIGURE 17.1**

Waveform and Wigner-Ville distribution of an LFM signal. (a) Real-part of the waveform. (b) Imaginary-part of waveform. (c) FFT magnitude. (d) Wigner-Ville distribution.

For an LFM signal  $x(t)$ , whose IF is expressed as  $f(t) = f_0 + \beta t$ , integrating the WVD  $D_{xx}(t, f)$  over the t-f line segments yields high peak values, and thus allows estimation of the chirp rate  $\beta$  and the initial frequency  $f_0$ , which uniquely describe the signature of the LFM signal. The following line integration,

$$L(f_0, \beta) = \int D_{xx}(t, f_0 + \beta t) dt \quad (17.6)$$

**FIGURE 17.2**

TFDs of two LFM signals corresponding to different kernels. (a) Wigner-Ville distribution. (b) Pseudo Wigner-Ville distribution. (c) Choi-Williams distribution. (d) Zhao-Atlas-Marks distribution.

is referred to as the Wigner-Radon transform. The above integration is equivalent to dechirping, i.e., signal multiplication with the conjugation of the LFM signal, followed by power spectral calculation of the product [37,38]. For multi-component LFM signals, the Wigner-Radon transform yields peaks in the respective  $(f_0, \beta)$  positions in the time-frequency plane, whereas the cross-terms are effectively mitigated due to their positive-negative oscillating behavior.

The FrFT, on the other hand, is a generalization of the classical Fourier transform. The FrFT was first developed for quantum mechanics [39], but has found broad applications since it was introduced to the signal processing community [28]. Ref. [28] also analyzed the relationship between the FrFT and

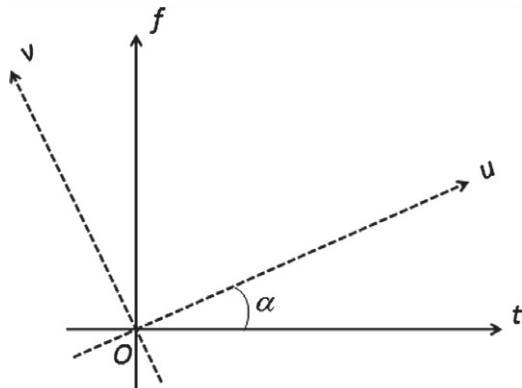
**FIGURE 17.3**

Illustration of rotation in the time-frequency domain through FrFT.

the WVD. FrFT is, in essence, a rotation of the signal representation in the t-f domain (Figure 17.3). The FrFT defines an operator, denoted as  $R^\alpha$ , that performs counterclockwise rotation of the signal by an angle of  $\alpha$  in the t-f plane. In this sense, the classical Fourier transform becomes a specific case by setting  $\alpha = \pi/2$ .

The FrFT is defined by the following transformation kernel:

$$K_\alpha(t, u) = \begin{cases} \sqrt{\frac{1-j \cot \alpha}{2\pi}} e^{j \frac{t^2+u^2}{2} \cot \alpha - jut \csc \alpha}, & \text{if } \alpha \text{ is not a multiple of } \pi, \\ \delta(t-u), & \text{if } \alpha \text{ is a multiple of } 2\pi, \\ \delta(t+u), & \text{if } \alpha + \pi \text{ is a multiple of } 2\pi, \end{cases} \quad (17.7)$$

where  $u$  is the axis of the transformed domain. The FrFT of a signal  $x(t)$  is then expressed as

$$\begin{aligned} X_\alpha(u) = R^\alpha x(t) &= \int_{-\infty}^{\infty} x(t) K_\alpha(t, u) dt \\ &= \begin{cases} \sqrt{\frac{1-j \cot \alpha}{2\pi}} e^{j \frac{u^2}{2} \cot \alpha} \int_{-\infty}^{\infty} x(t) e^{j \frac{t^2}{2} \cot \alpha - jut \csc \alpha} dt, & \text{if } \alpha \text{ is not a multiple of } \pi, \\ x(t), & \text{if } \alpha \text{ is a multiple of } 2\pi, \\ x(-t), & \text{if } \alpha + \pi \text{ is a multiple of } 2\pi. \end{cases} \end{aligned} \quad (17.8)$$

By properly choosing the rotating angle, an LFM signal would be well localized in the transformed domain. Readers interested in this subject can refer to [40] for additional information about FrFT.

### 3.17.2.4 Polynomial phase signals and parameter estimations

A polynomial phase signal (PPS) is an extension of LFM signals which include higher phase orders. The parameter estimation of PPS signals can be used for DOA estimation through signal stationarization, as described in Section 3.17.4.4.

Mathematically, a PPS can be expressed as

$$x(t) = Ae^{j\phi(t)} = Ae^{j \sum_{k=0}^K a_k t^k}, \quad (17.9)$$

where  $K$  is the polynomial order of the phase  $\phi(t)$ ,  $\{a_0, \dots, a_K\}$  are the polynomial coefficients, and  $A$  is the signal amplitude.

Several techniques have been developed to estimate PPS parameters. Commonly used parametric methods estimate the IF through polynomial phase transform, Hough transform, and high-order ambiguity function (HAF) [41–43].

The HAF is defined as the Fourier transform of the high-order instantaneous moment (HIM) which is given, for a signal  $s(t)$ , by the following relation:

$$\text{HIM}_K[s(t); \tau] = \prod_{q=0}^{K-1} [s^{(*q)}(t - q\tau)]^{\binom{K-1}{q}}, \quad (17.10)$$

where  $K$  is the HIM order,  $\tau$  is the lag, and  $(\cdot)^{(*q)}$  is an operator defined as

$$s^{(*q)}(t) = \begin{cases} s(t), & \text{if } q \text{ is even,} \\ s^*(t), & \text{if } q \text{ is odd.} \end{cases} \quad (17.11)$$

The  $N$ th order HIM of a PPS given in (17.9) is reduced to a constant amplitude harmonic [44,45]

$$\text{HIM}_N[s(t); \tau] = A^{2^{N-1}} e^{j(\tilde{\omega}_N t + \tilde{\phi}_N)}, \quad (17.12)$$

where

$$\tilde{\omega}_N = N! \tau^{N-1} a_N, \quad \tilde{\phi}_N = (N-1)! \tau^{N-1} a_{N-1} - 0.5 N! (N-1) \tau^N a_N. \quad (17.13)$$

A natural way to take advantage of this property is to compute the Fourier transform of the  $N$ th-order HIM, which leads to the HAF definition:

$$\text{HAF}_N[s; \omega, \tau] = \int_{-\infty}^{\infty} \text{HIM}_N[s(t); \tau] e^{-j\omega t} dt. \quad (17.14)$$

The  $N$ th order polynomial coefficient can be estimated via

$$\hat{a}_N = \frac{1}{N! \tau^{N-1}} \arg \max_{\omega} \{|\text{HAF}_N(s; \omega, \tau)|\}. \quad (17.15)$$

Using this estimate, the effect of the phase term of the  $N$ th order can be removed:

$$s^{(N-1)}(t) = s(t) e^{-j\hat{a}_N t^N}. \quad (17.16)$$

This process can be repeated to obtain lower order coefficients.

It was pointed out in [45,46] that the classical procedure for polynomial phase modeling based on HAF method is challenged by the noise robustness and the cross-terms presence. To overcome these problems, the multilag HAF (mlHAF) concept, which is a generalization of the HIM, was proposed in [46]. Consider that the  $K$ th-order HIM as the second-order HIM of the  $(K-1)$ th-order HIM,

$$\text{HIM}_K(s(t); \tau) = \text{HIM}_2[\text{HIM}_{K-1}[s(t); \tau]; \tau]. \quad (17.17)$$

In mlHAF based technique, the HIM is replaced by the multilag HIM (mlHIM), which is expressed as

$$\text{mlHIM}_K(s(t); \tau_{K-1}) = \text{mlHIM}_{K-1}[s(t + \tau_{K-1}); \tau_{K-2}] \times \text{mlHIM}_{K-1}^*[s(t - \tau_{K-1}); \tau_{K-2}], \quad (17.18)$$

where  $\tau_N = [\tau_1, \tau_2, \dots, \tau_K]$  is the set of lags. The mlHAF is defined as the Fourier transform of mlHIM in a manner similar to HAF. The performance of HAF-based parameter estimation of multi-component PPS signals are provided in, e.g., [47].

When a nonstationary signal, which is characterized by its IF, is considered over a substantial time period, it may become difficult to use a PPS to model the entire waveform. Rather, it is more practical to partition the waveform into multiple segments, each represents a PPS signal. The segments can be nonoverlapping or partially overlapping. The PPS coefficients are estimated over each segment and then merged together to achieve the global phase behavior of the entire FM signal [44,48–50]. In this case, multiple PPS parameter estimates can be made in each segment, and the phase continuity over neighboring segments can be utilized as additional constraint for the selection of the most likely parameter set [44,48].

### 3.17.3 Spatial time-frequency distribution

In this section, we first introduce the concept of STFD. Analysis of the subspace estimates based on STFD matrices is then provided. The robustness of these estimates compared to those corresponding to the covariance matrices is the main motivation of using the STFD platform for direction finding of nonstationary signals.

#### 3.17.3.1 Spatial time-frequency distribution

Consider  $n$  narrowband nonstationary signals impinging on an array consisting of  $m$  sensors. By *narrowband* signals, we mean that the steering vector does not change with frequency within the signal bandwidth. For simplicity, we assume a 1-D DOA estimation problem (e.g., only the azimuth angle is considered), but the extension to a 2-D problem (i.e., both the azimuth and elevation angles are considered) is straightforward. The  $m \times 1$  received data vector  $\mathbf{x}(t)$  and the  $n \times 1$  source signal vector  $\mathbf{d}(t)$  are related by

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{d}(t) + \mathbf{n}(t), \quad (17.19)$$

where  $\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_n)]$  is the  $m \times n$  mixing matrix that holds the steering vectors of the  $n$  signals,  $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_n]$ , and  $\mathbf{a}(\theta_q)$  is the steering vector for the  $q$ th source, who signal  $d_q(t)$  arrives from direction  $\theta_q$ . Each element of  $\mathbf{d}(t) = [d_1(t), d_2(t), \dots, d_n(t)]^T$  is assumed to be

a mono-component signal. Due to the signal mixing occurring at each sensor, the elements of  $\mathbf{x}(t)$  become multi-component signals.  $\mathbf{n}(t)$  is an  $m \times 1$  additive noise vector that consists of independent and identically distributed (i.i.d.) zero-mean, white and complex Gaussian distributed processes with variance  $\sigma_n^2$ . The noise elements are assumed to be independent of the signals, which are assumed to be deterministic.

The STFD matrix of vector  $\mathbf{x}(t)$  is expressed as [4]

$$\mathbf{D}_{\mathbf{xx}}(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(t-u, \tau) \mathbf{x} \left( u + \frac{\tau}{2} \right) \mathbf{x}^H \left( u - \frac{\tau}{2} \right) e^{-j2\pi f\tau} du d\tau, \quad (17.20)$$

where the  $(i, k)$ th element of  $\mathbf{D}_{\mathbf{xx}}(t, f)$  is given in (17.2) for  $i, k = 1, 2, \dots, m$ . The noise-free STFD matrix is obtained by substituting (17.19) into (17.20), resulting in

$$\mathbf{D}_{\mathbf{xx}}(t, f) = \mathbf{A}(\theta) \mathbf{D}_{\mathbf{dd}}(t, f) \mathbf{A}^H(\theta), \quad (17.21)$$

where  $\mathbf{D}_{\mathbf{dd}}(t, f)$  is the TFD matrix of  $\mathbf{d}(t)$  which consists of auto-source TFDs as its diagonal elements and cross-source TFDs as its off-diagonal elements. With the presence of noise, the expected value of  $\mathbf{D}_{\mathbf{xx}}(t, f)$  becomes

$$\mathbb{E}[\mathbf{D}_{\mathbf{xx}}(t, f)] = \mathbf{A}(\theta) \mathbf{D}_{\mathbf{dd}}(t, f) \mathbf{A}^H(\theta) + \sigma_n^2 \mathbf{I}_m. \quad (17.22)$$

Equation (17.22) relates the STFD matrix to the source TFD matrix in a manner similar to the formula that is commonly used in narrowband array processing problems, relating the source covariance matrix to the sensor spatial covariance matrix. It is clear, therefore, that the two subspaces spanned by the principle eigenvectors of  $\mathbf{D}_{\mathbf{xx}}(t, f)$  and the columns of  $\mathbf{A}(\theta)$  are identical. As discussed below, the construction of the STFD matrix from the t-f points of highly localized signal energy allows the corresponding signal and noise subspace estimates to become more robust to noise than their counterparts obtained using the data covariance matrix [4,6]. Further, source elimination, rendered through the selection of specific t-f regions, improves DOA estimations [6].

### 3.17.3.2 SNR enhancement

To provide insights into the properties of STFDs, we consider the case of frequency modulated (FM) signals and the simplest form of TFD, namely, the pseudo Wigner-Ville distribution (PWVD) [6]. The consideration of FM signals is motivated by the fact that these signals are uniquely characterized by their IFs, and therefore, they have clear t-f signatures that can be utilized by the STFD approach. Also, FM the signals have constant amplitudes. The FM signals can be modeled as

$$\mathbf{d}(t) = [d_1(t), \dots, d_n(t)]^T = \left[ D_1 e^{j\psi_1(t)}, \dots, D_n e^{j\psi_n(t)} \right]^T, \quad (17.23)$$

where  $D_i$  and  $\psi_i(t)$  are the fixed amplitude and time-varying phase of  $i$ th source signal. For each sampling time  $t$ ,  $d_i(t)$  has an IF of  $f_i(t) = d\psi_i(t)/(2\pi dt)$ . For the simplicity of the analysis, we further assume that the third-order derivative of the phase is negligible over the window length  $L$ .

The discrete form of PWVD of a signal  $x(t)$ , using a rectangular window of odd length  $L$ , is given by

$$D_{xx}(t, f) = \sum_{\tau=-(L-1)/2}^{(L-1)/2} x(t+\tau)x^*(t-\tau)e^{-j4\pi f\tau}. \quad (17.24)$$

Similarly, the spatial pseudo Wigner-Ville distribution (SPWVD) matrix is obtained as

$$\mathbf{D}_{\mathbf{xx}}(t, f) = \sum_{\tau=-(L-1)/2}^{(L-1)/2} \mathbf{x}(t + \tau) \mathbf{x}^H(t - \tau) e^{-j4\pi f\tau}. \quad (17.25)$$

The  $i$ th diagonal element of PWVD matrix  $\mathbf{D}_{\mathbf{dd}}(t, f)$  is given by

$$D_{d_i d_i}(t, f) = \sum_{\tau=-(L-1)/2}^{(L-1)/2} D_i^2 e^{j[\psi_i(t+\tau) - \psi_i(t-\tau)] - j4\pi f\tau}. \quad (17.26)$$

Assume that the third-order derivative of the phase is negligible over the window length  $L$ , then along the true t-f points of the  $i$ th signal,  $f_i(t) = d\psi_i(t)/(2\pi dt)$ , and  $\psi_i(t + \tau) - \psi_i(t - \tau) - 4\pi f_i(t)\tau = 0$ . Accordingly, for  $(L - 1)/2 \leq t \leq N - (L - 1)/2$ ,

$$D_{d_i d_i}(t, f_i(t)) = \sum_{\tau=-(L-1)/2}^{(L-1)/2} D_i^2 = LD_i^2. \quad (17.27)$$

Similarly, the noise SPWVD matrix  $\mathbf{D}_{\mathbf{nn}}(t, f)$  is

$$\mathbf{D}_{\mathbf{nn}}(t, f) = \sum_{\tau=-(L-1)/2}^{(L-1)/2} \mathbf{n}(t + \tau) \mathbf{n}^H(t - \tau) e^{-j4\pi f\tau}, \quad (17.28)$$

whose statistical expectation is  $E[\mathbf{D}_{\mathbf{nn}}(t, f)] = \sigma_n^2 \mathbf{I}_m$ . Therefore, when we select the t-f points along the t-f signature or the IF of the  $i$ th FM signal, the SNR in the STFD matrix  $E[\mathbf{D}_{\mathbf{xx}}(t, f)]$  becomes  $LD_i^2/\sigma_n^2$ , which has an improved factor  $L$  over the covariance matrix  $\mathbf{R}_{\mathbf{xx}} = E[\mathbf{x}(t)\mathbf{x}^H(t)]$ .

The PWVD of each FM source has a constant value over the observation period, providing that we leave out the rising and falling power distributions at both ends of the data record. For convenience, we select those  $N' = N - L + 1$  t-f points of constant distribution value for each source signal. In the case where the STFD matrices  $n_o$  sources, i.e., a total of  $n_o N'$  t-f points, the result is given by

$$\widehat{\mathbf{D}} = \frac{1}{n_o N'} \sum_{q=1}^{n_o} \sum_{i=1}^{N'} \mathbf{D}_{\mathbf{xx}}(t_i, f_{q,i}(t_i)), \quad (17.29)$$

where  $f_{q,i}(t_i)$  is the IF of the  $q$ th signal at the  $i$ th time sample. The expectation of the averaged STFD matrix is

$$\mathbf{D} = \frac{1}{n_o} \sum_{q=1}^{n_o} \left[ L D_q^2 \mathbf{a}_q \mathbf{a}_q^H + \sigma_n^2 \mathbf{I} \right] = \frac{L}{n_o} \mathbf{A}^o \mathbf{R}_{\mathbf{dd}}^o (\mathbf{A}^o)^H + \sigma_n^2 \mathbf{I}, \quad (17.30)$$

where  $\mathbf{R}_{\mathbf{dd}}^o = \text{Diag}[D_i^2, i = 1, 2, \dots, n_o]$  and  $\mathbf{A}^o = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{n_o}]$  represent the signal correlation matrix and the mixing matrix formulated by considering  $n_o$  signals out of the total number of  $n$  signal arrivals, respectively.

It is clear from (17.30) that the SNR improvement  $G = L/n_o$  (we assume  $L > n_o$ ) is inversely proportional to the number of sources contributing to matrix  $\mathbf{D}$ . Therefore, from the SNR perspective, it is best to set  $n_o = 1$ , i.e., to select the sets of  $N'$  t-f points that belong to individual signals one set at a time, and then separately evaluate the respective STFD matrices.

This procedure is made possible by the fact that STFD-based array processing is, in essence, a discriminatory technique in the sense that it does not require simultaneous localization and extraction of all unknown signals received by the array. Array processing can be performed using STFDs of a subclass of the impinging signals with specific t-f signatures. In this respect, the t-f based direction finding techniques have implicit spatial filtering, removing the undesired signals from consideration. It is also important to note that with the ability to construct the STFD matrix from one or few signal arrivals, the well known  $m > n$  condition on source localization using arrays can be relaxed to  $m > n_o$ , i.e., we can perform direction finding or source separation with the number of array sensors smaller than the number of impinging signals. Further, from the angular resolution perspective, closely spaced sources with different t-f signatures can be resolved by constructing two separate STFDs, each corresponding to one source, and then proceed with subspace decomposition for each STFD matrix, followed by an appropriate source localization method (MUSIC, for example). The drawback using different STFD matrices separately is of course the need for repeated computations. Relevant work for noise analysis and SNR enhancement in the t-f domain can be found in [51–54].

### 3.17.3.3 Subspace analysis

Analysis of the eigendecomposition of the STFD matrix is closely related to the analysis of subspace decomposition of the covariance matrix [55]. Before elaborating on this relationship, we present the case of FM signals using the conventional covariance matrix approach.

In Eq. (17.19), it is assumed that the number of sensors is greater than the number of sources, i.e.,  $m > n$ . Further, matrix  $\mathbf{A}$  is full column rank. We further assume that the correlation matrix  $\mathbf{R}_{\mathbf{xx}} = \mathbb{E}[\mathbf{x}(t)\mathbf{x}^H(t)]$  is nonsingular, and the observation period consists of  $N$  snapshots with  $N > m$ . Under the above assumptions, the correlation matrix is given by

$$\mathbf{R}_{\mathbf{xx}} = \mathbb{E}[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{A}\mathbf{R}_{\mathbf{dd}}\mathbf{A}^H + \sigma_n^2\mathbf{I}_m, \quad (17.31)$$

where  $\mathbf{R}_{\mathbf{dd}} = \mathbb{E}[\mathbf{d}(t)\mathbf{d}^H(t)]$  is the source correlation matrix.

Let  $\lambda_1 > \lambda_2 > \dots > \lambda_n > \lambda_{n+1} = \lambda_{n+2} = \dots = \lambda_m = \sigma_n^2$  denote the eigenvalues of  $\mathbf{R}_{\mathbf{xx}}$ . It is assumed that  $\lambda_i$ ,  $i = 1, \dots, n$ , are distinct. The unit-norm eigenvectors associated with  $\lambda_1, \dots, \lambda_n$  constitute the columns of matrix  $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_n]$  that spans the signal subspace, and those corresponding to  $\lambda_{n+1}, \dots, \lambda_m$  make up matrix  $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_{m-n}]$  that spans the noise subspace. Since the columns of  $\mathbf{A}$  and  $\mathbf{S}$  span the same subspace, then  $\mathbf{A}^H\mathbf{G} = \mathbf{0}$ .

In practice,  $\mathbf{R}_{\mathbf{xx}}$  is unknown, and therefore should be estimated from the available data samples (snapshots)  $\mathbf{x}(i)$ ,  $i = 1, 2, \dots, N$ . The estimated correlation matrix is given by  $\widehat{\mathbf{R}}_{\mathbf{xx}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}(i)\mathbf{x}^H(i)$ . Let  $\{\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_n, \hat{\mathbf{g}}_1, \dots, \hat{\mathbf{g}}_{m-n}\}$  denote the unit-norm eigenvectors of  $\widehat{\mathbf{R}}_{\mathbf{xx}}$ , arranged in the descending order of the associated eigenvalues, and let  $\widehat{\mathbf{S}} = [\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_n]$  and  $\widehat{\mathbf{G}} = [\hat{\mathbf{g}}_1, \dots, \hat{\mathbf{g}}_{m-n}]$ .

We assume that the transmitted signals propagate in a stationary environment and are mutually uncorrelated over the observation period  $1 \leq t \leq N$ , i.e.,  $\frac{1}{N} \sum_{k=1}^N d_l(k)d_l^*(k) = 0$ , for  $i \neq l, i$ ,

$l = 1, \dots, n$ . In this case, the signal correlation matrix is

$$\mathbf{R}_{\mathbf{d}\mathbf{d}} = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbf{d}(t) \mathbf{d}^H(t) = \text{Diag} \left[ D_1^2, \dots, D_n^2 \right]. \quad (17.32)$$

**Lemma 1 [6].** *For uncorrelated FM signals with additive white Gaussian noise, the orthogonal projections of  $\{\hat{\mathbf{g}}_i\}$  onto the column space of  $\mathbf{S}$  are asymptotically (for large  $N$ ) jointly Gaussian distributed with zero means and covariance matrices given by*

$$E \left[ (\mathbf{S} \mathbf{S}^H \hat{\mathbf{g}}_i) (\mathbf{S} \mathbf{S}^H \hat{\mathbf{g}}_j)^H \right] = \frac{\sigma_n^2}{N} \left[ \sum_{k=1}^n \frac{\lambda_k}{(\sigma_n^2 - \lambda_k)^2} \mathbf{s}_k \mathbf{s}_k^H \right] \delta_{i,j} \triangleq \frac{1}{N} \mathbf{U} \delta_{i,j}, \quad (17.33)$$

$$E \left[ (\mathbf{S} \mathbf{S}^H \hat{\mathbf{g}}_i) (\mathbf{S} \mathbf{S}^H \hat{\mathbf{g}}_j)^T \right] = 0 \quad \text{for all } i, j. \quad (17.34)$$

The following Lemma provides the relationship between the eigendecompositions of the STFD matrices and the data covariance matrices used in conventional array processing.

**Lemma 2 [6].** *Let  $\lambda_1^o > \lambda_2^o > \dots > \lambda_{n_o}^o > \lambda_{n_o+1}^o = \lambda_{n_o+2}^o = \dots = \lambda_m^o = \sigma_n^2$  denote the eigenvalues of  $\mathbf{R}_{\mathbf{x}\mathbf{x}}^o = \mathbf{A}^o \mathbf{R}_{\mathbf{d}\mathbf{d}}^o (\mathbf{A}^o)^H + \sigma_n^2 \mathbf{I}_m$ , which is defined from a data record of a mixture of the  $n_o$  selected FM signals. Denote the unit-norm eigenvectors associated with  $\lambda_1^o, \dots, \lambda_{n_o}^o$  by the columns of  $\mathbf{S}^o = [\mathbf{s}_1^o, \dots, \mathbf{s}_{n_o}^o]$ , and those corresponding to  $\lambda_{n_o+1}^o, \dots, \lambda_m^o$  by the columns of  $\mathbf{G}^o = [\mathbf{g}_1^o, \dots, \mathbf{g}_{m-n_o}^o]$ . We also denote  $\lambda_1^{tf} > \lambda_2^{tf} > \dots > \lambda_{n_o}^{tf} > \lambda_{n_o+1}^{tf} = \lambda_{n_o+2}^{tf} = \dots = \lambda_m^{tf} = (\sigma_n^{tf})^2$  as the eigenvalues of  $\mathbf{D}$  defined in (17.30). The superscript  $tf$  denotes that the associated term is derived from the STFD matrix  $\mathbf{D}$ . The unit-norm eigenvectors associated with  $\lambda_1^{tf}, \dots, \lambda_{n_o}^{tf}$  are represented by the columns of  $\mathbf{S}^{tf} = [\mathbf{s}_1^{tf}, \dots, \mathbf{s}_{n_o}^{tf}]$ , and those corresponding to  $\lambda_{n_o+1}^{tf}, \dots, \lambda_m^{tf}$  are represented by the columns of  $\mathbf{G}^{tf} = [\mathbf{g}_1^{tf}, \dots, \mathbf{g}_{m-n_o}^{tf}]$ . Then,*

- a. The signal and noise subspaces of  $\mathbf{S}^{tf}$  and  $\mathbf{G}^{tf}$  are the same as  $\mathbf{S}^o$  and  $\mathbf{G}^o$ , respectively.
- b. The eigenvalues have the following relationship:

$$\lambda_i^{tf} = \begin{cases} \frac{L}{n_o} (\lambda_i^o - \sigma_n^2) + \sigma_n^2 = \frac{L}{n_o} \lambda_i^o + \left(1 - \frac{L}{n_o}\right) \sigma_n^2, & i \leq n_o, \\ \left(\sigma_n^{tf}\right)^2 = \sigma_n^2, & n_o < i \leq m. \end{cases} \quad (17.35)$$

An important conclusion from Lemma 2 is that, the largest  $n_o$  eigenvalues are amplified using STFD analysis. This improves detection of the number of the impinging signals on the array, as it widens the separation between dominant and noise-level eigenvalues. Determination of the number of signals is key to establishing the proper signal and noise subspaces, and subsequently plays a fundamental role in subspace-based applications. When the input SNR is low, or the signals are closely spaced, the number of signals may often be underdetermined. When the STFD is applied, the SNR threshold level and/or angle separation necessary for the correct determination of the number of signals are greatly reduced.

Next we consider the signal and noise subspace estimates from a finite number of data samples. We form the STFD matrix based on the true  $(t, f)$  points along the IF of the  $n_o$  FM signals.

**Lemma 3 [15,6].** *If the third-order derivative of the phase of the FM signals is negligible over the time-period  $[t - L + 1, t + L - 1]$ , then the orthogonal projections of  $\{\hat{\mathbf{g}}_i^{tf}\}$  onto the column space of  $\mathbf{S}^{tf}$  are asymptotically (for  $N \gg L$ ) jointly Gaussian distributed with zero means and covariance matrices given by*

$$\begin{aligned} E \left( \mathbf{S}^{tf} (\mathbf{S}^{tf})^H \hat{\mathbf{g}}_i^{tf} \right) \left( \mathbf{S}^{tf} (\mathbf{S}^{tf})^H \hat{\mathbf{g}}_j^{tf} \right)^H &= \frac{\sigma_n^2 L}{n_o N'} \left[ \sum_{k=1}^{n_o} \frac{\lambda_k^{tf}}{(\sigma_n^2 - \lambda_k^{tf})^2} \mathbf{s}_k^{tf} (\mathbf{s}_k^{tf})^H \right] \delta_{i,j} \\ &= \frac{\sigma_n^2}{N'} \left[ \sum_{k=1}^{n_o} \frac{(\lambda_k^o - \sigma_n^2) + \frac{n_o \sigma_n^2}{L}}{(\sigma_n^2 - \lambda_k^o)^2} \mathbf{s}_k^o (\mathbf{s}_k^o)^H \right] \delta_{i,j} \\ &\triangleq \frac{1}{N'} \mathbf{U}^{tf} \delta_{i,j}, \end{aligned} \quad (17.36)$$

$$E \left( \mathbf{S}^{tf} (\mathbf{S}^{tf})^H \hat{\mathbf{g}}_i^{tf} \right) \left( \mathbf{S}^{tf} (\mathbf{S}^{tf})^H \hat{\mathbf{g}}_j^{tf} \right)^T = \mathbf{0} \quad \text{for all } i, j. \quad (17.37)$$

From (17.36) and (17.37), two important observations are in order. First, if the signals are both localizable and separable in the t-f domain, then the reduction of the number of signals from  $n$  to  $n_o$  greatly reduces the estimation error, specifically when the signals are closely spaced. The second observation relates to SNR enhancements. The above equations show that error reductions using STFDs are more pronounced for the cases of low SNR and/or closely spaced signals. It is clear from (17.36) and (17.37) that, when  $\lambda_k^o \gg \sigma_n^2$  for all  $k = 1, 2, \dots, n_o$ , the results are almost independent of  $L$  (suppose  $N \gg L$  so that  $N' = N - L + 1 \simeq N$ ), and therefore there would be no obvious improvement in using the STFD over conventional array processing. On the other hand, when some of the eigenvalues are close to  $\sigma_n^2$  ( $\lambda_k^o \simeq \sigma_n^2$ , for some  $k = 1, 2, \dots, n_o$ ), which is the case of weak or closely spaced signals, all the results of above three equations are reduced by a factor of up to  $G = L/n_o$ , respectively. This factor represents, in essence, the gain achieved from using STFD processing.

### 3.17.4 DOA estimation techniques

In this section, we first introduce the STFD-based DOA estimation techniques under the narrowband signal model. T-f MUSIC and t-f maximum likelihood (ML) are used as examples. These techniques demonstrate the advantages of the STFD framework, as described in the previous section. The t-f MUSIC is relatively simple, whereas t-f ML is more computationally demanding, but allows high-resolution DOA estimation of coherent signals. We address the effect of t-f cross-terms on direction finding performance. We then introduce the DOA estimation techniques based on parametric models of nonstationary signals. Depending on the characteristics of the nonstationary signals, different techniques can be used. Fractional transform is discussed for LFM signals, whereas techniques based on signal stationarization allows high-resolution DOA estimations of higher-order polynomial phase signals. DOA estimation based on spatial joint-variable domain distributions, such as the spatial ambiguity function (SAF), is also introduced. Finally, STFD-based DOA estimation of wideband signals is discussed.

### 3.17.4.1 Time-frequency MUSIC

Without loss of generality, we consider 1-D direction finding where the DOAs are described by  $\theta$ . First, recall that the DOAs are estimated in the MUSIC technique by determining the  $n$  values of  $\theta$  for which the following spatial spectrum is maximized [56],

$$f_{\text{MU}}(\theta) = \left[ \mathbf{a}^H(\theta) \widehat{\mathbf{G}} \widehat{\mathbf{G}}^H \mathbf{a}(\theta) \right]^{-1} = \left[ \mathbf{a}^H(\theta) (\mathbf{I} - \widehat{\mathbf{S}} \widehat{\mathbf{S}}^H) \mathbf{a}(\theta) \right]^{-1}, \quad (17.38)$$

where  $\mathbf{a}(\theta)$  is the steering vector corresponds to  $\theta$ . The variance of those estimates in the MUSIC technique, assuming white noise processes, is given by Stoica and Nehorai [55]

$$\mathbb{E} (\hat{\omega}_i - \omega_i)^2 = \frac{1}{2N} \frac{\mathbf{a}^H(\theta_i) \mathbf{U} \mathbf{a}(\theta_i)}{h(\theta_i)}, \quad (17.39)$$

where  $\omega_i = (2\pi d/\lambda) \sin \theta_i$  is the spatial frequency associated with DOA  $\theta_i$ ,  $d$  is the interelement spacing, and  $\lambda$  is the wavelength.  $\hat{\omega}_i$  is the estimate of  $\omega$  obtained from the MUSIC. Moreover,  $\mathbf{U}$  is defined in (17.33), and

$$h(\theta_i) = \mathbf{d}^H(\theta_i) \mathbf{G} \mathbf{G}^H \mathbf{d}(\theta_i), \quad \text{with } \mathbf{d}(\theta_i) = d\mathbf{a}(\theta_i)/d\omega. \quad (17.40)$$

Similarly, for t-f MUSIC with  $n_o$  signals selected, the DOAs are determined by locating the  $n_o$  peaks of the spatial spectrum defined from the  $n_o$  signals' t-f regions,

$$f_{\text{MU}}^{tf}(\theta) = \left[ \mathbf{a}^H(\theta) \widehat{\mathbf{G}}^{tf} (\widehat{\mathbf{G}}^{tf})^H \mathbf{a}(\theta) \right]^{-1} = \left[ \mathbf{a}^H(\theta) (\mathbf{I} - \widehat{\mathbf{S}}^{tf} (\widehat{\mathbf{S}}^{tf})^H) \mathbf{a}(\theta) \right]^{-1}. \quad (17.41)$$

$\widehat{\mathbf{G}}^{tf}$  and  $\widehat{\mathbf{S}}^{tf}$  can be obtained by using either joint block diagonalization (JBD) [5] or t-f averaging. When the t-f averaging is used, the variance of the DOA estimates based on t-f MUSIC is obtained, from the results of Lemmas 2 and 3, as [6],

$$\mathbb{E} (\hat{\omega}_i^{tf} - \omega_i)^2 = \frac{1}{2N'} \frac{\mathbf{a}^H(\theta_i) \mathbf{U}^{tf} \mathbf{a}(\theta_i)}{h^{tf}(\theta_i)}, \quad (17.42)$$

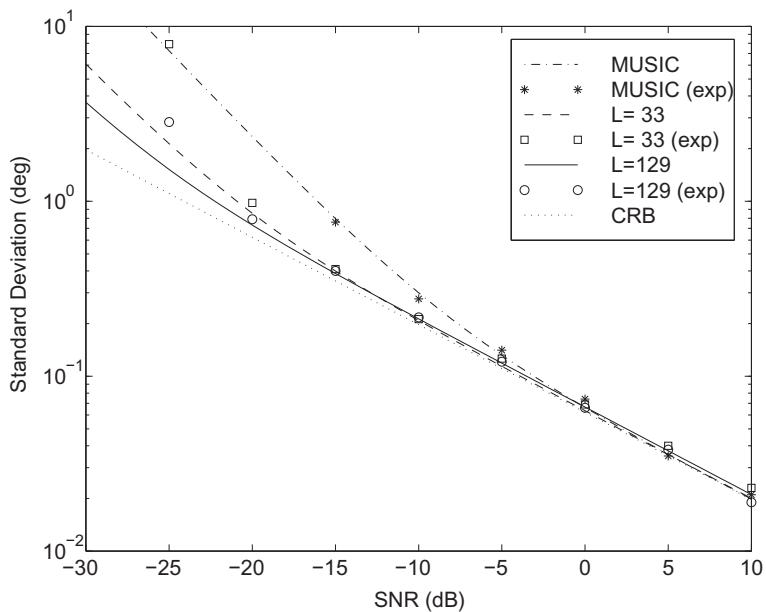
where  $\hat{\omega}_i^{tf}$  is the estimate of  $\omega_i$ ,  $\mathbf{U}^{tf}$  is defined in (17.36), and

$$h^{tf}(\theta_i) = \mathbf{d}^H(\theta_i) \mathbf{G}^{tf} (\mathbf{G}^{tf})^H \mathbf{d}(\theta_i). \quad (17.43)$$

Note that  $h^{tf}(\theta) = h(\theta_i)$  if  $n_o = n$ .

#### 3.17.4.1.1 Examples

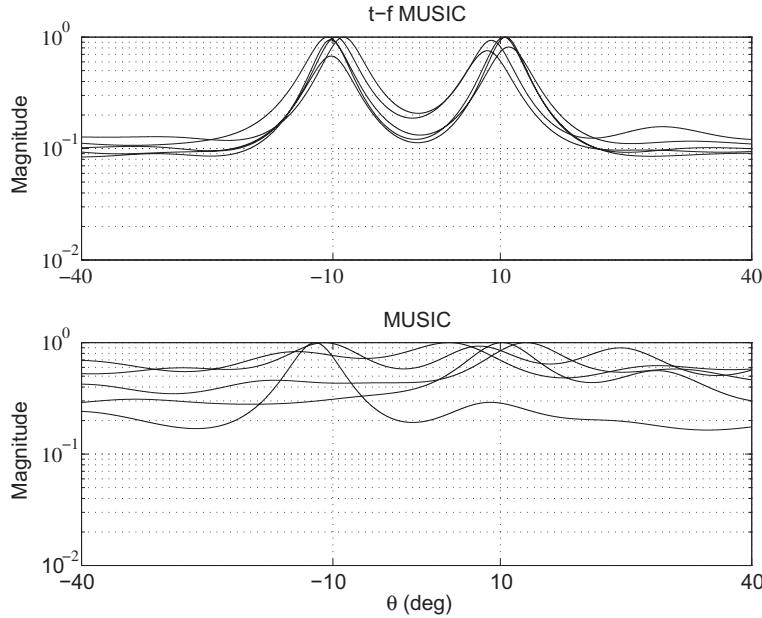
Consider a uniform linear array of eight sensors spaced by half a wavelength, and an observation period of 1024 samples. Two chirp signals emitted from two sources positioned at angles  $\theta_1$  and  $\theta_2$ . The start and end frequencies of the signal source at  $\theta_1$  are  $\omega_{s1} = 0$  and  $\omega_{e1} = \pi$ , while the corresponding two frequencies for the other source at  $\theta_2$  are  $\omega_{s2} = \pi$  and  $\omega_{e2} = 0$ , respectively.

**FIGURE 17.4**

Variance of DOA estimation versus input SNR.

Figure 17.4 displays the variance of the estimated DOA  $\hat{\theta}_1$  versus SNR for the case  $(\theta_1, \theta_2) = (-10^\circ, 10^\circ)$ . The curves in this figure show the theoretical and simulation results of the conventional MUSIC and t-f MUSIC (for  $L = 33$  and 129). The Cramer-Rao bound (CRB) is also shown in Figure 17.4 for comparison. Both signals were selected when performing t-f MUSIC ( $n_o = n = 2$ ). Simulation results were averaged over 100 independent Monte-Carlo runs. The advantages of t-f MUSIC in low SNR cases are evident from this figure. The simulation results deviate from the theoretical results for low SNR. This is due to considering only the lowest coefficient order of the perturbation expansion in deriving the theoretical results [6]. Figure 17.5 shows estimated spatial spectra at SNR = -20 dB based on t-f MUSIC ( $L = 129$ ) and the conventional MUSIC. The t-f MUSIC spectral peaks are clearly resolved.

Figure 17.6 shows examples of the estimated spatial spectrum based on t-f MUSIC ( $L = 129$ ) and the conventional MUSIC where the angle separation is small ( $\theta_1 = -2.5^\circ, \theta_2 = 2.5^\circ$ ). The input SNR is -5 dB. Two t-f MUSIC algorithms are performed using two sets of t-f points, each set belongs to the t-f signature of one source ( $n_o = 1$ ). It is evident that the two signals cannot be resolved when the conventional MUSIC is applied, whereas by utilizing the signals' distinct t-f signatures and applying t-f MUSIC separately for each signal, the two signals become clearly separated and a reasonable DOA estimation is achieved. It is noted that there is a small bias in the estimates of t-f MUSIC due to the imperfect separation of the two signals in the t-f domain.

**FIGURE 17.5**

Estimated spatial spectra of MUSIC and t-f MUSIC.

### 3.17.4.2 Time-frequency maximum likelihood method

In this section, we introduce the time-frequency maximum likelihood (t-f ML) method that can deal with coherent nonstationary sources [15, 57]. For conventional ML methods, the joint density function of the sampled data vectors  $\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)$ , is given by Ziskind [58]

$$f(\mathbf{x}(1), \dots, \mathbf{x}(N)) = \prod_{i=1}^N \frac{1}{\pi^m \det[\sigma_n^2 \mathbf{I}]} \exp \left( -\frac{1}{\sigma_n^2} [\mathbf{x}(i) - \mathbf{Ad}(i)]^H [\mathbf{x}(i) - \mathbf{Ad}(i)] \right). \quad (17.44)$$

It follows from (17.44) that the log-likelihood function of the observations  $\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)$ , is given by

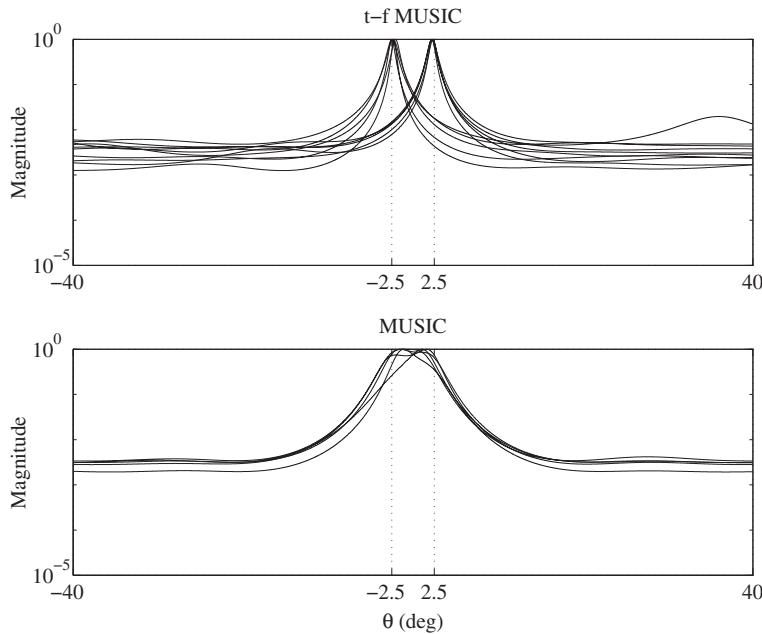
$$L = -mN \ln \sigma_n^2 - \frac{1}{\sigma_n^2} \sum_{i=1}^N [\mathbf{x}(i) - \mathbf{Ad}(i)]^H [\mathbf{x}(i) - \mathbf{Ad}(i)]. \quad (17.45)$$

To carry out this minimization, we fix  $\mathbf{A}$  and minimize (17.45) with respect to  $\mathbf{d}$ . This yields the well-known solution

$$\hat{\mathbf{d}}(i) = [\mathbf{A}^H \mathbf{A}]^{-1} \mathbf{A}^H \mathbf{x}(i). \quad (17.46)$$

We can obtain the concentrated likelihood function as [58]

$$F_{\text{ML}}(\theta) = \text{tr} \left\{ \left[ \mathbf{I}_m - \hat{\mathbf{A}} (\hat{\mathbf{A}}^H \hat{\mathbf{A}})^{-1} \hat{\mathbf{A}}^H \right] \hat{\mathbf{R}}_{\mathbf{xx}} \right\}. \quad (17.47)$$

**FIGURE 17.6**

Estimated spatial spectra of MUSIC and t-f MUSIC for closely spaced signals.

The ML estimate of  $\theta$  is obtained as the minimizer of (17.47). Let  $\omega_i$  and  $\hat{\omega}_i$ , respectively, denote the spatial frequency and its ML estimate associated with  $\theta_i$ , then the estimation error ( $\hat{\omega}_i - \omega_i$ ) are asymptotically (for large  $N$ ) jointly Gaussian distributed with zero means and the covariance matrix [55]

$$E[(\hat{\omega}_i - \omega_i)^2] = \frac{1}{2N} \left[ \operatorname{Re}(\mathbf{H} \odot \mathbf{R}_{dd}^T) \right]^{-1} \cdot \operatorname{Re} \left[ \mathbf{H} \odot \left( \mathbf{R}_{dd} \mathbf{A}^H \mathbf{U} \mathbf{A} \mathbf{R}_{dd} \right)^T \right] \left[ \operatorname{Re}(\mathbf{H} \odot \mathbf{R}_{dd}^T) \right]^{-1}, \quad (17.48)$$

where  $\mathbf{U}$  is defined in (17.33). Moreover,

$$\mathbf{H} = \mathbf{C}^H \left[ \mathbf{I} - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \right] \mathbf{C}, \quad \text{with } \mathbf{C} = d\mathbf{A}/d\omega. \quad (17.49)$$

Next we consider the t-f ML method. As we discussed in the previous section, we select  $n_o \leq n$  signals in the t-f domain. The concentrated likelihood function defined from the STFD matrix is similar to (17.47) and is obtained by replacing  $\widehat{\mathbf{R}}_{xx}$  by  $\widehat{\mathbf{D}}$ ,

$$F_{ML}^{tf}(\boldsymbol{\theta}) = \operatorname{tr} \left\{ \left[ \mathbf{I} - \widehat{\mathbf{A}}^o \left( (\widehat{\mathbf{A}}^o)^H \widehat{\mathbf{A}}^o \right)^{-1} (\widehat{\mathbf{A}}^o)^H \right] \widehat{\mathbf{D}} \right\}. \quad (17.50)$$

Therefore, the estimation error  $(\hat{\omega}_i^{tf} - \omega_i)$  associated with the t-f ML method are asymptotically (for  $N \gg L$ ) jointly Gaussian distributed with zero means and the covariance matrix [15]

$$\begin{aligned} E\left[\left(\hat{\omega}_i^{tf} - \omega_i\right)^2\right] &= \frac{\sigma_n^2}{2N'} \left[ \operatorname{Re}\left(\mathbf{H}^o \odot \mathbf{D}_{dd}^T\right) \right]^{-1} \cdot \operatorname{Re}\left[\mathbf{H}^o \odot \left(\mathbf{D}_{dd}(\mathbf{A}^o)^H \mathbf{U}^{tf} \mathbf{A}^o \mathbf{D}_{dd}\right)^T\right] \\ &\quad \times \left[\operatorname{Re}\left(\mathbf{H}^o \odot \mathbf{D}_{dd}^T\right)\right]^{-1} \\ &= \frac{\sigma_n^2}{2N'} \left[ \operatorname{Re}\left(\mathbf{H}^o \odot (\mathbf{R}_{dd}^o)^T\right) \right]^{-1} \cdot \operatorname{Re}\left[\mathbf{H}^o \odot \left(\mathbf{R}_{dd}^o(\mathbf{A}^o)^H \mathbf{U}^{tf} \mathbf{A}^o \mathbf{R}_{dd}^o\right)^T\right] \\ &\quad \times \left[\operatorname{Re}\left(\mathbf{H}^o \odot \mathbf{R}_{dd}^o\right)^T\right]^{-1}, \end{aligned} \quad (17.51)$$

where  $\mathbf{U}^{tf}$  is defined in (17.36), and

$$\mathbf{H}^o = (\mathbf{C}^o)^H \left[ \mathbf{I} - \mathbf{A}^o \left( (\mathbf{A}^o)^H \mathbf{A}^o \right)^{-1} (\mathbf{A}^o)^H \right] \mathbf{C}^o, \quad \text{with } \mathbf{C}^o = d\mathbf{A}^o/d\omega. \quad (17.52)$$

In the case of  $n_o = n$ , then  $\mathbf{H}^o = \mathbf{H}$ , and  $\mathbf{C}^o = \mathbf{C}$ .

The signal localization in the t-f domain enables us to select fewer signal arrivals. This fact is not only important in improving the estimation performance, particularly when the signals are closely spaced, but also reduces the dimension of the optimization problem solved by the maximum likelihood algorithm, and subsequently reduces the computational requirement.

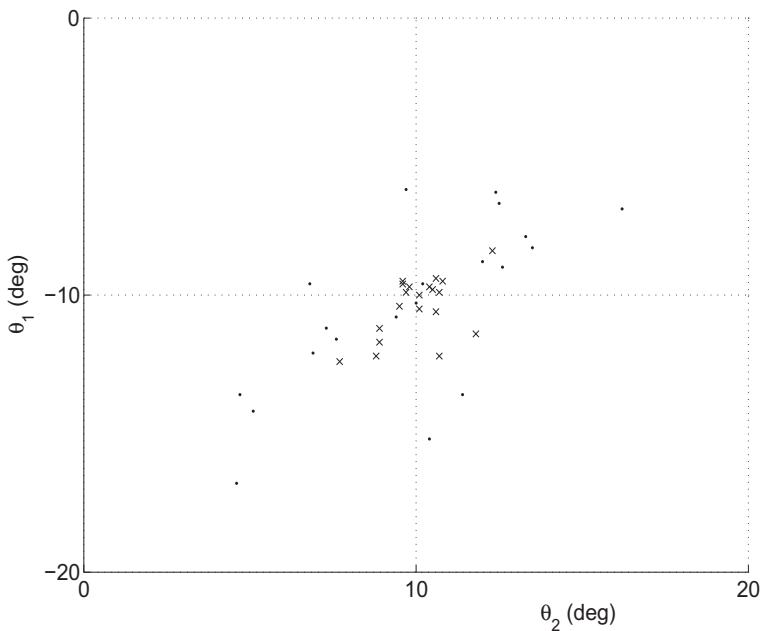
### 3.17.4.2.1 Examples

To demonstrate the advantages of t-f ML over both the conventional ML and the t-f MUSIC, consider a uniform linear array of eight sensors separated by half a wavelength. Two FM signals arrive from  $(\theta_1, \theta_2) = (-10^\circ, 10^\circ)$  with the IFs  $f_1(t) = 0.2 + 0.1t/N + 0.2 \sin(2\pi t/N)$  and  $f_2(t) = 0.2 + 0.1t/N + 0.2 \sin(2\pi t/N + \pi/2)$ ,  $t = 1, \dots, N$ . The SNR of both signals is  $-20$  dB, and the number of snapshots used in the simulation is  $N = 1024$ . We use  $L = 129$  for t-f ML. Figure 17.7 shows  $(\theta_1, \theta_2)$  that yield the minimum values of the likelihood function of the t-f ML and the ML methods for 20 independent trials. It is evident that the t-f ML provides much improved DOA estimation over the conventional ML.

In the next example, the t-f ML and the t-f MUSIC are compared for coherent sources. The two coherent FM signals have common IFs  $f_{1,2}(t) = 0.2 + 0.1t/N + 0.2 \sin(2\pi t/N)$ ,  $t = 1, \dots, N$ , with a  $\pi/2$  phase difference. The signals arrive at  $(\theta_1, \theta_2) = (-2^\circ, 2^\circ)$ . The SNR of both signals is  $5$  dB and the number of snapshots is  $1024$ . Figure 17.8 shows the contour plots of the likelihood function of the t-f ML and the estimated spectra of t-f MUSIC for three independent trials. It is clear that the t-f ML can separate the two signals, whereas the t-f MUSIC fails.

### 3.17.4.3 Effect of cross-terms

Auto-term and cross-term t-f points have different roles and contribute differently in DOA estimation. This section considers the behavior of cross-terms in DOA estimation and addresses the proper selection of auto-term and cross-term points.

**FIGURE 17.7**

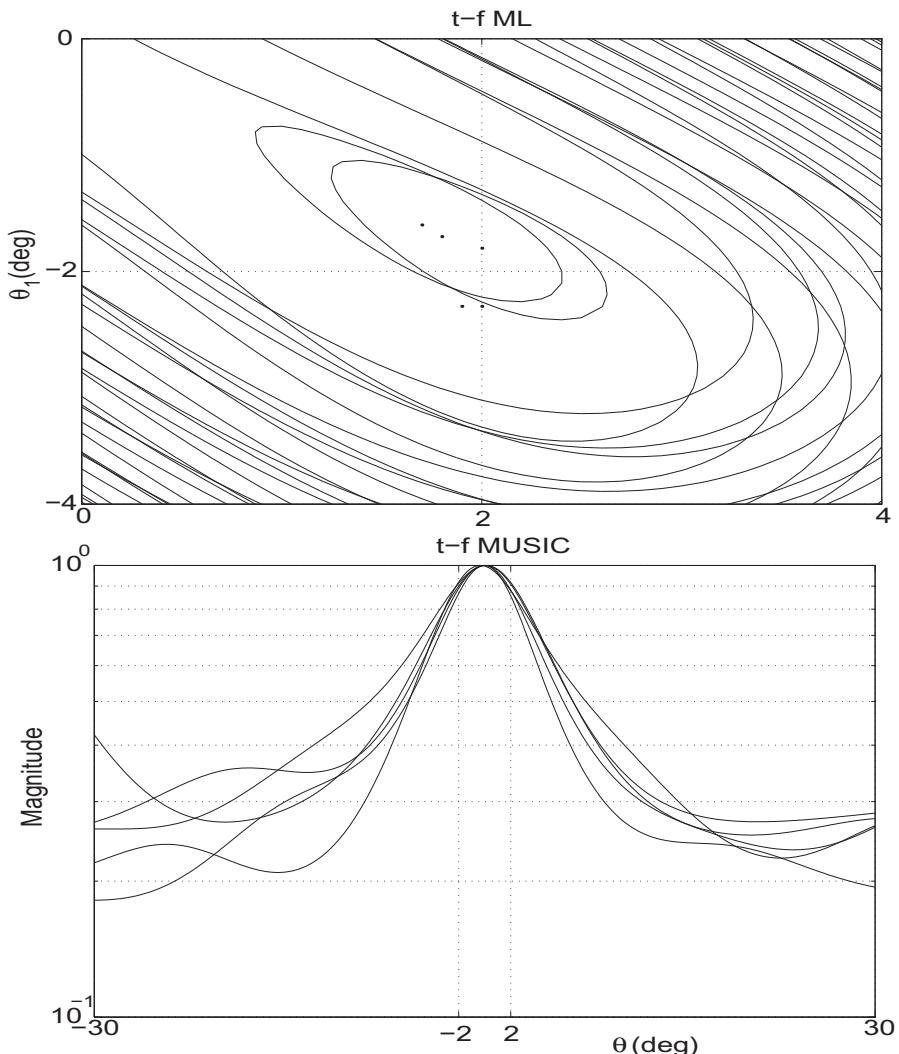
$(\theta_1, \theta_2)$  which minimize the t-f ML ("x") and ML ("·") likelihood functions.

As we discussed in Section 3.17.2, cross-terms are a by-product of the TFD due to its bilinearity. Although different kernels have different ways of mitigating cross-terms [30,59], complete removal of cross-terms, nevertheless, is in general difficult to achieve.

There are two types of cross-terms in the underlying DOA estimation problems. The first type is due to the interactions between the components of the same source signal. These cross-terms always reside, along with the auto-terms, on the main diagonal of the source TFD matrix. This type of cross-terms shares the same steering vector as the auto-terms and thus can be similarly treated. The other type of cross-terms is those generated from the interactions between two signal components belonging to two different sources. These cross-terms are associated with cross-TFD of the source signals and, at any given t-f point, they constitute the off-diagonal entries of the source TFD matrices. Here we consider the second type of cross-terms.

To understand the role of cross-terms in DOA estimation, it is important to compare the cross-terms to the cross-correlation between signals in conventional array processing, whose properties are well studied. The source TFD matrix takes the following general form:

$$\mathbf{D}_{dd}(t, f) = \begin{bmatrix} D_{d_1 d_1}(t, f) & D_{d_1 d_2}(t, f) & \cdots & D_{d_1 d_n}(t, f) \\ D_{d_2 d_1}(t, f) & D_{d_2 d_2}(t, f) & \cdots & D_{d_2 d_n}(t, f) \\ \vdots & \vdots & \ddots & \vdots \\ D_{d_n d_1}(t, f) & D_{d_n d_2}(t, f) & \cdots & D_{d_n d_n}(t, f) \end{bmatrix}. \quad (17.53)$$

**FIGURE 17.8**

Contour plots of t-f ML likelihood function (upper) and spatial spectra of t-f MUSIC (lower).

On the other hand, the covariance matrix of correlated source signals is given at the form

$$\mathbf{R}_{dd} = \begin{bmatrix} R_{d_1 d_1} & R_{d_1 d_2} & \cdots & R_{d_1 d_n} \\ R_{d_2 d_1} & R_{d_2 d_2} & \cdots & R_{d_2 d_n} \\ \vdots & \vdots & \ddots & \vdots \\ R_{d_n d_1} & R_{d_n d_2} & \cdots & R_{d_n d_n} \end{bmatrix}, \quad (17.54)$$

where the off-diagonal element  $R_{d_i d_j} = E[d_i(t)d_j^*(t)]$  represents the correlation between source signals  $d_i$  and  $d_j$ . Direction finding problems can usually be solved when the signals are partially correlated, however, full rank property of the source covariance matrix  $\mathbf{R}_{dd}$  is a necessary condition.

Comparing Eqs. (17.53) and (17.54), it is clear that the cross-correlation terms and the cross-terms have analogous forms. When cross-terms are present at the selected t-f point, these cross-terms appear as off-diagonal elements in the source TFD matrix. On the other hand, when signals are correlated, the off-diagonal elements of the covariance matrix of the source signals represent the cross-correlation between two source signals. DOA estimation problems can usually be solved when the signals are partially correlated, provided that the full rank property of the covariance matrix of the source signals is maintained. The cross-correlation terms and the cross-term TFDs have an analogous form and similar function. That is, cross-term TFDs can be exploited in the DOA estimation as long as the full rank subspace of the STFD matrix is achievable [10]. It is noted that the covariance matrix is obtained as a results of statistical or ensemble averages, whereas the STFD matrix is defined at a  $(t, f)$  point and its value usually varies with respect to time  $t$  and frequency  $f$ . When multiple  $(t, f)$  points are incorporated, the effect of a cross-term may be reduced, since the cross-term usually oscillates with respect to time.

### 3.17.4.4 DOA estimation based on signal stationarization

As we discussed in Section 3.17.2.3, FrFT can “rotate” LFM signals in the t-f domain and become sinusoidal signals in a transformed coordinate system. For narrowband LFM signals, the t-f signatures of the signals are identical for different array sensors, thus the same rotating operation stationarizes a signal component at all array sensors. Further, mask operations can be applied in the transformed domain to remove the effects of other components, whether they correspond to LFM or nonlinear FM signals. As such, the DOA estimation problem of LFM signals becomes equivalent to that of a single sinusoidal signal [60].

In general, for FM signals that are characterized by PPS or other time-varying IFs, their parameters can be estimated, as discussed in Section 3.17.2.4. The signal stationarization process converts an FM signal into a sinusoid or DC signal and, as such, allows a similar treatment [49]. The DOA estimation technique based on signal stationarization is proposed in [61].

For receiver array signals

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{d}(t) + \mathbf{n}(t) = \sum_{i=1}^n \mathbf{a}_i d_i(t) + \mathbf{n}(t), \quad (17.55)$$

where  $d_i(t) = D_i e^{j\phi_i(t)}$ , stationarization is performed by multiplying  $\mathbf{x}(t)$  with the conjugation of the  $k$ th signal component [61],

$$\mathbf{x}^{[k]}(t) = \mathbf{x}(t)e^{-j\phi_k(t)} = \mathbf{a}_k D_k + \sum_{i=1, i \neq k}^n \mathbf{a}_i e^{j(\phi_i(t) - \phi_k(t))}(t) + \mathbf{n}(t). \quad (17.56)$$

This operation transforms the selected  $k$ th signal from an FM signal to a DC signal. Other signal components, shown as the second term at the right-hand side of the above equation, will likely to have nonzero frequencies whenever the corresponding frequencies satisfy  $d\phi_i(t)/dt \neq d\phi_k(t)/dt$  for  $i \neq k$ .

As such, even with some perturbations induced due to imperfect stationarization, a mask around the DC region can be applied to only keep the  $k$ th signal, which is subsequently used for DOA estimation.

In [62], the signal stationarization is applied for the direction finding problem of multipath signals in an over-the-horizon radar (OTHR) system. It is shown that the stationarization operation allows separation of multipath signals, which have both closely separated Doppler signatures and close angular separation. This enables DOA estimations of individual components which are otherwise difficult to perform without pre-processing.

### 3.17.4.5 DOA estimation based on spatial joint-variable domain distributions

So far, we have considered the TFD which transforms a 1-D (time-domain) signal into a 2-D representation in the joint t-f domain. It is known that a nonstationary signal can be also represented in other joint-variable domains, such as the joint domain of time-lag and Doppler (frequency-lag), time and time-lag, and frequency and Doppler (frequency-lag) [7].

The ambiguity function of a signal  $x(t)$  is defined as

$$B_{xx}(\nu, \tau) = \int_{-\infty}^{\infty} x\left(u + \frac{\tau}{2}\right) x^*\left(u - \frac{\tau}{2}\right) e^{-j\nu\tau} du, \quad (17.57)$$

where  $\nu$  and  $\tau$  are the frequency lag and the time lag, respectively.

For the signal observed at an sensor array, we define the spatial ambiguity function (SAF) matrix of a signal vector  $\mathbf{x}(t)$  in a similar way to the STFD as [22]

$$\mathbf{B}_{xx}(\nu, \tau) = \int_{-\infty}^{\infty} \mathbf{x}\left(u + \frac{\tau}{2}\right) \mathbf{x}^H\left(u - \frac{\tau}{2}\right) e^{-j\nu\tau} du. \quad (17.58)$$

In a noise-free environment,  $\mathbf{x}(t) = \mathbf{A}\mathbf{d}(t)$ , the SAF is related to the source ambiguity matrix  $\mathbf{B}_{dd}(\nu, \tau)$  by

$$\mathbf{B}_{xx}(\nu, \tau) = \mathbf{A}\mathbf{B}_{dd}(\nu, \tau)\mathbf{A}^H. \quad (17.59)$$

Equations (17.58) and (17.59) are similar to the STFD matrix and thus the SAF inherits the properties of the STFD.

The SAFs have the following two important offerings that distinguish them from other array spatial functions. (1) The cross-terms in between source signals reside on the off-diagonal entries of source ambiguity matrix  $\mathbf{B}_{dd}(\nu, \tau)$ . In the ambiguity domain, the signal auto-terms are positioned near and at the origin, making it easier to leave out cross-terms from matrix construction. (2) In the ambiguity domain, the auto-terms of all narrowband signals, regardless of their frequencies and phases, fall on the time-lag axis ( $\nu = 0$ ), while those of the wideband signals fall on a different  $(\nu, \tau)$  region or spread over the entire ambiguity domain. Therefore, the SAF is a natural choice for recovering and spatially localizing narrowband sources in broadband signal platforms.

### 3.17.4.6 DOA estimation of wideband nonstationary signals

The discussion so far has been focused on the DOA estimation of narrowband nonstationary signals. In many applications, the signals are rather wideband. In this case, the DOA estimator should consider the fact that the steering vector is now frequency-dependent.

In order to estimate the DOA for a general class of wideband signals, the conventional techniques usually use Fourier transform to decompose the wideband signals into a set of narrowband components. The narrowband signals can then be processed either incoherently or coherently. The incoherent-based approaches are relatively simple and estimate the DOA from the average of the spatial spectra corresponding to different frequency bins. However, coherent approaches are often preferred due to their superior performance compared to incoherent ones. A popularly used technique, namely, the coherent signal-subspace (CSS) processing technique, was proposed by Wang and Kaveh [63] and was further developed in several papers (see [64] and references therein). The fundamental concept of the CSS techniques is to use a set of focusing matrices that map the steering vector at different frequencies into that at a reference frequency prior to coherent combining.

Several t-f and ambiguity domain based DOA estimation methods have been developed for the estimation of wideband LFM signals [11,23,65]. Wang and Xia [65] employs the t-f analysis to estimate the chirp rates and compensates the signal chirp structure in an iterative manner. A good estimate of the signal DOAs is required to initialize the iterative processing. By assuming that the wideband signals are separable in the t-f domain and their IFs do not rapidly change, [11] uses a sufficiently short sliding window to construct the STFD matrices so as to preserve the narrowband structure of the array manifold. The focusing matrices are then applied to the STFD matrices at selected t-f points corresponding to the source t-f signatures. Ma and Goh [23] considers the ambiguity domain for the DOA estimation of wideband LFM signals whose chirp rates are assumed to be known. Multiple chirps with identical chirp rates are allowed in this technique. Performance of incoherent and coherent processing techniques is also compared in [23].

In essence, STFD framework permits wideband DOA estimation methods incorporating the CSS approaches for nonstationary signals. The nature of the LFM signals and the offering of t-f signal representations may be utilized in several aspects. (i) The decomposition of the LFM signals into a spectrum of frequency bins is inherently performed in the t-f analysis. (ii) For LFM signals that have distinct characteristics in the t-f domain, DOA estimation can be performed on individual sources. (iii) LFM signals are instantaneous narrowband, allowing the focusing matrices to be applied to t-f points.

### 3.17.5 Joint DOD/DOA estimation in MIMO radar systems

In this section, we discuss the STFD framework in the context of joint direction-of-departure (DOD)/direction-of-arrival (DOA) estimation in multiple-input multiple-output (MIMO) radar configurations [25]. MIMO radar is an emerging technology that has attracted much interest in the radar community [66,67]. By emitting orthogonal waveforms from the transmit array antennas and utilizing matched filterbanks in the receivers to extract the orthogonal waveform components, MIMO radar systems can exploit the spatial diversity and the higher number of degrees of freedom to improve resolution, clutter mitigation, and classification performance. In particular, a monostatic MIMO radar system can provide effective array designs to achieve an extended virtual array, which is the sum coarray of the transmit array and the receive arrays [67,68]. A bistatic MIMO radar, on the other hand, is capable to jointly estimate the DOD and DOA of targets for enhanced target localization [69–72]. Bistatic radars, in which the transmitters and receivers are separated by a considerable distance, have received

increasing attentions because of many potential advantages, such as detection of stealthy targets, covert receivers for safe operation, and increased coverage [73]. The DOD and DOA information obtained from a bistatic radar system is particularly important in narrowband radar systems, such as over-the-horizon radar, which do not have a high range resolution [25, 74]. It is shown in [75] that nonstationary processing in a MIMO radar platform also yields improved estimation of motion parameters whose Doppler law is characterized by PPS models.

### 3.17.5.1 Signal model

Consider a bistatic MIMO radar system consisting of  $N_t$  closely spaced transmit antennas and  $N_r$  closely spaced receive antennas. Denote  $\mathbf{S} \in \mathbb{C}^{N_t \times T}$  as the narrowband waveform matrix which contains orthogonal waveforms to be transmitted from  $N_t$  antennas over a pulse-repetition period of  $T$  fast-time samples. We assume that the waveform orthogonality is achieved in the fast-time domain. That is, by denoting  $\mathbf{s}_i$  as the  $i$ th row of matrix  $\mathbf{S}$ ,  $\mathbf{s}_i$  and  $\mathbf{s}_l$  are orthogonal for any  $i \neq l$  with different delays, and  $\mathbf{s}_i$  is orthogonal to the delayed version of itself. We also assume that  $\mathbf{s}_i$  has a unit norm, i.e.,  $\mathbf{S}\mathbf{S}^H = \mathbf{I}_{N_t}$ .

Consider a far-field range cell where  $L$  point targets are present with DOD  $\theta_l$  and DOA  $\phi_l$ , where  $l = 1, \dots, L$ . Then, the signal data received at the receive array corresponding to the range cell is expressed as the following  $N_r \times 1$  complex vector,

$$\mathbf{X}(t) = \mathbf{A}_r \boldsymbol{\Gamma}(t) \mathbf{A}_t^H \mathbf{S} + \mathbf{N}(t), \quad (17.60)$$

where  $t$  is the slow time index,  $\mathbf{A}_r = [\mathbf{a}_r(\phi_1), \dots, \mathbf{a}_r(\phi_L)]$  and  $\mathbf{A}_t = [\mathbf{a}_t(\theta_1), \dots, \mathbf{a}_t(\theta_L)]$ , with  $\mathbf{a}_r(\phi_l) \in \mathbb{C}^{N_r \times 1}$  and  $\mathbf{a}_t(\theta_l) \in \mathbb{C}^{N_t \times 1}$  respectively denoting the receive steering vector corresponding to DOA  $\phi_l$  and the transmit steering vector corresponding to DOD  $\theta_l$ . In addition,  $\boldsymbol{\Gamma}(t) = \text{Diag}[\gamma_1(t), \dots, \gamma_L(t)]$  where  $\gamma_l(t) = \rho_l e^{j2\pi\beta(f_{D,l}(t), t)}$  denotes the complex reflection coefficient of the  $l$ th target during the  $t$ th pulse repetition period. The complex reflection coefficient is a function of the radar cross section (RCS), represented by  $\rho_l$ , and the phase term, denoted as  $\beta(f_{D,l}(t), t)$ , which depends on the Doppler frequency  $f_{D,l}(t)$  of the slow time index  $t$ . Moreover,  $\mathbf{N}(t) \in \mathbb{C}^{N_r \times T}$  is an additive noise matrix, whose elements are assumed to be i.i.d. complex Gaussian random variables with zero mean and variance  $\sigma_n^2$ . To qualify expression (17.60), it is assumed that the steering vectors remain unchanged during the entire slow-time processing period, which is often the case for far-field targets. The nonstationary signatures result from the maneuvering flights of targets, represented by the Doppler frequency  $f_{D,l}(t)$ .

By post-multiplying (17.60) by  $\mathbf{S}^H$  and utilizing the orthogonality of the transmitted waveforms, we obtain  $\mathbf{Y}(t) \in \mathbb{C}^{N_t \times N_r}$  as

$$\mathbf{Y}(t) = \mathbf{A}_r \boldsymbol{\Gamma}(t) \mathbf{A}_t^H + \mathbf{Z}(t), \quad (17.61)$$

where  $\mathbf{Z}(t) = \mathbf{N}(t) \mathbf{S}^H$ . Vectorizing  $\mathbf{Y}(t)$  in (17.61) yields the following  $N_t N_r \times 1$  vector

$$\mathbf{y}(t) = \mathbf{w}(t) + \mathbf{z}(t) = \mathbf{A} \boldsymbol{\gamma}(t) + \mathbf{z}(t), \quad (17.62)$$

where  $\mathbf{w}(t) = \mathbf{A} \boldsymbol{\gamma}(t)$  is the noise-free portion of the signal vector,

$$\mathbf{A} = \mathbf{A}_t \diamond \mathbf{A}_r = \left[ \mathbf{a}_1^{[t]} \otimes \mathbf{a}_1^{[r]}, \dots, \mathbf{a}_L^{[t]} \otimes \mathbf{a}_L^{[r]} \right], \quad (17.63)$$

with  $\mathbf{a}_l^{[t]}$  and  $\mathbf{a}_l^{[r]}$  denoting the  $l$ th column of  $\mathbf{A}_t$  and  $\mathbf{A}_r$ , respectively. In addition,  $\boldsymbol{\gamma}(t) = \text{diag}(\boldsymbol{\Gamma}(t)) = [\gamma_1(t), \dots, \gamma_L(t)]^T$ , and  $\mathbf{z}(t) = \text{vec}(\mathbf{Z}(t))$ .

The noise component corresponding to the  $m$ th transmit waveform and the  $n$ th receive antenna is given by  $z_{n,m}(t) = [\mathbf{z}(t)]_{(m-1)N_r+n} = \tilde{\mathbf{n}}_n(t)\tilde{\mathbf{s}}_m^H$ , where  $\tilde{\mathbf{n}}_n(t)$  is the  $n$ th row of the receive noise matrix  $\mathbf{N}(\mathbf{t})$ , and  $\tilde{\mathbf{s}}_m$  is the  $m$ th row of waveform matrix  $\mathbf{S}$ ,  $m = 1, \dots, N_t$  and  $n = 1, \dots, N_r$ . Notice that we used  $(\cdot)$  to emphasize a row vector. It is clear that vector  $\mathbf{z}(t)$  has a zero mean, spatially white across the virtual sensors, and its covariance matrix can be shown to be  $\sigma_n^2 \mathbf{I}_{N_t N_r}$  because

$$\mathbb{E}[z_{n_1,m_1}(t)z_{n_2,m_2}^*(t)] = \mathbb{E}\left[\tilde{\mathbf{n}}_{n_1}(t)\tilde{\mathbf{s}}_{m_1}^H \left(\tilde{\mathbf{n}}_{n_2}(t)\tilde{\mathbf{s}}_{m_2}^H\right)^*\right] = \mathbb{E}\left[\tilde{\mathbf{s}}_{m_2}\tilde{\mathbf{n}}_{n_2}^H(t)\tilde{\mathbf{n}}_{n_1}(t)\tilde{\mathbf{s}}_{m_1}^H\right] = \sigma_n^2 \delta_{n_1,n_2} \delta_{m_1,m_2}. \quad (17.64)$$

### 3.17.5.2 Joint DOD/DOA estimations

In bistatic radars, the DOD and DOA information can be synthesized to locate targets. For multiple targets, the combination of estimated DOD and DOA yields paring ambiguity. Several techniques have been developed to void or to automatically obtain pairing operation [69, 72]. These approaches, based on ESPRIT [76], or combined ESPRIT-MUSIC, can be extended to the t-f framework. We consider, as an example, the combined ESPRIT-MUSIC technique developed in [69] which only requires two decoupled one-dimensional direction finding operations where the DOD and DOA are automatically paired. In this section, we extend this technique into the STFD framework. The DODs of the targets are first estimated using t-f ESPRIT [12] and their DOAs are then obtained using t-f MUSIC [5]. To apply ESPRIT-based method, both arrays are assumed to be uniform and linear, but the interelement spacings of the two arrays, respectively denoted as  $d_t$  and  $d_r$ , may differ.

Consider a t-f region  $\Omega_0$  that contains signal returns from  $L_0 \leq L$  targets. An STFD matrix, denoted as  $\mathbf{D}_{yy}(\Omega_0)$ , can be obtained through weighted average of the STFD matrices across region  $\Omega_0$ , i.e.,

$$\mathbf{D}_{yy}(\Omega_0) = \sum_{(t,f) \in \Omega_0} w(t, f) \mathbf{D}_{yy}(t, f), \quad (17.65)$$

where  $w(t, f)$  is the weighting coefficients, which can be chosen to be identical or proportional to the TFD magnitude. The signal subspace of matrix  $\mathbf{D}_{yy}(\Omega_0)$  corresponds to the  $L_0$  target signals contained in the selected t-f region  $\Omega_0$ . In other words, it spans the same subspace as  $\mathbf{A}_0$ , where  $\mathbf{A}_0 = \mathbf{A}_{0,t} \diamond \mathbf{A}_{0,r}$  is a  $N_t N_r \times L_0$  submatrix of  $\mathbf{A}$  that contains the  $L_0$  columns of matrix  $\mathbf{A}$ , corresponding to the  $L_0$  signals included in the selected t-f region.

Performing eigendecomposition of  $\mathbf{D}_{yy}(\Omega_0)$  and denote  $\mathbf{U}_{s,0}$  as its  $N_t N_r \times L_0$  signal subspace, whereas  $\mathbf{U}_{n,0}$  as the  $N_t N_r \times (N_t N_r - L_0)$  noise subspace. Then,  $\mathbf{U}_{s,0}$  and  $\mathbf{A}_0$  are related by an unknown transformation matrix  $\mathbf{T}$  as

$$\mathbf{U}_{s,0} = \mathbf{A}_0 \mathbf{T}. \quad (17.66)$$

Divide the virtual array into two overlapping subarrays, respectively consisting of the first and last  $(N_t - 1)N_r$  virtual antennas. Denote  $\mathbf{A}_{0,t}^{(1)}$  and  $\mathbf{A}_{0,t}^{(2)}$  as the first and last  $N_t - 1$  rows of  $\mathbf{A}_{0,t}$ , and let  $\mathbf{A}_0^{(t1)} = \mathbf{A}_{0,t}^{(1)} \diamond \mathbf{A}_{0,r}$  and  $\mathbf{A}_0^{(t2)} = \mathbf{A}_{0,t}^{(2)} \diamond \mathbf{A}_{0,r}$ . Further, denote the averaged STFD matrices defined in these subarrays as  $\mathbf{D}_{yy}^{(1)}(\Omega_0)$  and  $\mathbf{D}_{yy}^{(2)}(\Omega_0)$ , respectively. Then, their respective signal subspaces relate to  $\mathbf{A}_0^{(t1)}$  and  $\mathbf{A}_0^{(t2)}$  through

$$\mathbf{U}_{s,0}^{(1)} = \mathbf{A}_0^{(t1)} \mathbf{T}, \quad \mathbf{U}_{s,0}^{(2)} = \mathbf{A}_0^{(t2)} \mathbf{T}. \quad (17.67)$$

$\mathbf{A}_0^{(t2)}$  and  $\mathbf{A}_0^{(t1)}$  differ due to the antenna position and thus are related by

$$\mathbf{A}_0^{(t2)} = \mathbf{A}_0^{(t1)} \Phi_{[t]}, \quad (17.68)$$

where  $\Phi_{[t]}$  is a diagonal matrix with diagonal elements  $[\Phi_{[t]}]_{i,i} = \exp(j2\pi d_t \sin(\theta_i)/\lambda)$ ,  $i = 1, \dots, L_0$ . Similarly,  $\mathbf{U}_{s,0}^{(1)}$  and  $\mathbf{U}_{s,0}^{(2)}$  are related by

$$\mathbf{U}_{s,0}^{(2)} = \mathbf{U}_{s,0}^{(1)} \Psi_{[t]}. \quad (17.69)$$

From the above results,  $\Psi_{[t]}$  can be obtained from  $\mathbf{U}_{s,0}^{(1)}$  and  $\mathbf{U}_{s,0}^{(2)}$ . Substituting (17.68) into (17.69), we obtain

$$\mathbf{U}_{s,0}^{(1)} = \mathbf{A}_0^{(t1)} \mathbf{T}, \quad \mathbf{U}_{s,0}^{(2)} = \mathbf{A}_0^{(t1)} \Phi_{[t]} \mathbf{T}. \quad (17.70)$$

Therefore, it is concluded from (17.69) and (17.70) that  $\Psi_{[t]}$  and  $\Phi_{[t]}$  are related by  $\Psi_{[t]} = \mathbf{T}^{-1} \Phi_{[t]} \mathbf{T}$ , that is,  $\Phi_{[t]}$  can be obtained as the eigenvalues of  $\Psi_{[t]}$ . As such, the DODs  $\theta_i$  can be obtained for  $i = 1, \dots, L_0$ .

