Amitava Chatterjee
Hadi Nobahari
Patrick Siarry *Editors*

Advances in Heuristic Signal Processing and Applications

Springer

Advances in Heuristic Signal Processing and Applications

Amitava Chatterjee • Hadi Nobahari •
Patrick Siarry
Editors

Advances in Heuristic Signal Processing and Applications

Springer

Editors
Amitava Chatterjee
Electrical Engineering Department
Jadavpur University
Kolkata, West Bengal, India

Patrick Siarry
Laboratory LiSSi
University of Paris 12
Créteil, France

Hadi Nobahari
Dept. of Aerospace Engineering
Sharif University of Technology
Tehran, Iran

ISBN 978-3-642-37879-9 ISBN 978-3-642-37880-5 (eBook)
DOI 10.1007/978-3-642-37880-5
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013941398

ACM Computing Classification (1998): I.2, I.4, J.2

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The development of sophisticated, state-of-the-art, signal processing algorithms and their suitable applications has evolved and remained a research activity of primary interest for more than 40 years now with special emphasis on the domains of communication, control theory, estimation, pattern recognition, design of electrical and electronic systems, mechanical systems, etc. Both one dimensional signal processing and multidimensional signal processing, namely image and video processing, have received tremendous research attention in recent years. Although the original focus was to develop traditional algorithms, perform in-depth analysis and then try to improve upon their performance from different perspectives, in recent times there has been a significant interest in applying heuristic based methods in solving signal processing problems.

This book makes a humble attempt in offering a collection of notable works that have recently contributed in the domain of heuristic signal processing, both in developing suitable general purpose algorithms and also in solving specialized application problems. Special emphasis was put on collecting several works that attempt to propose several heuristic, iterative optimization methods, essentially employing modern evolutionary and swarm intelligence based optimization techniques, specially employed for solving several relevant signal and image processing problems. Many of these problems under consideration originate from several important domains like the fields of communication engineering, estimation and tracking problems, the evergreen digital filter design problems, wireless sensor network problems, bioelectric signal classification problems, image denoising, image feature tracking problems, etc. We do hope that the sheer variety of the problems discussed in this book along with different techniques employed in solving them will arouse great interest among a large section of signal and image processing researchers all over the world.

Chapter 1 by Ling, Ho, and Teo shows how a two-channel linear phase FIR quadrature mirror filter bank minimax design problem can be formulated in terms of a nonconvex optimization problem. This optimization problem attempts to minimize a weighted sum of several performance indices like the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum

stopband ripple magnitude of the prototype filter. The chapter discusses in great detail how a joint norm relaxed sequential quadratic programming and filled function method can be utilized for finding the global minimum of the nonconvex optimization problem. The authors show the effectiveness and utility of the proposed method using several suitable case studies.

Chapter 2 by de Lamare focuses on developing robust reduced-rank linearly constrained minimum variance (LCMV) beamforming algorithms, utilizing the notion of the joint iterative optimization of parameters. The author utilizes the concept of constrained robust joint iterative optimization of parameters based on the minimum variance criterion. He shows how this optimization procedure can be suitably utilized to adjust the parameters of a rank-reduction matrix, a reduced-rank beamformer, and the diagonal loading in an alternating manner. de Lamare also demonstrates how stochastic gradient and recursive least-squares adaptive algorithms can be devised for suitable implementation of this optimization technique based robust beamforming methodology.

In Chap. 3, Sen, Tang, and Nehorai propose a multi-objective optimization based methodology that can be used to design an orthogonal frequency division multiplexing (OFDM) radar signal, used for detection of a moving target in the presence of multipath reflections. They discuss in detail the development of a parametric OFDM measurement model for a particular range cell under test, and how it can be converted to an equivalent sparse-model by considering the target returns over all the possible signal paths and target velocities. They utilize the multi-objective optimization procedure for designing the spectral-parameters of the transmitting OFDM waveform by simultaneously optimizing three objective functions: (i) maximizing the Mahalanobis distance, (ii) minimizing the weighted trace of the Cramer–Rao bound matrix for the unknown parameters, and (iii) minimizing the upper bound on the sparse recovery error.

Chapter 4 by Kwolek demonstrates the utilization of the particle swarm optimization (PSO) algorithm for multi-target tracking problems. Here PSO is employed to track the local mode of the similarity measure and a suitable objective function is developed utilizing the region covariance matrix and multi-patch based object representation. In this process, the target locations and velocities are determined and thereafter they are employed in a PSO based procedure for further refinements of the extracted trajectories. In the last stage, the algorithm utilizes a conjugate method for the final optimization, and the suitability of the proposed algorithm is aptly demonstrated by evaluating performance on publicly available datasets.

Chapter 5 by Boussaïd, Chatterjee, Siarry, and Ahmed-Nacer studies the performance of a wireless sensor network in the context of binary detection of a deterministic signal. This work considers a decentralized organization of spatially distributed sensor nodes and considers the development of an optimal power allocation scheme that will minimize the total power spent by the whole sensor network under a desired detection error probability. The chapter considers two scenarios for the fusion of binary decisions, depending on whether the observations are independent and identically distributed or correlated. This work shows how this problem can be solved utilizing different variations of biogeography-based optimization algorithms and compares their performances vis-à-vis similar problems solved using GA and PSO.

In Chap. 6, Aslan and Saranlı focus on the problem of detection threshold optimization in a tracker-aware manner so that a feedback from the tracker to the detector is established, in a bid to maximize the overall system performance. This chapter puts high emphasis on the development of optimization schemes for the probabilistic data association filter utilizing the modified Riccati equation (MRE) and the hybrid conditional averaging (HYCA) algorithm. The chapter also proposes a closed-form solution for the MRE-based dynamic threshold optimization problem.

Chapter 7 by Luitel and Venayagamoorthy introduces the iterative design of finite impulse response filters using PSO with the quantum infusion (PSO-QI) algorithm. The design specification utilizes two methods for calculating performance indices: (i) minimizing the mean-squared error between the actual and the ideal filter response and (ii) minimizing the mean-squared error between the ripples in the passband and the stopband of the designed filter and the desired filter specification. The chapter shows performance comparisons vis-à-vis the constrained least-squares method of filter design.

Chapter 8 by Sengupta, Chakraborti, and Konar introduces Invasive Weed Optimization (IWO) based algorithms for solving two-dimensional IIR digital filter design problems. The authors develop an improved variant of IWO by introducing a constriction factor in the seed dispersal phase. The design algorithm is developed using temporal difference Q-Learning and it falls under the category of special types of adaptive Memetic Algorithms.

Chapter 9 by Castella, Moreau, and Zarzoso discusses a survey of kurtosis optimization schemes employed for MISO source separation and equalization problems. The chapter provides an in-depth review of some of the most widely employed iterative algorithms utilizing kurtosis for MISO source separation and equalization. These methods include gradient and Newton search based methods, algorithms with optimal step-size selection, and also algorithms based on reference signals. The authors show the efficacy of these algorithms by presenting the performance evaluation for case studies chosen from the fields of digital communications and biomedical signal processing.

Chapter 10 by Nobahari, Sharifi, and MohammadKarimi proposes a new class of filters, based on swarm intelligence, for the purpose of nonlinear systems state estimation. The authors show how such swarm filters can be formulated for a nonlinear system state estimation problem as a stochastic dynamic optimization problem. The chapter shows how successfully PSO and ant colony optimization can be utilized for this estimation purpose and how they perform vis-à-vis other popular nonlinear filters such as the unscented Kalman filter, etc.

In Chap. 11, Pan and Chang demonstrate how the design of multiplier-less digital filters can be carried out utilizing Canonic Signed Digit (CSD) code. The chapter solves the optimum design problem for digital filters utilizing GA. This chapter introduces the concept of CSD coded GA which can effectively reduce the time consumed in the process of the evolution and thus can accelerate the training speed. This chapter also introduces a new hybrid code for the filter coefficients that can be instrumental in improving the precision of the coefficients of a digital filter. The chapter examines the design of both Finite-Impulse Response and Infinite-Impulse Response Filters in this context.