To estimate the DOAs after DODs are obtained, the ESPRIT-MUSIC method is based on the fact that noise subspace and the steering vector of the virtual array are orthogonal [69]. In the t-f framework, this leads to a t-f MUSIC based approach for each estimated  $\theta_i$ ,  $i = 1, \dots, L_0$ , i.e., estimating the paired  $\phi_i$  by finding the peaks of the following pseudo spatial spectrum

$$f(\phi) = \frac{1}{\mathbf{a}_r^H(\phi) [\mathbf{a}_t(\theta_i) \otimes \mathbf{I}_{N_r}]^H \mathbf{U}_{n,0} \mathbf{U}_{n,0}^H [\mathbf{a}_t(\theta_i) \otimes \mathbf{I}_{N_r}] \mathbf{a}_r(\phi)}. \quad (17.71)$$

When the receive array is uniform linear, for which the receive steering vector can be expressed as a polynomial function of  $z = \exp(-j2\pi d_r \sin(\phi)/\lambda)$ , i.e.,

$$\mathbf{a}_r(\phi) = \left[ 1, e^{-\frac{j2\pi d_r}{\lambda} \sin(\phi)}, \dots, e^{-\frac{j2\pi(N_r-1)d_r}{\lambda} \sin(\phi)} \right]^T = \left[ 1, z, \dots, z^{N_r-1} \right]^T, \quad (17.72)$$

the paired DOA  $\phi_i$  can be solved using the simpler t-f root-MUSIC approach that finds the root inside and closest to the unit circle of the following polynomial:

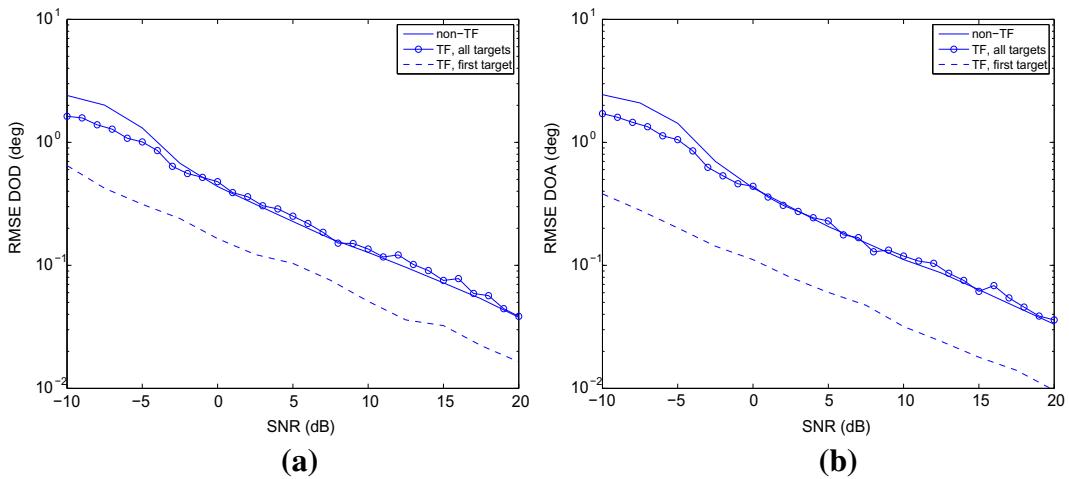
$$\mathbf{a}_r^H(\phi) [\mathbf{a}_t(\theta_i) \otimes \mathbf{I}_{N_r}]^H \mathbf{U}_{n,0} \mathbf{U}_{n,0}^H [\mathbf{a}_t(\theta_i) \otimes \mathbf{I}_{N_r}] \mathbf{a}_r(\phi) = 0. \quad (17.73)$$

For the directions of other  $L - L_0$  targets, the same procedure can be carried out in different t-f regions where these signals are included.

By exploiting source selection/discrimination through t-f region selection, significant performance improvement can be achieved. This is particularly true in the challenging situations when multiple targets are closely spaced in angle but are separable in the time-frequency domain. Specifically, when a t-f region corresponding to a single target is identified, the DOD and DOA can be estimated with simple phase examinations, and no paring operation is needed.

### 3.17.5.2.1 Example

Consider a scenario in which two moving targets appear in a specific range bin of interest. The bistatic radar consists of a linear transmit array consisting of  $N_t = 4$  antennas and a linear receive array of  $N_r = 6$  antennas. The transmit and receive arrays are assumed to be distantly separated. Half wavelength

**FIGURE 17.9**

Comparison of RMSE performance. (a) DOD estimation. (b) DOA estimation.

interelement spacing is set for both transmit and receive arrays. The waveforms emitted from different transmit antennas are considered orthogonal, i.e., their crosscorrelations are ignored. The total number of samples is 256 for each waveform. The two targets have close DODs ( $10^\circ$  and  $15^\circ$ ) and DOAs ( $5^\circ$  and  $20^\circ$ ). The input SNR of all the return signals are assumed to be identical. The start frequencies of the two chirp signals are 0.15 and 0.18, and the respective ending frequencies are 0.35 and 0.38. The increasing Doppler signature of each target indicates the target movement towards the transmit and receive arrays in a way that the sum two-way slant range decreases over time.

In Figure 17.9, the root-mean-square error (RMSE) of the DOD and DOA estimation results of the first target are compared for three different scenarios, namely, joint ESPRIT-MUSIC without the use of time-frequency analysis, time-frequency ESPRIT-MUSIC with both signals selected for consideration, and time-frequency ESPRIT-MUSIC that only considers the signal corresponding to the first target. The results are averaged over 100 independent trials. It is evident that when both signals are selected, the t-f ESPRIT-MUSIC still benefits from the SNR enhancement over low SNR regions. It is also clear that the performance of both DOD and DOA estimates is significantly improved through target discrimination by considering only the first target. This improvement stems from overcoming the close angular separation of the targets at both the transmitter and receiver sides using t-f signature selections.

### 3.17.6 Conclusion

This chapter discussed direction of arrival estimation of nonstationary signals that are characterized by instantaneous frequency laws. Conventional direction finding methods, including high resolution techniques, do not properly account for the instantaneous frequency characterization of the signals impinging on the antenna arrays. We discussed the spatial time-frequency distribution (STFD) framework which

permits eigenstructure subspace methods to utilize the signal-to-noise ratio enhancement, brought about by incorporating the time-frequency regions of high power concentration. The latter are typically found at and around the signal time-frequency signature. High SNR data enable robustness of DOA estimates. It was also shown that distinction in the time-frequency signatures of closely spaced sources provides a discriminatory capability, within the STFD framework, which allows reducing the number of sources in the field of view to a single, or a subgroup of the sources. This permits processing more sources than sensors and reduces the variance of the source angular estimate. We extended the STFD framework to include multiple-input multiple-output (MIMO) configurations and estimated both the source direction-of-departure and direction-of-arrival. Although the focus of the chapter was on bilinear distributions of nonstationary signals, we also addressed linear time-frequency methods and their applications to DOA estimation of polynomial phase sources.

## Glossary

Time-frequency distribution	distribution of the signal power over both the time and frequency variables
Spatial time-frequency distribution	a matrix whose entries are the time-frequency distributions associated with the outerproducts of the data observation vectors measured across a sensor array
Auto-term	a sample in the time-frequency domain which pertains to the time-frequency distribution of an individual component of the signal
Cross-term	an artifact in the time-frequency domain which is introduced by the bilinear product of two components of the input signal
Source discrimination	isolation of individual source signals in single variable or joint-variables domains, such as time, frequency, space, and time-frequency domains

### *Relevant Theory:* Statistical Signal Processing

See this Volume, [Chapter 1](#) Introduction: Statistical Signal Processing

See this Volume, [Chapter 3](#) Non-stationary Signal Analysis

## References

- [1] M.G. Amin, Y. Zhang, Spatial time-frequency distributions and their applications, in: B. Boashash (Ed.), Time-Frequency Signal Analysis and Processing, Elsevier, Oxford, UK, 2003.
- [2] M.G. Amin, Y. Zhang, Spatial time-frequency distributions and DOA estimation, in: E. Tuncer, B. Friedlander (Eds.), Classical and Modern Direction of Arrival Estimation, Academic Press, Burlington, MA, 2009.
- [3] M.G. Amin, Y. Zhang, G.J. Frazer, A.R. Lindsey, Spatial time-frequency distributions: theory and applications, in: L. Debnath (Ed.), Wavelets and Signal Processing, Birkhauser, Boston, MA, 2003.

- [4] A. Belouchrani, M.G. Amin, Blind source separation based on time-frequency signal representations, *IEEE Trans. Signal Process.* 46 (11) (1998) 2888–2897.
- [5] A. Belouchrani, M.G. Amin, Time-frequency MUSIC, *IEEE Signal Process. Lett.* 6 (5) (1999) 109–110.
- [6] Y. Zhang, W. Mu, M.G. Amin, Subspace analysis of spatial time-frequency distribution matrices, *IEEE Trans. Signal Process.* 49 (4) (2001) 747–759.
- [7] B. Boashash, Theory of quadratic TFDs, in: B. Boashash (Ed.), *Time-Frequency Signal Analysis and Processing*, Elsevier, Oxford, UK, 2003.
- [8] S. Qian, D. Chen, *Joint Time-Frequency Analysis—Methods and Applications*, Prentice Hall, Englewood Cliffs, NJ, 1996.
- [9] M.G. Amin, Interference mitigation in spread spectrum communication systems using time-frequency distributions, *IEEE Trans. Signal Process.* 45 (1) (1997) 90–101.
- [10] M.G. Amin, Y. Zhang, Direction finding based on spatial time-frequency distribution matrices, *Digit. Signal Process.* 10 (4) (2000) 325–339.
- [11] A. Gershman, M.G. Amin, Wideband direction-of-arrival estimation of multiple chirp signals using spatial time-frequency distributions, *IEEE Signal Process. Lett.* 7 (6) (2000) 152–155.
- [12] A. Hassanien, A.B. Gershman, M.G. Amin, Time-frequency ESPRIT for direction-of-arrival estimation of chirp signals, in: Proceedings of the IEEE Sensor Array and Multichannel Signal Processing Workshop, Rosslyn, VA, August 2002, pp. 337–341.
- [13] K. Sekihara, S. Nagarajan, D. Poeppel, Y. Miyashita, Time-frequency MEG-MUSIC algorithm, *IEEE Trans. Med. Imag.* 18 (1) (1999) 92–97.
- [14] Q. Wang, C. Wu, A high reliability DOA estimation method—TF-ESPRIT method, in: Proceedings of the Int. Conf. Signal Process., Beijing, China, August 2002, pp. 374–377.
- [15] Y. Zhang, W. Mu, M.G. Amin, Time-frequency maximum likelihood methods for direction finding, *J. Franklin Inst.* 337 (4) (2000) 483–497.
- [16] S. Rickard, F. Dietrich, DOA estimation of many W-disjoint orthogonal sources from two mixtures using DUET, in: Proceedings of the IEEE Workshop on Statistical Signal and Array Processing, Pocono, PA, August 2000, pp. 311–314.
- [17] A. Belouchrani, M.G. Amin, N. Thirion-Moreau, Y.D. Zhang, Source separation and localization using time-frequency distributions, *IEEE Signal Process. Mag.* (in press).
- [18] M.G. Amin, Minimum variance time-frequency distribution kernels for signal in additive noise, *IEEE Trans. Signal Process.* 44 (9) (1996) 2352–2356.
- [19] W. Mu, M.G. Amin, Y. Zhang, Bilinear signal synthesis in array processing, *IEEE Trans. Signal Process.* 51 (1) (2003) 90–100.
- [20] N. Linh-Trung, A. Belouchrani, K. Abed-Meraim, B. Boashash, Separating more sources than sensors using time-frequency distributions, *EURASIP J. Appl. Signal Process.* 2005 (17) (2005) 2828–2847.
- [21] Y. Zhang, M.G. Amin, Blind separation of nonstationary sources based on spatial time-frequency distributions, *EURASIP J. Appl. Signal Process.* 2006 (2006) 13 (Article ID 64785).
- [22] M.G. Amin, A. Belouchrani, Y. Zhang, The spatial ambiguity function and its applications, *IEEE Signal Process. Lett.* 7 (6) (2000) 138–140.
- [23] N. Ma, J.T. Goh, Ambiguity-function-based techniques to estimate DOA of broadband chirp signals, *IEEE Trans. Signal Process.* 54 (5) (2006) 1826–1839.
- [24] B.A. Obeidat, Y. Zhang, M.G. Amin, DOA and polarization estimation for wideband sources, in: Proceedings of the Asilomar Conference on Signals, System, Computers, Pacific Grove, CA, November 2004.
- [25] Y.D. Zhang, M.G. Amin, B. Himed, Joint DOD/DOA estimation in MIMO radar exploiting time-frequency signal representations, *EURASIP J. Adv. Signal Process.* 2012 (1) (2012) 102.
- [26] F. Auger, P. Flandrin, P. Gonçalvés, O. Lemoine, Time-frequency toolbox for use with Matlab. <<http://tftb.nongnu.org/tutorial.pdf>>.

- [27] A. Papandreou-Suppappola, Applications in Time-Frequency Signal Processing, CRC Press, Boca Raton, FL, 2003.
- [28] L.B. Almeida, The fractional Fourier transform and time-frequency representations, *IEEE Trans. Signal Process.* 42 (11) (1994) 3084–3091.
- [29] L. Cohen, Time-frequency distributions—a review, *Proc. IEEE* 77 (7) (1989) 941–981.
- [30] L. Cohen, Time-Frequency Analysis, Prentice Hall, Englewood Cliffs, NJ, 1995.
- [31] B. Boashash, G.R. Putland, Discrete time-frequency distributions, in: B. Boashash (Ed.), *Time-Frequency Signal Analysis and Processing*, Elsevier, Oxford, UK, 2003.
- [32] W. Mu, M.G. Amin, SNR analysis of time-frequency distributions, in: Proceedings of the IEEE Int. Conf. Acoust. Speech Signal Process. Istanbul, Turkey, June 2000, pp. II645–II648.
- [33] X-G. Xia, V. Chen, A quantitative SNR analysis for the pseudo Wigner-Ville distribution, *IEEE Trans. Signal Process.* 47 (10) (1999) 2891–2894.
- [34] H.I. Choi, W.J. Williams, Improved time-frequency representation of multicomponent signals using exponential kernels, *IEEE Trans. Acoust. Speech Signal Process. ASSP-37* (6) (1989) 862–871.
- [35] Y. Zhao, L.E. Atlas, R.J. Marks, The use of cone-shaped kernels for generalized time-frequency representations of non-stationary signals, *IEEE Trans. Acoust. Speech Signal Process. ASSP-38* (1990) 1084–1091.
- [36] R.G. Baraniuk, D.L. Jones, A signal-dependent time-frequency representation: optimal kernel design, *IEEE Trans. Signal Process.* 41 (1993) 1589–1602.
- [37] W. Li, Wigner distribution method equivalent to dechirp method for detecting a chirp signal, *IEEE Trans. Acoust. Speech Signal Process. ASSP-35* (1987) 1210–1211.
- [38] J.C. Wood, D.T. Barry, Radon transformation of time-frequency distributions for analysis of multicomponent signals, *IEEE Trans. Signal Process.* 42 (11) (1994) 3166–3177.
- [39] V. Namias, The fractional order Fourier transform and its application to quantum mechanics, *J. Inst. Math. Appl.* 25 (1980) 241–265.
- [40] S. Das, I. Pan, Fractional Order Signal Processing: Introductory Concepts and Applications, Springer, 2012.
- [41] S. Barbarossa, Analysis of multicomponent LFM signals by a combined Wigner-Hough transform, *IEEE Trans. Signal Process.* 43 (6) (1995) 1511–1515.
- [42] S. Barbarossa, O. Lemoine, Analysis of nonlinear FM signals by pattern recognition of their time-frequency representation, *IEEE Signal Process. Lett.* 3 (4) (1996) 112–115.
- [43] S. Peleg, B. Porat, Estimation and classification of polynomial-phase signals, *IEEE Trans. Inform. Theory* 37 (1991).
- [44] C. Ioana, A. Quinquis, Time-frequency analysis using warped-based high-order phase modeling, *EURASIP J. Applied Signal Process.* 2005 (17) (2005) 2856–2873.
- [45] B. Porat, Digital Processing of Random Signals: Theory and Methods, Prentice Hall, Englewood Cliffs, NJ, 1993.
- [46] S. Barbarossa, A. Scaglione, G.B. Giannakis, Product high-order ambiguity function for multicomponent polynomial-phase signal modeling, *IEEE Trans. Signal Process.* 46 (3) (1998) 691–708.
- [47] D.S. Pham, A.M. Zoubir, Analysis of multicomponent polynomial phase signals, *IEEE Trans. Signal Process.* 55 (1) (2007) 56–65.
- [48] S. Djukanović, M. Daković, L. Stanković, Local polynomial Fourier transform receiver for nonstationary interference excision in DSSS communications, *IEEE Trans. Signal Process.* 56 (4) (2008) 1627–1636.
- [49] C. Ioana, Y.D. Zhang, M.G. Amin, F. Ahmad, B. Himed, Time-frequency analysis of multipath Doppler signatures of maneuvering targets, in: Proceedings of the IEEE International Conference Acoustics, Speech, Signal Process., Kyoto, Japan, March 2012.
- [50] C. Ioana, Y.D. Zhang, M.G. Amin, F. Ahmad, G. Frazer, B. Himed, Time-frequency characterization of micro-multipath signals in over-the-horizon radar, in: Proceedings of the IEEE International Radar Conference, Atlanta, GA, May 2012, pp. 671–675.

- [51] M.G. Amin, Time-frequency spectrum analysis and estimation for nonstationary random processes, in: B. Boashash (Ed.), *Time-Frequency Signal Analysis: Methods and Applications*, Longman Cheshire, 1992.
- [52] S. Hearon, M.G. Amin, Minimum variance time-frequency distribution kernels, *IEEE Trans. Signal Process.* 43 (1995) 1258–1262.
- [53] L. Stankovic, A time-frequency distribution concentrated along the instantaneous frequency, *IEEE Signal Process. Lett.* 3 (3) (1996) 89–91.
- [54] L. Stankovic, Analysis of noise in time-frequency distributions, *IEEE Signal Process. Lett.* 9 (9) (2002) 286–289.
- [55] P. Stoica, A. Nehorai, MUSIC, maximum likelihood and Cramer-Rao bound, *IEEE Trans. Acoust. Speech Signal Process. ASSP-37* (5) (1989) 720–741.
- [56] R.O. Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Trans. Antennas Propagat. AP-34* (3) (1986) 276–280.
- [57] M.G. Amin, Spatial time-frequency distributions for direction finding and blind source separation, in: *Proceedings of the SPIE Wavelet Conference*, Orlando, FL, April 1999.
- [58] I. Ziskind, M. Wax, Maximum likelihood localization of multiple sources by alternating projection, *IEEE Trans. Acoust. Speech Signal Process. ASSP-36* (10) (1988) 1553–1560.
- [59] J. Jeong, W.J. Williams, Kernel design for reduced interference distributions, *IEEE Trans. Signal Process.* 42 (1992) 402–412.
- [60] H. Qu, R. Wang, W. Qu, P. Zhao, Research on DOA estimation of multi-component LFM signals based on the FRFT, *Wirel. Sens. Network* 2009 (3) (2009) 171–181.
- [61] Y.D. Zhang, M.G. Amin, B. Himed, Direction-of-arrival estimation of nonstationary signals exploiting signal characteristics, in: *International Conference Information Science Signal Processing Their Applications*, Montreal, Canada, July 2012.
- [62] Y.D. Zhang, M.G. Amin, B. Himed, Altitude estimation of maneuvering targets in MIMO over-the-horizon radar, in: *IEEE Sensor Array and Multichannel Signal Processing Workshop*, Stevens, NJ, June 2012, pp. 261–264.
- [63] H. Wang, M. Kaveh, Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wideband sources, *IEEE Trans. Acoust. Speech Signal Process. ASSP-33* (1985) 823–831.
- [64] B. Friedlander, J. Weiss, Direction finding for wide-band signals using an interpolated array, *IEEE Trans. Signal Process.* 41 (1993) 1618–1634.
- [65] G. Wang, X.-G. Xia, Iterative algorithm for direction of arrival estimation with wideband chirp signals, *IEE Proc. Radar Sonar Navig.* 147 (5) (2000) 233–238.
- [66] E. Fisher, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, R. Valenzuela, MIMO radar: an idea whose time has come, in: *Proceedings of the IEEE Radar Conference*, April 2004, pp. 71–78.
- [67] J. Li, P. Stoica (Eds.), *MIMO Radar Signal Processing*, Wiley-IEEE Press, New York, NY, 2009.
- [68] J. Li, P. Stoica, MIMO radar with colocated antennas, *IEEE Signal Process. Mag.* 25 (5) (2007) 106–114.
- [69] M.L. Bencheikh, Y. Wang, Joint DOD-DOA estimation using combined ESPRIT-MUSIC approach in MIMO radar, *Electron. Lett.* 46 (15) (2010).
- [70] J. Chen, H. Gu, W. Su, A new method for joint DOD and DOA estimation in bistatic MIMO radar, *Signal Process.* 90 (2010) 714–719.
- [71] C. Duofang, C. Baixiao, Q. Guodong, Angle estimation using ESPRIT in MIMO radar, *Electron. Lett.* 44 (12) (2008).
- [72] M. Jin, G. Liao, J. Li, Joint DOD and DOA estimation for bistatic MIMO radar, *Signal Process.* 89 (2009) 244–251.
- [73] N.J. Willis, H.D. Griffiths (Eds.), *Advances in Bistatic Radar*, SciTech Publishing, 2007.
- [74] Y. Zhang, G.J. Frazer, M.G. Amin, Concurrent operation of two over-thehorizon radars, *IEEE J. Sel. Topics Signal Process.* 1 (1) (2007) 114–123.

- [75] A. Hassani, S.A. Vorobyov, A.B. Gershman, Moving target parameters estimation in noncoherent MIMO radar systems, *IEEE Trans. Signal Process.* 60 (5) (2012) 2354–2361.
- [76] R. Roy, T. Kailath, ESPRIT-estimation of signal parameters via rotational invariance techniques, *IEEE Trans. Acoust. Speech Signal Process. ASSP-37* (7) (1989) 984–995.

# Source Localization and Tracking

# 18

Yu Hen Hu

Department of Electronics and Communication Engineering, University of Wisconsin-Madison, Madison, WI, USA

## 3.18.1 Introduction

In this chapter, the task of source localization and tracking will be discussed. The goal of source localization is to estimate the location of one or more events (e.g., earthquake) or targets based on signals emitted from these locations and received at one or more sensors. It is often assumed that the *cross section* of the target or the event is very small compared to the spread of the sensors and hence the *point source* assumption is valid. If source locations move with respect to time, then *tracking* will be performed to facilitate accurate forecasting of future target positions. Diverse applications of source localizations have been found, such as Global Positioning System (GPS) [1,2], sonar [3,4], radar [5,6], seismic event localization [7–9], brain imaging [10], teleconference [11,12], wireless sensor networks [13–19], among many others.

The basic approach of source localization is *geometric triangulation*: Given the distance or angles between the unknown source location and known reference positions, the coordinates of the source locations can be computed algebraically. However, these distance or angle measurements often need to be inferred indirectly from a *signal propagation model* that describes the received source signal as a function of distance and incidence angle between the source and the sensor. Such a model facilitates statistical estimation of spatial locations of the sources based on features such as time of arrival, attenuation of signal intensity, or phase lags. Specific features that may be exploited are also dependent on the specific signal modalities such as acoustic signal, radio waves, seismic vibrations, infra-red light, or visible light. In this chapter, various source signal propagation models will be reviewed, and statistical inference methods will be surveyed.

Very often, source localization will be performed consecutively to *track* a moving source over time. Using a *dynamic model* to describe the movement of the source, one may *predict* the probability distribution function (*pdf*) of source location using previously received source signals. With this estimated *prior distribution* of source location, Bayesian estimation may be applied to *update* the source location using current sensory measurements. Thus, source localization and tracking are often intimately related.

In the remaining of this chapter, the basic idea of triangulation will first be reviewed. Next, the signal propagation and channel models will be introduced. Statistical methods that implicitly or explicitly leverage the triangulation approach to estimate source locations will then be presented. Bayesian tracking methods such as Kalman filter will also be briefly surveyed.

### 3.18.2 Problem formulation

Assume  $N$  sensors are deployed over a sensing field with known positions. Specifically, the position of the  $n$ th sensor is denoted by  $\mathbf{x}_n$ . It is also assumed that there are  $K$  targets in the sensing field with *unknown* locations  $\{\mathbf{r}_k; 1 \leq k \leq K\}$ . Each target is emitting a source signal denoted by  $s_k(t)$  at time  $t$ . The  $n$ th sensor will receive a delayed, and sometimes distorted version of the  $k$ th source signal  $y_{n,k}(t)$ . In general,  $y_{n,k}(t)$  is a function of the past source signal up to time  $t$ ,  $\{s_k(t-m); 1 \leq k \leq K, m = 0, 1, \dots\}$ , the sensor locations  $\mathbf{x}_n$ , the target locations  $\mathbf{r}_k$ , as well as the propagation medium of the signal. This *source signal propagation model* will be discussed in a moment. It is noted that prior to target localization, a target detection task must be performed and the presence of  $K$  targets within the sensing field must have been estimated. The issues of target detection and target number estimation will not be included in this discussion.

We further assume that the measurements at the  $n$ th sensor, denoted by  $y_n(t)$  is a superimposition of  $\{y_{n,k}(t); 1 \leq k \leq K\}$ . That is,

$$y_n(t) = \sum_{k=1}^K y_{n,k}(t) + e_n(t), \quad (18.1)$$

where  $e_n(t)$  is the observation noise at the  $n$ th sensor. The objective of *source localization* is to estimate the source locations  $\{\mathbf{r}_k; 1 \leq k \leq K\}$  based on sensor readings  $\{y_n(t-\ell); 1 \leq n \leq N, \ell = 0, 1, \dots\}$  given known sensor positions  $\{\mathbf{x}_n; 1 \leq n \leq N\}$  and the number of targets  $K$ .

### 3.18.3 Triangulation

In a sensor network source localization problem setting, the sensor locations are the reference positions. Based on signals received at sensors from the sources, two types of measurements may be inferred: (i) distance between each sensor and each source and (ii) incidence angle of a wavefront of the source signal impinged upon a sensor relative to an absolute reference direction (e.g., north). In this section, we will derive three sets of formula that make use of (a) distance only, (b) angle only, and (c) distance and angle to deduce the source location. For convenience, the single source situation will be used, with discussions on potential generalization to multiple sources.

#### 3.18.3.1 Distance based triangulation

Denote  $d_n$  to be the Euclidean distance between the position of the  $n$ th sensor  $\mathbf{x}_n$  and the (unknown) source location  $\mathbf{r}$ , namely,  $d_n = \|\mathbf{x}_n - \mathbf{r}\|$ .  $d_n$  may be estimated from the sensor observations  $y_n(t)$  for certain type of sensors such as laser or infrared light. One may write down  $N$  quadratic equations:

$$d_n^2 = \|\mathbf{x}_n - \mathbf{r}\|^2 = \|\mathbf{x}_n\|^2 + \|\mathbf{r}\|^2 - 2\mathbf{x}_n^T \mathbf{r}, \quad 1 \leq n \leq N. \quad (18.2)$$

Subtracting both sides of each pair of successive equations above to eliminate the unknown term  $\|\mathbf{r}\|^2$ , it leads to  $N - 1$  linear equations

$$(\mathbf{x}_{n+1} - \mathbf{x}_n)^T \mathbf{r} = \frac{1}{2} \left[ \left( \|\mathbf{x}_{n+1}\|^2 - \|\mathbf{x}_n\|^2 \right) - \left( d_{n+1}^2 - d_n^2 \right) \right], \quad 1 \leq n \leq N - 1. \quad (18.3)$$

These  $N - 1$  linear systems of equations may be expressed in a matrix form:

$$\mathbf{Xr} = \mathbf{d}. \quad (18.4)$$

In general, the accuracy of the distance estimate may be compromised by estimation errors. Thus, a more realistic model should include an estimation noise term:

$$\mathbf{Xr} = \mathbf{d} + \mathbf{e}, \quad (18.5)$$

where  $\mathbf{e}$  is a zero-mean, uncorrelated random noise vector such that

$$E\{\mathbf{e}\} = \mathbf{0}, \quad \text{cov}\{\mathbf{e}\} = \Sigma = \text{diag}\left\{\sigma_1^2, \dots, \sigma_N^2\right\}.$$

Then  $\mathbf{r}$  may be estimated using weighted least square method that seeks to minimize a cost function

$$\left\|(\mathbf{Xr} - \mathbf{d})^T \Sigma^{-1} (\mathbf{Xr} - \mathbf{d})\right\|^2.$$

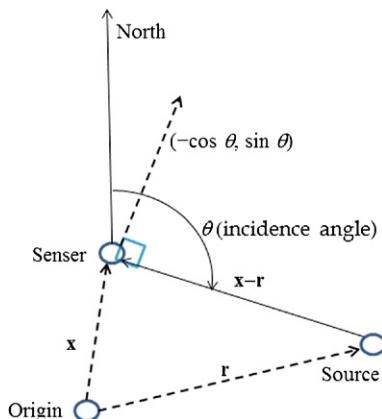
This leads to

$$\hat{\mathbf{r}}_{Dis, WLS} = \left(\mathbf{X}^T \Sigma^{-1} \mathbf{X}\right)^{-1} \mathbf{X}^T \Sigma^{-1} \mathbf{d}. \quad (18.6)$$

### 3.18.3.2 Angle based triangulation

Denote  $\theta_n$  to be *incidence angle* from the source into the  $n$ th sensor using the north direction as the reference direction. We further assume that the angle is positive along the clock-wise direction. Referring to Figure 18.1, it is easily verified that

$$0 = [-\cos \theta_n \ \sin \theta_n] (\mathbf{x}_n - \mathbf{r}). \quad (18.7)$$



**FIGURE 18.1**

Geometric positions for Triangulation.

Rearranging terms and collecting all  $N$  equations, one has

$$\underbrace{\begin{bmatrix} -\cos \theta_1 & \sin \theta_1 \\ -\cos \theta_2 & \sin \theta_2 \\ \vdots & \vdots \\ -\cos \theta_N & \sin \theta_N \end{bmatrix}}_{\mathbf{C}} \cdot \mathbf{r} = \underbrace{\begin{bmatrix} [-\cos \theta_1 \sin \theta_1] \mathbf{x}_1 \\ [-\cos \theta_2 \sin \theta_2] \mathbf{x}_2 \\ \vdots \\ [-\cos \theta_N \sin \theta_N] \mathbf{x}_N \end{bmatrix}}_{\mathbf{p}}. \quad (18.8)$$

Or, in matrix formation:

$$\mathbf{C} \cdot \mathbf{r} = \mathbf{p}. \quad (18.9)$$

Similar to Eq. (18.5), the observation  $\mathbf{p}$  may be contaminated with noise. As such, the source location may be obtained via a weighted least square estimate. For the sake of notation simplicity, one may use  $\mathbf{e}$  to denote the noise vector as in Eq. (18.5). Then, the weighted least square estimation of  $\mathbf{r}$  becomes:

$$\hat{\mathbf{r}}_{Ang, WLS} = (\mathbf{C}^T \Sigma^{-1} \mathbf{C})^{-1} \mathbf{C}^T \Sigma^{-1} \mathbf{p}. \quad (18.10)$$

### 3.18.3.3 Triangulation: generalizations

In Sections 3.18.3.1 and 3.18.3.2, single target triangulation localization algorithms for distance measurements and incidence angle measurements have been discussed. Both of these situations lead to an over-determined linear systems of Eqs. (18.5) and (18.9). Therefore, when both distance measurements and incidence angle measurements are available, these two equations may be combined and solved jointly.

When there are two or more sources (targets), a number of issues will need to be addressed. First of all, when a sensor receives signals emitted from two or more sources simultaneously, it may not be able to distinguish one signal from another if these signals overlap in time, space and frequency domains. This is the traditional *signal (source) separation problem* [20–23]. Secondly, even individual sources may be separated by individual sensors, which of the signals received by different sensors correspond to the same source is not always easy to tell. This is the so called *data association problem* [24,25]. A comprehensive survey of these two issues is beyond the scope of this chapter.

As discussed earlier, while localization may be accomplished via triangulation, the measurements of sensor to source distance or source to sensor incidence angle must be estimated based on received signals. The mathematical model of the received source signals, known as the signal propagation model will be surveyed in the next section.

---

## 3.18.4 Signal propagation models

Depending on specific applications, in many occasions, the source signal may be modeled as a narrow band signal characterized by a single sinusoid:

$$s_k(t) = A_k \exp(j2\pi f_k t + \phi_k), \quad (18.11)$$

where  $A_k$  is the amplitude,  $f_k$  is the frequency, and  $\phi_k$  is the phase. While for certain special cases that all or a portions of these parameters are known, in most applications, they are assumed unknown.

In other occasions, the source signal may be a broad band signal which contains numerous harmonics and is difficult to be expressed analytically.

Between each source-sensor pair there is a communication channel whose characteristics depend on specific medium (air, vacuum, water, etc.). The net effects of the communication channels on the received signal  $y_{n,k}(t)$  can be summarized in the following categories:

**Attenuation:** The amplitude of the source signal often attenuates rapidly as source to sensor distance increases. Hence, examining relative attenuation of signal strength provides an indirect way to estimate source to sensor distance. For a point source, the rate of attenuation is often inversely proportional to  $\|\mathbf{x}_n - \mathbf{r}_k\|$ . The communication channel between the sensor and the source may also be frequency selective such that the attenuation rates are different from different frequency bands. For example, high frequency sound often attenuate much faster than low frequency sound. Thus the amplitude waveform as well as the energy of the source signal at the sensor may be distorted. If the signal propagation is subject to multi-path distortion, the attenuation rate may also be affected. Yet another factor that affects the measured amplitude of received signal is the sensor gain. Before the received analog signal is to be digitized, its magnitude will be amplified with adaptive gain control to ensure the dynamic range of the analog-to-digital converter (ADC) is not saturated. Furthermore, the signal strength attenuation may not be uniform over all directions, and the point source assumption may not be valid at short distance.

**Time delay:** Denote  $v$  to be the signal propagation speed in the corresponding medium, the time for the source signal traveling to the sensor can be evaluated as

$$D_{n,k} = \|\mathbf{x}_n - \mathbf{r}_k\|/v. \quad (18.12)$$

However, the time delay due to signal propagation may be impacted by non-homogeneous mediums. The consequence may be refraction or deflection of signal propagation and the accuracy of time delay estimation may be compromised.

**Phase distortion:** The nonlinear phase distortion due to frequency selective channel property may also distort the morphology of the signal waveform, making it difficult to estimate any phase difference between received signals at different sensors.

**Noise:** While the source signal travels to each sensor, it may suffer from (additive) channel noise or interferences by other sources. It is often assumed that the background noise observed at each sensor is a zero-mean, un-correlated, and wide-sense stationary random process, having Gaussian distribution. Moreover, the background noise processes at different sensors are assumed to be statistically independent to each other. Such assumptions may need to be modified for situations when the background noise also include high energy impulsive noise or interferences.

A commonly used channel model in wireless communication theory is the convolution model that models the frequency selective characteristics of the channel as a *finite impulse response* digital filter  $\{h_{n,k}(m); 0 \leq m \leq M - 1\}$ . Thus, using the notation defined in this chapter,

$$y_{n,k}(t) = \sum_{m=0}^{M-1} h_{n,k}(m)s_k(t - m) + \epsilon_{n,k}(t). \quad (18.13)$$

The channel parameters are functions of both the sensor and source locations. For source localization application, above model is often simplified to emphasize specific features.

### 3.18.4.0.1 Received signal strength indicator (RSSI)

A popular simplified model concerns only amplitude attenuation:

$$y_{n,k}(t) = \frac{g_n}{\|\mathbf{x}_n - \mathbf{r}_k\|} \cdot s_k(t). \quad (18.14)$$

Here the propagation delay, phase distortion, and noise are all ignored. Instead,  $g_n$  is used to denote the sensor *gain* of the  $n$ th sensor. Thus, one may write

$$y_n(t) = g_n \cdot \sum_{k=1}^K \frac{s_k(t)}{\|\mathbf{x}_n - \mathbf{r}_k\|} + e_n(t). \quad (18.15)$$

Averaging over a short time interval centered at the sampling time  $t$ , one may express the *energy* of the received signal during this short period as

$$Y_n(t) = \frac{1}{T} \int_{t-T/2}^{t+T/2} y_n^2(u) du \simeq G_n \cdot \sum_{k=1}^K \frac{S_k(t)}{\|\mathbf{x}_n - \mathbf{r}_k\|^2} + \zeta_n(t), \quad (18.16)$$

where  $Y_n(t)$  is the received signal energy at time  $t$  at sensor  $n$ ,  $G_n = g_n^2$ , and

$$S_k(t) = \frac{1}{T} \int_{t-T/2}^{t+T/2} [s_k(u)]^2 du.$$

It is assumed that the  $K$  source signals are statistically, mutually independent, such that

$$\frac{1}{T} \int_{t-T/2}^{t+T/2} s_k(u)s_j(u) du \simeq 0, \quad k \neq j.$$

Moreover, the background noise  $e_n(t)$  is also independent to the signal, besides being *i.i.d.* random variables such that

$$\frac{1}{T} \int_{t-T/2}^{t+T/2} s_k(u)e_j(u) du \simeq 0, \quad \forall k, j$$

and

$$\frac{1}{T} \int_{t-T/2}^{t+T/2} e_k(u)e_j(u) du = \begin{cases} \zeta_n(t) & k = j, \\ 0 & k \neq j. \end{cases} \quad (18.17)$$

Here  $\zeta_n(t)$  is a random variable with a  $\chi^2$  distribution. However, as discussed in [19], for practical purposes,  $\zeta_n(t)$  can be modeled as a Gaussian random variable with a positive mean value  $\mu_n > 0$  and variance  $\sigma_n^2$ .

Equations (18.15) or (18.16) are often used in source localization algorithms that are based on *Received Signal Strength Indicator* (RSSI) [26,27] to infer the target location.

### 3.18.4.1 Time delay estimation

Time delay estimation [28,29] has a long history of signal processing applications [30–32]. It is assumed that

$$y_{n,k}(t) = \sum_{m=0}^{M-1} h_{n,k}(t) s_k(t - m - D_{n,k}) + \epsilon_{n,k}(t). \quad (18.18)$$

If  $D_{n,k}$  can be estimated accurately, the source to sensor distance may be estimated using Eq. (18.12). Thus, the key issue is to estimate  $D_{n,k}$ .

If the original source signal  $s_k(t)$  and the received signal  $y_{n,k}(t)$  are both available, then the time delay may be estimated using a number of approaches.

Assume  $s_k(t)$  and  $y_{n,k}(t)$  are both zero-mean white sense stationary random processes over a time interval  $[-T/2, T/2]$ , the cross-correlation between them is defined as

$$R_{s,y}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} s_k(t) y_{n,k}(t + \tau) dt. \quad (18.19)$$

Substituting  $y_{n,k}(t)$  in Eq. (18.18) into Eq. (18.19), and assume  $h_{n,k}(t) = 1$  if  $t = 0$  and  $h_{n,k} = 0$  otherwise, one has

$$\begin{aligned} R_{s,y}(\tau) &= \frac{1}{T} \int_{-T/2}^{T/2} s_k(t) s_k(t - D_{n,k} + \tau) dt + \underbrace{\frac{1}{T} \int_{-T/2}^{T/2} s_k(t) \cdot \epsilon_{n,k}(t) dt}_{=0} \\ &= R_{s,s}(\tau - D_{n,k}) \leq R_{s,s}(0) = \frac{1}{T} \int_{-T/2}^{T/2} |s_k(t)|^2 dt. \end{aligned} \quad (18.20)$$

Therefore, a maximum likelihood estimate of the time delay  $D_{n,k}$  will be

$$\hat{D}_{n,k} = \arg_{\tau} \max R_{s,y}(\tau). \quad (18.21)$$

There is a fundamental difficulty in applying Eq. (18.21) to estimate source to sensor signal propagation delay:  $s_k(t)$  may *not* be available at the sensor. This is often the case when the source is non-cooperative such as an intruder. On the other hand, if the source is cooperative, it may transmit a signal that consists of a time stamp. If the clocks at the sensor and at the source are *synchronized* as in the case of the global positioning system (GPS) [1], the sensor can compare the receiving time stamp against the sending time stamp and deduce the transit time without using cross correlation.

In some applications, the direction of the source is known but the distance between the source and the sensor is to be measured. Then, a *round trip* signal propagation delay may be estimated using cross-correlation method by emitting a signal  $s_k(t)$  from the sensor toward the source and bounding back to the sensor. Then, both the transmitted signal and received signal will be available at the sensor and above cross-correlation method may be applied to estimate the source to sensor distance.

In general, when  $s_k(t)$  is unavailable at the sensor, a *difference* of time of arrival from the same source at different sensors will allow one to estimate the incidence angle of source signal. This is the *Time Difference of Arrival* (TDoA) feature mentioned in literatures [33,34]. With TDoA, a *Generalized Cross Correlation* (GCC) [28] may be applied to estimate  $\delta_{m,n}(k) = D_{m,k} - D_{n,k}$ ,  $m \neq n$ .

Ideally, one would hope  $y_{m,k}(t) = h_{m,k}(0)s_k(t - D_{m,k})$  and  $y_{n,k}(t) = h_{n,k}(0)s_k(t - D_{n,k})$ . As such, one computes the cross correlation

$$\begin{aligned} R_{m,n}(\tau) &= \frac{1}{T} \int_{-T/2}^{T/2} y_{m,k}(t)y_{n,k}(t + \tau)dt \\ &= \frac{1}{T} \int_{-T/2}^{T/2} s_k(t - D_{m,k})s_k(t - D_{n,k} + \tau)dt \\ &= R_{ss}(\tau - (D_{n,k} - D_{m,k})) \leq R_{ss}(0). \end{aligned} \quad (18.22)$$

Thus,

$$\hat{\delta}_{m,n}(k) = D_{m,k} - D_{n,k} = \arg_{\tau} \max R_{m,n}(\tau). \quad (18.23)$$

With the presence of channel noise  $\epsilon_{n,k}(t)$  and channel model  $\{h_{n,k}(t)\}$ , Knapp and Carter [28] proposed to pre-filter  $y_{m,k}(t)$  and  $y_{n,k}(t)$  to improve the accuracy of the TDOA estimate by enhancing the signal to noise ratio (SNR) of  $R_{m,n}(\tau)$  estimate. Specifically, denote  $Y_{m,k}(\omega)$  and  $Y_{n,k}(\omega)$  respectively as the Fourier transform of the received signal  $y_{m,k}(t)$  and  $y_{n,k}(t)$ , and also denote  $W_{m,k}(\omega)$  and  $W_{n,k}(\omega)$  respectively as the spectrum of the pre-filters for  $y_{m,k}(t)$  and  $y_{n,k}(t)$ , the GCC is defined as:

$$R_{m,n}^{\text{GCC}}(\tau) = \frac{1}{2\pi T} \int_{-T/2}^{T/2} W_{m,k}(\omega) W_{n,k}^*(\omega) Y_{m,k}(\omega) Y_{n,k}^*(\omega) \exp(j\omega\tau) d\omega. \quad (18.24)$$

Once  $\hat{\delta}_{m,n}(k)$  is estimated, one has the following relation:

$$\hat{\delta}_{m,n}(k) \cdot v = \|\mathbf{x}_m - \mathbf{r}\| - \|\mathbf{x}_n - \mathbf{r}\|. \quad (18.25)$$

Hence TDOA provides a relative distance measure from the source to two different sensors. We note by passing that Eq. (18.25) defines a parabolic trajectory for potential target location  $\mathbf{r}$ .

### 3.18.4.2 Angle of arrival estimation

For narrow band source signal, if sensors are fully synchronized, phase difference between received sensor signals may be used to estimate the incidence angle of source signal.

Assume a point source emitting a narrow band (single harmonic) signal described in Eq. (18.11). Under a *far field* assumption, that is

$$\max_{m,n} \|\mathbf{x}_m - \mathbf{x}_n\| \ll \min_n \|\mathbf{r} - \mathbf{x}_n\| \quad (18.26)$$

the waveform of the source signal can be modeled as a *plane wave* such that the incidence angle of the source to each of the sensor nodes will be identical. This makes it easier to represent the time difference of arrival as a *steering vector* which is a function of the incidence angle and the sensor array geometry.

Again, let north be a reference direction of the incidence angle  $\theta$  which increases along the clockwise direction. Then a unit vector along the incidence angle can be expressed as  $[-\sin \theta, -\cos \theta]^T$ . The time difference of arrival between sensor nodes  $m$  and  $N$  can be expressed as:

$$\Delta t_{m,N}(\theta) = (\mathbf{x}_m - \mathbf{x}_N)^T \begin{bmatrix} -\sin \theta \\ -\cos \theta \end{bmatrix} / v \quad (18.27)$$

Substitute the expression of a narrow band signal as shown in Eq. (18.11) into the expression of received signal described in Eq. (18.18), one has

$$\begin{aligned} y_{m,k}(t) &= \{h_{m,k}(t)\} * \{s_k(t - D_{m,k})\} + \epsilon_{m,k}(t) \\ &= \{h_{m,k}(t)\} * \{s_k(t - (D_{N,k} + \Delta t_{m,N}(\theta_k)))\} + \epsilon_{m,k}(t) \\ &= H_{m,k}(f_k) \cdot A_k \exp(j2\pi f_k(t - (D_{N,k} + \Delta t_{m,N}(\theta_k))) + \phi_k) + \epsilon_{m,k}(t) \\ &= H_k \exp(-j2\pi f_k \Delta t_{m,N}(\theta_k)) \cdot s_k(t - D_{N,k}) + \epsilon_{m,k}(t), \end{aligned} \quad (18.28)$$

where  $H_{n,k}(f_k) = H_k$  is the frequency response of  $h_{n,k}(t)$  evaluated at  $f = f_k$  and is independent of  $\mathbf{r}_k$ , and  $\mathbf{x}_n$  under the far field assumption Eq. (18.26). Assume  $K = 1$ , then the received signal of all  $N$  sensors may be represented by

$$\mathbf{y}_k(t) = \mathbf{a}_k^H(\theta_k) s_k(t - D_{N,k}) \cdot H_k + \boldsymbol{\epsilon}_k(t), \quad (18.29)$$

where

$$\mathbf{a}_k(\theta_k) = [\exp(-j2\pi f_k \Delta t_{1,N}(\theta_k)) \quad \cdots \quad \exp(-j2\pi f_k \Delta t_{N-1,N}(\theta_k)) \quad 1]^H$$

is a *steering vector* with respect to a narrow band source with incidence angle  $\theta_k$  using the signal received at the  $N$ th sensor as a reference. It represents the phase difference of the received sensor signals with respect to a plane wave traveling at an incidence angle  $\theta_k$ . With  $K$  narrow band sources, the received signal vector than may be expressed as:

$$\mathbf{y}(t) = \sum_{k=1}^K \mathbf{y}_k(t) = \mathbf{A}^H(\theta) \mathbf{s}(t) + \boldsymbol{\epsilon}(t), \quad (18.30)$$

where

$$\mathbf{A}(\theta) = [\mathbf{a}_1(\theta_1) \quad \mathbf{a}_2(\theta_2) \quad \cdots \quad \mathbf{a}_K(\theta_K)]^H,$$

$$\mathbf{s}(t) = [s_1(t - D_{N,1})H_1 \quad s_2(t - D_{N,2})H_2 \quad \cdots \quad s_K(t - D_{N,K})H_K]^H,$$

and

$$\boldsymbol{\epsilon}(t) = \sum_{k=1}^K \boldsymbol{\epsilon}_k(t).$$

Equation (18.30) is the basis of many array signal processing algorithms [3,35,36] such as MUSIC [37]. Specifically, the covariance matrix of  $\mathbf{y}(t)$  may be decomposed into two parts:

$$\mathbf{R}_{yy} = E\{\mathbf{y}\mathbf{y}^H\} = \mathbf{A}(\theta) \mathbf{S} \mathbf{A}^H(\theta) + \sigma^2 \mathbf{I} = \mathbf{R}_s + \mathbf{R}_n, \quad (18.31)$$

where  $\mathbf{R}_s$  and  $\mathbf{R}_n$  are the *signal* and *noise* covariance matrix respectively. In particular,  $\mathbf{R}_s$  has a rank equal to  $K$ . In practice, one would estimate the covariance matrix from received signal and perform eigenvalue decomposition:

$$\widehat{\mathbf{R}}_{yy} = \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{y}(t) \mathbf{y}^H(t) = \mathbf{U}_s \Lambda_s \mathbf{U}_s^H + \mathbf{U}_n \Lambda_n \mathbf{U}_n^H, \quad (18.32)$$

where  $\Lambda_s = \text{diag}\{\lambda_1, \dots, \lambda_K\}$  are the largest  $K$  eigenvalues and columns of  $\mathbf{U}_s$  are corresponding eigenvectors. The  $K$  dimensional subspace spanned by columns of the  $\mathbf{U}_s$  matrix is called the *signal subspace*. On the other hand,  $\Lambda_n = \text{diag}\{\lambda_{K+1}, \dots, \lambda_N\}$  are the remaining  $N - K$  eigenvalues. The  $N - K$  dimensional subspace spanned by columns of the  $\mathbf{U}_n$  matrix is called the *noise subspace*. Define

$$\mathbf{v}_k(\theta) = [\exp(-j2\pi f_k \Delta t_{1,N}(\theta)) \quad \cdots \quad \exp(-j2\pi f_k \Delta t_{N-1,N}(\theta)) \quad 1]^H \quad (18.33)$$

as a generic steering vector, the MUSIC method evaluates the function

$$P_{\text{MUSIC}}(\theta) = \frac{\|\mathbf{v}(\theta)\|^2}{\|\mathbf{v}^H(\theta)\mathbf{U}_n\|^2}. \quad (18.34)$$

This *MUSIC spectrum* then will show high peaks at  $\theta = \theta_k$ ,  $1 \leq k \leq K$ .

### 3.18.5 Source localization algorithms

#### 3.18.5.1 Bayesian source localization based on RSSI

For convenience of discussion, let us rewrite Eq. (18.16) here after a linear transformation of the random variables

$$Z_n(t) = (Y_n(t) - \mu_n)/\sigma_n \quad \text{and} \quad \tilde{\zeta}_n(t) = (\zeta_n(t) - \mu_n)/\sigma_n. \quad (18.35)$$

Then,

$$\begin{bmatrix} Z_1(t) \\ Z_2(t) \\ \vdots \\ Z_N(t) \end{bmatrix} = \begin{bmatrix} G_1/(\sigma_1 \cdot \|\mathbf{x}_1 - \mathbf{r}_1\|^2) & G_1/(\sigma_1 \cdot \|\mathbf{x}_1 - \mathbf{r}_2\|^2) & \cdots & G_1/(\sigma_1 \cdot \|\mathbf{x}_1 - \mathbf{r}_K\|^2) \\ G_2/(\sigma_2 \cdot \|\mathbf{x}_2 - \mathbf{r}_1\|^2) & G_2/(\sigma_2 \cdot \|\mathbf{x}_2 - \mathbf{r}_2\|^2) & \cdots & G_2/(\sigma_2 \cdot \|\mathbf{x}_2 - \mathbf{r}_K\|^2) \\ \vdots & \vdots & \ddots & \vdots \\ G_N/(\sigma_N \cdot \|\mathbf{x}_N - \mathbf{r}_1\|^2) & G_N/(\sigma_N \cdot \|\mathbf{x}_N - \mathbf{r}_2\|^2) & \cdots & G_N/(\sigma_N \cdot \|\mathbf{x}_N - \mathbf{r}_K\|^2) \end{bmatrix} \\ \times \begin{bmatrix} S_1(t) \\ S_2(t) \\ \vdots \\ S_K(t) \end{bmatrix} + \begin{bmatrix} \tilde{\zeta}_1(t) \\ \tilde{\zeta}_2(t) \\ \vdots \\ \tilde{\zeta}_N(t) \end{bmatrix} \quad (18.36)$$

or in matrix notation

$$\mathbf{z} = \mathbf{Hs} + \tilde{\zeta}. \quad (18.37)$$

Since  $\tilde{\zeta}$  is a normalized Gaussian random vector with zero mean and identity matrix as its covariance, the *likelihood function* that  $\mathbf{z}$  is observed at  $N$  sensors, given the  $K$  source signal energy and source locations  $\{\mathbf{r}_k, S_k(t); 1 \leq k \leq K\}$  can be expressed as:

$$L(\mathbf{r}_k, S_k(t); 1 \leq k \leq K) = P\{\mathbf{z}|\mathbf{r}_k, S_k(t); 1 \leq k \leq K\} \propto \exp\left\{-\frac{1}{2}(\mathbf{z} - \mathbf{Hs})^T(\mathbf{z} - \mathbf{Hs})\right\}. \quad (18.38)$$

A maximum likelihood (ML) estimate of  $\{\mathbf{r}_k, S_k(t); 1 \leq k \leq K\}$  may be obtained by maximizing  $L$  or equivalently, minimizing the *negative log likelihood function*

$$\ell(\mathbf{r}_k, S_k(t); 1 \leq k \leq K) = -\log L = \|\mathbf{z} - \mathbf{Hs}\|^2. \quad (18.39)$$

Setting the gradient of  $\ell$  against  $\mathbf{s}$  equal to 0, one may express the estimate of the source signal energy vector as

$$\hat{\mathbf{s}} = \mathbf{H}^\dagger \mathbf{z}, \quad (18.40)$$

where  $\mathbf{H}^\dagger$  is the pseudo inverse of the  $\mathbf{H}$  matrix. Substituting Eq. (18.40) into Eq. (18.39), one has

$$\ell(\mathbf{r}_k; 1 \leq k \leq K) = \|(\mathbf{I} - \mathbf{HH}^\dagger)\mathbf{z}\|^2. \quad (18.41)$$

Equation (18.41) is significant in several ways: (a) The number of unknown parameters is reduced from  $3K$  to  $2K$  assuming that the dimension of  $\mathbf{r}_k$  is 2. (b) To minimize  $\ell$ , the source locations should be chosen such that the (normalized) energy vector  $\mathbf{z}$  falls within the subspace spanned by columns of the  $\mathbf{H}$  matrix as close as possible. In [19], this property is leveraged to derive a multi-resolution projection method for solving the source locations.

In deriving the ML estimates of source locations, no prior information about source locations is used. If, as in a tracking scenario, the prior probability of source locations,  $p(\mathbf{r}_k; 1 \leq k \leq K)$  is available, then the *a posterior* probability may be expressed as:

$$P\{\mathbf{r}_k, S_k(t); 1 \leq k \leq K | \mathbf{z}\} \propto p(\mathbf{r}_k; 1 \leq k \leq K) \cdot \exp\left\{-\frac{1}{2}(\mathbf{z} - \mathbf{Hs})^T(\mathbf{z} - \mathbf{Hs})\right\}. \quad (18.42)$$

Maximizing above expression then will lead to the Bayesian estimate of the source locations and corresponding source emitted energies during  $[t - T/2, t + T/2]$ .

### 3.18.5.2 Non-linear least square source localization using RSSI

Assume a scenario of a single source ( $K = 1$ ). If one ignores the noise energy term, Eq. (18.16) can be expressed as (using notation in Eq. (18.35))

$$\|\mathbf{x}_n - \mathbf{r}\|^2 = S \cdot \frac{G_n}{\sigma_n \cdot Z_n(t)} \quad 1 \leq n \leq N. \quad (18.43)$$

Based on this approximated distance, the source location  $\mathbf{r}$  and the source energy  $S$  may be estimated.

#### 3.18.5.2.1 Nonlinear quadratic optimization

Based on Eq. (18.43), one evaluate the ratio

$$\frac{\|\mathbf{x}_n - \mathbf{r}\|^2}{\|\mathbf{x}_m - \mathbf{r}\|^2} = \frac{G_n}{G_m} \cdot \frac{\sigma_m \cdot Z_m(t)}{\sigma_n \cdot Z_n(t)} = \kappa_{n,m}^2, \quad n \neq m. \quad (18.44)$$

After simplification, above equation can be simplified as

$$\|\mathbf{r} - \mathbf{c}_{m,n}\| = \rho_{m,n} \quad \text{where} \quad \mathbf{c}_{m,n} = \frac{\mathbf{x}_n - \kappa_{m,n}^2 \mathbf{x}_m}{1 - \kappa_{m,n}^2} \quad \text{and} \quad \rho_{m,n} = \frac{\kappa_{m,n} \|\mathbf{x}_n - \mathbf{x}_m\|}{1 - \kappa_{m,n}^2}. \quad (18.45)$$

For convenience, one may set  $m = n + 1$  and writes  $\mathbf{c}_n$ , and  $\kappa_n$  instead of  $\mathbf{c}_{m,n}$ , and  $\kappa_{m,n}$ . The target location may be solved by minimizing a nonlinear cost function:

$$J(\mathbf{r}) = \sum_{n=1}^{N-1} (\|\mathbf{r} - \mathbf{c}_n\| - \rho_n)^2. \quad (18.46)$$

In Eq. (18.46), it is assumed that  $\kappa_n \neq 1$ . It can easily be updated to deal the situation when  $\kappa_n \rightarrow 1$ .

### 3.18.5.2.2 Least square solution

If one ignores the background noise, Eq. (18.45) can be seen as a distance measurement of the unknown source location  $\mathbf{r}$ . Recall the distance based triangulation method described in Section 3.18.3.1. One may consider squaring both sides of Eq. (18.45) using sensor readings of sensor indices  $n$ ,  $n+1$ , and  $m$ ,  $m+1$ :

$$\begin{aligned} \|\mathbf{r} - \mathbf{c}_{n,n+1}\|^2 &= \rho_{n,n+1}^2 \Rightarrow \|\mathbf{r}\|^2 = 2\mathbf{r}^T \mathbf{c}_{n,n+1} + \rho_{n,n+1}^2 - \|\mathbf{c}_{n,n+1}\|^2, \\ \|\mathbf{r} - \mathbf{c}_{m,m+1}\|^2 &= \rho_{m,m+1}^2 \Rightarrow \|\mathbf{r}\|^2 = 2\mathbf{r}^T \mathbf{c}_{m,m+1} + \rho_{m,m+1}^2 - \|\mathbf{c}_{m,m+1}\|^2. \end{aligned}$$

After simplification, one has

$$(\mathbf{c}_{n,n+1} - \mathbf{c}_{m,m+1})^T \mathbf{r} = \rho_{m,m+1}^2 - \rho_{n,n+1}^2 + \|\mathbf{c}_{n,n+1}\|^2 - \|\mathbf{c}_{m,m+1}\|^2. \quad (18.47)$$

Combining above equations for different indices of  $m, n$ , one may solve for  $\mathbf{r}$  by solving an over-determined linear system

$$\mathbf{C} \mathbf{r} = \mathbf{h} \Rightarrow \hat{\mathbf{r}}_{LS} = \mathbf{C}^\dagger \mathbf{h}. \quad (18.48)$$

### 3.18.5.2.3 Source localization using table look-up

Consider a single target ( $K = 1$ ) scenario. The normalized sensor observation vector  $\mathbf{z}(t)$  can be regarded as a signature vector of a source at location  $\mathbf{r}$ . Hence, by collecting the corresponding signature vectors for all possible locations  $\mathbf{r}$  in a sensing field, the source location may be estimated using table look-up method. To account for variations of source intensity, the signature vector may be normalized to have a unity norm.

### 3.18.5.3 Source localization using time difference of arrival

Denote

$$d_N = \|\mathbf{x}_N - \mathbf{r}\|. \quad (18.49)$$

Substituting Eq. (18.49) into Eq. (18.25) with  $m = N$ , one has

$$\|\mathbf{r} - \mathbf{x}_n\| = \delta_n \cdot v + d_N, \quad n = 1, 2, \dots, N-1. \quad (18.50)$$

Squaring both sides of above equation for both indices  $m$  and  $n$ ,

$$\begin{aligned} \|\mathbf{r}\|^2 + \|\mathbf{x}_m\|^2 - 2\mathbf{x}_m^T \mathbf{r} &= (\delta_m \cdot v)^2 + d_N^2 + 2d_N \delta_m v, \\ \|\mathbf{r}\|^2 + \|\mathbf{x}_n\|^2 - 2\mathbf{x}_n^T \mathbf{r} &= (\delta_n \cdot v)^2 + d_N^2 + 2d_N \delta_n v. \end{aligned}$$

Subtracting both sides of above equations, it yields (for  $1 \leq m, n \leq N - 1$ )

$$(\mathbf{x}_n - \mathbf{x}_m)^T \mathbf{r} + (\delta_n - \delta_m) \cdot v \cdot d_N = \frac{1}{2} \left\{ \|\mathbf{x}_n\|^2 - \|\mathbf{x}_m\|^2 + (\delta_m^2 - \delta_n^2) \cdot v^2 \right\} = p_{n,m}.$$

Restricting  $m = n + 1$ , one may express above into a matrix format:

$$\begin{bmatrix} (\mathbf{x}_1 - \mathbf{x}_2)^T & (\delta_1 - \delta_2) \cdot v \\ (\mathbf{x}_2 - \mathbf{x}_3)^T & (\delta_2 - \delta_3) \cdot v \\ \dots & \dots \\ (\mathbf{x}_{N-1} - \mathbf{x}_N)^T & (\delta_{N-1} - \delta_N) \cdot v \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ d_N \end{bmatrix} = \begin{bmatrix} p_{1,2} \\ p_{2,3} \\ \vdots \\ p_{N-1,N} \end{bmatrix}. \quad (18.51)$$

The source location  $\mathbf{r}$  may be solved from above equation using least square estimate subject to the constraint quadratic equality constraint according to Eq. (18.49):

$$[\mathbf{r}^T \ d_N] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ d_N \end{bmatrix} + [-2\mathbf{x}_N^T \ 0] \begin{bmatrix} \mathbf{r} \\ d_N \end{bmatrix} + \|\mathbf{x}_N\|^2 = 0. \quad (18.52)$$

### 3.18.5.4 Source localization using angle of arrival

When there is only a single source in the sensing field, the angle based triangulation method discussed in Section 3.18.3.2 can be applied to estimate the source location. With more than one sources, a data correspondence problem must be resolved. Such a problem has been addressed in terms of *N-Ocular stereo* [38].

Assume that each of the  $n$ th sensor detects  $K$  distinct incidence angles from the  $K$  sources. Thus, there are  $N \cdot K$  incidence angles  $\{\theta_{n,k}; 1 \leq n \leq N, 1 \leq k \leq K\}$ . If at the  $n$ th sensor, it receives a source signal with incidence angle  $\theta_{n,k}$ , then the source location may be expressed as

$$\mathbf{r}_k = \mathbf{x}_n + \alpha \cdot \mathbf{u}(\theta_{n,k}) = \mathbf{x}_n + \alpha \cdot \begin{bmatrix} \sin \theta_{n,k} \\ \cos \theta_{n,k} \end{bmatrix}, \quad \alpha > 0. \quad (18.53)$$

Similarly, for the  $m$ th sensor, for the  $\ell$ th incidence angle, the source location is at

$$\mathbf{r}_\ell = \mathbf{x}_m + \beta \cdot \mathbf{u}(\theta_{m,\ell}) \quad \beta > 0.$$

Therefore, if these two sources are the same source (e.g.,  $\mathbf{r}_k = \mathbf{r}_\ell$ ), then one must have a valid solution, namely,  $\alpha > 0$ , and  $\beta > 0$  to the following linear system of equations:

$$\mathbf{x}_m - \mathbf{x}_n = \alpha \mathbf{u}(\theta_{n,k}) - \beta \mathbf{u}(\theta_{m,\ell}) = \begin{bmatrix} \sin \theta_{n,k} & \sin \theta_{m,\ell} \\ \cos \theta_{n,k} & \cos \theta_{m,\ell} \end{bmatrix} \begin{bmatrix} \alpha \\ -\beta \end{bmatrix}. \quad (18.54)$$

The solution can be represented expressively as:

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} -\cos \theta_{m,\ell} & \sin \theta_{m,\ell} \\ -\cos \theta_{n,k} & \sin \theta_{n,k} \end{bmatrix} \cdot (\mathbf{x}_m - \mathbf{x}_n). \quad (18.55)$$

If both  $\alpha$  and  $\beta$  are positive, then  $\mathbf{x}_n + \alpha \mathbf{u}(\theta_{n,k})$  is a potential source location. Since it is derived from observation of two sensors, it is called a *bi-ocular* solution. Otherwise, the solution will be discarded. Substituting the  $K^2$  pairs angles  $\{(\theta_{n,k}, \theta_{m,\ell}); 1 \leq k, \ell \leq K\}$  into Eq. (18.55), one may obtain up to  $K^2$

valid solutions which are candidates for the  $K$  possible source locations, assuming there is no occlusion. The collection of these solutions will be denoted by  $P(2)$  indicating they are consistent with two sensor observations.

Now for a third sensor at  $\mathbf{x}_q$  with incidence angles  $\{\theta_{q,k}; 1 \leq k \leq K\}$ , one may test if any of the potential solutions in  $P(2)$ ,  $\mathbf{r}$ , lies in any of the  $K$  incidence rays originated from  $\mathbf{x}_q$ . This is easily accomplished by evaluating

$$\mathbf{u}^T(\theta_{q,k})(\mathbf{r} - \mathbf{x}_q) = \mathbf{u}^T(\theta_{q,k}) \cdot \{\alpha \mathbf{u}(\theta_{q,k})\} = \alpha. \quad (18.56)$$

If the resulting  $\alpha > 0$ , then the candidate solution  $\mathbf{r}$  will be promoted into a *tri-ocular* solution set  $P(3)$  for being consistent with 3 sensor observations. Repeat above procedure until  $P(N)$  is obtained or when the size of the solution set reduces to  $K$ . Then the procedure is terminated and the  $K$  source positions are obtained.

Two issues will need to be addressed when applying above procedures: (a) A *false matching* solution may be obtained where many incidence rays intersect but no source exists. Fortunately, each incidence ray passing through a false matching position should have two or more matching points. Thus, a false matching solution may be identified if every incidence rays passing through it have more than one matching point. (b) The azimuth incidence angle estimates may be inaccurate. Hence the intersections may not coincide at the same position. Several remedies of this problem have been discussed in [38]. In practice, one may partition the sensing field into mesh grids whose size roughly equal to the angle estimation accuracy. The position of a potential  $N$ -ocular solution then will be registered with the corresponding mesh grid rather than a specific point. A mesh grid will be included into  $P(m)$  if there are  $m$  intersections fall within its range.

### 3.18.6 Target tracking algorithm

So far, in this chapter, source localization is performed based solely on a single snapshot at time  $t$  of sensor readings at  $N$  sensors in the sensing field. If past sensor readings about the same sources can be incorporated, the source location estimates are likely to be much more accurate.

#### 3.18.6.1 Dynamic and observation models

Statistically, tracking is modeled as a *sequential Bayesian estimation* problem of a dynamic system whose states obey a Markov model. In the context of source localization and tracking, the dynamic system that describes a single target moving in a 2D plane has the form

$$\begin{aligned} \begin{bmatrix} \mathbf{r}(t) \\ \dot{\mathbf{r}}(t) \end{bmatrix} &= \mathbf{z}(t) = \mathbf{F}\mathbf{z}(t-1) + \mathbf{G}\mathbf{w}(t) \\ &= \begin{bmatrix} 1 & 0 & T_0 & 0 \\ 0 & 1 & 0 & T_0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}(t-1) \\ \dot{\mathbf{r}}(t-1) \end{bmatrix} + \begin{bmatrix} T_0^2/2 & 0 \\ T_0 & 0 \\ 0 & T_0^2/2 \\ 0 & T_0 \end{bmatrix} \mathbf{w}(t), \end{aligned} \quad (18.57)$$

where  $T_0$  is the time duration between  $t$  and  $t-1$ , and  $\mathbf{w}(t)$  is a zero mean Gaussian random number with variance  $\sigma_w^2$  which models the acceleration.

Signal received at  $N$  sensors as a function of source location  $\mathbf{r}$  and source signal  $s(t)$  gives the *observation model*. Examples of observation model include Eqs. (18.15) and (18.18). These nonlinear models may be expressed as:

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{z}(t)) + \mathbf{v}(t), \quad (18.58)$$

where the *observation noise*  $\mathbf{v}(t)$  is a zero mean Gaussian random variable with variance  $\sigma_v^2$ . Sometimes, intermediate observation model, such as sensor to source distance estimate Eq. (18.12), Eq. (18.25), or source signal incidence angle estimate such as Eq. (18.53).

In general, the sensor observation  $\mathbf{y}(t)$ , or the intermediate measurements are highly nonlinear equations of the source position and speed  $\mathbf{z}(t)$ . Alternatively, one may apply methods discussed in this chapter to estimate  $\mathbf{z}(t)$  based only on current observation  $\mathbf{y}(t)$ , and express a derived observation equation as:

$$\mathbf{y}(t) = \mathbf{H} \cdot \mathbf{z}(t) + \mathbf{v}(t) = [\mathbf{I} \ \mathbf{0}] \mathbf{z}(t) + \mathbf{v}(t). \quad (18.59)$$

### 3.18.6.2 Sequential Bayesian estimation

The state transition described in Eqs. (18.57) and (18.58) can be described by a Markov chain model such that

$$P\{\mathbf{z}(t)|\mathbf{z}(t-1), \dots, \mathbf{z}(0)\} = P\{\mathbf{z}(t)|\mathbf{z}(t-1)\}$$

and

$$P\{\mathbf{y}(t)|\mathbf{z}(t), \mathbf{z}(t-1), \dots, \mathbf{z}(0)\} = P\{\mathbf{y}(t)|\mathbf{z}(t)\}.$$

As such, it is easily verified that

$$P\{\mathbf{z}(t), \dots, \mathbf{z}(0); \mathbf{y}(t), \dots, \mathbf{y}(1)\} = P\{\mathbf{z}(0)\} \cdot \prod_{m=1}^t P\{\mathbf{y}(m)|\mathbf{z}(m)\} \cdot P\{\mathbf{z}(m)|\mathbf{z}(m-1)\}. \quad (18.60)$$

Denote  $\mathbf{Y}(t) = \{\mathbf{y}(t), \mathbf{y}(t-1), \dots, \mathbf{y}(0)\}$  to be the observations up to time  $t$ , and  $P\{\mathbf{z}(t)|\mathbf{Y}(t)\}$  to be the conditional probability of  $\mathbf{z}(t)$  given  $\mathbf{Y}(t)$ . Given the state estimation (location and speed of the source) at previous time step  $P\{\mathbf{z}(t-1)|\mathbf{Y}(t-1)\}$ , the dynamic model Eq. (18.57) allows the *prediction* of the location and speed of the source at time  $t$ :

$$P\{\mathbf{z}(t)|\mathbf{Y}(t-1)\} = \int P\{\mathbf{z}(t)|\mathbf{z}(t-1)\} P\{\mathbf{z}(t-1)|\mathbf{Y}(t-1)\} d\mathbf{z}(t-1), \quad (18.61)$$

where

$$P\{\mathbf{z}(t)|\mathbf{z}(t-1)\} \sim \mathcal{N}\left(\mathbf{F} \cdot \mathbf{z}(t-1), \sigma_w^2 \mathbf{G} \mathbf{G}^T\right) \quad (18.62)$$

has a normal distribution. Similarly,

$$P\{\mathbf{y}(t)|\mathbf{z}(t)\} \sim \mathcal{N}\left(\mathbf{h}(\mathbf{z}(t)), \sigma_v^2 \mathbf{I}\right). \quad (18.63)$$

Applying Bayesian rule, one has

$$\begin{aligned} P\{\mathbf{z}(t)|\mathbf{Y}(t)\} &= \frac{P\{\mathbf{z}(t)|\mathbf{y}(t)\} P\{\mathbf{z}(t)|\mathbf{Y}(t-1)\}}{P\{\mathbf{y}(t)|\mathbf{Y}(t-1)\}} \\ &= \frac{P\{\mathbf{z}(t)|\mathbf{y}(t)\} P\{\mathbf{z}(t)|\mathbf{Y}(t-1)\}}{\int P\{\mathbf{y}(t)|\mathbf{z}(t)\} P\{\mathbf{z}(t)|\mathbf{Y}(t-1)\} d\mathbf{z}(t)}. \end{aligned} \quad (18.64)$$

### 3.18.6.3 Kalman filter

Based on the sequential Bayesian formulation, one may deduce the well-known Kalman filter for the linear observation model (Eq. (18.59)). Specifically, the Kalman filter computes the mean and covariance matrix of the probability distribution

$$P\{\mathbf{z}(t)|\mathbf{Y}(t)\} \sim \mathcal{N}(\hat{\mathbf{z}}(t), \mathbf{P}(t)),$$

where

$$\hat{\mathbf{z}}(t) = E\{P\{\mathbf{z}(t)|\mathbf{Y}(t)\}\} \quad \text{and} \quad \mathbf{P}(t) = E\left\{(\mathbf{z}(t) - \hat{\mathbf{z}}(t))(\mathbf{z}(t) - \hat{\mathbf{z}}(t))^T\right\}.$$

#### 3.18.6.3.1 Prediction phase

Given  $\hat{\mathbf{z}}(t-1)$ , the dynamic Eq. (18.57) allows one to predict the a priori estimate of the current state:

$$\mathbf{z}(t|t-1) = \mathbf{F} \cdot \hat{\mathbf{z}}(t-1). \quad (18.65)$$

Hence,

$$\mathbf{z}(t) - \mathbf{z}(t|t-1) = \mathbf{F} \cdot \mathbf{z}(t-1) + \mathbf{w}(t) - \mathbf{F} \cdot \hat{\mathbf{z}}(t-1) = \mathbf{F}(\mathbf{z}(t-1) - \hat{\mathbf{z}}(t-1)).$$

The corresponding prediction error covariance matrix is:

$$\begin{aligned} \mathbf{P}(t|t-1) &= E\left\{(\mathbf{z}(t) - \mathbf{z}(t|t-1))(\mathbf{z}(t) - \mathbf{z}(t|t-1))^T\right\} \\ &= \mathbf{F}\mathbf{P}(t-1)\mathbf{F}^T + \sigma_w^2\mathbf{I}. \end{aligned} \quad (18.66)$$

Equations (18.65) and (18.66) constitute the *prediction phase* of a Kalman filter.

#### 3.18.6.3.2 Update phase

Given the predicted source position and speed  $\mathbf{z}(t|t-1)$ , each sensor may compute an expected received signal  $\mathbf{y}(t|t-1) = \mathbf{H} \cdot \mathbf{z}(t|t-1)$  and the corresponding innovation (*prediction error*) when compared to the actual received signal  $\mathbf{y}(t)$  as

$$\tilde{\mathbf{y}}(t) = \mathbf{y}(t) - \mathbf{y}(t|t-1) = \mathbf{y}(t) - \mathbf{H} \cdot \mathbf{z}(t|t-1). \quad (18.67)$$

The corresponding covariance matrix then is

$$\mathbf{Q}(t) = E\left\{\tilde{\mathbf{y}}(t)\tilde{\mathbf{y}}^T(t)\right\} = \mathbf{H}\mathbf{P}(t|t-1)\mathbf{H}^T + \sigma_v^2\mathbf{I}. \quad (18.68)$$

Applying least square principle, the optimal *Kalman gain* matrix may be expressed as:

$$\mathbf{K}(t) = \mathbf{P}(t|t-1)\mathbf{H}^T\mathbf{Q}^{-1}(t). \quad (18.69)$$

Finally, the update equation of the state estimation is:

$$\hat{\mathbf{z}}(t) = \mathbf{z}(t|t-1) + \mathbf{K}(t)\tilde{\mathbf{y}}(t) = (\mathbf{I} - \mathbf{K}(t)\mathbf{H})\mathbf{z}(t|t-1) + \mathbf{K}(t)\mathbf{y}(t). \quad (18.70)$$

In other words, the optimal estimate of the source location and speed is a linear combination of the predicted location and speed and a correction term based on sensor observations. Moreover, the covariance matrix of estimation error will also be updated:

$$\mathbf{P}(t) = (\mathbf{I} - \mathbf{K}(t)\mathbf{H})\mathbf{P}(t|t-1). \quad (18.71)$$

### 3.18.6.3.3 Nonlinear observation model

The Kalman filter tracking equations developed so far is based on the linear observation model Eq. (18.59). It facilitates the close-form expression of the prediction error covariance matrix Eq. (18.68). However, as discussed earlier, many practical observation models are non-linear in nature. A number of techniques have been developed to deal with this challenge.

With an *extended Kalman filter* (EKF), the nonlinear observation model will be replaced by an approximated linear model such that

$$\begin{aligned}\mathbf{Q}(t) &= \widehat{\mathbf{H}}\mathbf{P}(t|t-1)\widehat{\mathbf{H}}^T + \sigma_v^2\mathbf{I}, \\ \mathbf{K}(t) &= \mathbf{P}(t|t-1)\widehat{\mathbf{H}}^T\mathbf{Q}^{-1}(t), \\ \mathbf{P}(t) &= (\mathbf{I} - \mathbf{K}(t)\widehat{\mathbf{H}})\mathbf{P}(t|t-1),\end{aligned}$$

where  $\widehat{\mathbf{H}} = \nabla_{\mathbf{z}}\mathbf{z}(t)|_{\mathbf{z}(t|t-1)}$ . There are also *Unscented Kalman filter* (UKF) [39] that further enhance the accuracy of the EKF.

For extremely nonlinear models, *particle filter* [18, 40, 41] may be applied to facilitate more accurate, albeit more computationally intensive, tracking. Briefly speaking, in a particle filter, the probability distribution is approximated by a probability mass function (*pmf*) evaluated at a set of randomly sampled points (*particles*). Then, the sequential Bayesian estimation is carried out on individual particles.

## 3.18.7 Conclusion

In this chapter, source localization algorithms in the context of wireless sensor network are discussed. A distinct approach of this chapter is to separate the received source signal model from the basic triangulation algorithms. The development also revealed basic relations among several well studied families of localization algorithms. The significance of tracking algorithm in the localization task is also discussed and some basic tracking algorithms are reviewed.

*Relevant Theory:* Signal Processing Theory, Machine Learning Statistical Signal processing, and Array Signal Processing

See [Vol. 1, Chapter 2](#), Continuous-Time Signals and Systems

See [Vol. 1, Chapter 3](#), Discrete-Time Signals and Systems

See [Vol. 1, Chapter 4](#), Random Signals and Stochastic Processes

See [Vol. 1, Chapter 11](#), Parametric Estimation

See [Vol. 1, Chapter 19](#) A Tutorial Introduction to Monte Carlo Methods

See this volume, [Chapter 5](#), Distributed Signal Detection

See this volume, [Chapter 7](#), Geolocation—Maps, Measurements, Models, and Methods

See this volume, [Chapter 19](#), Array Processing in the Face of Nonidealities

## References

- [1] P. Daly, Electron. Commun. Eng. J. 5 (1993) 349–357.
- [2] B.W. Parkinson, J.J. Spilker (Eds.), Global Positioning System: Theory and Applications, vol. 1, American Institute of Astronautics and Aeronautics, 1996.

- [3] N.L. Owsley, in: S. Haykin (Ed.), *Array Signal Processing*, Prentice-Hall, Englewood-Cliffs, NJ, 1991.
- [4] S. Zhou, P. Willett, *IEEE Trans. Signal Process.* 55 (2007) 3104–3115.
- [5] P. Valin, A. Jouan, E. Bosse, in: Proc. SPIE-1999 Sensor Fusion: Architectures, Algorithms, and Applications III, vol. 3719, Society of Photo-Optical Instrumentation Engineers, Bellingham, WA, USA, Orlando, FL, USA, 1999, pp. 126–138.
- [6] G.L. Duckworth, M.L. Frey, C.E. Remer, S. Ritter, G. Vidaver, in: Proc. SPIE, vol. 2344, The International Society for Optical Engineering, 1995, pp. 16–29.
- [7] C. Friedrich, U. Wegler, *Geophys. Res. Lett.* 32 (2005) L14312.
- [8] D. Gajewski, K. Sommer, C. Vanelle, R. Patzig, *Geophysics* 74 (2009) WB55–WB61.
- [9] J. Zhao, *Seismic signal processing for near-field source localization*, Ph.D. Dissertation, University of California, Los Angeles, 2007.
- [10] R.R. Ramirez, *Scholarpedia* 3 (11) (2008) 1073.
- [11] B.C. Basu, S.A. Pentland, in: Proceedings of ICASSP'01, vol. 5, pp. 3361–3364.
- [12] P. Aarabi, A. Mahdavi, in: Proceedings of ICASSP'02, IEEE, 2002, pp. 273–276.
- [13] J.C. Chen, K. Yao, R.E. Hudson, *IEEE Signal Process. Mag.* 19 (2002) 30–39.
- [14] J. Chen, R.E. Hudson, K. Yao, *IEEE Trans. Signal Process.* 50 (2002) 1843–1854.
- [15] Y.H. Hu, X. Sheng, D. Li, in: *IEEE Workshop on Multimedia Signal Processing*, IEEE, St. Thomas, Virgin Island, 2002.
- [16] D. Li, Y.H. Hu, *EURASIP J. Appl. Signal Process.* (2003) 321–337.
- [17] X. Sheng, Y.H. Hu, in: *Proceedings of the International Symposium on Information Processing in Sensor Networks (IPSN'03)*, Springer-Verlag, Palo Alto, CA, 2003, pp. 285–300.
- [18] X. Sheng, Y.H. Hu, in: *Proceedings of ICASSP'04*, vol. 3, IEEE, Montreal, Canada, 2004, pp. 972–975.
- [19] X. Sheng, Y.H. Hu, *IEEE Trans. Signal Process.* 53 (2005) 44–53.
- [20] S. Amari, A. Cichocki, H.H. Yang, in: *Proceedings of Advances in Neural Information Processing Systems (NIPS'96)*, MIT Press, 1996, pp. 757–763.
- [21] J.F. Cardoso, *Proc. IEEE* 86 (1998) 2009–2025.
- [22] S.C. Douglas, in: Y.H. Hu, J.N. Hwang (Eds.), *Handbook of Neural Network Signal Processing*, CRC Press, Boca Raton, FL, 2001 (Chapter 7).
- [23] G. Gelle, M. Colas, G. Delaunay, *Mech. Syst. Signal Process.* 14 (2000) 427–442.
- [24] K.-C. Chang, C.-Y. Chong, Y. Bar-Shalom, *IEEE Trans. Automatic Control* 31 (1986) 889–897.
- [25] M. Ito, S. Tsujimichi, Y. Kosuge, in: *Proceedings of the International Conference on Industrial Electronics, Control and Instrumentation*, New Orleans, LA, vol. 3, pp. 1260–1264.
- [26] C. Savarese, J.M. Rabaey, J. Beutel, in: *Proceedings of ICASSP'2001*, IEEE, Salt Lake City, UT, 2001, pp. 2676–2679.
- [27] V. Seshadri, G. Zaruba, M.A. Huber, in: *Proceedings of the International Conference on Pervasive Computing and Communications (PerCom'05)*, IEEE, 2005, pp. 75–84.
- [28] C.H. Knapp, G.C. Carter, *IEEE Trans. Acoust. Speech Signal Process.* 24 (1976) 320–327.
- [29] G.C. Carter (Ed.), *Coherence and Time Delay Estimation*, IEEE Press, 1993.
- [30] Y.T. Chan, R.V. Hattin, J.B. Plant, *IEEE Trans. Acoust. Speech Signal Process.* 26 (1978) 217–222.
- [31] T.L. Tung, K. Yao, C.W. Reed, R.E. Hudson, D. Chen, J.C. Chen, in: *Proceedings of SPIE*, vol. 3807, The International Society for Optical Engineering, 1999, pp. 220–233.
- [32] J. Benesty, J. Chen, Y. Huang, *IEEE Trans. Speech Audio Process.* 1542 (2004) 509–519.
- [33] F. Gustafsson, F. Gunnarsson, in: *Proceedings of ICASSP'03*, vol. 6, IEEE, 2003, pp. 553–556.
- [34] L. Yang, K.C. Ho, *IEEE Trans. Signal Process.* 57 (2009) 4598–4615.
- [35] H. Krim, M. Viberg, *IEEE Signal Process. Mag.* 1583 (1996) 67–94.
- [36] E. Tuncer, B. Friedlander (Eds.), *Classical and Modern Direction-of-Arrival Estimation*, Academic Press, 2010.

- [37] P. Stoica, A. Nehorai, IEEE Trans. Acoust. Speech Signal Process. 37 (1989) 720–741.
- [38] T. Sogo, H. Ishiguro, M.M. Trivedi, in: Proceedings of IEEE Workshop Omnidirectional Vision, IEEE, 2000, pp. 153–160.
- [39] S.J. Julier, J.K. Uhlmann, Proc. IEEE 92 (2004) 401–422.
- [40] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, IEEE Trans. Signal Process. 50 (2002) 174–188.
- [41] X. Sheng, Y.H. Hu, in: Proceedings of 4th International Symposium on Information Processing in Sensor Networks (IPSN '05), IEEE, Los Angeles, CA, 2005.

# Array Processing in the Face of Nonidealities

# 19

Mário Costa<sup>\*</sup>, Visa Koivunen<sup>\*</sup>, and Mats Viberg<sup>†</sup>

<sup>\*</sup>Department of Signal Processing and Acoustics, School of Electrical Engineering,  
Aalto University/SMARTAD CoE, Finland

<sup>†</sup>Division of Signal Processing and Antennas, Chalmers University of Technology, Sweden

## 3.19.1 Introduction

In array signal processing one is typically interested in characterizing, synthesizing, enhancing or attenuating certain aspects of propagating wavefields by employing a collection of sensors, known as a *sensor array*. Characterizing a propagating wavefield refers to determining its *spatial spectrum*, i.e., the angular distribution of energy, so that information regarding the location of the sources generating the wavefield can be obtained, for example [1]. Synthesizing or producing a wavefield refers to generating a propagating wavefield with a desired spatial spectrum in order to focus the transmitted energy towards certain locations in space. Finally, attenuating or enhancing a received wavefield based on its spatial spectrum refers to the ability of canceling interfering sources or improving the signal-to-interference-plus-noise ratio (SINR) and maximizing the energy received from certain directions. Examples of characterization, synthesis and enhancement of propagating wavefields include direction-of-arrival (DoA) estimation, angle spread estimation in channel sounding as well as transmit and receive beamforming [2].

Traditionally, array signal processing has found applications in radar and sonar, defense systems, signal intelligence (SIGINT), and surveillance as well as imaging and biomedical applications. Radioastronomy also employs many array processing techniques [3]. More recently, it has been used in wireless communication systems, in particular in basestations that utilize beamforming techniques in controlling the interference and enhancing the signal quality. Moreover, creating advanced, measurement-based models of the radio channels in communication systems such as long-term evolution (LTE) requires capturing the directional properties of the propagation channel [4]. In global navigation satellite systems, receive beamforming techniques for anti-jamming purposes are often employed as well as DoA estimation techniques in indoor navigation systems [5]. See also Chapter 20 “Applications of Array Signal Processing” in this volume.

The propagating wavefield is typically parameterized by the angular location of sources generating such a wavefield, their polarization state, bandwidth, delay profile, and Doppler shift. These are called *wavefield parameters* and are given in terms of a reference system, which in the case of angular information is typically assumed to be the coordinate system common to the sensor array and propagating wavefield. Such a reference system is typically assumed to be within the physical extent of the sensor array, known as *array aperture*, and the angular parameterization of the propagating wavefield refers

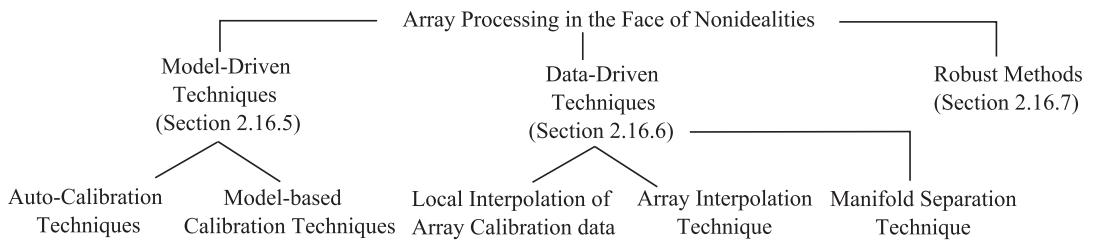
to the DoAs or directions-of-departure (DoDs), characterizing the spatial spectrum of the wavefield received or transmitted by the sensor array.

In addition to the propagating wavefield, a model describing the response of the sensor array as a function of the wavefield parameters, such as the angle-of-arrival or departure, is typically required in array processing. Such a model is defined in terms of *array steering vectors* and allows us to estimate the wavefield parameters from data acquired by the sensor array, and use them to design a beamformer at the receiver. Similarly, steering vectors are used to synthesize a desired wavefield and employ transmit beamforming techniques. In order to simplify these signal processing tasks and models, the standard approach to array signal processing assumes rather idealistic array steering vector models. In particular, all array elements are typically assumed to have similar omnidirectional gain patterns and the employed sensor array is assumed to have a regular geometry such as a uniform linear array (ULA), a uniform rectangular array (URA), or a uniform circular array (UCA). The resulting array steering vector models have then a very convenient form for various signal processing tasks; see Section 3.19.2. However, most of real-world sensor arrays are not well described using such an ideal array model. In fact, elements of real-world arrays have individual beampatterns, not necessarily omnidirectional, and may be subject to severe mutual coupling. Phase centers of the elements may not be exactly in the assumed positions. Moreover, mounting platform reflections and cross-polarization effects are also very common in real-world arrays.

In practical array processing applications employing ideal array steering vector models leads to a performance degradation and typically to loss of optimality for optimal array processors [6]. The limiting factor in the performance of high-resolution array processing algorithms as well as in the tightness of related theoretical performance bounds is known to be the accuracy of the employed array model rather than measurement noise [6, 7]. Similarly, misspecified sensor array models may lead to a severe performance loss of beamforming techniques. Effects include steering energy towards unwanted directions, cancelation of the signal of interest (SOI) as well as amplification of interfering sources [8].

In this chapter we present techniques that allow the practitioner to acquire a realistic array steering vector model by taking into account array nonidealities such as mutual coupling, mounting platform reflections, cross-polarization effects, errors in element positions as well as individual directional beampatterns. This facilitates achieving optimal performance in the presence of nonidealities as well as mitigating problems related to beam-steering, SOI and interference cancellation. We also describe how the various approaches can be applied in the context of high-resolution direction finding and beamforming. Emphasis is given to the case when the array response, along with its nonidealities, is obtained from array calibration measurements. However, the methods and techniques discussed in this chapter are also applicable when the array response is obtained from EM simulation software, or even when ideal array models are employed. Typically, EM simulation software does not capture manufacturing errors while calibration measurement noise is unavoidable in array calibration measurements. Techniques for denoising array calibration measurements are included in this chapter as well. Sensor failure in array signal processing is not addressed herein, and the interested reader is referred to [9, 10] and references therein. Many of such techniques aim at determining the inoperable sensors' outputs from the available array snapshots, and proceed with the array processing tasks as if the sensor array were fully operable. Then, realistic array steering vector models are still required and the methods discussed herein may also be useful in such circumstances.

The classification used in this chapter for the various techniques capable of dealing with array nonidealities is given in Figure 19.1. We classify the methods trying to capture the nonidealities as

**FIGURE 19.1**

Classification of techniques for array processing in the face of nonidealities.

model-driven and data-driven techniques. Robust methods are a third class of methods that acknowledge that the array model contains errors without trying to characterize such nonidealities. Instead, robust estimation methods trade-off desirable properties such as high-resolution or optimality for reliability in the face of uncertainties in the array response. In model-driven techniques, the array nonidealities are described using an explicit formulation for each nonideality. The parameters of such a formulation may be estimated from array calibration measurements or simultaneously with wavefield parameters. The latter approach is called auto-calibration technique [7, 11–14]. Data-driven techniques use array calibration data as a starting point and capture the nonidealities implicitly by using basis function expansion, interpolation, approximation or nonparametric estimation techniques. Data-driven methods include local interpolation of array calibration data [6], array interpolation technique [15–17], and manifold separation technique [18, 19] which stems from the wavefield modeling principle [20–22]. These techniques do not employ any explicit model for the array nonidealities. In data-driven techniques the array nonidealities are described by the basis function coefficients, which may be estimated from array calibration measurements. Hence, they allow the practitioner to develop array processing algorithms that do not require explicit formulation for the nonidealities, are independent of the sensor array, including its geometry and individual element beampatterns while obtaining close to optimal performance. Finally, robust methods try to bound the influence of modeling errors in the estimation process instead of trying to capture them.

This chapter is organized as follows. First, conventional array steering vector models and widely employed techniques in array processing are briefly described in Section 3.19.2. Then, typical explicit formulations for array nonidealities are described in Section 3.19.3. In Section 3.19.4, array calibration measurements in controlled environments are briefly described. Section 3.19.5 includes model-driven techniques that are based on explicit formulations of the array nonidealities. Section 3.19.6 considers data-driven techniques. In Section 3.19.7, robust methods are described. Section 3.19.8 includes extensive array processing examples. Conclusions are given in Section 3.19.9.

## 3.19.2 Ideal array signal models

The conventional narrowband  $N$ -element array output model due to a propagating wavefield, generated by  $P \in \mathbb{N}$  far-field sources, is

$$\mathbf{x}(k) = \mathbf{A}(\boldsymbol{\theta}, \boldsymbol{\phi})\mathbf{s}(k) + \mathbf{n}(k), \quad (19.1)$$

where  $\mathbf{A}(\theta, \phi) \in \mathbb{C}^{N \times P}$ ,  $\mathbf{s}(k) \in \mathbb{C}^{P \times 1}$ , and  $\mathbf{n}(k) \in \mathbb{C}^{N \times 1}$  denote the array steering matrix, transmitted waveforms, and sensor noise, respectively. The discrete time instant is denoted by  $k \in \mathbb{N}$  while  $\theta \in \mathbb{R}^{P \times 1}$  and  $\phi \in \mathbb{R}^{P \times 1}$  represent the co-elevation and azimuth angles of the  $P$  sources generating the propagating wavefield, respectively. Typically, the co-elevation angle ( $\theta \in [0, \pi]$ ) is measured down from the  $z$ -axis and the azimuth angle ( $\phi \in [0, 2\pi]$ ) is measured counter-clockwise in the  $xy$ -plane. In Eq. (19.1), the  $N$ -dimensional observation vector  $\mathbf{x}(k) \in \mathbb{C}^{N \times 1}$  is known as *array snapshot*. The array steering matrix  $\mathbf{A}(\theta, \phi)$  is composed of  $P$  array steering vectors  $\mathbf{a}(\theta, \phi) \in \mathbb{C}^{N \times 1}$ , each representing the array response to a plane-wave impinging on the sensor array from directions  $\phi_1, \dots, \phi_P$ . In array processing, the employed sensor array is typically assumed to be *unambiguous* in the sense that any collection of  $P$  ( $P < N$ ) steering vectors with different angles forms a linearly independent set.

Assuming that the employed sensor array lies in the  $xy$ -plane, and is not subject to nonidealities such as mutual coupling or cross-polarization effects, the corresponding array steering vector model may be written as

$$\mathbf{a}(\theta, \phi) = [g_1(\theta, \phi)e^{jk(x_1 \sin(\theta) \cos(\phi) + y_1 \sin(\theta) \sin(\phi))}, \dots, g_N(\theta, \phi)e^{jk(x_N \sin(\theta) \cos(\phi) + y_N \sin(\theta) \sin(\phi))}], \quad (19.2)$$

where  $g_n(\theta, \phi) \in \mathbb{R}$  denotes the gain function of the  $n$ th element. In (19.2),  $\kappa = 2\pi/\lambda$  and  $\lambda$  denote the angular wavenumber and wavelength, respectively. Moreover,  $x_n, y_n \in \mathbb{R}$  denote the location (in meters) of the  $n$ th element in the  $xy$ -plane, relative to the origin of the assumed coordinate system. Note that other wavefield parameters such as the polarization of the sources may also be included in the array steering vector model in (19.2). This is briefly discussed in Section 3.19.8.

Typically, in array signal processing the steering vector model in (19.2) is further simplified by assuming that the array elements are all identical and have omnidirectional gain functions, i.e.,  $g_n(\theta, \phi) = 1$ , and are arranged in regular geometries. Commonly used ideal steering vector models include those of ULAs, UCAs, and URAs:

$$\mathbf{a}_{\text{ULA}}(\phi) = [1, e^{jk d \cos(\phi)}, \dots, e^{jk d(N-1) \cos(\phi)}]^T, \quad (19.3a)$$

$$\mathbf{a}_{\text{UCA}}(\theta, \phi) = [e^{jk r \sin(\theta) \cos(\phi - \gamma_1)}, \dots, e^{jk r \sin(\theta) \cos(\phi - \gamma_N)}]^T, \quad (19.3b)$$

$$\begin{aligned} \mathbf{a}_{\text{URA}}(\theta, \phi) = & [1, e^{jk d_x \sin(\theta) \cos(\phi)}, \dots, e^{jk d_x (N_x-1) \sin(\theta) \cos(\phi)}] \\ & \otimes [1, e^{jk d_y \sin(\theta) \cos(\phi)}, \dots, e^{jk d_y (N_y-1) \sin(\theta) \cos(\phi)}]^T, \end{aligned} \quad (19.3c)$$

where  $\otimes$  and  $d$  denote the Kronecker product and the inter-element spacing, respectively. In (19.3b),  $r \in \mathbb{R}$  and  $\gamma_n = 2\pi n/N$  denote the radius of the circular array and the angular position of the  $n$ th element, respectively.

Assuming wavefield propagation in the  $xy$ -plane as well as uncorrelation between transmitted signals and sensor noise, the array covariance matrix of (19.1) is given by

$$\mathbf{R}_X = \mathbf{A}(\phi)\mathbf{R}_S\mathbf{A}(\phi)^H + \mathbf{R}_N, \quad (19.4)$$

where  $\mathbf{R}_S \in \mathbb{C}^{P \times P}$  and  $\mathbf{R}_N \in \mathbb{C}^{N \times N}$  denote the covariance matrices of the transmitted signals and sensor noise, respectively. The signal covariance matrix  $\mathbf{R}_S$  may be rank deficient, with rank  $P'$  ( $P' \leq P$ ), due to highly correlated or coherent sources that may be caused by specular multipath propagation, for example. Sensor noise is typically assumed to be zero-mean complex-circular Gaussian distributed  $\mathcal{N}_C(\mathbf{0}, \sigma^2 \mathbf{I}_N)$ . In practice, the exact covariance matrix in (19.4) is unknown and it is typically estimated

from a collection of  $K$  ( $K \geq N$ ) array snapshots as

$$\widehat{\mathbf{R}}_X = \frac{1}{K} \sum_{k=1}^K \mathbf{x}(k) \mathbf{x}(k)^H. \quad (19.5)$$

Signal models (19.1) and (19.4) are used in most array processing tasks such as beamforming and direction finding. In estimation problems, maximum likelihood methods are commonly used to find the optimal parameter estimates whereas beamformers typically target at enhancing the signal by maximizing the SINR at the array output.

A popular criterion for evaluating the performance of beamformers is the array output SINR [8]:

$$\text{SINR} = \frac{\sigma_S^2 |\mathbf{w}^H \mathbf{a}(\phi_S)|^2}{\mathbf{w}^H \mathbf{R}_{I+N} \mathbf{w}}, \quad (19.6)$$

where  $\phi_S \in [0, 2\pi)$  and  $\sigma_S^2 \in \mathbb{R}$  denote the angle from where the SOI impinges on the sensor array and the corresponding signal power, respectively. Moreover,  $\mathbf{w} \in \mathbb{C}^{N \times 1}$  denotes the beamformer weight vector and  $\mathbf{R}_{I+N} = \mathbf{A}(\phi) \mathbf{R}_I \mathbf{A}(\phi)^H + \mathbf{R}_N \in \mathbb{C}^{N \times N}$  the covariance matrix due to both interfering signals  $\mathbf{R}_I$  and sensor noise. The optimal weight vector that maximizes (19.6) is [8]

$$\mathbf{w}_{\text{OPT}} = \alpha \mathbf{R}_{I+N}^{-1} \mathbf{a}(\phi_S), \quad (19.7)$$

where  $\alpha \in \mathbb{R}$  may be arbitrary since it does not affect the SINR in (19.6). Choosing  $\alpha = 1/(\mathbf{a}(\phi_S)^H \mathbf{R}_{I+N}^{-1} \mathbf{a}(\phi_S))$  leads to the well-known minimum variance distortionless response (MVDR) beamformer, also known as Capon beamformer [8, 23]. Note that using the exact  $\mathbf{R}_X$  in place of  $\mathbf{R}_{I+N}$  in (19.7) does not affect the array output SINR.

In Section 3.19.1, we have mentioned that the DoAs of the sources generating the propagating wavefield may be found from its spatial spectrum. The location of the sources are associated with the angles of the spatial spectrum with larger power. We may therefore view DoA estimation as a spectrum estimation problem and employ beamforming techniques for estimating the angular distribution of power of the wavefield received by the sensor array. Such an approach is called nonparametric or spectral-based approach to DoA estimation since it does not require a parametric model for the sources, nor the number of sources generating the wavefield. These techniques are versatile but typically have poor *resolution* and lead to suboptimal DoA estimates. Informally, resolution refers to the ability of distinguishing between two closely spaced sources. The resolution of beamforming techniques is limited by the array aperture, i.e., the physical size of the array in wavelengths, as well as SNR, and does not improve with increasing number of array snapshots.

One way of improving the resolution limit imposed by the array aperture is by making further assumptions regarding the sources generating the propagating wavefield. In particular, we first assume that the number of sources  $P$  as well as rank of the signal covariance matrix  $\mathbf{R}_S$  are known or have been correctly estimated from the array output. The radiating sources are also assumed to be located in the far-field of the sensor arrays as well as point-emitters in the sense that the field radiated by each source can be assumed to have originated from a single location in space. Finally, we assume that the number of sources generating the wavefield is smaller than the number of array elements. Then, the array output

in (19.1) is known as low-rank signal model and the array covariance matrix in (19.4) can be written as

$$\mathbf{R}_X = \mathbf{E}_S \boldsymbol{\Lambda}_S \mathbf{E}_S^H + \mathbf{E}_N \boldsymbol{\Lambda}_N \mathbf{E}_N^H. \quad (19.8)$$

Here,  $\mathbf{E}_S \in \mathbb{C}^{N \times P'}$  and  $\mathbf{E}_N \in \mathbb{C}^{N \times (N-P')}$  contain the eigenvectors of  $\mathbf{R}_X$  spanning the so-called signal and noise subspaces while  $\boldsymbol{\Lambda}_S \in \mathbb{C}^{P' \times P'}$  and  $\boldsymbol{\Lambda}_N \in \mathbb{C}^{(N-P') \times (N-P')}$  contain the corresponding eigenvalues in their diagonal. Techniques employing the low-rank signal model are called subspace methods and are a class of high-resolution DoA estimation algorithms [2]. They exploit the fact that the columns of  $\mathbf{E}_S$  span the same subspace as the columns of the steering matrix  $\mathbf{A}(\phi)$  (in the case of coherent signals  $\mathbf{E}_S$  is contained in the subspace spanned by the columns of  $\mathbf{A}(\phi)$ ), and that both  $\mathbf{E}_S$  and  $\mathbf{A}(\phi)$  are orthogonal to  $\mathbf{E}_N$ . Unlike beamforming techniques, the resolution of subspace methods improves with increasing number of array snapshots.

A commonly used lower bound on the estimation error variance of any unbiased estimator is the Cramér-Rao lower Bound (CRB) [24]. Assuming that both signal and noise are zero-mean complex-circular Gaussian distributed, the unconditional CRB for azimuth angle estimation is [25]

$$\text{CRB}(\phi) = \frac{\sigma^2}{2N} \left[ \Re \left\{ \left( \dot{\mathbf{A}}(\phi)^H \boldsymbol{\Pi}_A^\perp \dot{\mathbf{A}}(\phi) \right) \odot \left( \mathbf{R}_S \mathbf{A}(\phi)^H \mathbf{R}_X^{-1} \mathbf{A}(\phi) \mathbf{R}_S \right)^T \right\} \right]^{-1}, \quad (19.9)$$

where  $\dot{\mathbf{A}}(\phi) = \begin{bmatrix} \frac{\partial \mathbf{a}(\phi_1)}{\partial \phi_1} & \dots & \frac{\partial \mathbf{a}(\phi_P)}{\partial \phi_P} \end{bmatrix} \in \mathbb{C}^{N \times P}$ . Moreover,  $\odot$  and  $\boldsymbol{\Pi}_A^\perp \in \mathbb{C}^{N \times N}$  denote the Hadamard-Schur product and a projection matrix onto the nullspace of  $\mathbf{A}(\phi)^H$ , respectively. An estimator with an error covariance matrix that equals (19.9) is called statistically efficient. In particular, the stochastic maximum likelihood estimator is asymptotically ( $K \rightarrow +\infty$ ) statistically efficient, and the azimuth-angle estimates are obtained as [26]

$$\hat{\phi} = \arg \min_{\phi} \det \left\{ \mathbf{A}(\phi) \hat{\mathbf{R}}_S \mathbf{A}(\phi)^H + \hat{\sigma}^2 \mathbf{I} \right\}. \quad (19.10)$$

Here,  $\det \{\cdot\}$  denotes the determinant of a matrix. Moreover,  $\hat{\mathbf{R}}_S$  and  $\hat{\sigma}^2$  denote estimates of the signal covariance matrix and sensor noise, respectively. See [26] and chapter DOA Estimation Methods and Algorithms of this book for details.

Asymptotically optimal DoA estimation algorithms such as the stochastic maximum likelihood estimator, and beamforming techniques such as the Capon beamformer, are often very sensitive to uncertainties in the array steering vector model and sensor noise (and interferers) statistics [6,8]. Under these scenarios, optimal DoA estimators may be subject to bias and increased variance while optimum beamformers may suffer from SOI cancellation effect. Uncertainty in noise statistics may be due to outliers, i.e., highly deviating observations that do not follow the same pattern as the majority of the data, or incorrect assumptions on the noise environment. For example, man-made interference has typically a non-Gaussian heavy-tailed distribution, for which (19.5) may no longer be a consistent estimator of  $\mathbf{R}_X$  [27]. Uncertainties in the steering vector model are often due to misspecification or lack of knowledge of various array nonidealities. These include:

- Uncertainty in array elements' beampatterns and positions.
- Mutual coupling.
- Mounting platform reflections.
- Cross-polarization effects.

- Departures from narrowband, far-field and point-source assumptions.
- Errors introduced by the receiver front-end architecture.
- Effects of nonlinear elements.

In subsequent sections we describe in detail various rigorous and practical approaches for taking the aforementioned nonidealities of real-world arrays into account by array processing techniques. For details on array processing under uncertainty in noise statistics the reader is referred to [27].

### 3.19.3 Examples of array nonidealities

This section describes the main nonidealities experienced in real-world sensor arrays. We discuss their effects in array processing techniques, and provide explicit formulations describing such array nonidealities that can be found in the literature. These models are rather specific but allow one to understand the departure from the ideal array model as well as to incorporate such nonidealities into both DoA estimators and beamforming techniques.

#### 3.19.3.1 Mutual coupling

Mutual coupling (also known as cross-talk) refers to interactions among array elements. The signal received by an element affects the signals received by the other array elements, and similarly for signal transmission [28, Chapter 2]. Typically, mutual coupling is inversely proportional to the inter-element spacing as well as isolation among array elements. Mutual coupling distorts the elements' radiation patterns and decreases the efficiency of sensor arrays. Mobile wireless terminals equipped with antenna arrays are a typical example where mutual coupling is significant since the whole chassis, along with its other components, can be considered part of the antenna [5]. Array processing algorithms typically experience a significant loss of performance when the employed array steering vector model does not account for mutual coupling [6,29].

The steering vector of a sensor array subject to mutual coupling is typically modeled as [6]

$$\mathbf{a}(\phi) = \mathbf{C}\mathbf{a}_0(\phi), \quad (19.11)$$

where  $\mathbf{C} \in \mathbb{C}^{N \times N}$  denotes the mutual coupling matrix and  $\mathbf{a}_0(\phi) \in \mathbb{C}^{N \times 1}$  is known as *nominal* array steering vector. The  $\mathbf{C}(m, n)$  element describes the contribution of the  $n$ th array element to the output of the  $m$ th sensor. Nominal sensor arrays are typically considered to be ideal uniform arrays with regular geometries of the form (16.3), and the motivation for their use is based on the assumption that real-world arrays may be described as a perturbation from an ideal sensor array. For example, if the nominal array is assumed to be an ideal UCA the mutual coupling matrix takes the form of a circulant matrix [30]. Note that a diagonal matrix  $\mathbf{C}$  in (19.11) may also describe errors due to the receiver front-end such as imbalance in the I/Q channels. This is also discussed in Section 3.19.3.5.

In practice, the mutual coupling matrix needs to be determined from array calibration measurements and a nominal array steering vector should be specified by the practitioner. Typically, the mutual coupling matrix is obtained from the measured network parameters of the sensor array, including scattering and transmission coefficients, which may be a rather tedious task [31,32, Chapter 2]. Moreover, determining the nominal array steering vector is typically based on trial and error, and rely on visual inspection of the real-world sensor array.

### 3.19.3.2 Uncertainty in array elements' beampatterns and positions

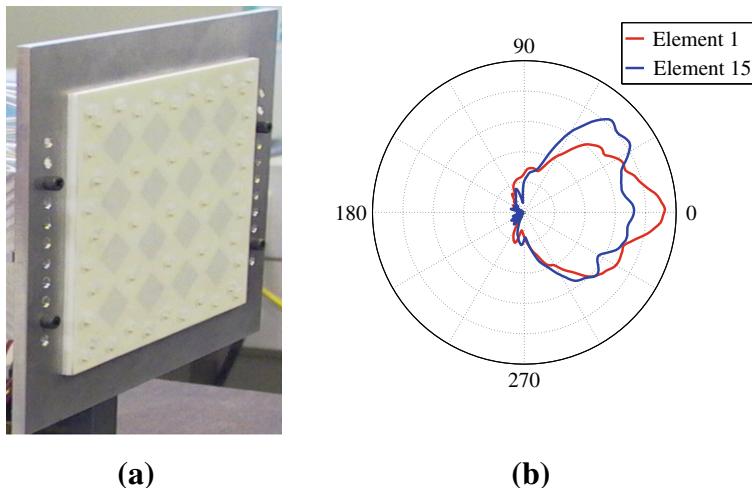
Often, elements' beampatterns and positions in real-world arrays are not fully known. This may be caused by normal variability in the manufacturing process in the sense that each array element has an individual beampattern or may suffer from manufacturing errors. In fact, elements' phase centers in real-world arrays do not typically correspond to their physical locations due to interactions with other array elements and mounting platform. Misspecification of the array element's beampatterns and phase centers leads to loss of performance in array processing techniques.

A commonly employed model taking into account individual beampatterns and position errors is

$$\mathbf{a}(\phi) = \text{diag} \left\{ g_1(\phi) e^{jk(\tilde{x}_1 \cos \phi + \tilde{y}_1 \sin \phi)}, \dots, g_N(\phi) e^{jk(\tilde{x}_N \cos \phi + \tilde{y}_N \sin \phi)} \right\} \mathbf{a}_0(\phi), \quad (19.12)$$

where  $\tilde{x}_n, \tilde{y}_n \in \mathbb{R}$ , and  $g_n(\phi) \in \mathbb{R}$  denote the error in the  $n$ th element's position, with respect to the nominal array steering vector, and corresponding directional beampattern. Parameters  $\tilde{x}_n, \tilde{y}_n$  may be estimated from calibration measurements taken in controlled environments while the gain function  $g_n(\phi)$  may be measured at a discrete set of points and interpolated using appropriate basis functions such as splines [6]. The latter approach leads to a technique known as array interpolation and it is described in Section 3.19.6.

Alternatively, one may specify a parametric model for  $g_n(\phi)$  (i.e., functionally dependent on  $\phi$ ) by trial and error, and visual inspection. For example, in the case of electrically short (relative to the wavelength)  $x$ -oriented dipoles one can use the following approximation  $g_n(\phi) \approx \sin \phi$ . However, in the general case of electrically large antennas and patch elements, specifying a parametric model for  $g_n(\phi)$  may be very challenging. An example of two gain functions of a real-world array is illustrated in Figure 19.2b.



**FIGURE 19.2**

(a) Real-world rectangular array with  $N = 4 \times 4$  dual-polarized patch elements. (b) Gain patterns of two elements of the real-world rectangular array. Courtesy of the Department of Radio Science and Technology, Aalto University, Finland.

### 3.19.3.3 Cross-polarization effects

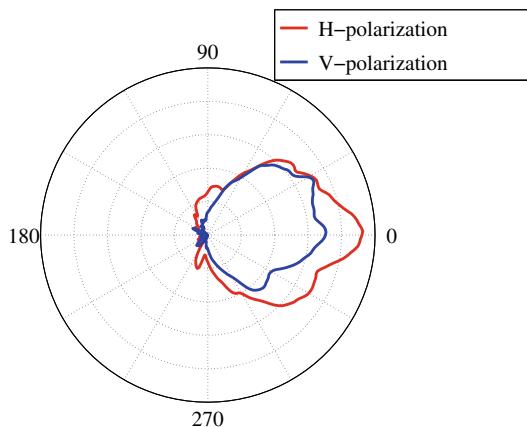
Cross-polarization effects refer to “leakage” that, for example, a vertically polarized element suffers from an horizontally polarized wavefield. They are typically characterized by the cross-polarization discrimination (XPD), denoting the ratio between the power received by an antenna due to co-polarized and cross-polarized wavefields. The ratio between the powers received in different polarizations is commonly expressed in dB scale. XPD defines quantitatively how well the two received channels that use different polarization orientations are isolated. Antennas with a large XPD are essentially insensitive to cross-polarization effects and the power received from cross-polarized wavefields may be neglected. When mounted on an array the antennas’ XPD may change significantly due to complex EM interactions among the array elements, scatterers, and mounting platform. In such cases, high-resolution DoA estimators that do not take cross-polarization effects into account typically lead to estimates that may contain significant bias and excess variance [33].

The steering vector of a sensor array that is subject to cross-polarization effects may be described as

$$\mathbf{a}(\phi) = \mathbf{a}_{\text{co}}(\phi)\alpha_{\text{co}} + \mathbf{a}_{\text{cross}}(\phi)\alpha_{\text{cross}}, \quad (19.13)$$

where  $\mathbf{a}_{\text{co}}(\phi) \in \mathbb{C}^{N \times 1}$  and  $\mathbf{a}_{\text{cross}}(\phi) \in \mathbb{C}^{N \times 1}$  denote the array responses due to a co-polarized and cross-polarized wavefields, respectively. Moreover,  $\alpha_{\text{co}}, \alpha_{\text{cross}} \in \mathbb{C}$  define the polarization of the wavefield. For example, for a co-polarized wavefield Eq. (19.13) simplifies to  $\mathbf{a}(\phi) = \mathbf{a}_{\text{co}}(\phi)$  while for a cross-polarized wavefield we have  $\mathbf{a}(\phi) = \mathbf{a}_{\text{cross}}(\phi)$ .

Parametric modeling of  $\mathbf{a}_{\text{co}}(\phi)$  and  $\mathbf{a}_{\text{cross}}(\phi)$  in (19.13) may now be done by employing models (19.11) and (19.12). However, specifying a nominal array steering vector model for the cross-polarized component  $\mathbf{a}_{\text{cross}}(\phi)$  is even more challenging than that of the co-polarized component, where one may approximate the element’s gain function as  $g_n(\phi) \approx \sin(\phi)$ . Typically, visual inspection does not help much in determining a parametric model for  $g_n(\phi)$ . An example of gain functions corresponding to an horizontally and vertically polarized wavefield is illustrated in Figure 19.3.



**FIGURE 19.3**

Gain functions corresponding to the horizontal and vertical polarization components of an element of the real-world rectangular array from Figure 19.2a.

### 3.19.3.4 Departures from narrowband assumption

The narrowband signal model commonly used in array processing (see Section 3.19.2) assumes that the *time-bandwidth product* is “small,” i.e.,

$$B_s \tau \ll 1, \quad (19.14)$$

where  $B_s$  and  $\tau$  denote the bandwidth of the transmitted signal and the wavefront’s propagation delay across the array aperture, respectively. A rule of thumb for considering a signal narrowband is  $\text{sinc}(B_s \tau) \approx 1$  [34]. In practice, the time-bandwidth product may be such that the narrowband assumption no longer holds true. This is also the case in focusing-based wideband array processing, where signals’ bandwidth is divided into a set of narrowband channels and narrowband processing is applied to each narrowband bin or to a focused covariance matrix [35]. Alternatively, genuine space-time signal processing can be employed as in STAP radar systems [36,37].

Assuming an array with a flat frequency response (with linear phase) over the signals’ bandwidth, the array covariance matrix may be modeled as [34,38]

$$\mathbf{R}_X = \sum_{p=1}^P \left( \mathbf{a}(f_p, \phi_p) \mathbf{a}^H(f_p, \phi_p) \odot \mathbf{R}_p \right) + \sigma^2 \mathbf{I}_N, \quad (19.15)$$

where  $\odot$  denotes the element-wise Hadamard-Schur product. Moreover,  $\mathbf{a}(f, \phi) \in \mathbb{C}^{N \times 1}$  denotes the array steering vector with a linear phase response over frequency and  $\mathbf{R}_p \in \mathbb{R}^{N \times N}$  contains the correlation of the  $p$ th signal among the array elements. For signals with small time-bandwidth product  $\mathbf{R}_p$  equals a matrix of ones and (19.15) reduces to (19.4). However, when  $B_s \tau$  is non-negligible the rank of  $(\mathbf{a}(f_p, \phi_p) \mathbf{a}^H(f_p, \phi_p) \odot \mathbf{R}_p)$  due to a single signal is larger than one, and the low-rank structure of the array covariance matrix in (19.4) is lost. Thus, high-resolution subspace methods may not be applicable anymore [34]. The influence of non-negligible time-bandwidth product on DoA estimators and beamforming techniques has been addressed in [34,38]. In most cases, the error due to non-negligible time-bandwidth product can be neglected when compared to finite sample effects. However, when sources are closely spaced or have large difference in power, such an error may be significant.

### 3.19.3.5 Errors due to receiver front-end architectures

Receiver architectures are commonly classified as superheterodyne, low-IF (intermediate frequency) or direct-conversion receivers. Each front-end architecture is subject to various nonidealities such as I/Q-imbalance, DC-offset, and interfering image frequencies that impact the performance of array processors.

Superheterodyne receivers typically consist of two or more IF stages in order to convert the radio-frequency (RF) signal to baseband. They require image rejection filters at each downconversion stage, which may be difficult to integrate on-chip with other components. Power consumption and size may be significant [39]. Low-IF receivers convert the RF signal to baseband in two IF stages, similarly to the superheterodyne receiver. The need for image rejection filters is overcome by the use of two mixers, one for each I/Q channels, in order to cancel the image. In practice, gain and phase imbalances in the I/Q channels limit the effectiveness of such image cancellation approach. Direct-conversion (or zero-IF) receivers downconvert the RF signal to baseband in a single stage. There is no need for image rejection

filters nor image cancellation. However, they typically suffer from DC offset, and I/Q imbalances in the demodulation process are still present.

I/Q imbalances in the demodulation process are common to all of the aforementioned receivers and have been studied in the context of array signal processing in [40]. They appear as phase and amplitude distortions in the I and Q branches of the demodulated signal. Similarly to errors caused by departures from narrowband assumption, the rank of the covariance matrix due to received signals increases by two and the noise eigenvalues are no longer identical. The low-rank structure of the array covariance matrix may be lost and subspace-based array processing methods experience a performance degradation.

One should note that I/Q imbalances may be significantly mitigated by using advanced digital down-converters, either at intermediate or radio frequencies. The commonly used low-rank model is then a good approximation of the array covariance matrix, given that a sufficient number of quantization bits are used [41]. Power consumption and high cost of ADCs operating at GHz and large bandwidths may be a limiting factor in practice.

We also note that antenna arrays using a single receiver, known as switched or time division multiplexing receivers, are often employed in practice due to their low-power, cost, and size [41, 42]. Typically, switched array receivers suffer from phase errors that need to be estimated and taken into account by array processing methods.

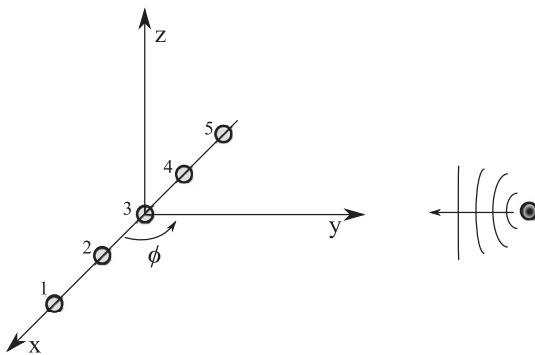
### 3.19.3.6 Effects of nonlinear elements

Linearity of the sensor array is among the most common assumptions in the array processing literature. Linearity, in the sense of superposition principle, essentially means that the array output due to a propagating wavefield generated by multiple sources equals the sum of array outputs due to a wavefield generated by each source separately. Most sensor arrays can be considered linear systems and the superposition principle may then be employed [32]. However, active elements such as low-noise amplifiers (LNAs) commonly used in receiver architectures are nonlinear systems and the superposition principle at the array output (after the RF front-end) may no longer hold true [43]. Typically, LNAs trade-off linearity for gain and noise figures. Some waveforms that have large peak to average power ratio such as OFDM (orthogonal frequency-division multiplexing) are particularly sensitive to nonlinearities of power amplifiers. Digital signal processing techniques for mitigating nonlinear effects due to RF front-end include pre- as well as post-distortion techniques [44].

---

## 3.19.4 Array calibration

The goal of array calibration is to capture the combined effects of sensor positions, their unknown gain and phase, mutual coupling characteristics as well as cross-polarization effects and mounting platform reflections. In array calibration one typically acquires the so-called array measurement matrix  $\tilde{\mathbf{A}}(\phi_c) \in \mathbb{C}^{N \times Q}$ . The array measurement matrix is composed of a collection of  $Q$  steering vectors corresponding to the angles contained in the vector  $\phi_c \in \mathbb{R}^{Q \times 1}$ . The standard approach of obtaining the array measurement matrix is by taking the sensor array into an anechoic chamber and measuring its response to a source, known as probe, from  $Q \in \mathbb{N}$  different known angles. The antenna array is usually mounted on a mechanical device, called positioner, that rotates the sensor array in azimuth (and

**FIGURE 19.4**

Example of the standard array calibration setup. A ULA is rotated in the  $xy$ -plane around its center element while a probe is held fixed in the far-field of the ULA.

possibly in elevation) while the probe is held fixed; see Figure 19.4. Note that the coordinate system employed for DoA estimation is defined by both the positioner and probe.

The array measurement matrix fully describes a given real-world sensor array as well as all its nonidealities. However, array calibration measurements are typically taken in controlled environments such as anechoic chambers, and may be subject to various errors including sensor noise, reflections from the anechoic chamber, imperfections of the employed positioner, attenuations and phase-drifts due to cabling, small distance between the antenna array and probe (i.e., not in far-field), and effects of the probe (e.g., not a point-source). These errors need to be corrected so that array processing techniques may employ an accurate model of the real-world array response. Approaches for reducing the aforementioned errors occurring during array calibration measurements can be found in [32, 33, 45]. Moreover, array calibration measurements do not provide any steering vector model, i.e., an explicit formulation describing the array response as a function of the wavefield parameters. Such difficulties may be alleviated by employing data-driven techniques described in Section 3.19.6.

Typically, the array measurement matrix contains the array response to angles spanning the whole angular region, such as  $\phi_C \in (0, 2\pi]$  in the azimuth-only case. However, in some applications the array response may only be measured over a small angular sector, and a partial array calibration is obtained. Examples include applications where the antenna array is deployed on an environment where the sources are known *a priori* to be confined to an angular sector or when the array dimensions do not allow a full calibration. Data-driven techniques described in Section 3.19.6 are also applicable in these cases.

### 3.19.5 Model-driven techniques

In this section, we describe techniques that assume an explicit model describing each array nonideality. The array nonidealities may be estimated, by employing such a model, from array calibration measurements or directly from the array output data, simultaneously with the wavefield parameters. The latter

approach is known as auto-calibration technique. Moreover, the array nonidealities may be assumed to be unknown deterministic or random parameters with known prior distribution. Hence, methods from estimation theory may be applied to estimate the array nonidealities.

### 3.19.5.1 Deterministic approach

Consider an  $N$ -element planar array lying in the  $xy$ -plane and denote the uncertainty in the array elements' positions by  $\rho = [\tilde{x}_1, \tilde{y}_1, \dots, \tilde{x}_N, \tilde{y}_N]^T \in \mathbb{R}^{2N \times 1}$ . In this case, the array is not assumed to be subject to other nonidealities such as cross-polarization effects or individual beampatterns. We may estimate the sensors' misplacements  $\rho$  from array calibration measurements as [6]

$$\hat{\rho} = \arg \min_{\rho} \|\tilde{\mathbf{A}}(\phi_c) - \mathbf{A}(\phi_c, \rho)\|_F^2, \quad (19.16)$$

where  $\mathbf{A}(\phi_c, \rho) \in \mathbb{C}^{N \times Q}$  denotes a matrix composed of a collection of array steering vectors describing  $\rho$  in a closed-form, such as in (19.12), and  $\|\cdot\|_F$  denotes the Frobenius norm of a matrix. In some cases, a closed-form solution to (19.16) may be obtained. For example, using the mutual coupling matrix  $\mathbf{C}$  in place of  $\rho$  in (19.16) yields the following solution:

$$\hat{\mathbf{C}} = \tilde{\mathbf{A}}(\phi_c) \mathbf{A}^\dagger(\phi_c), \quad (19.17)$$

where we have used the array model (19.11) and assumed that  $\mathbf{A}(\phi_c)$  has full row-rank. Notation  $(\cdot)^\dagger$  in (19.17) denotes the Moore-Penrose pseudo-inverse of a matrix. Similarly, by letting  $\rho$  denote angle-independent gain and phase errors such as  $\mathbf{A}(\phi_c, \rho) = \text{diag}\{\rho\} \mathbf{A}(\phi_c)$ , we obtain the following solution:

$$\hat{\rho} = \tilde{\mathbf{a}}_n(\phi_c) \mathbf{a}_n^H(\phi_c) \left( \mathbf{a}_n(\phi_c) \mathbf{a}_n^H(\phi_c) \right)^{-1}, \quad (19.18)$$

where  $\tilde{\mathbf{a}}_n(\phi_c) \in \mathbb{C}^{1 \times Q}$  and  $\mathbf{a}_n(\phi_c) \in \mathbb{C}^{1 \times Q}$  denote the  $n$ th row of  $\tilde{\mathbf{A}}(\phi_c)$  and  $\mathbf{A}(\phi_c)$ , respectively. In case one is dealing with electrically large uniform linear arrays, the mutual coupling matrix may be approximated by a banded matrix. Recall that mutual coupling is typically inversely proportional to inter-element spacing thus, such an effect may be negligible among sensors located at both ends of the linear array. A least-squares estimator to such a structured mutual coupling matrix may also be found in a closed-form. Computationally efficient solutions may employ appropriate  $LU$ -factorization and back-substitution methods [46, Chapter 4]. A summary of model-driven calibration is given in Table 19.1.

Alternatively, we may jointly estimate array nonidealities and wavefield parameters from the output of the sensor array [11–13]. For example, joint estimation of the nonidealities  $\rho$  and azimuth angles of  $P$  sources ( $\phi \in \mathbb{R}^{P \times 1}$ ) generating a propagating wavefield may be accomplished by employing the following nonlinear least-squares estimator [12]

$$\{\hat{\phi}, \hat{\rho}\} = \arg \min_{\hat{\phi}, \hat{\rho}} \text{Tr} \left\{ \Pi_A^\perp(\phi, \rho) \hat{\mathbf{R}}_X \right\}. \quad (19.19)$$

Typically, criterion in (19.19) is minimized in an alternating manner between the array nonidealities  $\rho$  and the DoAs  $\phi$  [47]. Since (19.19) is highly nonlinear, and potentially with multiple local minima,

**Table 19.1** Steps of (Deterministic) Model-Driven Calibration**Offline**

- Step 1* Acquire the array calibration matrix  $\tilde{\mathbf{A}}(\phi_c) \in \mathbb{C}^{N \times Q}$   
*Step 2* Specify an array steering vector model  $\mathbf{a}(\phi, \rho) \in \mathbb{C}^{N \times 1}$   
describing both array nonidealities and wavefield parameters  
*Step 3* Estimate the array nonidealities  $\hat{\rho}$

**Online**

- Step 4* Acquire the output of the real array  $\mathbf{x}(k)$ ,  $k = 1, \dots, L$   
*Step 5* Apply DoA estimators and beamforming techniques  
using the resulting array steering vector  $\mathbf{a}(\phi, \hat{\rho})$

the minimization should be initialized with “good enough” initial values so that the global minimum can be attained. Note that the array nonidealities  $\rho$  are typically nuisance parameters. The statistical performance of the wavefield parameter estimates may suffer due to the higher dimension of the parametric model as well as estimation errors in the nuisance parameters.

The main issue with auto-calibration techniques is that of parameter identifiability [6]. In general, both  $\rho$  and  $\phi$  cannot be uniquely estimated unless a nonlinear sensor array is employed and additional assumptions regarding sensors’ locations as well as DoAs are made [12, 13]. For example, if the array orientation is unknown the DoAs may not be uniquely estimated since they represent the angles relative to the orientation of the sensor array. Alternatively, one may assume that the array nonidealities are random parameters with a known prior distribution, and employ Bayesian estimators. This is discussed next.

### 3.19.5.2 Bayesian approach

In the previous section we have seen that the identifiability problem in auto-calibration techniques could be alleviated by making additional assumptions regarding the nonidealities or wavefield parameters. One such an assumption considers the array nonidealities to be random parameters with a known prior distribution. In addition to alleviating the identifiability problem, such an assumption allows one (at least in principle) to “integrate out” the array nonidealities and focus on the wavefield parameters, instead [24]. For example, in mass production of sensor arrays one could model array elements’ misplacements due to manufacturing errors as bivariate Gaussian distributed and proceed with Bayesian type of estimators for the wavefield parameters [7, 14].

One such an estimator is the generalized weighted subspace fitting (GWSF) algorithm proposed in [7]. It extends the MODE [48] and WSF [49] by taking into account prior information (first and second moments) of the array nonidealities in an optimal manner. It provides asymptotically efficient estimates provided the assumption on the prior Gaussian distribution is valid and array parameterization is known. The DoA estimates obtained by the GWSF are given by [7]

$$\hat{\phi} = \arg \min_{\phi} \hat{\mathbf{e}}_S^H \mathbf{\Pi}_A^\perp \mathbf{W} \mathbf{\Pi}_A^\perp \hat{\mathbf{e}}_S \quad (19.20)$$

**Table 19.2** Steps of Auto-Calibration Techniques

<b>Offline</b>	
<i>Step 1</i>	Specify an array steering vector model $\mathbf{a}(\phi, \rho) \in \mathbb{C}^{N \times 1}$ describing both array nonidealities and wavefield parameters
<i>Step 2</i>	Model the array nonidealities $\rho$ either as unknown deterministic or random parameters with a prior distribution
<b>Online</b>	
<i>Step 3</i>	Acquire the output of the real array $\mathbf{x}(k), k = 1, \dots, L$
<i>Step 4</i>	Determine the number of signals generating the received wavefield
<i>Step 5</i>	Estimate wavefield parameters $\hat{\phi}$ by employing a suitable estimator e.g., the NNLS or GWSF estimator
<i>Step 6 (optional)</i>	Estimate the array nonidealities $\hat{\rho}$ and apply beamforming techniques using the resulting array steering vector $\mathbf{a}(\phi, \hat{\rho})$

where  $\hat{\mathbf{e}}_S^H = \text{vec} \left\{ [\hat{\mathbf{E}}_S^T \hat{\mathbf{E}}_S^H]^T \right\} \in \mathbb{C}^{2NP' \times 1}$  and  $\Pi_A^\perp \in \mathbb{C}^{2NP' \times 2NP'}$  denotes a projection matrix onto the orthogonal complement of

$$\bar{\mathbf{A}}(\phi, \rho) = \begin{bmatrix} (\mathbf{I}_{P'} \otimes \mathbf{A}(\phi, \rho)) & \mathbf{0} \\ \mathbf{0} & (\mathbf{I}_{P'} \otimes \mathbf{A}^c(\phi, \rho)) \end{bmatrix}. \quad (19.21)$$

Furthermore,  $\hat{\mathbf{W}} \in \mathbb{C}^{2NP' \times 2NP'}$  in (19.20) denotes a (positive-definite) weighting matrix that ensures asymptotically minimum variance unbiased estimates and the superscript  $(\cdot)^c$  denotes complex-conjugate. The first and second-order moments of the array nonidealities enter criterion (19.20) through  $\hat{\mathbf{W}}$ ; see [7] for details. Criterion (19.20) is an asymptotic approximation of the *maximum a posteriori* estimator for simultaneous estimation of array and wavefield parameters [14]. It may be implemented by means of polynomial rooting techniques when the nominal array steering vector has a form similar to that of an ideal ULA. A summary of auto-calibration techniques is given in Table 19.2.

The main difficulty with the Bayesian approach is related to the well-known problem of choosing appropriate prior distributions.<sup>1</sup> Assumed prior knowledge may not exist or it may be difficult to express in the form of a pdf. Moreover, specifying a parametric model for the array nonidealities may be very challenging in practice, similarly to the deterministic approach. In case of uncertainties in the array elements' locations, the nominal steering vector model may be obtained by visual inspection of the real-world sensor array [7]. However, when considering cross-polarization effects, sensors with individual beampatterns and mutual coupling, such a procedure is of little help in determining a nominal array steering vector.

<sup>1</sup>This may be alleviated by means of uncertainty sets on the array steering vector, to be discussed in Section 3.19.7. For example, uncertainties in the array elements' locations may be bounded by the array aperture, thus avoiding the difficulties that may arise in deriving Bayesian type of estimators with truncated prior distributions.

In practice, array calibration measurements may be necessary even in auto-calibration techniques. Hence, it may be worth considering alternative techniques for dealing with array nonidealities that assume array calibration measurements but do not suffer from the difficulties in specifying explicit formulations for the nonidealities. This is discussed in the next section.

### 3.19.6 Data-driven techniques

Data-driven techniques take into account all array nonidealities simultaneously through array calibration measurements or synthesized array response using e.g., electromagnetic simulation software. These techniques do not require any explicit formulation describing the array nonidealities in a closed-form. Examples of nonidealities that may be handled with data-driven techniques include mutual coupling, individual beampatterns, mounting platform reflections, and cross-polarization effects. This section describes the array interpolation technique [15] and wavefield modeling principle [20], also known as manifold separation technique [18].

In particular, array interpolation technique may be understood as a linear interpolation method that fits array calibration measurements with some ideal array steering vector model. The manifold separation techniques stems from the wavefield modeling principle and can be seen as an orthogonal expansion in Fourier basis (in azimuth-angle processing) of each array element. The expansion coefficients describe the array nonidealities in a combined manner and may be estimated from array calibration measurements.

#### 3.19.6.1 Local interpolation of the array calibration matrix

The columns of the array calibration matrix  $\tilde{\mathbf{A}}$  discussed in Section 3.19.4 describe the array response, with the combined effects due to array nonidealities, to a set of angles. The angular grid employed in array calibration measurements is typically sparse due to time and cost limitations. Hence, optimal array processing methods using the array calibration matrix may loose their high-resolution properties and suffer from SOI cancellation effects. Perhaps the most intuitive approach to overcome such a limitation consists in interpolating the array calibration matrix using local basis functions such as splines.

Let  $\alpha(\phi)$  denote a vector composed of local basis functions such as splines or other polynomial functions. The practitioner should choose local basis functions with desirable properties such as smoothness, differentiability, and minimum energy. Also, let  $\widehat{\mathbf{C}}_n$  denote a coefficient matrix obtained by interpolating the array calibration matrix  $\tilde{\mathbf{A}}$  over an angular sector  $\mathcal{C}_n$  using  $\alpha(\phi)$ .  $\mathcal{C}_n$  may correspond to two or more columns of  $\tilde{\mathbf{A}}$ . Using local basis as the interpolating functions leads to the following piece-wise estimate of the real-world array steering vector:

$$\hat{\mathbf{a}}(\phi) = \begin{cases} \widehat{\mathbf{C}}_1\alpha(\phi), & \phi \in \mathcal{C}_1, \\ \widehat{\mathbf{C}}_2\alpha(\phi), & \phi \in \mathcal{C}_2, \\ \vdots \\ \widehat{\mathbf{C}}_n\alpha(\phi), & \phi \in \mathcal{C}_n. \end{cases} \quad (19.22)$$

**Table 19.3** Steps of Local Interpolation Technique

<b>Offline</b>	
<b>Step 1</b>	Acquire the array calibration matrix $\tilde{\mathbf{A}}(\phi_{\mathbf{C}}) \in \mathbb{C}^{N \times Q}$ and divide it into $L$ angular sectors $\{\tilde{\mathbf{A}}_l \in \mathbb{C}^{N_v \times Q/L}\}_{l=1}^L$
<b>Step 2</b>	Specify a nominal array model $\mathbf{a}_0(\phi) \in \mathbb{C}^{N \times 1}$
<b>Step 3</b>	Find the $L$ coefficient matrices $\{\mathbf{C}_l(\phi) \in \mathbb{C}^{N \times N_l}\}_{l=1}^L$
<b>Online</b>	
<b>Step 4</b>	Acquire the output of the real array $\mathbf{x}(k)$ , $k = 1, \dots, K$
<b>Step 5</b>	Apply DoA estimators and beamforming techniques to $\mathbf{x}(k)$ using the nominal array model $\mathbf{a}_0(\phi)$ and coefficient matrices $\{\mathbf{C}_l(\phi)\}_{l=1}^L$

Alternatively, one may use a nominal array steering vector  $\mathbf{a}_0(\phi)$  in place of  $\boldsymbol{\alpha}(\phi)$ , and use a local basis expansion for the coefficients matrices in (19.22), instead. More precisely, we may have [6]:

$$\hat{\mathbf{a}}(\phi) = \begin{cases} \widehat{\mathbf{C}}_1(\phi)\mathbf{a}_0(\phi), & \phi \in \mathcal{C}_1, \\ \widehat{\mathbf{C}}_2(\phi)\mathbf{a}_0(\phi), & \phi \in \mathcal{C}_2, \\ \vdots \\ \widehat{\mathbf{C}}_n(\phi)\mathbf{a}_0(\phi), & \phi \in \mathcal{C}_n, \end{cases} \quad (19.23)$$

where the  $n$ th matrix  $\widehat{\mathbf{C}}_n(\phi)$  is modeled as

$$\widehat{\mathbf{C}}_n(\phi) = \text{diag}\{\widetilde{\mathbf{C}}_n \boldsymbol{\alpha}(\phi)\}. \quad (19.24)$$

Here,  $\widetilde{\mathbf{C}}_n$  denotes a local coefficients matrix; see [6] and references therein for details. The rationale for (19.23) is that  $\widehat{\mathbf{C}}_n(\phi)$  is typically a smoother function of the angles than the array response, thus allowing for sparser calibration grids than those employed in (19.22). A summary of local interpolation technique is given in Table 19.3.

Expressions (19.22) and (19.23) may be useful in cases when the sensor array is deployed on an environment where the sources are known to be confined to an angular sector. If this is not the case, and the sources may span the whole angular region, using local interpolation techniques may lead to a significant increase on the computationally complexity of array processing methods and may compromise the convergence rate of gradient-based optimization techniques. For example, using (19.22) with the root-MUSIC algorithm requires finding the roots of  $n$  different polynomials while the maximum step-size of gradient-based methods is limited by the size of each angular sector  $\mathcal{C}_n$ . Hence, wavefield modeling and manifold separation, discussed later in this section, are generally preferred when a sensor array is deployed on an environment where the sources may span the whole angular region.

### 3.19.6.2 Array interpolation technique

The array interpolation technique was originally proposed in [15] and further studied in, e.g., [16, 17, 50]. The idea is to linearly transform the real-world array so that its response approximates that of a specified ideal array, such as an ULA, known as *virtual array*. The steering vector model of the virtual array needs to be specified by the designer, and it is typically based on some array processing technique. For example, if the virtual array is that of an ULA, one may employ polynomial rooting techniques for DoA estimation. We note that for UCAs a technique called Beamspace transform may be also employed [51]. However, we do not consider Beamspace transform in this chapter due to the restriction on the array geometry; see [51] and references therein.

Let  $\tilde{\mathbf{A}}(\phi_c) \in \mathbb{C}^{N \times Q}$  denote the calibration measurement matrix of the real-world array and  $\mathbf{A}_v(\phi_c) \in \mathbb{C}^{N_v \times Q}$  a collection of steering vectors of the virtual array. In its simplest form, the array interpolation technique consists in determining the transformation matrix  $\mathbf{T} \in \mathbb{C}^{N_v \times N}$  that minimizes the following quadratic error:

$$\mathbf{T} = \arg \min_{\mathbf{T}} \|\mathbf{T}\tilde{\mathbf{A}}(\phi_c) - \mathbf{A}_v(\phi_c)\|_F^2. \quad (19.25)$$

The solution to (19.25) is well-known to be

$$\mathbf{T} = \mathbf{A}_v(\phi_c)\tilde{\mathbf{A}}^\dagger(\phi_c). \quad (19.26)$$

Given the output of the real-world array  $\mathbf{x}(k) \in \mathbb{C}^{N \times 1}$ , the output of the virtual array  $\mathbf{y}(k) \in \mathbb{C}^{N_v \times 1}$  and its sample covariance matrix  $\hat{\mathbf{R}}_Y \in \mathbb{C}^{N_v \times N_v}$  are found as  $\mathbf{y}(k) = \mathbf{T}\mathbf{x}(k)$  and  $\hat{\mathbf{R}}_Y = \mathbf{T}\hat{\mathbf{R}}_X\mathbf{T}^H$ , respectively. Array processing techniques may then be developed for the virtual array and be employed to real-world arrays without explicitly modeling their nonidealities. We note that the virtual array should be designed so that both  $\mathbf{T}^H$  and  $\hat{\mathbf{R}}_Y$  are of full column-rank. In case the condition  $N_v \leq N$  does not lead to a full-rank virtual array covariance matrix, the virtual array should be re-designed. A summary of array interpolation techniques is given in Table 19.4.

**Table 19.4** Steps of Array Interpolation Technique

#### Offline

- Step 1** Acquire the array calibration matrix  $\tilde{\mathbf{A}}(\phi_c) \in \mathbb{C}^{N \times Q}$  and divide it into  $L$  angular sectors  $\{\tilde{\mathbf{A}}_l \in \mathbb{C}^{N_v \times Q/L}\}_{l=1}^L$
- Step 2** Specify  $L$  virtual array models  $\{\mathbf{a}_l(\phi) \in \mathbb{C}^{N_v \times 1}\}_{l=1}^L$  and generate the corresponding virtual array calibration matrices  $\{\mathbf{A}_v^l \in \mathbb{C}^{N_v \times Q/L}\}_{l=1}^L$
- Step 3** Find the  $L$  transformation matrices  $\{\mathbf{T}_l \in \mathbb{C}^{N_v \times N}\}_{l=1}^L$   
If  $\mathbf{T}_l$  is ill-conditioned go to Step 2 and design a new virtual array model

#### Online

- Step 4** Acquire the output of the real array  $\mathbf{x}(k)$ ,  $k = 1, \dots, K$
- Step 5** Obtain  $L$  virtual array outputs  $\mathbf{y}_l(k) = \mathbf{T}_l\mathbf{x}(k)$ ,  $l = 1, \dots, L$
- Step 6** Apply DoA estimators and beamforming techniques to  $\{\mathbf{y}_l(k)\}_{l=1}^L$  using the virtual array models  $\{\mathbf{a}_l(\phi)\}_{l=1}^L$

The array interpolation technique has two important drawbacks. First, the virtual array, including its configuration, orientation, number of elements, and inter-element spacing, needs to be specified by the designer. Even though this offers some versatility for employing low-complexity DoA estimators or beamforming techniques with arbitrary array configurations (e.g., root-MUSIC algorithm using virtual ULAs), designing virtual arrays is always a heuristic and subjective task. For example, suppose one wants to establish a performance bound such as the widely used Cramér-Rao lower Bound (CRB) for a specific real-world array [25]. In order to take into account the array nonidealities it would be appealing to use array calibration measurements and array interpolation techniques for guaranteeing the tightness of such a bound. However, the resulting CRB depends on the choice of the user-specified virtual array configuration and its parameterization, even though the true physical array remains unmodified. Typically, the specified virtual array employed in array interpolation does not provide insight into the achievable performance by an array built in practice.

Second, the quadratic error in the mapping (19.25) is typically very large if one considers the whole range of angles at once, i.e.,  $\phi_c \in [0, 2\pi]$ . In order to reduce such an error, array interpolation technique typically proceed by dividing the visible region of the real-world array into angular sectors and optimizing a transformation matrix for each sector. In case the sensor array is deployed on an environment where the sources are known *a priori* to be confined to an angular sector such a requirement of array interpolation techniques is not a serious limitation. However, in environments where the sources may span the whole angular region, array interpolation technique require sector-by-sector processing, which is known to be sensitivity to out-of-sector sources [52]. Moreover, it may also need a prohibitively large number of sectors in azimuth and elevation processing.

Array interpolation techniques are typically more flexible than local interpolation of the array calibration matrix. For example, one cannot (in general) employ the ESPRIT algorithm with arbitrary array configurations using (19.22) simply by choosing a “shift-invariant” local basis vector. Typically, the approximation  $\mathbf{a}(\phi) \approx \mathbf{C}_n \boldsymbol{\alpha}(\phi)$  is not shift-invariant. On the other hand, array interpolation techniques requires more design parameters than local interpolation methods. Finally, array interpolation techniques, and to some extent local interpolation methods, typically interpolate exactly all of the measured data including calibration measurement noise. Next, we show that wavefield modeling and manifold separation can be formulated in a model fitting approach in order to minimize the contribution of calibration measurement noise.

### 3.19.6.3 Wavefield modeling principle and manifold separation technique

The wavefield modeling principle was proposed in the seminal work of Doron and Doron [20–22]. It has been further studied and applied to high-resolution direction finding in [18, 19], and extended to vector-fields such as completely polarized electromagnetic wavefields in [53].

Let us first recall some results regarding propagating wavefields and wave equation. For the sake of clarity, we consider scalar-fields such as acoustic pressure fields, narrowband signals, and drop the carrier term  $e^{j\omega t}$ . The extension to completely polarized EM wavefields is briefly described in Section 3.19.8. Let  $\Psi(t, \mathbf{r}) \in \mathbb{C}$  represent a (scalar) wavefield propagating in the  $xy$ -plane and  $\mathbf{r} \in \mathbb{R}^{2 \times 1}$  denote a point in 2-D Euclidean space. The propagating wavefield takes the form of  $\Psi(t, \mathbf{r}) = \sum_{p=1}^P s_p(t) e^{-j\mathbf{r}^T \mathbf{k}_p}$  in the case of  $P$  far-field point sources or  $\Psi(t, \mathbf{r}) = s(t) \int_{\mathcal{S}^1} \varrho(\phi) e^{-j\mathbf{r}^T \mathbf{k}} d\phi$  in the case of a (far-field) spatially distributed source.  $\varrho(\phi) \in \mathbb{C}$  denotes a density function and  $\mathbf{k} \in \mathbb{R}^{2 \times 1}$  is known as the direction vector

since it is a function of  $\phi$ . Spatially distributed sources may be caused by scattering nearby the transmitter. Wavefields of time-harmonic nature, i.e., that have a representation in terms of Fourier integral, may be written as  $\Psi(t, \mathbf{r}) = \sum_{m=-\infty}^{+\infty} \psi_m(t) h_m(\mathbf{r})$ , where  $\{h_m(\mathbf{r}) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$  and  $\{\psi_m(t) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$  denote an orthogonal set of *spatial* basis functions and the coefficients of the expansion, respectively. An important outcome of such an expansion is that the coefficients  $\{\psi_m(t) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$  uniquely describe the spatial characteristics of the propagating wavefield, such as the DoAs of the sources generating the wavefield, in addition to the transmitted signals. For example, by letting  $\{h_m(\mathbf{r}) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$  denote circular wave functions, the  $m$ th wavefield coefficient is given by  $\psi_m(t) = s(t) \int_{S^1} e^{im\phi} \varrho(\phi) d\phi$  for a spatially distributed source and  $\psi_m(t) = \sum_{p=1}^P s_p(t) e^{im\phi_p}$  for  $P$  point-sources.

Let us now assume that the employed real-world array satisfies the superposition principle (see Section 3.19.3). Then, the wavefield modeling principle shows that the array output, at a given frequency, is a linear function of the wavefield coefficients  $\{\psi_m(t) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$ . In particular, after discretization the narrowband array output in (19.1) can be written as

$$\mathbf{x}(k) = \mathbf{G}\boldsymbol{\psi}(k) + \mathbf{n}(k), \quad (19.27a)$$

$$= \mathbf{G} \sum_{p=1}^P \mathbf{d}(\phi_p) s_p(k) + \mathbf{n}(k), \quad (19.27b)$$

where  $\mathbf{G} \in \mathbb{C}^{N \times M}$  denotes the so-called array sampling matrix and  $\boldsymbol{\psi}(k) \in \mathbb{C}^{M \times 1}$  contains the (discretized) wavefield coefficients  $\{\psi_m(k) \in \mathbb{C}\}_{m=-\infty}^{+\infty}$ . Recall that, due to the circular wave basis function employed by the spatial decomposition of the propagating wavefield,  $\mathbf{d}(\phi) \in \mathbb{C}^{M \times 1}$  in (19.27b) is a Vandermonde vector composed of Fourier basis. Hence,  $\mathbf{d}(\phi)$  is called basis functions vector. In case of a spatially distributed source, the sum in (19.27a) is replaced by an integral over the angles, and weighted by  $\varrho(\phi)$ . The number of coefficients employed in (19.27a) and (19.27b) is denoted by  $M$ . Exact equality in (19.27a) and (19.27b) is achieved with  $M = \infty$  but in practice a (very) accurate approximation of the array output can be obtained with a relatively small  $M$  (see Figure 19.7).

The result in (19.27a) and (19.27b) shows that the (noise-free) array output can be decomposed into two parts. One, represented by the array sampling matrix, characterizes the employed sensor array and it is independent of the wavefield. The second part, represented by the basis functions vector, characterizes the propagating wavefield and it is independent of the employed sensor array. In fact, a corollary of the wavefield modeling principle shows that the array steering vector may be decomposed as

$$\mathbf{a}(\phi) = \mathbf{G} \mathbf{d}(\phi). \quad (19.28)$$

The result in (19.28) is known as manifold separation technique [18] and reveals an interesting interpretation for the array sampling matrix. It represents the spatial Fourier spectrum of the array steering vector

$$\mathbf{G} = \int_{S^1} \mathbf{a}(\phi) \mathbf{d}^H(\phi) d\phi, \quad (19.29)$$

where each row of the array sampling matrix contains the spatial Fourier coefficients of each array element. Hence, the array output (at each frequency) can be seen as the product between the spatial Fourier spectrum of the array steering vector and that of the propagating wavefield.

The array sampling matrix fully and uniquely characterizes a given sensor array since it contains the coefficients of an orthogonal spectral decomposition of the corresponding array steering vector.

For example,  $\mathbf{G}$  contains information about the array configuration, sensors beampatterns (gain and phase response), mutual coupling, cross-polarization effects, mounting platform reflections, etc. In short, it contains all the effects that can be represented by the array steering vector  $\mathbf{a}(\phi)$ . Closed-form expressions for the array sampling matrix for some ideal sensor arrays can be found in [20]. However,  $\mathbf{G}$  may also be estimated in a non-parametric manner from array calibration measurements, without explicit formulations for the array nonidealities. In particular, the estimated array sampling matrix  $\widehat{\mathbf{G}}$  obtained as

$$\widetilde{\mathbf{G}} = \widetilde{\mathbf{A}}(\phi_c) \mathbf{F}, \quad (19.30a)$$

$$\widehat{\mathbf{G}} = \widetilde{\mathbf{G}} \mathbf{S} \quad (19.30b)$$

is known as effective aperture distribution function (EADF) [18,33]. In (16.30),  $\mathbf{F} \in \mathbb{C}^{Q \times Q}$  denotes the unitary discrete Fourier transform (DFT) matrix and  $\mathbf{S} \in \mathbb{N}^{Q \times M}$  a selection matrix that optimally trades-off between complexity and accuracy of the resulting array steering vector model in (19.28).  $\mathbf{S}$  may be estimated using state-of-the-art model order estimators [54].

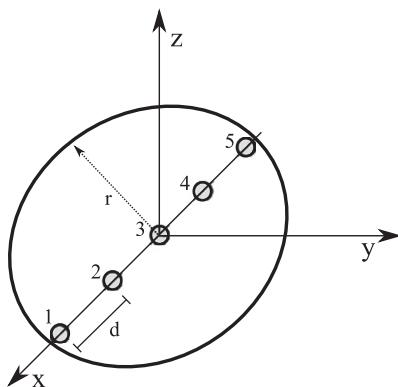
We have mentioned that exact equality in (19.27b, 19.27a) and (19.28) requires that  $\mathbf{G}$  is composed of infinitely many columns.<sup>2</sup> One may feel that such a requirement makes the wavefield modeling principle, or manifold separation technique, of theoretical interest only. The crucial property of the array sampling matrix that makes both wavefield modeling principle and manifold separation technique practical is known as superexponential decay. In particular, the magnitude of the columns of the sampling matrix,  $[\mathbf{G}]_m$ , decay faster than exponential (i.e., superexponential) as  $m \rightarrow \infty$  beyond  $|m| = \kappa r$ .  $\kappa$  and  $r$  denote the angular wavenumber and radius of the smallest sphere enclosing the array structure (centered at the assumed coordinate system), respectively.

In practice, the superexponential property tells us that (19.27b, 19.27a) and (19.28) are (very) well approximated by a few columns  $M$  of the array sampling matrix since the norm-convergence rate of the expansion in (19.28) is faster than exponential [18,20]; see Figure 19.7. The superexponential property may be understood by interpreting sensor arrays as spatial filters. In fact,  $\mathbf{G}$  may be seen as the array's spatial frequency response, where the passband, stopband, and cutoff frequencies are given by  $|m| < \kappa r$ ,  $|m| > \kappa r$ , and  $|m| = \kappa r$ , respectively. For example, sensor arrays with large aperture have increased resolution since they sample the propagating wavefield over a large area. This is reflected on the array sampling matrix by an increase of the passband  $|m| < \kappa r$  ( $r$  increases). This leads to an increase of the amount of energy received from such a wavefield since a large number of wavefield coefficients  $\{\psi_m \in \mathbb{C}\}_{m=-\infty}^{+\infty}$  are taken into account by the employed sensor array.

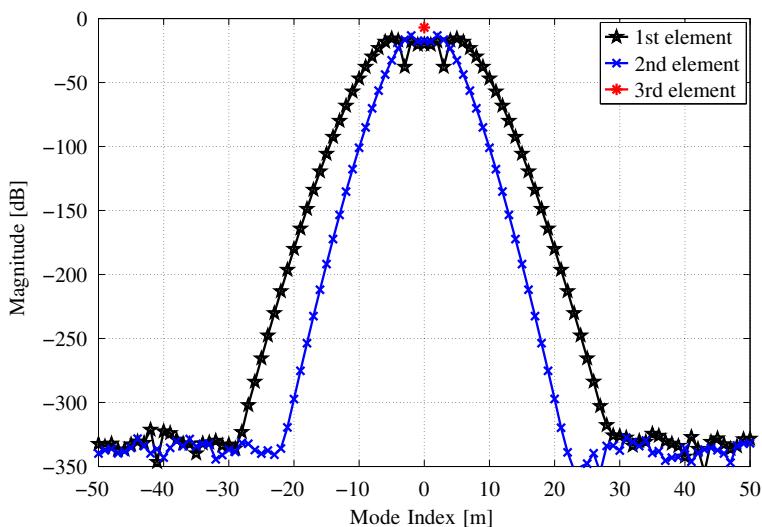
In the following, the main concepts and properties of wavefield modeling principle and manifold separation technique are illustrated using an ideal ULA. We emphasize that using ULAs does not limit the generality of the discussion. We employ it for the sake of clarity here. Let us consider the 5-element ULA from Figure 19.5, where the smallest sphere (a circle in this case) enclosing the array structure is depicted as well. Figure 19.6 illustrates three rows of the array sampling matrix, i.e., the spatial Fourier coefficients of the first three array elements. The ideal ULA is composed of omnidirectional elements with an inter-element spacing of  $\lambda/2$ . In Figure 19.7a, the norm of each column of the array sampling matrix,  $\|[\mathbf{G}]_m\|$ , is illustrated for two ideal ULAs with inter-element spacings of  $d = \lambda/2$  and  $d = \lambda/4$ .

---

<sup>2</sup>In the limiting case of an infinitely small aperture the array sampling matrix is finite.

**FIGURE 19.5**

Ideal uniform linear array with an inter-element spacing denoted by  $d$ . The smallest sphere (a circle in this case) enclosing the array structure is depicted as well. The concept of smallest sphere provides a measure of the array aperture, including the mounting platform.

**FIGURE 19.6**

Three first rows of the array sampling matrix corresponding to the first three elements of the ideal ULA depicted in Figure 19.5. They represent the spatial Fourier coefficients of the first three array elements. The coefficients of the third array element are zero except at  $m = 0$  whereas that of the outer elements exhibit the superexponential property.

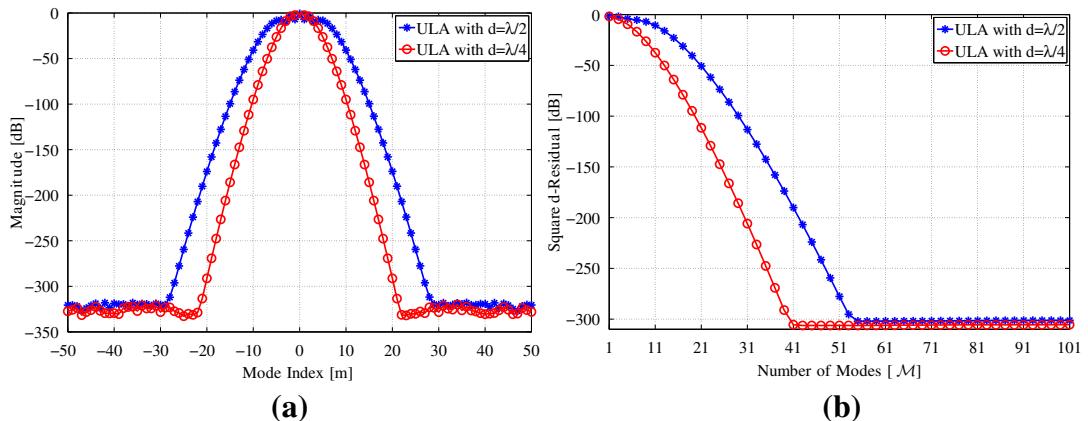


FIGURE 19.7

In (a) the norm of each column of the array sampling matrix for two ideal ULAs with different inter-element spacing. In (b) the average squared-residual of the manifold separation technique as a function of the number of modes. The saturation floor observed at  $\sim -300$  dB is due to arithmetic precision of Matlab. The superexponential property of the array sampling matrix is a consequence of the finite aperture of sensor arrays.

Finally, Figure 19.7b, illustrates the following (average) squared-residual:

$$\frac{1}{2\pi} \int_{S^1} \| \mathbf{a}(\phi) - \sum_{m=-(M-1)/2}^{(M-1)/2} [\mathbf{G}]_m [\mathbf{d}(\phi)]_m \|^2 d\phi, \quad (19.31)$$

as a function of  $M$ , the number of columns of  $\mathbf{G}$ .

The simulation results show that the “passband” of the array sampling matrix increases for large apertures.<sup>3</sup> In the limiting case of an infinitely small aperture, such as the omnidirectional array element located at the center of the coordinate system in Figure 19.5, the array sampling matrix is finite (see Figure 19.6). This is because only the magnitude response of such an array needs to be modeled (the first spatial harmonic in the case of omnidirectional elements) since the relative phase across such an aperture is zero.

A summary of the wavefield modeling principle/manifold separation technique is given in Table 19.5, where we assume that array calibration measurements are taken over the whole angular region, i.e.,  $\phi_c \in [0, 2\pi]$ . We emphasize that such an assumption does not limit the generality of the wavefield modeling principle/manifold separation technique since it is simply related with the choice of orthogonal basis functions  $\{h_m(\mathbf{r})\}_{m=-\infty}^{+\infty}$ , employed for decomposing the propagating wavefield. More precisely, wavefield modeling principle/manifold separation technique are also applicable when only partial array calibration measurements are acquired, or when the sources are known *a priori* to be confined to an angular sector. The superexponential property of the sampling matrix is also retained in such cases [20].

<sup>3</sup>The correct term should be electric dimensions since the superexponential property is inversely proportional to the wavelength, in addition to the relationship with the physical dimension of the array.