Chapter 12 by Dutta, Chatterjee, and Munshi presents a thorough discussion on the development of robust algorithms for pathological classification of human gait signals. The chapter shows how cross-correlograms can be utilized for feature extraction and how both time and frequency domain based features can be extracted from cross-correlation procedures. These features are used as inputs for Elman's recurrent neural network (ERNN) based classifiers for automatic identification of healthy subjects and those with neurological disorder, and also the type of disorder, e.g., people suffering from Parkinson's disease (PD) or Huntington's disease (HD), or Amyotrophic Lateral Sclerosis (ALS). The chapter discusses how modular ERNNs can be utilized to develop composite classifiers to improve classification accuracy. The performances of such systems developed are compared with similar systems developed utilizing back propagation neural network (BPNN), learning vector quantization (LVQ), and least-squares support vector machine (LS-SVM) based classification algorithms.

In Chap. 13, Abdeldjalil Ouahabi presents a detailed review of image denoising techniques using wavelets which are specifically aimed at medical imaging applications. He specifically considers medical ultrasound and magnetic resonance images and discusses the denoising performances using the well known indices of SNR (or PSNR) and visual aspects of image quality. In the process, he highlights an important fact that image denoising using wavelet-based multi-resolution analysis requires employment of a judicious compromise between noise reduction and preserving significant image details. Hence, the author emphasizes the importance of employing heuristics to supplement theory and making it simpler for practical applications to involve less complexity.

Chapter 14 by Guo and Ruan presents a sparse representation method for single-channel signal separation with a priori knowledge. The key features of the proposed method include dictionary constructions and pursuit algorithms for finding sparse representations. The chapter also presents an overview of popular schemes that are commonly employed to achieve these two key features. The performance evaluation shows that the proposed method can efficiently separate the overlapping resonances and the baseline.

Chapter 15 by Raphael, Philippe, and Christine performs a detailed review of different approaches to the introduction of a color monogenic wavelet transform, that offer a geometric representation of grayscale images through an AM/FM model that facilitates invariance of coefficients to translations and rotations. The authors start from the grayscale monogenic wavelets together with a color extension of the monogenic signal based on geometric algebra and move on by giving a step-by-step description.

Chapter 16 by Pissaloux, Maybank, and Velázquez gives a vivid description of the state-of-the-art in image and feature matching, both in 2D and 3D, specifically aimed at the embedded or wearable real-time system implementations. The chapter first discusses relaxation, maximal clique, tree search, region growing, and dynamic programming based methods. This is followed by a discussion on the popular correlation-based methods, with a fixed size or adaptive sized window, pyramidal methods, the iterative closest point (ICP) algorithm, and probability (saliency)-based approaches.

In the end, we, the editors of this volume, would like to thank everyone who has contributed directly or indirectly in making this project happen. We would specially like to thank all chapter contributors who have made notable contributions in their own ways by writing their chapters and enriching this book. Their timely contributions and active cooperation helped the process to be smooth. Now we sincerely hope that the final product will satisfy our readers all over the world and will be useful, in a small way, in further enriching their subject knowledge and will help them to be better equipped in their future research endeavors.

Kolkata, India — Amitava Chatterjee
Tehran, Iran — Hadi Nobahari
Paris, France — Patrick Siarry
November 2012

Contents

Contributors

Mohamed Ahmed-Nacer University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria

Murat Şamil Aslan İLTAREN, Advanced Technologies Research Institute, TÜBİTAK BİLGEM, Ankara, Turkey

Ilhem Boussaïd University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria

Philippe Carré XLIM Laboratory, UMR CNRS 7252 University of Poitiers, Futuroscope Chasseneuil Cedex, France

Marc Castella Institut Mines-Télécom/Télécom SudParis, CNRS UMR 5157 SAMOVAR, Evry Cedex, France

Tathagata Chakraborti Dept. of Electronics and Telecommunication Eng., Jadavpur University, Kolkata, India

Cheng-Yuan Chang Department of Electrical Engineering, Chung Yuan Christian University, Jhongli, Taiwan, R.O.C.

Amitava Chatterjee Electrical Engineering Department, Jadavpur University, Kolkata, West Bengal, India

Rodrigo C. de Lamare Communications Research Group, Department of Electronics, University of York, York, UK

Saibal Dutta Electrical Engineering Department, Heritage Institute of Technology, Kolkata, West Bengal, India

Christine Fernandez-Maloigne XLIM Laboratory, UMR CNRS 7252 University of Poitiers, Futuroscope Chasseneuil Cedex, France

Yu Guo Department of Biomedical Engineering, Tianjin University, Tianjin, China

Charlotte Yuk-Fan Ho School of Mathematical Sciences, Queen Mary University of London, London, UK

Amit Konar Dept. of Electronics and Telecommunication Eng., Jadavpur University, Kolkata, India

Bogdan Kwolek Rzeszow University of Technology, Rzeszów, Poland

Bingo Wing-Kuen Ling Department of Information Engineering, Guangdong University of Technology, Guangzhou, Guangdong Province, China

Bipul Luitel Real-Time Power and Intelligent Systems (RTPIS) Lab, Clemson University, Clemson, SC, USA

Steve Maybank Department of Computer Science and Information Systems, Birkbeck College, University of London, London, UK

Hamed MohammadKarimi Guidance and Control Research Center, Sharif University of Technology, Tehran, Iran

Eric Moreau ISITV, LSIS UMR-CNRS 7296, University of the South Toulon-Var, La Valette du Var Cedex, France

Sugata Munshi Electrical Engineering Department, Jadavpur University, Kolkata, India

Arye Nehorai Preston M. Green Department of Electrical & Systems Engineering, Washington University in St. Louis, Saint Louis, MO, USA

Hadi Nobahari Guidance and Control Research Center, Sharif University of Technology, Tehran, Iran

Abdeldjalil Ouahabi Polytech Tours, Tours University, Tours, France

Shing-Tai Pan Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, R.O.C.

Edwige E. Pissaloux ISIR (Institut des Systèmes Intelligents et de Robotique), UPMC (Université Paris 6), Paris, France

Su Ruan Laboratoire LITIS(EA 4108), Equipe Quantif, Université de Rouen, Rouen, France

Afşar Saranlı Dept. of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey

Satyabrata Sen Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA

Abhronil Sengupta Dept. of Electronics and Telecommunication Eng., Jadavpur University, Kolkata, India

Alireza Sharifi Guidance and Control Research Center, Sharif University of Technology, Tehran, Iran

Patrick Siarry LiSSi (EA 3956), Université de Paris-Est Créteil Val de Marne, Créteil, France

Raphael Soulard XLIM Laboratory, UMR CNRS 7252 University of Poitiers, Futuroscope Chasseneuil Cedex, France

Gongguo Tang Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, WI, USA

Kok-Lay Teo Department of Mathematics and Statistics, Curtin University of Technology, Perth, Australia

Ramiro Velázquez Mecatrónica y Control de Sistemas, Universidad Panamericana, Aguascalientes, Mexico

Ganesh Kumar Venayagamoorthy Real-Time Power and Intelligent Systems (RTPIS) Lab, Clemson University, Clemson, SC, USA

Vicente Zarzoso I3S Laboratory, University of Nice Sophia Antipolis, CNRS, Les Algorithmes, Euclide-B, Sophia Antipolis Cedex, France

Chapter 1
Nonconvex Optimization via Joint Norm Relaxed SQP and Filled Function Method with Application to Minimax Two-Channel Linear Phase FIR QMF Bank Design

Bingo Wing-Kuen Ling, Charlotte Yuk-Fan Ho, and Kok-Lay Teo

Abstract In this chapter, a two-channel linear phase finite impulse response (FIR) quadrature mirror filter (QMF) bank minimax design problem is formulated as a nonconvex optimization problem so that a weighted sum of the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter is minimized subject to specifications on these performances. A joint norm relaxed sequential quadratic programming and filled function method is proposed for finding the global minimum of the nonconvex optimization problem. Computer numerical simulations show that our proposed design method is efficient and effective.

1.1 Introduction

Filters are fundamental building blocks of many engineering systems, such as in multimedia [13] and communication [1] systems. Hence, developing efficient and effective filter design methods are essential. There are many filter design techniques such as the window design method. However, these design methods have some limitations. For example, it is necessary to find a closed form of the impulse responses of the filters. Also, the bandwidths of the transition bands and the ripples of different frequency bands are approximately the same. Because of these limitations, it is required to develop new methodologies for designing filters [8–11, 14, 28].

B.W.-K. Ling (✉)
Department of Information Engineering, Guangdong University of Technology, Guangzhou, Guangdong Province, 510006, China
e-mail: yongquanling@gdut.edu.cn

C.Y.-F. Ho
School of Mathematical Sciences, Queen Mary University of London, London, E1 4NS, UK
e-mail: c.ho@qmul.ac.uk

K.-L. Teo
Department of Mathematics and Statistics, Curtin University of Technology, Perth, CRICOS Provider Code 00301J, Australia
e-mail: K.L.Teo@curtin.edu.au

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_1,

On the other hand, there are many advantages of minimax two-channel linear phase FIR QMF banks. Since transition bandwidths of the filters in two-channel filter banks are usually larger than those in multi-channel filter banks, lengths of the filters in two-channel filter banks are usually shorter than those in multi-channel filter banks. Moreover, as only a single prototype filter is required for the design of a QMF bank and all other filters are derived from the prototype filter, the total number of filter coefficients required for the design of a QMF bank is usually smaller than those in general filter banks. Furthermore, as the linear phase property of the filters guarantees no phase distortion of the filter bank and the FIR property of the filters guarantees the bounded input bounded output (BIBO) stability of the filter bank, two-channel linear phase FIR QMF banks find many applications in image and video signal processing [22].

However, there are different considerations for two-channel linear phase FIR QMF bank designs compared to other filter bank designs [5–7, 12]. For example, unlike a multi-channel QMF bank [24, 26], a two-channel QMF bank cannot achieve the exact perfect reconstruction with the prototype filter having very good frequency selectivity [23]. Hence, it is useful to design a two-channel QMF bank so that a weighted sum of the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude and the maximum stopband ripple magnitude of the prototype filter is minimized subject to specifications on these performances. Nevertheless, this QMF bank minimax design problem is a nonconvex optimization problem subject to many nonlinear constraints. Since there are many nonlinear constraints, the corresponding dual problem consists of many Lagrange multipliers and the objective function of the dual problem is a nonlinear function of many variables. It is very difficult to find a locally optimal solution of the dual problem. Moreover, as nonconvex optimization problems usually consist of many local minima [32], it is usually stuck at these local minima. Hence, it is very difficult to find the global minima.

Gradient descent based methods are the most common approaches for finding locally optimal solutions of optimization problems. Due to the convergence issues, adaptive step sizes are used. However, when the step sizes are changed adaptively, in general it is not guaranteed to reach locally optimal solutions. There are mainly two different approaches for finding the global minima of nonconvex optimization problems. The first type of the approaches is nongradient based approaches, such as evolutionary algorithm based approaches [27, 29]. These approaches keep generating evaluation points randomly. Those evaluation points with better performances are kept, while those evaluation points with poor performances are ignored. However, these nongradient based approaches are not efficient. This is because most of the evaluation points are ignored and computational efforts are wasted. The second type of the approaches is filled function approaches [3, 4, 25, 30, 31, 33]. The working principles of the filled function methods are to find a sequence of locally optimal solutions and guarantee that their objective functional values are monotonic decreasing. For those optimization problems with a finite number of local minima, this method guarantees to reach globally optimal solutions. As the objective functional values of the locally optimal solutions are monotonic decreasing, better solutions are guaranteed for next iterations. Hence, this method is very efficient.

In this chapter, a joint norm relaxed sequential quadratic programming and filled function method is proposed for finding the global minimum of a two-channel linear phase FIR QMF bank minimax design problem. A locally optimal solution of the optimization problem in each iteration of the filled function method is found by the norm relaxed sequence quadratic programming method [2, 15–21, 34]. The globally optimal solution of the original optimization problem is found by the filled function method. The outline of this chapter is as follows. In Sect. 1.2, a two-channel linear phase FIR QMF bank minimax design problem is formulated as a nonconvex optimization problem. In Sect. 1.3, the globally optimal solution of the optimization problem is found by a joint norm relaxed sequential quadratic programming and filled function method. In Sect. 1.4, computer numerical simulations are presented. Finally, conclusions are drawn in Sect. 1.5.

1.2 Problem Formulation

Let us denote the transpose operator, the conjugate operator, and the conjugate transpose operator by the superscripts T, $*$ and $+$, respectively, and the modulus operator as $|\cdot|$. Let the transfer functions of the lowpass and the highpass analysis filters of a two-channel linear phase FIR QMF bank be $H_0(z)$ and $H_1(z)$, respectively, and those of the synthesis filters of the filter bank be $F_0(z)$ and $F_1(z)$, respectively. Here, $H_0(z)$ is the transfer function of the prototype filter. Let us denote the impulse response of the prototype filter as $h(n)$, the passband and the stopband of the prototype filter as B_p and B_s, respectively, the length of the prototype filter as N, the maximum passband ripple magnitude and the maximum stopband ripple magnitude of the prototype filter as δ_p and δ_s, respectively, the specifications on the acceptable bounds on the maximum passband ripple magnitude and the maximum stopband ripple magnitude of the prototype filter as ε_p and ε_s, respectively, and the desired magnitude response of the prototype filter as $D(\omega)$. In this chapter, it is assumed that the prototype filter is of even length and symmetric. Let the polyphase components of $H_0(z)$ be $E_0(z^2)$ and $E_1(z^2)$, that is,

$$H_0(z) \equiv E_0\left(z^2\right) + z^{-1} E_1\left(z^2\right). \tag{1.1}$$

Let us denote the transfer function of the filter bank as $T(z)$, the maximum amplitude distortion of the filter bank as δ_a, and the specification on the acceptable bound on the maximum amplitude distortion of the filter bank as ε_a. Let the vector containing these distortions and the even-time index filter coefficients be $\underline{x}$, that is,

$$\underline{x} \equiv \left[\delta_a, \delta_p, \delta_s, h(0), h(2), \ldots, h(N-2)\right]^T. \tag{1.2}$$

In order to achieve both the aliasing-free condition and the QMF pairs condition, the relationships among the analysis filters and the synthesis filters are governed by

$$H_1(z) = H_0(-z), \tag{1.3}$$

$$F_0(z) = 2H_0(z), \tag{1.4}$$

and

$$F_1(z) = -2H_0(-z). \tag{1.5}$$

As the prototype filter is of even length and symmetric, we have

$$H_0(z) = \sum_{n=0}^{\frac{N}{2}-1} h(2n)z^{-2n} + z^{-1}\sum_{n=0}^{\frac{N}{2}-1} h(2n)z^{-(N-2-2n)}, \tag{1.6}$$

$$E_0(z) = \sum_{n=0}^{\frac{N}{2}-1} h(2n)z^{-n}, \tag{1.7}$$

$$E_1(z) = \sum_{n=0}^{\frac{N}{2}-1} h(2n)z^{-(\frac{N}{2}-1-n)} = z^{-(\frac{N}{2}-1)}E_0\big(z^{-1}\big), \tag{1.8}$$

and

$$T(z) = 4z^{-1}E_0\big(z^2\big)E_1\big(z^2\big) = 4z^{-(N-1)}E_0\big(z^2\big)E_0\big(z^{-2}\big). \tag{1.9}$$