**Table 19.5** Steps of Wavefield Modeling Principle/Manifold Separation Technique

**Offline**

- Step 1* Acquire the array calibration matrix  $\tilde{\mathbf{A}}(\phi_c) \in \mathbb{C}^{N \times Q}$ , with  $\phi_c \in [0, 2\pi)$
- Step 2* Find  $\tilde{\mathbf{G}} \in \mathbb{C}^{N \times Q}$  by taking a  $Q$ -point FFT of  $\tilde{\mathbf{A}}(\phi_c)$
- Step 3* Estimate the array sampling matrix  $\hat{\mathbf{G}} \in \mathbb{C}^{N \times M}$  by employing state-of-the-art model order estimators such as normalized MDL

**Online**

- Step 4* Acquire the output of the real-world array  $\mathbf{x}(k)$ ,  $k = 1, \dots, K$
- Step 5* Apply DoA estimators and beamforming techniques to  $\mathbf{x}(k)$  using the array model  $\mathbf{a}(\phi) = \hat{\mathbf{G}}\mathbf{d}(\phi)$

The wavefield modeling principle/manifold separation technique may also be employed as a complement to the auto-calibration techniques from Section 3.19.5 as well as to array interpolation technique and uncertainty sets (to be discussed in the next section). For example, in the case of uncertainties in the array elements' positions, one may parameterize the array sampling matrix using the closed-form expressions in [20] and employ the auto-calibration techniques described in Section 3.19.5. The advantage of such an approach is the simplicity of using Fourier basis regardless of the configuration of the real-world array. One may also employ manifold separation technique with array interpolation technique to determine the conditions under which the real array output can be transformed to that of a virtual array, up to a specified mapping error [20]. Such conditions can be found with or without sector-by-sector processing. Finally, we note that many antenna measurement techniques, including the spherical near-field approach, are based on wavefield modeling [32, 33]. This suggests that the wavefield modeling principle/manifold separation technique may play a fundamental role in any sensor array application.

### 3.19.7 Robust methods

Robust array processing procedures trade-off optimality to high reliability when the assumptions on the nominal signal or noise model do not hold [8, 55–57]. The derivation of an optimal method is typically performed using strict assumptions on the sensor array model, propagation environment, source signals, as well as statistical properties of interference and noise. A shortcoming of the optimal array processing procedures is that they are extremely sensitive even to small deviations from the assumed model. In reality the underlying assumptions may not be valid and a significant degradation from the optimal performance is experienced.

Robust methods acknowledge that the assumptions on signal model may not be valid and do not try to recover the nonidealities from the observed or calibration data. Instead, they aim at bounding the influence of array modeling errors so that small departures from the nominal model lead only to small errors in the array processor output. One robust approach is based on *minimax* design which protects

against a worst possible scenario, i.e., it is the best in the worst case. Hence, robust methods are also applicable if the conditions where the calibration was done are very different from the conditions where the sensor array system is deployed.

One simple approach is to assume that these errors are random, independent, and zero mean. Hence, they just decrease the SNR by increasing the noise variance. Since the array covariance matrix plays an important role in most array processing algorithms it is of interest to study how such a quantity behaves in nonstandard conditions. Assuming random errors, the perturbations may be expressed using the array covariance matrix as follows [58]:

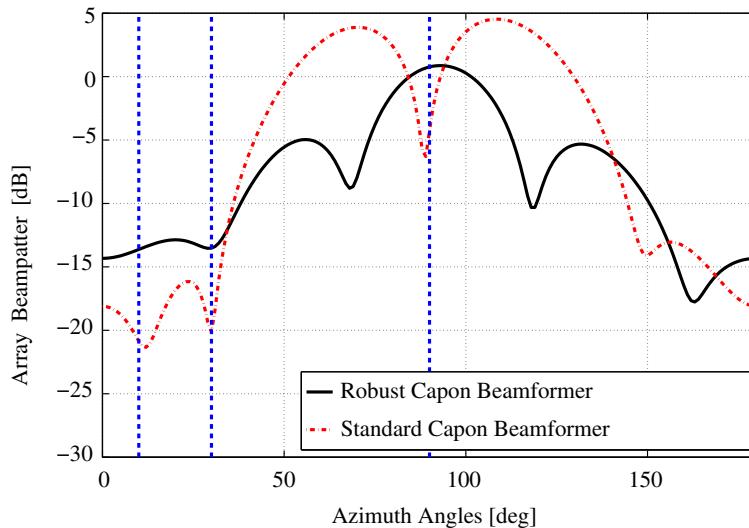
$$\widehat{\mathbf{R}}_X = (\mathbf{I} + \Delta)[(\mathbf{A}(\phi) + \widetilde{\mathbf{A}}(\phi))\mathbf{R}_S(\mathbf{A}(\phi) + \widetilde{\mathbf{A}}(\phi))^H + \sigma^2(\mathbf{I} + \widetilde{\mathbf{R}}_N)](\mathbf{I} + \Delta)^H,$$

where matrix  $\Delta \in \mathbb{C}^{N \times N}$  is associated with errors that influence both the signal and noise components of the data. Departures from the nominal array response are included in the matrix  $\widetilde{\mathbf{A}}(\phi) \in \mathbb{C}^{N \times P}$ . This matrix contains the perturbations in element positions, errors in gain and phase responses of the sensors and mutual coupling. The term  $\widetilde{\mathbf{R}}_N$  describes the deviation of the noise covariance matrix from the nominal matrix  $\mathbf{I}$  (noise is commonly assumed to be zero-mean complex-circular white Gaussian distributed and  $\mathbf{R}_N$  an identity matrix). The effects of the above perturbations to high resolution DoA estimation as well as signal and noise subspaces may be studied using the first-order analysis introduced in [58] or by using tools from matrix perturbation theory. In the following, we will consider an example of robust beamforming that is optimized for the worst case scenario. A more detailed discussion can be found in chapter Adaptive and Robust Beamforming of this book.

### 3.19.7.1 Robust technique based on worst-case performance optimization and uncertainty sets

We have seen that steering vectors of real-world arrays are not exactly known in practice and that lack of knowledge or uncertainties about the array model may lead to a significant performance degradation in most array processors. For example, one may steer energy towards unwanted directions, cancel the SOI as well as amplify interfering sources or jammers; see Figure 19.8. The wavefield modeling principle/manifold separation technique aims at optimally describing (in the MSE sense, for example) nonidealities from array calibration measurements as well as incorporating such nonidealities into DoA estimators and beamforming techniques. However, real-world arrays may also be subject to nonidealities that change as a function of time or cannot be measured in controlled environments. For example, random fluctuations of the array response due to nearby scatterers or motion of the platform where the array is mounted are not captured in the calibration stage. In addition, imprecise knowledge of the DoAs may also lead to a performance degradation of beamforming techniques, a problem known as pointing angle or look direction errors. The idea of worst-case performance optimization techniques [59, 60] employing so-called uncertainty sets [23, 61], is to develop array processing techniques that are robust to general uncertainties in the steering vector of the real-world array.

Let us denote the exact (but unknown) steering vector of the real-world array by  $\mathbf{a}(\phi) \in \mathbb{C}^{N \times 1}$ . Also, let  $\tilde{\mathbf{a}}(\phi) \in \mathbb{C}^{N \times 1}$  denote the known but imprecise array steering vector, where  $\tilde{\mathbf{a}}(\phi) = \mathbf{a}(\phi) + \Delta\mathbf{a}$ .  $\tilde{\mathbf{a}}(\phi)$  may be acquired from array calibration measurements, EM simulation software or may represent an ideal ULA, for example. The vector  $\Delta\mathbf{a} \in \mathbb{C}^{N \times 1}$  denotes the uncertainty we have about  $\mathbf{a}(\phi)$  and may be due to errors in pointing angle or imprecise gain of array elements. Worst-case performance

**FIGURE 19.8**

Example of array beampattern of both standard and robust Capon beamformer in case of steering vector uncertainties. The standard Capon beamformer cancels the SOI, located at  $\phi = 90^\circ$ .

optimization techniques proceed by assuming that the norm of the uncertainty vector can be bounded by a *known*  $\epsilon > 0$  such that  $\|\Delta \mathbf{a}\|^2 \leq \epsilon$ . This is equivalent to assuming that the imprecise steering vector  $\tilde{\mathbf{a}}(\phi)$  belongs to a known hyper-ellipsoid centered at  $\mathbf{a}(\phi)$  [23, 61]:

$$(\tilde{\mathbf{a}}(\phi) - \mathbf{a}(\phi))^H \mathbf{C}^{-1} (\tilde{\mathbf{a}}(\phi) - \mathbf{a}(\phi)) \leq 1. \quad (19.32)$$

Here,  $\mathbf{C} \in \mathbb{C}^{N \times N}$  denotes a known positive-definite matrix that characterizes the shape of the ellipsoid and defines the maximum uncertainty we have about  $\mathbf{a}(\phi)$ . Ellipsoids of the form of (19.32) are called nondegenerate ellipsoids. If the uncertainty ellipsoids fall into a lower-dimensional space they are called flat (or degenerate) ellipsoids [61]. Flat ellipsoids are employed to make the uncertainty set as tight as possible but require more prior information about the maximum uncertainty than that of the nondegenerate ellipsoids.

Worst-case performance optimization techniques have been mostly used in the context of robust minimum variance beamforming [23, 59–61]. The resulting robust beamformers can be shown to belong to the class of diagonal loading approaches. In fact, the optimal diagonal loading value can be found exactly from  $\mathbf{C}$ , unlike most of the ad hoc diagonal loading techniques [8]. However, the value of  $\epsilon$  and the shape of the uncertainty ellipsoid are typically specified by the designer. See e.g., [62] for alternative approaches to find the loading factor automatically.

Let us consider the eigenvalue decomposition of the sample covariance matrix  $\widehat{\mathbf{R}}_X = \widehat{\mathbf{E}} \widehat{\boldsymbol{\Lambda}} \widehat{\mathbf{E}}^H$ , with  $\{\lambda_n \in \mathbb{R}\}_{n=1}^N$  denoting the corresponding eigenvalues, and assume that  $\mathbf{C} = \epsilon \mathbf{I}_N$ . The array steering

vector found by employing robust techniques based on uncertainty sets is [23]

$$\mathbf{a}(\phi) = \tilde{\mathbf{a}}(\phi) - \widehat{\mathbf{E}}(\mathbf{I}_N + \gamma(\phi)\widehat{\boldsymbol{\Lambda}})^{-1}\widehat{\mathbf{E}}^H\tilde{\mathbf{a}}(\phi), \quad (19.33)$$

where  $\gamma(\phi) \in \mathbb{R}$  is the solution of [23]

$$\sum_{n=1}^N \frac{|[\mathbf{z}]_n|^2}{(1 + \gamma(\phi)\lambda_n)^2} = \epsilon. \quad (19.34)$$

Here,  $\mathbf{z} = \widehat{\mathbf{E}}^H\tilde{\mathbf{a}}(\phi)$  and  $\gamma(\phi) > 0$  can be shown to belong to the following interval [23]

$$\frac{\|\tilde{\mathbf{a}}(\phi)\| - \sqrt{\epsilon}}{\lambda_1\sqrt{\epsilon}} \leq \gamma(\phi) \leq \min \left\{ \left( \frac{1}{\epsilon} \sum_{n=1}^N \frac{|[\mathbf{z}]_n|^2}{\lambda_n^2} \right)^{1/2}, \frac{\|\tilde{\mathbf{a}}(\phi)\| - \sqrt{\epsilon}}{\lambda_N\sqrt{\epsilon}} \right\}. \quad (19.35)$$

We may now use the steering vector model in (19.33) with the Capon beamformer expression from Section 3.19.2 in order to have a robust approach for both beamforming as well as DoA estimation. A summary of this robust technique is provided in Table 19.6.

Typically, robust techniques based on uncertainty sets have a prohibitively large computational complexity. For example, the value  $\gamma(\phi)$  in (19.33) needs to be found for every angle and may not be found offline since it is a function of  $\widehat{\mathbf{E}}$ . An exception that is worth mentioning is the robust beamformer of [60], where the beamformer weights may be updated snapshot-by-snapshot using subspace tracking techniques. However, such an approach is still not practical in DoA estimation since it requires finding a principal eigenvector for each grid point of the spectral search.

**Table 19.6** Steps of Robust Technique based on Uncertainty Sets

**Offline**

*Step 1* Acquire the array calibration matrix  $\widetilde{\mathbf{A}}(\phi_C) \in \mathbb{C}^{N \times Q}$

*Step 2* Determine the maximum uncertainty ellipsoid by choosing an appropriate matrix  $\mathbf{C}$

**Online**

*Step 3* Acquire the output of the real-world array  $\mathbf{x}(k)$ ,  $k = 1, \dots, K$

*Step 4* Find the EVD of  $\widehat{\mathbf{R}}_X = 1/K \sum_{k=1}^K \mathbf{x}(k)\mathbf{x}(k)^H$

*Step 5* Determine  $\gamma(\phi)$  by solving (19.34)

*Step 6* Apply the Capon beamformer to  $\mathbf{x}(k)$

using the array model  $\mathbf{a}(\phi) = \tilde{\mathbf{a}}(\phi) - \widehat{\mathbf{E}}(\mathbf{I}_N + \gamma(\phi)\widehat{\boldsymbol{\Lambda}})^{-1}\widehat{\mathbf{E}}^H\tilde{\mathbf{a}}(\phi)$

### 3.19.8 Array processing examples

#### 3.19.8.1 DoA estimation using an ideal uniform linear array and manifold separation technique

Let us consider that a propagating wavefield, generated by two equi-power and uncorrelated point-sources, impinge on an ideal ULA from  $\phi = [90^\circ, 85^\circ]$ , measured from the endfire of the array. The ULA is composed of 5 omnidirectional elements with an inter-element spacing of  $d = \lambda/2$ . We employ the root-MUSIC [63] and element-space (ES) root-MUSIC [18] algorithms. Since the employed sensor array is composed of omnidirectional elements the array sampling matrix  $\mathbf{G}$  may be found in a closed-form [20]. Figure 19.9 illustrates the performance of both root-MUSIC and ES root-MUSIC algorithms in terms of root mean-squared error (RMSE). Only the results for  $\phi_1 = 90^\circ$  are illustrated but similar results are obtained for  $\phi_2 = 85^\circ$ . Figure 19.9a illustrates the RMSE as a function of snapshots, with SNR = 10 dB, while Figure 19.9b illustrates the RMSE as a function of SNR, with  $K = 20$  snapshots. The number of columns of  $\mathbf{G}$  employed by the ES root-MUSIC is  $\mathcal{M} = 25$ . Results show that the ES root-MUSIC has a performance very close to the root-MUSIC even though the former employs an approximation of the array steering vector.

The complexity of the ES root-MUSIC is, of course, higher than that of the root-MUSIC algorithm, and if the employed sensor array is indeed an ideal ULA one should resort to the standard root-MUSIC algorithm. However, if the task is estimate DoAs using real-world arrays with imperfections, the ES

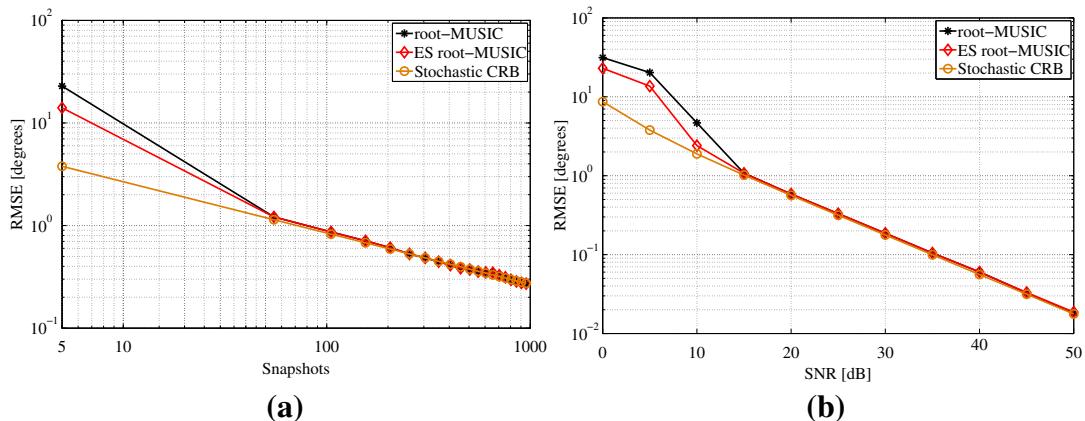
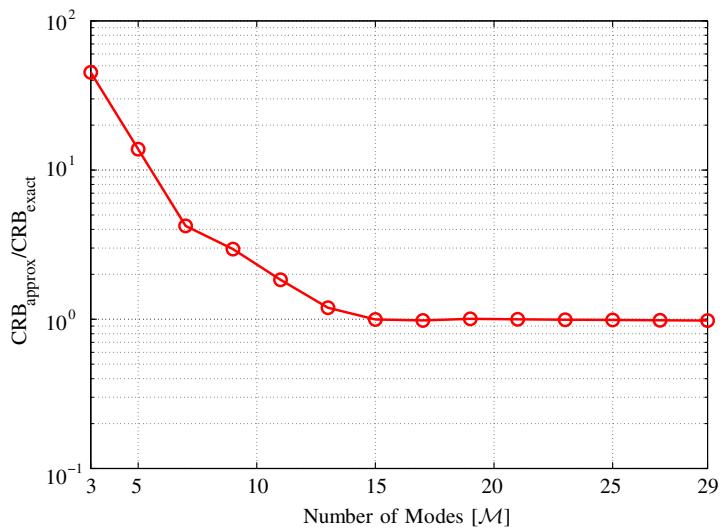


FIGURE 19.9

Performance of both standard root-MUSIC and ES root-MUSIC algorithms as a function of (a) snapshots and (b) SNR. A propagating wavefield, generated by two equi-power and uncorrelated sources, impinge on an ideal ULA from  $\phi = [90^\circ, 85^\circ]$ . Only the results for  $\phi_1 = 90^\circ$  are illustrated but similar results are obtained for  $\phi_2 = 85^\circ$ . The number of columns of  $\mathbf{G}$  employed by the ES root-MUSIC is  $\mathcal{M} = 25$ . Results show that there is practically no loss of performance, even though the ES root-MUSIC employs an approximation of the array steering vector.

root-MUSIC is a very attractive choice. Note that the complexity of the ES root-MUSIC algorithm may be reduced by means of Schur factorization and Arnoldi iterations [64].

Let us now suppose that the only information we have about a real-world array is by means of its array measurement matrix  $\tilde{\mathbf{A}}(\phi_c)$ , and the stochastic CRB expression for array processing in (19.9) needs to be found. One may employ the manifold separation technique in (19.28) with (19.9) in order to find an approximate CRB expression that is tight even for real-world arrays with nonidealities, excluding the low SNR regime where the CRB is not a tight bound in general. To illustrate this, let us assume that the array measurement matrix of the 5-element ULA from Figure 19.5 is obtained from  $Q = 30$  points ( $\phi_c \in [-\pi, \pi]$ ) with  $\text{SNR}_{\text{cal}} = 20$  dB. The EADF of such an ULA is obtained from the array measurement matrix using Eq. (16.30). The resulting array steering vector model, Eq. (19.28), is employed with the CRB expression in (19.9) in order to obtain an approximate CRB expression. Figure 19.10 illustrates the ratio between the approximate and exact stochastic CRBs as a function of the number of columns of  $\tilde{\mathbf{G}}$ . The results have been averaged over  $\phi \in [-70^\circ, 70^\circ]$  and 100 realizations of calibration noise. The approximate CRB expression obtained by employing the manifold separation principle is accurate since the ratio  $\text{CRB}_{\text{approx}}/\text{CRB}_{\text{exact}}$  converges to unity. We also note that the accuracy of the approximate CRB expression is related to the electrical dimension of the employed



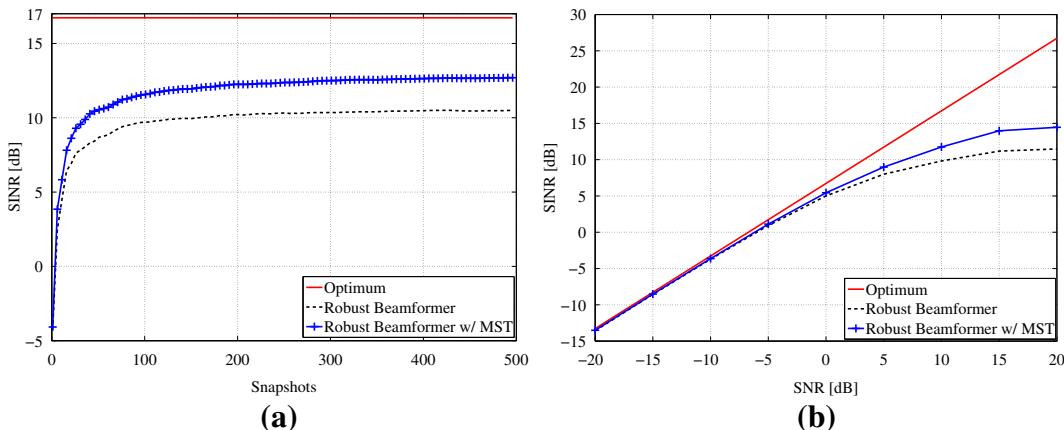
**FIGURE 19.10**

Accuracy of the approximate CRB expression obtained by employing the manifold separation technique. The ratio between the approximate and exact CRBs is illustrated as a function of the number of columns of the EADF. The approximate CRB expression is accurate since the ratio  $\text{CRB}_{\text{approx}}/\text{CRB}_{\text{exact}}$  converges to one around  $M = 15$ , which is a measure of the electrical dimension of the employed sensor array. The approximate CRB expression obtained by employing the manifold separation technique is tight and very useful for real-world arrays with imperfections.

sensor array. In fact, the ratio  $\text{CRB}_{\text{approx}}/\text{CRB}_{\text{exact}}$  converges to unity around  $M = 15$  modes which, as discussed in Section 3.19.6, is the point where the magnitude of the array sampling matrix starts decaying superexponentially.

### 3.19.8.2 Robust beamforming using array calibration

Let us consider that a propagating wavefield, generated by three uncorrelated point-sources, impinge on an ideal ULA that is identical to the one employed in the previous example. The SOI and interfering sources impinge on the sensor array from  $\phi_S = 90^\circ$  and  $(\phi_1 = 30^\circ, \phi_2 = 10^\circ)$ , respectively. The signal powers of the interferers are  $\sigma_1^2 = \sigma_2^2 = 20$  dB. We consider two sources of uncertainty in the array steering vector  $\mathbf{a}(\phi_S)$ , namely due to array calibration noise and error in the look direction. In particular, we employ the array measurement matrix  $\tilde{\mathbf{A}}(\phi_C)$  of the ULA, found with an  $\text{SNR}_{\text{cal}} = 20$  dB and  $Q = 181$  calibration points. In addition, we consider that there is an error of two degrees in the DoA of the SOI. We employ the robust Capon beamformer, obtained by using (19.33) in (19.7), with the imprecise array steering vector  $\tilde{\mathbf{a}}(\phi_S + 2^\circ)$  found directly from  $\tilde{\mathbf{A}}(\phi_C)$ , and by employing the manifold separation technique, i.e., using the EADF  $\widehat{\mathbf{G}}$ . The uncertainty ellipsoid is fixed with  $\mathbf{C} = \epsilon \mathbf{I}_N$ , where  $\epsilon = 0.1$ . Figure 19.11a illustrates the array output SINR as a function of snapshots with  $\sigma_S^2 = 10$  dB while Figure 19.11b illustrates the array output SINR as a function of the SNR of the SOI with  $K = 100$  snapshots. Employing the manifold separation technique leads to improved performance since it attenuates calibration measurement noise [18].



**FIGURE 19.11**

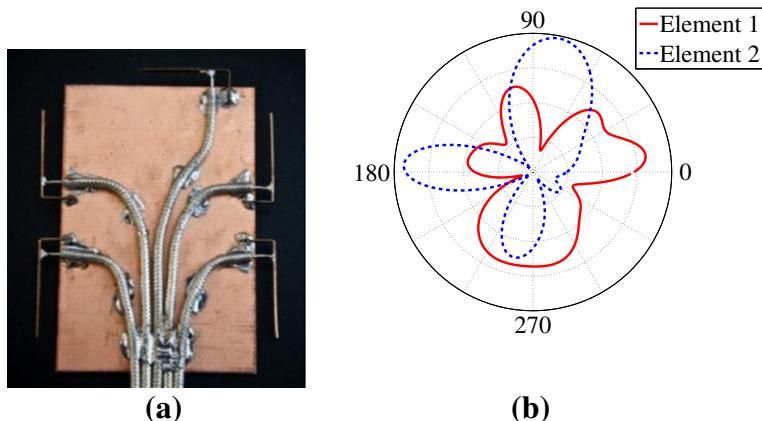
Performance of the robust beamformer based on uncertainty sets in terms of (a) array output SINR as a function of snapshots and (b) array output SINR as a function of SNR of the SOI. Two sources of uncertainty in the array steering vector are considered, namely due to array calibration noise and error in the look direction. Employing the manifold separation technique leads to improved performance since it allows reducing calibration measurement noise while modeling array nonidealities.

### 3.19.8.3 Polynomial rooting techniques for real-world arrays with nonidealities

Let us now consider DoA estimation algorithms based on polynomial rooting techniques that can be employed regardless of the array configuration and nonidealities. Assume that a propagating wavefield, generated by two point-sources, is observed by the real-world array from Figure 19.12 [65]. The sources are located in the far-field of the sensor array and their DoAs are  $\phi = [60^\circ, 65^\circ]^T$ . The sources are assumed to be located in the same plane ( $xy$ -plane) as the employed antenna array. Only the array measurement matrix  $\tilde{\mathbf{A}}(\phi_c)$  of the real-world array is known. The interpolated root-MUSIC algorithm [15] as well as the ES root-MUSIC [18] and the Fourier-domain (FD) root-MUSIC [50] are used to provide the angle estimates as follows.

The interpolated root-MUSIC algorithm is employed by first determining 12 array mapping matrices  $\{\mathbf{T}_s\}_{s=1}^{12}$ , one for each  $30^\circ$ -sector, from  $\tilde{\mathbf{A}}(\phi_c)$ . Each of the 12 virtual ULAs (one per sector) is located at the center of the minimum circle enclosing the employed sensor array and oriented so that its broadside corresponds to the middle of each sector. The virtual ULAs are composed of five elements with an inter-element spacing of  $\lambda/2$ , so that the aperture of the virtual ULAs is maximized while guaranteeing a small condition number of the mapping matrices. The remaining steps of the interpolated root-MUSIC algorithm are implemented as described in [15].

The ES root-MUSIC algorithm is implemented by first determining the EADF, denoted by  $\hat{\mathbf{G}}$  as described in (16.6.9). This includes taking the FFT of the array measurement matrix and determining the number of columns  $M$  by means of a model order estimation technique such as MDL [54]. After determining the noise subspace  $\hat{\mathbf{E}}_N$  of the sample covariance matrix  $\hat{\mathbf{R}}_X$ , the DoA estimates are found from the phase-angles of the  $P$  roots closest to the unit circle (either inside or outside the unit circle) of



**FIGURE 19.12**

- (a) Five-element Inverted-F Antenna (IFA) array built for direction-finding purposes using handheld terminals. Its dimensions, center frequency and bandwidth are  $6 \text{ cm} \times 4 \text{ cm}$ , 3.5 GHz, and  $\sim 100 \text{ MHz}$ , respectively.
- (b) Magnitude of the radiation pattern of two elements of the array. Each antenna has a different radiation characteristic which is far from the ideal omnidirectional pattern. Courtesy of the Department of Radio Science and Technology, Aalto University, Finland.

the following polynomial

$$c_{2\mathcal{M}-2} z^{2\mathcal{M}-2} + \cdots + c_1 z + c_0 = 0. \quad (19.36)$$

Note that the coefficients in (19.36) may be found in a computationally efficient manner using the FFT:

$$\mathbf{c} = \text{FFT} \left\{ \sum_{n=1}^{N-P} \left| \text{IFFT} \left\{ \widehat{\mathbf{G}}^H [\widehat{\mathbf{E}}_N]_n \right\} \right|^2 \right\}, \quad (19.37)$$

where  $\mathbf{c} = [c_{\mathcal{M}} \dots c_0 c_{2\mathcal{M}-2} \dots c_{\mathcal{M}+1}]^T \in \mathbb{C}^{(2\mathcal{M}-1) \times 1}$ .

The FD root-MUSIC algorithm is implemented by first determining the sampled MUSIC nullspectrum  $\mathbf{f}(\phi_c) \in \mathbb{C}^{Q \times 1}$ :

$$\mathbf{f}(\phi_c) = \text{diag} \left\{ \widetilde{\mathbf{A}}(\phi_c)^H \widehat{\mathbf{E}}_N \widehat{\mathbf{E}}_N^H \widetilde{\mathbf{A}}(\phi_c) \right\}. \quad (19.38)$$

Then, the coefficients of the FD root-MUSIC polynomial

$$g_{2\mathcal{M}-2} z^{2\mathcal{M}-2} + \cdots + g_1 z + g_0 = 0 \quad (19.39)$$

are obtained as

$$\tilde{\mathbf{g}} = \text{FFT}\{\mathbf{f}(\phi_c)\}, \quad (19.40a)$$

$$\mathbf{g} = \mathbf{S} \tilde{\mathbf{g}}, \quad (19.40b)$$

where  $\mathbf{g} = [g_{\mathcal{M}} \dots g_0, g_{2\mathcal{M}-2} \dots g_{\mathcal{M}+1}]^T \in \mathbb{C}^{(2\mathcal{M}-1) \times 1}$  and the selection matrix  $\mathbf{S} \in \mathbb{C}^{(2\mathcal{M}-1) \times Q}$  is obtained in a similar fashion as with the ES root-MUSIC.

Figure 19.13 illustrates the performance of the three polynomial rooting approaches in terms of RMSE as a function of (a) snapshots with SNR = 20 dB and (b) SNR with  $K = 100$  snapshots. Only the results for  $\phi_2 = 65^\circ$  are shown but similar results are obtained for  $\phi_1 = 60^\circ$ . The algorithms exploit a noise-free array measurement matrix with  $Q = 180$  calibration points. The degree of both ES root-MUSIC and FD root-MUSIC polynomials are equal with  $\mathcal{M} = 21$ . Results show that the ES root-MUSIC as well as the FD root-MUSIC have similar performance and are closed to the stochastic CRB while the estimates obtained by the interpolated root-MUSIC have large bias and excess variance.

Figure 19.14 illustrates the sensitivity of both ES root-MUSIC and FD root-MUSIC algorithms, for a fixed  $\mathcal{M} = 21$ , to (a) calibration SNR ( $\text{SNR}_{\text{cal}}$ ) with  $Q = 31$  and (b) calibration points  $Q$  with  $\text{SNR}_{\text{cal}} = 30$  dB. Two equi-power sources impinge on the 5-element IFA array from  $\phi = [5^\circ, 15^\circ]^T$  with an SNR = 20 dB and  $K = 100$ . Only the results for  $\phi_1 = 5^\circ$  are illustrated but similar results are obtained for  $\phi_2 = 15^\circ$ . Results show that the ES root-MUSIC algorithm outperforms the FD root-MUSIC when the array measurement matrix is corrupted by calibration noise.

### 3.19.8.4 Azimuth, elevation, and polarization estimation

Let us now consider the general case of estimating both azimuth  $\phi$  and elevation  $\theta$  angles of a completely polarized EM propagating wavefield. Typically, the polarization of the wavefield is unknown and needs to be estimated along with the DoAs. The narrowband array output model is now given by

$$\mathbf{x}(k) = [\mathbf{A}_\phi(\phi, \theta) \quad \mathbf{A}_\theta(\phi, \theta)] \mathbf{V}(\gamma, \beta) \mathbf{s}(k) + \mathbf{n}(k), \quad (19.41)$$

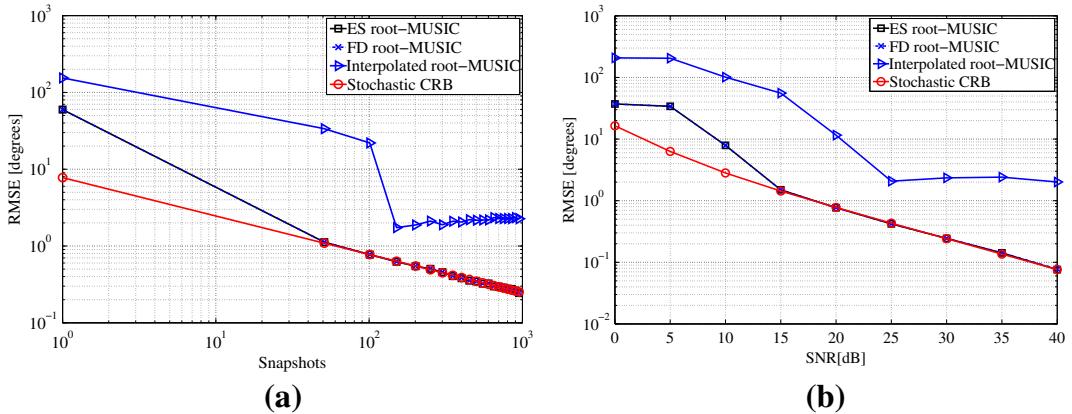


FIGURE 19.13

Performance of interpolated root-MUSIC, ES root-MUSIC, and FD root-MUSIC algorithms as a function of (a) snapshots and (b) SNR. A propagating wavefield, generated by two equi-power and uncorrelated sources located at  $\phi = [60^\circ, 65^\circ]^T$ , impinge the sensor array from Figure 19.12. Only the results for  $\phi_2 = 65^\circ$  are illustrated but similar results are obtained for  $\phi_1 = 60^\circ$ . The array calibration matrix is noise-free and  $Q = 180$  points have been taken. Results show that the ES root-MUSIC as well as the FD root-MUSIC have similar performance and are closed to the stochastic CRB while the estimates obtained by the interpolated root-MUSIC have large bias and excess variance.

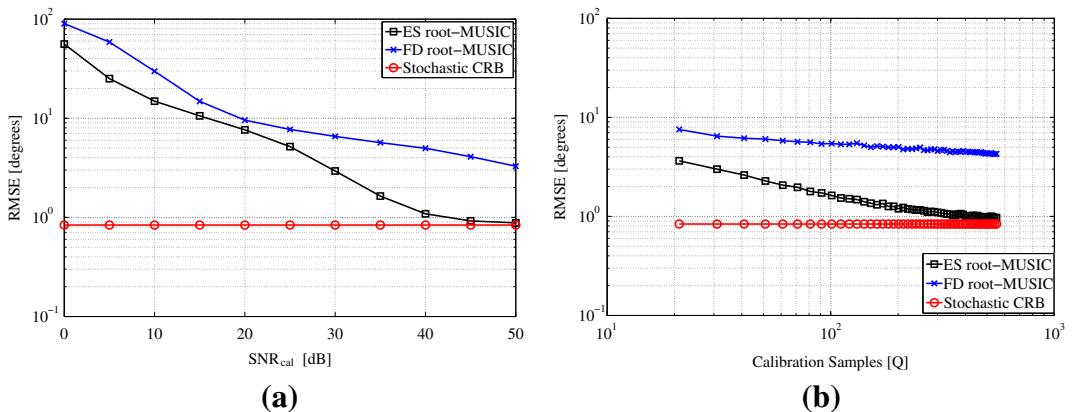


FIGURE 19.14

Sensitivity of ES root-MUSIC and FD root-MUSIC algorithms, with a fixed  $M = 21$ , to (a) calibration SNR ( $\text{SNR}_{\text{cal}}$ ) with  $Q = 31$  and (b) number of calibration points  $Q$  with  $\text{SNR}_{\text{cal}} = 30$  dB. Two equi-power sources impinge on the 5-element IFA array from  $\phi = [5^\circ, 15^\circ]^T$  with an SNR = 20 dB and  $K = 100$ . Results show that the ES root-MUSIC algorithm outperforms the FD root-MUSIC when the array measurement matrix is corrupted by calibration noise.

where  $\mathbf{A}_\phi(\boldsymbol{\phi}, \boldsymbol{\theta}), \mathbf{A}_\theta(\boldsymbol{\phi}, \boldsymbol{\theta}) \in \mathbb{C}^{N \times P}$  denote the array steering matrices due to a vertical and horizontal polarized wavefields, respectively.  $\mathbf{V}(\boldsymbol{\gamma}, \boldsymbol{\beta}) = [\mathbf{V}_\phi(\boldsymbol{\gamma}), \mathbf{V}_\theta(\boldsymbol{\gamma}, \boldsymbol{\beta})]^T \in \mathbb{C}^{2P \times P}$  describes the polarization of the wavefield and it is typically given by

$$\mathbf{V}_\phi(\boldsymbol{\gamma}) = \text{diag}\{v_\phi(\gamma_1), \dots, v_\phi(\gamma_P)\}, \quad (19.42a)$$

$$\mathbf{V}_\theta(\boldsymbol{\gamma}, \boldsymbol{\beta}) = \text{diag}\{v_\theta(\gamma_1, \beta_1), \dots, v_\theta(\gamma_P, \beta_P)\}, \quad (19.42b)$$

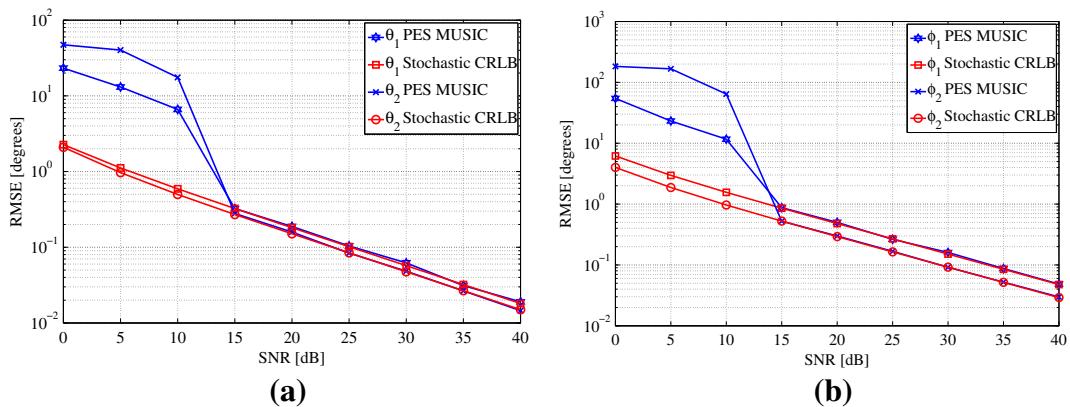
where  $v_\phi(\gamma_p) = \cos(\gamma_p)$  and  $v_\theta(\gamma_p, \beta_p) = \sin(\gamma_p)e^{j\beta_p}$ . The parameters  $\gamma$  and  $\beta$  describe the polarization ellipse of the received Electric-field and take values over  $0 \leq \gamma \leq \pi/2$  and  $-\pi \leq \beta < \pi$ , respectively. The techniques described in Sections 3.19.5 and 3.19.6 may now be employed to (19.41) by modeling the array steering vector  $\mathbf{a}(\theta, \phi, \gamma, \beta) \in \mathbb{C}^{N \times 1}$  either in a parametric or non-parametric manner.

In particular, the wavefield modeling principle/manifold separation technique for completely polarized EM wavefields consists in employing so-called vector spherical harmonics, instead of Fourier basis in (19.27b, 19.27a) and (19.28). The motivation for using vector spherical harmonics follows from the fact that such basis functions form a complete and orthonormal set on the 2-sphere, similarly to Fourier basis in azimuth-only processing. However, vector spherical harmonics, which have a rather cumbersome algebraic form, are less attractive than Fourier basis for the purposes of array processing and may be subject to numerical instabilities. In fact, a formulation of the wavefield modeling principle/manifold separation technique involving 2-D Fourier basis may be found in [53], where it is employed the so-called equivalence matrix [19]. A summary of the wavefield modeling principle/manifold separation technique for completely polarized EM wavefields is given in Table 19.7.

Let us consider that a propagating EM wavefield, generated by two equi-power far-field point-sources, is received by the 5-element IFA from Figure 19.12. The DoAs and polarization parameters

**Table 19.7** Steps of Wavefield Modeling Principle/Manifold Separation Technique for Completely Polarized EM Wavefields

<b>Offline</b>	
<i>Step 1</i>	Acquire the array calibration matrices $\tilde{\mathbf{A}}_\phi(\boldsymbol{\phi}_C, \boldsymbol{\theta}_C), \tilde{\mathbf{A}}_\theta(\boldsymbol{\phi}_C, \boldsymbol{\theta}_C)$ with $\boldsymbol{\phi}_C \in [0, 2\pi]$ and $\boldsymbol{\theta}_C \in [0, \pi]$
<i>Step 2</i>	Find $\tilde{\mathbf{G}}$ by taking a discrete vector spherical harmonic transform of $\tilde{\mathbf{A}}_\phi(\boldsymbol{\phi}_C, \boldsymbol{\theta}_C), \tilde{\mathbf{A}}_\theta(\boldsymbol{\phi}_C, \boldsymbol{\theta}_C)$
<i>Step 3</i>	Estimate the array sampling matrix $\hat{\mathbf{G}}$ by employing model order estimation techniques
<i>Step 4</i>	Employ the equivalence matrix $\Xi$ and find $\hat{\Gamma} = \hat{\mathbf{G}}\Xi$
<b>Online</b>	
<i>Step 5</i>	Acquire the output of the real array $\mathbf{x}(k), k = 1, \dots, K$
<i>Step 6</i>	Apply DoA estimators and beamforming techniques to $\mathbf{x}(k)$ using the array model $\hat{\mathbf{a}}(\theta, \phi, \gamma, \beta) = \hat{\Gamma}(\mathbf{I}_2 \otimes \mathbf{d}(\theta, \phi))\mathbf{v}(\gamma, \beta)$ , where $\mathbf{d}(\theta, \phi) = \mathbf{d}(\phi) \otimes \mathbf{d}(\theta)$



**FIGURE 19.15**

Statistical performance of the PES MUSIC algorithm using the 5-element IFA from Figure 19.12. The DoAs and polarization parameters are ( $\theta_1 = 20^\circ$ ,  $\theta_2 = 40^\circ$ ), ( $\phi_1 = 25^\circ$ ,  $\phi_2 = 60^\circ$ ) and ( $\gamma_1 = 10^\circ$ ,  $\gamma_2 = 20^\circ$ ), ( $\beta_1 = 10^\circ$ ,  $\beta_2 = 50^\circ$ ), respectively. The PES MUSIC algorithm takes into account array nonidealities and has a performance close to the stochastic CRB.

are ( $\theta_1 = 20^\circ$ ,  $\theta_2 = 40^\circ$ ), ( $\phi_1 = 25^\circ$ ,  $\phi_2 = 60^\circ$ ), and ( $\gamma_1 = 10^\circ$ ,  $\gamma_2 = 20^\circ$ ), ( $\beta_1 = 10^\circ$ ,  $\beta_2 = 50^\circ$ ), respectively.  $K = 100$  snapshots are acquired at the array output. The array response was obtained from an EM simulation software; see [65] for details. The manifold separation technique is employed for determining the array steering vector model along with its nonidealities, as described in Table 19.7. The polarimetric element-space (PES) MUSIC algorithm [53] is used for estimating both DoAs and polarization parameters of the sources generating the wavefield.

Figure 19.15 illustrates the statistical performance of the PES MUSIC in terms of RMSE as a function of SNR. Only the angle estimates are shown but similar results are obtained for the polarization parameters. Results show that the PES MUSIC algorithm has a performance close to the stochastic CRB since it takes into account array nonidealities.

### 3.19.9 Conclusion

In this chapter, array signal processing in face of array nonidealities was addressed. Real-world arrays are always subject to nonidealities such as mutual coupling, mounting platform reflections, cross-polarization effects, sensors with individual beampatterns as well as sensors' position errors. We have seen that such nonidealities typically lead to a significant performance degradation as well as loss of optimality of DoA estimators and beamforming techniques.

Various techniques for dealing with array nonidealities have been described. They are classified as model-driven techniques, data-driven, and robust methods. Model-driven techniques use an explicit formulation describing each nonideality, which may be challenging to specify, and include auto-calibration techniques as well as so-called parametric calibration methods. Data-driven techniques do not assume any explicit formulation for the nonidealities but employ array calibration measurements. They cap-

ture the nonidealities implicitly by using basis function expansion, interpolation, approximation or nonparameteric estimation techniques. Data-driven techniques include local interpolation of the array calibration matrix, array interpolation as well as manifold separation techniques. Finally, robust methods try to bound the influence of nonidealities in the estimation process instead of trying to capture them. Typically, robust methods trade-off optimality for reliability.

Extensive array processing examples have been included. They explain in detail how array processing techniques may deal with array nonidealities by employing data-driven techniques as well as robust methods.

We note that the various techniques for dealing with array nonidealities described in this chapter have different features and may complement each other. Future research work may thus be based on combining auto-calibration techniques and robust methods with manifold separation.

*Relevant Theory:* Signal Processing Theory, Statistical Signal Processing and Array Signal Processing

See Vol. 1, Chapter 11 Parametric Estimation

See this Volume, Chapter 2 Model Order Selection

See this Volume, Chapter 16 Performance Bounds and Statistical Analysis of DOA Estimation

---

## References

- [1] P. Stoica, R. Moses, *Introduction to Spectral Analysis*, Wiley, 1997.
- [2] H. Krim, M. Viberg, Two decades of array signal processing, *IEEE Signal Process. Mag.* (1996) 67–94.
- [3] S. Wijnholds, S. van der Tol, R. Nijboer, A. van der Veen, Calibration challenges for future radio telescopes, *IEEE Signal Process. Mag.* 27 (1) (2010) 30–42.
- [4] D. Gesbert, C. van Rensburg, F. Tosato, F. Kaltenberger, Multiple antenna techniques, in: S. Sesia, I. Toufik, M. Baker (Eds.), *LTE—The UMTS Long Term Evolution: From Theory to Practice*, John Wiley and Sons, 2009, pp. 243–283 (Chapter 11).
- [5] F. Belloni, V. Ranki, A. Kainulainen, A. Richter, Angle-based indoor positioning system for open indoor environments, in: *Workshop on Positioning, Navigation and Communication*, 2009, pp. 261–265.
- [6] M. Viberg, M. Lanne, A. Lundgren, Calibration in array processing, in: T. Tuncer, B. Friedlander (Eds.), *Classical and Modern Direction-of-Arrival Estimation*, Academic Press, Burlington, MA, USA, 2009, pp. 93–124 (Chapter 3).
- [7] M. Jansson, A. Swindlehurst, B. Ottersten, Weighted subspace fitting for general array error models, *IEEE Trans. Signal Process.* 46 (9) (1998) 2484–2498.
- [8] A. Gershman, Robust adaptive beamforming in sensor arrays, *Int. J. Electron. Commun.* 53 (6) (1999) 305–314.
- [9] R. Mailloux, Array failure correction with a digitally beamformed array, *IEEE Trans. Antennas Propag.* 44 (12) (1996) 1543–1550.
- [10] A. Waters, V. Cevher, Distributed bearing estimation via matrix completion, in: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2010, pp. 2590–2593.
- [11] A. Weiss, B. Friedlander, Eigenstructure methods for direction finding with sensor gain and phase uncertainties, *Circ. Syst. Signal Process.* 9 (3) (1990) 271–300.
- [12] A. Weiss, B. Friedlander, Array shape calibration using sources in unknown locations—a maximum likelihood approach, *IEEE Trans. Acoust. Speech Signal Process.* 37 (12) (1989) 1958–1966.

- [13] Y. Rockah, P. Schultheiss, Array shape calibration using sources in unknown locations—Part I: far-field sources, *IEEE Trans. Acoust. Speech Signal Process.* 35 (3) (1987) 286–299.
- [14] M. Viberg, A. Swindlehurst, A Bayesian approach to auto-calibration for parametric array signal processing, *IEEE Trans. Signal Process.* 42 (12) (1994) 3495–3507.
- [15] B. Friedlander, The root-MUSIC algorithm for direction finding with interpolated arrays, *Signal Process.* 30 (1993) 15–19.
- [16] P. Hyberg, M. Jansson, B. Ottersten, Array interpolation and DOA MSE reduction, *IEEE Trans. Signal Process.* 53 (12) (2005) 4464–4471.
- [17] A. Gershman, J. Böhme, A note on most favorable array geometries for DOA estimation and array interpolation, *IEEE Signal Process. Lett.* 4 (8) (1997) 232–235.
- [18] F. Belloni, A. Richter, V. Koivunen, DoA estimation via manifold separation for arbitrary array structures, *IEEE Trans. Signal Process.* 55 (10) (2007) 4800–4810.
- [19] M. Costa, A. Richter, V. Koivunen, Unified array manifold decomposition based on spherical harmonics and 2-D Fourier basis, *IEEE Trans. Signal Process.* 58 (9) (2010) 4634–4645.
- [20] M. Doron, E. Doron, Wavefield modeling and array processing, Part I—Spatial sampling, *IEEE Trans. Signal Process.* 42 (10) (1994) 2549–2559.
- [21] M. Doron, E. Doron, Wavefield modeling and array processing, Part II—Algorithms, *IEEE Trans. Signal Process.* 42 (10) (1994) 2560–2570.
- [22] M. Doron, E. Doron, Wavefield modeling and array processing, Part III—Resolution capacity, *IEEE Trans. Signal Process.* 42 (10) (1994) 2571–2580.
- [23] J. Li, P. Stoica, Z. Wang, On robust Capon beamforming and diagonal loading, *IEEE Trans. Signal Process.* 51 (7) (2003) 1702–1715.
- [24] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall, 1993.
- [25] P. Stoica, E. Larsson, A. Gershman, The stochastic CRB for array processing: a textbook derivation, *IEEE Signal Process. Lett.* 8 (5) (2001) 148–150.
- [26] P. Stoica, B. Ottersten, M. Viberg, R. Moses, Maximum likelihood array processing for stochastic coherent sources, *IEEE Trans. Signal Process.* 44 (1) (1996) 96–105.
- [27] V. Koivunen, E. Ollila, Direction of arrival estimation under uncertainty, in: S. Chandran (Ed.), *Advances in Direction of Arrival Estimation*, Artech House, 2006, pp. 241–258 (Chapter 12).
- [28] R. Mailloux, *Phased Array Antenna Handbook*, second ed., Artech House, 2005.
- [29] K. Dandekar, H. Ling, G. Xu, Effect of mutual coupling on direction finding in smart antenna applications, *Electron. Lett.* 36 (22) (2000) 1889–1891.
- [30] R. Goossens, H. Rogier, A hybrid UCA-RARE/root-MUSIC approach for 2-D direction of arrival estimation in uniform circular arrays in the presence of mutual coupling, *IEEE Trans. Antennas Propag.* 55 (3) (2007) 841–849.
- [31] H. Steyskal, J. Herd, Mutual coupling compensation in small array antennas, *IEEE Trans. Antennas Propag.* 38 (12) (1990) 1971–1975.
- [32] J. Hansen (Ed.), *Spherical Near-Field Antenna Measurements*, Peter Peregrinus Ltd., 1988.
- [33] M. Landmann, M. Käske, R. Thomä, Impact of incomplete and inaccurate data models on high resolution parameter estimation in multidimensional channel sounding, *IEEE Trans. Antennas Propag.* 60 (2) (2012) 557–573.
- [34] M. Zatman, How narrowband is narrowband? *IEE Proc. Radar Sonar Navig.* 145 (2) (1998) 85–91.
- [35] H. Wang, M. Kaveh, Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wide-band sources, *IEEE Trans. Acoust. Speech Signal Process.* 33 (4) (1985) 823–831.
- [36] R. Klemm, *Principles of Space-Time Adaptive Processing*, The Institution of Electrical Engineers, 2002.
- [37] S. Werner, M. With, V. Koivunen, Householder multistage Wiener filter for space-time navigation receivers, *IEEE Trans. Aerosp. Electron. Syst.* 43 (3) (2007) 975–988.

- [38] J. Sorelius, R. Moses, T. Söderström, A. Swindlehurst, Effects of nonzero bandwidth on direction of arrival estimators in array signal processing, *IEEE Proc. Radar Sonar Navig.* 145 (6) (1998) 317–324.
- [39] A. Abidi, Direct-conversion radio transceivers for digital communications, *IEEE J. Solid State Circ.* 30 (12) (1995) 1399–1410.
- [40] U. Nickel, On the influence of channel errors on array signal processing methods, *Int. J. Electron. Commun.* 47 (4) (1993) 209–219.
- [41] F. Demmel, Practical aspects of design and application of direction-finding systems, in: T. Tuncer, B. Friedlander (Eds.), *Classical and Modern Direction-of-Arrival Estimation*, Academic Press, Burlington, MA, USA, 2009, pp. 53–92.
- [42] G. Krishnamurthy, K. Gard, Time division multiplexing front-ends for multiantenna integrated wireless receivers, *IEEE Trans. Circ. Syst. I: Regular Papers* 57 (6) (2010) 1231–1243.
- [43] S. Loyka, The influence of electromagnetic environment on operation of active array antennas: analysis and simulation techniques, *IEEE Antennas Propag. Mag.* 41 (6) (1999) 23–39.
- [44] M. Valkama, A. Springer, G. Hueber, Digital signal processing for reducing the effects of RF imperfections in radio devices—an overview, in: *Proceedings of the IEEE International Symposium on Circuits and Systems*, 2010, pp. 813–816.
- [45] J. Toivanen, T. Laitinen, P. Vainikainen, Modified test zone field compensation for small-antenna measurements, *IEEE Trans. Antennas Propag.* 58 (11) (2010) 3471–3479.
- [46] G. Golub, C. Loan, *Matrix Computations*, third ed., John Hopkins University Press, 1996.
- [47] J. Fessler, A. Hero, Space-alternating generalized expectation-maximization algorithm, *IEEE Trans. Signal Process.* 42 (10) (1994) 2664–2677.
- [48] P. Stoica, K. Sharman, Maximum likelihood methods for direction-of-arrival estimation, *IEEE Trans. Acoust. Speech Signal Process.* 38 (7) (1990) 1132–1143.
- [49] M. Viberg, B. Ottersten, Sensor array processing based on subspace fitting, *IEEE Trans. Signal Process.* 39 (5) (1991) 1110–1121.
- [50] M. Rübsamen, A. Gershman, Direction-of-arrival estimation for nonuniform sensor arrays: from manifold separation to Fourier domain MUSIC methods, *IEEE Trans. Signal Process.* 57 (2) (2009) 588–599.
- [51] C. Mathews, M. Zoltowski, Eigenstructure techniques for 2-D angle estimation with uniform circular arrays, *IEEE Trans. Signal Process.* 42 (9) (1994) 2395–2407.
- [52] M. Pesavento, A. Gershman, Z. Luo, Robust array interpolation using second-order cone programming, *IEEE Signal Process. Lett.* 9 (1) (2002) 8–11.
- [53] M. Costa, A. Richter, V. Koivunen, DoA and polarization estimation for arbitrary array configurations, *IEEE Trans. Signal Process.* 60 (5) (2012) 2330–2343.
- [54] J. Rissanen, MDL denoising, *IEEE Trans. Inform. Theory* 46 (7) (2000) 2537–2543.
- [55] J. Li, P. Stoica (Eds.), *Robust Adaptive Beamforming*, John Wiley and Sons, 2006.
- [56] A. Zoubir, V. Koivunen, Y. Chakhchoukh, M. Muma, Robust estimation in signal processing, *IEEE Signal Process. Mag.* 29 (2012).
- [57] E. Ollila, V. Koivunen, Robust estimation techniques for complex-valued random vectors, in: S. Haykin, T. Adali (Eds.), *Adaptive Signal Processing: Next Generation Solutions*, Wiley, 2009, pp. 87–142.
- [58] A. Swindlehurst, T. Kailath, A performance analysis of subspace-based methods in the presence of model errors—Part I: the MUSIC algorithm, *IEEE Trans. Signal Process.* 40 (7) (1992) 1758–1774.
- [59] S. Vorobyov, A. Gershman, Z. Luo, Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem, *IEEE Trans. Signal Process.* 51 (2) (2003) 313–324.
- [60] S. Shahbazpanahi, A. Gershman, Z. Luo, K. Wong, Robust adaptive beamforming for general-rank signal models, *IEEE Trans. Signal Process.* 51 (9) (2003) 2257–2269.
- [61] R. Lorenz, S. Boyd, Robust minimum variance beamforming, *IEEE Trans. Signal Process.* 53 (5) (2005) 1684–1696.

- [62] L. Du, T. Yardibi, J. Li, P. Stoica, Review of user parameter-free robust adaptive beamforming algorithms, *Digit. Signal Process.* 19 (4) (2009) 567–582.
- [63] A. Barabell, Improving the resolution performance of the eigenstructure-based direction-finding algorithms, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 1983, pp. 336–339.
- [64] J. Zhuang, W. Li, A. Manikas, Fast root-MUSIC for arbitrary arrays, *Electron. Lett.* 46 (2) (2010) 174–176.
- [65] A. Azremi, M. Kyro, J. Ilvonen, J. Holopainen, S. Ranvier, C. Icheln, P. Vainikainen, Five-element inverted-F antenna array for MIMO communications and radio direction finding on mobile terminal, in: Loughborough Antennas and Propagation Conference, 2009, pp. 557–560.

# Applications of Array Signal Processing

# 20

A. Lee Swindlehurst<sup>\*</sup>, Brian D. Jeffs<sup>†</sup>, Gonzalo Seco-Granados<sup>‡</sup>, and Jian Li<sup>§</sup>

<sup>\*</sup>*Department of Electrical Engineering and Computer Science, University of California, Irvine, CA, USA*

<sup>†</sup>*Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT, USA*

<sup>‡</sup>*Department of Telecommunications and Systems Engineering, Universitat Autònoma de Barcelona, Bellaterra, Barcelona, Spain*

<sup>§</sup>*Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA*

## 3.20.1 Introduction and background

The principles behind obtaining information from measuring an acoustic or electro-magnetic field at different points in space have been understood for many years. Techniques for long-baseline optical interferometry were known in the mid-19th century, where widely separated telescopes were proposed for high-resolution astronomical imaging. The idea that direction finding can be performed with two acoustic sensors has been around at least as long as the physiology of human hearing has been understood. The mathematical duality observed between sampling a signal either uniformly in time or uniformly in space is ultimately just an elegant expression of Einstein's theory of relativity. However, most of the technical advances in array signal processing have occurred in the last 30 years, with the development and proliferation of inexpensive and high-rate analog-to-digital (A/D) converters together with flexible and very powerful digital signal processors (DSPs). These devices have made the chore of collecting data from multiple sensors relatively easy, and helped give birth to the use of sensor arrays in many different areas.

Parallel to the advances in hardware that facilitated the construction of sensor array platforms were breakthroughs in the mathematical tools and models used to exploit sensor array data. Finite impulse response (FIR) filter design methods originally developed for time-domain applications were soon applied to uniform linear arrays in implementing digital beamformers. Powerful data-adaptive beamformers with constrained look directions were conceived and applied with great success in applications where the rejection of strong interference was required. Least-mean square (LMS) and recursive least-squares (RLS) time-adaptive techniques were developed for time-varying scenarios. So-called “blind” adaptive beamforming algorithms were devised that exploited known temporal properties of the desired signal rather than its direction-of-arrival (DOA).

For applications where a sensor array was to be used for locating a signal source, for example finding the source's DOA, one of the key theoretical developments was the parametric vector-space formulation introduced by Schmidt and others in the 1980s. They popularized a vector space signal model with a parameterized array manifold that helped connect problems in array signal processing to advanced estimation theoretic tools such as Maximum Likelihood (ML), Minimum Mean-Square Estimation (MMSE) the Likelihood Ratio Test (LRT) and the Cramér-Rao Bound (CRB). With these

tools, one could rigorously define the meaning of the term “optimal” and performance could be compared against theoretical bounds. Trade-offs between computation and performance led to the development of efficient algorithms that exploited certain types of array geometries. Later, concerns about the fidelity of array manifold models motivated researchers to study more robust designs and to focus on models that exploited properties of the received signals themselves.

The driving applications for many of the advances in array signal processing mentioned above have come from military problems involving radar and sonar. For obvious reasons, the military has great interest in the ability of multi-sensor surveillance systems to locate and track multiple “sources of interest” with high resolution. Furthermore, the potential to null co-channel interference through beamforming (or perhaps more precisely, “null-steering”) is a critical advantage gained by using multiple antennas for sensing and communication. The interference mitigation capabilities of antenna arrays and information theoretic analyses promising large capacity gains has given rise to a surge of applications for arrays in multi-input, multi-output (MIMO) wireless communications in the last 15 years. Essentially all current and planned cellular networks and wireless standards rely on the use of antenna arrays for extending range, minimizing transmit power, increasing throughput, and reducing interference. From peering to the edge of the universe with arrays of radio telescopes to probing the structure of the brain using electrode arrays for electroencephalography (EEG), many other applications have benefited from advances in array signal processing.

In this chapter, we explore some of the many applications in which array signal processing has proven to be useful. We place emphasis on the word “some” here, since our discussion will not be exhaustive. We will discuss several popular applications across a wide variety of disciplines to indicate the breadth of the field, rather than delve deeply into any one or try to list them all. Our emphasis will be on developing a data model for each application that falls within the common mathematical framework typically assumed in array processing problems. We will spend little time on algorithms, presuming that such material is covered elsewhere in this collection; algorithm issues will only be addressed when the model structure for a given application has unique implications on algorithm choice and implementation. Since radar and wireless communications problems are discussed in extensive detail elsewhere in the book, our discussion of these topics will be relatively brief.

---

### 3.20.2 Radar applications

We begin with the application area for which array signal processing has had the most long-lasting impact, dating back to at least World War II. Early radar surveillance systems, and even many still in use today, obtain high angular resolution by employing a radar dish that is mechanically steered in order to scan a region of interest. While such slow scanning speeds are suitable for weather or navigation purposes, they are less tolerable in military applications where split-second decisions must be made regarding targets (e.g., missiles) that may be moving at several thousand miles per hour. The advent of electronically scanned phased arrays addressed this problem, and ushered in the era of modern array signal processing.

Phased arrays are composed of from a few up to several thousand individual antennas laid out in a line, circle, rectangle or even randomly. Directionality is achieved by the process of *beamforming*: multiplying the output of each antenna by a complex weight with a properly designed phase (hence the term “phased” array), and then summing these weighted outputs together. The conventional “delay-and-sum”



**FIGURE 20.1**

A phased array radar enclosed in the nose of a fighter jet.

beamforming scheme involves choosing the weights to phase delay the individual antenna outputs such that signals from a chosen direction add constructively and those from other directions do not. Since the weights are applied electronically, they can be rapidly changed in order to focus the array in many different directions in a very short period of time. Modern phased arrays can scan an entire hemisphere of directions thousands of times per second. Figures 20.1 and 20.2 show examples of airborne and ground-based phased array radars.

For scanning phased arrays, a fixed set of beamforming weights is repeatedly applied to the antennas over and over again, in order to provide coverage of some area of interest. Techniques borrowed from time-domain filter design such as windowing or frequency sampling can be used to determine the beamformer weights, and the primary trade-off is beamwidth/resolution versus sidelobe levels. Adaptive weight design is required if interference or clutter must be mitigated. In principle, the phased array beamformer can be implemented with either analog or digital hardware, or a combination of both. For arrays with a very large number of antennas (e.g., the Patriot radar has in excess of 5000 elements), analog techniques are often employed due to the hardware and energy expense required in implementing a separate RF receive chain for each antenna. Hybrid implementations are also used in which analog beamforming over subsets of the array is used to create a smaller number of signal streams, which are then processed by a digital beamformer. This is a common approach, for example, in shipborne radar systems, where the targets of interest (e.g., low altitude cruise missiles) are typically located near the horizon. In such systems, analog beamforming with vertically-oriented strips of antennas are used to create a set of narrow azimuthal beams whose outputs can be flexibly combined using digital signal processing.

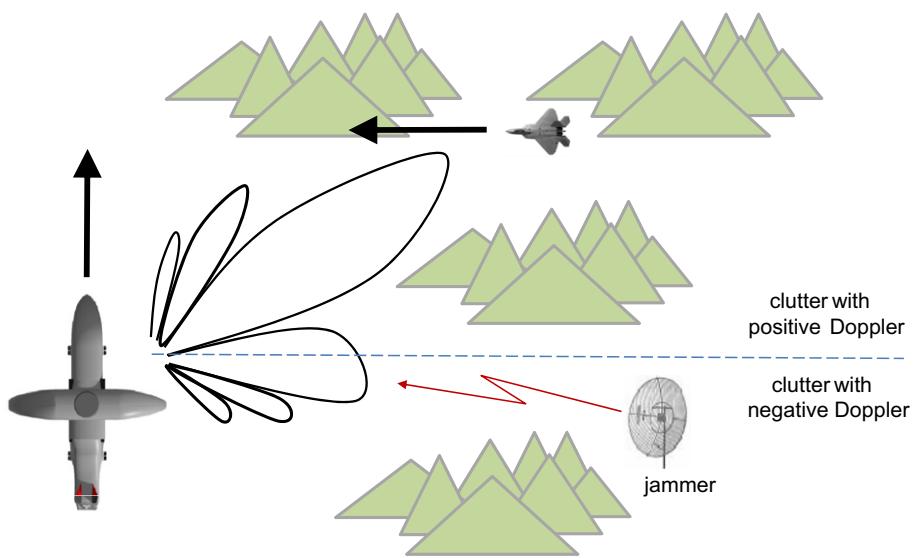
**FIGURE 20.2**

The phased array used for targeting the Patriot surface-to-air missile system, composed of over 5000 individual elements.

In this section, we will briefly discuss the two radar array applications that have received the most attention in the signal processing literature: space-time adaptive processing (STAP) and MIMO radar. Since these are discussed in detail elsewhere in the book, our discussion will not be comprehensive. While STAP and MIMO radar applications are typically used in active radar systems, arrays are also useful for passive radars, such as those employed in radio astronomy. We will devote a separate section to array signal processing for radio astronomy and discuss this application in much more detail, since it is not addressed elsewhere in the book.

### 3.20.2.1 Space-time adaptive processing

In many tactical military applications, airborne surveillance radars are tasked with providing location and tracking information about moving objects both on the ground and in the air. These radars typically

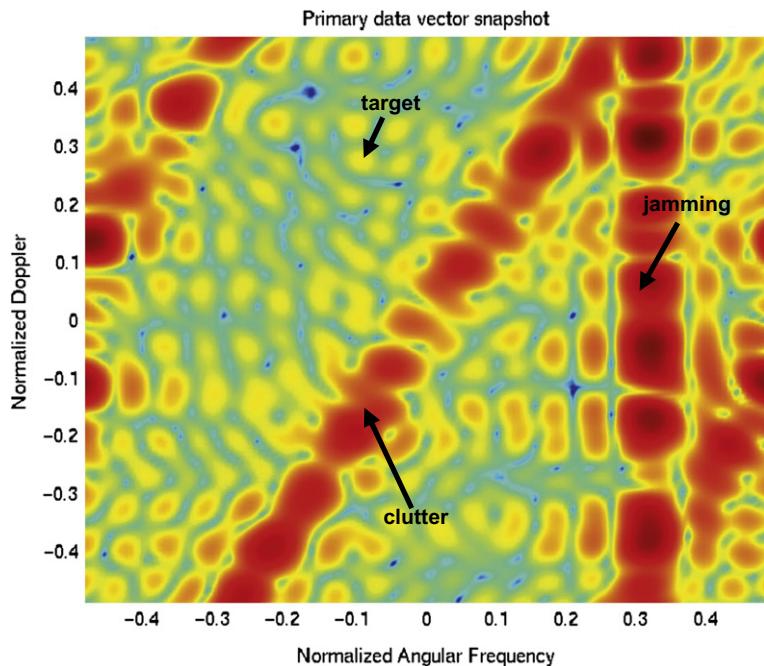
**FIGURE 20.3**

Airborne STAP scenario with clutter and jamming.

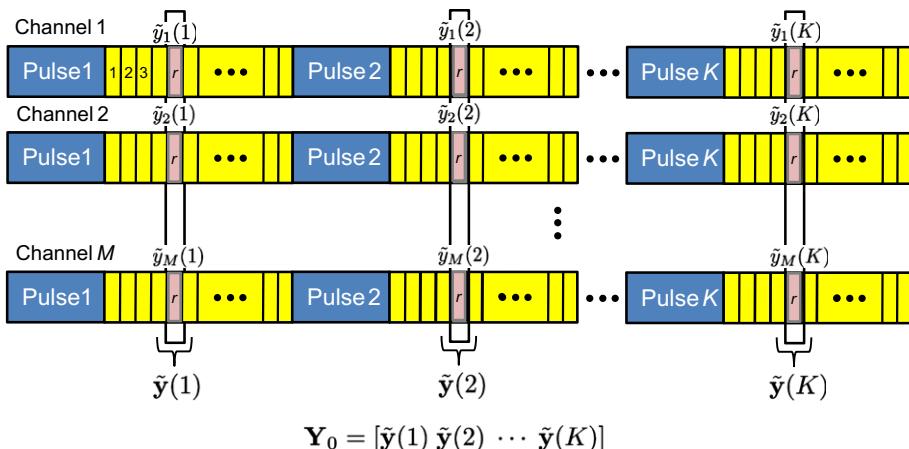
use pulse-Doppler techniques since measuring the velocity of the objects of interest (or “targets”) is a key to accurately tracking them. As depicted in Figure 20.3, even when the targets are airborne, the transmit mainbeam and sidelobes will still illuminate the ground, especially when the radar look-direction is at a negative elevation angle (the targets may be below the radar platform). This means that the radar returns will contain significant energy from ground reflections, referred to as *clutter*. In addition, since pulse-Doppler techniques require an active radar, the frequency support of the radar signal is known, and an adversary can employ strong jamming to further mask the target returns. Often, the target signal is many tens of dB (e.g., 50 or more) weaker than the combination of jamming and clutter.

The difficulty of the situation is revealed by Figure 20.4, which shows the angle-Doppler power spectrum of data that contains a target together with clutter and jamming at a particular range. The jamming signal is due to a point source, so it is confined to a single arrival angle, but the jamming signal extends across the entire bandwidth of the data. The clutter energy lies on a ridge that cuts across the angle-Doppler space in a direction that is a function of the heading, altitude and velocity of the radar, and the current range bin of interest. Clutter in front of the radar will have a positive Doppler, and that behind it will be negative (as seen in Figure 20.3). Compared with the clutter and jamming, the target signal is weak and cannot be distinguished from the background due to the limited dynamic range of the receiver. Doppler filtering alone is not sufficient to reveal the target, since the jamming signal cuts across the entire bandwidth of the signal. On the other hand, using spatial filtering (beamforming) to null the jammer will still leave most of the clutter untouched. What is needed is a two-dimensional space-time filter. The process of designing and applying such a filter is referred to as space-time adaptive processing (STAP).

To better place STAP in the context of array signal processing problems, consider Figure 20.5 which depicts how data is organized in an  $M$ -antenna pulse-Doppler radar. The radar transmits a series of  $K$

**FIGURE 20.4**

Angle-Doppler spectrum with weak target in the presence of clutter and jamming.

**FIGURE 20.5**

Organization of data for range bin  $r$  in STAP pulsed-Doppler radar.

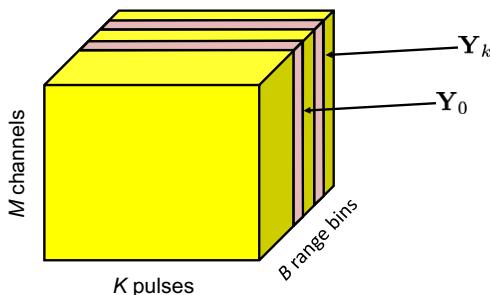
pulses separated in time by a fixed pulse repetition interval (PRI). In order to focus sufficient energy to obtain a measurable return from a target, the transmitted pulse is typically a very spatially focused signal steered towards a particular azimuth and elevation angle or look direction. However, the mathematical description of the STAP process can be described independently of this assumption. In between the pulses, the radar collects the returns from each of the  $M$  antennas, which are sampled after the received data is passed through a pulse-compression matched filter. Each sample corresponds to the aggregate contribution of scatterers (clutter and targets, if such exist) at a particular range together with any noise, jamming or other interference that may be present. The range for a given sample is given by the speed of light multiplied by half the time interval between transmission of the pulse and the sampling instant. Suppose we are interested in a particular range bin  $r$ . As shown in the figure, we will let

$$\tilde{\mathbf{y}}(t) = \begin{bmatrix} \tilde{y}_1(t) \\ \vdots \\ \tilde{y}_M(t) \end{bmatrix}, \quad (20.1)$$

$$\mathbf{Y}_0 = [\tilde{\mathbf{y}}(1) \ \cdots \ \tilde{\mathbf{y}}(K)] \quad (20.2)$$

represent the  $M \times 1$  vector of returns from the array after pulse  $t$  and the  $M \times K$  matrix of returns from all  $K$  pulses for range bin  $r$ , respectively.

Alternatively, as shown in Figure 20.6, the data can be viewed as forming a cube over  $M$  antennas,  $K$  pulses, and  $B$  total range bins. Each range bin corresponds to a different slice of the data cube. Data from adjacent range bins  $\mathbf{Y}_k$  will be used to counter the effect of clutter and jamming in the range bin of interest, which we index with  $k = 0$ . The time required to collect the data cube for a given look direction is referred to as a coherent processing interval (CPI). If the radar employs multiple look directions, a separate CPI is required for each. Assuming the target, clutter and jamming are stationary over different CPIs, data from these CPIs can be combined to perform target detection and localization. However, in our discussion here we will assume that data from only a single CPI is available to determine the presence or absence of a target in range bin  $r$ .



**FIGURE 20.6**

STAP data cube showing slices for range bin of interest ( $\mathbf{Y}_0$ ) and secondary range bin ( $\mathbf{Y}_k$ ).

If a target is present in the data set  $\mathbf{Y}_0$ , then the received signal can be modeled as

$$\tilde{\mathbf{y}}(t) = b_0 \mathbf{a}(\theta_0, \phi_0) e^{j\omega_0 t} + \underbrace{\sum_{i=1}^{D_c} b_i \mathbf{a}(\theta_i, \phi_i) e^{j\omega_i t} + \sum_{j=1}^{D_j} \mathbf{a}(\theta_j^*, \phi_j^*) x_j(t)}_{\tilde{\mathbf{e}}(t)} + \mathbf{n}(t), \quad (20.3)$$

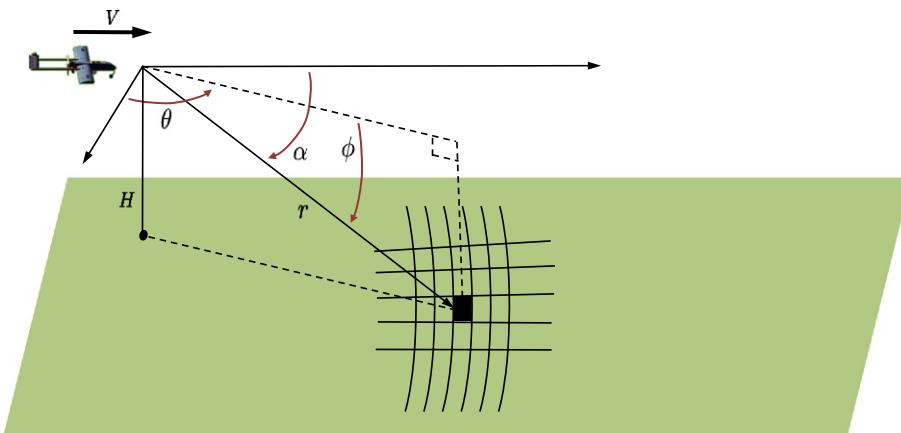
where  $b_i$  is the amplitude of the return from the  $i$ th scatterer ( $i = 0$  corresponds to the target),  $(\theta_i, \phi_i)$  are the azimuth and elevation angles of the  $i$ th scatterer,  $\omega_i$  is the corresponding Doppler frequency,  $\mathbf{a}(\theta, \phi)$  is the response of the  $M$ -element receive array to a signal from direction  $(\theta_i, \phi_i)$ ,  $x_j(t)$  is the signal transmitted by the  $j$ th jammer,  $(\theta_j^*, \phi_j^*)$  denote the DOA of the  $j$ th jammer signal,  $D_c$  represents the number of distinct clutter sources,  $D_j$  the number of jammers, and  $\mathbf{n}(t)$  corresponds to any remaining background noise and interference. We have also defined  $\tilde{\mathbf{e}}(t)$  to contain all received signals except that of the target. Note that the above model assumes the relative velocity of the radar and all scatterers is constant over the CPI, so that the Doppler effect can be described as a complex sinusoid.

Technically, the amplitude and Doppler terms  $b_i$  and  $\omega_i$  will also depend on the azimuth and elevation angles of the  $i$ th scatterer since the Doppler frequency is position-dependent and the strength of the return is a function of the transmit beampattern in addition to the intrinsic radar cross section (RCS) of the scatterer. This is clear from Figure 20.7, which shows the geometry of the airborne radar with respect a clutter patch on the ground at some range  $r$ . The Doppler frequency for the given clutter patch at azimuth  $\theta$  and elevation  $\phi$  can be determined from the following equations:

$$\sin \phi = \frac{H}{r} + \frac{r^2 - H^2}{2r(r_e + H)}, \quad (20.4)$$

$$\cos \alpha = \sin \theta \cos \phi, \quad (20.5)$$

$$\omega = \frac{4\pi V}{\lambda} \cos \alpha, \quad (20.6)$$



**FIGURE 20.7**

Geometry for determining the Doppler frequency due to a ground clutter patch at range  $r$ .

where  $r_e$  denotes the earth's radius,  $H$  is the altitude of the radar, and  $\alpha$  is the angle between the velocity vector of the radar and the clutter patch. To simplify the notation, we have dropped the explicit dependence of  $b_i$  and  $\omega_i$  on  $\theta_i, \phi_i$ . While the highest Doppler frequencies obviously occur for small  $\alpha$  (forward- or rear-looking radar), the fact that  $\cos \alpha$  changes relatively slowly for small  $\alpha$  compared with  $\alpha$  near  $90^\circ$  means that the Doppler spread of the clutter for a forward- or rear-looking radar will be smaller than that for the side-looking case.

Rather than working with the data matrix  $\mathbf{Y}_0$ , for STAP it is convenient to vectorize the data as follows:

$$\mathbf{y}_0 = \text{vec}(\mathbf{Y}_0) = \begin{bmatrix} \tilde{\mathbf{y}}(1) \\ \vdots \\ \tilde{\mathbf{y}}(K) \end{bmatrix} = b_0 \mathbf{s}(\theta_0, \phi_0, \omega_0) + \mathbf{e}_0, \quad (20.7)$$

where  $\mathbf{e}_0$  is defined similarly to  $\mathbf{y}_0$  for the clutter and jamming, and where

$$\mathbf{s}(\theta_0, \phi_0, \omega_0) = \text{vec} \left( \mathbf{a}(\theta_0, \phi_0) \begin{bmatrix} e^{j\omega_0} & e^{j2\omega_0} & \dots & e^{jK\omega_0} \end{bmatrix} \right) \quad (20.8)$$

$$= \begin{bmatrix} e^{j\omega_0} \\ \vdots \\ e^{jK\omega_0} \end{bmatrix} \otimes \mathbf{a}(\theta_0, \phi_0). \quad (20.9)$$

The  $MK \times 1$  vector  $\mathbf{y}_0$  is the space-time snapshot associated with the given range bin ( $r$ ) of interest. To detect whether or not a target signal was present in  $\mathbf{y}_0$ , one may be tempted to use a minimum-variance distortionless response (MVDR) space-time filter of the form

$$\mathbf{w}(\theta, \phi, \omega) = \frac{\mathbf{R}_{y_0}^{-1} \mathbf{s}(\theta, \phi, \omega)}{\mathbf{s}^H(\theta, \phi, \omega) \mathbf{R}_{y_0}^{-1} \mathbf{s}(\theta, \phi, \omega)}, \quad (20.10)$$

apply it to  $\mathbf{y}_0$  for various choices of  $(\theta, \phi, \omega)$ , which then should lead to a peak in the filter output when  $(\theta, \phi, \omega)$  corresponds to the parameters of the target. The problem with this approach is that we will not have enough data available to estimate the covariance  $\mathbf{R}_{y_0}$ ; if the target signal is only present in this range bin, then with a single CPI we only have a single snapshot that possesses this covariance.

Fortunately, an alternative approach exists, since it can be shown via the matrix inversion lemma (MIL) that the optimal MVDR space-time filter is proportional to another vector that can be more readily estimated:

$$\mathbf{w}(\theta, \phi, \omega) \propto \mathbf{R}_{e_0}^{-1} \mathbf{s}(\theta, \phi, \omega), \quad (20.11)$$

which depends on the covariance  $\mathbf{R}_{e_0}$  of the clutter and jamming. In particular, STAP relies on the assumption that the statistics of the clutter and jamming in range bins near the one in question are similar, and can be used to estimate  $\mathbf{R}_{e_0}$ . For example, let  $S_0 = \{k_1, k_2, \dots, k_{N_s}\}$  represent a set containing the indices of  $N_s$  target-free range bins near  $r$  (since the target signal may leak into range

bins immediately adjacent to bin  $r$ , these are typically excluded), then a sample estimate of  $\mathbf{R}_{e_0}$  may be formed as

$$\widehat{\mathbf{R}}_{e_0} = \sum_{k \in S_0} \mathbf{y}_k \mathbf{y}_k^H = \boldsymbol{\Gamma} \boldsymbol{\Gamma}^H, \quad \boldsymbol{\Gamma} = [\mathbf{y}_{k_1} \dots \mathbf{y}_{k_{N_s}}], \quad (20.12)$$

where  $\mathbf{y}_k$  is the space-time snapshot from range bin  $k$ . The  $N_s$  samples that compose  $\boldsymbol{\Gamma}$  are referred to as secondary data vectors.

Implementation of the space-time filter in (20.11) using a covariance estimate such as (20.12) is referred to as the “fully adaptive” STAP algorithm. The number  $N_s$  of secondary data vectors chosen to estimate  $\mathbf{R}_{e_0}$  is a critical parameter. If it is too small, a poor estimate will be obtained; if it is too large, then the assumption of statistical similarity may be strained. Another critical parameter is the rank of  $\mathbf{R}_{e_0}$ . While in theory  $\mathbf{R}_{e_0}$  may be full rank, in practice its effective rank  $\rho$  is typically much smaller than its dimension  $MK$ , since the clutter and jamming are usually orders of magnitude stronger than the background noise. According to Brennan’s rule [1], the value of  $\rho$  for a uniform linear array is  $M + (K - 1)\beta$ , where  $\beta$  is a factor that depends on the speed of the array platform and the pulse repetition frequency (PRF), and is usually between 0.5 and 1.5. The rank of  $\mathbf{R}_{e_0}$  for non-linear array geometries will be greater, although no concise formula exists in the general case. Factors influencing the rank of  $\mathbf{R}_{e_0}$  include the beamwidth and sidelobes of the transmit pulse (narrower pulses and lower sidelobes mean smaller  $\rho$ ), the presence of intrinsic clutter motion (e.g., leaves on trees in a forest) or clutter discrete (strong specular reflectors), and whether the radar is forward- or side-looking (the Doppler spread of the clutter and hence  $\rho$  is much smaller in the forward-looking case).

The rank of  $\mathbf{R}_{e_0}$  is important in determining the minimum value for  $N_s$  required to form a sufficiently accurate sample estimate. A general rule of thumb is that the number of required samples is on the order of  $2\rho-5\rho$ . Even when these many stationary secondary range bins are available,  $N_s$  may still be much smaller than  $MK$ , and  $\widehat{\mathbf{R}}_{e_0}$  will not be invertible. In such situations, a common remedy is to employ a diagonal loading factor  $\delta$ , and use the MIL to simplify calculation of the inverse:

$$(\widehat{\mathbf{R}}_{e_0} + \delta \mathbf{I})^{-1} = (\boldsymbol{\Gamma} \boldsymbol{\Gamma}^H + \delta \mathbf{I})^{-1} \quad (20.13)$$

$$= \frac{1}{\delta} \left( \mathbf{I} - \boldsymbol{\Gamma} (\boldsymbol{\Gamma}^H \boldsymbol{\Gamma} + \delta \mathbf{I})^{-1} \boldsymbol{\Gamma}^H \right). \quad (20.14)$$

Another approach is to use a pseudo-inverse based on principal components.

Still, the computation involved in implementing the fully adaptive STAP algorithm is often prohibitive. The dimension  $MK$  of  $\mathbf{R}_{e_0}$  is often in the hundreds, and computational costs add up quickly when one realizes the STAP filtering must be performed in multiple range bins for each look direction. Most of the STAP research in recent years has been aimed at reducing the computational load to more reasonable levels. Two main classes of approaches have been proposed: (1) partially adaptive STAP and (2) parametric modeling. In the partially adaptive approach, the dimensions of the space-time data slice are reduced by means of linear transformations in space or time or both:

$$\mathbf{Y}_0 \rightarrow \mathbf{T}_a \mathbf{Y}_0 \mathbf{T}_\omega^H. \quad (20.15)$$

Techniques for choosing the transformation matrices include beamspace methods, Doppler binning, PRI staggering, etc. The classical moving target indicator (MTI) approach can be thought of as falling

in this class of algorithms for the special case where  $\mathbf{T}_a$  is one-dimensional. The dimension reduction achieved by partially adaptive methods not only reduces the computational load, but it improves the numerical conditioning and decreases the required secondary sample support as well.

The parametric approach is based on the observation that in (20.14), as  $\delta \rightarrow 0$ , we have

$$\lim_{\delta \rightarrow 0} (\widehat{\mathbf{R}}_{e_0} + \delta \mathbf{I})^{-1} \propto \left( \mathbf{I} - \boldsymbol{\Gamma} \left( \boldsymbol{\Gamma}^H \boldsymbol{\Gamma} \right)^{-1} \boldsymbol{\Gamma}^H \right). \quad (20.16)$$

Thus, the effect of  $\mathbf{R}_{e_0}^{-1}$  is to approximately project the space-time signal vector onto the space orthogonal to the clutter and jamming. While  $\boldsymbol{\Gamma}$  could be used to define this subspace, a more efficient approach has been proposed based on vector autoregressive (VAR) filtering. To see this, note from (20.3) and (20.7) that the clutter and jamming vector  $\mathbf{e}_k$  for range bin  $k$  over the full CPI can be partitioned into samples for each individual pulse within the CPI:

$$\mathbf{e}_k = \begin{bmatrix} \mathbf{n}_k(1) \\ \mathbf{n}_k(2) \\ \vdots \\ \mathbf{n}_k(K) \end{bmatrix}. \quad (20.17)$$

The VAR approach assumes that the clutter and jamming obey the following model for each pulse  $t$ :

$$\mathbf{H}_0 \mathbf{n}_k(t) + \mathbf{H}_1 \mathbf{n}_k(t-1) + \cdots + \mathbf{H}_L \mathbf{n}_k(t-L+1) = 0, \quad (20.18)$$

where  $L$  is typically assumed to be small (e.g., less than 5–7) and each matrix  $\mathbf{H}_i$  is  $M' \times M$  for some chosen value of  $M'$ . The matrix coefficients of the VAR can be estimated for example by solving a standard least-squares problem of the form

$$\min_{\mathcal{H}} \sum_{k=1}^{N_s} \|\mathcal{H} \mathbf{e}_k\|^2 \quad \text{s.t.} \quad \mathcal{H}^H \mathcal{H} = \mathbf{I}, \quad (20.19)$$

where

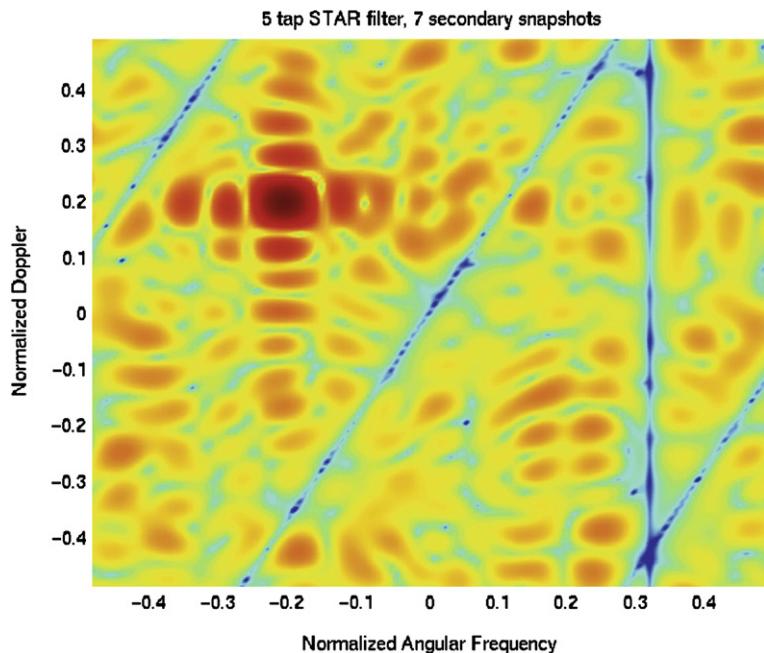
$$\mathcal{H} = \begin{bmatrix} \mathbf{H}_L & \mathbf{H}_{L-1} & \cdots & \mathbf{H}_0 & & \\ & \mathbf{H}_L & \mathbf{H}_{L-1} & \cdots & \mathbf{H}_0 & \\ & & \ddots & & & \ddots \\ & & & \mathbf{H}_L & \mathbf{H}_{L-1} & \cdots & \mathbf{H}_0 \end{bmatrix} \quad (20.20)$$

and the constraint  $\mathcal{H}^H \mathcal{H} = \mathbf{I}$  is used to prevent a trivial solution. The matrix  $\mathcal{H}^H$  will approximately span the subspace orthogonal to  $\boldsymbol{\Gamma}$ , and based on (20.16) a suitable space-time filter would be given by

$$\mathbf{w} = \mathbf{P}_{\mathcal{H}^H} \mathbf{s}(\theta, \phi, \omega), \quad (20.21)$$

where

$$\mathbf{P}_{\mathcal{H}^H} = \mathcal{H}^H \left( \mathcal{H} \mathcal{H}^H \right)^{-1} \mathcal{H}. \quad (20.22)$$

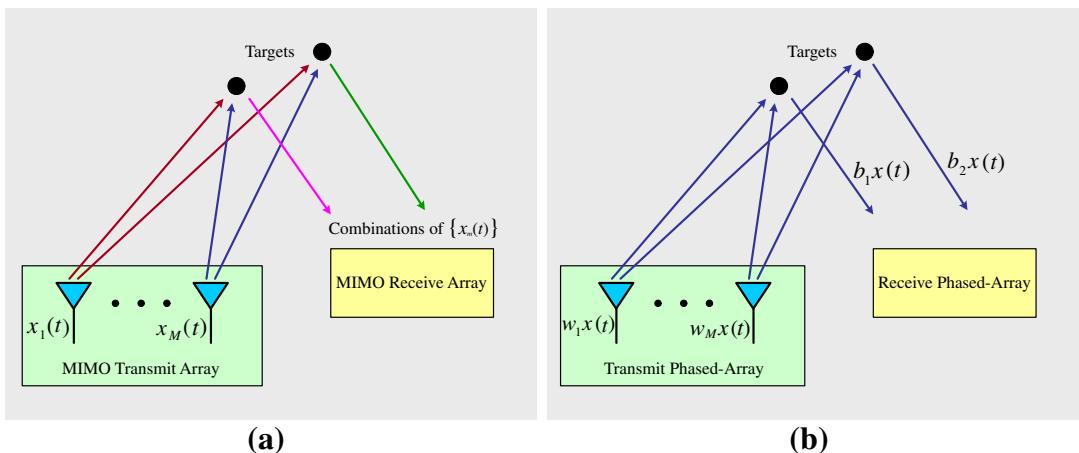
**FIGURE 20.8**

Angle-Doppler spectra after STAP filtering.

This approach is referred to as the space-time autoregressive (STAR) filter. An example of the performance of the STAR filter is given in Figure 20.8 for a case with  $L = 4$  and  $N_s = 7$ . These results are for the same data set that generated the unfiltered angle-Doppler spectrum in Figure 20.4. Note that the clutter and jamming have been removed, and the target is plainly visible. Similar results were obtained in this case with the fully adaptive STAP method with diagonal loading, but required a value of  $N_s$  near 60.

### 3.20.2.2 MIMO radar

Multi-input multi-output (MIMO) radar is beginning to attract a significant amount of attention from researchers and practitioners alike due to its potential of advancing the state-of-the-art of modern radar. Unlike a standard phased-array radar, which transmits scaled versions of a single waveform, a MIMO radar system can transmit via its antennas multiple probing signals that may be chosen quite freely (see Figure 20.9). This waveform diversity enables superior capabilities compared with a standard phased-array radar. For example, the angular diversity offered by widely separated transmit/receive antenna elements can be exploited for enhanced target detection performance. For collocated transmit and receive antennas, the MIMO radar paradigm has been shown to offer many advantages including long virtual array aperture sizes and the ability to untangle multiple paths. Array signal processing plays critical roles in reaping the benefits afforded by the MIMO radar systems. In our discussion here, we focus on array signal processing for MIMO radar with collocated transmit and receive antennas.

**FIGURE 20.9**

(a) MIMO radar and (b) phased-array radar.

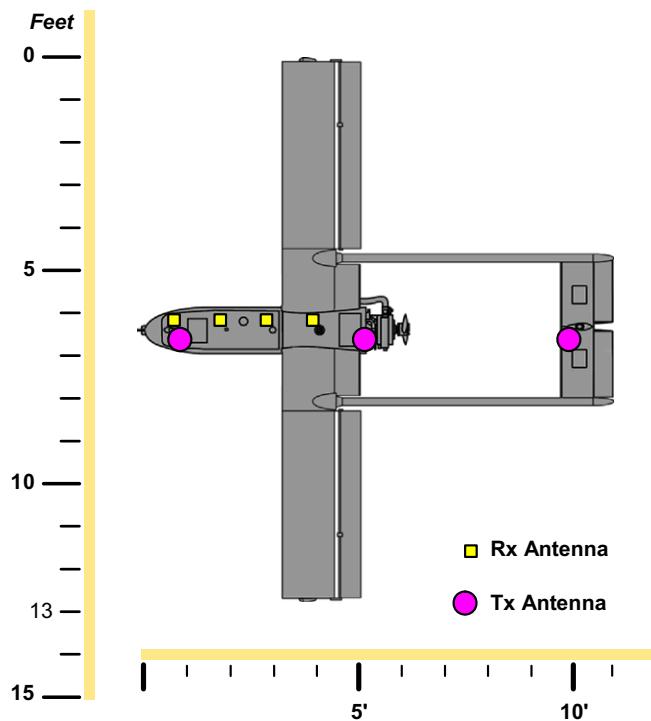
An example of a UAV equipped with a MIMO radar system is shown in Figure 20.10, where the transmit array is sparse and the receive array is a filled (half-wavelength inter-element spacing) uniform linear array. When the transmit antennas transmit orthogonal waveforms, *the virtual array of the radar system is a filled array with an aperture up to  $M$  times that of the receive array, where  $M$  is the number of transmit antennas*. Many advantages of MIMO radar with collocated antennas result directly from this significantly increased virtual aperture size. For example, for small aerial vehicles (with medium or short range applications), a conventional phased-array system could be problematic since it usually weighs too much, consumes too much power, takes up too much space, and is too expensive. In contrast, MIMO radar offers the advantages of reduced complexity, power consumption, weight and cost by obviating phase shifts and affording significantly increased virtual aperture size.

Some typical examples of array processing in MIMO radar include transmit beampattern synthesis, transmit and receive array design, and adaptive array processing for diverse MIMO radar applications. We briefly describe these array processing examples in MIMO radar.

### 3.20.2.2.1 Flexible transmit beampattern synthesis

The probing waveforms transmitted by a MIMO radar system can be designed to approximate a desired transmit beampattern and also to minimize the cross-correlation of the signals reflected from various targets of interest—an operation that would hardly be possible for a phased-array radar.

The recently proposed techniques for transmit narrowband beampattern design have focused on the optimization of the covariance matrix  $\mathbf{R}$  of the waveforms. Instead of designing  $\mathbf{R}$ , we might think of directly designing the probing signals by optimizing a given performance measure with respect to the matrix  $\mathbf{X}$  of the signal waveforms. However, compared with optimizing the same performance measure with respect to the covariance matrix  $\mathbf{R}$  of the transmitted waveforms, optimizing directly with respect to

**FIGURE 20.10**

A UAV equipped with a MIMO radar.

**X** is a more complicated problem. This is so because **X** has more unknowns than **R** and the dependence of various performance measures on **X** is more intricate than the dependence on **R**.

There are several recent methods that can be used to efficiently compute an optimal covariance matrix **R**, with respect to several performance metrics. One of the metrics consists of choosing **R**, under a uniform elemental power constraint (i.e., under the constraint that the diagonal elements of **R** are equal), to achieve the following goals:

- Maximize the total spatial power at a number of given target locations, or more generally, match a desired transmit beampattern.
- Minimize the cross-correlation between the probing signals at a number of given target locations.

Another beampattern design problem is to choose **R**, under the uniform elemental power constraint, to achieve the following goals:

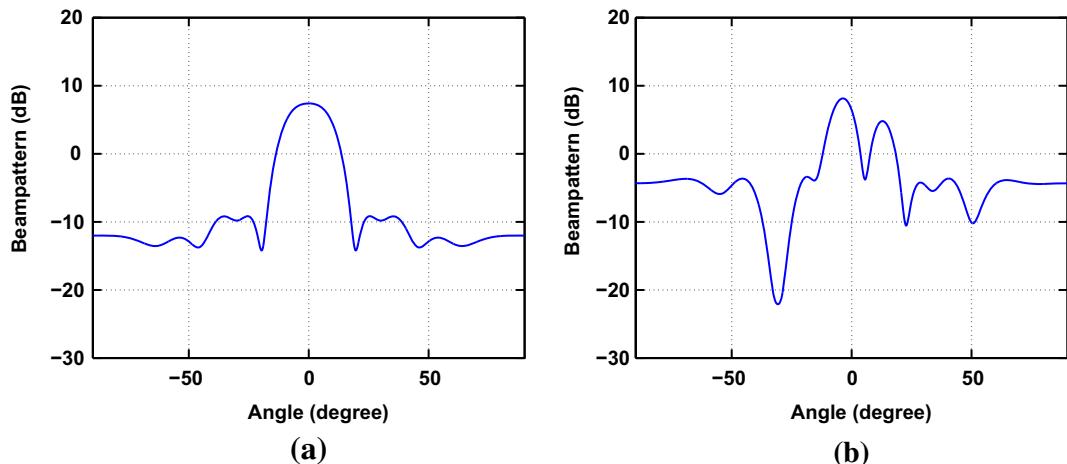
- Minimize the sidelobe level in a prescribed region.
- Achieve a predetermined 3 dB main-beam width.

It can be shown that both design problems can be efficiently solved in polynomial time as a semi-definite quadratic program (SQP).

We comment in passing on the conventional phased-array beampattern design problem in which only the array weight vector can be adjusted and therefore all antennas transmit the same differently-scaled waveform. We can readily modify the MIMO beampattern designs for the case of phased-arrays by adding the constraint that the rank of  $\mathbf{R}$  is one. However, due to the rank-one constraint, both of these originally convex optimization problems become non-convex. The lack of convexity makes the rank-one constrained problems much harder to solve than the original convex optimization problems. Semi-definite relaxation (SDR) is often used to obtain approximate solutions to such rank-constrained optimization problems. The SDR is obtained by omitting the rank constraint. Hence, interestingly, *the MIMO beampattern design problems are the SDRs of the corresponding phased-array beampattern design problems*.

We now provide a numerical example below, where we have used a Newton-like algorithm to solve the rank-one constrained design problems for phased-arrays. This algorithm uses SDR to obtain an initial solution, which is the exact solution to the corresponding MIMO beampattern design problem. Although the convergence of the said Newton-like algorithm is not guaranteed, we did not encounter any apparent problem in our numerical simulations.

Consider the beampattern design problem with  $M = 10$  transmit antennas. The main-beam is centered at  $\theta_0 = 0^\circ$ , with a 3 dB width equal to  $20^\circ$  ( $\theta_1 = -10^\circ, \theta_2 = 10^\circ$ ). The sidelobe region is  $\Omega = [-90^\circ, -20^\circ] \cup [20^\circ, 90^\circ]$ . The minimum-sidelobe beampattern design is shown in Figure 20.11a. Note that the peak sidelobe level achieved by the MIMO design is approximately 18 dB below the mainlobe peak level. Figure 20.11b shows the corresponding phased-array beampattern obtained by using the additional constraint  $\text{rank}(\mathbf{R}) = 1$ . The phased-array design fails to provide a proper mainlobe (it suffers from peak splitting) and its peak sidelobe level is much higher than that of its



**FIGURE 20.11**

Minimum sidelobe beampattern designs, under the uniform elemental power constraint, when the 3 dB main-beam width is  $20^\circ$ . (a) MIMO and (b) phased-array.

MIMO counterpart. We note that, under the uniform elemental power constraint, the number of degrees of freedom (DOF) of the phased-array that can be used for beampattern design is equal to only  $M - 1$ ; consequently, it is difficult for the phased-array to synthesize a proper beampattern. The MIMO design, on the other hand, can be used to achieve a much better beampattern due to its much larger number of DOF, viz.  $M^2 - M$ .

The radar waveforms are generally desired to possess constant modulus and excellent auto- and cross-correlation properties. Consequently, the probing waveforms can be synthesized in two stages: at the first stage, the covariance matrix  $\mathbf{R}$  of the transmitted waveforms is optimized, and at the second stage, a signal waveform matrix  $\mathbf{X}$  is determined whose covariance matrix is equal or close to the optimal  $\mathbf{R}$ , and which also satisfies some practically motivated constraints (such as constant modulus or low peak-to-average-power ratio (PAR) constraints). A cyclic algorithm for example, can be used for the synthesis of such an  $\mathbf{X}$ , where the synthesized waveforms are required to have good auto- and cross-correlation properties in time.

### **3.20.2.2.2 Array design**

For a phased-array radar system, the transmission of coherent waveforms allows for a narrow mainbeam and, thus, a high signal-to-noise ratio (SNR) upon reception. When the locations of targets in a scene are unknown, phase shifts can be applied to the transmitting antennas to steer the focal beam across an angular region of interest. In contrast, MIMO radar systems, by transmitting different, possibly orthogonal waveforms, can be used to illuminate an extended angular region over a single processing interval, as we have demonstrated above.

Waveform diversity permits higher degrees of freedom, which enables the MIMO radar system to achieve increased flexibility for transmit beampattern design. The assumptions used in the discussions above are that the positions of the transmitting antennas, which also affect the shape of the beampattern, are fixed prior to the construction of  $\mathbf{R}$  followed by the synthesis of  $\mathbf{X}$ . At the receiver, sparse, or thinned, array design has been the subject of an abundance of literature during the last 50 years. The purpose of sparse array design has been to reduce the number of antennas (and thus reduce the cost) needed to produce desirable spatial receiving beampatterns. The ideas behind sparse receive array methodologies can be extended to that of sparse, MIMO array design. For example, cyclic algorithms can be used to approximate desired transmit and receive beampatterns via the design of sparse antenna arrays. These algorithms can be seen as extensions to iterative receive beampattern designs.

### **3.20.2.2.3 Adaptive array processing at radar receivers**

Adaptive array processing plays a vital role at radar receivers, including those of MIMO radar. Conventional data-independent algorithms, such as the delay-and-sum approach for array processing, suffer from poor resolution and high sidelobe level problems. Data-adaptive algorithms, such as MVDR (Capon) receivers, have been widely used in radar receivers. These adaptive signal processing algorithms offer much higher resolution and lower sidelobe levels than the data-independent approaches. However, these algorithms can be sensitive to steering vector errors and also require a substantial number of snapshots to determine the second-order statistics (covariance matrices). To mitigate these problems, diagonal loading has been used extensively in practical applications to make adaptive algorithms feasible. However, too much diagonal loading makes the adaptive algorithm degenerate into

data-independent methods, and the diagonal loading level may be hard to determine in practice. Parametric methods tend to be sensitive to data model errors and are not as widely used as the aforementioned data-adaptive algorithms.

In MIMO radar, adaptive array processing is essential, especially because many of the simple tricks used to achieve the longer virtual arrays, such as randomized antenna switching (also called randomized time-division multiple access (R-TDMA)) and slow-time code-division multiple access (ST-CDMA), provide sparse random sampling. Because of such sampling, the high sidelobe level problem suffered by data-independent approaches are exacerbated. Moreover, most of the radar signal processing problems encountered in practice do not have multiple snapshots. In fact, in most practical applications, only a single data measurement snapshot is available for adaptive signal processing. For example, in synthetic aperture radar (SAR) imaging, just a single phase history matrix is available for SAR image formation. Moreover the phase history matrix may not be uniformly sampled. In MIMO radar applications, including MIMO-radar-based space-time adaptive processing (STAP), synergistic MIMO SAR imaging and ground moving target indication (GMTI), and untangling multiple paths for diverse radar operations such as those encountered by MIMO over-the-horizon radar (OTHR), we essentially have just a single snapshot available at the radar receiver, especially in a heterogeneous clutter environment.

Fortunately, the recent advent of iterative adaptive algorithms, such as the iterative adaptive approach (IAA) and sparse learning via iterative minimization (SLIM), obviate the need of multiple snapshots and the uniform sampling requirements but retain desirable properties, including high resolution, low side-lobe level, and robustness against data model errors, of the conventional adaptive array processing methods. Moreover, for uniformly sampled data, various fast implementation strategies of these algorithms have been devised to exploit the Toeplitz matrix structures. These iterative adaptive algorithms are particularly suited for signal processing at radar receivers. They can also be used in diverse other applications, such as in sonar, radio astronomy, and channel estimation for underwater acoustic communications.

---

### 3.20.3 Radio astronomy

Radio astronomy is the study of our universe by passive observation of extra-terrestrial radio frequency emissions. Sources of interest for astronomers include (among others) radio galaxies, pulsars, supernova remnants, synchrotron radiation from excited material in a star's magnetic field, ejection jets from black holes, narrowband emission and absorption lines from diffuse elemental or chemical compound matter that can be assayed by their characteristic spectral structure, and continuum thermal black body radiation emitted by objects ranging from stars to interstellar dust and gasses. The radio universe provides quite a different and complementary view to that which is visible to more familiar optical telescopes. Radio astronomy has enabled a much fuller understanding of the structure of our universe than would have been possible with visible light alone. With Doppler red shifting, the spectrum of interest ranges from as low as the shortwave regime near 10 MHz, to well over 100 GHz in the millimeter and submillimeter bands, and there are radio telescopes either in use or under development to cover much of this spectrum.

From the earliest days of radio astronomy, detecting faint deep space sources has pushed available technology to extreme performance limits. Early progress was driven by improvements in hardware with relatively straightforward signal processing and detection techniques. With the advent of large

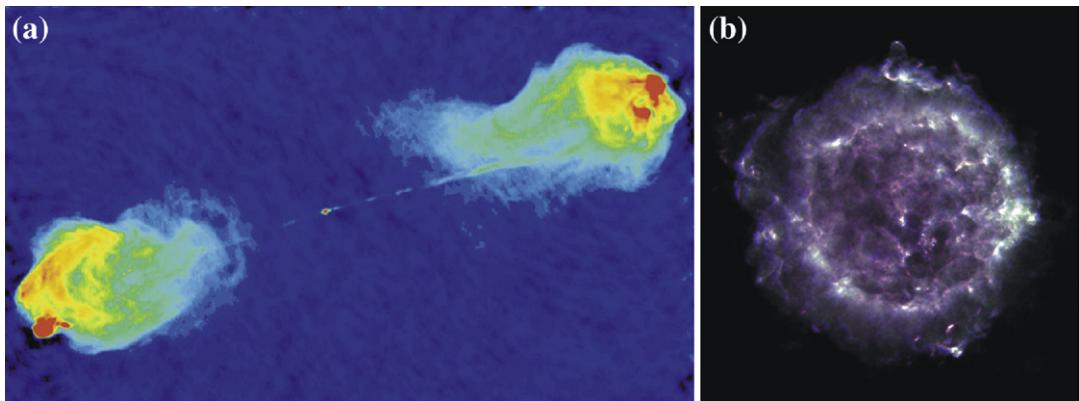
synthesis arrays, signal processing algorithms increased in sophistication. More recently, interest in phased array feeds (PAFs) has opened a new frontier for array signal processing algorithm development for radio astronomical observations.

Radio astronomy presents unique challenges as compared to typical applications in communications, radar, sonar, or remote sensing:

- *Low SNR*: Deep space signals are extremely faint. SNRs of  $-30$  to  $-50$  dB are routine.
- *Radiometric detection*: A basic observational mode in radio astronomy is “on-source minus off-source” radiometric detection where the source level is well below the noise floor and can only be seen by differencing with a noise only estimate. This requires stable power estimates of (i) system noise plus weak signal of interest (SOI) and (ii) noise power alone with the sensor steered off the SOI. The standard deviation of the noise power estimate determines the minimum detectable signal level, so that long integration times (minutes to hours) are required.
- *Low system temperatures*: With cryogenically cooled first stage low noise amplifiers, system noise temperatures can be as low as 15 K at L-band, including LNA noise, waveguide ohmic losses, downstream receiver noise, and spillover noise from warm ground observed beyond the rim of a dish reflector.
- *Stability*: System gain fluctuations increase the receiver output variance and place a limit on achievable sensitivity that cannot be overcome with increased integration time. High stability in gain, phase, noise, and beamshape response over hours is required to enable long term integrations to tease out detection of the weakest sources.
- *Bandwidth*: Some scientific observations require broad bandwidths of an octave or more. Digital processing over such large bandwidths poses serious computational burdens.
- *Radio frequency interference (RFI)*: Observations in RFI environments outside protected frequency bands are common. Interference levels below the noise floor may be as problematic as strong interferers, since they are hard to identify and attenuate. Cancelation approaches also cause pattern rumble which limits sensitivity.

### 3.20.3.1 Synthesis imaging

Radio astronomical synthesis imaging uses interferometric techniques and some of the world’s largest sensor arrays to form high resolution images of the distribution of radio sources in deep space. Figure 20.12 presents two examples of the beautiful high resolution detail revealed by synthesis imaging from the Very Large Array (VLA) in New Mexico, and Figure 20.13 shows the VLA with its antennas configured in a compact central core configuration. The key to this technology is coherent cross-correlation processing (i.e., interferometry) of RF signals seen by pairs of widely separated antennas (up to 10s of kilometers and more). Each such antenna typically consists of a high gain dish reflector of 12–45 m diameter which serves as a single element in the larger array. At lower frequencies, in order to avoid difficulties of physically steering the large aperture needed for high gain, array elements may themselves be built up as electronically steered beamforming aperture “stations” using clusters of fixed bare antennas without a reflector (for example, the LOFAR array). Whether implemented with a collection of large dish telescopes, or with a beamforming array, these elements of the full imaging array provide a sparse spatial sampling of the waveform that would have been observed by a much larger, imaginary “synthetic” encompassing dish. Though the array cannot match the collecting areas of the



**FIGURE 20.12**

VLA images of radio sources not visible to optical astronomy. (a) An early image of the gas jet structures in Cygnus A (ejected from the spinning core of the radio galaxy in the constellation Cygnus) seen at 5.0 GHz 1983 by Perley, Carilli, and Dreher. (b) Supernova remnant Cassiopeia A, 1994 composite of 1.4, 4.0, and 8.4 GHz images, by Rudnick, Delaney, Keohane, Koralesky, and Rector.

*Credits: National Radio Astronomy Observatory/Associated Universities, Inc./National Science Foundation.*



**FIGURE 20.13**

The central core of the Very Large Array (VLA) in compact configuration.

*Credit: Dave Finley, National Radio Astronomy Observatory/Associated Universities, Inc./National Science Foundation.*

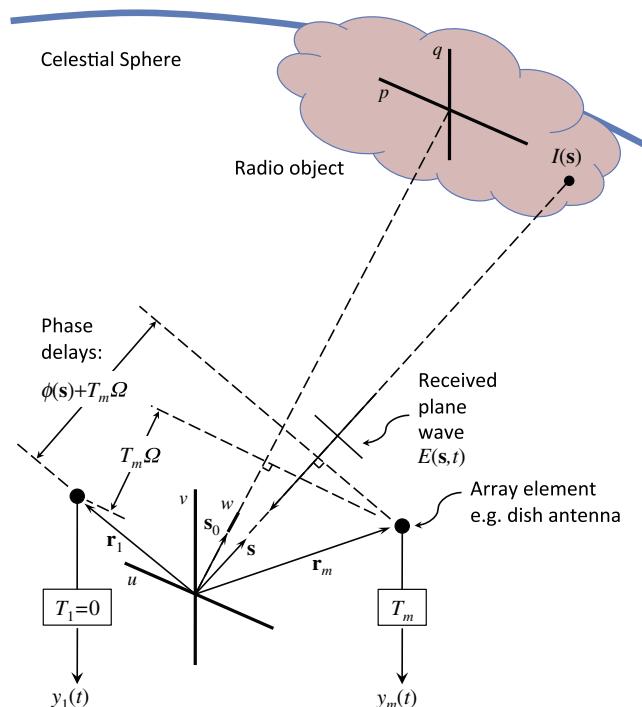
synthesized aperture, the long “baseline” distances between antennas yield spatial imaging resolution comparable to that of the encompassing dish aperture which inscribes the baseline vectors. Exploiting the earth’s rotation over time relative to the distant celestial sky patch being observed fills in sampling gaps between sparse array elements.

There are a number of aspects of synthesis imaging arrays that are distinct from many other array signal processing applications. Due to wide separation there is no mutual coupling and noise

is truly independent across the array. The large scale, long baselines, and critical dependence on phase relationships require very long coherent signal transport or precision time stamping of data sets using atomic clock references. Each array element is itself a high gain, highly directive antenna with a sizable aperture. Precision array calibration is required, but due to large scale hardware this cannot be done in a laboratory or on an antenna range. Self calibration methods are employed that use known point-source deep space objects in the field of view to properly phase the array. Array geometry is sparse with either random-like or log-scaled spacing. Extreme stability is required due to the need for coherent integration over hours, and bandwidths of interest can cover and octave or more.

### 3.20.3.1.1 The imaging equation

While the signals of interest are broadband, processing typically takes place in frequency subchannels so that narrowband models can typically be used. Further, since deep space sources are typically seen through line-of-sight propagation, multipath scattering is limited and occurs only locally as reflections off antenna support structures. Thus the propagation channel can be considered to be memoryless (zero delay spread). The synthesis imaging equations relate the observed cross correlation between pairs of array elements to the expected electromagnetic source intensity spatial distribution over a patch of the



**FIGURE 20.14**

Geometry and signal definitions for the synthesis imaging equations.

celestial sphere. Figure 20.14 illustrates the geometry, signal definitions, and coordinate systems for one of the baseline pairs of antennas used to develop the imaging equations.

Consider the electric field  $E(\mathbf{s}, t) = E(\mathbf{s})e^{j\Omega t}$  observed by the array at frequency  $\Omega$  due to a narrowband plane wave signal arriving from the direction pointed to by the unit length 3-space vector  $\mathbf{s}$ . We consider only the quasi monochromatic case where a single radiation frequency  $\Omega$  is observed by subband processing. To simplify discussion, polarization effects are not considered so  $E(\mathbf{s})$  is treated as a scalar rather than vector quantity, though working synthesis arrays typically have dual polarized antennas and receiver systems to permit studying source polarization. Since distance is indeterminate to the array, in our model the observed  $E(\mathbf{s})$  and its corresponding intensity distribution  $I(\mathbf{s}) = E[|E(\mathbf{s})|^2]$  are projected without time or phase shifting onto a hypothetical far-field celestial sphere that is interior to the nearest observed object. The goal of synthesis imaging is to estimate  $I(\mathbf{s})$  from observations of sensor array  $\mathbf{y}(t)$ .

Define the image coordinate axes  $(p, q)$  to be fixed on the celestial sphere and centered in the imaging field of view patch. Since  $\mathbf{s}$  is unit length, we may use these coordinates to express it as  $\mathbf{s} = (p, q, \sqrt{1 - p^2 - q^2})$ . Let  $\mathbf{s}_0$  point to the  $(p = 0, q = 0)$  origin, thus  $\mathbf{s}_0 = (0, 0, 1)$ . For small values of  $p$  and  $q$ , such as being contained within a field-of-view limited by the narrow beamwidth of array antennas,  $\mathbf{s} \approx (p, q, 1)$ . Time delays  $T_m$  are inserted in the signal paths for receiver outputs  $y_m(t)$  to compensate for the differential propagation times of a plane wave originating from the  $(p, q)$  origin. The most distant antenna is arbitrarily designated as the  $m = 1$ st element, and  $T_1 = 0$ . Thus the array is co-phased for a signal propagating along  $\mathbf{s}_0$ .

Receiver output voltage signal  $y_m(t)$ ,  $1 \leq m \leq M$ , is given by the superposition of scaled electric field contributions from across the full celestial sphere surface  $S$ , plus local sensor noise:

$$y_m(t) = \int_S A(\mathbf{s}) E(\mathbf{s}) e^{j(\Omega t + \phi_m(\mathbf{s}))} d\mathbf{s} + n_m(t), \quad (20.23)$$

where  $A(\mathbf{s})$  represents the known antenna element directivity pattern and downstream receiver gain terms,  $\phi_m(\mathbf{s})$  is the phase shift due to differential geometric propagation distances for a source from  $\mathbf{s}$  relative to a co-phased source from  $\mathbf{s}_0$  as shown in Figure 20.14, and  $\mathbf{n}_m(t)$  is the noise seen in the  $m$ th array element. For simple imaging algorithms, it is assumed that all elements (e.g., dish antennas) have identical spatial response patterns and that each is steered mechanically or electronically to align its beam mainlobe with  $\mathbf{s}_0$ , so  $A(\mathbf{s})$  does not depend on  $m$  and sources outside the elemental beams are strongly attenuated. The beamwidth defined by  $A(\mathbf{s})$  determines the maximum imaging field of view, or patch size. Considering the full array, (20.23) can be expressed in vector form as:

$$\mathbf{y}(t) = \int_S A(\mathbf{s}) E(\mathbf{s}) e^{j(\Omega t + \phi(\mathbf{s}))} d\mathbf{s} + \mathbf{n}(t) \quad (20.24)$$

where  $\phi(\mathbf{s}) = [\phi_1(\mathbf{s}) \cdots \phi_M(\mathbf{s})]^T$ .

Consider the vector distance between two array elements,  $(\mathbf{r}_l - \mathbf{r}_m)$ ,  $l \neq m$ , where  $\mathbf{r}_m$  is the location of the  $m$ th antenna. This is known as an interferometric “baseline,” and it plays a critical role in synthesis imaging. Longer baselines yield higher resolution images by increasing the synthetic array aperture diameter, and using more antennas provides more distinct baseline vectors which will be shown to more fully sample the image in the angular spectrum domain. In the following all functions of element

position depend only on such vector differences, so it is convenient to define a relative coordinate system  $(u, v, w)$  in the vicinity of the array to express the difference as  $(\mathbf{r}_l - \mathbf{r}_m) = (u, v, w)$ . Align  $(u, v)$  with  $(p, q)$ , and  $w$  with  $\mathbf{s}_0$ . Scale these axes so distance is measured in wavelengths, i.e., so that a unit distance corresponds to one wavelength  $\lambda = \frac{2\pi c}{\Omega}$ , where  $c$  is the speed of light. In this coordinate system we have by simple geometry

$$\phi_m(\mathbf{s}) + T_m \Omega = -2\pi \mathbf{s}(\mathbf{r}_m - \mathbf{r}_1), \text{ and } T_m \Omega = -2\pi \mathbf{s}_0(\mathbf{r}_m - \mathbf{r}_1). \quad (20.25)$$

At array outputs  $y_m(t)$ , after the inserted delays  $T_m$ , the effective phase difference between two array elements is then

$$\phi_l(\mathbf{s}) - \phi_m(\mathbf{s}) = -2\pi (\mathbf{s} - \mathbf{s}_0)^T (\mathbf{r}_l - \mathbf{r}_m). \quad (20.26)$$

Using the signal models of (20.23) and (20.26), the cross correlation of two antenna signals as a function of their positions is given by:

$$R(\mathbf{r}_l, \mathbf{r}_m) = E[y_l(t)y_m^*(t)] \quad \text{for } l \neq m \quad (20.27)$$

$$\begin{aligned} &= E \left[ \left( \int_S A(\mathbf{s}) E(\mathbf{s}) e^{j(\Omega t + \phi_l(\mathbf{s}))} d\mathbf{s} + n_l(t) \right) \right. \\ &\quad \times \left. \left( \int_S A(\mathbf{s}') E(\mathbf{s}') e^{j(\Omega t + \phi_m(\mathbf{s}'))} d\mathbf{s}' + n_m(t) \right)^* \right] \end{aligned} \quad (20.28)$$

$$\begin{aligned} &= \int_S |A(\mathbf{s})|^2 I(\mathbf{s}) e^{-j2\pi(\mathbf{s}-\mathbf{s}_0)^T(\mathbf{r}_l-\mathbf{r}_m)} d\mathbf{s} \\ &= \int_S |A(\mathbf{s})|^2 I(\mathbf{s}) e^{-j2\pi(p,q,a-1)^T(u,v,w)} d\mathbf{s} \end{aligned} \quad (20.29)$$

$$= \iint_{-\infty}^{\infty} |A(p, q)|^2 \frac{1}{a} I(p, q) e^{-j2\pi(up+vq+w(a-1))} dp dq \quad (20.30)$$

$$\approx \iint_{-\infty}^{\infty} |A(p, q)|^2 I(p, q) e^{-j2\pi(up+vq)} dp dq = R(u, v) \quad \text{for } u, v \neq 0, \quad (20.31)$$

where  $a = \sqrt{1 - p^2 - q^2}$ . We have assumed zero mean spatially independent radiators for  $E(\mathbf{s})$  and  $n_m(t)$ , a narrow field of view so  $a \approx 1$ , and that  $\mathbf{s}_0 = (0, 0, 1)$ . The quantity  $R(u, v)$  is known by radio astronomers as a “visibility function” where arguments  $\mathbf{r}_l$  and  $\mathbf{r}_m$  are replaced by  $u$  and  $v$  since the final expression depends only on these terms. A cursory inspection of (20.31) reveals that it is precisely a 2-D Fourier transform relationship, so the inversion method to obtain  $I(\mathbf{s})$  from visibilities  $R(u, v)$  suggests itself:

$$I(p, q) = \frac{1}{|A(p, q)|^2} \iint_{-\infty}^{\infty} R(u, v) e^{j2\pi(up+vq)} du dv, \quad \forall \{(p, q) | A(p, q) \gg 0\} \quad (20.32)$$

$$= \frac{1}{|A(p, q)|^2} F^{-1}(R(u, v)), \quad (20.33)$$

where  $F^{-1}(\cdot)$  is the inverse 2-D Fourier transform. This is the well known synthesis imaging equation. Since only cross correlations between distinct antennas are measured by this imaging interferometer,

the self power terms  $R(\mathbf{r}_l, \mathbf{r}_m)|_{\mathbf{r}_l=\mathbf{r}_m} = R(0, 0)$  are not computed or used in the Fourier inverse. The d.c. level in the image which normally depends on these terms must rather be adjusted to provide a black, zero valued background.

### 3.20.3.1.2 Algorithms for solving the imaging equation

The geometry of the imaging problem described in (20.32) and illustrated in Figure 20.14 is continually changing due to Earth rotation. The fixed ground antenna positions  $\mathbf{r}_m$  rotate relative to the  $(u, v)$  axis, which remains aligned to the  $(p, q)$  axis fixed on the celestial sphere. On one hand, this is a negative effect because it limits the integration time that can be used to estimate  $R(u, v)$  under a stationarity assumption. On the other hand, rotation produces new baseline vectors  $(\mathbf{r}_l - \mathbf{r}_m)$  with distinct orientations, filling in the Fourier space coverage for  $R(u, v)$  and improving image quality. To exploit rotation, imaging observations are made over long time periods, up to 12 h, to form a single image.

Receiver outputs are sampled as  $\mathbf{y}(i) \equiv \mathbf{y}(iT_s)$  at frequency  $f_s = 1/T_s$ , and sample covariance estimates of the visibility function (assuming zero mean signals) are obtained as

$$\widehat{\mathbf{R}}_k = \frac{1}{N} \sum_{i=kN}^{(k+1)N-1} \mathbf{y}(i)\mathbf{y}^H(i), \quad (20.34)$$

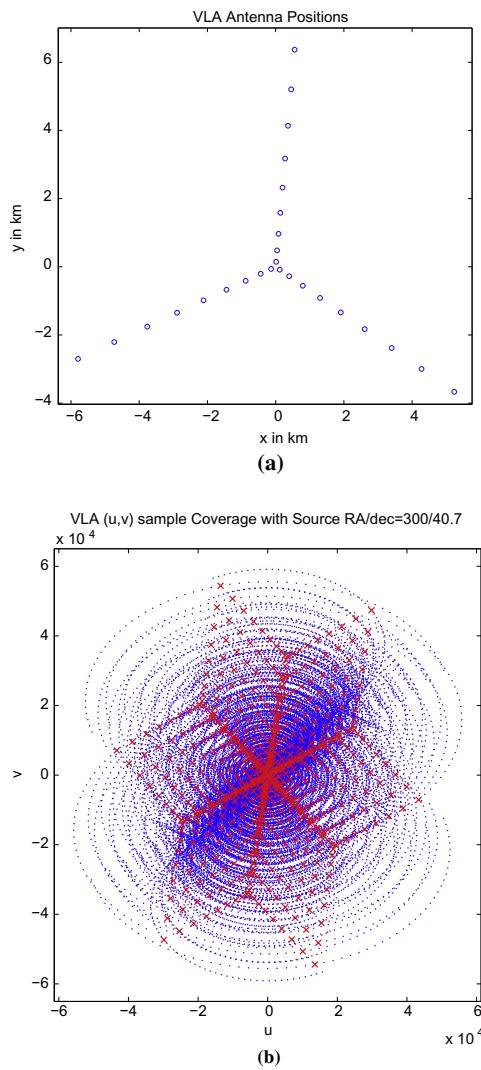
where  $N$  is the number of samples in the long term integration (LTI) window over which the imaging geometry and thus cross correlations may be assumed to be approximately stationary, and  $k$  is the LTI index. (We will later introduce a short term integration window length  $N_{\text{sti}}$  over which moving interference sources appear statistically stationary.)

Since covariance estimates are only available at discrete time intervals (one per LTI index  $k$ ), and the antennas have fixed Earth positions, only samples of  $R(u, v)$  are available with irregular spacing in the  $(u, v)$  plane, so (20.32) must be solved with discreet approximations. However, noting that due to Earth rotation, the corresponding antenna position vector orientations  $\mathbf{r}_m$  depend on time through  $k$ , a new set of  $(u, v)$  samples with different locations is available at each LTI. Index  $k$  is thus added to the notation to distinguish distinct baseline vectors  $(\mathbf{r}_{k,l} - \mathbf{r}_{k,m})$  for the same antenna pairs during different LTIs. So the  $(l, m)$ th element of  $\widehat{\mathbf{R}}_k$  relates to the sampled visibility function as

$$\{\widehat{\mathbf{R}}_k\}_{lm} = \widehat{R}_{k,lm} \approx R(u_{k,lm}, v_{k,lm}), \quad (20.35)$$

where  $(u_{k,lm}, v_{k,lm}, w_{k,lm}) = (\mathbf{r}_{k,l} - \mathbf{r}_{k,m})$  and where as in (20.26) and (20.31), due to inserted time delays  $T_m$  we may take  $w_{k,lm}$  to be zero. For simplicity we will use a single index  $\kappa$  to represent unique LTI-antenna index triples  $\{k, lm\}$  to specify vector samples in the  $(u, v)$  plane, so  $(u_{k,lm}, v_{k,lm}) = (u_\kappa, v_\kappa)$  and  $\widehat{R}_{k,lm} = \widehat{R}_\kappa$ . Thus elements of the sequence of matrices  $\widehat{\mathbf{R}}_k$  provide a non-uniformly sampled representation of the visibility function, or frequency domain image. Consistent with the treatment of  $R(0, 0)$  in (20.32), diagonal elements in  $\widehat{\mathbf{R}}_k$  are set to zero.

Figure 20.15a presents an example of a certain VLA geometry, and Figure 20.15b shows where the  $(u_\kappa, v_\kappa)$  samples would lie, with each point representing a unique sample  $\kappa$ . This plot includes 61 LTIs (i.e.,  $0 \leq k \leq 60$ ) over a 12 h VLA observation for the Cygnus A radio galaxy of Figure 20.12a. This sample pattern would change for sources with different positions on the celestial sphere (expressed by astronomers in right ascension and declination).

**FIGURE 20.15**

(a) An example VLA antenna element geometry with the repositionable 25 m dishes in a compact log spacing along the arms. Axis units are in kilometers. (b) Corresponding  $(u, v)$  sample grid for a 12 h observation of Cygnus A. Each point represents a  $(u_k, v_k)$  sample corresponding to a unique baseline vector where a visibility estimate  $\hat{R}_k$  is available. Red crosses denote baselines from a single LTI midway through the observation, and blue points are additional samples available using Earth rotation, with a new  $\hat{R}_k$  computed every 12 min. Observation is at 1.61 GHz and axis units are in wavelengths. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this book.)

With this frequency domain sampling and including noise effects (20.32) becomes

$$\widehat{I}_D(p, q) = \frac{1}{|A(p, q)|^2} \iint_{-\infty}^{\infty} \Psi(u, v) (R(u, v) + \widetilde{R}(u, v)) e^{j2\pi(up+vq)} du dv \quad (20.36)$$

$$= \frac{1}{|A(p, q)|^2} \sum_{\kappa} \widehat{R}_{\kappa} e^{j2\pi(u_{\kappa}p+v_{\kappa}q)}, \quad (20.37)$$

where  $\widehat{I}_D(p, q)$  is known as the “dirty image,” the sampling function  $\Psi(u, v) = \sum_{\kappa} \delta(u - u_{\kappa}, v - v_{\kappa})$ , and  $\widetilde{R}(u, v)$  represents sample estimation error in the covariance/visibility. Since the  $(u, v)$  plane is sparsely sampled,  $\Psi(u, v)$  introduces a bias in the inverse which must be removed by deconvolution as described below. This also means that (20.37) is not a true inverse Fourier transform due to the limited set of basis functions used. It is referred to as the “direct Fourier inverse” solution.

There are two common approaches to solving (20.36) or (20.37) for  $\widehat{I}_D(p, q)$  given a set of LTI covariances  $\widehat{R}_{\kappa}$ . The most straightforward though computationally intensive method is a brute force evaluation of (20.37) given knowledge of the  $(u_{\kappa}, v_{\kappa})$  sample locations (e.g., as in Figure 20.15). Alternately, the efficiencies of a 2-D inverse FFT can be exploited if these samples and corresponding visibilities  $\widehat{R}_{\kappa}$  are re-sampled on a uniform rectilinear grid in the  $(u, v)$  plane. “Cell averaging” assigns the average of visibility samples contained in a local cell region to the new rectilinear grid point in the middle of the cell. Other re-gridding methods based on higher order 2-D interpolation have also been used successfully. When large fields of view are required, or array elements are not coplanar, then any of these approaches based on (20.31) will not work and a solution to the more complete expression of (20.30) must be found. Cornwell has developed the W-Projection method to address these conditions [27].

An alternate “parametric matrix” representation of (20.31) and (20.37) has been developed. This is particularly convenient because it models the imaging system in a familiar array signal processing form that lends itself readily to analysis, adaptive array processing and interference canceling, and opens up additional options for solving the synthesis imaging and image restoration problems. Returning to the indexing notation of (20.34), note that since  $(\mathbf{r}_l - \mathbf{r}_m) = (\mathbf{r}_l - \mathbf{r}_1) - (\mathbf{r}_m - \mathbf{r}_1)$  one may express  $(u_{k,l}, v_{k,l})$  as  $(u_{k,l1} - u_{k,m1}, v_{k,l1} - v_{k,m1})$ . Let  $J(p, q) = |A(p, q)|^2 I(p, q)$  be the desired image as scaled (i.e., vigneted) by the antenna beam pattern, and sample it on a regular 2-D grid of pixels  $(p_d, q_d)$ ,  $1 \leq d \leq D$ . The conventional visibility Eq. (20.31) then becomes

$$R_{k,lm} = \sum_{d=1}^D J(p_d, q_d) e^{-j2\pi(u_{k,lm}p_d+v_{k,lm}q_d)} + \sigma_n^2 \delta(l-m) \quad (20.38)$$

$$= \sum_{d=1}^D e^{-j2\pi(u_{k,l1}p_d+v_{k,l1}q_d)} J(p_d, q_d) e^{j2\pi(u_{k,m1}p_d+v_{k,m1}q_d)} + \sigma_n^2 \delta(l-m), \quad (20.39)$$

which in matrix form is

$$\mathbf{R}_k = \mathbf{A}_k \mathbf{J} \mathbf{A}_k^H + \sigma_n^2 \mathbf{I}, \text{ where} \quad (20.40)$$

$$\mathbf{A}_k = [\mathbf{a}_{k,1}, \dots, \mathbf{a}_{k,D}], \quad (20.41)$$

$$\mathbf{a}_{k,d} = \left[ e^{-j2\pi(u_{k,11}p_d+v_{k,11}q_d)}, \dots, e^{-j2\pi(u_{k,M1}p_d+v_{k,M1}q_d)} \right]^T, \quad (20.42)$$

and where  $\mathbf{J} = \text{Diag}([J(p_1, q_D), \dots, J(p_D, q_D)])$  is the diagonal image matrix representation of sampled  $J(p, q)$ ,  $M$  is the total number of array elements, and though noise is independent across antennas, the self noise terms have been included to allow for the  $l = m$  case that contributes to the diagonal of full matrix  $\mathbf{R}_k$ . The matrix discrete “direct Fourier inverse” relationship corresponding to (20.37) is

$$\widehat{\mathbf{J}} = \frac{1}{K} \sum_{k=1}^K \mathbf{A}_k^H \mathbf{R}_k \mathbf{A}_k, \quad (20.43)$$

where  $K$  is the number of available LTIs. Equations (20.40) and (20.43) are well suited to address synthesis imaging as an estimation problem, facilitating use of Maximum Likelihood, maximum a posteriori, constrained minimum variance, or robust beamforming techniques. Note that (20.43) is not a complete discrete inverse Fourier transform, indeed, often  $D > MK$  so a one-to-one inverse relationship between  $\mathbf{R}_k$  and  $\mathbf{J}$  does not exist and  $\widehat{\mathbf{J}}$  is significantly blurred.

By the Fourier convolution theorem, the effect of frequency sampling by  $\Psi(u, v)$  in (20.36) is to convolve the desired image  $I(p, q)$  with the “dirty beam response”  $\psi_D(p, q) = F^{-1}(\Psi(u, v))$ . Neglecting the effect of individual antenna directivity pattern  $A(p, q)$ ,  $\psi_D(p, q)$  can be interpreted as the point spread function, or synthetic beam pattern of the imaging array for the given observation scenario. Significant reduction of this blurring effect can be achieved by an image restoration/deconvolution step. The dirty image of (20.36) may be expressed as

$$\widehat{I}_D(p, q) = \psi_D(p, q) * (I(p, q) + \tilde{I}(p, q)), \quad (20.44)$$

where  $\tilde{I}(p, q) = F^{-1}(\tilde{R}(u, v))$  is due to sample estimation error in the visibilities. Since antenna locations in the rotating  $(u, v)$  plane are known precisely over the full observation,  $\psi_D(p, q)$  is known to high accuracy, and with well calibrated dish antennas so is  $A(p, q)$ . Thus (20.44) may be solved as

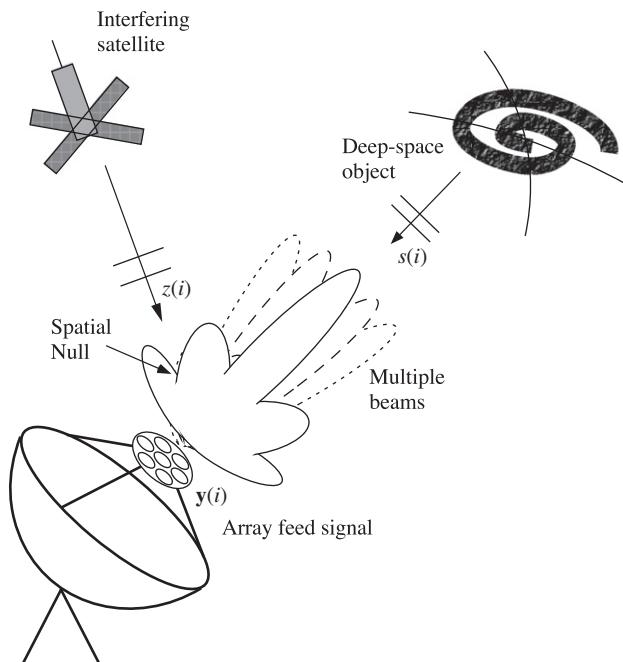
$$\widehat{I}(p, q) = \widehat{I}_D(p, q) *^{-1} \psi_D(p, q), \quad (20.45)$$

where “ $*^{-1}$ ” denotes deconvolution with respect to the right argument. Due to the spatial lowpass nature of dirty beam  $\psi_D(p, q)$  this problem is ill conditioned and must be regularized by imposing some assumptions about the image. The most popular reconstruction methods impose a sparse source distribution model and use an iterative source subtraction approach related to the original CLEAN algorithm [32]. The sparse model is justifiable for point-source images of star fields, and works well even with more complex distributions of gas and nebular structures given that much of the field of view is expected to be dark. Several variants and extensions to CLEAN have been proposed, some applying source subtraction in the spatial  $(p, q)$  domain, and some in the frequency  $(u, v)$  domain. Typically these have performance tuning parameters which astronomers adjust for most pleasing results. Thus the effective regularization term or mathematical optimization expression is often not known precisely and the process is a bit ad hoc, but solutions with higher contrast and resolution, and with reduced noise and reconstruction artifacts are preferred. Maximum entropy reconstruction has also been used effectively.

### 3.20.3.2 Astronomical phased array feeds

A new application for array signal processing in radio astronomy is phased array feeds (PAFs) where the traditional single large horn antenna feed at the focus of large telescope dish is replaced with a closely spaced (order of 1/2 wavelength) 2-D planar array of small antennas located at the dish focal plane. The primary motivation for such a system, as shown in Figure 20.16 is to form multiple simultaneous beams steered to cover a grid pattern in a field of view that is many times larger than the single pixel horn fed dish. PAFs are ideal for wide-field and survey instruments where it is desired to cover large regions of the sky in the shortest possible time. They provide the ability to capture a small image over the field of view, with one pixel per simultaneously formed beam, using a single snapshot pointing of the dish. Such systems have been referred to as “radio cameras.” Additional advantages of PAFs include sensitivity optimization with respect to the noise environment, and spatial interference cancellation capabilities (see Figure 20.16 and Section 3.20.3.3) albeit at the expense of increased hardware and processing complexity.

In some ways PAF processing is simply conventional beamforming for an array of microwave receiving antennas, but there are several unique aspects of the application that provide some challenges.



Radio telescope dish with a phased array feed

**FIGURE 20.16**

The primary advantage of FPA telescopes is increased field of view provided by multiple, simultaneously formed beams. Spatial cancellation of interfering signals is also possible, but very deep nulls are required.

The following technical hurdles are why PAFs have not been previously adopted in radio astronomy, but these issues have largely been resolved and working platforms have now been demonstrated.

First, the PAF is not a bare aperture array but operates in conjunction with a very large reflector which for an on-axis far field point source focusses a tight Airy pattern spot of energy at the array that spans little more than a single array element. For off-axis sources the spot moves across the array and undergoes coma shaped pattern distortion. So, though noise and interference are seen on all elements, only a few antennas see much of the SOI. The combined dish and PAF can be viewed as a dense array of small but high gain, highly directive elements, but not all of these have equal SNR. Elements outside the focal spot must however be used in beamforming to control the illumination pattern on the dish and thus reduce spillover noise from observing warm ground beyond the edge of the dish. The focal properties of the dish also limit the achievable field of view, even with electronic steering, since deviation from the boresight axis beyond a few beamwidths leads to defocusing and loss of gain, no matter how large the PAF is.

Second, array calibration is critical to achieve maximum sensitivity (gain over noise power) and due to the huge sizes of these instruments, must be performed *in situ* using known deep space objects as calibration sources of opportunity. Calibrations must be performed periodically (order of weeks) to account for electronic and structural drift, and must estimate array response vectors in every direction that a beam is to be steered or a response constraint is to be placed.

Third, beamformer weight calculation is non-trivial. Astronomers want maximum sensitivity and stable beampatterns on the sky, but these competing requirements are challenging. The variable correlated noise field environment of a radio telescope calls for an adaptive approach, but it is difficult to obtain low error array calibrations at enough points to control beam sidelobe structure. Also, due to complexity of the antenna structures, it is impossible to design usable beamformer weights from even a very detailed electromagnetic system simulation.

Fourth, as discussed in Section 3.20.3.3, many of the conventional adaptive canceling beamforming methods are not very effective for astronomical PAFs. This is because observations are frequently done when both the SOI and interference power levels are well below the noise floor. New approaches are required to form deeper spatial nulls in scenarios where it is difficult to estimate interference parameters.

Fifth, replacing a single horn feed channel with 38, or 200 array elements, as have been proposed for PAFs, has major implications on the back end processing. Processed bandwidths of 300 MHz or more per antenna are needed, so a real-time DSP processor with capacity to serve as digital receiver, multiple beamformer, and array correlator for calibrationm constitutes a major infrastructure investment.

And finally, in a field where cryogenically cooled antennas and LNAs are the norm to reduce receiver noise, cooling a large array is daunting. Most current development projects have opted for room temperature arrays and trade off the then necessary longer integration times with faster survey speeds possible with multiple beams.

### 3.20.3.2.1 Signal model

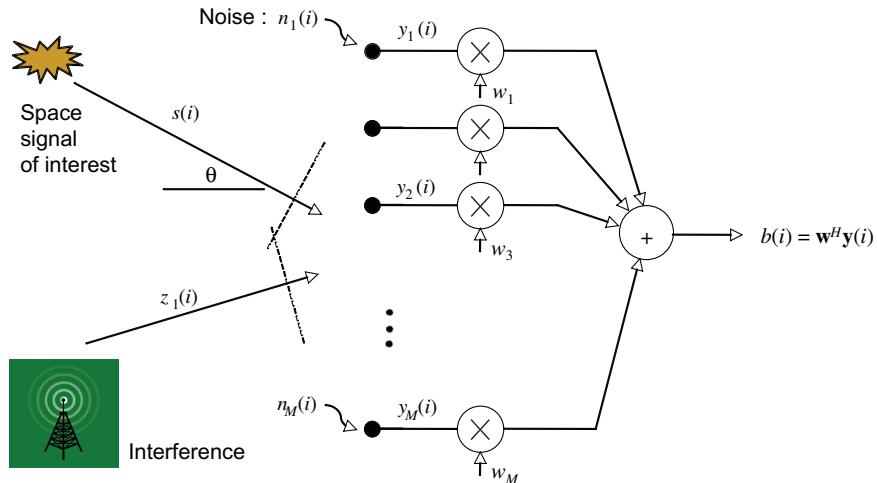
After analog frequency down conversion, sampling, and complex baseband bandshifting, the array signal time sample vector of Figure 20.16 is modeled as

$$\mathbf{y}(i) = \mathbf{as}(i) + \sum_{d=1}^D \mathbf{v}_d(i) z_d(i) + \mathbf{n}(i), \quad (20.46)$$

where  $\mathbf{a}$  is the array response vector for signal of interest (SOI)  $s(i)$ ,  $\mathbf{v}_d(i)$  is the time varying array response for the  $d$ th independent interfering source  $z_d(i)$ , and  $\mathbf{n}(i)$  is the noise vector. Source response  $\mathbf{a}$  is assumed to be constant, even for observation times on the order of an hour because the dish mechanically tracks a point in the sky. Even fixed ground interference sources must be modeled as moving (thus  $\mathbf{v}_d(i)$  depends on  $i$ ) due to this tracking motion of the dish. Approaches to address man-made interference are discussed in Section 3.20.3.3. This model for  $z_d(i)$  can also include natural deep space sources which are bright enough to overwhelm the SOI even when seen in the beam sidelobe pattern. Their apparent rotational motion about the SOI is due to Earth rotation. When the corresponding  $\mathbf{v}_d(i)$  is known accurately, these can be removed through a successive subtraction algorithm known as peeling. As with synthesis imaging, broadband processing for PAFs is accomplished by FFT based subband decomposition, often with thousands of frequency bins. So in the following we consider only a single frequency channel and adopt the standard narrowband array processing model.

Any array signal processing, including beamforming, must take into account the fact that, unlike synthesis imaging, the PAF noise vector  $\mathbf{n}(i)$  is correlated across the array. Even with cryogenic cooling, first stage amplifier LNA noise is correlated due to electromagnetic mutual coupling at the elements. Another major component, spillover noise from warm ground black body radiation as seen by the feed array, is spatially correlated because it is not isotropic since it stops above the horizon and is blocked over a large solid angle by the dish.

In a practical PAF scenario the beams are steered in a rectangular or hexagonal grid pattern with crossover points at the  $-1$  to  $-3$  dB levels. The total number of beams,  $J$ , is limited by the maximum steering angle which is determined by the diameter of the array feed and the focal properties of the dish, by the acceptable limit for beamshape distortion, and by the available processing capacity for real-time simultaneous computation of multiple beams. As illustrated in Figure 20.17, the time series output for



**FIGURE 20.17**

Beamformer architecture. Narrowband operation is assumed and for PAF beamforming, interaction with the large reflector dish is not shown.

a beam steered in the  $j$ th direction is given by

$$b_j(i) = \mathbf{w}_j^H \mathbf{y}(i), \quad (20.47)$$

where  $\mathbf{w}_j$  is the vector of complex weights for beamformer  $j$ ,  $1 \leq j \leq J$ . Weights are designed based on array calibration data and the desired response pattern constraints and optimization as described in the following two sections. Separate beamformers with their own sets of  $J$  distinct weight vectors are computed for each frequency channel, though we consider only a single channel in this discussion.

### 3.20.3.2.2 Calibration

Since multiple simultaneous beams are formed with a PAF as shown in Figure 20.16, a calibration for the signal array response vector  $\mathbf{a}_j$  must be performed for each direction,  $\mathbf{s}_j$ , corresponding to each formed beam's boresight direction, and any additional directions where point constraints in the beam pattern response will be placed. Periodic re-calibration is necessary due to strict beam pattern stability requirements, to correct for differential electronic phase and gain drift, and to characterize changes in receiver noise temperatures. Calibration is based on sample array covariance estimates  $\hat{\mathbf{R}}$  as described in (20.34) while observing a dominant bright calibration point source in the sky. For example, in the northern hemisphere, Cassiopeia A and Cygnus A shown in Figure 20.12 are the brightest continuum (broadband) sources, and with a typical single dish telescope aperture they are unresolved and appear as point sources. Both have been used as calibrators.

### 3.20.3.2.3 Beamformer calculation

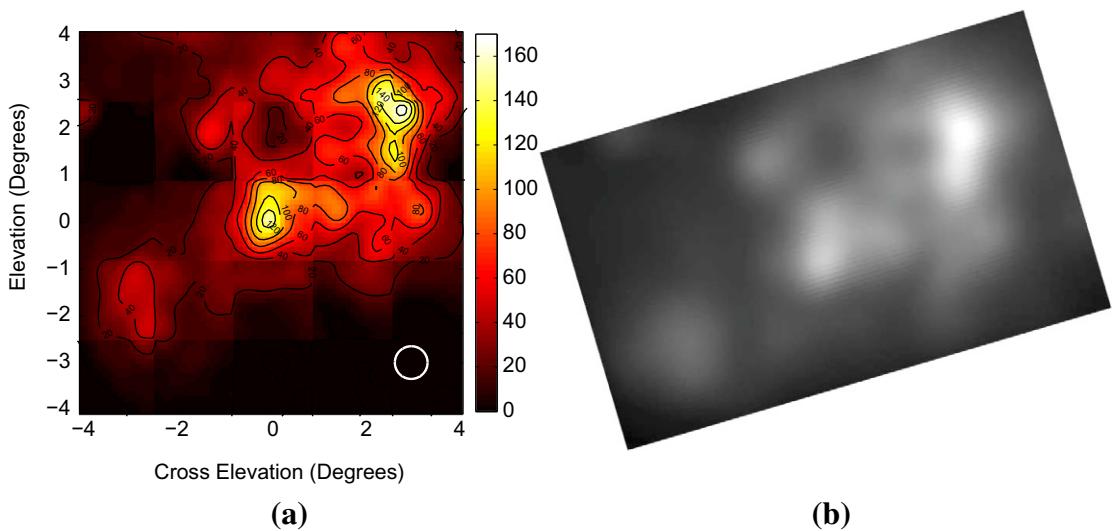
Since discovery of the weakest, most distant sources is a primary aim of radio astronomers, it is paramount to design a dish and feed combination to achieve high sensitivity, which has been derived for a phased array feed to be

$$\frac{A_e}{T_{\text{sys}}} = \frac{k_b B}{F_s} \frac{\mathbf{w}^H \mathbf{R}_s \mathbf{w}}{\mathbf{w}^H \mathbf{R}_n \mathbf{w}} \quad (\text{m}^2/\text{K}), \quad (20.48)$$

where  $A_e$  (in  $\text{m}^2$ ) represents directivity in terms of the effective antenna aperture collecting area,  $T_{\text{sys}}$  is system noise power at the beamformer output expressed as a black body temperature,  $k_b$  is Boltzmann's constant,  $B$  is system bandwidth,  $F_s$ , (in Watts/ $\text{m}^2$ ) is the signal flux density in one polarization, and  $\mathbf{R}_s$  and  $\mathbf{R}_n$  are the signal and noise components of  $\mathbf{R}$  respectively. Here we have assumed  $D = 0$ , i.e., that there are no interferers. For a reflector antenna with a traditional horn feed, maximizing sensitivity involves a hardware-only tradeoff between aperture efficiency, which determines the received signal power, and spillover efficiency, which determines the spillover noise contribution. With a PAF, sensitivity is determined by the beamforming weights as well as the array and receivers. Adjusting  $\mathbf{w}$  controls both the PAF illumination pattern on the dish which affects  $A_e$ , and the response to the noise field, which affects  $T_{\text{sys}}$ . Noting that all other right hand side terms in (20.48) are constant, sensitivity can be maximized with the well known maximum signal to noise ratio (SNR) beamformer

$$\mathbf{w}_{\text{snr}} = \arg \max_{\mathbf{w}} \frac{\mathbf{w}^H \mathbf{R}_s \mathbf{w}}{\mathbf{w}^H \mathbf{R}_n \mathbf{w}}. \quad (20.49)$$

To date all hardware demonstrated PAF telescopes have used this maximum sensitivity beamformer. However, a hybrid beamformer design method for PAFs that parametrically trades off sensitivity maximization with constraining mainlobe shape and sidelobe levels has been proposed.



**FIGURE 20.18**

(a) Cygnus X region at 1600 MHz.  $5 \times 5$  mosaic of images using the 19-element prototype PAF on the Green Bank 20-Meter Telescope. The circle indicates the half-power beamwidth. (b) Canadian Galactic Plane Survey image [38] convolved to the 20-m effective beamwidth. The center of the map is approximately  $20^{\text{h}}44^{\text{m}}, +42^{\circ}$  (J2000) with north to the upper left.

*Credit: Karl Warnick in [33].*

### 3.20.3.2.4 Radio camera results

In 2008, ASTRON and BYU/NRAO independently demonstrated the first radio camera images with a PAF fed dish. Figure 20.18 presents an example of the BYU work as a mosaic image of a complex source distribution in the Cygnus X region. As a comparison, the right image is from the Canadian Galactic Plane Survey image, but blurred by convolution with the equivalent beam pattern of the 20-Meter Telescope to match resolution scales. We expect that the image artifacts caused by discontinuities at mosaic tile boundaries could be eliminated with more sophisticated processing. The Cygnus X radio camera image contains approximately 3000 pixels. A more practical coarse grid spacing of about half the HPBW would require about 600 pixels. A single horn feed would require 600 pointings (one for each pixel) to form such an image, compared to 25 (one for each mosaic tile) for the radio camera. Thus for equal integration times per pixel, this radio camera provides an imaging speed up of 24 times.

### 3.20.3.3 Interference mitigation for radio astronomy

From a regulatory and spectrum management point of view, radio astronomy is a passive wireless service which must co-exist with many other licensed communications activities. Though international treaties have long been established to protect a few important frequency bands for astronomical use only (e.g., around 1420 MHz for emission lines of abundant deep space neutral Hydrogen) these precautions

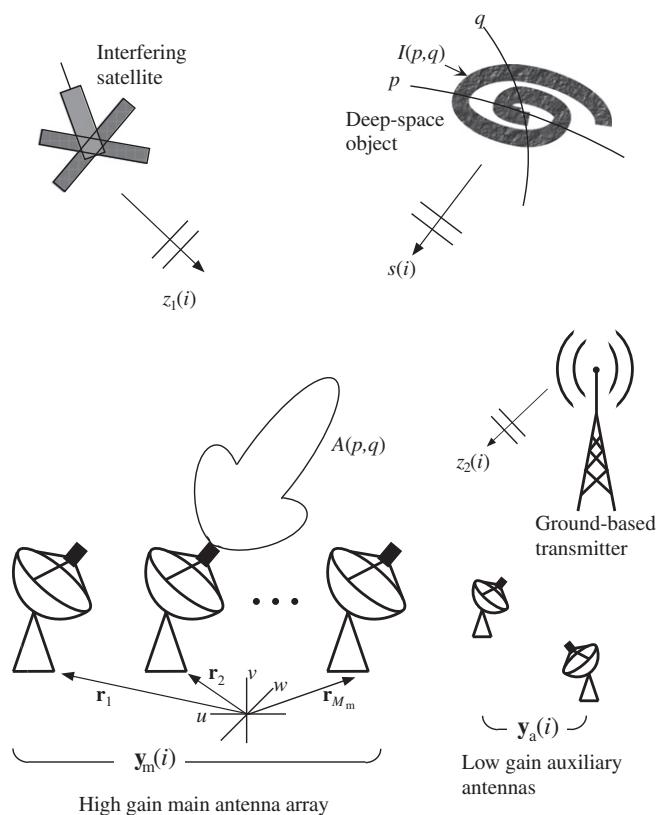
have become wholly inadequate. Astronomers' current scientific goals require observing emissions across the radio spectrum from molecules of more exotic gas compounds, from broad spectrum sources such as pulsars, and from highly red shifted objects nearing the edge of the observable universe where Doppler effects dramatically reduce the frequencies. Thus there is virtually no frequency band devoid of interesting sources to study. Astronomers cannot rely solely on protected bands and must develop methods to mitigate ubiquitous man-made radio transmission interference.

The problem is further exacerbated because one of the fundamental aims of radio astronomy is to discover the weakest of sources which are often at signal levels many tens of decibels below the noise floor. Successful detection usually requires long integration times on the order of hours to average out noise induced sample estimation error variance, combined with on-source minus off-source subtraction to find subtle differences in power levels between a noise-only background and noise plus SOI. Thus even very weak interference levels that would hardly hinder wireless communications can completely obscure an astronomical source of interest.

There is a long laundry list of troublesome RFI sources for radio astronomy. Examples of man-made signals encountered at radio observatories for which mitigation strategies have been demonstrated include: satellite downlink transmissions, radar systems, air navigation aids, wireless communications, and digital television broadcasts. Even locating instruments in undeveloped areas with regulatory protection for radio quiet zones does not avoid many man-made sources such as satellite downlinks. Low frequency synthesis arrays such as LOFAR, PAPER, LWA, and the Murchison Widefield Array operate in the heavily used VHF bands (30–300 MHz) to detect highly redshifted emissions, and as such must contend with very powerful commercial TV and FM radio broadcasts, as well as two-way mobile communications services.

There are a variety of RFI mitigation methods used in radio astronomy. The major approaches include *avoidance* (simply wait until the interference stops or observe in a different frequency band), *temporal excision* (blank out only the small percentage of data samples corrupted by impulsive interference), waveform subtraction (estimate parameters for known structured interference and subtract a synthetic copy of this signal from the data), *anti-coincidence* (remove local interference by retaining only signals common to two distant observing stations), and *spatial filtering* (adaptive array processing to place spatial nulls on interference). Since this present article emphasizes array signal processing, we will address spatial filtering in the following discussion.

Figures 20.16 and 20.19 illustrate interference scenarios for a phased array feed and synthesis imaging array respectively. For PAFs the closely packed antennas in the feed enable for the first time adaptive spatial filtering on single dish telescopes. This would also be possible with PAFs on the multiple dishes of a large imaging array, but even with just typical single horn feeds (as in Figure 20.19) the covariance matrix used to compute imaging visibilities as in (20.34) and (20.40) can also be used for interference canceling. Some proposed algorithms use only the main imaging array antennas, while others achieve improved performance with additional smaller auxiliary antennas trained on the interferers as shown in the figure. The various algorithm approaches will be discussed below. Most spatial filtering work to-date has been at frequencies in L-band (1–2 GHz) and below because this includes important astronomical sources and because of the abundance of man-made interference in these bands.

**FIGURE 20.19**

An RFI scenario at a synthesis imaging array. Two independent interference sources are illustrated: a satellite downlink and a ground-based broadcast transmitter. The main imaging array consists of typical single feed dishes (i.e., PAF feeds are not used here). In addition to the main array, a subarray of smaller auxiliary antennas is shown which can be used with some algorithms discussed below to improve cancelation performance. If tracking information is available, these auxiliaries are steered to the offending sources to provide a high INR copy of the interference.

### 3.20.3.3.1 Challenges and solutions to radio astronomical spatial filtering

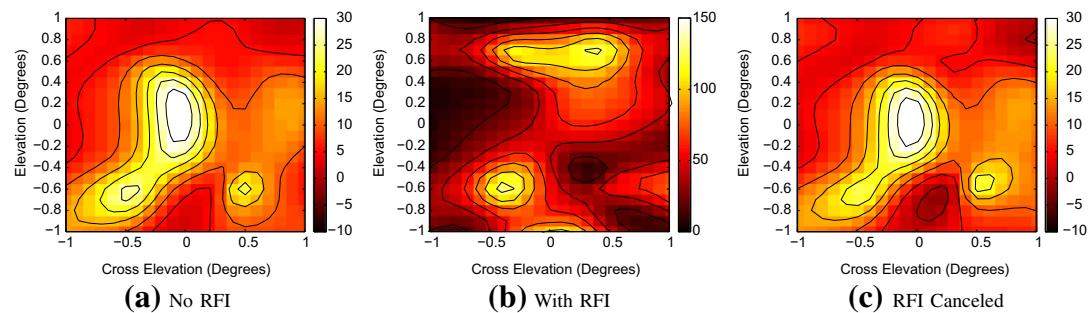
Many of the well-known adaptive beamforming algorithms appear at first glance to be promising candidates for interference mitigation in astronomical array processing, including maximum SNR, minimum variance distortionless response (MVDR or Capon), linearly constrained minimum variance (LCMV), generalized sidelobe canceler (GSC), Wiener filtering, and other algorithms. Robust canceling beamformers which are less sensitive to calibration error have also been considered for aperture arrays used as stations in large low frequency imaging arrays like LOFAR. However, due to several challenging characteristics of the radio astronomical RFI problem, most of these approaches are less successful here

than they would be in typical radar, sonar, wireless communications, or signal intercept applications. These problems have made many astronomers reluctant to adopt the use of adaptive array processing methods for regular scientific observations. We note though that the intrinsic motivations to observe in RFI corrupted bands are becoming strong enough that rapid progress toward adoption is necessary and is anticipated by most practitioners. New algorithm adaptations are being introduced which are better suited for radio astronomical spatial filtering. We consider below some of the significant aspects of radio astronomy that complicate spatial filtering.

The typical astronomical SOI power level is 30 dB or more below the system noise over comparable bandwidth, even when cryogenically cooled LNAs are used with instruments located in radio quiet zones. Canceling nulls must therefore be deep enough to drive interference below the SOI level, i.e., below the on-source minus off-source detection limit, not just down to the system noise level. Most algorithms require a dominant interferer to form deep nulls because minimum variance methods (MVDR, LCMV, max SNR, Wiener Filtering, etc.) which balance noise variance with residual interference power cannot drive a weaker interferer far below the noise floor. The residual will remain well above the SOI level.

Another promising solution to limited null depth is a zero forcing beamformer like subspace projection (SP) where the null in the estimated vector subspace for interference is theoretically infinitely deep. A number of proposed radio astronomical RFI cancelers have adopted the SP approach and some experimental demonstration results have appeared. Figure 20.20 illustrates the first use of subspace projection RFI mitigation with a PAF as reported in [53]. Data were collected from a 19 element PAF mounted on the 20-Meter Telescope at the NRAO Green Bank, West Virginia observatory while observing the deep space Hydroxyl Ion (OH) maser radiation source designated in star catalog as “W3OH.” An FM-modulated RFI source overlapping the W3OH spectral line at 1665 MHz was created artificially using a signal generator. The RFI was removed using the subspace projection algorithm. Snapshot radio camera images (see Section 3.20.3.2) of the source with and without RFI mitigation are shown in Figure 20.20. The source which was completely obscured by interference is now clearly visible.