Let us denote

$$\eta(\omega) \equiv \big[0, 0, 0, 1, e^{-j\omega}, \ldots, e^{-j(\frac{N}{2}-1)\omega}\big]^T, \tag{1.10}$$

then

$$T(\omega) = 4e^{-j\omega(N-1)}\underline{x}^T\big(\eta(2\omega)\big)^*\big(\eta(2\omega)\big)^T\underline{x}. \tag{1.11}$$

Obviously, the filter bank does not suffer from the phase distortion, and the amplitude distortion of the filter bank can be expressed as $|4\underline{x}^T(\eta(2\omega))^*(\eta(2\omega))^T\underline{x} - 1|$. Let us denote

$$\underline{Q}(\omega) = 8\big(\eta(2\omega)\big)^*\big(\eta(2\omega)\big)^T, \tag{1.12}$$

then the amplitude distortion of the filter bank can be expressed as $|\frac{1}{2}\underline{x}^T\underline{Q}(\omega)\underline{x} - 1|$. Let us denote

$$\iota_a \equiv [1, 0, \ldots, 0]^T, \tag{1.13}$$

then the constraint on the maximum amplitude distortion of the filter bank can be expressed as

$$\frac{1}{2}\underline{x}^T\underline{Q}(\omega)\underline{x} - \iota_a^T\underline{x} - 1 \le 0 \tag{1.14}$$

and

$$-\frac{1}{2}\underline{x}^T\underline{Q}(\omega)\underline{x} - \iota_a^T\underline{x} + 1 \le 0 \quad \forall\omega \in [-\pi, \pi]. \tag{1.15}$$

Let us denote

$$\kappa(\omega) \equiv 2\left[0, 0, 0, \cos\left(\frac{N-1}{2}\omega\right), \cos\left(\frac{N-5}{2}\omega\right), \ldots, \cos\left(\frac{3-N}{2}\omega\right)\right]^T, \quad (1.16)$$

then

$$H_0(\omega) = \left(\eta(2\omega)\right)^T \underline{x} + e^{-j\omega(N-1)}\left(\eta(2\omega)\right)^+ \underline{x}, \quad (1.17)$$

therefore

$$H_0(\omega) = e^{-j\omega\frac{N-1}{2}} \times \left(\left[0, 0, 0, e^{j\omega\frac{N-1}{2}}, e^{j\omega\frac{N-5}{2}}, \ldots, e^{-j\omega\frac{N-3}{2}}\right]\underline{x} + \left[0, 0, 0, e^{-j\omega\frac{N-1}{2}}, e^{-j\omega\frac{N-5}{2}}, \ldots, e^{j\omega\frac{N-3}{2}}\right]\underline{x}\right). \quad (1.18)$$

Finally,

$$H_0(\omega) = e^{-j\omega\frac{N-1}{2}}\left(\kappa(\omega)\right)^T \underline{x}, \quad (1.19)$$

and the passband ripple magnitude of the prototype filter can be expressed as $|(\kappa(\omega))^T \underline{x} - D(\omega)|\ \forall\omega \in B_p$. Let us define

$$\iota_p \equiv [0, 1, 0, \ldots, 0]^T, \quad (1.20)$$

then the constraint on the maximum passband ripple magnitude of the prototype filter can be expressed as

$$\left|\left(\kappa(\omega)\right)^T \underline{x} - D(\omega)\right| \leq \iota_p^T \underline{x} \quad \forall\omega \in B_p. \quad (1.21)$$

Let us define

$$\underline{A}_p(\omega) \equiv \left[\kappa(\omega) - \iota_p, -\kappa(\omega) - \iota_p\right]^T \quad (1.22)$$

and

$$\underline{c}_p(\omega) \equiv \left[D(\omega), -D(\omega)\right]^T, \quad (1.23)$$

then the constraint on the maximum passband ripple magnitude of the prototype filter can be further expressed as

$$\underline{A}_p(\omega)\underline{x} - \underline{c}_p(\omega) \leq \underline{0} \quad \forall\omega \in B_p. \quad (1.24)$$

Similarly, let us define

$$\iota_s \equiv [0, 0, 1, 0, \ldots, 0]^T, \quad (1.25)$$

$$\underline{A}_s(\omega) \equiv \left[\kappa(\omega) - \iota_s, -\kappa(\omega) - \iota_s\right]^T, \quad (1.26)$$

and

$$\underline{c}_s(\omega) \equiv \left[D(\omega), -D(\omega)\right]^T, \quad (1.27)$$

then the constraint on the maximum stopband ripple magnitude of the prototype filter can be expressed as

$$\underline{\mathrm{A}}_s(\omega)\underline{\mathrm{x}} - \underline{\mathrm{c}}_s(\omega) \leq \underline{0} \quad \forall \omega \in B_s. \tag{1.28}$$

Let us define

$$\underline{\mathrm{A}}_b \equiv [\underline{\mathrm{I}}, \underline{0}] \tag{1.29}$$

and

$$\underline{\mathrm{c}}_b \equiv [\varepsilon_a, \varepsilon_p, \varepsilon_s]^T, \tag{1.30}$$

in which $\underline{\mathrm{I}}$ is the 3×3 identity matrix, then the specifications on the acceptable bounds on the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter can be expressed as

$$\underline{\mathrm{A}}_b\underline{\mathrm{x}} - \underline{\mathrm{c}}_b \leq \underline{0}. \tag{1.31}$$

In order to minimize a weighted sum of the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter subject to the specifications on these performances, the filter bank design problem is formulated as the following optimization problem:

Problem ($\underline{\mathrm{P}}$)

$$\min_{\underline{\mathrm{x}}} f(\underline{\mathrm{x}}) \equiv (\alpha\iota_a + \beta\iota_p + \gamma\iota_s)^T\underline{\mathrm{x}}, \tag{1.32}$$

subject to

$$g_1(\underline{\mathrm{x}}, \omega) \equiv \frac{1}{2}\underline{\mathrm{x}}^T\underline{\mathrm{Q}}(\omega)\underline{\mathrm{x}} - \iota_a^T\underline{\mathrm{x}} - 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.33}$$

$$g_2(\underline{\mathrm{x}}, \omega) \equiv -\frac{1}{2}\underline{\mathrm{x}}^T\underline{\mathrm{Q}}(\omega)\underline{\mathrm{x}} - \iota_a^T\underline{\mathrm{x}} + 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.34}$$

$$g_3(\underline{\mathrm{x}}, \omega) \equiv \underline{\mathrm{A}}_p(\omega)\underline{\mathrm{x}} - \underline{\mathrm{c}}_p(\omega) \leq \underline{0} \quad \forall \omega \in B_p, \tag{1.35}$$

$$g_4(\underline{\mathrm{x}}, \omega) \equiv \underline{\mathrm{A}}_s(\omega)\underline{\mathrm{x}} - \underline{\mathrm{c}}_s(\omega) \leq \underline{0} \quad \forall \omega \in B_s, \tag{1.36}$$

and

$$g_5(\underline{\mathrm{x}}) \equiv \underline{\mathrm{A}}_b\underline{\mathrm{x}} - \underline{\mathrm{c}}_b \leq \underline{0}, \tag{1.37}$$

where α, β, and γ are the weights of different criteria for formulating the objective function, $f(\underline{\mathrm{x}})$ is the objective function, and $g_1(\underline{\mathrm{x}}, \omega)$, $g_2(\underline{\mathrm{x}}, \omega)$, $g_3(\underline{\mathrm{x}}, \omega)$, $g_4(\underline{\mathrm{x}}, \omega)$, and $g_5(\underline{\mathrm{x}})$ are the constraint functions of the optimization problem.

As the set of the filter coefficients satisfying the constraints (1.33) and (1.34) is nonconvex, the optimization problem is a nonconvex optimization problem. In general, it is difficult to find the global minimum of the nonconvex optimization problem.