Typically interference subspace estimation is poor in SP and all other cancelers without a dominant RFI signal so null depth suffers at lower INR levels. Short integration times, needed to avoid subspace



**FIGURE 20.20**

W3OH image with and without RFI. The color scale is equivalent antenna temperature (K).

smearing with moving interference, increase covariance sample estimation error which also limits null depth. To address these issues, an SP canceler using auxiliary antennas as in Figure 20.19 and a new parametric model-based SP approach for tracking low INR moving interferers have been proposed which significantly improves null depth [50].

Adaptive beamformers must distort the desired quiescent (interference free) beam pattern in order to place deep nulls on interferers. For astronomy, even modest beamshape distortions can be unacceptable. A small pointing shift in mainlobe peak response, or coma in the beam mainlobe can corrupt sensitive calibrated measurements of object brightness spatial distribution. Due to strict gain stability requirements it has been preferable to lose some observation time and frequency bands to interference rather than draw false scientific conclusions from corrupted on-sky beam patterns.

For PAF beamforming a potential solution is to use one of several classical constrained adaptive beamformers. Due to the inherent tendency for off-axis steered beams with a parabolic dish reflector to develop a mainlobe coma distortion, it would be necessary to employ several mainlobe point constraints to maintain a consistent symmetric beampattern. It has also been demonstrated that without multiple mainlobe constraints, RFI canceling nulls in the beampattern sidelobes can cause significant distortion in the mainlobe.

A more subtle undesirable effect for both PAF and synthesis imaging arrays is that variations in the effective sidelobe patterns due to moving RFI nulling can translate directly to an increase in the minimum detectable signal level for the radiometer. Weak astronomical sources can only be observed by integrating the received power for a long period to obtain separate low variance estimates of signal plus noise power (on source), and noise only (off source). Both signal and noise (including leakage from other deep space source through beam sidelobe patterns) must be stable to an extreme tolerance requirement over the full integration time. Even small variations in the sidelobe structure can significantly perturb background source and noise signal levels, causing intolerable time variation. This sidelobe pattern rumble due to adaptive cancelation increases the “confusion limit” to detection since unstable noise and background are not fully canceled in the on-source minus off-source subtraction. This occurs even if the beam pattern mainlobe is held stable using constrained or robust beamformer techniques.

### 3.20.4 Positioning and navigation

The Global Positioning System (GPS) is the most widely adopted positioning system in the world. It is a prominent example of what is known as Global Navigation Satellite Systems or GNSS, which represent any system that provides position information to users equipped with appropriate receivers at any time and anywhere around the globe based on signals transmitted from satellites. Currently there are two operating GNSS: GPS (developed by the USA) and Glonass (developed by the former USSR and now by Russia), while there are a number of systems under deployment, such as Galileo in Europe and Compass in China. Despite the differences in the satellite constellation, signal parameters, etc., all of these systems share the same operating principles and use similar types of signals. Therefore, while we will often refer to the case of GPS, all of what we discuss here is also applicable to the other systems as well.

The GPS constellation is formed by approximately 30 satellites orbiting at a distance of about 26,560 km from earth's center. Each satellite transmits several Direct-Sequence Spread-Spectrum

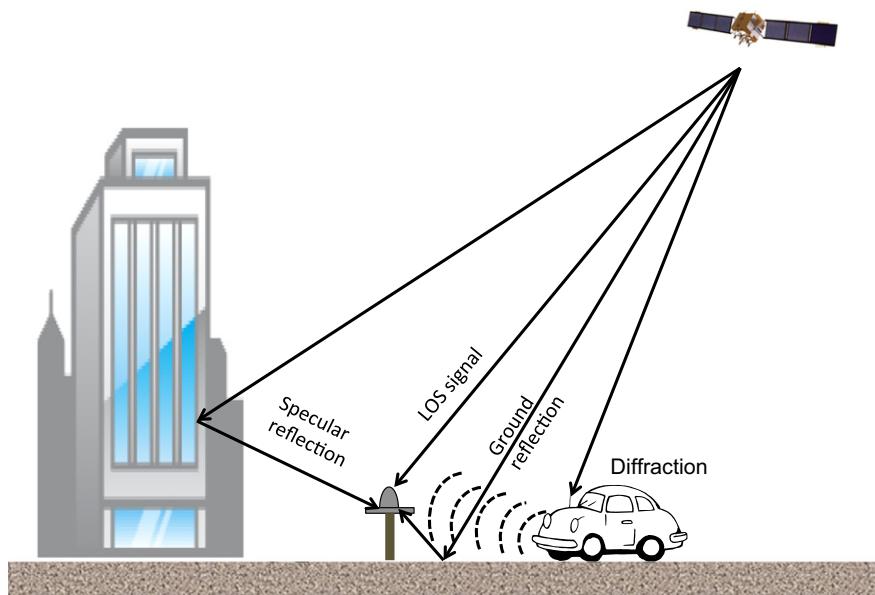
(DS-SS) signals, and the main task of a GPS receiver is to measure the distances to the satellites via the time delay of the signals. In applications requiring high-accuracy positions, the phase of the received signal is also used as a source of information about the propagation delay of the signal. Once the receiver has obtained these distances, it can compute its position by solving a geometrical problem. Apart from the satellites themselves, the core of a GNSS is the ground segment that consists of a set of ground stations monitoring the satellites and computing their positions.

Unlike communication receivers, where timing and phase synchronization are intermediary steps to recovering the transmitted information, for positioning receivers it is the synchronization that is the information. Significantly greater synchronization precision is required in a GNSS receiver than in a communications system. As discussed below, the positioning accuracy of GNSS is degraded by many effects. Multipath propagation and certain types of interference are very difficult to mitigate with single-antenna receivers. Spatial processing has proven to be the most effective approach to combat these sources of degradation, making it possible to obtain in some cases the same accuracy as in a multipath- and interference-free scenario. The next two sections describe the error sources in GNSS, with special emphasis on the multipath effects, and an appropriate signal model for spatial processing. They serve as a justification of why the use of antenna arrays in the context of GNSS has been receiving considerable attention since the mid-1990s. The rest of the sections discuss the advantages and limitations of different approaches for exploiting the spatial degrees of freedom or spatial diversity in satellite-based navigation systems.

### 3.20.4.1 Error sources and the benefits of antenna arrays in GNSS

The synchronization accuracy demanded by GPS receivers is very stringent, on the order of a few nanoseconds, and exceeds by far the levels usually required in communications receivers. The difficulties in achieving such ranging accuracy are due to the presence of different sources of error, which can be categorized in three groups: (i) the errors due to the ground segment and the satellites, (ii) propagation-induced errors, and (iii) local errors at the receiver. The first category includes the discrepancy between the estimates of the satellite positions and clocks, which are computed by the ground segment and broadcast by the satellites themselves, and the actual values. The second category corresponds to the changes in the propagation delay, phase and amplitude of the signals caused by the atmosphere. Finally, local errors refer to the effects of thermal noise, interference and multipath components.

The largest contributors to the total error budget are typically the ionospheric delay and local effects. The size of the errors in the first category is progressively decreasing as the ground segment and satellites are modernized. Moreover, one can also access alternative providers of more accurate satellite coordinates and clocks. Another option is to use differential methods, where the user receiver makes use of corrections computed by another receiver at a known position, or relative methods, where the position relative to that second receiver is computed. The use of differential or relative methods virtually eliminates the errors from the first category. These methods also help mitigate the propagation-induced errors. Alternatively, the ionospheric delay can be essentially canceled using measurements at two or more frequency bands. In short, the errors from the first two categories can typically be mitigated at the measurement or system levels, and hence the local errors remain as the limiting factor in the ultimate accuracy achievable with GNSS. This is the reason why it is of high interest to use signal processing techniques, and in particular antenna array-based methods, to combat multipath and interference effects in GNSS.

**FIGURE 20.21**

Environment with multipath propagation.

As in other systems, interference obviously affects the quality of time delay and phase estimates in GNSS. On the other hand, the study of multipath effects requires a different treatment to the one that is typically employed in wireless communications. While multipath components can be useful in communications systems as a source of diversity or to increase the total received signal power, they are always a source of error in navigation systems, and can lead to positioning inaccuracies reaching up to many tens of meters. For the case of a satellite-based transmission, multipath is produced by objects that are close to the receiver, as depicted in Figure 20.21. The only signal of interest in a navigation receiver is the line-of-sight (LOS) signal, since it conveys information about the transmitter-receiver distance through its time delay and phase information. While the multipath in a frequency-flat channel with zero delay-spread theoretically arrives at the same time as the LOS, the resulting fading can lead to signal drop-outs and poor localization performance. A second antenna (i.e., forming a small array) can be used to overcome this difficulty. More challenging are multipath signals that arrive with non-zero delay relative to the LOS, but still within 1–1.5 chip periods of the LOS (for civilian GPS, the chip period is  $1 \mu\text{s}$ , corresponding to about 300 m). Such signals are commonly referred to as *coherent* multipath, and cause biases in the LOS signal time delay and carrier phase estimates. Signal replicas with delays greater than about 1.5 chip periods can essentially be eliminated via the despreading process.

Narrowband or pulsed interference can be canceled in single antenna receivers using excision filters or pulse blanking. Wideband non-pulsed interference cannot be combatted with time-domain processing, but it is in principle an easy target for array processing. Harmful interference usually stands out clearly

above the noise, and this makes its identification and subsequent nulling with a spatial filter relatively easy. On the other hand, multipath mitigation is an extremely difficult task in single-antenna receivers and also a difficult problem when using antenna arrays. In the single-antenna case where time-domain methods must be used, the problem is ill-conditioned since one is attempting to estimate the parameters of signal replicas that are very similar to each other. If a reflection and the LOS signal differ by a very small delay (compared to the inverse of the signal bandwidth), they are almost identical and it is very difficult to accurately measure the exact LOS signal delay. On the other hand, the spatial selectivity offered by antenna arrays can be used to differentiate the LOS signal from multipath, since the multipath will arrive from directions different from the LOS (it is very unlikely to have reflectors close to the direct propagation path). The application of spatial processing for multipath mitigation is not without difficulties. The main problem is that the LOS signal and the coherent multipath are strongly correlated, which causes problems for many array processing techniques.

### 3.20.4.2 Signal model for positioning applications

The signal received by the antenna array can be written as

$$\mathbf{y}(t) = \sum_{k=0}^D \alpha_k \mathbf{a}_k x(t - \tau_k) e^{j2\pi f_k t} + \mathbf{n}(t). \quad (20.50)$$

In particular, in our problem the sources are not different signals, but delayed replicas of a single signal. Each replica is shifted by a different Doppler frequency  $f_k$ , and its complex amplitude is  $\alpha_k$ . The subindex 0 is reserved for the LOS signal, and this implies that  $\tau_i > \tau_0, \forall i$ . The term  $\mathbf{n}(t)$  includes the thermal noise and any (possibly directional) interference. The key parameters of interest for positioning applications are  $\tau_0$  and possibly the argument of  $\alpha_0$  (i.e.,  $\angle \alpha_0$ , which is the carrier phase of the LOS signal).

According to the discussion above, we assume that the delays of the replicas are in the range  $[\tau_0, \tau_0 + 1.5T_c]$ , where  $T_c$  is the chip duration. Each replica may represent a single reflection or a cluster of reflections with very similar delays. This leads to different possible parameterizations for the vectors  $\mathbf{a}_k$ , as listed below:

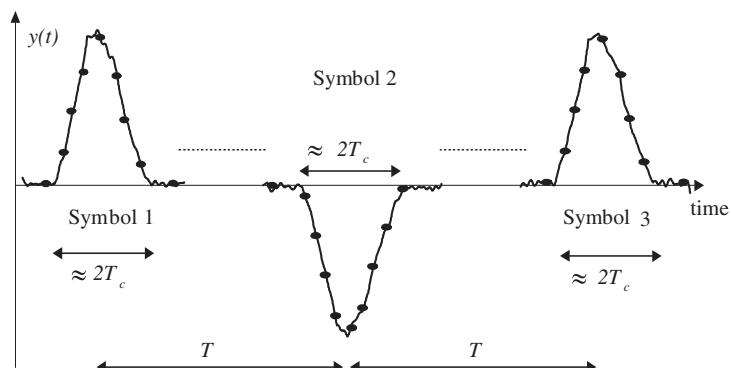
1. an unstructured spatial signature (i.e., each  $\mathbf{a}_k$  is an arbitrary complex vector). In this case, there is an inherent ambiguity between the definition of  $\alpha_k$  and  $\mathbf{a}_k$ , which can be simply avoided by defining  $\alpha_k \mathbf{a}_k$  as the overall spatial signature. One element of the spatial signature is identified as  $\alpha_k$ , and hence the carrier phase of the LOS signal is given by the argument of that element of the spatial signature.
2. a steering vector (or also referred to as structured spatial signature), which is a function of the DOA.
3. a weighted sum of steering vectors:  $\mathbf{a}_k = \sum_{l=1}^{D_k} \alpha_{k,l} \mathbf{a}_{k,l}(\theta_{k,l}, \phi_{k,l})$ , where each term corresponds to the amplitude and the steering vector of one of the reflections of the cluster. In this case, the ambiguity between  $\alpha_k$  and  $\mathbf{a}_k$  can be handled in the same way as in the first model.

The signal  $x(t)$  may represent the GNSS signal itself or the signal after some processing. The most common case of processing in our context is the despreading operation, which consists in cross-correlating the received signal with a local replica of the pseudorandom or pseudonoise (PN) sequence.

In this case, the variable  $t$  in the signals may be interpreted as the correlation lag. Unlike communications receivers, a single correlation lag is not sufficient. A single correlation lag may be appropriate for data detection but in a GNSS receiver, where the timing of the PN sequence has to be measured, several correlation lags are required. The correlation of the incoming signal with the local sequence is usually computed as a multiply-integrate-and-dump operation, which is carried out for each lag. However, the despread signal, depicted in Figure 20.22, can also be interpreted as a portion of the output of a matched filter. Figure 20.23 shows how the reception of multiple replicas affects the shape of the despread signal, and it is clear from there that identifying the components that form the signal is a very complicated task.

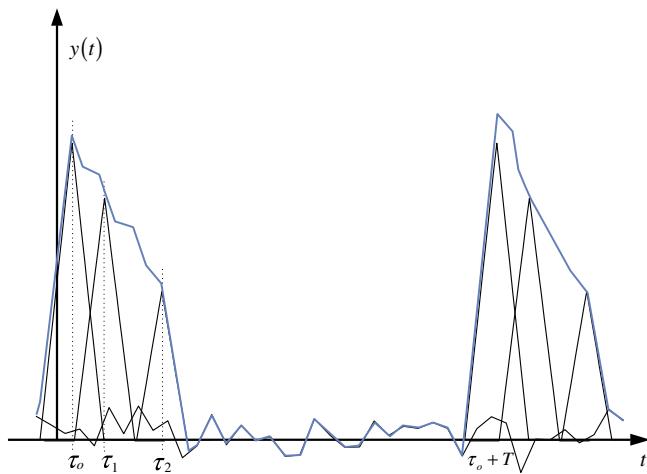
The choice of whether to base the computation of beamformers or other estimation methods on the pre-despread (pre-correlation) or post-despread (post-correlation) signal has a crucial impact on the performance and limitations of the array processing algorithms. GNSS signals typically have a Carrier-Power-to-Noise-Spectral-Density ( $C/N_0$ ) of about 45 dB Hz. The chip rate and hence the bandwidth is greater than 1 MHz, so this results in an SNR on the order of  $-15$  dB or less. This means that the GNSS signals and also their reflections are buried in the background noise. If one computes the spatial correlation matrix  $\mathbf{R}_{yy} = E\{\mathbf{y}(t)\mathbf{y}^H(t)\}$  in a pre-correlation scheme, only the noise and interference have a noticeable contribution to the matrix, so in practical terms the “total” correlation matrix  $\mathbf{R}_{yy}$  really only represents the noise-plus-interference correlation matrix.

The situation is completely different in the post-correlation scheme. The SNR of the correlation maximum is equal to  $C/N_0$  times the duration of the local reference. The duration of PN sequences in GNSS is several milliseconds, so the SNR of the maximum is typically on the order of several tens of dBs. The average SNR of the signal depends on the length of the portion of the correlation around the maximum that is taken as the observation window. This length is normally not too large, usually only a few chips, so the average SNR stays at the level of tens of dBs. In this case, the post-correlation matrix  $\mathbf{R}_{yy}$  includes noticeable contributions from the LOS and reflected signals besides the noise and



**FIGURE 20.22**

Qualitative example of the signal at one antenna after the despread (parameter  $T$  is the symbol period: 20 ms in GPS C/A, and  $T_c$  is the chip duration:  $\sim 1 \mu\text{s}$  in GPS C/A.)

**FIGURE 20.23**

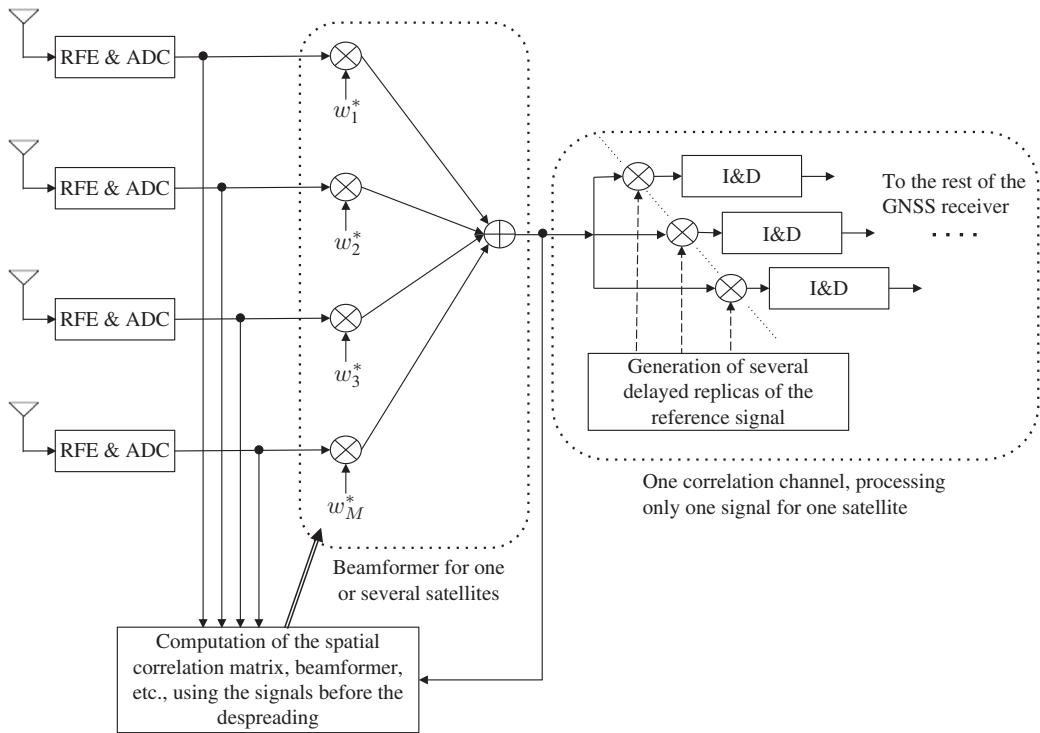
Qualitative example of the despreaded signal composed of the LOS component and two reflections. These reflections are considered as coherent multipath because their contributions overlap with that of the LOS component.

interference. To conclude, in order to make multipath *visible* in the spatial correlation matrix, one has to work with the post-despread correlation matrix. If one wants to hide multipath from the spatial correlation matrix, the pre-despread correlation matrix has to be used.

The location of the beamformer (if any) with respect to the despreader has an impact on computational complexity, but it does not have an effect on performance since only its position within a set of linear operations is changed. Note that we are referring here to the placement of the beamformer in the receive chain, and not to the input data used for its computation, which is a totally different aspect as explained above. Some examples of the placement of the beamformer as well as the input data used for its computation are shown in Figures 20.24 and 20.25. Note that all combinations are in principle possible, although some cases, such as pre-despread beamforming with weights computed using the post-despread signals, do not have a clear justification. As an example of a typical approach, the beamforming vector is computed using the pre-despread correlation matrix as  $\mathbf{w} = \mathbf{R}_{yy}^{-1} \mathbf{a}_0$ , and applied to the despread signal to obtain  $z(t) = \mathbf{w}^H \mathbf{y}(t)$ . In the first formula, the symbol  $\mathbf{y}$  refers to signals before despread, whereas in the second formula it refers to the despread signals.

### 3.20.4.3 Beamforming

The objective is to synthesize an array pattern that attenuates the reflections and interference. In the context of GNSS, antenna-array beamformers are customarily referred to as CRPAs (Controlled Reception Pattern Antennas). Adaptive (or data-dependent) beamforming is appropriate for situations where little *a priori* information about the scenario is available, or when the scenario is likely to change with time.



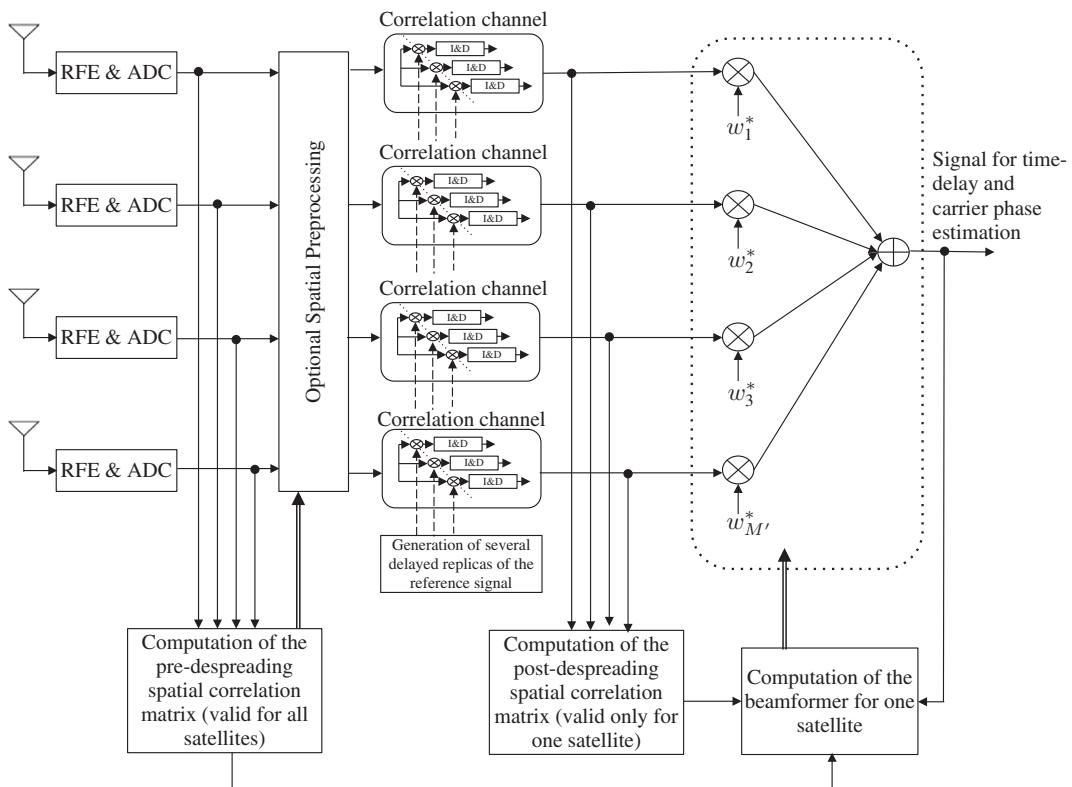
**FIGURE 20.24**

Example of a GNSS receiver using an antenna array where the beamformer is applied before despreading and it is computed using the pre-despread signals. The output of the beamformer is processed by a conventional GNSS receiver channel, as if it was the signal coming from a single antenna. Either option is possible: the beamformer can be the same for all satellites, or different beamformers for different satellites can be used. The complexity bottleneck is due to the fact that the beamformer weights are applied to high-rate samples coming from the RF front-end.

This is the typical situation for user receivers. On the other hand, deterministic (or data-independent) beamforming is more suitable for static and relatively controlled scenarios. This is typically the case for ground reference stations. These reference stations refer to both the receivers that form part of the ground segment of the GNSS (i.e., those receivers providing the measurements used to compute the position of the satellites) and the user receivers that are static and typically used as references in differential or relative positioning.

### 3.20.4.3.1 Adaptive beamforming

As outlined below, several different types of adaptive beamforming algorithms have been proposed for GNSS. Some of these are adaptations of standard algorithms, others have been designed specifically for conditions specific to positioning applications.

**FIGURE 20.25**

Example of a GNSS receiver using an antenna array where the beamformer is applied after despreading. The beamformer vector is calculated using the pre-despreading or the post-despreading spatial correlation matrix. An optional spatial preprocessing block is included, which can be used to cancel some spatial sectors. The number of outputs of the preprocessing block,  $M'$ , is equal to or smaller than the number of antennas,  $M$ . In this configuration, the application of the beamformer weights do not entail a significant computational load because the correlation channels generate samples at a very low rate. Hence the fact that a different beamformer is applied for each satellite is not a problem. Here the computational bottleneck comes from the need to use a correlation channel at each antenna or at each output of the preprocessing block.

*Algorithms employing a spatial reference:* These approaches are based on knowledge of the steering vector of the LOS signal,  $\mathbf{a}_0$ . Assuming this *a priori* information is reasonable in some GNSS applications since the satellite position can be known thanks to the navigation message (transmitted by the satellite itself) or to assistance from ground stations, and a rough estimate of the receiver position may be available from previous position fixes or from the application of a basic positioning algorithm (e.g., using only one antenna and not exploiting the antenna array). Moreover, the accuracy of the satellite and receiver positions is not important in determining the DOA of the signal; errors of several hundreds

of meters can be tolerated without affecting the satellite DOA estimate, given that the satellite-receiver distance is more than 20,000 km. However, the assumption of a known  $\mathbf{a}_0$  relies on the availability of array calibration and especially on the knowledge of the receiver orientation (also known as attitude in the GNSS literature). Errors or uncertainty in the array response correspond to the standard calibration problem found in many applications of antenna arrays, and robust methods developed for generic applications are also applicable here. On the other hand, the need to know the receiver orientation is a feature more specific to GNSS receivers. Assuming that  $\mathbf{a}_0$  is known, the use of the MVDR beamformer (and variants) is possible, and it is most appropriate to apply them in a pre-despread scheme. If these beamformers are computed with the post-despread correlation matrix and multipath components are present, they will suffer from the cancellation of the desired signal.

*Algorithms employing a temporal reference:* These methods are based on knowledge of the GNSS signal waveform. Knowledge of the waveform can be exploited to design a beamformer that minimizes the difference between its output and the reference signal (e.g., as in a Wiener filter). In practice, the situation is not that straightforward because even though the shape of the signal is known, the delay and frequency shift are not, so the beamformer weights and the signal parameters have to be computed jointly or iteratively. The expression for the beamformer is

$$\mathbf{w}_T = \mathbf{R}_{yy}^{-1} \mathbf{r}_{yx}(\hat{\tau}_0, \hat{f}_0), \quad (20.51)$$

where  $\mathbf{r}_{yx}(\hat{\tau}_0, \hat{f}_0)$  is the cross-correlation between the array output and a local replica of the LOS signal generated using estimates of its delay and frequency shift,  $\hat{\tau}_0$  and  $\hat{f}_0$ , respectively. In this case, it only makes sense to work with correlations computed after the despread, otherwise the contribution of the GNSS signals is hardly present in the correlations. This beamformer is able to cancel interference, but its performance in the presence of multipath is not satisfactory, although not as bad as with spatial-reference beamformers. The temporal reference beamformer combines the multipath and the LOS signal in a constructive manner, so as to increase the SNR. This is useful behavior in communications but not in navigation systems, since the increase in SNR comes at the price of a bias in the estimation of the delay and phase due to the presence of strong multipath at the beamformer output.

*Hybrid beamformers:* The opposite behavior of the spatial-reference and temporal-reference beamformers suggests that their combination may have good properties. Both of them provide the LOS signal at the output, but the former changes the phase of the multipath so that it is roughly in counter-phase with the LOS signal, whereas the latter modifies the multipath phase to align it with that of the LOS signal. Therefore, if the output of both beamformers is added together, the multipath will tend to cancel. This observation has led to the proposal of a hybrid beamformer that can be expressed as

$$\mathbf{w}_H = \beta \mathbf{w}_T + \gamma \mathbf{w}_S, \quad (20.52)$$

where  $\mathbf{w}_S$  is a spatial-reference beamformer, and  $\beta$  and  $\gamma$  are two scalars weighting the contribution of each beamformer. When  $\mathbf{w}_S$  is chosen as the MVDR beamformer, it can be shown that the optimal weights are

$$\begin{aligned} \beta &= \alpha_0, \\ \gamma &= 1 - \alpha_0 \mathbf{a}_0^H \mathbf{R}_{yy}^{-1} \mathbf{r}_{yx}(\tau_0, f_0). \end{aligned} \quad (20.53)$$

Since the optimal weights depend on the unknown parameters to be estimated, a practical way to proceed is to use an iterative algorithm where the calculation of the beamformer according to (20.52) is done using the previous estimates of  $\{\alpha_0, \tau_0, f_0\}$ , and next these estimates are updated using the output of the just computed beamformer.

*Blind algorithms:* This class of methods refers to techniques that do not exploit *a priori* knowledge of the exact signal or the steering vector, and hence are more robust to errors in these assumptions. Examples of such methods include those based on the constant modulus (CM) assumption, cyclostationarity and the power inversion approach. The civil GPS signal in current use, referred to as the C/A signal, has constant modulus because it is formed by almost rectangular chips. Most other GNSS signals also satisfy the CM property. However, this property cannot be exploited before despreading since the array cannot provide enough SNR gain to bring the signal above the noise. Therefore, the CM beamformer has to be applied after despreading, but in order to do so the despread samples corresponding to the LOS signal have to be CM. This happens when only one sample per integration period is used. However, the presence of multipath does not alter the constant-modulus property of the signal, so the CM beamformer is not useful in combating multipath.

GNSS signals are obviously cyclostationary since the repeated use of the PN spreading sequence introduces periodicity into the statistics of the signal. The fact that several repetitions of the PN code are present during a bit time (a property sometimes referred to as self-coherence) can also be exploited, as depicted in Figure 20.26. Interference will not have in general this structure, so it is possible to design the beamformer by imposing that its output should be as similar as possible to a version of itself delayed by a time equal to the PN code duration. Because of the same reasons as in the case of the CM beamformer, this technique should be applied to the despread signals and it will only be effective against interference and not multipath.

A very simple but rather effective approach is the power inversion beamformer. The weights are obtained as the beamformer vector that minimizes the total output power subject to a simple constraint to avoid the null solution. The constraint is chosen without using any information about the signal, typically forcing a given beamformer coefficient to be equal to one. This method has to be applied to the signals before despreading and, since the response is independent of the GNSS signals, it may happen that some nulls of the reception pattern are near to the DOAs of some of the GNSS signals. However, this situation can often be accepted since it is assumed that many GNSS satellites will be

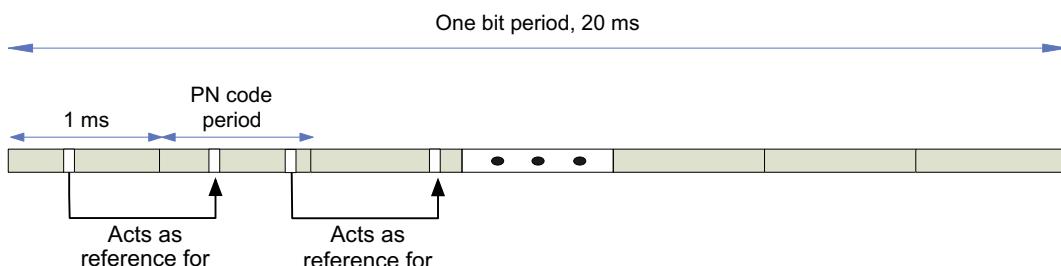


FIGURE 20.26

Structure of the GPS signal that allows to implement self-coherence restoration beamforming methods.

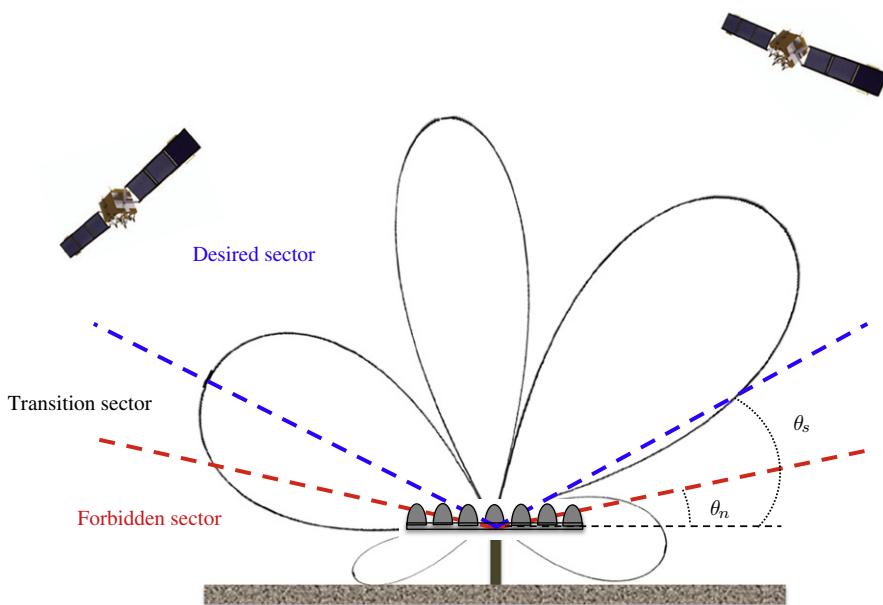
visible (around 10 satellites), and if a few of them are lost due to the coincidence of the pattern nulls with their DOAs, there will still be a sufficient number of satellite signals available (i.e., four or more) to compute the position. In this method, all satellites are received through the same beamformer, so it offers the possibility of being deployed as an add-onto existing single-antenna receivers. This is an important advantage of this method; more sophisticated beamformers that require information provided by the receiver (e.g., the DOA or the delay of the LOS signal) or that generate one beam per satellite cannot be coupled with existing single-antenna receivers and require the development of a completely new receiver. The number of antennas used with power inversion should be large enough to cancel the existing interference sources, but not much larger in order not to increase the number of nulls in the pattern and thus the probability that a GNSS signal is canceled. The power inversion approach is popular in military systems where jamming from highly maneuverable sources (fighter jets) is a pivotal concern. The fast maneuvers of these vehicles makes the use of spatial-reference beamformers virtually impossible, and therefore a simple and robust method like power inversion, that does not need any reference or calibration procedure, is an excellent option.

#### **3.20.4.3.2 Deterministic beamforming**

Although it is recognized that data-dependent beamformers are more powerful in general than deterministic versions, there are some situations where the latter may be advantageous. Deterministic beamformers are clearly more robust against calibration errors and other uncertainties in the signal parameters. Moreover, if the desired and non-desired signals are known to be confined to distinct spatial regions, the deterministic design may offer an adequate solution since the problem reduces to designing a spatial filter with given pass and stop bands. This *a priori* spatial separability occurs in several circumstances in GNSS, particularly in GNSS ground stations. In this case, the interference is normally ground-based, and the multipath normally arises from ground-based scatterers, so both interference and multipath impinge on the receiver from relatively low elevation angles. This is contrasted with the satellite signals, which originate from the entire upper hemisphere. Thus, as illustrated in Figure 20.27, a fixed beamformer can be designed to minimize reception of signals from these low elevations. The complicating factor here is that an upwards-facing array typically cannot provide a sharp stop-band to pass-band transition for directions near end-fire. Another advantage of deterministic beamformers is that they allow an easier control of the trade-off between array gain (understood here as the increase of the ratio between the desired signal power and the white noise power) and interference cancellation. In adaptive beamformers, these two characteristics are tightly coupled. For instance, with MVDR, the presence of a strong interference gives rise to a deep null in the pattern, and this null necessarily increases the beampattern in other directions, thus degrading the array gain.

#### **3.20.4.4 DOA estimation**

DOA estimation algorithms can be used as a processing stage prior to beamforming. If the DOAs of the LOS and reflected signals and interferences can be determined, then it is possible to design a beamformer that, for instance, attenuates the reflections and interferences while maximizing the SNR of the LOS signal. Given the identifiability limitations of DOA estimation methods and their sensitivity to highly correlated signals, such a method would likely only be suitable in situations where there were a small number of multipath and interference arrivals in addition to the LOS signal. The DOAs of the

**FIGURE 20.27**

Desired and forbidden regions for the design of deterministic beamformers.

non-desired signals can typically be obtained in two stages; interference sources can be localized prior to despreading, while the DOAs for multipath sources would have to be found after despreading. Even when it is not possible to use estimation methods to determine the DOAs of all signals, such methods can still be useful for detecting scenarios where the LOS signal is obstructed. Such information is critical in tracking applications, since highly erroneous estimates due to non-LOS measurements can be eliminated.

### 3.20.4.5 Array-based parameter estimators

Probably the most rigorous approach to the use of antenna arrays for multipath and interference mitigation consists not in focusing on the use of the array to synthesize a beam that attenuates those unwanted signals, but in formulating the measurement of the time delay and carrier phase of the LOS signal as an estimation problem. The Maximum Likelihood (ML) approach used in many other areas of array processing can also be used here. There is a large variety of models and assumptions that have been used to derive ML estimators, as we briefly outline below. Recall the expression of  $\mathbf{y}(t)$  in (20.50) and assume that  $K$  samples or snapshots are taken from the array, which form the columns of a matrix  $\mathbf{Y}$ . This matrix can be expressed as

$$\mathbf{Y} = \mathbf{A}\boldsymbol{\Gamma}(\boldsymbol{\tau}) \odot \mathbf{D}(\mathbf{f}) + \mathbf{N}, \quad (20.54)$$

where  $\mathbf{A} = [\mathbf{a}_0, \dots, \mathbf{a}_D]$ ,  $\boldsymbol{\Gamma} = \text{diag}\{\boldsymbol{\alpha}\} = \text{diag}\{\alpha_0, \dots, \alpha_D\}$ ,  $\boldsymbol{\tau} = [\tau_0, \dots, \tau_D]$ , and  $\mathbf{f} = [f_0, \dots, f_D]$ . The  $(k,n)$ th components of  $\mathbf{X}$  and  $\mathbf{D}$  are:  $[\mathbf{X}(\boldsymbol{\tau})]_{k,n} = x(t_n - \tau_k)$  and  $[\mathbf{D}(\mathbf{f})]_{k,n} = e^{j2\pi f_k t_n}$ . Matrix  $\mathbf{N}$  contains the snapshots of  $\mathbf{n}(t)$ , which includes all disturbances present in the received signal except for

the multipath components. In particular, as it can include directional interference, it is logical to assume that  $\mathbf{N}$  is spatially colored. The spatial correlation matrix is denoted as  $\mathbf{Q}$  and it is in general assumed to be unknown. If we further assume for simplicity that  $\mathbf{N}$  is temporally white, zero-mean and circularly-symmetric complex Gaussian distributed, the negative log-likelihood function can be expressed as

$$\begin{aligned} L(\mathbf{Y}, \mathbf{A}, \boldsymbol{\alpha}, \boldsymbol{\tau}, \mathbf{f}) = M \ln \det(\mathbf{Q}) \\ + \text{tr} \left( \mathbf{Q}^{-1} (\mathbf{Y} - \mathbf{A}\boldsymbol{\Gamma}(\mathbf{X}(\boldsymbol{\tau}) \odot \mathbf{D}(\mathbf{f}))) (\mathbf{Y} - \mathbf{A}\boldsymbol{\Gamma}(\mathbf{X}(\boldsymbol{\tau}) \odot \mathbf{D}(\mathbf{f})))^H \right), \end{aligned} \quad (20.55)$$

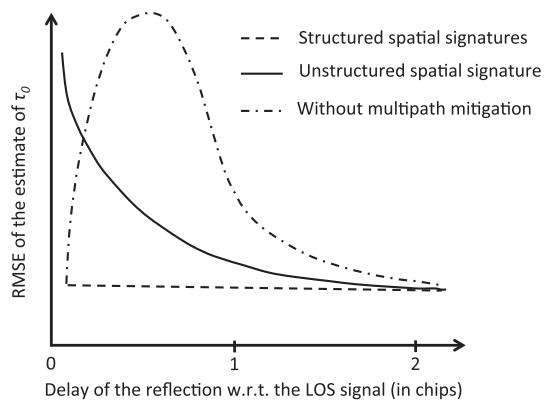
where  $\mathbf{A}$  can be replaced with the alternative parameterization discussed in Section 3.20.4.2. The ML estimates of the different parameters can be obtained as the arguments that minimize (20.55). This optimization problem, in its most general form, cannot be easily tackled because it has a large number of variables and it is highly non-linear (and non-convex). As discussed below, simplifications are possible depending on various modeling assumptions and how the problem is parameterized.

*Unstructured spatial signatures and spatially white noise:* The assumption of spatially white noise allows determinant in (20.55) to be eliminated, and the ML problem turns into a least squares problem. Since the resulting problem has the same structure as the estimation of the DOAs of unknown deterministic signals, most DOA estimations algorithms can be adapted to the estimation of time delays and frequencies in this new setup. A number of algorithms have been developed based on this parallelism between conventional DOA estimation and time delay estimation with unstructured spatial signatures. However, all techniques derived under the assumption of spatial whiteness suffer from a lack of interference mitigation capability. In addition, the use of unstructured spatial signatures causes the variance of the estimates to grow when the difference in delay and frequency shift of the replicas becomes smaller (see Figure 20.28).

*Structured spatial signatures and spatially white noise:* In this approach, the steering vectors  $\mathbf{a}_k$  are parameterized by the corresponding DOAs instead of being arbitrary complex vectors. This change makes the estimation problem more non-linear and hence more complex to solve, but on the other hand it provides in general more accurate estimates because the model parsimony is improved. The increased accuracy is largely observed when the signals are very close to each other in the delay and frequency dimension.

*Unstructured spatial signatures and unknown spatial correlation:* The ML estimator for this model involves the determinant of the correlation matrix of the fitting residuals, and hence it does not correspond to a least squares problem like the techniques derived under the assumptions of unstructured spatial signatures and spatial white noise. Consequently, it is not possible to establish a clear parallelism with DOA estimation algorithms, but techniques have been developed that are robust to directional interference. In addition, asymptotically equivalent algorithms have been proposed that admit a simple solution based on polynomial rooting.

*Structured spatial signatures and unknown spatial correlation:* This constitutes the most detailed model for the problem at hand, and also the one that leads to the best performance as long as there are no severe model mismatches with respect to reality. A direct optimization for this model requires a highly non-linear search in the DOA, time delay and frequency spaces, which cannot be implemented easily in an efficient manner. This limitation has been recently overcome by applying the Extended Invariance

**FIGURE 20.28**

Qualitative representation of the behavior achieved with the estimators derived under different models. It is assumed that the LOS signal and one reflection are received. The solid line corresponds to the model with unstructured spatial signatures. The dashed line corresponds to the model with structured spatial signatures; the line may be not totally constant, but in any case it shows a much smaller dependence with the delay than the solid line. When the reflection is not mitigated, the errors have the shape depicted by the dash-dotted line, where the increase in the RMSE is normally not due to an increase of the variance as in the other two cases, but to the existence of large bias.

**Principle (EXIP).** The EXIP technique begins with ML estimates corresponding to the model with unstructured spatial signatures and unknown spatial correlation described above, which can be obtained with relatively low complexity. Then, these estimates are refined by means of a weighted least-squares fit, resulting in improved estimates that have the same asymptotic accuracy as the exact ML estimates directly derived from the model with structured spatial signatures and unknown spatial correlation. The refinement approach boils down to a DOA estimation problem, and if the antenna array response has a Vandermonde structure, a polynomial-rooting based DOA estimator can be used. Thus, in the most difficult case involving DOAs and time delays of several replicas received in noise of unknown spatial correlation, estimates asymptotically equivalent to the ML ones can be obtained simply rooting two polynomials.

**GNSS-specific signal models:** Although the methods described above rigorously follow the logic of model-based estimation, they may present limitations in some practical conditions. This is exemplified by these two cases:

- The model in (20.54) assumes that the received signal is formed by several replicas of the transmitted GNSS signal. The real received signal may not be constituted by a few clearly defined reflections, but instead may consist of a large number or even a continuous distribution of components. In principle, the accurate modeling of this reality would require the use of a very large value for  $D$  in the model, and this would prohibitively increase the number of parameters to estimate and hence the complexity. One can argue that a reasonable model can be obtained using only a few replicas that capture most of the contribution of the actual multipath environment. But even if this is true, the estimation of the

appropriate value of  $D$  (large enough to represent well the received signal, but not too large to avoid overfitting of the model and excessive complexity) is an issue that needs to be addressed in any of the model-based estimators presented above.

- There are some particular aspects of the GNSS application that are not adequately exploited. For instance, the above methods provide estimates of all parameters in the model, but this is overkill in GNSS, where only the parameters of the LOS signal are of interest for positioning. Moreover, there is some side information that is not employed in the models, such as the *a priori* knowledge of the DOA of the LOS signal, and the fact that reflections always arrive later than the LOS signal and usually with smaller amplitudes.

As a consequence, a different way of proceeding consists in abandoning very detailed models attempting to provide a very precise description of the received signal (and maybe not achieving it because the signal includes other effects not accounted for in the model), in favor of simpler models that focus on particular aspects related to the GNSS application, even if they do not necessarily provide a comprehensive representation reality. For example, when LOS DOA can be assumed to be known, the vector  $\mathbf{a}_0$  can act as a spatial reference to the LOS signal, what makes it possible to approximately model the reflections as part of the noise term with unknown spatial correlation and, hence, the value of  $D$  is simply taken as zero. It can be shown that there is an equivalence between the estimator resulting from this simplified model and the hybrid beamformer presented in Section 3.20.4.3.1, while a model with  $D > 0$  would provide a more accurate representation of reality (at the expense of increased complexity), results have shown that there is typically not a large penalty in assuming  $D = 0$ . Problems will arise in situations when the delay of the reflections are close to that of the LOS signal, in which case the time delay and carrier phase estimates are biased. However, the degradation in such cases remains bounded.

### 3.20.5 Wireless communications

The use of antenna arrays in wireless communications provides one or more of the following types of advantages: diversity gain, array (or beamforming) gain and multiplexing gain. Given that in practice these gains lead to increases in capacity and spectral efficiency as well as improved robustness against fading, multiple antenna (or MIMO) techniques have been included in recent wireless communication standards. While the signal models and algorithms for multi-antenna wireless communications have been studied in detail in other chapters, here we focus on specific ways in which multiple antennas are exploited in current wireless standards.

#### 3.20.5.1 Multiple antennas techniques in LTE

MIMO constitutes an essential element of LTE in order to achieve the highly demanding requirements for transmission rate and spectral efficiency. LTE exploits multiple antennas for both diversity and multiplexing [78, 79, Ch.11], and also for both the downlink and uplink portions of the network.

##### 3.20.5.1.1 Diversity schemes

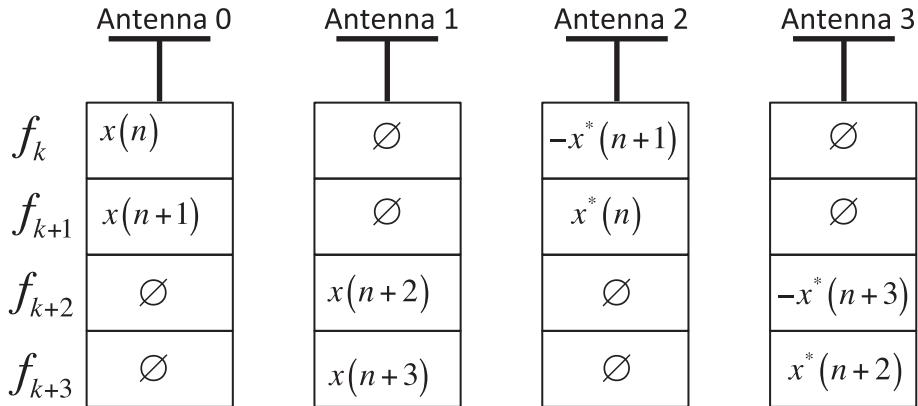
Various sources of diversity are available to average out channel variations due to fading. This includes time and frequency diversity, as well as transmit and receive diversity. Receive diversity is mandatory

for user handsets, usually referred to as UE's (User Equipment). It is the baseline receiver functionality for which performance requirements are defined. The typical method consists in performing maximum ratio combining (MRC) of the signals received at several antennas. We will focus however on transmit diversity since many schemes were analyzed in detail during the standardization phase of LTE. Some of the characteristics sought for the final selection of techniques were:

- Absence of puncturing in the presence of correlated channels. This eliminated the use of Cyclic Delay Diversity (CDD) and Precoding Vector Switching (PVS) in favor of block-code-based schemes.
- Low decoding complexity, which eliminated the option of non-orthogonal block codes.
- Power efficiency. Each antenna has an instantaneous power limitation, so the power that is not employed during one OFDM symbol (referred to as a “resource element” in LTE) cannot be shifted to the following ones. On the other hand, power can be adaptively allocated in the frequency domain; power that is not used in some subcarriers can be employed in others. The objective of using the maximum available power makes it advisable to select schemes where all antennas transmit at all times, though not necessarily in all subcarriers. This objective takes precedence over achieving a uniform power distribution in the frequency domain.
- Robustness to channel estimation errors. Orthogonal block codes lose the orthogonality property due to channel estimation errors at the receiver. As a certain level of error is unavoidable, it has to be checked that these errors do not cause large interfering terms. Estimation errors are not the only source of loss of orthogonality; variations of the channel that violate some design assumptions (such as the channel is constant across a certain group of subcarriers or during some symbols) may also create self-interference. It is desired to use techniques that do not make stringent assumptions about the evolution of the channel in time or frequency. Moreover, the quality of the channel estimation is not necessarily the same at all antennas. This means that all antennas are not statistically equivalent on average and a proper balancing of the symbols among them is needed.
- Good adaptation to the structure of the signals. The signals are mapped to two-dimensional resource blocks formed by a certain number of symbols and subcarriers. Some codes must be applied over a number of symbols or subcarriers that is a multiple of a given value (typically two or four). It may be easier to achieve the structure required by the code in one of the two dimensions. Since there are many more subcarriers than symbols forming a resource block, it is usually simpler to apply the code in the frequency domain because selecting a certain number of subcarriers is more manageable than changing the number of symbols in a block. Furthermore, in LTE the number of available OFDM symbols in a resource block is often odd.
- Reduced inter-cell interference. One must consider the impact of diversity techniques on the interference produced in neighboring cells.

For two-transmit-antenna diversity, the well-known Alamouti code is applied in the frequency domain, constituting a Space-Frequency Block Code (SFBC). If  $y^{(p)}(k)$  denotes the symbols transmitted from the  $p$ th antenna on the  $k$ th subcarrier, at a given OFDM symbol period, the transmission strategy of the eNodeB (i.e., the base station in LTE terminology) can be represented as follows:

$$\begin{bmatrix} y^{(0)}(k) & y^{(0)}(k+1) \\ y^{(1)}(k) & y^{(1)}(k+1) \end{bmatrix} = \begin{bmatrix} x(n) & x(n+1) \\ -x^*(n+1) & x^*(n) \end{bmatrix}, \quad (20.56)$$

**FIGURE 20.29**

Space-Frequency Block Code for four antennas used in LTE.

where  $x(n)$  represents the stream of symbols to be transmitted. In the case of four transmit antennas, the previous code is applied to each pair of antennas. Each pair of antennas uses a different set of frequencies, and hence the scheme is referred to as SFBC-FSTD, where FSTD stands for Frequency Shift Transmit Diversity (also known as Frequency Switched Transmit Diversity). This is depicted in Figure 20.29, and can be expressed as

$$\begin{bmatrix} y^{(0)}(k) & y^{(0)}(k+1) & y^{(0)}(k+2) & y^{(0)}(k+3) \\ y^{(1)}(k) & y^{(1)}(k+1) & y^{(1)}(k+2) & y^{(1)}(k+3) \\ y^{(2)}(k) & y^{(2)}(k+1) & y^{(2)}(k+2) & y^{(2)}(k+3) \\ y^{(3)}(k) & y^{(3)}(k+1) & y^{(3)}(k+2) & y^{(3)}(k+3) \end{bmatrix} = \begin{bmatrix} x(n) & x(n+1) & 0 & 0 \\ 0 & 0 & x(n+2) & x(n+3) \\ -x^*(n+1) & x^*(n) & 0 & 0 \\ 0 & 0 & -x^*(n+3) & x^*(n+2) \end{bmatrix}. \quad (20.57)$$

This mapping is a full-rate orthogonal code with diversity order equal to two, which is smaller than the possible maximum of four since full-rate full-diversity orthogonal codes do not exist for four antennas and complex symbols. Note also that each pair of symbols uses antennas {0, 2} and {1, 3}. This is because the channel estimates are better in antennas 0 and 1 since more pilot symbols are employed for these antennas than for antennas 2 and 3. Thus, each pair of symbols makes use of one of the antennas for which the receiver can obtain better channel estimates and another antenna for which the estimates are worse.

### 3.20.5.1.2 Multiplexing schemes

LTE supports closed-loop and open-loop MIMO transmission in the downlink using  $P = 2$  or  $4$  antennas and a number of multiplexing layers equal to  $v = 1, 2, 3$ , or  $4$ . A layer is a term used in LTE to denote the

different data streams to be transmitted simultaneously using spatial multiplexing. As a consequence, the number of layers represents the multiplexing gain and cannot exceed the number of transmit antennas; thus,  $v \leq P$ . The number of layers is also referred to as the rank of the transmission. The mapping of between codewords (i.e., an independently encoded data block) and layers is also specified in LTE. For a transmission rank greater than 1, up to two codewords can be transmitted. In this case, each codeword is assigned to each layer if  $v = 2$ , one codeword is assigned to one layer and the other codeword is split between the other two layers if  $v = 3$ , and each codeword is mapped to a different pair of layers if  $v = 4$ . Multi-codeword transmission allows for the use of the computationally simpler MMSE-SIC (Minimum Mean Square Error-Successive Interference Cancellation) detector, providing comparable performance to the more complex ML detector applied to the single-codeword case, which on the other hand enjoys an advantage in terms of ARQ ACK/NACK signaling.

The relation between the symbols at the antenna ports,  $y^{(p)}(n)$ , and the symbols in layer  $l$ ,  $x^{(l)}(n)$ , is

$$\begin{bmatrix} y^{(0)}(n) \\ \vdots \\ y^{(P-1)}(n) \end{bmatrix} = \mathbf{W}(n) \begin{bmatrix} x^{(0)}(n) \\ \vdots \\ x^{(v-1)}(n) \end{bmatrix}, \quad P = 1, 2, \text{ or } 4, \quad P \geq v = 1, 2, 3, \text{ or } 4, \quad (20.58)$$

where  $\mathbf{W}(n)$  is a  $P \times v$  precoding matrix. Next we describe the closed- and open-loop approaches to forming this matrix.

*Closed-loop multiplexing schemes:* The precoding matrices belong to a codebook. The receiver selects the best precoding matrix based on its current channel estimates and feeds back an index to the transmitter. For the case of rank-1 transmission with 2 antennas, the precoders are

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ j \end{bmatrix}, \quad \begin{bmatrix} 1 \\ -j \end{bmatrix}. \quad (20.59)$$

The elements of the precoders are limited to the QPSK alphabet  $\{\pm 1, \pm j\}$  to reduce computational complexity at the UE by avoiding the use of complex multiplications. Moreover, there are no amplitude differences between antennas because it is desired to use the maximum available power at each antenna. These two properties are also valid for the other cases, with the caveat that with four antennas the elements of the matrices can also belong to the 8-PSK alphabet:  $\{\pm 1, \pm j, (\pm 1 \pm j)/\sqrt{2}\}$ .

For the case of rank-2 transmission with 2 antennas, the precoders are

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ 1 & -j \end{bmatrix}. \quad (20.60)$$

The codebook for four antenna ports is formed by 16 matrices, which are obtained from 16 generating vectors,  $\mathbf{v}_k$ , whose components belong to the 8-PSK alphabet by applying the Householder matrix definition:  $\mathbf{I} - 2\mathbf{v}_k\mathbf{v}_k^H$ . The precoders for ranks lower than four are obtained by a selected subset of the columns of each matrix. This makes it straightforward to fulfill the nested property, whereby columns of lower rank precoders are subsets of the columns of higher rank precoders, which considerably facilitates the precoder evaluation at the UE. LTE admits both frequency-selective precoding, in which precoding

weights are selected independently for different sub-bands of bandwidth ranging from 360 kHz to 1.44 MHz, and also wideband precoding, where a single set of single precoding weights are applied to the entire transmission band.

Note that rank-1 transmission amounts to beamforming. Besides the beamforming case, LTE also allows for UE-specific beamforming, which is not based on the feedback of precoding-related information, but on channel state information obtained by the eNodeB using for instance DOAs measured from the uplink signals or exploiting reciprocity in TDD scenarios.

*Open-loop multiplexing schemes:* The same diversity schemes as described in Section 3.20.5.1.1 are used for rank-1 open-loop communication. For higher ranks, the general approach is to employ layer cycling together with precoder cycling. Layer cycling is implemented by means of CDD (Cyclic Delay Diversity), and the net effect is to circularly change the order of the columns of the precoding matrix. Specifically, this type of CDD is called long-delay CDD in LTE terminology. This means each layer is transmitted using a different column of the precoding matrix at successive OFDM symbols. The precoder cycling consists simply of changing the precoding matrix after each set of  $v$  resource elements, that is to say, when a complete circular shift of the current matrix has been done. The logic behind this approach is that precoder cycling provides a new realization of SINRs across the layers every time the precoding matrix is changed, and layer cycling makes each codeword experience an SINR that is the average of the SINRs of the layers because each codeword ends up using all columns of the precoding matrix.

In order to put the description above in formulas, we can consider for example the case of  $P = 4$  antennas and  $v = 3$  layers. The description is also valid for two or four layers with obvious modifications. The same relation as in (20.58) is valid with the replacement of  $\mathbf{W}(n)$  with

$$\mathbf{W} \left( \left\lfloor \frac{n}{v} \right\rfloor \bmod 4 \right) \mathbf{D}(n) \mathbf{U}, \quad (20.61)$$

where  $\mathbf{W}(n)$  represents one of the 16 matrices previously mentioned, matrix  $\mathbf{D}(n)$  applies CDD in the frequency domain and  $\mathbf{U}$  is the  $v \times v$  DFT matrix:

$$\mathbf{U} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & e^{-j2\pi/3} & e^{-j4\pi/3} \\ 1 & e^{-j4\pi/3} & e^{-j8\pi/3} \end{bmatrix}. \quad (20.62)$$

The cyclic delay applied to the  $l$ th layer is equal to a fraction  $l/v$  of the symbol duration, and then the CDD matrix is

$$\mathbf{D}(n) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & e^{-j2\pi n/3} & 0 \\ 0 & 0 & e^{-j4\pi n/3} \end{bmatrix}. \quad (20.63)$$

The set of possible precoding matrices contains four elements (for any number of layers), and the index  $(\lfloor \frac{n}{v} \rfloor \bmod 4)$  selects another matrix every time a given matrix has been used for  $v$  symbols. The important

fact is that the combined effect of the CDD and the DFT matrices is

$$\begin{aligned}\mathbf{D}(3m)\mathbf{U} &= \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & e^{-j2\pi/3} & e^{-j4\pi/3} \\ 1 & e^{-j4\pi/3} & e^{-j8\pi/3} \end{bmatrix}, \\ \mathbf{D}(3m+1)\mathbf{U} &= \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ e^{-j2\pi/3} & e^{-j4\pi/3} & 1 \\ e^{-j4\pi/3} & e^{-j8\pi/3} & 1 \end{bmatrix}, \\ \mathbf{D}(3m+2)\mathbf{U} &= \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ e^{-j4\pi/3} & 1 & e^{-j2\pi/3} \\ e^{-j8\pi/3} & 1 & e^{-j4\pi/3} \end{bmatrix},\end{aligned}\quad (20.64)$$

which means that columns of the resulting matrix are shifted for successive symbols.

The case for two antennas is simpler because the precoding matrix  $\mathbf{W}(n)$  is always the identity matrix (so no precoder cycling is applied) and

$$\mathbf{D}(n) = \begin{bmatrix} 1 & 0 \\ 0 & (-1)^n \end{bmatrix}. \quad (20.65)$$

This implies that

$$\begin{aligned}\mathbf{D}(2m)\mathbf{U} &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \\ \mathbf{D}(2m+1)\mathbf{U} &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix},\end{aligned}\quad (20.66)$$

which simply represents a swap of the columns between the two layers for consecutive symbols.

### 3.20.5.1.3 Multiple user MIMO (MU-MIMO)

The previous description is based on Release 8 of the LTE standard and has considered only Single-User MIMO (SU-MIMO). That release includes a rather minimal MU-MIMO transmission scheme. It is based on codebook feedback and uses the same codebooks as SU-MIMO. Actually, only the rank-1 precoders are employed because only one layer is utilized by each UE. The performance of this MU-MIMO scheme is limited by the coarse codebook quantization and the lack of support for cross-talk suppression at the UE. As a consequence, MU-MIMO only offers marginal gain with respect to SU-MIMO. The shortcomings of this simple MU-MIMO approach are fixed in the subsequent releases [80–82]. The set of new features included in Release 10 of the standard has made it possible to reach spectral efficiencies of 30 bits/s/Hz in the downlink and 15 bits/s/Hz in the uplink [83, Section 7.3].

Release 9 allows for beamforming for up to four UEs. The beamformers are constructed by exploiting channel reciprocity. It also includes the option of rank-2 transmissions to two UEs. Release 10 (also known as LTE-A or LTE-Advanced) supports configurations with up to  $8 \times 8$  MIMO with eight transmission layers, and as a consequence the set of precoding codebooks has also been extended using the dual-codebook approach. That is, the precoding matrix is obtained as the multiplication of two matrices,  $\mathbf{W}_1$  and  $\mathbf{W}_2$ , where  $\mathbf{W}_1$  is a block diagonal matrix matching the spatial covariance matrix of

the dual-polarized antenna setup, and  $\mathbf{W}_2$  is the antenna selection and cophasing matrix. The LTE-A UEs have to provide feedback information for both  $\mathbf{W}_1$  and  $\mathbf{W}_2$ . When only two or four antennas are used at the eNodeB,  $\mathbf{W}_1$  is the identity matrix and backwards compatibility with Releases 8 and 9 is achieved. For the 8-transmit antenna configuration,  $\mathbf{W}_1$  is obtained from the coefficients of the DFT.

An important contribution in LTE-A is the inclusion of Coordinated Multipoint transmission (CoMP), whereby multiple eNodeBs can cooperate to determine the scheduling, transmission parameters, and transmit antenna weights for a specific UE [84, 85]. The objective is to reduce interference at the UEs, making universal frequency reuse possible and, hence improving cell-edge throughput as well as average sector throughput with little complexity increase at the receiver. Two major types of CoMP transmission are identified for the downlink (DL) of LTE-A:

- *Coordinated beamforming/coordinated scheduling (CB/CS)* refers to techniques that do not require data sharing between cells. However, CSI may be shared among cells. This family of techniques includes coordinated beamforming/scheduling, adaptive fractional frequency reuse, interference alignment, PMI (Precoding Matrix Indicators) coordinations, etc.
- *Joint processing* is characterized by the fact that data are shared, and it includes techniques such as dynamic cell selection and joint transmission (network MIMO).

The CoMP concept can also be employed in the uplink by coordinating multiple cells to perform joint reception of the transmitted signal at multiple receiving eNodeBs and/or by taking coordinated scheduling decisions. Nevertheless, CoMP transmission/reception is an active area of research and further studies are needed to reliably evaluate the gains of CoMP in LTE.

#### **3.20.5.1.4 Uplink MIMO**

In Release 8, only one antenna of the UE can be used for transmission, so it is possible to achieve transmit diversity using an antenna selection mechanism, but single-user spatial multiplexing is not feasible. However, the uplink (UL) can support MU-MIMO transparently with 2–6 UEs (although in practice only two UEs are considered in order to limit receiver complexity). The number of UEs that can share a resource block is determined by the number of orthogonal reference signals that can be assigned to the UEs. The different reference signals are used by the eNodeB to estimate the channels of each UE, from which a multiuser detector (e.g., using the MMSE criterion) is derived. In Release 10, spatial multiplexing with 1, 2, or 4 transmit antennas at the UE and up to four layers is introduced. Open-loop and closed-loop spatial multiplexing as well as transmit diversity are supported. Closed-loop multiplexing relies on codebook-based precoding, and the codebooks are optimized to maintain a low PAPR.

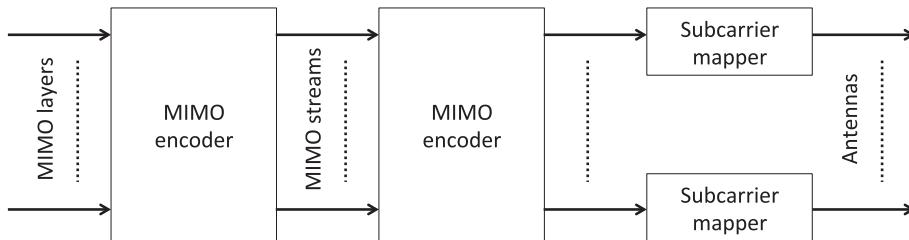
To sum up, the evolution of MIMO techniques in the different releases of the standard is summarized in Table 20.1.

#### **3.20.5.2 Multiple antennas techniques in WiMAX**

The IEEE 802.16m standard is the core technology for WiMAX Release 2 (WiMAX-2 in short), and it contains the addition of several MIMO technologies to the ones included in IEEE 802.16e (which was the basis of WiMAX Release 1) [86]. MIMO plays an essential role in WiMAX-2, as well as in LTE-A, in order to meet the IMT-Advanced 4G requirements. Although terminologies in the IEEE 802.16 and 3GPP LTE standards differ and the comparison may be confusing, the MIMO techniques used in both WiMAX-2 and LTE-A, while different in various details, share in general the same fundamental approaches.

**Table 20.1** Evolution of the Support of MIMO Techniques in LTE

	<b>LTE (Rel-8)</b>	<b>LTE (Rel-9)</b>	<b>LTE-A (Rel-10)</b>
Downlink	<ul style="list-style-type: none"> <li>• Codebook-based SU- &amp; MU-MIMO</li> <li>• Transmit diversity</li> <li>• Dedicated reference signal-based beamforming</li> </ul>	<ul style="list-style-type: none"> <li>• Dual-stream beamforming</li> </ul>	<ul style="list-style-type: none"> <li>• Non-codebook-based precoding for eight layers</li> <li>• Enhanced MU-MIMO</li> <li>• Inclusion of CoMP</li> </ul>
Uplink	<ul style="list-style-type: none"> <li>• MU-MIMO</li> <li>• Antenna selection</li> </ul>		<ul style="list-style-type: none"> <li>• Spatial multiplexing with codebook-based precoding</li> <li>• Transmit diversity</li> </ul>

**FIGURE 20.30**

WiMAX DL MIMO architecture (as shown in [89]).

Therefore, rather than describing the details of MIMO techniques in WiMAX-2, for which an excellent review can be found in [87, Ch.10], we will focus on the similarities between LTE-A and WiMAX (a compared overview can be found in [88]), and comment on some specifics aspects of the latter.

Both 802.16m and LTE-A support MIMO implementations with the same sets of antennas: 2, 4, or 8 transmit antennas and a minimum of 2 receive antennas in the DL; 1, 2, or 4 transmit antennas and a minimum of 2 receive antennas in the UL. The two systems also specify schemes for: open-loop transmit diversity, open- and closed-loop spatial multiplexing, and MU-MIMO both in the UL and DL. The WiMAX downlink architecture is represented in Figure 20.30. For open-loop transmit diversity, WiMAX employs SFBC encoding combined with precoder cycling, whereas LTE employs either SFBC or SFBC-FSTD. As far as open-loop spatial multiplexing is concerned, both systems propose precoder cycling, but WiMAX does not include layer permutation with CDD. Closed-loop spatial multiplexing relies on codebook-based precoding, and WiMAX has three feedback mechanisms: base mode, transformation mode, and differential mode.

Codebook adaptation is defined in 802.16m, and it consists in changing the codeword distribution according to long-term channel statistics. Each vector codeword of the rank-1 base codebook is linearly

transformed and normalized to create a codeword in the new codebook. As a result, more codewords are steered towards the ideal beamformer vectors and the codebook quantization error is reduced. Moreover, codebook adaptation is also useful in achieving robustness against calibration errors in the antenna array and transceiver chains. The derivation of the adaptive precoding matrix is specific to the implementation and is not included in the standard. The case where the columns of the precoding matrix are orthogonal to each other is called unitary precoding. Otherwise, it is defined as non-unitary precoding. Non-unitary precoding is only allowed with closed-loop MU-MIMO. Advanced beamforming is also enabled by this precoding mechanism. Besides the closed-loop MU-MIMO scheme, which is also present in LTE, WiMAX-2 allows for open-loop MU-MIMO, where each terminal selects the preferred column from a unitary matrix that has been preset for each frequency-domain resource. Each terminal reports the channel quality indicator (not the spatial correlation matrix, which is the reason why the scheme is considered to be open-loop), and the technique shows good performance with limited feedback in uncorrelated and semi-correlated channels typically corresponding to urban areas with high user density and no line-of-sight. A summary of the MIMO modes proposed for the downlink and uplink of WiMAX-2 are summarized in Tables 20.2 and 20.3.

### 3.20.5.3 Multiple Antenna Techniques in IEEE 802.11

MIMO techniques play an essential role in the significant increase (54–600 Mbits) in the maximum data rate provided by the IEEE 802.11n amendment to the IEEE 802.11-2007 standard. The single largest contributor to the rate increment comes from the use of multiple antennas, which has the effect of a fourfold increase. A factor of 2 can be attributed to the widening of the channels from 20 MHz to

**Table 20.2** Downlink MIMO Modes

Mode index	Description	MIMO Encoding Format	MIMO Precoding
0	Open-loop single-user (transmit diversity)	Alamouti encoding in space-frequency	Non-adaptive
1	Open-loop single-user (spatial multiplexing)	Transparent encoding	Non-adaptive
2	Closed-loop single-user (spatial multiplexing)	Transparent encoding	Adaptive
3	Open-loop multiple-user (spatial multiplexing)	Multi-layer encoding	Non-adaptive
4	Closed-loop multiple-user (spatial multiplexing)	Multi-layer encoding	Adaptive
5	Open-loop single-user (transmit diversity)	Conjugate data repetition	Non-adaptive

**Table 20.3** Uplink MIMO Modes

Mode index	Description	MIMO Encoding Format	MIMO Precoding
0	Open-loop single-user (transmit diversity)	Alamouti encoding in space-frequency	Non-adaptive
1	Open-loop single-user (spatial multiplexing)	Transparent encoding	Non-adaptive
2	Closed-loop single-user (spatial multiplexing)	Transparent encoding	Adaptive
3	Open-loop multiple-user (collaborative spatial multiplexing)	Transparent encoding	Non-adaptive
4	Closed-loop multiple-user (collaborative spatial multiplexing)	Transparent encoding	Adaptive

40 MHz; and the rest of the improvement (roughly about 40%) to reducing the overhead in the signal [90]. IEEE 802.11a/g allowed only for a very basic exploitation of multiple antennas. Its method for obtaining diversity was simple antenna selection, while IEEE 802.11n allows for the use of Space-Time Block Codes (STBC), spatial multiplexing and transmit beamforming. Any number of transmit and receive antennas with a maximum number of 4 at each side is permitted, and up to four data streams can be multiplexed.

Two processing blocks are sequentially applied to the spatial streams to obtain the data streams to be transmitted from each antenna [91,92]:

- *STBC encoder:* Spreads constellation points from  $N_{SS}$  spatial streams into  $N_{STS}$  space-time streams using a space-time block code. The STBC encoder is used only when  $N_{SS} < N_{STS}$ , otherwise it is a transparent block. If  $N_{SS} = 1$  and  $N_{STS} = 2$ , the Alamouti code is employed; if  $N_{SS} = 2$  and  $N_{STS} = 3$ , one spatial stream is encoded by the Alamouti approach and the other stream is directly mapped to the third space-time stream; if  $N_{SS} = 2$  and  $N_{STS} = 4$ , two disjoint pairs of space-time streams are obtained by applying the Alamouti code to each spatial stream; and finally if  $N_{SS} = 3$  and  $N_{STS} = 4$ , one spatial stream is coded with the Alamouti code and the other two streams are directly mapped to the output. The cases for a single spatial stream  $N_{SS} = 1$  with three or four antennas are handled through the use of spatial expansion, which is mentioned below.
- *Spatial mapper:* Maps the  $N_{STS}$  space-time streams to the  $N_{TX}$  antennas (where  $N_{TX} \geq N_{STS}$ ) by multiplying the space-time streams by a matrix, which is then passed along to each transmit chain. Different matrices can be used for different subcarriers. Some examples of spatial mapping are presented below, but other alternatives are possible and the standard does not restricts the implementation to these instances.
  - *Direct mapping:* Each space-time stream is directly assigned to each antenna (only possible when  $N_{TX} = N_{STS}$ ), possibly after multiplication by a complex exponential in order to implement CDD.

- *Indirect mapping*: The two sets of streams are related by a square unitary matrix such as the Hadamard matrix or the Fourier matrix.
- *Spatial expansion*: The standard proposes several binary-valued (ones and zeros) matrices covering the different combinations of the values of  $N_{TX}$  and  $N_{STS}$ . The effect of these matrices is simply to translate each of the  $N_{STS}$  streams to one or several antennas. For instance, if  $N_{STS} = 1$  and  $N_{TX} = 3$ , the following matrix (vector, in this case) is proposed:  $\mathbf{D} = 1/\sqrt{3}[1 \ 1 \ 1]^T$ , which implies that the same symbols are transmitted simultaneously from the three antennas.
- *Beamforming matrix*: Represents any matrix that improves the reception based on some knowledge of the channel between the transmitter and the receiver. Two mechanisms are considered in the standard to obtain CSI at the transmit side. The first is called *implicit feedback*, which relies on reciprocity in the TDD operation mode to estimate the channel based on a reference signal transmitted by the device that will act as receiver in the subsequent communication. In the second mechanism, denoted as *explicit feedback*, the receiver sends to the transmitter either the measured channel response or a beamforming matrix that it has computed based on the measured channel. In the latter case, there are two possibilities, namely, to simply transmit the coefficients of the beamforming matrix (called noncompressed beamforming feedback matrix) or a set of angles and phases that parameterize that matrix (called compressed beamforming feedback matrix).

As a final remark, it is worth mentioning that a cyclic shift can also be applied to the signal in each antenna to prevent unintentional beamforming. The shift can be inserted either in the frequency or in the time domain (i.e., before or after the IDFT).

---

### 3.20.6 Biomedical

There is an immense interest in processing the electrical, magnetic and acoustic signals that originate from physiological processes, and extracting information that is useful for diagnosis and treatment. Examples of such signals include those obtained via electrocardiography, measurements of the electrical behavior of the heart; electroencephalography and magnetoencephalography, which measure the electrical and magnetic activity of the brain; electromyography, observations of electrical signals in muscle tissue, and so on. In addition, active measurement approaches exist that collect the response of the body to magnetic or acoustic stimulation, as in magnetic resonance or ultrasonic imaging systems. Arrays of sensors are used in many of these applications, primarily for localizing the source of the signals in passive measurement systems, or for non-invasively imaging the internal structure of the body in active systems.

In this section, we will briefly discuss three biomedical applications of array signal processing that are widely employed in both clinical or research settings. These are by no means exhaustive; a notable omission is magnetic resonance imaging (MRI), which uses a large array of coils to detect the precession of molecules in response to applied external magnetic fields. However, these examples serve to illustrate the important role array signal processing has in biomedicine.

### 3.20.6.1 Ultrasonic imaging

Ultrasonic arrays for biomedical imaging are in widespread clinical use today, most commonly for monitoring fetal development and for real-time imaging of heart valve operation and related blood flow. Ultrasound imaging is relatively inexpensive compared with other imaging modalities, and the array and associated equipment is relatively compact and portable. Ultrasonic images can achieve sub-millimeter resolution, but the imaging process is more susceptible to noise and unpredictable propagation effects than, say, MRI.

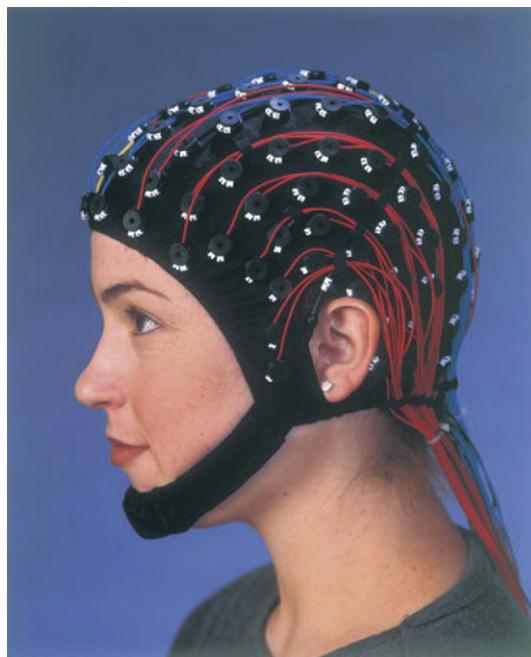
Ultrasound imaging is based on pulse-echo signal processing, much like an active radar. An array of from many tens to a few hundred piezoelectric transducers transmits baseband signals with bandwidths up to tens of MHz and then receives the resulting echoes. Broadband signals are usually employed for imaging human tissue, while narrowband CW signals are employed for Doppler measurements of blood flow velocities. An ultrasonic array is relatively compact, with an aperture of 50 mm or less, and can be condensed to fit in a handheld wand that is manually placed on the body and oriented in some direction of interest. The speed of sound in human tissue is approximately 1500 m/s, so the resulting wavelength is typically much less than 1 mm. Consequently, near-field modeling of the acoustic wavefronts is often necessary. The array typically has a slight inward curve to create a larger “fan-beam” image.

Traditionally, ultrasonic imagers have employed delay-and-sum beamforming to focus both the transmit and receive signals, although with improvements in computational power, systems are now being designed with adaptive (e.g., MVDR) beamforming to improve resolution and eliminate artifacts due to interference entering through sidelobes. An important difference compared to radar is the severe range-dependent attenuation the ultrasonic signal undergoes—with signal intensity decreasing by a factor of two at 5 MHz for approximately every cm of distance the signal travels. This necessitates the use of gain compensation on receive and leads to low SNRs at longer ranges. As focus moves towards higher frequencies for better resolution, the attenuation problem increases.

### 3.20.6.2 EEG and MEG signal processing

Electroencephalography (EEG) and magnetoencephalography (MEG) are widely used in both clinical practice and research since they provide direct measurement of cerebral activity with much higher temporal resolution than other non-invasive methods such as functional MRI (fMRI). The analysis of EEG/MEG signals is used for detecting and diagnosing neurological disorders such as epileptic seizures, monitoring brain activity during sleep or anesthesia, analyzing the extent of brain damage due to stroke or traumatic injury, etc. Such signals are also currently being investigated as a tool for brain-computer interface applications that would allow individuals with sensory-motor impairments (e.g., a paraplegic) to control a wheelchair, prosthetic limb or a computer input device via focused cognitive activity. Although EEG/MEG techniques have lower spatial resolution than, for example, fMRI, relatively high-resolution techniques for locating sources of cerebral activity have been proposed to cope with this issue.

To obtain high spatial precision, EEG/MEG localization requires a large array of sensors or electrodes (an example of a typical EEG array is shown in Figure 20.31), which leads to a high-dimensional inverse problem that in general does not have a unique solution. Thus, in practice, a “forward” propagation model for the brain, skull and scalp is adopted, and one attempts to estimate the parameters of the model corresponding to the source activity. A common approach is to model the signal source in a

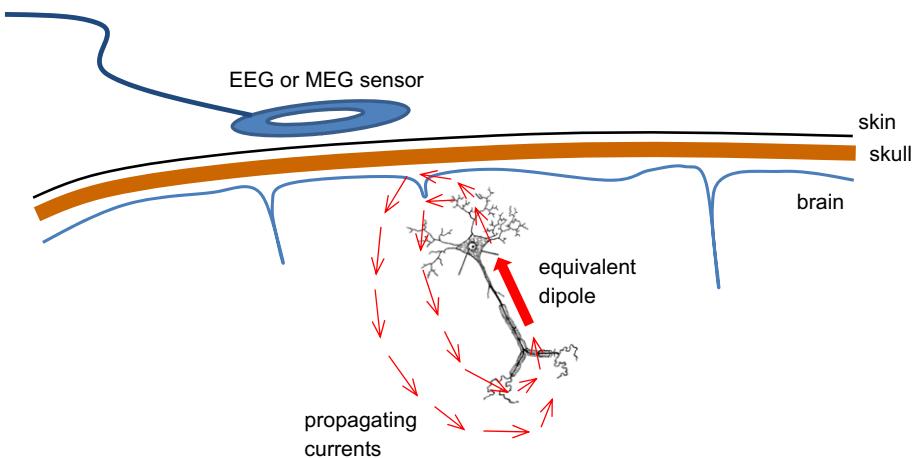
**FIGURE 20.31**

Cap with electrodes for collecting EEG data.

small region of the brain as originating from an equivalent current dipole, treating the dipole location, the orientation and magnitude of the dipole moment as dipole parameters to be estimated. Figure 20.32 depicts the equivalent dipole model, where the charge difference along a neuron (or neuron cluster) causes a flow of current whose resulting electric or magnetic field can be measured by a sensor. While EEG caps like the one in Figure 20.31 are used to place the sensor as close to the brain as possible, there is increasing interest in and use of intracranial EEG measurements, or electrocorticography (ECoG), where the electrode is placed beneath the skull immediately adjacent to the neural area of interest. ECoG signals avoid the attenuation of the skull and scalp and provide a much higher SNR, but at the expense of an invasive implantation. In the next section, we describe the mathematical model that results from the equivalent dipole assumption to illustrate its connection with other array signal processing applications.

### **3.20.6.2.1 Unified dipole model for EEG/MEG/ECoG measurements**

Assume a current dipole located at position  $\mathbf{r} \in \mathbb{R}^{3 \times 1}$  and an electric or magnetic sensor located at  $\mathbf{s} \in \mathbb{R}^{3 \times 1}$ . In the following, we derive expressions for the electric or magnetic field  $\mathbf{y}(t)$  at  $\mathbf{s}$  due to the dipole at  $\mathbf{r}$  for the three measurement modalities EEG, MEG, and ECoG. We will see that all three lead to expressions with a similar structure that allows us to formulate a unifying model for the output of arrays of such sensors.

**FIGURE 20.32**

Equivalent current dipole model for EEG measurements.

**EEG:** The electric potential at  $\mathbf{s}$  caused by a current density  $\mathbf{J}_S$  at location  $\mathbf{r}$  under a quasistatic assumption (i.e., setting all time derivatives in Maxwell's equations equal to zero) can be expressed as

$$y(t) = -\frac{1}{4\pi\sigma} \int_V \frac{\nabla \mathbf{J}_S(\mathbf{r}, t)}{\|\mathbf{s} - \mathbf{r}\|} d\mathbf{r}, \quad (20.67)$$

where  $\nabla$  is the divergence operator and  $V$  is the volume of interest. A current dipole can be idealized as a source and sink with equal magnitude, denoted by  $I_0(t)$ , and separated by a very small distance  $d$ , which leads to

$$\nabla \mathbf{J}_S = -I_0(t)[\delta(\mathbf{s} - \mathbf{r}_+) - \delta(\mathbf{s} - \mathbf{r}_-)], \quad (20.68)$$

where  $\delta$  is the Dirac delta function, and  $\mathbf{r}_+$  ( $\mathbf{r}_-$ ) is the source (sink) location. The dipole location  $\mathbf{r}$  is assumed to be at the midpoint between  $\mathbf{r}_+$  and  $\mathbf{r}_-$ , and the orientation of the dipole is in the direction of  $\mathbf{d} = \mathbf{r}_+ - \mathbf{r}_- = \frac{1}{2}(\mathbf{r}_+ - \mathbf{r}) = \frac{1}{2}(\mathbf{r} - \mathbf{r}_-)$ . Substituting (20.68) into (20.67), the potential generated by the ideal dipole source becomes

$$y(t) = \frac{I_0(t)}{4\pi\sigma\|\mathbf{s} - \mathbf{r}_+\|} - \frac{I_0(t)}{4\pi\sigma\|\mathbf{s} - \mathbf{r}_-\|}. \quad (20.69)$$

Assuming that  $\|\mathbf{s} - \mathbf{r}_+\| \gg d$ , and similarly for  $\mathbf{r}_-$ , then

$$\begin{aligned} \frac{1}{\|\mathbf{s} - \mathbf{r}_+\|} &\approx \frac{1}{\|\mathbf{s} - \mathbf{r}\|} + \frac{(\mathbf{s} - \mathbf{r})^T \mathbf{d}}{2\|\mathbf{s} - \mathbf{r}\|^3}, \\ \frac{1}{\|\mathbf{s} - \mathbf{r}_-\|} &\approx \frac{1}{\|\mathbf{s} - \mathbf{r}\|} - \frac{(\mathbf{s} - \mathbf{r})^T \mathbf{d}}{2\|\mathbf{s} - \mathbf{r}\|^3}. \end{aligned}$$

Substituting this approximation into (20.69), the potential received at  $\mathbf{r}$  becomes

$$y(t) = \frac{I_0(t)}{4\pi\sigma} \left[ \frac{1}{\|\mathbf{s} - \mathbf{r}\|} + \frac{(\mathbf{s} - \mathbf{r})^T \mathbf{d}}{2\|\mathbf{s} - \mathbf{r}\|^3} - \left( \frac{1}{\|\mathbf{s} - \mathbf{r}\|} - \frac{(\mathbf{s} - \mathbf{r})^T \mathbf{d}}{2\|\mathbf{s} - \mathbf{r}\|^3} \right) \right] = \frac{1}{4\pi\sigma} \frac{(\mathbf{s} - \mathbf{r})^T}{\|\mathbf{s} - \mathbf{r}\|^3} \mathbf{m}(t), \quad (20.70)$$

where the dipole moment is defined as  $\mathbf{m}(t) = \mathbf{d}I_0(t)$ . In the sequel, we will write  $\mathbf{m}(t) = \boldsymbol{\phi}s(t)$ , where  $\boldsymbol{\phi} = \mathbf{d}/\|\mathbf{d}\|$  is the unit-magnitude dipole orientation, and  $s(t) = I_0(t)\|\mathbf{d}\|$  is the moment magnitude.

**MEG:** Extracranial magnetic fields produced by neuronal activity within the brain can be calculated using Biot-Savart's law. A dipole source at  $\mathbf{r}$  with dipole moment  $\mathbf{m}(t)$  will generate a magnetic field  $y(t)$  at sensor location  $\mathbf{s}$  given by

$$y(t) = \frac{\mu_0(\mathbf{m}(t) \times (\mathbf{s} - \mathbf{r}))^T}{4\pi\|\mathbf{s} - \mathbf{r}\|^3} \mathbf{t} = \frac{\mu_0((\mathbf{s} - \mathbf{r}) \times \mathbf{t})^T}{4\pi\|\mathbf{s} - \mathbf{r}\|^3} \mathbf{m}(t), \quad (20.71)$$

where  $\times$  denotes the vector cross product and  $\mathbf{t}$  is a unit vector defining the orientation of the sensor. As in the case of EEG, we will write  $\mathbf{m}(t) = \boldsymbol{\phi}s(t)$  with  $\boldsymbol{\phi}$  defining the orientation of the dipole moment. Note that a dipole inside a spherically symmetric conductor with  $\boldsymbol{\phi}$  aligned with the sphere's radius will produce no external magnetic field. Consequently, for MEG applications, the orientation  $\boldsymbol{\phi}$  (and hence the moment  $\mathbf{m}(t)$ ) is often expressed using only two rather than three coordinates.

**ECoG:** More involved models have been developed for ECoG settings due to the presence of local currents and higher SNR. For a sensor inside the skull on the surface of the brain at position  $\mathbf{s}$ , the measured potential  $y(t)$  due to a dipole source at  $\mathbf{r}$  with moment  $\mathbf{m}(t)$  in a homogeneous conducting sphere is given by

$$y(t) = \frac{1}{4\pi\sigma} \left( 2 \frac{\mathbf{r} - \mathbf{s}}{r_d^3} + \frac{1}{\|\mathbf{r}\|^2 r_d} \left[ \mathbf{r} + \frac{\mathbf{r}\|\mathbf{s}\| \cos \theta - \|\mathbf{r}\|\mathbf{s}}{\|\mathbf{r}\| + r_d - \|\mathbf{s}\| \cos \theta} \right] \right)^T \mathbf{m}(t), \quad (20.72)$$

where  $\sigma$  is the conductivity value for the brain,  $\theta$  denotes the angle between  $\mathbf{r}$  and  $\mathbf{s}$  and  $r_d = \|\mathbf{r} - \mathbf{s}\|$ .

**General multi-source multi-sensor model:** In all three cases discussed above, the equation for the electric or magnetic field has the same general form. In particular, for an array of  $M$  sensors at positions  $\mathbf{s}_i$ ,  $i = 1, \dots, M$ , the field can be represented as

$$y_i(t) = \mathbf{g}(\mathbf{s}_i, \mathbf{r})^T \mathbf{m}(t), \quad (20.73)$$

where the gain vector depends on which measurement system is employed:

$$\text{EEG: } \mathbf{g}(\mathbf{s}_i, \mathbf{r}) = \frac{1}{4\pi\sigma} \frac{\mathbf{s}_i - \mathbf{r}}{\|\mathbf{s}_i - \mathbf{r}\|^3}, \quad (20.74)$$

$$\text{MEG: } \mathbf{g}(\mathbf{s}_i, \mathbf{r}) = \frac{\mu_0(\mathbf{s}_i - \mathbf{r}) \times \mathbf{t}_i}{4\pi\|\mathbf{s}_i - \mathbf{r}\|^3}, \quad (20.75)$$

$$\text{ECoG: } \mathbf{g}(\mathbf{s}_i, \mathbf{r}) = \frac{1}{4\pi\sigma} \left( 2 \frac{\mathbf{r} - \mathbf{s}_i}{r_{d,i}^3} + \frac{1}{\|\mathbf{r}\|^2 r_{d,i}} \left[ \mathbf{r} + \frac{\mathbf{r}\|\mathbf{s}_i\| \cos \theta_i - \|\mathbf{r}\|\mathbf{s}_i}{\|\mathbf{r}\| + r_{d,i} - \|\mathbf{s}_i\| \cos \theta_i} \right] \right), \quad (20.76)$$

where all variables are as defined above, with the subscript  $i$  referencing sensor  $i$ . Thus, for all three models, stacking the outputs of the  $M$  sensors together in the vector  $\mathbf{y}(t)$  yields the same general equation:

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{g}^T(\mathbf{s}_1, \mathbf{r}) \\ \vdots \\ \mathbf{g}^T(\mathbf{s}_M, \mathbf{r}) \end{bmatrix} \mathbf{m}(t) = \mathbf{G}(\mathbf{r})\mathbf{m}(t) = \mathbf{a}(\mathbf{r}, \boldsymbol{\phi})s(t), \quad (20.77)$$

where  $\mathbf{G}(\mathbf{r})$  is  $M \times 3$  (or possibly  $M \times 2$  in the case of MEG data), and where the steering vector for the source depends on its location and dipole orientation:

$$\mathbf{a}(\mathbf{r}, \boldsymbol{\phi}) = \mathbf{G}(\mathbf{r})\boldsymbol{\phi}. \quad (20.78)$$

The steering vector model in (20.78) has the same form as in RF applications with diversely polarized signals. Finally, augmenting the model with the superposition of  $N$  sources as well as background interference  $\mathbf{n}(t)$ , we end up with the standard array processing equation:

$$\mathbf{y}(t) = [\mathbf{a}(\mathbf{r}_1, \boldsymbol{\phi}_1) \cdots \mathbf{a}(\mathbf{r}_N, \boldsymbol{\phi}_N)] \begin{bmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{bmatrix} + \mathbf{n}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t) + \mathbf{n}(t), \quad (20.79)$$

where the vector  $\boldsymbol{\theta}$  contains the source location and dipole orientation parameters.

The fact that the steering or array manifold vectors depend linearly on the dipole orientations, and that these vectors are assumed to satisfy  $\boldsymbol{\phi}_k^T \boldsymbol{\phi}_k = 1$ , lead to special types of solutions when estimating the parameters. For example, a direct implementation of the MUSIC algorithm leads to

$$\hat{\mathbf{r}}, \hat{\boldsymbol{\phi}} = \arg \min_{\mathbf{r}, \boldsymbol{\phi}} \frac{\boldsymbol{\phi}^T \mathbf{G}^T(\mathbf{r}) \mathbf{E}_n \mathbf{E}_n^T \mathbf{G}(\mathbf{r}) \boldsymbol{\phi}}{\boldsymbol{\phi}^T \mathbf{G}^T(\mathbf{r}) \mathbf{G}(\mathbf{r}) \boldsymbol{\phi}} \quad \text{s.t. } \boldsymbol{\phi}^T \boldsymbol{\phi} = 1, \quad (20.80)$$

where  $\mathbf{E}_n$  are the noise subspace eigenvectors of the covariance of  $\mathbf{y}(t)$ . It is straightforward to show that minimizing the MUSIC criterion is equivalent to solving the following generalized eigenvalue problem as a function of  $\mathbf{r}$ :

$$\hat{\mathbf{r}} = \arg \min_{\mathbf{r}} \lambda_{\min}(\mathbf{r}), \quad (20.81)$$

$$\mathbf{G}^T(\hat{\mathbf{r}}) \mathbf{E}_n \mathbf{E}_n^T \mathbf{G}(\hat{\mathbf{r}}) \hat{\boldsymbol{\phi}} = \lambda_{\min}(\hat{\mathbf{r}}) \mathbf{G}^T(\hat{\mathbf{r}}) \mathbf{G}(\hat{\mathbf{r}}) \hat{\boldsymbol{\phi}}, \quad (20.82)$$

where  $\lambda_{\min}(\mathbf{r})$  is the smallest generalized eigenvalue for a given  $\mathbf{r}$ . The position estimates  $\hat{\mathbf{r}}$  are found by searching for the value of  $\mathbf{r}$  for which  $\lambda_{\min}(\mathbf{r})$  is minimized, and the dipole orientation estimate is then given by the generalized eigenvector associated with  $\lambda_{\min}(\hat{\mathbf{r}})$ .

### 3.20.6.2.2 Interference mitigation

For EEG and MEG measurements, where the sensors are separated from the brain by the skull and scalp, the signals of interest are very weak, and embedded in strong, spatially correlated noise and interference due primarily to background brain activity not related to the stimulus of interest. If standard source

localization algorithms are applied without some attempt at mitigating this interference, the results are typically very poor. A common strategy in such situations is to design experiments with dual conditions, one (control state) prior to application of the stimulus and one (activity state) after the stimulus has been applied. In principle, the control state data will contain only background interference and sensor noise, while the activity state data will contain statistically similar noise and interference as well as the event-related signals. Prewhitening approaches are typically applied in dual-condition experiments like these. In these approaches, the control state data are first used to estimate the spatial covariance matrix of the interference plus noise using, for example, the following sample average:

$$\widehat{\mathbf{R}}_C = \frac{1}{n_C} \sum_{t=1}^{n_C} \mathbf{y}_C(t) \mathbf{y}_C^T(t), \quad (20.83)$$

where  $n_C$  is the number of control state samples. The activity state data,  $\mathbf{y}_A(t)$ , is then prewhitened in an attempt to eliminate the influence of the interference and noise as follows:

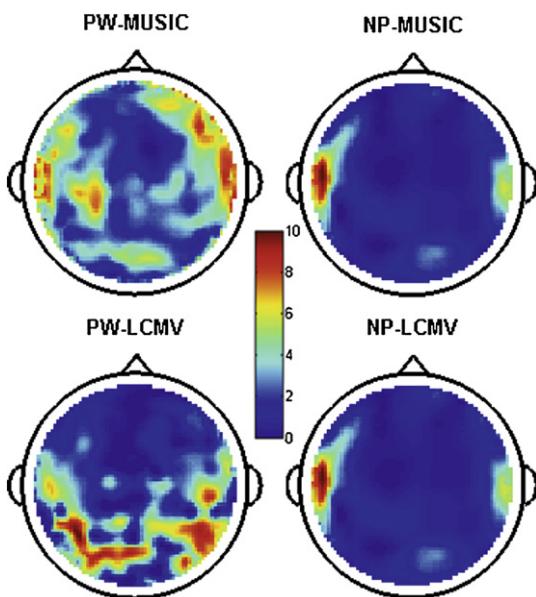
$$\mathbf{y}'_A(t) = \widehat{\mathbf{R}}_C^{-1/2} \mathbf{y}_A(t). \quad (20.84)$$

A drawback to the use of prewhitening is that it requires that the spatial and temporal statistics of the interference and noise during the control state be identical to those during the activity state. If the assumption of stationarity between these two states is violated, then methods based on prewhitening can suffer a significant performance degradation. An alternative is to use projection-based methods that estimate a spatial-only subspace in which the bulk of the interference energy lies during the control state, and then project away this subspace in the activity state data. This method eliminates the need for temporal stationarity, and relies only on the assumption that the locations of the interference in the control and activity states remain unchanged. This is a reasonable assumption since any “new” source that appears during the activity state is considered to be related to the stimulus, and is thus a source of interest.

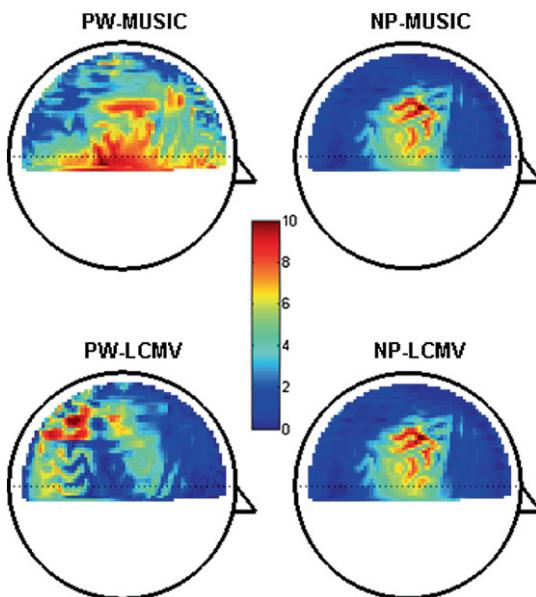
As an example, we present here the results of an experimental study with real EEG data. Experiments with an auditory stimulus applied to the left ear were conducted with a single human subject to elicit auditory-evoked potentials. MUSIC and LCMV were applied to the resulting data using both prewhitening (PW) and the projection (NP) technique and assuming the number of sources was one. Figures 20.33 and 20.34 show the spatial spectra of the four algorithm combinations. The projection-based methods provide an activity map that closely corresponds to what one would expect, with energy confined to the auditory cortex. On the other hand, the prewhitening-based methods contain a number of apparently unrelated artifacts, and the PW-LCMV method does not even show any energy near the auditory cortex.

### 3.20.6.3 Multi-sensor extracellular probes

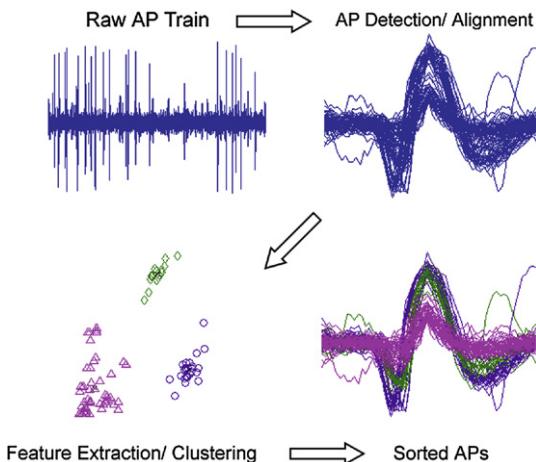
Direct measurement of neural action potentials (APs) using electrodes inserted directly into biological tissue has become an important neurological research and diagnostic tool. The goal is to record the APs of individual neurons, often referred to as “single-unit activity,” in order to obtain a more precise view of the underlying neurophysiology. Information from these recordings are potentially useful in the development of artificial prostheses and in the diagnosis and treatment of paralysis and brain disorders such as epilepsy and memory loss. Even though the electrodes are small and can be inserted with high accuracy to target a specific location, they will typically record the superposition of the activity from

**FIGURE 20.33**

Spatial spectra of four algorithm combinations using experimental data (top view).

**FIGURE 20.34**

Same as previous figure (side view).

**FIGURE 20.35**

Automated procedures for AP sorting.

several neurons. The process of separating out the single-unit activity of individual neurons from the multi-unit activity in the noisy electrode measurements is often referred to as AP or “spike” sorting.

In practice, manual sorting of APs in large volumes of experimental data is prohibitively time-consuming, and automated procedures for AP sorting have become essential. As depicted in Figure 20.35, an automated AP sorting algorithm can be divided into three main steps: (1) *AP detection and time alignment*: determining the locations of the APs in the electrode time series and arranging the isolated AP waveforms so that they “line up” in time, (2) *feature extraction*: extracting a low-dimensional set of parameters for each detected AP that can be used to discriminate between different sources, and (3) *clustering*: grouping the extracted features into clusters in order to associate them with individual neurons. The feature extraction step is crucial since it reduces the effect of noise and removes redundant information in the input data so that clustering algorithms can work efficiently. The three most common feature categories discussed in the literature are: (1) AP shape-related features, such as AP height, width, peak-to-peak amplitude, inter-AP interval, and first-order derivative, (2) wavelet coefficients, and (3) principal components (PCs). One common characteristic of these features is that they only capture “temporal” information since they are obtained by processing single-sensor measurements. However, AP sorting based only on temporal features is challenging since neurons with similar geometries located at roughly equal distances to the electrode can generate very similar AP waveforms and therefore similar features.

To overcome this problem, multi-sensor extracellular probes (e.g., tetrodes) that record a time-aligned multi-channel data set have been suggested. The simplest way to use the data from multi-sensor probes is to apply standard feature extraction techniques to all of the channels individually, and then combine all the extracted features as inputs for clustering. Other approaches use the availability of spatially distinct channel measurements to obtain neuron location estimates or independent components as feature vectors for clustering. Independent component analysis (ICA) is a computational method for separating a multivariate signal into additive subcomponents. While ICA can potentially resolve overlapping spikes,

it requires strong assumptions regarding the non-Gaussianity and independence of the APs, and a separate feature extraction step is still required to identify the source of the recovered AP waveform. Matched subspace (MS) techniques attempt to detect the presence of a signal that lies in an *a priori* unknown low-dimensional subspace of the data. Unlike multi-sensor principal component analysis and algorithms based on location estimates, which respectively allow only temporal or spatial information to be extracted, the MS approach provides a joint spatio-temporal feature vector that is more effective for differentiating between individual neurons. Furthermore, the spatial information obtained by MS techniques is achieved without the need for a forward propagation model as required by location-based methods.

### 3.20.6.3.1 Data model

Assume that the APs have been accurately detected in a previous step using existing approaches. A block of samples around each detected AP peak is isolated, and it is time-aligned with data blocks obtained for other detected APs. Assuming  $M$  electrodes and  $N$  samples per block, the data for the  $i$ th detected AP will consist of an  $M \times N$  matrix  $\mathbf{Y}_i$ , which is referred to as an AP “bundle.” Assuming that each bundle consists of an AP from a single neuron, and assuming that the AP signal results in an instantaneous mixture at the electrode array (i.e., a rank-one signal component), an appropriate mathematical model for  $\mathbf{Y}_i$  is:

$$\mathbf{Y}_i = \mathbf{S}_i + \mathbf{W}_i = \mathbf{a}_i \mathbf{v}_i^T + \mathbf{W}_i, \quad (20.85)$$

where  $\mathbf{S}_i \in \mathbb{R}^{M \times N}$  represents the noise-free multi-sensor signal corresponding to the AP,  $\mathbf{W}_i$  is composed of zero-mean background neural and sensor noise,  $\mathbf{a}_i \in \mathbb{R}^{M \times 1}$  is the spatial signature of the target neuron, and  $\mathbf{v}_i \in \mathbb{R}^{N \times 1}$  corresponds to the sampled AP waveform.

By vectorizing the data matrix, we obtain

$$\mathbf{y}_i = \mathbf{s}_i + \mathbf{w}_i = \mathbf{v}_i \otimes \mathbf{a}_i + \mathbf{w}_i$$

$$= \Phi \mathbf{c}_i \otimes \mathbf{a}_i + \mathbf{w}_i \quad (20.86)$$

$$= (\Phi \otimes \mathbf{a}_i) \mathbf{c}_i + \mathbf{w}_i, \quad (20.87)$$

where  $\otimes$  denotes the Kronecker product and  $\mathbf{y}_i$ ,  $\mathbf{s}_i$ , and  $\mathbf{w}_i$  are  $MN \times 1$  vectors formed from  $\mathbf{Y}_i$ ,  $\mathbf{S}_i$ , and  $\mathbf{W}_i$ , respectively. The term  $\mathbf{v}_i = \Phi \mathbf{c}_i$  models the AP in the absence of any specific information about the waveform, with matrix  $\Phi \in \mathbb{R}^{N \times p}$  ( $p \leq N$ ) representing a chosen orthonormal basis and  $\mathbf{c} \in \mathbb{R}^{p \times 1}$  representing the corresponding coefficient vector. Modeling the AP signal in this way not only provides the possibility of a compact representation for the AP but also eliminates the need for AP templates, which enables unsupervised spike sorting. Although  $\Phi$  can be any orthonormal basis, one with a compact support such as the wavelet basis is preferred in general since APs tend to be pulse-like.

### 3.20.6.3.2 Multi-sensor feature extraction

In the discussion that follows, we briefly describe several popular algorithms for feature extraction from a given  $MN \times 1$  AP bundle  $\mathbf{y}_i$ . In some cases, the methods either assume a single sensor ( $M = 1$ ) or they operate on each sensor independently. We will use the notation  $\mathbf{y}_i^{(k)} = \mathbf{y}_i(k : M : M(N - 1) + k)$  to represent the data from the  $k$ th sensor, where the indexing  $k : M : M(N - 1) + k$  indicates we select every  $M$ th sample from  $\mathbf{y}_i$  starting with sample  $k$ . The variable  $p$  will be used to denote the dimension of the extracted feature vector.

*Discrete wavelet transform:* The wavelet transform is a popular choice for feature extraction in the spike sorting application since it offers simultaneous interpretation of the signal in both time and scale (frequency), which allows local, transient or intermittent components to be elucidated. It has advantages over the traditional Fourier transform in analyzing physical signals since it can provide a compact signal representation in both the time and scale domains. The discrete wavelet transform (DWT) decomposes the data from a single sensor as follows:

$$\mathbf{y}_i^{(k)} = \Phi_w \mathbf{c}_i^{(k)}, \quad (20.88)$$

where  $\Phi_w \in \mathbb{R}^{N \times N}$  is a basis matrix that defines the DWT, and  $\mathbf{c}_i^{(k)} \in \mathbb{R}^{N \times 1}$  represents the DWT coefficient vector. The DWT basis is typically assumed to be orthonormal, so the coefficient vector is found by simply computing  $\mathbf{c}_i^{(k)} = \Phi_w^T \mathbf{y}_i^{(k)}$ . The feature vector  $\hat{\mathbf{c}}_i^{(k)} \in \mathbb{R}^{p \times 1}$  is determined by choosing a subset of  $p$  of the coefficients in the full DWT vector  $\mathbf{c}_i^{(k)}$ . The choice of which  $p$  coefficients to use can in principle be different for each sensor  $k$ , but must be the same for each AP bundle. For example, feature reduction for the DWT can be achieved by selecting the  $p$  coefficients that have the largest average magnitudes. Once a reduced-dimension set of features is chosen for each sensor, the complete feature vector is formed by stacking them all together:

$$\hat{\mathbf{c}}_w = \text{vec} \left( \begin{bmatrix} \hat{\mathbf{c}}_w^{(1)} & \dots & \hat{\mathbf{c}}_w^{(m)} \end{bmatrix} \right). \quad (20.89)$$

*Principal component analysis:* A difficulty associated with the DWT approach is there is no systematic way to choose the wavelet basis so that it is somehow optimized for the signals at hand. Principal component analysis (PCA) addresses this issue by calculating a data-dependent basis that corresponds to the principle subspace where most of the signal energy resides. This is most commonly achieved by performing the singular value decomposition (SVD) on a subset of  $n > p$  of the AP bundles

$$\mathbf{U}_k \Sigma_k \mathbf{V}_k^T = \begin{bmatrix} \mathbf{y}_{i_1}^{(k)} & \mathbf{y}_{i_2}^{(k)} & \dots & \mathbf{y}_{i_n}^{(k)} \end{bmatrix}, \quad (20.90)$$

where  $i_1, i_2, \dots, i_n$  are the indices corresponding to the  $n$  AP bundles chosen for the analysis. The PCA basis  $\Phi_p^{(k)} \in \mathbb{R}^{N \times p}$  is then taken to be the first  $p$  columns of  $\mathbf{U}_k$ , and the PCA feature vector (sometimes referred to as the “score” vector) is calculated as  $\hat{\mathbf{c}}_i^{(k)} = \Phi_p^{(k)T} \mathbf{y}_i^{(k)}$ . In most applications of PCA to this problem,  $p$  is chosen to be between two to three. Alternatively, a single basis for all  $k$  can be found by including AP bundles from all sensors in the SVD of (20.90). As in the DWT approach, once features are extracted for each  $k$ , the complete feature vector is found by stacking them together as in (20.89).

*Matched Subspace Detector:* The Matched Subspace Detector (MSD) can be thought of as a generalization of the well-known matched filter from signal processing, where a noisy signal  $\mathbf{y}_i$  is correlated with a parameterized version of the signal of interest  $\mathbf{s}$  to produce the output  $\mathbf{s}^T \mathbf{y}_i$ . The parameters are chosen as those that maximize the resulting correlation. The single-sensor versions of the DWT and PCA algorithms described in the previous section, where  $\mathbf{s} = \Phi \mathbf{c}$ , can be thought of as implementing a simple matched filter:

$$\begin{aligned} \hat{\mathbf{c}}_i &= \arg \max_{\mathbf{c}} \|\mathbf{s}^T \mathbf{y}_i\|^2 = \arg \max_{\mathbf{c}} \|\mathbf{c}^T \Phi^T \mathbf{y}_i\|^2 \\ \text{s.t. } \|\mathbf{c}\| &= \|\mathbf{y}_i\| = \alpha_i, \end{aligned} \quad (20.91)$$

where the constraint on  $\mathbf{c}$  is used to maintain energy equivalence. The solution to (20.91) is the same as that given earlier for DWT and PCA:  $\hat{\mathbf{c}}_i = \Phi^T \mathbf{y}_i$ .

The general MSD approach can be viewed as a natural multi-sensor extension of the single-sensor DWT or PCA approaches. Instead of the single-sensor parameterization  $\mathbf{s} = \Phi \mathbf{c}$ , the multi-sensor parameterization in (20.87) is used. In particular, MSD solves the following generalized version of (20.91):

$$\begin{aligned}\hat{\mathbf{a}}_i, \hat{\mathbf{c}}_i &= \arg \max_{\mathbf{a}, \mathbf{c}} \left\| \mathbf{s}^T \mathbf{y}_i \right\|^2 = \arg \max_{\mathbf{a}, \mathbf{c}} \left\| (\Phi \mathbf{c} \otimes \mathbf{a})^T \mathbf{y}_i \right\|^2 \\ \text{s.t. } \|\mathbf{a}\| &= 1, \|\mathbf{c}\| = \|\mathbf{y}_i\| = \alpha_i,\end{aligned}\quad (20.92)$$

where the constraints match those used in the model to ensure identifiability. It is straightforward to find a closed form solution for both  $\hat{\mathbf{a}}_i, \hat{\mathbf{c}}_i$ .

The MSD algorithm can be used in conjunction with either the DWT or PCA, or any other choice of the temporal basis matrix  $\Phi$ . The number of features produced by the MSD algorithm will be the  $M$  spatial features from the elements of  $\hat{\mathbf{a}}_i$ , plus however many temporal features are provided by  $\hat{\mathbf{c}}_i$ , which in turn depends on the dimension of  $\Phi$ . For the case of PCA, where  $\Phi \in \mathbb{R}^{N \times p}$  and typically  $p \ll N$ , the temporal feature vector  $\hat{\mathbf{c}}_i$  will have  $p$  elements. For the DWT, where  $\Phi$  is  $N \times N$ ,  $\hat{\mathbf{c}}_i$  will have  $N$  elements. Note that in this case,  $\hat{\mathbf{a}}_i$  can be found from the SVD of  $\mathbf{X}'_i$  rather than  $\mathbf{X}'_i \Phi$ , since both matrices have the same set of left singular vectors. Whether the total number of space-time features obtained by MSD is  $M + N$  or  $M + P$ , it is often desirable to reduce the number of features to a more manageable number.

### 3.20.7 Sonar

In the context of naval warfare, sonar is used to detect, locate, track and identify surface and submerged vehicles. This is one of the classical early applications of digital array signal processing. Since required bandwidths are often small and operating frequencies are low (10s of Hz to about 30 kHz) due to propagation limitations at higher frequencies in the ocean environment, corresponding sample and data rates are low enough to have been accommodated by ADCs and signal processing computers available in the 1970s and 1980s. The well funded military applications spurred rapid development in that era, and the transition from analog to digital systems enabled significant capability enhancements and increased processing complexity. Many of the classical array processing and statistically optimal beamforming algorithms were first demonstrated in sonar applications.

We will address two major classes of sonar: active and passive. Active sonar has much in common with radar systems in that a pulse, or sequence of pulses, is transmitted and the return echo signal is analyzed to detect vehicle range, direction, and range rate (radial velocity). This can be viewed as a non-cooperative digital wireless communications problem where the transmitted pulse corresponds to communications symbol, and two-way propagation effects including reflection from the target correspond to the communications channel.

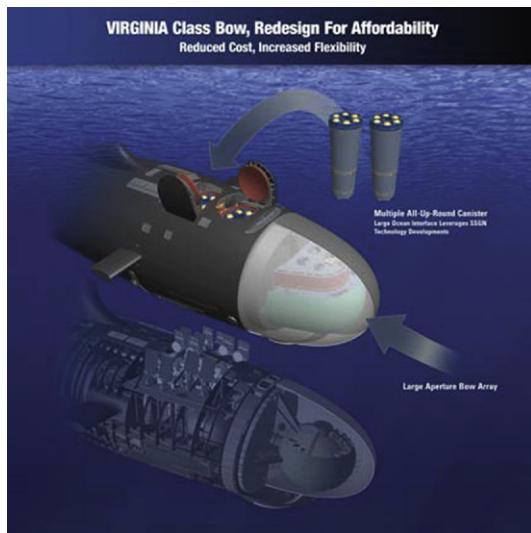
Passive sonar is a “listen only” mode used when stealthy operation is important so as not to reveal one’s own position with a transmitted pulse. Acoustic radiation is detected from the target’s turning propeller, internal machinery, occupants, or flow turbulence as it moves through the water. Passive systems can typically estimate target direction, and can classify the source as to speed, vehicle type, or even specific hull number by comparing the signal spectrum to previously obtained acoustic signature data bases.

The primary use of sensor and transmitting arrays in sonar is to exploit spatial information in the channel, including estimating directions of arrival, improving gain, and mitigating against noise and other interfering sources. We will also discuss in Section 3.20.7.3 how array processing combined with good acoustic propagation models can be used in passive sonar to estimate range and depth at a distance, without access to two-way propagation time-of-flight information.

### 3.20.7.1 Sonar arrays

The sensors used in sonar arrays are hydrophones that act as underwater microphones and acoustic drivers. Most hydrophones are constructed of ceramic piezoelectric transducer material, which operates effectively over the range of a few Hz to tens of kHz. In active sonar the same hydrophone elements are used for both transmit and receive. The total instantaneous array output power for long range sonar systems can be many tens of kilowatts. At very low frequencies some systems use electromagnetic linear motors (like speaker driver coils) or hydraulic actuators. Infrasonic pulses have also been generated using explosive charges.

Depending on the intended application and the supporting platform, sonar arrays are found in a variety of physical forms. Spherical or cylindrical arrays as seen in Figure 20.36 are housed in the bulbous bow protrusions below the water line of many military surface ships, and encased in the streamlined bow of submarines. These typically operate in the 1–6 kHz range and are capable of steering pencil beams in both azimuth and depression angle. An example of a spherical array is the US Navy AN/BSY-2 sonar on the Sea Wolf submarine.



**FIGURE 20.36**

An illustration of the spherical array in the Virginia III class of submarines.

*Credit: Defense Industry Daily.*

Conformal arrays use a thinly layered grid of hydrophones mounted on the nose or sides of vessels so as to blend smoothly with the contours of the hull design. Though this may be a less than ideal geometry for acoustic beamforming, it has the benefit of maintaining a streamlined structure for reduced drag and flow noise turbulence while allowing a larger aperture than is practical with a spherical array.

Long tubular towed array lines are pulled behind surface ships, submarines, and barges. Many hydrophones are spaced regularly inside a garden-hose-like tube that can be thousands of feet long. Depth is controlled either by adjusting the payout of the tow cable, or with an actively controlled tow body at the end of the array or at the tow cable attachment point. This enables steering to, and maintaining a desired depth (see Figure 20.37). Because of their length, towed arrays offer very large apertures for increased bearing resolution, narrow beams, high sensitivity due to many sensors and separation from ship self noise, and lower frequency operation as compared to hull mounted arrays. One drawback with the towed array is its one-dimensional linear geometry which leads to annularly symmetric (donut shaped) formed beampatterns. This yields no directivity in the vertical dimension, and a left-right ambiguity that requires the support ship to make turn maneuvers to resolve. The low frequency, long range, barge-towed US Navy SURTASS system is an example of a towed array sonar.

When mobility is essential or when submarine detection is needed at the far perimeter of the sonar reach from a naval battle group, then helicopter-borne dipping sonar is highly effective. A sonar array is reeled down to great depth from a hovering helicopter. Figure 20.38 shows a 1980s era system that is still in service, the US Navy AN/AQS-13 sonar. More recent developments like the US Navy AN/AQS-22 ALFS dipping sonar include extendable hydrophone support arms which increase aperture and permit lower frequency operation.

Modern torpedoes like the US Navy Mk 48 ADCAP shown in Figure 20.39 are quite autonomous. They are able to search out, detect, track, and target surface ships and submarines without the necessity



**FIGURE 20.37**

A French type F70 frigate (the Motte-Picquet) fitted with VDS (Variable Depth Sonar) type DUBV43 or DUBV43C towed array sonars. The array reeling mechanism and tow depth control body can be seen.

*Credit: Used by permission, NetMarine. Photographer: Jean-Michel Roche.*

**FIGURE 20.38**

A US Navy 1980s era Sikorsky SH-3H Sea King helicopter lowers its AN/AQS-13 dipping sonar.

*Credit: US DefenseImagery ([www.defenseimagery.mil](http://www.defenseimagery.mil)), PH1 R.O. Overholt, USN.*

**FIGURE 20.39**

Maintenance on an early development model of the US Navy Mk 48 ADCAP torpedo. The sonar hydrophone array lies behind the rubber shielded flat front nose plate.

*Credit: US DefenseImagery ([www.defenseimagery.mil](http://www.defenseimagery.mil)).*

of guidance and control from the launching boat. These tasks are performed using a nose-mounted planar array and on-board signal processing. The Mk 48 array is a nose-mounted grid of piezoelectric hydrophones which steer pencil beams for detection and direction finding.

Other sonar arrays are not mobile, but are permanently moored to the ocean floor. We will discuss in Section 3.20.7.3 how a fixed vertical line array can be used in matched field processing to estimate source range and depth using only passive observations. There are a number of very large and widely dispersed bottom-affixed passive sensor arrays used for surveillance in strategic ocean regions, including the US Navy's SOSUS network.

### 3.20.7.2 The undersea acoustic channel

Effective sonar signal processing requires an understanding of the challenging characteristics of sound propagation in an the ocean environment. In many ways sonar propagation is more complex and variable than the radio frequency channel encountered in wireless communications, radio astronomy, or radar. Fortunately though, propagation in the deep ocean is well understood, can be modeled accurately, and coherent processing across a large sensor array is possible even for distant sources.

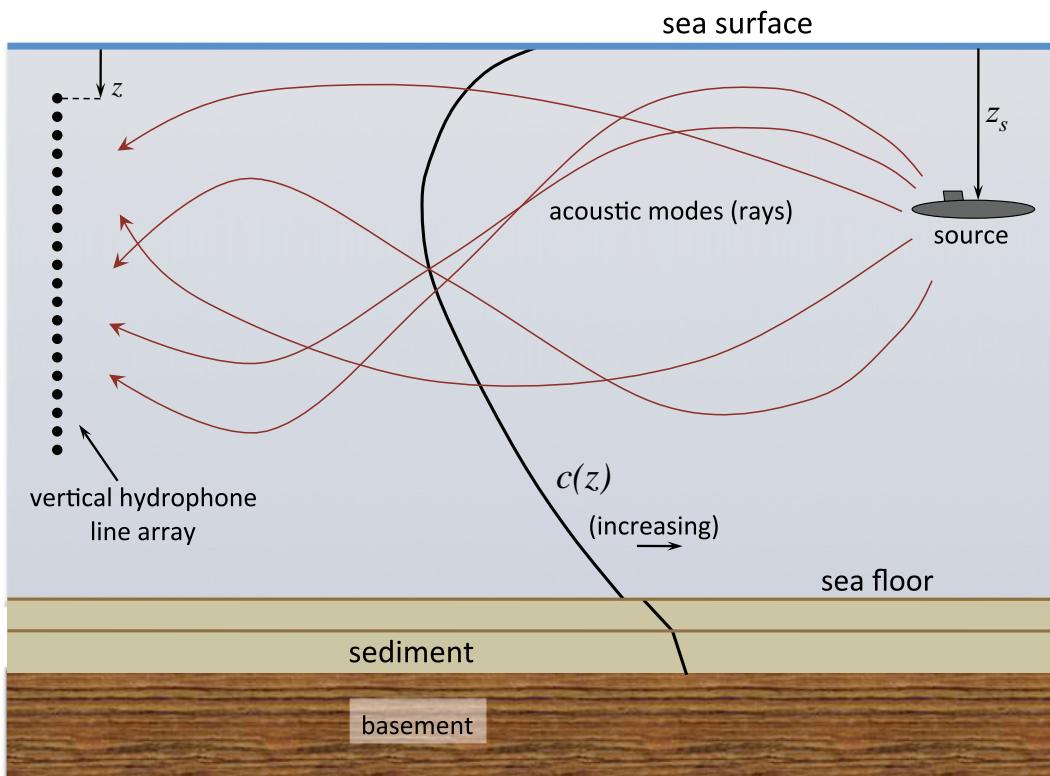
#### 3.20.7.2.1 Propagation models

Sound velocity in salt water is nominally 5000 ft per second, but this varies significantly with depth, sea temperature, and local salinity. Figure 20.40 illustrates a representative depth-dependent sound velocity profile  $c(z)$  and the resulting ray propagation characteristics. The increasing velocity near the sea floor is due to greatly increased pressure in the deep isothermal layer below about 3000 ft. Velocity also increases as depth decreases between the deep sound channel and the surface duct due to rising temperature as depth decreases in the main thermocline layer. The cross-over between these two effects leads to a velocity minimum that focuses acoustic energy in the stable deep sound channel which can propagate great distances and maintain coherency across the ray paths.

Propagation effects near the surface are much more variable and depend on diurnal heating and cooling, surface mixing due to wind and wave action, latitude, and formation of seasonal thermocline layers. A relatively shallow surface layer duct often forms which traps energy near the surface, allowing sonar propagation and detection with shallow arrays. This may be the only possibility if the source also lies in the duct. However, due to sea roughness and losses at the sea-air interface, transmission loss is greater in this layer and rays die out more rapidly than in the deep sound channel.

Another important propagation effect not illustrated in Figure 20.40 is the convergence zone. Rays of higher angular incidence (at the sensor array or source) will periodically extend beyond the deep sound channel and intersect the surface. This forms a ring on the surface at a fixed range from the sonar array of convergent ray paths that enable surface ship detection at great distances. The radial separation between the successive convergence zones is typically on the order of 20 miles.

Optimal placement of sensor arrays (in depth), identifying convergence zones, and estimating range require specific knowledge of  $c(z)$  to enable numerical ray path modeling of the sound channel. This information is obtained to great depths by bathythermograph and velocimeter sounders which are dropped overboard from surface ships or non-retrievably deployed from helicopters or other naval support fixed wing aircraft (e.g., the PC-3 Orion). When there are multiple vessels in the operational theater it is possible to collect these environmental data periodically over a large area. In situ measurements are supplemented with historical data, seasonal and weather models, and ocean bottom topography data to provide quite accurate sound velocity profile results. These enable useful acoustic channel ray trace modeling.



**FIGURE 20.40**

Acoustic propagation model for a horizontally stratified ocean. The convex-to-the left sound velocity function  $c(z)$  forms long distance propagating modes (or acoustic ray paths) in the deep sound channel. The complex wavefront seen at the hydrophone array depends on source depth  $z$  and range, enabling estimation of these source parameters using MFP.

### 3.20.7.2.2 Transmission loss

Signal loss in the undersea acoustic channel is due to physical wavefront spreading, volume absorption, and leakage and scattering at the surface and bottom. One would expect near-field propagation to follow a spherical spreading law with loss proportional to  $1/r^2$ , and long distance propagation confined by the ocean surface and sea bottom to a planar disc to have cylindrical spreading loss proportional to  $1/r$ , where  $r$  is range to the source. However extensive field measurements suggest that due to scattering and leakage, spherical spreading with  $1/r^2$  loss is a better match over a wide range of conditions. A commonly used model for sonar transmission loss in dB is

$$TL \approx (20 \log_{10} r) + \alpha r \times 10^{-3}, \quad (20.93)$$

$$\alpha = \frac{16\pi^2}{3\rho c^3} \left( \mu_s + \frac{3}{4}\mu_v \right) f^2, \quad (20.94)$$

where  $\alpha$  is defined in units of decibels loss per thousand yards,  $r$  is (following sonar convention) in yards, and  $f$  is frequency in Hz. The first term in (20.93) is due to spherical spreading. Other parameters for pure distilled water are density  $\rho \approx 1 \text{ gm/cm}^3$ , sound velocity  $c \approx 1.5 \times 10^5 \text{ cm/s}$ , shear viscosity  $\mu_s \approx 0.01 \text{ poises}$ , and volume viscosity  $\mu_v \approx 0.0281 \text{ poises}$ . Below about 100 Hz the effective  $\alpha$  in sea water increases (as compared to distilled water) by a factor of 30 due primarily to dissolved magnesium sulfate.

The fact that attenuation in dB is proportional to  $f^2$  suggests that for long range detection it will be highly advantageous to use low frequencies. This is born out in practice where short range and targeting sonars with small arrays typically operate at tens of kHz, medium range hull mounted arrays and helicopter dipping sonars are at 1–6 kHz, and long range towed array sonars cover 10 Hz to a few 100 Hz. Even at low frequencies, spreading losses are significant at long ranges so in order to put sufficient energy into the water, active sonar systems typically use very long transmit pulses on the order of several seconds. Fortunately the sound channel is stable over such long pulse periods and coherent matched filter detection processing of echoes is possible.

### 3.20.7.2.3 Noise and reverberation

Sonar systems must detect weak signals in an inherently very noisy environment. Noise sources are many and varied, but we will mention approximate average levels for some significant sources in deep water conditions.

- Between 1 Hz and 10 Hz there are a variety of sources that contribute to an average level of approximately 105 dB rel 1  $\mu\text{Pa}$  at 1 Hz, which declines with a slope of  $-30 \text{ dB}$  per decade of frequency increase.
- Between about 10 Hz and 150 Hz, the dominant source is mechanical and turbulence noise from distant surface shipping. Acoustic levels range from 60 to 85 dB rel 1  $\mu\text{Pa}$  for light to heavy shipping traffic conditions, respectively.
- Between about 100 Hz and 100 kHz The dominant source is surface noise from wind and wave action. Levels decline with increased frequency with a slope of about  $-20 \text{ dB}$  per decade. At 1 kHz surface noise levels range from 44 to 70 dB rel 1  $\mu\text{Pa}$  for sea state 0–6, respectively.

Additional external noise sources include biologics such as shrimp (one of the loudest) and marine mammals. Self-generated noise from the platform vehicle is of course very local and potentially strong. It includes flow noise due to turbulence across the hydrophone surfaces for a moving platform, and propulsion, machinery and other noise associated with the support vehicle.

Flow noise is very local to each hydrophone and is thus well modeled as statistically independent per sensor, and often i.i.d. Surface sea state noise is typically quite widespread and can often be approximately modeled as isotropic within a horizontal plane containing the array. Shipping noise can be directional (spatially colored) since it is concentrated in well traveled shipping lanes. Biologic sources include very direction-dependent and distant marine mammals and more local swarms of shrimp-like noise makers. Beamforming algorithms which place nulls on the nearby directional noise sources can be very effective in improving SNR in this environment.

Detection processing in active sonar must deal with significant reverberation. There are three main sources: volume reverberation, surface reflections, and bottom backscatter from rough clutter topographic features. Volume reflection is due to widely distributed particulate matter and marine life in the path of the transmit beam. It is strongest during the early portion of the pulse period. The initial surface

reflection arrives from directly above the array, with additional backscatter occurring as ray paths intersect the surface during rough sea-air interface conditions due to higher sea state. Multiple bottom-surface reflections lead to a nearly periodic structure for reverberation peaks within an exponentially decaying envelope. Reverberation can be reduced by extending the vertical array aperture to narrow the transmit and receive beampatterns in the vertical dimension. A time-dependent automatic gain control is also used in the receiver to avoid signal clipping during strong reverberation early in the pulse period. When the target of interest is moving, Doppler gating can also be used to reject reverberation from stationary clutter and to highlight the frequency-shifted target echo return.

### 3.20.7.3 Matched field processing

Matched field processing (MFP) is an interesting passive sonar application where modeled ray propagation is compared with the signal spatial structure at the receive array to estimate parameters of interest. The classical MFP application is to estimate source (target) range and depth at great distances, though it is theoretically possible to use the technique to identify environmental parameters such as ocean floor geography, propagation medium inhomogeneities, and even global undersea tomography. Unlike active sonar where two-way time of flight is used to directly measure range, and depth is not usually observable, in passive MFP it is possible to infer these parameters from the spatial phase and amplitude structure of the wavefront at the sensor array. Source localization can be viewed as an acoustic channel inversion problem exploiting sufficient complexity in the spatial signal distribution across a sensor array and a well-modeled channel transfer function between any candidate source position and each array sensor. This is not simple wavefront curvature estimation which can only be used effectively at shorter ranges within the Fresnel limits of the array. Success depends on the ability to accurately model the ducted, wave-guided acoustic propagation from source to sensor in a planar channel constrained by the ocean surface above, sea floor below, and a thermal-gradient-induced refractive deep-sound channel which directs (bends) acoustic rays with shallow incidence angles back toward the channel center. As illustrated in Figure 20.40, signals arrive at the array as a finite set of multipath rays, or acoustic modes, which are the discrete solutions to the frequency domain wave Eq. (20.95) and whose angular and depth distribution, or spatial spectrum, depends on the channel structure and source and sensor element positions.

To solve the channel inversion problem, source position parameters are varied within a propagation model for an optimization search to find the best “match” between the predicted and observed acoustic “field” at the sensor array. MFP relies on accurate full wave parametric modeling of acoustic waveguide propagation between the source and the array of hydrophone sensors, typically constructed as a vertical line array. Model mismatch of course impairs localization performance, but at low frequencies (10–100s of Hz) the sea channel maintains remarkable phase coherency across propagation modes and ray paths and existing models are sufficiently accurate. The MFP approach has been successfully demonstrated over ranges of hundreds of kilometers. Externally provided environmental parameters needed in the model include the sound velocity profile, bottom composition, and bottom topography. Contemporary measurements are obtained within a few hours of the MFP observations by deploying sounding instruments overboard to record sound velocity to great depths.

In this section we will follow in part the development found in [131]. Assuming waveguided propagation and a distant source, surface and bottom scattered signal components are attenuated to the point where they are negligible relative to modal components. These modes represent the multiple ray paths in

the deep sound channel between the source and individual array elements, and are the discrete solutions to the temporal frequency domain wave equation

$$\left[ \nabla^2 + K^2(\mathbf{v}) \right] g(\mathbf{v}, \mathbf{v}_s) = -\delta(\mathbf{v} - \mathbf{v}_s), \quad (20.95)$$

where  $\mathbf{v} = [x, y, z]^T$  is a position vector for an arbitrary point in the ocean channel,  $g(\mathbf{v})$  is the normalized (assuming a unit amplitude source) velocity potential or pressure,  $\nabla^2$  is the spatial Laplacian operator,  $K(\mathbf{v}) = \frac{\Omega}{c(\mathbf{v})}$  is the position dependent medium wave number,  $\Omega$  is the radian frequency of the narrowband acoustic source,  $c(\mathbf{v})$  is the local sound velocity,  $\mathbf{v}_s$  is the source position, and  $\delta(\cdot)$  is the 3D delta function. Since the source is modeled as a fixed point radiator, solutions  $g(\mathbf{v})$  are interpreted as the Green's function for the propagation channel between  $\mathbf{v}_s$  and  $\mathbf{v}$ .

It is often possible to model the acoustic channel with a horizontally stratified ocean as shown in Figure 20.40, where sound speed  $c(\mathbf{v}) = c(z)$  is a function of depth only and surface and bottom act as partially reflecting parallel plates. At greater distances this model is more accurate since only rays confined in the deep sound channel have survived surface and bottom scattering and attenuation. In this case we may separate out dependence on  $z$  in (20.95) and simplify by centering the  $(x, y)$  coordinate system on the source so  $g(\mathbf{v}, \mathbf{v}_s)$  may be re-parameterized as  $g(\bar{\mathbf{v}}, z, z_s)$  where  $\bar{\mathbf{v}} = [x - x_s, y - y_s]^T$ . The 2-D inverse spatial Fourier transform relationship with respect to only  $x$  and  $y$  is then

$$g(\bar{\mathbf{v}}, z, z_s) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} G(\mathbf{k}, z, z_s) e^{j\mathbf{k}^T \bar{\mathbf{v}}} d\mathbf{k}, \quad (20.96)$$

where  $\mathbf{k} = [k_x, k_y]^T$  is the 2-D horizontal wavenumber vector. Note that the medium wavenumber is given by  $K^2 = k_x^2 + k_y^2 + k_z^2$ . Acoustic pressure  $g(\bar{\mathbf{v}}, z, z_s)$  is interpreted as that seen by a hydrophone at depth  $z$  and 2D range  $\bar{\mathbf{v}}$  relative to the source, which is at depth  $z_s$ . The wave equation is then expressed in the 2-D spatial frequency domain by substituting (20.96) into (20.95)

$$\frac{1}{4\pi^2} \int_{-\infty}^{\infty} \left[ \nabla^2 + K^2(z) \right] G(\mathbf{k}, z, z_s) e^{j\mathbf{k}^T \bar{\mathbf{v}}} d\mathbf{k} = -\frac{1}{4\pi^2} \int_{-\infty}^{\infty} \delta(z - z_s) e^{j\mathbf{k}^T \bar{\mathbf{v}}} d\mathbf{k}, \quad (20.97)$$

$$\left[ \frac{\partial^2}{\partial z^2} + K^2(z) - |\mathbf{k}|^2 \right] G(\mathbf{k}, z, z_s) = -\delta(z - z_s), \quad (20.98)$$

where in (20.97) we have used  $\delta(\mathbf{v} - \mathbf{v}_s) = \delta(\bar{\mathbf{v}})\delta(z - z_s)$  and  $-\frac{1}{4\pi^2} \int_{-\infty}^{\infty} e^{j\mathbf{k}^T \bar{\mathbf{v}}} d\mathbf{k} = \delta(\bar{\mathbf{v}})$ . Equation (20.98) follows by matching the Fourier transform arguments, noting that  $G(\mathbf{k}, z, z_s)$  does not depend on  $x$  or  $y$ , and that the vertical wave number is given by  $k_z^2 = K^2 - |\mathbf{k}|^2$ . Solutions  $G(\mathbf{k}, z, z_s)$  to (20.98) are known as the depth dependent Green's functions.

In general these solutions span a continuous range of  $\mathbf{k}$  corresponding to the different directions of arrival for the wave fronts at  $z$ . But in the horizontally stratified, waveguided case we are considering, only discrete values of  $\mathbf{k}$  for distinct ray paths represent propagation modes with significant energy, and these directions of arrival at  $(\bar{\mathbf{v}}, z)$  are confined to the vertical plane containing the source and sensor. Thus  $G(\mathbf{k}, z, z_s)$  typically consists of a series of complex-amplitude-scaled delta functions at discrete wavenumbers  $\mathbf{k}_n$  corresponding to the  $n$ th propagation ray. The inverse transform of (20.96) then takes the form

$$g(\bar{\mathbf{v}}, z, z_s) = \frac{1}{4\pi^2} \sum_n G(\mathbf{k}_n, z, z_s) e^{j\mathbf{k}_n^T \bar{\mathbf{v}}}. \quad (20.99)$$

A central component of every MFP algorithm is a numerical simulation for the forward propagation model. Given a sound velocity profile  $c(z)$ , the simulation solves (20.98) for the discrete rays and uses (20.99) to compute  $g(\bar{\mathbf{v}}, z, z_s)$  as a function of all  $\bar{\mathbf{v}}$  and  $z_s$  values in the search range and for every  $z$  corresponding to a sensor array element depth. The fundamental MFP strategy is to solve the inverse propagation problem with an exhaustive search for the best match with respect to  $\bar{\mathbf{v}}$  and  $z_s$  between the forward modeled responses and the measured sensor array response structure seen in output samples  $\mathbf{y}(i)$ .

Assuming a vertical line array of  $M$  elements and an MFP search in range and depth, the sampled data vector at the array is

$$\mathbf{y}(i) = \mathbf{a}(\bar{\mathbf{v}}, z_s)s(i) + \mathbf{n}(i), \quad (20.100)$$

where the parametric array spatial response vector is given by

$$\mathbf{a}(\bar{\mathbf{v}}, z_s) = \begin{bmatrix} g(\bar{\mathbf{v}}, z_1, z_s) \\ \vdots \\ g(\bar{\mathbf{v}}, z_M, z_s) \end{bmatrix} \quad (20.101)$$

and where  $z_m$ ,  $1 \leq m \leq M$ , is the depth of the  $m$ th array hydrophone sensor,  $s(i)$  is a zero-mean Gaussian random source process with variance  $\sigma_s^2$  and  $\mathbf{n}(i)$  is the noise sample vector including flow noise, surface winds and shipping, biologics, etc. Assuming wide sense stationarity, the covariance matrix is

$$\mathbf{R} = E[\mathbf{y}(i)\mathbf{y}^H(i)] = \sigma_s^2 \mathbf{a}(\bar{\mathbf{v}}, z_s)\mathbf{a}^H(\bar{\mathbf{v}}, z_s) + \mathbf{R}_n = \mathbf{R}_s(\bar{\mathbf{v}}, z_s) + \mathbf{R}_n. \quad (20.102)$$

As an MFP performance metric one can quantify how unique the array spatial response is for distinct values of the parameters  $\bar{\mathbf{v}}$  and  $z_s$ , since this is related to the invertibility of the channel. To this end, a very useful measure is the ambiguity function defined as

$$\phi(\bar{\mathbf{v}}_1, z_1; \bar{\mathbf{v}}_2, z_2) = \left| \frac{\mathbf{a}^H(\bar{\mathbf{v}}_1, z_1)}{\|\mathbf{a}(\bar{\mathbf{v}}_1, z_1)\|} \cdot \frac{\mathbf{a}(\bar{\mathbf{v}}_2, z_2)}{\|\mathbf{a}(\bar{\mathbf{v}}_2, z_2)\|} \right|^2. \quad (20.103)$$

If  $\phi(\bar{\mathbf{v}}_1, z_1; \bar{\mathbf{v}}_2, z_2)$  has multiple equally large peaks, then the MFP solution is ambiguous. Ideally it would have a single narrow peak at  $(\bar{\mathbf{v}}_1 = \bar{\mathbf{v}}_2; z_1 = z_2)$  for all  $\bar{\mathbf{v}}_1, z_1$ , which would yield high resolution unique solutions, but significant sidelobe patterns are common. Array length and depth and the sound velocity profile  $c(z)$  affect the shape of the ambiguity function and thus channel invertibility.

### 3.20.7.4 Acoustic vector sensors

As we transition to microphone arrays for aeroacoustical applications, we briefly mention here a relatively new type of acoustical *vector* sensor (AVS), which essentially amounts to an “array on a sensor.” An AVS can measure the vector-valued acoustic particle velocity in addition to the scalar-valued sound pressure, and such sensors have been manufactured for acoustic measurements in both air and water. An example of an aeroacoustic vector sensor is shown in Figure 20.41. The output of a general AVS in free space can be represented as

$$\mathbf{y}(t) = \begin{bmatrix} 1 \\ \mathbf{u}(\theta, \phi) \end{bmatrix} x(t) + \mathbf{n}(t), \quad (20.104)$$

**FIGURE 20.41**

A vector acoustic sensor manufactured by Microflown Technologies, The Netherlands.

where  $x(t)$  represents the sound pressure,  $\mathbf{n}(t)$  noise and interference, and

$$\mathbf{u}^T(\theta, \phi) = [\cos(\theta) \cos(\phi) \quad \sin(\theta) \cos(\phi) \quad \sin(\phi)]^T$$

is a unit vector at the sensor pointing towards the source at azimuth angle  $\theta$  and elevation angle  $\phi$ . If the sensor is located near a reflecting surface (e.g., a wall or the ocean floor), then a reflection term is added to (20.104) to account for the source image.

The key distinguishing feature of an AVS is the fact that it produces a four-dimensional measurement at essentially a single point in space. A single AVS can be used to localize two separate sources, and additional resolving power can be obtained by an array of AVS within a relatively small aperture. AVS provide an interesting alternative to standard hydrophones or microphones in acoustic source localization and signal recovery.

### 3.20.8 Microphone arrays

The processing of acoustic signals in the air using arrays of microphones has received significant attention, although considerably less than for underwater acoustics due to the ubiquitous use of sonar in naval operations. While the use of microphone arrays has also been proposed for military applications, such as localization or identification of vehicles, helicopters, sniper fire, etc., such arrays have perhaps enjoyed more success in commercial settings, particularly those related to speech recovery or enhancement. For example, the ability of microphone arrays to locate an acoustic source such as a speaker, extract an acoustic signal in a noisy and reverberant environment, and synthesize arbitrary sound fields has led to

**FIGURE 20.42**

4 × 4 microphone array manufactured by iSEMcon GmbH, Germany.

their use in advanced video-conferencing systems, “hands-free” communication systems, surveillance of criminal activity and simulation of concert hall acoustics in high-end audio systems. Figure 20.42 shows a 4 × 4 array used for sound field mapping and source localization. A simple array typical of those used for teleconferencing applications is shown in Figure 20.43.

Similar to underwater acoustics, signal processing with microphone arrays relies on wideband data models, where the propagation time across the array is usually much greater than the inverse bandwidth. For example, it takes about 6 ms for sound to travel one half meter (a typical array aperture), while the inverse bandwidth of a speech signal is around 0.2–0.5 ms. On the other hand, since the speed of sound in air is over four times slower than in water, and since the frequencies of interest for aeroustics are usually higher than in sonar, microphone arrays can be much more compact. At 1 kHz, the wavelength of sound is about 30 cm, so arrays with apertures under a few meters are common. Consequently, plane-wave propagation models are typically assumed, at least locally, in the vicinity of the array. Propagation in outdoor environments is complicated by wind and temperature gradients that make precise localization difficult over long ranges. Even in situations where straight-line propagation can be assumed, random fluctuations in the air and temperature will cause a transmitted and received acoustic signal to lose temporal coherence if the signal travels a large distance. Indoors, the main obstacle to overcome is reverberation due to reflections of the sound from floors, walls, ceilings, furniture, etc.

**FIGURE 20.43**

Microphone array used in video-conferencing applications. Manufactured by Polycom, Inc., San Jose, CA.

Consequently, the focus of most microphone array applications in outdoor settings is source localization, while indoors the most common problem is reconstruction of a desired acoustic source in the presence of noise and multipath. We briefly discuss aspects of these two problems below.

### 3.20.8.1 Aeroacoustic source localization

The term “acoustic camera” is often used to refer to microphone arrays that are used to characterize sound fields and locate sources of acoustic energy. Since the aeroacoustic signals used for localization are typically wideband, models for the problem tend to be formulated in the frequency domain. Let  $\mathbf{y}(t) = [y_1(t) \cdots y_M(t)]^T$  denote the output of an  $M$ -microphone array. Assuming zero-mean wide-sense stationary signals, the array output is characterized by its cross-correlation matrix

$$\mathbf{R}_y(\tau) = \mathcal{E}\{\mathbf{y}(t + \tau)\mathbf{y}(t)^T\} \quad (20.105)$$

and the corresponding cross spectral density (CSD) matrix  $\mathbf{G}_y(\omega)$  whose  $i$ ,  $j$ th element is defined as

$$G_{y,ij}(\omega) = \int_{-\infty}^{\infty} R_{y,ij}(\tau) e^{-j\omega\tau} d\tau, \quad (20.106)$$

where  $R_{y,ij}(\tau)$  is element  $i$ ,  $j$  of  $\mathbf{R}_y(\tau)$ .

In general, the elements of the CSD may be expressed as

$$G_{y,ij}(\omega) = e^{-j\omega\tau_{ij}(\mathbf{p})} G_{s,ij}(\omega) + \sigma^2 \delta_{ij}(\omega), \quad (20.107)$$

where  $\sigma^2(\omega)$  is the CSD of the noise (assumed to be uncorrelated at each microphone),  $\tau_{ij}(\mathbf{p})$  is the propagation delay between the two microphones, which is a function of the location of the source  $\mathbf{p}$ , and

$$G_{s,ij}(\omega) = \gamma_{s,ij}(\omega) [G_{s,i}(\omega) G_{s,j}(\omega)]^{1/2}, \quad (20.108)$$

where  $\gamma_{s,ij}(\omega)$  is the spectral coherence function for the two sensors satisfying  $0 \leq |\gamma_{s,ij}(\omega)| \leq 1$ , and  $G_{s,i}(\omega)$  represents the CSD of the source at microphone  $i$ . In general,  $G_{s,i}(\omega) \neq G_{s,j}(\omega)$  when  $i \neq j$  due to propagation inhomogeneities that occur as the signal travels between the two microphones. If microphones  $i$  and  $j$  are close enough together such that one can assume spatially coherent planewave propagation between them, then  $\gamma_{s,ij}(\omega) = 1$  and  $G_{s,i}(\omega) = G_{s,j}(\omega)$ .

A convenient and very general approach is to assume an array-of-arrays situation, where several (say,  $K$ ) arrays of closely-spaced microphones with locally coherent propagation are distributed over a larger area and separated by distances over which coherent propagation cannot generally be assumed. This model subsumes the two cases discussed above. If the vector outputs of each array  $\mathbf{y}_k(t)$  are stacked on top of each other to form the super-vector  $\mathbf{y}(t) = [\mathbf{y}_1(t)^T \cdots \mathbf{y}_K(t)^T]^T$ , then the  $MK \times MK$  CSD matrix will be given by

$$\mathbf{G}_y(\omega, \mathbf{p}) = \begin{bmatrix} \mathbf{a}_1(\omega, \mathbf{p})\mathbf{a}_1^H(\omega, \mathbf{p})G_{s,1}(\omega) & \cdots & \mathbf{a}_1(\omega, \mathbf{p})\mathbf{a}_K^H(\omega, \mathbf{p})e^{-j\omega\tau_{1K}(\mathbf{p})}G_{s,1K}(\omega) \\ \vdots & \ddots & \vdots \\ \mathbf{a}_K(\omega, \mathbf{p})\mathbf{a}_1^H(\omega, \mathbf{p})e^{-j\omega\tau_{K1}(\mathbf{p})}G_{s,K1}(\omega) & \cdots & \mathbf{a}_K(\omega, \mathbf{p})\mathbf{a}_K^H(\omega, \mathbf{p})G_{s,K}(\omega) \end{bmatrix} + \sigma^2(\omega)\mathbf{I}, \quad (20.109)$$

where  $\mathbf{I}$  is an  $MK \times MK$  identity matrix (assuming for simplicity that the noise CSD is the same at each array),

$$\mathbf{a}_k(\omega, \mathbf{p}) = \begin{bmatrix} e^{-j\omega\tau_{k,11}(\mathbf{p})} \\ \vdots \\ e^{-j\omega\tau_{k,1M_k}(\mathbf{p})} \end{bmatrix}, \quad (20.110)$$

and where  $\tau_{k,ij}(\mathbf{p})$  represents the propagation delay between microphones  $i$  and  $j$  for array  $k$  with  $M_k$  elements.

If one has access to the outputs of all  $K$  of the arrays and the various source CSDs  $G_{s,ij}(\omega)$  are known, a procedure for estimating the source location  $\mathbf{p}$  based on samples of  $\mathbf{G}_y(\omega, \mathbf{p})$  at different frequencies can easily be formulated. Such an approach would require the arrays to share all their data with a fusion center, which incurs a large communication overhead. In addition, knowledge of  $G_{s,i}(\omega)$  implies that the arrays can somehow obtain time-aligned measurements of the source CSD, which is problematic without knowledge of the source location. The latter issue can be resolved by absorbing the time-delay terms  $e^{-j\omega\tau_{kl}(\mathbf{p})}$  between arrays  $k$  and  $l$  into  $G_{s,kl}(\omega)$ , and basing the estimate of  $\mathbf{p}$  on just the intra-array phase shifts. An alternative approach is to estimate the direction-of-arrival (DOA) of the source signal at each array using only the locally calculated CSD matrix  $\mathbf{a}_k(\omega, \mathbf{p})\mathbf{a}_k^H(\omega, \mathbf{p})G_{s,k}(\omega)$  (the location of the source is not identifiable at each array individually, only the source DOA). Each array would then forward only its estimated DOA to the fusion center, which would then estimate  $\mathbf{p}$  via triangulation. Various studies of the Cramér-Rao Bound have been conducted to determine the difference in achievable performance for these approaches.

### 3.20.8.2 Wideband adaptive beamforming

As mentioned above, in many microphone array applications, locating an acoustic source is less important than extracting its waveform in a reverberant and noisy environment. In relatively short-range

indoor settings where factors that influence acoustic propagation (temperature, pressure, wind, etc.) are uniform, Doppler and dispersion effects can be ignored, and to a very good approximation the array will simply receive scaled and delayed versions of the source via a (potentially large) number of reverberant paths. In particular, at microphone  $m$ , the received acoustic signal can be represented as

$$y_m(t) = \sum_{i=1}^N \alpha_{i,m} s(t - \tau_{i,m}) + n_m(t), \quad (20.111)$$

where  $s(t)$  is the desired source,  $N$  denotes the number of multipath echoes from the source to the microphone,  $\{\alpha_{i,m}, \tau_{i,m}\}$  are the amplitude and the delay corresponding to path  $i$  at microphone  $m$ , and  $n_m(t)$  is due to all other background noise and interference.

The most common approach to extracting  $s(t)$  from the  $M$ -element microphone array output is via a wideband beamformer:

$$\hat{s}(t - t_0) = \sum_{m=1}^M \sum_{l=0}^L w_{ml} y_m(t - lT_s), \quad (20.112)$$

where  $T_s$  is the sampling period of the array,  $w_{ml}$  is the beamformer weight for microphone  $m$  at sample  $l$ , and  $t_0$  is an arbitrary delay. This essentially amounts to a space-time equalizer similar to what might be employed in a frequency-selective wireless RF channel. The difference in the microphone array application is that one typically does not have access to periodic “training” data from the source to facilitate updates of the beamformer/equalizer weights, either in time via the LMS or RLS algorithms, or using a data-adaptive approach like MVDR beamforming. Instead, other factors must be exploited to adapt the weights. For example, one may know or be able to estimate the approximate location or DOA of the source, as in automobile voice-enhancement or video conferencing systems where the speaker(s) are confined to certain positions. Likewise, knowledge of the location of strong sources of acoustic interference (e.g., TVs, air conditioners, windows, etc.) can also be taken advantage of to help the filter focus on the source of interest. Adaptive noise canceling approaches are possible if reference waveforms are available for the interference, obtained for example by placing a microphone near the interfering source. One can also exploit situations where the source or interference is known to have strong components at certain frequencies, although in this case it is advantageous to implement the filter in the frequency domain:

$$\hat{S}(\omega_k) = \sum_{m=1}^M W_m(\omega_k) Y_m(\omega_k). \quad (20.113)$$

Important factors to consider when implementing a wideband beamformer in microphone array applications are the sampling period  $T_s$  and the length  $L$  of the equalizer, which in many situations can be quite large for the required value of  $T_s$ . For example, to reconstruct a speech signal with a 3 kHz bandwidth in a room where the path lengths of the echos may vary by 5 m could require a value of  $L$  on the order of 300–400. For this reason, in computationally constrained scenarios, one may be forced to settle for a space-only beamformer followed by an adaptive echo canceler.

### 3.20.9 Chemical sensor arrays

In recent years, the use of model-based signal processing with chemical sensor arrays has received significant interest. Driving this interest has been important applications such as environmental monitoring of air and water quality, chemical spills, detection and localization of air- or waterborne chemical weapons and even landmines. The ability to quickly discover and accurately locate sources of toxic chemicals is obviously a critical factor in mitigating their negative impact. The term “model-based” is used here to contrast against classical methods that simply use arrays of sensors to improve coverage or increase the probability of detecting a chemical event. While these are clearly important, our focus below will be on approaches that employ parametric models to describe chemical flow across the sensors, and as such can be used to locate and quantify other properties of the source(s) in addition to simply detecting their presence.

The key differentiating feature of applications involving chemical sensor arrays compared with others considered in this chapter is the fact that the signals of interest propagate according to diffusion rather than wave equations. Additional complications such as imprecisely known wind/currents, turbulence, eddys, vortices and boundary effects make it difficult to obtain an accurate mathematical model except in fairly simple circumstances. Nevertheless, results obtained from simplified models of the environment can serve as valuable approximations that provide useful information. Furthermore, they can be used to focus the local implementation of more complicated numerical operations that would be too involved to perform globally.

To illustrate the application of sensor arrays in localizing a diffusing chemical source, we will consider a simple example involving a point source in an open environment (all surfaces and other boundaries are far enough removed from the source and array so that their effects can be neglected) with homogeneous diffusivity in all directions. Assume the source is located at position  $\mathbf{r}_0 = [x_0, y_0, z_0]^T$  and at time  $t_0$  begins emitting the chemical substance at a constant rate of  $\mu$  kg/s. Assume also that a wind/current is present with constant velocity vector  $\mathbf{v}$ . For this case, the diffusion equation that governs the concentration  $c(\mathbf{r}, t)$  of the substance at some position  $\mathbf{r}$  at time  $t > t_0$  is given by

$$\frac{\partial c(\mathbf{r}, t)}{\partial t} = \kappa \nabla^2 c(\mathbf{r}, t) - \nabla c \cdot \mathbf{v}, \quad (20.114)$$

where  $\kappa$  measured in  $\text{m}^2/\text{s}$  is the diffusivity of the medium, which in the above expression is assumed to be incompressible. The solution to (20.114) is given by  $c(\mathbf{r}, t) = \mu a(\mathbf{r}, t)$ , where

$$a(\mathbf{r}, t) = \frac{1}{8\pi\kappa\|\mathbf{r} - \mathbf{r}_0\|} \exp\left\{\frac{(\mathbf{r} - \mathbf{r}_0)^T \mathbf{v}}{2\kappa}\right\} \times \left[ \exp\left\{\frac{\|\mathbf{r} - \mathbf{r}_0\| \|\mathbf{v}\|}{2\kappa}\right\} \operatorname{erfc}\left(\frac{\|\mathbf{r} - \mathbf{r}_0\|}{2\sqrt{\kappa(t - t_0)}} + \|\mathbf{v}\| \sqrt{\frac{t - t_0}{4\kappa}}\right) \right. \quad (20.115)$$

$$\left. \times \exp\left\{-\frac{\|\mathbf{r} - \mathbf{r}_0\| \|\mathbf{v}\|}{2\kappa}\right\} \operatorname{erfc}\left(\frac{\|\mathbf{r} - \mathbf{r}_0\|}{2\sqrt{\kappa(t - t_0)}} - \|\mathbf{v}\| \sqrt{\frac{t - t_0}{4\kappa}}\right) \right], \quad (20.116)$$

where  $\text{erfc}(x) = \frac{2}{\pi} \int_x^\infty e^{-y^2} dy$  is the complementary error function. While technically the above model is appropriate for molecular diffusion, it can be applied to larger scale scenarios involving convective diffusion by adjusting the value of the diffusivity  $\kappa$ .

To characterize the chemical concentration at any point in space or time, one would need to know  $\mu$ , the “strength” of the source, as well as the parameters in the vector  $\boldsymbol{\theta} = [\mathbf{r}_0^T \ \kappa \ t_0]^T$ , which include the location of the source and the time it became active. The diffusivity  $\kappa$  is also treated as an unknown constant, since it will depend on environmental factors (temperature, humidity, etc.) in a complicated way. To determine these unknowns, an array of sensors that measure the concentration of the chemical can be deployed. For example, a given sensor located at position  $\mathbf{r}_i$  would observe the following concentration at some specific time  $t_k$ :

$$y_i(t_k) = a(\mathbf{r}_i, \boldsymbol{\theta}, t_k)\mu + n_i(t_k),$$

where  $n_i(t)$  represents noise or modeling errors, and  $a$  is written as an explicit function of  $\boldsymbol{\theta}$  to emphasize its dependence on the parameters of interest. If the observations from  $M$  sensors taken at  $K$  distinct time samples are stacked into a single observation vector  $\mathbf{y}$ , where element  $p$  of  $\mathbf{y}$  is indexed according to  $p = M(k-1) + i$  for  $k = 1, \dots, K$  and  $i = 1, \dots, M$ , the standard array processing model is obtained:

$$\mathbf{y} = \begin{bmatrix} a(\mathbf{r}_1, \boldsymbol{\theta}, t_1) \\ \vdots \\ a(\mathbf{r}_M, \boldsymbol{\theta}, t_1) \\ a(\mathbf{r}_1, \boldsymbol{\theta}, t_2) \\ \vdots \\ a(\mathbf{r}_M, \boldsymbol{\theta}, t_K) \end{bmatrix} \mu + \mathbf{n} = \mathbf{a}(\boldsymbol{\theta})\mu + \mathbf{n}, \quad (20.117)$$

where the vector of noise samples  $\mathbf{n}$  is organized like  $\mathbf{y}$ , and element  $p = M(k-1) + i$  of the “steering” vector  $\mathbf{a}(\boldsymbol{\theta})$  is given by  $a(\mathbf{r}_i, \boldsymbol{\theta}, t_k)$ . With the model of (20.117) in hand, one can apply standard array processing techniques to estimate  $\mu$  and  $\boldsymbol{\theta}$ , provided that (20.117) is identifiable. In principle, unique identification of the three location parameters in  $\boldsymbol{\theta}$ , namely  $\mathbf{r}_0$ , requires that  $M \geq 4$ , and of course we require that the total number of observations  $MK$  exceed the number of free parameters (six in this model). In practice, of course,  $MK$  will likely need to be much larger than six in order to combat the effects of noise.