1.3 Joint Norm Relaxed Sequential Quadratic Programming and Filled Function Method

A joint norm relaxed sequential quadratic programming and the filled function method is proposed for finding the globally optimal solution of Problem (P). The details of the proposed method are discussed below.

1.3.1 Filled Function Method

Some terminologies related to filled functions are discussed below. Notably, a basin of a function is defined as the subset of the domain of the optimization variables such that any points in this subset will yield the same local minimum of the function via conventional gradient based optimization methods. A hill of a function is defined as the subset of the domain of the optimization variables such that any points in this subset will yield the same local maximum of the function via conventional gradient based optimization methods. A higher basin of a function is a basin of the function with the objective functional value of the local minimum of the basin being higher than that of the current basin of the function. A lower basin of a function is a basin of the function with the objective functional value of the local minimum of the basin being lower than that of the current basin of the function.

A filled function is a function satisfying the following properties: (a) the current local minimum of the original objective function is the current local maximum of the filled function; (b) the whole current basin of the original objective function is a part of the current hill of the filled function; (c) the filled function has no stationary point in any higher basins of the original objective function; and (d) there exists a local minimum of the filled function which is in a lower basin of the original objective function.

The working principles of the filled function method are as follows. Due to property (a), by evaluating the filled function at a point slightly deviated from the current local minimum of the original objective function, a lower functional value will be obtained. Hence, the filled function could kick out from the current local minimum of the original objective function. Due to properties (b)–(d), the current local minimum of the filled function is neither in the current basin nor in any higher basins of the original objective function. Hence, the current local minimum of the filled function is in a lower basin of the original objective function. As a result, by finding the next local minimum of the original objective function via conventional gradient based methods with the initial point being the current local minimum of the filled function, a better local minimum of the original objective function can be obtained. Following these procedures, if the original objective function contains a finite number of local minima, then the global minimum of the original objective function will be eventually reached.

Based on the above working principles, the algorithm is summarized below.

Algorithm 1

Step 1: Initialize a minimum improvement factor ε, an acceptable error ε', an initial search point $\underline{\tilde{x}}_1$, a positive definite matrix $\underline{R}$, and an iteration index $k = 1$.

Step 2: Find a local minimum of the following optimization Problem ($\underline{P}_f$) using the norm relaxed sequential quadratic programming method with the initial search point $\underline{\tilde{x}}_k$.

Problem ($\underline{P}_f$)

$$\min_{\underline{x}} f(\underline{x}) = (\alpha \iota_a + \beta \iota_p + \gamma \iota_s)^T \underline{x}, \tag{1.38}$$

subject to

$$g_1(\underline{x}, \omega) = \frac{1}{2} \underline{x}^T \underline{Q}(\omega) \underline{x} - \iota_a^T \underline{x} - 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.39}$$

$$g_2(\underline{x}, \omega) = -\frac{1}{2} \underline{x}^T \underline{Q}(\omega) \underline{x} - \iota_a^T \underline{x} + 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.40}$$

$$g_3(\underline{x}, \omega) = \underline{A}_p(\omega) \underline{x} - \underline{c}_p(\omega) \leq \underline{0} \quad \forall \omega \in B_p, \tag{1.41}$$

$$g_4(\underline{x}, \omega) = \underline{A}_s(\omega) \underline{x} - \underline{c}_s(\omega) \leq \underline{0} \quad \forall \omega \in B_s, \tag{1.42}$$

$$g_5(\underline{x}) = \underline{A}_b \underline{x} - \underline{c}_b \leq \underline{0}, \tag{1.43}$$

$$g_6(\underline{x}) \equiv \iota_a^T \left(\underline{x} - (1 - \varepsilon) \underline{\tilde{x}}_k \right) \leq \underline{0}, \tag{1.44}$$

$$g_7(\underline{x}) \equiv \iota_p^T \left(\underline{x} - (1 - \varepsilon) \underline{\tilde{x}}_k \right) \leq \underline{0}, \tag{1.45}$$

and

$$g_8(\underline{x}) \equiv \iota_s^T \left(\underline{x} - (1 - \varepsilon) \underline{\tilde{x}}_k \right) \leq \underline{0}, \tag{1.46}$$

where $g_6(\underline{x})$, $g_7(\underline{x})$, and $g_8(\underline{x})$ are the constraint functions we imposed. Let us denote the obtained local minimum as $\underline{x}_k^*$.

Step 3: Find a local minimum of the following optimization Problem ($\underline{P}_H$) using the norm relaxed sequential quadratic programming method with the initial search point $\underline{x}_k^*$.

Problem ($\underline{P}_H$)

$$\min_{\underline{x}} H(\underline{x}) \equiv (\alpha \iota_a + \beta \iota_p + \gamma \iota_s)^T \underline{x} + \frac{1}{(\underline{x} - \underline{x}_k^*)^T \underline{R} (\underline{x} - \underline{x}_k^*)}, \tag{1.47}$$

subject to

$$g_1(\underline{x}, \omega) = \frac{1}{2} \underline{x}^T \underline{Q}(\omega) \underline{x} - \iota_a^T \underline{x} - 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.48}$$

$$g_2(\underline{x}, \omega) = -\frac{1}{2} \underline{x}^T \underline{Q}(\omega) \underline{x} - \iota_a^T \underline{x} + 1 \leq 0 \quad \forall \omega \in [-\pi, \pi], \tag{1.49}$$

$$g_3(\underline{x}, \omega) = \underline{A}_p(\omega) \underline{x} - \underline{c}_p(\omega) \leq \underline{0} \quad \forall \omega \in B_p, \tag{1.50}$$

$$g_4(\underline{x}, \omega) = \underline{A}_s(\omega)\underline{x} - \underline{c}_s(\omega) \leq \underline{0} \quad \forall \omega \in B_s, \tag{1.51}$$

$$g_5(\underline{x}) = \underline{A}_b\underline{x} - \underline{c}_b \leq \underline{0}, \tag{1.52}$$

$$g_6'(\underline{x}) \equiv \iota_a^T\left(\underline{x} - (1-\varepsilon)\underline{x}_k^*\right) \leq \underline{0}, \tag{1.53}$$

$$g_7'(\underline{x}) \equiv \iota_p^T\left(\underline{x} - (1-\varepsilon)\underline{x}_k^*\right) \leq \underline{0}, \tag{1.54}$$

and

$$g_8'(\underline{x}) \equiv \iota_s^T\left(\underline{x} - (1-\varepsilon)\underline{x}_k^*\right) \leq \underline{0}, \tag{1.55}$$

where $H(\underline{x})$ is the filled function we defined, $g_6'(\underline{x})$, $g_7'(\underline{x})$, and $g_8'(\underline{x})$ are the constraint functions we imposed. Let us denote the obtained local minimum as $\tilde{\underline{x}}_{k+1}$. Increment the value of k.

Step 4: Iterate Step 2 and Step 3 until

$$\left\|(\alpha\iota_a + \beta\iota_p + \gamma\iota_s)^T\left(\underline{x}_k^* - \underline{x}_{k+1}^*\right)\right\| \leq \varepsilon'. \tag{1.56}$$

Take the final vector of $\underline{x}_k^*$ as the global minimum of the original optimization problem.

Step 1 is an initialization of the proposed algorithm. In order not to terminate the algorithm when the convergence of the algorithm is slow and to have a high accuracy of the solution, both ε and ε' should be chosen as small values. Also, as $\tilde{\underline{x}}_1$ is an initial search point of the optimization algorithm, this initial search point should be in the feasible set. However, in general it is difficult to guarantee that $\tilde{\underline{x}}_1$ is in the feasible set, so it is chosen in such a way that most of the constraints are satisfied. Moreover, as $\underline{R}$ is a positive definite matrix, it controls the spread of the hill of $H(\underline{x})$ at $\underline{x}_k^*$. If $\underline{R}$ is a diagonal matrix with all diagonal elements being the same and positive, then small values of these diagonal elements will result to a wide spread of the hill of $H(\underline{x})$ at $\underline{x}_k^*$ and vice versa. Since the local minima of nonconvex optimization problems could be located far away from each other, the spread of the hill of $H(\underline{x})$ at $\underline{x}_k^*$ should be large and the diagonal elements of $\underline{R}$ should be chosen as small positive numbers. Step 2 is to find a local minimum of $f(\underline{x})$. As the constraints $g_6(\underline{x})$, $g_7(\underline{x})$, and $g_8(\underline{x})$ are imposed on the Problem $(\underline{P}_f)$, the maximum amplitude distortion of the filter bank, the maximum ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter corresponding to the new obtained local minimum are guaranteed to be lower than those corresponding to $\tilde{\underline{x}}_k$. Similarly, Step 3 is to find a local minimum of $H(\underline{x})$. As the constraints $g_6'(\underline{x})$, $g_7'(\underline{x})$, and $g_8'(\underline{x})$ are imposed on the Problem $(\underline{P}_H)$, the maximum amplitude distortion of the filter bank, the maximum ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter are guaranteed to be lower than those corresponding to $\underline{x}_k^*$. Step 4 is a termination test procedure. If the difference of the weighted performance between two consecutive iterations is smaller than a certain bound ε', then the algorithm is terminated.

It has been discussed above that conventional filled function methods require that (a) the current local minimum of the original objective function be the current local

maximum of the filled function; (b) the whole current basin of the original objective function be a part of the current hill of the filled function; (c) the filled function have no stationary point in any higher basins of the original objective function; and (d) there exist a local minimum of the filled function which is in a lower basin of the original objective function. As $\underline{\mathrm{R}}$ is a positive definite matrix and $\underline{\mathrm{x}}_k^*$ is in the denominator of $H(\underline{\mathrm{x}})$, $H(\underline{\mathrm{x}}) \to +\infty$ as $\underline{\mathrm{x}} \to \underline{\mathrm{x}}_k^*$. Hence, $\underline{\mathrm{x}}_k^*$ is the global maximum of $H(\underline{\mathrm{x}})$ and property (a) is guaranteed to be satisfied. As the constraints $g_6'(\underline{\mathrm{x}})$, $g_7'(\underline{\mathrm{x}})$, and $g_8'(\underline{\mathrm{x}})$ are imposed on the Problem $(\underline{\mathrm{P}}_H)$, when a new local minimum of $H(\underline{\mathrm{x}})$ is found, this new local minimum of $H(\underline{\mathrm{x}})$ will not be located at $\underline{\mathrm{x}}_k^*$ and the original objective value evaluated at $\underline{\tilde{\mathrm{x}}}_{k+1}$ will guarantee to be lower than that at $\underline{\mathrm{x}}_k^*$. Hence, properties (b)–(d) are guaranteed to be satisfied. As a result, the proposed algorithm is guaranteed to reach the global minimum of the nonconvex optimization problem.

As the efficiency of general nonconvex optimization algorithms would depend on the initial search points, the total number of local minima of the optimization problems, and the stopping criteria of the optimization algorithms, there is always a tradeoff between the accuracy of the obtained solutions and the efficiency of the optimization algorithms. For nongradient based approaches, as most of the evaluation points are ignored, the effectiveness of these algorithms is low. On the other hand, our proposed method is guaranteed to obtain a local minimum in each iteration, the effectiveness of our proposed algorithm is high. Hence, for the same period of time, our proposed method will obtain a better solution than those of nongradient based approaches.

1.3.2 Norm Relaxed Sequential Quadratic Programming

To find locally optimal solutions of both Problem $(\underline{\mathrm{P}}_f)$ and Problem $(\underline{\mathrm{P}}_H)$, the norm relaxed sequential quadratic programming method is employed. This method is based on the assumptions that the initial point is in the feasible set of the optimization problem and both the objective function and the constraint functions are smooth. The working principles of the norm relaxed sequential quadratic programming method are to find directions of descents via solving quadratic programming problems and to construct a set of new points based on the obtained directions of descents, where the new points are in the feasible set of the original optimization problem and the objective functional values of these new points are monotonic decreasing.

Based on the above working principles, the algorithm is summarized below.

Algorithm 2

Step 1: Denote the initial searching vector for the norm relaxed sequential quadratic programming method obtained from Algorithm 1 as $\overline{\underline{\mathrm{x}}}_0$, initialize constants $\delta_1 > 0$, $\delta_2 > 0$, $\beta_{-1} > 0$, $\sigma \in (0, 1)$, $\alpha \in (0, 1)$, and $\overline{P} > 0$, as well as define a symmetric positive definite matrix $\underline{\mathrm{B}}_0$. Set $k = 0$. By using the integration approach,

the functional inequality constraints can be converted to conventional inequality constraints. Let us define

$$\overline{f}(\underline{x}) \equiv f(\underline{x}) \quad \texttt{for Problem}\ (\underline{P}_f), \tag{1.57}$$

$$\overline{f}(\underline{x}) \equiv H(\underline{x}) \quad \texttt{for Problem}\ (\underline{P}_H), \tag{1.58}$$

$$\overline{g}_j(\underline{x}) \equiv \int \left(\max\left(g_j(\underline{x}, \omega), 0\right)\right)^2 d\omega \quad \texttt{for } j = 1, \ldots, 4, \tag{1.59}$$

$$\overline{g}_5(\underline{x}) \equiv g_5(\underline{x}), \tag{1.60}$$

$$\overline{g}_j(\underline{x}) \equiv g_j(\underline{x}) \quad \texttt{for } j = 6, 7, 8, \texttt{ and for Problem}\ (\underline{P}_f), \tag{1.61}$$

$$\overline{g}_j(\underline{x}) \equiv g'_j(\underline{x}) \quad \texttt{for } j = 6, 7, 8, \texttt{ and for Problem}\ (\underline{P}_H), \tag{1.62}$$

$$P(\underline{x}) \equiv \max\left(0, \overline{g}_1(\underline{x}), \ldots, \overline{g}_8(\underline{x})\right), \tag{1.63}$$

and

$$I_0(\underline{x}) \equiv \left\{ j \in \{1, \ldots, 8\} \quad \texttt{such that} \quad \overline{g}_j(\underline{x}) = P(\underline{x}) \right\}. \tag{1.64}$$

Step 2: Solve the following quadratic programming problem:
Problem $(\underline{QP}_k)$

$$\min_{(\underline{d}, z)} z + \frac{1}{2} \underline{d}^T \underline{B}_k \underline{d}, \tag{1.65}$$

subject to

$$\nabla \overline{f}(\underline{\overline{x}}_k)^T \underline{d} \leq z \tag{1.66}$$

and

$$\overline{g}_j(\underline{\overline{x}}_k) + \nabla \overline{g}_j(\underline{\overline{x}}_k)^T \underline{d} \leq z, \quad \texttt{for } j = 1, \ldots, 8. \tag{1.67}$$

Let us denote the obtained solution as $(\underline{d}_k, z_k)$. If $\underline{d}_k = \underline{0}$, then the algorithm terminates.