While the discussion above was for the simple case of an infinite open environment, a similar approach can be taken for more complicated scenarios provided that the diffusion equation can be solved. Cases of particular interest that have been addressed include a semi-infinite medium (e.g., a source on the ocean floor) and a large room of known dimensions. Boundary conditions play an important role in such cases, and different results are obtained depending on whether or not the boundaries are permeable to the chemical of interest. Source models different from the step function model assumed above can also be employed, such as impulse or pulsed waveforms. In settings involving very complicated geometries (e.g., urban canyons, buildings with offices and hallways, etc.), moving sources or sensors, or when more realistic propagation effects are taken into account (e.g., turbulence, eddys, inhomogeneous diffusivity, etc.), numerical methods are required to evaluate the response of the array to the chemical source. Details for these different modeling assumptions can be found in the references at the end of the chapter.

---

### 3.20.10 Conclusion

As we have seen above, the applications of array signal processing stretch from locating the sources of electrical energy from tiny neurons in the brain to astronomical objects millions of light-years away to submarines deep below the surface of the ocean. Remarkably, all of these applications share a very consistent underlying mathematical model that often allows techniques developed for one problem to apply to others in different fields. For this reason, we see similar methods appearing in journals related to radar, sonar, neurophysiology, acoustics, radio astronomy, medical imaging, seismology, and navigation, although explained in many cases with different terminology or emphases. Superficial differences in language aside, while the methods in the literature of these different areas are similar, they are not identical; each application has its own peculiarities that warrant special attention. Thus, in addition to showing what is common among the problems considered, our goal has also been to highlight the unique features of each application, and hence to provide motivation for the particular methodologies researchers and practitioners have adopted for these applications. Clearly, our discussion has only scratched the surface, and many details have been glossed over. It is our hope that we have piqued the reader's interest enough to pursue some of these details in the reference list (which is itself a small subset of what is available).

*Relevant Theory:* Signal Processing Theory, Machine Learning, and Statistical Signal Processing

See [Vol. 1, Chapter 2](#) Continuous-Time Signals and Systems

See [Vol. 1, Chapter 3](#) Discrete-Time Signals and Systems

See [Vol. 1, Chapter 4](#) Random Signals and Stochastic Processes

See [Vol. 1, Chapter 5](#) Sampling and Quantization

See [Vol. 1, Chapter 6](#) Digital Filter Structures and Their Implementation

See [Vol. 1, Chapter 7](#) Multirate Signal Processing for Software Radio Architectures

See [Vol. 1, Chapter 8](#) Modern Transform Design for Practical Audio/Image/Video Coding Applications

See [Vol. 1, Chapter 9](#) Discrete Multi-Scale Transforms in Signal Processing

See [Vol. 1, Chapter 10](#) Frames in Signal Processing

See [Vol. 1, Chapter 11](#) Parametric Estimation

See [Vol. 1, Chapter 12](#) Adaptive Filters

See [Vol. 1, Chapter 20](#) Clustering

See [Vol. 1, Chapter 21](#) Unsupervised Learning Algorithms

See [Vol. 1, Chapter 25](#) A Tutorial on Model Selection

See this Volume, [Chapter 2](#) Model Order Selection

See this Volume, [Chapter 7](#) Geolocation—Maps, Measurements, Models, and Methods

See this Volume, [Chapter 8](#) Performance Analysis and Bounds

---

## References and Further Reading

### Space-time adaptive processing

- [1] L. Brennan, F. Staudaher, Subclutter Visibility Demonstration, Adaptive Sensors, Inc., Technical Report RL-TR-92-21, 1992.

- [2] I.S. Reed, J.D. Mallett, L.E. Brennan, Rapid convergence rate in adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* AES-10 (1974) 853–862.
- [3] B.D. Carlson, Covariance matrix estimation errors and diagonal loading in adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 24 (3) (1988) 397–401.
- [4] I. Kirsteins, D. Tufts, Adaptive detection using low rank approximation to a data matrix, *IEEE Trans. Aerosp. Electron. Syst.* AES-30 (1) (1994) 55–67.
- [5] P. Parker, A. Swindlehurst, space-time autoregressive filtering for matched subspace STAP, *IEEE Trans. Aerosp. Electron. Syst.* AES-39 (2) (2003) 510–520.
- [6] J. Guerci, J. Goldstein, I. Reed, Optimal and adaptive reduced-rank STAP, *IEEE Trans. Aerosp. Electron. Syst.* 36 (2) (2000) 647–663.
- [7] J. Guerci, Space-Time Adaptive Processing for Radar, Artech House, 2003.
- [8] R. Klemm, Space-Time Adaptive Processing: Principles and Applications, IEE Press, 1998.
- [9] J. Ward, Space-Time Adaptive Processing for Airborne Radar, MIT Lincoln Labs, Technical Report TR-1015, 1994.

## MIMO radar

- [10] J. Li, P. Stoica (Eds.), MIMO Radar Signal Processing, John Wiley & Sons, Inc., Hoboken, NJ, 2009.
- [11] E. Fishler, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, R. Valenzuela, MIMO radar: an idea whose time has come, in: Proceedings of the IEEE Radar Conference, April 2004, pp. 71–78.
- [12] E. Fishler, A. Haimovich, R. Blum, L. Cimini, D. Chizhik, R. Valenzuela, Spatial diversity in radars—models and detection performance, *IEEE Trans. Signal Process.* 54 (3) (2006) 823–838.
- [13] I. Bekerman, J. Tabrikian, Target detection and localization using MIMO radars and sonars, *IEEE Trans. Signal Process.* 54 (2006) 3873–3883.
- [14] P. Stoica, J. Li, Y. Xie, On probing signal design for MIMO radar, *IEEE Trans. Signal Process.* 55 (8) (2007) 4151–4161.
- [15] J. Li, P. Stoica, MIMO radar with colocated antennas: review of some recent work, *IEEE Signal Process. Mag.* 24 (5) (2007) 106–114.
- [16] L. Xu, J. Li, P. Stoica, Target detection and parameter estimation for MIMO radar systems, *IEEE Trans. Aerosp. Electron. Syst.* 44 (3) (2008) 927–939.
- [17] A.H. Haimovich, R.S. Blum, L.J. Cimini, MIMO radar with widely separated antennas, *IEEE Signal Process. Mag.* 25 (1) (2008) 116–129.
- [18] H. He, P. Stoica, J. Li, Designing unimodular sequence sets with good correlations—including an application to MIMO radar, *IEEE Trans. Signal Process.* 57 (11) (2009) 4391–4405.
- [19] H. He, J. Li, P. Stoica, Waveform Design for Active Sensing Systems—A Computational Approach, Cambridge University Press, 2012.
- [20] W. Roberts, P. Stoica, J. Li, T. Yardibi, F.A. Sadjadi, Iterative adaptive approaches to MIMO radar imaging, *IEEE J. Sel. Top. Signal Process.* 4 (1) (2010) 5–20.
- [21] X. Tan, W. Roberts, J. Li, P. Stoica, Sparse learning via iterative minimization with application to MIMO radar imaging, *IEEE Trans. Signal Process.* (2011) 1088–1101.

## Radio astronomy

- [22] D. John, Kraus, Radio Astronomy, second ed., Cygnus-Quasar Books, Powell, Ohio, 1986.
- [23] A.R. Thompson, J.M. Moran, G.W. Swenson Jr., Interferometry and Synthesis in Radio Astronomy, second ed., Wiley-Interscience, New York, 2001.

- [24] B.D. Jeffs, K.F. Warnick, J. Landon, J. Waldron, J.R. Fisher D. Jones, R.D. Norrod, Signal processing for phased array feeds in radio astronomical telescopes, *IEEE J. Sel. Top. Signal Process.* 2 (5) (2008) 635–646.
- [25] P.J. Napier, A.R. Thompson, R.D. Ekers, The very large array: design and performance of a modern synthesis radio telescope, *Proc. IEEE* 71 (1983) 1295–1320.
- [26] R. Levanda, A. Leshem, Synthetic aperture radio telescopes, *IEEE Signal Process. Mag.* 27 (1) (2010) 14–29.
- [27] R.J. Cornwell, K. Golap, S. Bhatnaggar, The noncoplanar baselines effect in radio interferometry: the W-projection algorithm, *IEEE J. Sel. Top. Signal Process.* 2 (5) (2008) 647–657.
- [28] A. Leshem, A.-J. van der Veen, A.-J. Boonstra, Multichannel interference mitigation techniques in radio astronomy, *Astrophys. J. Suppl.* 131 (1) (2000) 355–374.
- [29] A. Leshem, A.-J. van der Veen, Radio-astronomical imaging in the presence of strong radio interference, *IEEE Trans. Inform. Theory* 46 (5) (2000) 1730–1747.
- [30] J.A. Högbom, Aperture synthesis with a nonregular distribution of interferometer baselines, *Astron. Astrophys. Suppl.* 15 (1974) 417–426.
- [31] F.R. Schwab, Relaxing the isoplanarity assumption in self-calibration: application to low-frequency radio interferometry, *Astron. J.* 131 (6) (1984) 646–659 (pt. F).
- [32] R.J. Cornwell, Multiscale CLEAN deconvolutions of radio synthesis images, *IEEE J. Sel. Top. Signal Process.* 2 (5) (2008) 793–801.
- [33] J. Landon, M. Elmer, D. Jones, A. Stemmons, B.D. Jeffs, K.F. Warnick, J.R. Fisher, R.D. Norrod, Phased array feed calibration, beamforming, and imaging, *Astron. J.* 139 (3) (2010) 1154–1167.
- [34] K.F. Warnick, M.A. Jensen, Effects of mutual coupling on interference mitigation with a focal plane array, *IEEE Trans. Antennas Propag.* 53 (8) (2005) 2490–2498.
- [35] M. Elmer, B.D. Jeffs, K.F. Warnick, J.R. Fisher, R. Norrod, Beamformer design methods for radio astronomical phased array feeds, *IEEE Trans. Antennas Propag.* 60 (2) (2012) 903–914.
- [36] S. van der Tol, B.D. Jeffs, A.-J. van der Veen, Self calibration for the LOFAR radio astronomical array, *IEEE Trans. Signal Process.* 55 (9) (2007) 4497–4510.
- [37] K.F. Warnick, B.D. Jeffs, Gain and aperture efficiency for a reflector antenna with an array feed, *IEEE Antennas Propag. Lett.* 5 (2006) 499–502.
- [38] A.R. Taylor, S.J. Gibson, M. Peracaula, P.G. Martin, T.L. Landecker, C.M. Brunt, P.E. Dewdney, S.M. Dougherty, A.D. Gray, L.A. Higgs, C.R. Derton, L.B.G. Knee, R. Kothes, C.R. Purton, B. Uyaniker, B.J. Wallace, A.G. Willis, D. Durand, The Canadian galactic plane survey, *Astron. J.* 125 (2003) 3145–3164.
- [39] Y. Bhattacharjee, Radio astronomers take arms against a sea of signals, *Science* 330 (6003) (2010) 444–445.
- [40] J.F. Bell, S.W. Ellingson, J. Bunton, Removal of the GLONASS C/A signal from OH spectral line observations using a parametric modeling technique, *Astrophys. J. Suppl.* 135 (2001) 87–93.
- [41] A.J. Poulsen, B.D. Jeffs, K.F. Warnick, J.R. Fisher, Programmable real-time cancellation of GLONASS interference with the Green Bank telescope, *Astron. J.* 130 (6) (2005) 2916–2927.
- [42] W. Dong, B.D. Jeffs, J.R. Fisher, Radar interference blanking in radio astronomy using a Kalman tracker, *Radio Sci.* 40 (5) (2005).
- [43] B.D. Jeffs, W. Lazarte, J.R. Fisher, Bayesian detection of radar interference in radio astronomy, *Radio Sci.* 41 (2006).
- [44] S.W. Ellingson, G.A. Hampson, Mitigation of radar interference in L-band radio astronomy, *Astrophys. J. Suppl.* 147 (2003) 167–176.
- [45] Q. Zhang, Y. Zheng, S.G. Wilson, J.R. Fisher, R. Bradley, Combating pulsed radar interference in radio astronomy, *Astron. J.* 126 (2003) 1588–1594.
- [46] Q. Zhang, Y. Zheng, S.G. Wilson, J.R. Fisher, R. Bradley, Excision of distance measuring equipment interference from radio astronomy signals, *Astron. J.* 129 (6) (2005) 2933–2939.
- [47] J.R. Fisher, Q. Zhang, S.G. Wilson, Y. Zheng, R. Bradley, Mitigation of pulsed interference to redshifted HI and OH observations between 960 and 1215 megahertz, *Astron. J.* 129 (6) (2005) 2940–2949.

- [48] P.A. Fridman, W.A. Baan, RFI mitigation methods in radio astronomy, *Astron. Astrophys.* 378 (2001) 327–344.
- [49] C. Barnbaum, R.F. Bradley, A new approach to interference excision in radio astronomy: real-time adaptive cancellation, *Astron. J.* 116 (1998) 2598–2614.
- [50] B.D. Jeffs, L. Li, K.F. Wanick, Auxiliary antenna assisted interference mitigation for radio astronomy arrays, *IEEE Trans. Signal Process.* 53 (2) (2005) 439–451.
- [51] S.W. Ellingson, G.A. Hampson, A subspace-tracking approach to interference nulling for phased array-based radio telescopes, *IEEE Trans. Antennas Propag.* 50 (1) (2002) 25–30.
- [52] C.K. Hansen, K.F. Warnick, B.D. Jeffs, R. Bradley, Interference mitigation using a focal plane array, *Radio Sci.* 40 (2005).
- [53] J.R. Nagel, K.F. Warnick, B.D. Jeffs, J.R. Fisher, R. Bradley, Experimental verification of radio frequency interference mitigation with a focal plane array feed, *Radio Sci.* 42 (2007).
- [54] J. Raza, A.-J. Boonstra, A.-J. van der Veen, Spatial filtering of RF interference in radio astronomy, *IEEE Signal Process. Lett.* 9 (2) (2002) 64–67.

## Positioning and navigation

- [55] E.D. Kaplan, C. Hegarty (Eds.), *Understanding GPS: Principles and Applications*, second ed., Artech House, 2005.
- [56] Y.-H. Chen, J.-C.J.D.S.D. Lorenzo, J. Seo, S. Lo, P. Enge, D.M. Akos, Real-time software receiver for GPS controlled reception pattern antenna array processing, in: ION GNSS Conference, 2010.
- [57] R.G. Lorenz, S.P. Boyd, Robust beamforming in GPS arrays, in: ION National Technical Meeting, 2002.
- [58] S. Backén, On dynamic array processing for GNSS software receivers, Lulea University of Technology, Ph.D. Dissertation, 2011.
- [59] G. Seco-Granados, J. Fernandez-Rubio, C. Fernandez-Prades, ML estimator and hybrid beamformer for multipath and interference mitigation in GNSS receivers, *IEEE Trans. Signal Process.* 53 (3) (2005) 1194–1208.
- [60] M. Amin, W. Sun, A novel interference suppression scheme for global navigation satellite systems using antenna array, *IEEE J. Sel. Areas Commun.* 23 (5) (2005) 999–1012.
- [61] D. Lu, Q. Feng, R. Wu, Survey on interference mitigation via adaptive array processing in GPS, *PIERS Online* 2 (4) (2006) 357–362.
- [62] R. Fante, J. Vaccaro, Wideband cancellation of interference in a GPS receive array, *IEEE Trans. Aerosp. Electron. Syst.* 36 (2) (2000) 549–564.
- [63] S.-J. Kim, R. Iltis, STAP for GPS receiver synchronization, *IEEE Trans. Aerosp. Electron. Syst.* 40 (1) (2004) 132–144.
- [64] M. Amin, L. Zhao, A. Lindsey, Subspace array processing for the suppression of FM jamming in GPS receivers, *IEEE Trans. Aerosp. Electron. Syst.* 40 (1) (2004) 80–92.
- [65] M.T. Brenneman, Y.T. Morton, Q. Zhou, GPS multipath detection with ANOVA for adaptive arrays, *IEEE Trans. Aerosp. Electron. Syst.* 46 (3) (2010) 1171–1184.
- [66] J. Soubielle, I. Frijalkow, P. Duvaut, A. Bibaut, GPS positioning in a multipath environment, *IEEE Trans. Signal Process.* 50 (1) (2002) 141–150.
- [67] A. Swindlehurst, Time delay and spatial signature estimation using known asynchronous signals, *IEEE Trans. Signal Process.* 46 (2) (1998) 449–462.
- [68] A. Jakobsson, A. Swindlehurst, P. Stoica, subspace-based estimation of time delays and Doppler shifts, *IEEE Trans. Signal Process.* 46 (9) (1998) 2472–2483.
- [69] M. Wax, A. Leshein, Joint estimation of time delays and directions of arrival of multiple reflections of a known signal, *IEEE Trans. Signal Process.* 45 (10) (1997) 2477–2484.

- [70] H. Amindavar, A. Reza, A new simultaneous estimation of directions of arrival and channel parameters in a multipath environment, *IEEE Trans. Signal Process.* 53 (2) (2005) 471–483.
- [71] M. Vanderveen, A.-J. van der Veen, A. Paulraj, Estimation of multipath parameters in wireless communications, *IEEE Trans. Signal Process.* 46 (3) (1998) 682–690.
- [72] F. Antreich, J. Nossek, W. Utschick, Maximum likelihood delay estimation in a navigation receiver for aeronautical applications, *Aero. Sci. Technol.* 12 (3) (2008) 256–267 (online). <<http://www.sciencedirect.com/science/article/pii/S1270963807000843>>.
- [73] G. Seco, A.L. Swindlehurst, D. Astély, Exploiting antenna arrays for synchronization, in: G.B. Giannakis, Y. Hua, P. Stoica, L. Tong (Eds.), *Signal Processing Advances in Wireless Communications, Trends in Single- and Multi-User Systems*, vol. II, Prentice-Hall, 2000, pp. 403–430 (Chapter 10).
- [74] F. Antreich, J.A. Nossek, G. Seco-Granados, A.L. Swindlehurst, The extended invariance principle for signal parameter estimation in an unknown spatial field, *IEEE Trans. Signal Process.* 59 (7) (2011) 3213–3225.
- [75] J.-C. Juang, G.-S. Huang, Development of GPS-based attitude determination algorithms, *IEEE Trans. Aerosp. Electron. Syst.* 33 (3) (1997) 968–976.
- [76] J.K. Ray, M.E. Cannon, P.C. Fenton, Mitigation of static carrier-phase multipath effects using multiple closely spaced antennas, *Navigation: J. Inst. Navigation (ION)* 46 (3) (1999) 193–201.
- [77] J. Ray, M. Cannon, P. Fenton, GPS code and carrier multipath mitigation using a multiantenna system, *IEEE Trans. Aerosp. Electron. Syst.* 37 (1) (2001) 183–195.

## Wireless communications

- [78] F. Khan, *LTE for 4G Mobile Broadband—Air Interface Technologies and Performance*, Cambridge University Press, 2009.
- [79] S. Sesia, I. Toufik, M. Baker (Eds.), *LTE—The UMTS Long Term Evolution: From Theory to Practice*, Wiley, 2009.
- [80] J. Lee, J.-K. Han, J.C. Zhang, MIMO technologies in 3GPP LTE and LTE-advanced, *EURASIP J. Wireless Commun. Network*. (2009) (online). <<http://dx.doi.org/10.1155/2009/302092>>.
- [81] A. Ghosh, R. Ratasuk, B. Mondal, N. Mangalvedhe, T. Thomas, LTE-advanced: next-generation wireless broadband technology (invited paper), *IEEE Wireless Commun.* 17 (3) (2010) 10–22.
- [82] J. Duplacy, B. Badic, R. Balraj, P.H. Rizwan Ghaffar, F. Kaltenberger, R. Knopp, I.Z. Kovács, H.T. Nguyen, D. Tandur, G. Vivier, MU-MIMO in LTE systems, *EURASIP J. Wireless Commun. Network*. (2011).
- [83] E. Dahlman, S. Parkvall, J. Skold, *4G: LTE/LTE-Advanced for Mobile Broadband*, Academic Press, 2011.
- [84] H. Taoka, S. Nagata, K. Takeda, Y. Kakishima, X. She, K. Kusume, MIMO and CoMP in LTE-advanced, *NTT DOCOMO Techn. J.* 12 (2) (2010) 20–28.
- [85] E. Hossain, D.I. Kim, V.K. Bhargava, *Cooperative Cellular Wireless Networks*, Cambridge University Press, 2011.
- [86] Q. Li, X. Lin, J. Zhang, W. Roh, Advancement of MIMO technology in WiMAX: from IEEE 802.16d/e/j to 802.16m, *IEEE Commun. Mag.* 47 (6) (2009) 100–107.
- [87] S. Ahmadi, *Mobile WiMAX: A Systems Approach to Understanding IEEE 802.16m Radio Access Technology*, Academic Press, 2010.
- [88] Q. Li, G. Li, W. Lee, M. il Lee, D. Mazzarese, B. Clerckx, Z. Li, MIMO techniques in WiMAX and LTE: a feature overview, *IEEE Commun. Mag.* 48 (5) (2010) 86–92.
- [89] IEEE Std 802.16, IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems Amendment 3: Advanced Air Interface, IEEE Std 802.16m-2011(Amendment to IEEE Std 802.16-2009), 2011.
- [90] D. Halperin, W. Hu, A. Sheth, D. Wetherall, 802.11 with multiple antennas for dummies, *SIGCOMM Comput. Commun. Rev.* 40 (2010) 19–25 (online). <<http://doi.acm.org/10.1145/1672308.1672313>>.

- [91] E. Perahia, R. Stacey, Next Generation Wireless LANs: Throughput, Robustness, and Reliability in 802.11n, Cambridge University Press, 2008.
- [92] IEEE802.11n, IEEE Standard for Information technology—Telecommunications information exchange between systems—Local and metropolitan area networks—Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput, IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009), 2009.

## Biomedical applications

- [93] T. Szabo, Diagnostic Ultrasound Imaging: Inside Out, Elsevier Academic Press, 2004.
- [94] B. Steinberg, Digital beamforming in ultrasound, *IEEE Trans. Ultrasonics, Ferr. Freq. Cont.* 39 (6) (1992) 716–721.
- [95] J. Lu, H. Zou, J. Greenleaf, Biomedical ultrasound beamforming, *Ultrasound Med. Biol.* 20 (5) (1994) 403–428.
- [96] J. Quistgaard, Signal acquisition and processing in medical diagnostic ultrasound, *IEEE Signal Process. Mag.* 14 (1) (1997) 67–74.
- [97] Z. Wang, J. Li, R. Wu, Time-delay- and time-reversal-based robust Capon beamformers for ultrasound imaging, *IEEE Trans. Med. Imag.* 24 (2005) 1308–1322.
- [98] S. Sanei, J. Chambers, EEG Signal Processing, Wiley & Sons, Ltd., West Sussex, England, 2007.
- [99] P. Nunez, R. Srinivasan, Electric Fields of the Brain: The Neurophysics of EEG, Oxford University Press, 2006.
- [100] T. Handy (Ed.), Brain Signal Analysis, Advances in Neuroelectric and Neuromagnetic Methods, MIT Press, 2009.
- [101] Y. Salu, L.G. Cohen, D. Rose, S. Sato, C. Kufta, M. Hallett, An improved method for localizing electric brain dipoles, *IEEE Trans. Biomed. Eng.* 37 (7) (1990) 699–705.
- [102] J.C. de Munck, The estimation of time varying dipoles on the basis of evoked potentials, *Electroencephalogr. Clin. Neurophysiol.* 77 (1990) 156–160.
- [103] J. Mosher, P. Lewis, R. Leahy, Multiple dipole modeling and localization from spatio-temporal MEG data, *IEEE Trans. Biomed. Eng.* 39 (6) (1992) 541–557.
- [104] J.W. Phillips, R.M. Leahy, J.C. Mosher, MEG-based imaging of focal neuronal current sources, *IEEE Trans. Med. Imag.* 16 (3) (1997) 338–348.
- [105] B. van Veen, W. van Drongelen, M. Yuchtman, A. Suzuki, Localization of brain electrical activity via linearly constrained minimum variance spatial filtering, *IEEE Trans. Biomed. Eng.* 44 (9) (1997) 867–880.
- [106] B. Lutkenhoner, Dipole source localization by means of maximum likelihood estimation: theory and simulations, *Electroencephalogr. Clin. Neurophysiol.* 106 (1998) 314–321.
- [107] B. Cuffin, EEG dipole source localization, *IEEE Eng. Med. Bio. Mag.* 17 (5) (1998) 118–122.
- [108] Z. Koles, Trends in EEG source localizatoin, *Electroencephalogr. Clin. Neurophysiol.* 106 (1998) 127–137.
- [109] L. Zhukov, D. Weinstein, C. Johnson, Independent component analysis for EEG source localization, *IEEE Eng. Med. Bio. Mag.* 19 (3) (2000) 87–96.
- [110] D. Yao, Electric potential produced by a dipole in a homogeneous conducting sphere, *IEEE Trans. Biomed. Eng.* 47 (7) (2000) 964–966.
- [111] S. Baillet, J.C. Mosher, R.M. Leahy, Electromagnetic brain mapping, *IEEE Signal Process. Mag.* 18 (6) (2001) 14–30.

- [112] J.C. de Munck, H.M. Huijzena, L.J. Waldorp, R.A. Heethaar, Estimating stationary dipoles from MEG/EEG data contaminated with spatially and temporally correlated background noise, *IEEE Trans. Signal Process.* 50 (7) (2002) 1565–1572.
- [113] C. Michela, M. Murraya, G. Lantza, S. Gonzalez, L. Spinellib, R.G. de Peralta, EEG source imaging, *Clin. Neurophysiol.* 115 (2004) 2195–2222.
- [114] K. Sekihara, K. Hild, S. Nagarajan, A novel adaptive beamformer for MEG source reconstruction effective when large background brain activities exist, *IEEE Trans. Biomed. Eng.* 53 (9) (2006) 1755–1764.
- [115] T. Ferree, P. Nunez, Primer on electroencephalography for functional connectivity, in: V. Jirsa, A. McIntosh (Eds.), *Handbook of Brain Connectivity*, Springer-Verlag, 2007, pp. 169–200.
- [116] K. Sekihara, K. Hild, S.S. Dalal, S. Nagarajan, Performance of prewhitening beamforming in MEG dual experimental conditions, *IEEE Trans. Biomed. Eng.* 55 (3) (2008) 1112–1121.
- [117] S.C. Wu, A.L. Swindlehurst, P.T. Wang, Z. Nenadic, Projection vs. prewhitening for EEG interference suppression, *IEEE Trans. Biomed. Eng.* 59 (5) (2012) 1329–1338.
- [118] S.C. Wu, A.L. Swindlehurst, P.T. Wang, Z. Nenadic, Efficient dipole parameter estimation in EEG systems with near-ML performance, *IEEE Trans. Biomed. Eng.* 59 (5) (2012) 1339–1348.
- [119] S. Gibson, J.W. Judy, D. Markovic, Spike sorting: the first step in decoding the brain, *IEEE Signal Process. Mag.* 29 (1) (2012) 124–143.
- [120] C.M. Gray, P.E. Maldonado, M. Wilson, B. McNaughton, Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex, *J. Neurosci. Methods* 63 (1995) 43–54.
- [121] S. Takahashi, Y. Sakurai, M. Tsukada, Y. Anzai, Classification of neuronal activities from tetrode recordings using independent component analysis, *Neurocomputing* 49 (2002) 289–298.
- [122] S. Takahashi, Y. Anzai, Y. Sakurai, A new approach to spike sorting for multi-neuronal activities recorded with a tetrode: how ICA can be practical, *Neurosci. Res.* 46 (2003) 265–272.
- [123] M.I. Chelaru, M.S. Jog, Spike source localization with tetrodes, *J. Neurosci. Meth.* 142 (2005) 305–315.
- [124] S. Micera, L. Citi, J. Rigosa, J. Carpaneto, S. Raspopovic, G.D. Pino, L. Rossini, K. Yoshida, L. Denaro, P. Dario, P.M. Rossini, Decoding information from neural signals recorded using intraneuronal electrodes: toward the development of a neurocontrolled hand prosthesis, *Proc. IEEE* 98 (3) (2010) 407–417.

## Sonar

- [125] W.C. Knight, R.G. Pridham, S.M. Kay, Digital signal processing for sonar, *IEEE Proc.* 69 (11) (1981) 1451–1506.
- [126] R.J. Urick, *Principles of Underwater Sound*, third ed., John Wiley & Sons, West Sussex, England, 1983.
- [127] R.G. Fizell, S.C. Wales, Source localization in range and depth in an arctic environment, *J. Acoust. Soc. Am. Suppl.* 78 (1985) p. S57.
- [128] N.L. Owsley, *Array Signal Processing*, Prentice Hall, 1985.
- [129] L. Brekhovskikh, Y. Lysanov, *Fundamentals of Ocean Acoustics*, Springer-Verlag, New York, 1991.
- [130] Special issue on detection and estimation in matched-field processing, *IEEE J. Ocean. Eng.* 18 (3) (1993) 156–270.
- [131] A.B. Baggeroer, W.A. Kuperman, P.N. Mikhalevsky, An overview of matched field methods in ocean acoustics, *IEEE J. Ocean. Eng.* 18 (4) (1993) 401–424.
- [132] A. Tolstoy, *Matched Field Processing for Underwater Acoustics*, World Scientific, Singapore, 1993.
- [133] A. Nehorai, E. Paldi, Acoustic vector-sensor array processing, *IEEE Trans. Signal Process.* 42 (9) (1994) 2481–2491.
- [134] D.W. Tufts, J.P. Iannello, I. Lourtie, J.C. Preisig, J.M.F. Moura, The past, present, and future of underwater acoustic signal processing, *IEEE Signal Process. Mag.* 15 (4) (1998) 21–51.

- [135] M. Hawkes, A. Nehorai, Effects of sensor placement on acoustic vector-sensor array performance, *IEEE J. Oceanic Eng.* 24 (1999) 33–40.
- [136] A.D. Waite, *Sonar for Practising Engineers*, third ed., West Sussex, England, 2002.
- [137] M. Hawkes, A. Nehorai, Wideband source localization using a distributed acoustic vector-sensor array, *IEEE Trans. Signal Process.* 51 (6) (2003) 1479–1491.
- [138] W. Xu, A.B. Baggeroer, C.D. Richmond, Bayesian bounds for matched-field parameter estimation, *IEEE Trans. Signal Process.* 52 (12) (2004) 3293–3305.

## Microphone arrays

- [139] J. Flanagan, J. Johnston, R. Zahn, G. Elko, Computer-steered microphone arrays for sound transduction in large rooms, *J. Acoust. Soc. Am.* 78 (5) (1985) 1508–1518.
- [140] Y. Kaneda, J. Ohga, Adaptive microphone-array system for noise reduction, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-34 (6) (1986) 1391–1400.
- [141] Y. Grenier, A microphone array for car environments, *Speech Commun.* 12 (1993) 25–39.
- [142] B. Ferguson, B. Quinn, Application of the short-time Fourier transform and the Wigner-Ville distribution to the acoustic localization of aircraft, *J. Acoust. Soc. Am.* 96 (1994) 821–827.
- [143] M. Hoffman, K. Buckley, Robust time-domain processing of broadband microphone array data, *IEEE Trans. Speech Audio Process.* 3 (3) (1995) 193–203.
- [144] S. Fischer, K. Simmer, Beamforming microphone arrays for speech acquisition in noisy environments, *Speech Commun.* 20 (1996) 215–227.
- [145] G. Elko, Microphone array systems for hands-free telecommunication, *Speech Commun.* 20 (1996) 229–240.
- [146] S. Affes, Y. Grenier, A signal subspace tracking algorithm for microphone array processing of speech, *IEEE Trans. Speech Audio Process.* 5 (5) (1997) 425–437.
- [147] B. Ferguson, Time-delay estimation techniques applied to the acoustic detection of jet aircraft transits, *J. Acoust. Soc. Am.* 106 (1) (1999) 255–264.
- [148] M. Dahl, I. Claesson, Acoustic noise and echo canceling with microphone array, *IEEE Trans. Veh. Tech.* 48 (5) (1999) 1518–1526.
- [149] J. Benesty, Adaptive eigenvalue decomposition algorithm for passive acoustic source localization, *J. Acoust. Soc. Am.* 107 (1) (2000) 384–391.
- [150] Y. Huang, J. Benesty, G. Elko, M. Mersereau, Real-time passive source localization: a practical linear-correction least-squares approach, *IEEE Trans. Speech Audio Process.* 9 (8) (2001) 943–956.
- [151] J. Chen, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. Hudson, K. Yao, D. Estrin, Coherent acoustic array processing and localization on wireless sensor networks, *Proc. IEEE* 91 (8) (2003) 1154–1162.
- [152] T. Gustafsson, B. Rao, M. Trivedi, Source localization in reverberant environments: modeling and statistical analysis, *IEEE Trans. Speech Audio Process.* 11 (6) (2003) 791–803.
- [153] R. Kozick, B. Sadler, Source localization with distributed sensor arrays and partial spatial coherence, *IEEE Trans. Signal Process.* 52 (3) (2004) 601–616.
- [154] Z. Li, R. Duraiswami, Flexible and optimal design of spherical microphone arrays for beamforming, *IEEE Trans. Audio Speech Lang. Process.* 15 (2) (2007) 702–714.
- [155] J. Benesty, J. Chen, Y. Huang, J. Dmochowski, On microphone-array beamforming from a MIMO acoustic signal processing perspective, *IEEE Trans. Audio Speech Lang. Process.* 15 (3) (2007) 1053–1065.
- [156] X. Zhao, Z. Ou, Closely coupled array processing and model-based compensation for microphone array speech recognition, *IEEE Trans. Audio Speech Lang. Process.* 15 (3) (2007) 1114–1122.
- [157] J. Benesty, J. Chen, Y. Huang, *Microphone Array Signal Processing*, Springer Verlag, 2008.
- [158] M. Brandstein, D. Ward (Eds.), *Microphone Arrays—Signal Processing Techniques and Applications*, Springer Verlag, 2010.

## Chemical sensor arrays

- [159] A. Nehorai, B. Porat, E. Paldi, Detection and localization of vapor-emitting sources, *IEEE Trans. Signal Process.* 43 (1) (1995) 243–253.
- [160] A. Gershman, V. Turchin, Nonwave field processing using sensor array approach, *Signal Process.* 44 (1995) 197–210.
- [161] B. Porat, A. Nehorai, Localizing vapor-emitting sources by moving sensors, *IEEE Trans. Signal Process.* 44 (4) (1996) 1018–1021.
- [162] A. Jerémic, A. Nehorai, Design of chemical sensor arrays for monitoring disposal sites on the ocean floor, *IEEE J. Ocean. Eng.* 23 (4) (1998) 334–343.
- [163] Y. Nievergelt, Solution to an inverse problem in diffusion, *SIAM Rev.* 40 (1) (1998) 74–80.
- [164] A. Jerémic, A. Nehorai, Landmine detection and localization using chemical sensor array processing, *IEEE Trans. Signal Process.* 48 (5) (2000) 1295–1305.
- [165] J. Matthes, L. Gröll, H. Keller, Source localization based on pointwise concentration measurements, *Sensor. Actuat. A: Phys.* 115 (2004) 32–37.
- [166] J. Matthes, L. Gröll, H. Keller, Source localization by spatially distributed electronic noses for advection and diffusion, *IEEE Trans. Signal Process.* 53 (5) (2005) 1711–1719.
- [167] T. Zhao, A. Nehorai, Detecting and estimating biochemical dispersion of a moving source in a semi-infinite medium, *IEEE Trans. Signal Process.* 54 (6) (2006) 2213–2225.
- [168] S. Vijayakumaran, Y. Levinbook, T. Wong, Maximum likelihood localization of a diffusive point source using binary observations, *IEEE Trans. Signal Process.* 55 (2) (2007) 665–676.
- [169] M. Ortner, A. Nehorai, A. Jerémic, Biochemical transport modeling and Bayesian source estimation in realistic environments, *IEEE Trans. Signal Process.* 55 (6) (2007) 2520–2532.

# Index

## A

- Acoustic signal processing  
chemical sensor arrays, 943  
microphone arrays, 938, 940
- Adaptive ATC strategy, 394
- Adaptive beamforming algorithms, 482, 508  
algorithms employing  
spatial reference, 900  
temporal reference, 901  
basic principles, 508  
blind algorithms, 902  
general-rank source, 513  
gradient adaptive beamforming algorithms, 513  
hybrid beamformers, 901  
MVDR beamforming with data covariance matrix, 512  
optimal SINR, 512  
projection adaptive beamforming methods, 515  
reduced complexity approaches to adaptive  
beamforming, 516  
sample matrix inversion adaptive beamformer, 514  
wideband adaptive beamforming, 519
- Adaptive broadband beamforming, 584
- Adaptive combination weights, 401, 412
- Adaptive CTA strategy, 393
- Adaptive diffusion strategies with smoothing mechanisms, 415
- Adapt-then-combine (ATC) diffusion strategy, 353
- Aeroacoustic source localization, 940–941
- Affine transforms, 53–54
- AIC. *See* Akaike information criterion (AIC)
- Airborne fast vehicles, 278
- Airborne slow vehicles, 280
- Akaike information criterion (AIC), 12, 17, 635
- Algorithms using tensor-based subspace estimates, 698  
R-D NC standard tensor-ESPRIT, 700  
R-D NC unitary tensor-ESPRIT, 702–703  
R-D standard tensor-ESPRIT, 698–699  
R-D unitary tensor-ESPRIT, 699
- Ambiguity function, 77
- Angle-Doppler spectra, 870
- Aperture theory, 556
- Array aperture, 819–820
- Array-based parameter estimators, 904–905  
GNSS-specific signal models, 906  
structured spatial signatures and spatially white noise, 905  
structured spatial signatures and unknown spatial  
correlation, 905  
unstructured spatial signatures and spatially white noise, 905
- unstructured spatial signatures and unknown spatial  
correlation, 905
- Array calibration, 829–830  
robust beamforming using, 848
- Array geometries, 557
- Array interpolation technique, 836
- Array nonidealities, 825  
array elements' beampatterns and positions, 826  
cross-polarization effects, 827  
mutual coupling, 825  
narrowband signal model, 828  
nonlinear elements, effects of, 829  
receiver front-end architectures, 828
- Array processing  
beam forming and signal detection, 480  
adaptive beamforming, 482  
signal detection, 485  
spatial filter design, 480
- direction-of-arrival estimation, 486  
beamforming methods, 487  
modeling errors and array calibration, 492  
parametric methods, 489  
subspace methods, 488  
geometric data model, 465  
ideal data model, 466  
non-ideal data models, 470  
wave propagation, 465
- non-Coherent array applications, 493  
microwave and ultrasound imaging, 497  
sensor networks, source localization in, 497  
spread sources, 493  
time series modeling, 495
- spatial filtering and beam patterns, 471  
one-dimensional arrays, 472  
spatial filtering, 471  
two-dimensional arrays, 475  
wideband array response, 477
- Array signal processing  
adaptive and robust beamforming, 458  
applications of, 460  
array processing, 458  
azimuth, elevation, and polarization estimation, 850, 852  
biomedical applications, 917  
broadband beamforming and optimization, 458  
classification of techniques, 821  
DOA estimation, 846–848  
methods and algorithms, 458

- Array signal processing (*Continued*)
- of nonstationary signals, 459
  - performance bounds and statistical analysis of, 459
  - examples, 846
  - face of non idealities, 460
  - history, 457
  - ideal array signal models, 821
  - multi-input multi-output (MIMO) radar, 870–871
  - nonstationary signals, 459
  - outlook, 461
  - polynomial rooting techniques, 849
  - positioning and navigation, 893–894
  - radar applications, 860–862
  - radio astronomy, 875–876
  - robust beamforming using array calibration, 848
  - robust methods, 842–843
  - sonar, 928–929
  - source localization and tracking, 460
  - special array structures, subspace methods and exploitation of, 459
  - wireless communications, 907
- Array steering vectors, 820
- Astronomical phased array feeds, 885
- beamformer calculation, 888
  - calibration, 888
  - radio camera results, 889
  - signal model, 886–888
- Asymptotic analysis
- and central limit theorem, 313
  - and parametric models, 315
- Asymptotic distribution
- estimated DOA, 736
    - beamforming-based algorithms, 737–738
    - high-order algorithms, 750
    - maximum likelihood algorithms, 738
    - robustness of algorithms, 747
    - second-order algorithms, 743–744
    - subspace-based algorithms, 745
  - of statistics, 724
- Asymptotic regime, 194
- Attenuation, 803
- Autoregressive (AR) modeling, 328
- cooperative adaptation through diffusion, 334
  - linear model, 328
  - non-cooperative adaptive solution, 331
  - non-cooperative mean-square-error solution, 330
- B**
- Bandwidth, 876
- Bayesian computational methods, 5
- computational methods, 161
  - expectation-maximization (EM), for MAP estimation, 162
- Markov chain Monte Carlo (MCMC), 163
- parameter estimation, 143
- Bayesian inference, 147
  - Bayesian model averaging, 161
  - linear Gaussian model, 144
  - maximum likelihood (ML) estimation, 146
  - model uncertainty and Bayesian decision theory, 158
  - model uncertainty, structures for, 161
  - particle filtering and auxiliary sampling, 169
    - marginalized particle filters, 171
    - particle filters, 177
  - probability densities and integrals, 178
    - gamma density, 180
    - inverse Wishart distribution, 182
    - inverted-gamma distribution, 180
    - multivariate Gaussian, 178
    - normal-inverted-gamma distribution, 181
    - univariate Gaussian, 178
    - Wishart distribution, 181
  - state-space models and sequential inference, 164
    - linear Gaussian state-space models, 164
    - prediction error decomposition, 166
    - sequential Monte Carlo (SMC), 167
- Bayesian formulation, 190, 210
- Bayesian i.i.d. setting, 217
- Bayesian inference, 147
- covariance matrices, priors on, 157
  - G-prior, 157
  - hyperparameters and marginalization, of unwanted parameters, 152
  - linear Gaussian model, hyperparameters for, 153
  - linear Gaussian model, parameters in, 151
  - Marginal likelihood, 152
  - normal-inverted-gamma prior, 154
    - posterior inference and Bayesian cost functions, 149
  - Bayesian information criterion (BIC), 14
  - Bayesian model averaging, 161
  - Bayesian quickest change detection, 217
  - Bayesian source localization, 808–809
  - Beamformer architecture, 887
  - Beam forming and signal detection, 480
  - Beamforming-based algorithms, 737–738
  - Beamforming methods, 487
  - Beamforming process, 860, 898
    - adaptive beamforming algorithms, 899
    - deterministic, 903
  - Beamspace processing, 638
  - BIC. *See* Bayesian information criterion (BIC)
  - Binary hypothesis testing problem, 188
  - Binary RSS measurements, 289
  - Biomedical applications, array signal processing, 917
    - electroencephalography (EEG), 918
    - magnetoencephalography (MEG) signal processing, 918

**Biomedical applications (*Continued*)**

- multi-sensor extracellular probes, 923, 925–926
- ultrasound imaging, 918
- Biomedical signal analysis**, 137
- Blackman window, 32
- Block maximum norm, 435
- Bootstrap methods, 12
- Bootstrapping, 14
- Born-Jordan distribution, 83
- Broadband beamformer, 574
- Broadband beamforming and optimization**
  - adaptive broadband beamforming, 584
  - common signal modeling, 584
  - frequency domain, generalized sidelobe canceler in, 589
  - frequency domain Wiener filter, 591
  - generalized sidelobe canceler (GSC), 587
  - LCMV in frequency domain, 586
  - linearly constrained minimum variance (LCMV) beamforming, 584
  - Wiener filter, 590
- design in element space, 558
- broadband beamformer, 574
- Chebyshev design, 563
- design examples, 568
- model and robust formulation, 563
- robust Chebyshev design, 567
- robust total least squares design, 567
- robust WLS design, 566
- steerable broadband beamformer, 571
- total least squares design and Eigen-filters, 562
- weighted least square (WLS) design, 561
- design using wave equation, 574
- design examples, 582
- spherical broadband beamformer, 581
- wave equation, 579
- environment and channel modeling**, 556
  - aperture theory, 556
  - array geometries, 557
- examples for optimal beamformers, 593
- optimal near-field signal-to-noise plus interference beamformer (SNIB), 591
- frequency domain formulation, 593
- time domain formulation, 591

Butterworth distribution, 83

## C

- Capon beamformer. *See* Minimum variance distortionless response (MVDR) beamformer
- Car engine signal analysis, 137
- Cellular phones, 285
  - binary RSS measurements, 289
  - continuous RSS measurements, 286

- Chair-Varshney fusion rule, 200
- Channel aware distributed detection, 197
- Chebyshev design, 563
- Chemical sensor arrays, 943
- Chirplet transform, 53–54
- Closed-loop multiplexing schemes, 910–911
- Closely spaced sources, resolution of, 755
  - CRB, angular resolution limit, 757–758
  - detection theory, angular resolution limit, 758–759
  - mean null spectra, angular resolution limit, 755
- Cognitive radio, 250
- Cohen class of distributions, 80
  - auto-terms form, 87
  - reduced interference distributions, 83
- Coherent processing interval (CPI), 865
- Collaborative spectral sensing, 341
- Combination weights, 396
- Combine-then-adapt (CTA) diffusion strategy, 355
- Common signal modeling, 584
- Complex argument distribution, 114
- Computational methods, 161
- Computer network security, 250
- Conditional independence assumption, 190
  - asymptotic regime, 194
  - Bayesian formulation, 190
  - decision fusion problem, 192
  - Neyman-Pearson formulation, 191
- Consensus recursion, 442
- Consensus strategies, comparison with, 442
- Constant combination weights, 397
- Constrained Cramér-Rao bound (CCRB), 308
- Constrained maximum-likelihood estimation (CMLE), 310
- Continuous RSS measurements, 286
- Controlled Reception Pattern Antennas (CRPAs), 898
- Conventional beamformer, 605
- Convergence behavior, 368
- Convergence in mean, 409
- Cooperative adaptation through diffusion, 334
- Coordinated Multipoint transmission (CoMP), 913
- Copula theory, 199
- Covariance matching estimation methods, 626
- Covariance matrices, priors on, 157
- CPI. *See* Coherent processing interval
- Cramér-Rao bounds (CRB), 9, 303, 729
  - bias-informed, 301
  - Gaussian deterministic case, 732
  - Gaussian stochastic case, 730
  - general CRB expression, 301
  - non Gaussian case, 733–734
  - on parameter estimation, 299
  - properties, 302
  - transformations, 301
- Cramer-Rao Lower Bound (CRLB), 263, 824, 837

CRB. *See* Cramér-Rao bounds (CRB)  
 CRLB. *See* Cramer-Rao Lower Bound (CRLB)  
 Cross-polarization discrimination (XPD), 827  
 Cross-polarization effects, 827  
 Cross-validation (CV), 12  
 CRPAs. *See* Controlled Reception Pattern Antennas  
 CuSum procedure, 211  
 CV. *See* Cross-validation (CV)

**D**

Data and beamforming models, 504  
 narrowband case, 505  
     general-rank source, 506  
     point source, 505  
     wideband case, 507  
 Data association problem, 802  
 Data-driven techniques, 834  
     array calibration matrix, local interpolation of, 834–835  
     array interpolation technique, 836  
     manifold separation technique, 837, 842  
     wavefield modeling principle, 837, 842  
 Data-efficient quickest change detection, 244  
 Data model, 374, 654  
     general data model, 655, 657  
     non-circular data, 665, 667  
     notation, 654  
     special array structures, 657  
 Dead-reckoning model, 259, 272  
     dynamical models, 273  
     inertial models, 273  
     marginalization of speed, 273  
     odometric models, 272  
 Decision fusion problem, 192  
 Degrees of freedom (DOF), 16–17  
 DE-Shiryayev algorithm, 245  
 Design using wave equation, 574  
 Diagonally loaded SMI beamformer, 522  
 Diffusion adaptation over networks, 5  
     adaptive diffusion strategies, 359, 374  
     convergence in mean, 379  
     data model, 374  
     error recursions, 377  
     mean-square performance, of individual nodes, 387  
     mean-square stability, 381  
     network mean-square performance, 386  
     performance measures, 375  
     transient mean-square performance, 390  
     uniform data profile, 389  
 Block maximum norm, 435  
 combination weights, 396  
     adaptive combination weights, 401  
     constant combination weights, 397  
     optimizing combination weights, 398  
 consensus recursion, 442  
 consensus strategies, comparison with, 442  
 cooperative strategies, 391  
     ATC and CTA strategies, 392  
     information exchange, 393  
     non-cooperative strategy, 394  
 distributed optimization via diffusion strategies, 345  
     adapt-then-combine (ATC) diffusion strategy, 353  
     combine-then-adapt (CTA) diffusion strategy, 355  
     global cost to neighborhood costs, 347  
     properties of diffusion strategies, 357  
     steepest-descent iterations, 351  
 error recursion, 443  
 extensions and variations, 414  
     adaptive diffusion strategies with smoothing  
         mechanisms, 415  
     diffusion distributed optimization, 426  
     diffusion Kalman filtering, 423  
     diffusion recursive least-squares, 419  
     graph laplacian and network connectivity, 430  
     mean-square-error estimation, 327  
     autoregressive modeling, 328  
     collaborative spectral sensing, 341  
     tapped-delay-line models, 334  
     target localization, 336  
 motivation, 323  
     cooperation among agents, 326  
     networks and neighborhoods, 324  
     notation, 326  
 noisy information exchanges, 403  
     adaptive combination weights, 412  
     convergence in mean, 409  
     error recursion, 405  
     mean-square convergence, 410  
     noise sources over exchange links, 404  
 properties of Kronecker products, 430  
 steepest-descent diffusion strategies, 364  
     convergence behavior, 368  
     error recursions, 366  
     general diffusion model, 364  
     stochastic matrices, 433  
 Diffusion distributed optimization, 426  
     noiseless updates, 427  
     updates with gradient noise, 428  
 Diffusion Kalman filtering, 423  
 Diffusion recursive least-squares, 419  
 Direction of arrival (DOA), 823–824, 832  
     estimation techniques, 463, 486, 779  
     effect of cross-terms, 784  
     signal stationarization, 787–788

Direction of arrival (*Continued*)

- spatial joint-variable domain distributions, 788
- time-frequency maximum likelihood method, 782–784
- time-frequency MUSIC, 780
- wideband nonstationary signals, 788–789
- Discrete pseudo Wigner distribution, 75
- Discrete S-method, 91
- Discrete STFT, signal reconstruction form, 44
- Discrete wavelet transform, 927
- Distributed sensor systems, 247
- Distributed signal detection, 4
  - with dependent observations, 198
  - with independent observations
    - channel aware distributed detection, 197
    - conditional independence assumption, 190
    - energy efficient distributed detection, 197
    - multi-objective optimization, 197
    - network topologies, 194
    - nonparametric rules, in distributed detection, 196
- DOA. *See* Direction of arrival (DOA)
- DOA estimation methods and algorithms
  - background, 599
  - beamforming methods, 604
    - conventional beamformer, 605
    - minimum variance distortionless response (MVDR) beamformer, 606
    - numerical examples, 609
    - sparse data representation based approach, 607
  - beamspace processing, 638
  - data model, 600
    - frequency domain description, 601
    - uniqueness, 604
    - wave propagation, 600
  - distributed sources, 639
  - parametric methods, 617
    - covariance matching estimation methods, 626
    - implementation, 620
    - maximum likelihood approach, 618
    - numerical examples, 628
    - performance bound, 627
    - subspace fitting methods, 625
  - polarization sensitivity, 640
  - signal detection, 634
    - additional issues, 636
    - nonparametric methods, 634
    - parametric methods, 636
  - signals with known structures, 637
  - spatially correlated noise fields, 638
  - subspace methods, 610
    - estimation of signal parameters via rotational invariance techniques (ESPRIT) algorithm, 613
  - MUSIC algorithm, 612
  - numerical examples, 616
- signal coherence, 614
- tracking, 637
- wideband DOA estimation, 631
  - coherent signal subspace methods, 633
  - wideband maximum likelihood estimation, 632
- DOF. *See* Degrees of freedom (DOF)
- Doob's optional stopping theorem, 213
- Doppler velocity log, 282
- Doubly constrained robust adaptive beamforming, 539
- Dynamical models, 273

**E**

- EADF. *See* Effective aperture distribution function
- ECoG measurements
  - unified dipole model, 919
- EEG and MEG signal processing, 918
  - interference mitigation, 922–923
  - unified dipole model, 919
- Effective aperture distribution function (EADF), 838–839
- Eigen-filters, 562
- Eigenspace-based beamformer, 537
- Eigenvalue beamforming using multi-rank MVDR beamformer, 541
- EKF. *See* Extended Kalman filter
- Electroencephalography (EEG) signal processing, 918
- Energy efficient distributed detection, 197
- Environment and channel modeling, 556
- Error recursion, 405, 443
  - comparison with diffusion strategies, 447
  - convergence conditions, 443
  - rate of convergence, 445
- Error recursions, 366, 377
- ESPRIT algorithm. *See* Estimation of signal parameters via rotational invariance techniques (ESPRIT) algorithm
- Estimation of signal parameters via rotational invariance techniques (ESPRIT) algorithm, 613, 678
- Expectation-maximization (EM) algorithm, 5
  - for MAP estimation, 162
- Extended Invariance Principle (EXIP), 905
- Extended Kalman filter (EKF), 264, 815

**F**

- False alarm, 209–211
- False discovery rate (FDR), 196
- Far-field assumption, 601
- Fast Fourier Transform (FFT), 3, 519
- FDR. *See* False discovery rate (FDR)
- Filter bank STFT implementation, 38
- Finite impulse response digital filter, 803
- Finite Impulse Response (FIR) filtering, 463

Fourier domain root-MUSIC, 689–690

Fourier transform, 315, 602

Fractional Fourier Transform, 769–772

Frequency domain description, 601

general model, 602

narrow band data, 603

Frequency domain formulation, 593

Frequency domain, generalized sidelobe canceler in, 589

Frequency domain Wiener filter, 591

Frequency Shift Transmit Diversity (FSTD), 908

FSTD. *See* Frequency Shift Transmit Diversity

Functional analysis, 723

## G

Gabor transform, 45

Gamma density, 180

Gaussian case, 311

Gaussian deterministic case, 732

Gaussian model, 5

Gaussian stochastic case, 730

Gaussian window, 32

General asymptotic Bayesian theory, 221

General asymptotic minimax theory, 237

General diffusion model, 364

Generalized CuSum algorithm, 238

Generalized ESPRIT (GESPRIT), 681–682

Generalized likelihood ratio test (GLRT), 193, 196

Generalized likelihood ratio test (GLRT-) based sequential hypothesis testing, 10, 16

Generalized sidelobe canceler (GSC), 527, 587

Generalized weighted subspace fitting (GWSF) algorithm, 832

GeneralizedWigner Distribution (GWD), 60

Generalmaximumlikelihood theory, 16

General-rank signal model, 545

General statistical tools, DOA estimation, 723

AMVB and CRB, relationship, 735–736

asymptotically minimum variance bounds (AMVB), 734–735

Cramer-Rao bounds (CRB), 729–730

specific algorithm, performance analysis of, 723

Geographical information system (GIS), 258

Geolocation, 6

Geolocation-maps

estimation methods, 260

extended Kalman filter, 264

mathematical framework, 260–261

nonlinear filtering, 261

nonlinear filter theory, 261

particle filter (PF), 267

unscented Kalman filter (UKF), 265

mapping in practice, 292

maps and applications, 276

airborne fast vehicles, 278

airborne slow vehicles, 280

cellular phones, 285

road-bound vehicles, 276

small migrating animals, 290

surface vessels, 283

underwater vessels, 282

motion models, 270

dead-reckoning model, 272

kinematic model, 274

theory, 259

Geometric data model, 465

Geometric triangulation, 799

Global cost to neighborhood costs, 347

Global Navigation Satellite Systems (GNSS), 893–894

beamforming, 898

error sources and benefits of antenna arrays, 894–896

Global Positioning System (GPS), 893

GLRT-based sequential hypothesis testing. *See* Generalized

likelihood ratio test (GLRT-) based sequential hypothesis testing

GNSS. *See* Global Navigation Satellite Systems

G-prior, 157

Gradient adaptive beamforming algorithms, 513

Graph laplacian and network connectivity, 430

GWSF algorithm. *See* Generalized weighted subspace fitting algorithm

## H

Hann(ing) window, 32

Heuristic approach, 9

Higher order time-frequency representations, 107

Hybrid time and frequency varying windows, 41

## I

Ideal array signal models, 821

Ideal data model, 466

i.i.d. model with geometric prior, 225

Inertial models, 273

Information and coding theory based methods, 17

Inner interferences in Wigner distribution, 72

Instantaneous bandwidth, 65

Instantaneous frequency (IF)

distribution concentrated, 61

interpretation, 48

Inverse Wishart distribution, 182

Inverted-gamma distribution, 180

I/Q imbalances, 828–829

**K**

- Kaiser window, 32  
 Kalman filter, 814  
     nonlinear observation model, 815  
     prediction phase, 814  
     update phase, 814  
 Kalman gain, 175–176  
 Kernel constraint, 82  
 Kernel decomposition method, 88  
 Kernel transformations, 85  
 Kinematic model, 274  
*K-L divergence.* *See* Kullback-Leibler (K-L) divergence  
 Kronecker products, properties of, 430  
 Kullback-Leibler (K-L) divergence, 10, 17–19, 218

**L**

- LCMV in frequency domain, 586  
 Leaky Integrate-and-Fire model, 250  
 Least squares estimation, 316  
 Linear coordinate transforms, of Wigner distribution, 69  
 Linear Gaussian model, 144  
     hyperparameters for, 153  
     parameters in, 151  
 Linear Gaussian state-space models, 164  
 Linearly constrained minimum variance (LCMV)  
     beamforming, 584  
 Linear model, 328  
 Linear signal transforms, 28  
 Local polynomial Fourier transform (LPFT), 50  
 Log posterior density, 14  
 Look direction mismatch (pointing error) problem, 523  
 Lorden’s problem, 230  
 Low system temperatures, 876  
 LSMI adaptive beamformers, 538  
 L-statistics in time-frequency, 131  
 L-Wigner distribution realization, 117

**M**

- Magnetoencephalography (MEG) signal processing, 918  
 Manifold separation technique, 837, 842  
 Marginalization of speed, 275  
 Marginalized particle filters, 171  
 Marginal likelihood, 152  
     calculation of, 159  
 Markov chain Monte Carlo (MCMC), 5, 163  
 Markov transition matrix, 176  
 Martingales, 212  
 Matched field processing (MFP), 935  
 Matched subspace detector (MSD), 927–928  
 Mathematical framework, 260–261

- Mathematical preliminaries, 212  
 Matrix-based subspace estimation, 667  
 Matrix perturbations, 307  
 Maximum likelihood algorithms, 738  
     asymptotic properties, 739  
     large sample ML approximations, 741–742  
     stochastic and deterministic algorithms, 738–739  
 Maximum likelihood approach, 618  
     deterministic maximum likelihood, 618  
     stochastic maximum likelihood, 619  
 Maximum likelihood (ML) estimation, 146  
 Maximum likelihood estimation (MLE), 303  
 MCMC. *See* Markov chain Monte Carlo (MCMC)  
 MDL. *See* Minimum description length (MDL)  
 Mean-square convergence, 410  
 Mean-square error bound, 305  
 Mean-square-error estimation, 327  
 Mean-square performance, of individual nodes, 387  
 Mean-square stability, 381  
 Method of direction of arrival estimation (MODE), 682, 684  
 Microphone arrays, 938–940  
     aeroacoustic source localization, 940–941  
     wideband adaptive beamforming, 941–942  
 Microwave and ultrasound imaging, 497  
 MIMO. *See* Multiple-input multiple-output (MIMO)  
 Minimax algorithms, optimality properties of, 234  
 Minimax approach, 211  
 Minimax quickest change detection, 228  
 Minimum description length (MDL), 10–19  
 Minimum variance distortionless response (MVDR)  
     beamformer, 606, 823  
 Minimum-variance distortionless response (MVDR)  
     space-time filter, 867–868  
 Mixture Kalman filter, 171  
 ML estimation. *See* Maximum likelihood (ML) estimation  
 MODE. *See* Method of direction of arrival estimation  
 Model and robust formulation, 563  
 Model-driven techniques, 830–831  
     Bayesian approach, 832–834  
     deterministic approach, 831–832  
 Modeling errors and array calibration, 492  
 Model order selection, 6  
     information and coding theory based methods, 17  
     Akaike information criterion (AIC), 17  
     minimum description length, 19  
     regression, variable selection in, 11  
         AIC and stepwise regression, 12  
         bootstrap methods, 12  
         cross-validation (CV), 12  
         statistical inference paradigms  
             Bayesian information criterion (BIC), 14  
             GLRT-based sequential hypothesis testing, 16  
         subspace methods, signals in, 21

- Model uncertainty  
     and Bayesian decision theory, 158  
     structures for, 161
- Monte Carlo method (MCM), 317
- Multi-dimensional algorithms, 690  
     R-D MODE, 694–695  
     R-D NC standard ESPRIT, 696  
     R-D NC unitary ESPRIT, 697–698  
     R-D RARE, 691–694  
     R-D standard ESPRIT, 690–691  
     R-D unitary ESPRIT, 691
- Multi-objective optimization, 197
- Multiple antennas techniques  
     IEEE 802.11, 915–916  
     LTE  
         diversity schemes, 907–908,  
         multiple user MIMO (MU-MIMO), 912–913  
         multiplexing schemes, 909–910  
         uplink MIMO, 913  
     WiMAX, 913–914
- Multiple-input multiple-output (MIMO), 870–871  
     adaptive array processing at radar receivers, 874–875  
     direction-of-departure (DOD)/direction-of-arrival (DOA)  
         estimation, 789–790  
     example, 792–793  
     flexible transmit beampattern synthesis, 871  
     joint DOD/DOA estimations, 791  
     signal model, 790  
     UAV equipped with, 872
- Multiple signal classification (MUSIC) algorithm,  
     612, 674–675  
     Fourier domain root-MUSIC, 689–690  
     interpolated root-MUSIC, 687–688  
     root-MUSIC, 676  
     unitary root-MUSIC, 677  
     weighted MUSIC, 675
- Multiplicative and additive noise model, 310
- Multi-sensor extracellular probes, 923, 925–926  
     data model, 926  
     multi-sensor feature extraction, 926
- Multi-sensor feature extraction, 926  
     discrete wavelet transform, 927  
     matched subspace detector (MSD), 927–928  
     principal component analysis, 927
- Multi-time Wigner higher order distribution (MTWD), 110
- Multivariate Gaussian, 178
- MUSIC. *See* Multiple signal classification
- MVDR beamformer. *See* Minimum variance distortionless response beamformer
- MVDR beamforming with data covariance matrix, 512
- MVDR robust adaptive beamforming design, 536
- N**
- Network mean-square performance, 386
- Networks and neighborhoods, 324
- Network topologies, 194
- Neuroscience, 250
- Neyman-Pearson formulation, 191
- Noise, 803
- Noise eigenvectors, 610
- Noise sources over exchange links, 404
- Noise subspace, 807–808
- Noise vector, 11
- Noisy information exchanges, 403  
     adaptive combination weights, 412  
     convergence in mean, 409  
     error recursion, 405  
     mean-square convergence, 410  
     noise sources over exchange links, 404
- Non-coherent array applications, 493
- Non-cooperative adaptive solution, 331
- Non-cooperative mean-square-error solution, 330
- Non-cooperative strategy, 394
- Non Gaussian case, 733–734
- Non-Gaussian case, 311
- Non-ideal data models, 470
- Nonlinear filtering, 261
- Nonlinear filter theory, 261  
     Bayes optimal filter, 261  
     covariance bound, 263  
     Kalman filter, 264  
     mean and covariance, 263
- Non-linear least square source localization, 809  
     least square solution, 810  
     nonlinear quadratic optimization, 809  
     source localization using table look-up, 810
- Nonlinear observation model, 815
- Nonlinear renewal theory, 216
- Nonparametric rules, in distributed detection, 196
- Non-stationary signal analysis, 5
- Non-stationary signal analysis time-frequency approach  
     higher order time-frequency representations, 107  
     signal phase derivative and distributions definitions, 112  
     Wigner bispectrum, 107  
     Wigner higher order spectra, 108  
     Wigner multi-time distribution, 110
- linear signal transforms, 28  
     discrete form and realizations of STFT, 36  
     Gabor transform, 45  
     generalization, 56  
     local polynomial Fourier transform, 50  
     short-time Fourier transform, 28  
     stationary phase method, 46  
     STFT and continuous wavelet transform, 52

Non-stationary signal analysis (*Continued*)

- quadratic time-frequency distributions, 58
  - ambiguity function, 77
  - Cohen class of distributions, 80
  - Kernel decomposition method, 88
  - Rihaczek distribution, 58
  - S*-method, 89
  - time-frequency, reassignment in, 99
  - time-frequency representations, affine class of, 104
  - Wigner distribution, 60
- sparse signals in time-frequency, 124
  - compressive sensing, 131
  - concentration measures, 124
  - L*-statistics in time-frequency, 131
  - sparse signals, 126
- time-frequency analysis applications, 135
  - biomedical signal analysis, 137
  - car engine signal analysis, 137
  - seismic signal analysis, 137
  - spread spectrum systems, interference rejection in, 139
  - time-frequency radar signal processing, 135
  - time-variant filtering, 138
  - video sequence, velocities of moving objects, 138
  - watermarking, in space/spatial-frequency domain, 140
- Nonstationary signals, 767
  - and time-frequency representations, 767
- Normal-inverted-gamma distribution, 181
- Normal-inverted-gamma prior, 154
- Normalized maximum likelihood approach, 19
- Notation, 326
- Number of sources, detection of, 752
  - MDL criterion, 752–753

## O

- Odometric models, 272
- One-dimensional algorithms, 674
  - Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT), 678
    - Fourier domain root-MUSIC, 689–690
    - generalized ESPRIT (GESPRIT), 681–682
    - interpolated root-MUSIC, 687–688
    - manifold separation scheme, 688–689
    - method of direction of arrival estimation (MODE), 682, 684
    - MUSIC, 674–675
    - rank-reduction (RARE) DOA estimation method, 684, 686
    - root-MUSIC, 676
    - root-RARE, 686
    - unitary root-MUSIC, 677
    - weighted MUSIC, 675
  - One-dimensional arrays, 472
  - On-source minus off-source radiometric detection, 876
  - Open-loop multiplexing schemes, 911–912

- Optimal near-field signal-to-noise plus interference beamformer (SNIB), 591
- Optimal SINR, 512
- Optimizing combination weights, 398
- Orthogonal matching pursuit, 12

## P

- Parallel configuration, 189
- Parallel factor (PARAFAC) analysis, 652
- Parameter estimations, 772
- Parameter identifiability, 722–723
- Parametric array model, 720–721
- Parametric methods, 489
- Parametric statistical models, 298
- Parsimony, principle of, 12
- Particle filter (PF) illustration, 267, 269
- Particle filtering and auxiliary sampling, 169
- Path integration, 259
- Performance analysis, 6
- Performance analysis and bounds
  - asymptotic analysis and central limit theorem, 313
  - asymptotic analysis and parametric models, 315
    - Fourier transform, 315
    - least squares estimation, 316
    - asymptotic normality and MLE, 304
    - confidence intervals, 318
    - constrained Cramér-Rao bound and constrained MLE, 308
      - comments and properties of CCRB, 309
      - constrained CRB, 308
      - constrained MLE, 310
    - Cramér-Rao bound, 299
    - maximum likelihood estimation and CRB, 303
    - mean-square error bound, 305
  - Monte Carlo method (MCM), 317
    - approximate an expectation, 318
    - computing CRB via Monte Carlo, 318
  - multiplicative and non-Gaussian noise, 310
    - Gaussian case, 311
    - multiplicative and additive noise model, 310
    - Non-Gaussian case, 311
  - parametric statistical models, 298
  - perturbation methods, 306
    - matrix perturbations, 307
    - perturbation analysis of MLE, 308
    - perturbations and statistical analysis, 306
  - Performance bound, 627
    - ML methods, 627
    - subspace methods, 627
  - Performance measures, 375
  - Perturbation methods, 306
  - Phase distortion, 803

- Polarization sensitivity, 640  
 Pollak's problem, 231  
 Polynomial phase signal (PPS), 772  
 Polynomial Wigner-Ville distribution, 114, 118  
 Positioning and navigation, 893–894  
     array-based parameter estimators, 904–905  
     beamforming, 898  
     DOA estimation algorithms, 903  
     signal model, 896  
 Posterior inference and Bayesian cost functions, 149  
 Prediction error decomposition, 166  
 Predictor-corrector formulation, 169  
 Probability densities and integrals, 178  
 Probability distribution function (PDF), 799  
 Probability mass function (PMF), 815  
 Problem formulation, 721–722  
 Product higher order ambiguity function (PHAF), 124  
 Projection adaptive beamforming methods, 515  
 Pseudo and smoothed Wigner distribution, 73  
 Pseudo quantum signal representation, 64
- Q**
- Q-factor transform, 53–54  
 Quadratic time-frequency distributions, 58  
 Quickest change detection, 4  
     applications of, 250  
     Bayesian quickest change detection, 217  
         Bayesian i.i.d. setting, 217  
         general asymptotic Bayesian theory, 221  
         independent and identically distributed model with geometric prior, 225  
     mathematical preliminaries, 212  
         martingales, 212  
         renewal and nonlinear renewal theory, 214  
         stopping times, 213  
     minimax quickest change detection, 228  
         general asymptotic minimax theory, 237  
         minimax algorithms, optimality properties of, 234  
         Shiryav algorithm, minimax algorithms based on, 232  
     models, 240  
     variants and generalizations of the quickest change detection problem, 241  
         data-efficient quickest change detection, 244  
         distributed sensor systems, 247  
         with unknown pre- or post-change distributions, 241  
     variants of quickest change detection problem, 249
- R**
- Radar applications, 860–862  
     space-time adaptive processing, 862  
 Radio astronomy, 875–876
- astronomical phased array feeds, 885  
 challenges and solutions to, 891  
 synthesis imaging, 877–878  
 Radio frequency interference (RFI), 876  
 Rao-Blackwellized particle filter, 171  
 Real time distributions, 116  
 Received signal strength (RSS), 259  
 Received signal strength indicator (RSSI), 804  
 Rectangular window function, 30  
 Redundancy averaging, 622–623  
 Renewal and nonlinear renewal theory, 214  
 RFI. *See* Radio frequency interference  
 Rihaczek distribution, 58  
 Rissanen's minimum description length (MDL), 635  
 Road-bound vehicles, 276  
 Robust adaptive beamforming, 521  
     comparison by simulation, 544  
     diagonally loaded SMI beamformer, 522  
     doubly constrained robust adaptive beamforming, 539  
     eigenspace-based beamformer, 537  
     eigenvalue beamforming using multi-rank MVDR beamformer, 541  
     forward-backward averaging and spatial smoothing, 533  
     generalized sidelobe canceler, 527  
     general-rank signal model, 545  
     look direction mismatch (pointing error) problem, 523  
     LSMI adaptive beamformers, 538  
     motivations, 521  
     MVDR robust adaptive beamforming design, 536  
     probabilistically constrained robust adaptive beamforming, 540  
     rapidly moving interferences, 535  
     sequential quadratic programming-based robust adaptive beamforming, 540  
     SOI and interferences, 529  
     steering vector estimation, 542  
     wideband robust adaptive beamforming, 546  
     worst-case-based robust adaptive beamforming, 538  
 Robust Chebyshev design, 567  
 Robust methods, 842–843  
     worst-case performance/optimization/uncertainty sets, 843, 845  
 Robustness of algorithms, 747  
     robustness to array modeling errors, 749–750  
     robustness w.r.t. narrowband assumption, 747  
 Robust total least squares design, 567  
 Robust WLS design, 566  
 Role of adaptation, localization application, 340
- S**
- Sample covariance matrix (SCM), 22  
 Sample matrix inversion adaptive beamformer, 514

- Scalogram, 53–54  
*SCM.* *See* Sample covariance matrix (SCM)  
 Second-order algorithms, 743  
 Seismic signal analysis, 137  
 Semi-definite relaxation (SDR), 873  
 Sensor array, 819  
 Sensor networks, 250  
 Sensor networks, source localization in, 497  
 Sequential Monte Carlo (SMC), 167  
 Sequential quadratic programming-based robust adaptive beamforming, 540  
 Serial configuration, 195  
 Shiryaev algorithm, minimax algorithms based on, 232  
 Shiryaev-Roberts algorithm, 211  
 Shiryaev-Roberts-Pollak (SRP) algorithm, 232  
 Shiryaev-Roberts-*r* (SR-*r*) algorithm, 232  
 Shiryaev's formulation, 210  
 Shiryaev statistic evolution, 209  
 Short-time Fourier transform (STFT), 28  
   continuous STFT inversion, 34  
   duration measures and uncertainty principle, 32  
   of multi-component signals, 35  
   windows, 30  
 Sigma points, 266  
 Signal assumptions, 721–722  
 Signal detection, 485, 634  
 Signal eigenvectors, 610  
 Signal intelligence (SIGINT), 819  
 Signal phase derivative and distributions definitions, 112  
 Signal propagation models, 802  
   angle of arrival estimation, 806–808  
   received signal strength indicator (RSSI), 804  
   time delay estimation, 805  
 Signal reconstruction, 62  
 Signal (source) separation problem, 802  
 Signal subspace, 807–808  
 Signal-to-noise ratio (SNR), 603  
 Signal to noise ratio (SNR) beamformer, 888  
 Simulation results, 703  
   1-D algorithms using matrix-based subspace estimates, 703, 705–706  
   R-D algorithms using matrix-based subspace estimates, 706  
 Simultaneous localization and mapping (SLAM), 292  
 Sinc distribution, 83  
 SINR, 823  
 SLAM. *See* Simultaneous localization and mapping (SLAM)  
 Small migrating animals, 290  
 SMC. *See* Sequential Monte Carlo (SMC)  
 S-method, 89  
   discrete S-method, 91  
   multi-component signals, decomposition of, 97  
   *versus* smoothed spectrogram, 96  
 Soft-thresholding function, 249  
 Sonar, 928–929  
   acoustic vector sensors, 937–938  
   arrays, 929  
   matched field processing (MFP), 935  
   undersea acoustic channel, 932  
 Sound velocity, 932  
 Source localization algorithms, 808  
   Bayesian source localization, 808–809  
   non-linear least square source localization, 809  
   source localization using angle of arrival, 811–812  
   source localization using time difference of arrival, 810  
 Source localization and tracking  
   problem formulation, 800  
   signal propagation models, 802  
   source localization algorithms, 808  
   target tracking algorithm, 812  
   triangulation, 800  
 Source signal propagation model, 800  
 Space-time adaptive processing (STAP), 862, 602  
 Sparse data representation based approach, 607  
 Sparse signals in time-frequency, 124  
 Spatial filter design, 480  
 Spatial filtering, 471  
 Spatial filtering and beam patterns, 471  
 Spatial time-frequency distribution (STFD), 774  
   SNR enhancement, 775  
   subspace analysis, 777  
 SPC. *See* Statistical process control (SPC)  
 Special array structures, 657  
   centro-symmetric arrays, 659–660  
   minimum redundancy linear arrays, 658  
   multidimensional arrays, 661  
   partially calibrated arrays, 660–661  
   R-D shift invariance structure, 663  
   uniform circular array (UCA), 659–660  
   uniform linear arrays (ULAs), 657–658  
   uniform rectangular arrays (URAs), 659  
 Specific algorithm, performance analysis of, 723  
   asymptotic covariance and bias, 728–729  
   asymptotic distribution of estimated DOA, 727–728  
   asymptotic distribution of statistics, 724  
   functional analysis, 723  
 Spherical broadband beamformer, 581  
 Spread sources, 493  
 Spread spectrum systems, interference rejection in, 139  
 State-space models and sequential inference, 164  
 Stationary phase method, 46  
 Statistical process control (SPC), 250  
 Statistical signal processing  
   content, 3  
   contributions, 4  
   Bayesian computational methods, 5  
   bounds, 6

Statistical signal processing (*Continued*)

- diffusion adaptation over networks, 5
- distributed signal detection, 4
- geolocation, 6
- model order selection, 6
- non-stationary signal analysis, 5
- performance analysis, 6
- quickest change detection, 4
- time-frequency approach, 5
- historical recount, 3
- Steepest-descent iterations, 351
- Steerable broadband beamformer, 571
- Steering vector estimation, 542
- Steering vector model, 821–822
- Stepwise regression, 12
- STFD. *See* Spatial time-frequency distribution
- STFT and continuous wavelet transform, 52
- STFT, realizations of, 36
- Stochastic complexity, concept of, 19
- Stochastic matrices, 433
- Stopping times, 213
- Subspace-based algorithms, 674, 745
  - algorithms using tensor-based subspace estimates, 698
  - multi-dimensional algorithms, 690
  - one-dimensional algorithms, 674
  - simulation results, 703
- Subspace estimation, 667
  - forward-backward averaging and real-valued, 671–672
  - matrix-based subspace estimation, 667
  - with small number of snapshots, 669
  - tensor-based subspace estimation, 672
- Subspace fitting methods, 625
- Subspace methods, 488
- Synchronization issues, 252
- Synthesis imaging, 876–878
  - algorithms for solving the imaging equation, 881
  - geometry and signal definitions, 878
  - The imaging equation, 878

## T

- Tapped-delay-line models, 334
- Target localization, 336
- Target tracking algorithm, 812
  - dynamic and observation models, 812–813
  - Kalman filter, 814
  - sequential Bayesian estimation, 813
- TDoA. *See* Time Difference of Arrival
- Tensor-based subspace estimation, 672
- Time and frequency varying windows, 41
- Time delay, 803
- Time Difference of Arrival (TDoA), 805–806
- Time domain formulation, 591
- Time-frequency analysis applications, 135
- Time-frequency approach, 5

- Time-frequency maximum likelihood method, 782–784
  - examples, 784
- Time-frequency MUSIC, 780
  - examples, 780–781
- Time-frequency radar signal processing, 135
- Time-frequency representations
  - Cohen's class of, 767
  - examples, 769
- Time-frequency representations, affine class of, 104
- Time series modeling, 495
- Time-variant filtering, 138
- Time varying window, 41
- Total least squares design and Eigen-filters, 562
- Transient change detection, 251
- Transient mean-square performance, 390
- Triangulation, 800
  - angle based triangulation, 801–802
  - distance based triangulation, 800
  - generalizations, 802
  - geometric positions for, 801
- Two-dimensional arrays, 475

## U

- UCA. *See* Uniform circular array
- UKF. *See* Unscented Kalman filter
- ULAs. *See* Uniform linear arrays
- Uncertainty principle and Wigner distribution, 63
- Undersea acoustic channel, 932
  - noise and reverberation, 934
  - propagation models, 932
  - transmission loss, 933–934
- Uniform circular array (UCA), 659–660
- Uniform data profile, 389
- Uniform linear arrays (ULAs), 657–658
- Uniform rectangular arrays (URAs), 659
- Univariate Gaussian, 178
- Unscented Kalman filter (UKF), 265, 815
- URAs. *See* Uniform rectangular arrays

## V

- Variants and generalizations of the quickest change detection problem, 241
- Variants of quickest change detection problem, 249
- Video sequence, velocities of moving objects, 138

## W

- Wald's identity, 213
- Watermarking, in space/spatial-frequency domain, 140
- Wave equation, 579
- Wavefield modeling principle, 837, 842

Wavefield parameters, 819–820

Wave propagation, 465, 600

Weighted least square (WLS) design, 561

Wideband adaptive beamforming, 519, 941–942

Wideband array response, 477

Wideband DOA estimation, 631

Wideband maximum likelihood estimation, 632

Wideband robust adaptive beamforming, 546

Wiener filter (WF), 482, 590

Wigner bispectrum, 107

Wigner distribution, 60

auto-terms and cross-terms, 70

based inversion and synthesis, 76

discrete pseudo, 75

inner interferences in, 72

instantaneous bandwidth, 65

instantaneous frequency, distribution concentrated, 61

linear coordinate transforms of, 69

properties of, 67

pseudo and smoothed, 73

pseudo quantum signal representation, 64

signal reconstruction, 62

uncertainty principle and, 63

Wigner higher order spectra, 108

Wigner multi-time distribution, 110

Wigner-Radon transform, 769–772

Wireless communications, 907

multiple antennas techniques, LTE, 907

Wishart distribution, 181

Worst-case-based robust adaptive beamforming, 538

## Z

Zhao-Atlas-Marks distribution, 83