Step 3: Let us define

$$\triangle_k \equiv \frac{2P(\underline{\overline{x}}_k) - \underline{d}_k^T \underline{B}_k \underline{d}_k}{\underline{d}_k^T \underline{B}_k \underline{d}_k} + \delta_1, \tag{1.68}$$

$$\beta_k = \beta_{k-1} \quad \texttt{for } \beta_{k-1} \geq \triangle_k, \tag{1.69}$$

$$\beta_k = \triangle_k + \delta_2 \quad \texttt{for } \beta_{k-1} < \triangle_k, \tag{1.70}$$

$$\psi_{\beta_k}(\underline{x}) \equiv \overline{f}(\underline{x}) + \beta_k P(\underline{x}), \tag{1.71}$$

$$P'(\underline{\overline{x}}_k, \underline{d}_k) \equiv \max\left(\nabla \overline{g}_j(\underline{\overline{x}}_k)^T \underline{d}_k \quad \texttt{for } j \in I_0(\underline{\overline{x}}_k), 0\right) \quad \texttt{for } P(\underline{\overline{x}}_k) = 0, \tag{1.72}$$

$$P'(\underline{\overline{x}}_k, \underline{d}_k) \equiv \max\left(\nabla \overline{g}_j(\underline{\overline{x}}_k)^T \underline{d}_k \quad \texttt{for } j \in I_0(\underline{\overline{x}}_k)\right) \quad \texttt{for } P(\underline{\overline{x}}_k) > 0, \tag{1.73}$$

and

$$\psi'_{\beta_k}(\underline{\overline{x}}_k, \underline{d}_k) \equiv \nabla \overline{f}(\underline{\overline{x}}_k)^T \underline{d}_k + \beta_k P'(\underline{\overline{x}}_k, \underline{d}_k). \tag{1.74}$$

Find the step size t_k which is defined as the first value in the sequence $\{1, \sigma, \sigma^2, \ldots\}$ such that:

if $P(\underline{\overline{x}}_k) \leq \overline{P}$, then

$$\psi_{\beta_k}(\underline{\overline{x}}_k + t_k \underline{d}_k) \leq \psi_{\beta_k}(\underline{\overline{x}}_k) + \alpha t_k \psi'_{\beta_k}(\underline{\overline{x}}_k, \underline{d}_k), \tag{1.75}$$

if $P(\underline{\overline{x}}_k) > \overline{P}$, then

$$\psi_{\beta_k}(\underline{\overline{x}}_k + t_k \underline{d}_k) \leq \psi_{\beta_k}(\underline{\overline{x}}_k) + \alpha t_k \psi'_{\beta_k}(\underline{\overline{x}}_k, \underline{d}_k), \tag{1.76}$$

and

$$P(\underline{\overline{x}}_k + t_k \underline{d}_k) \leq P(\underline{\overline{x}}_k). \tag{1.77}$$

Step 4: Find a new symmetric positive definite matrix $\underline{B}_{k+1}$ using existing algorithms. Set

$$\underline{\overline{x}}_{k+1} = \underline{\overline{x}}_k + t_k \underline{d}_k. \tag{1.78}$$

Increment the value of k and go back to Step 2.

Step 1 is an initialization of the norm relaxed sequential quadratic programming method. Step 2 is to find the directions of descents via solving quadratic programming problems. The constraints imposed in Problem $(\underline{QP}_k)$ guarantee that the objective functional values of the obtained solutions are monotonic decreasing and within the feasible set of the original optimization problem. As a result, the converged solution of the quadratic programming problems is guaranteed to be a locally optimal solution of the original optimization problem.

As the quadratic programming problems are only subject to linear constraints, the corresponding dual problems only involve simple functions of these Lagrange multipliers. Hence, the quadratic programming problems can be solved via efficient algorithms such as interior point methods. As a result, the proposed method is very efficient and effective for finding locally optimal solutions of the optimization problems in each iteration of the filled function method.

1.4 Computer Numerical Simulation Results

In order to have a fair comparison, the performance of the QMF banks designed via our proposed method is compared to that designed via existing minimax approaches [23]. We choose the same passband, stopband, filter length, maximum

passband ripple magnitude, maximum stopband ripple magnitude, and desirable magnitude response of the prototype filter as that in [23], that is,

$$B_p = [-0.4\pi, 0.4\pi], \tag{1.79}$$

$$B_s = [0.6\pi, \pi] \cup [-\pi, -0.6\pi], \tag{1.80}$$

$$N = 36, \tag{1.81}$$

$$\varepsilon_p = -50 \text{ dB}, \tag{1.82}$$

$$\varepsilon_s = -50 \text{ dB}, \tag{1.83}$$

$$D(\omega) = 1 \quad \texttt{for } \omega \in B_p, \tag{1.84}$$

and

$$D(\omega) = 0 \quad \texttt{for } \omega \in B_s. \tag{1.85}$$

In order to guarantee that the performance of the QMF bank designed via our proposed method is better than that in [23], the specification on the maximum amplitude distortion of the filter bank is chosen as $\varepsilon_a = -58$ dB, which is better than that in [23] ($\varepsilon_a = 0.003 = -50.4576$ dB). In order not to have any bias among the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter, all the weights in the objective function are chosen to be the same, that is, $\alpha = \beta = \gamma = 1$. In this chapter, $\varepsilon = \varepsilon' = 10^{-6}$ are chosen, which is small enough for most applications. $\tilde{\underline{x}}_1$ is chosen as the filter coefficients obtained via the Remez exchange algorithm, which is guaranteed to satisfy the specifications on the maximum passband ripple magnitude and the maximum stopband ripple magnitude of the prototype filter. $\underline{R}$ is chosen as the diagonal matrix with all diagonal elements equal to 10^{-3}, which is small enough for most applications.

To compare the efficiency of the designed method, our proposed method only takes three iterations to converge and the total time required for the computer numerical simulations is 0.8 seconds. On the other hand, the method discussed in [23] takes 68 iterations to converge and the total time required for the computer numerical simulations is 80 seconds. Hence, it can be concluded that the method discussed in [23] requires more computational efforts than our proposed method and our proposed method is more efficient than that discussed in [23]. The magnitude responses of the filter banks as well as the magnitude responses of the prototype filters in both the passband and the stopband designed via our proposed method are shown in Figs. 1.1, 1.2, 1.3. It can be seen from these figures that the prototype filter designed by our proposed method can achieve $\delta_p = -64.2416$ dB and $\delta_s = -50.3625$ dB, and the QMF bank could achieve $\delta_a = -58.1557$ dB. It can be checked easily that the QMF bank designed via our proposed method achieves better performance with respect to the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude ripple of the prototype filter than that designed by the method discussed in [23]. This is because the QMF bank designed by the method discussed in [23] is not the global minimum, while that designed by our proposed method is the global minimum.

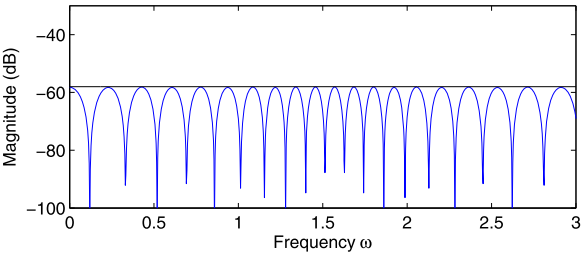


Fig. 1.1 Magnitude response of the filter bank

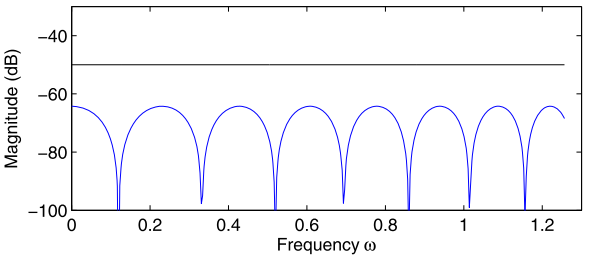


Fig. 1.2 Magnitude response of the prototype filter in the passband

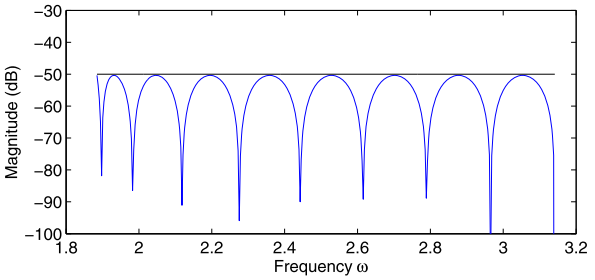


Fig. 1.3 Magnitude response of the prototype filter in the stopband

1.5 Conclusions

This chapter proposes a joint norm relaxed sequential quadratic programming and filled function method for the design of a two-channel linear phase FIR QMF bank so that a weighted sum of the maximum amplitude distortion of the filter bank, the maximum passband ripple magnitude, and the maximum stopband ripple magnitude of the prototype filter is minimized. In particular, a locally optimal solution of the optimization problem in each iteration of the filled function method is found by the norm relaxed sequence quadratic programming method. The globally optimal solution of the original optimization problem is found by the filled function method. Computer numerical simulation results show that the proposed method can find the global minimum of the nonconvex optimization problem efficiently.

References

1. Benmesbah, L., Ling, B.W.K., Chandrasekhar, V., Chu, X.L., Dohler, M.: Optimal decentralized spectral resource allocation for OFDMA downlink of femto networks via adaptive gradient vector step size approach. Am. J. Eng. Appl. Sci. (2011)
2. Chen, X.B., Kostreva, M.M.: A generalization of the norm-relaxed method of feasible directions. Appl. Math. Comput. (1999)
3. Ge, R.P.: A filled function method for finding a global minimizer of a function of several variables. Math. Program. (1990)
4. Ge, R.P., Qin, Y.F.: A class of filled functions for finding global minimizers of a function of several variables. J. Optim. Theory Appl. (1987)
5. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Optimal design of nonuniform FIR transmultiplexer using semi-infinite programming. IEEE Trans. Signal Process. (2005)
6. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Design of nonuniform near allpass complementary FIR filters via a semi-infinite programming technique. IEEE Trans. Signal Process. (2005)
7. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Efficient algorithm for solving semi-infinite programming problems and their applications to nonuniform filter bank designs. IEEE Trans. Signal Process. (2006)
8. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Optimal design of magnitude responses of rational infinite impulse response filters. IEEE Trans. Signal Process. (2006)
9. Ho, C.Y.F., Ling, B.W.K., Reiss, J.D., Liu, Y.Q., Teo, K.L.: Design of interpolative sigma–delta modulators via semi-infinite programming. IEEE Trans. Signal Process. (2006)
10. Ho, C.Y.F., Ling, B.W.K., Chi, Z.W., Shikh-Bahaei, M., Liu, Y.Q., Teo, K.L.: Design of near-allpass strictly stable minimal-phase real-valued rational IIR filters. IEEE Trans. Circuits Syst. II, Express Briefs (2008)
11. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Optimum design of discrete-time differentiators via semi-infinite programming approach. IEEE Trans. Instrum. Meas. (2008)
12. Ho, C.Y.F., Ling, B.W.K., Benmesbah, L., Kok, T.C.W., Siu, W.C., Teo, K.L.: Two-channel linear phase FIR QMF bank minimax design via global nonconvex optimization programming. IEEE Trans. Signal Process. (2010)
13. Ho, C.Y.F., Ling, B.W.K., Giovanni, S.B., Chi, Z.W., Siu, W.C.: Single step optimal block matched motion estimation with motion vectors having arbitrary pixel precisions. Am. J. Eng. Appl. Sci. (2011)
14. Ho, C.Y.F., Ling, B.W.K., Dam, H.H.H., Teo, K.L.: Minimax passband group delay nonlinear phase peak constrained FIR filter design without imposing desired phase response. Int. J. Innov. Comput. Inf. Control (2012)
15. Jian, J.B., Zheng, H.Y., Hu, Q.J., Tang, C.M.: A new norm-relaxed method of strongly sub-feasible direction for inequality constrained optimization. Appl. Math. Comput. (2005)
16. Jian, J.B., Hu, Q.J., Han, D.L.: A strongly convergent norm-relaxed method of strongly sub-feasible direction for optimization with nonlinear equality and inequality constraints. Appl. Math. Comput. (2006)
17. Jian, J.B., Zheng, H.Y., Tang, C.M., Hu, Q.J.: A new superlinearly convergent norm-relaxed method of strongly sub-feasible direction for inequality constrained optimization. Appl. Math. Comput. (2006)
18. Jian, J.B., Hu, Q.J., Han, D.L.: A norm-relaxed method of feasible directions for finely discretized problems from semi-infinite programming. Eur. J. Oper. Res. (2008)
19. Jian, J.B., Ke, X.Y., Cheng, W.X.: A superlinearly convergent norm-relaxed SQP method of strongly sub-feasible directions for constrained optimization without strict complementarity. Appl. Math. Comput. (2009)
20. Jian, J.B., Ke, X.Y., Zheng, H.Y., Tang, C.M.: A method combining norm-relaxed QP subproblems with systems of linear equations for constrained optimization. J. Comput. Appl. Math. (2009)

21. Jian, J.B., Tang, C.M., Zheng, H.Y.: Sequential quadratically constrained quadratic programming norm-relaxed algorithm of strongly sub-feasible directions. Eur. J. Oper. Res. (2010)
22. Johnston, J.D.: A filter family designed for use in quadrature mirror filter banks. In: International Conference on Acoustics, Speech, and Signal Processing (1980)
23. Kok, C.W., Siu, W.C., Law, Y.M.: Peak constrained least-squares QMF banks. Signal Process. (2008)
24. Lin, Y.P., Vaidyanathan, P.P.: Linear phase cosine modulated maximally decimated filter banks with perfect reconstruction. In: International Symposium on Circuits and Systems (1994)
25. Liu, X.: Finding global minima with a computable filled function. J. Glob. Optim. (2001)
26. Nguyen, T.Q.: Near-perfect-reconstruction pseudo-QMF banks. IEEE Trans. Signal Process. (1994)
27. Samadi, P., Ahmadi, M.: Genetic algorithm and its application for the design of QMF banks with canonical signed digit coefficients: a comparative study and new results. In: IEEE Northeast Workshop on Circuits and Systems (2007)
28. Subramaniam, S.R., Ling, B.W.K., Georgakis, A.: Filtering in rotated time-frequency domains with unknown noise statistics. IEEE Trans. Signal Process. (2012)
29. Uppalapati, H., Rastgar, H., Ahmadi, M., Sid-Ahmed, M.A.: Design of quadrature mirror filter banks with canonical signed digit coefficients using genetic algorithm. In: International Conference on Communications, Circuits and Systems (2005)
30. Wu, Z.Y., Lee, H.W.J., Zhang, L.S., Yang, X.M.: A novel filled function method and quasi-filled function method for global optimization. Comput. Optim. Appl. (2005)
31. Yiu, K.F.C., Liu, Y., Teo, K.L.: A hybrid descent method for global optimization. J. Glob. Optim. (2004)
32. Yiu, C.K.F., Grbic, N., Nordholm, S., Teo, K.L.: A hybrid method for the design of oversampled uniform DFT filter banks. Signal Process. (2006)
33. Zhang, Y., Sheng, L., Xu, Y.T.: New filled functions for nonsmooth global optimization. Appl. Math. Model. (2009)
34. Zheng, H.Y., Jian, J.B., Tang, C.M., Quan, R.: A new norm-relaxed SQP algorithm with global convergence. Appl. Math. Lett. (2010)

Chapter 2
Robust Reduced-Rank Adaptive LCMV Beamforming Algorithms Based on Joint Iterative Optimization of Parameters

Rodrigo C. de Lamare

Abstract This chapter presents robust reduced-rank linearly constrained minimum variance (LCMV) beamforming algorithms based on the concept of joint iterative optimization of parameters. The proposed robust reduced-rank scheme is based on a constrained robust joint iterative optimization (RJIO) of parameters according to the minimum variance criterion. The robust optimization procedure adjusts the parameters of a rank-reduction matrix, a reduced-rank beamformer, and the diagonal loading in an alternating manner. LCMV expressions are developed for the design of the rank-reduction matrix and the reduced-rank beamformer. Stochastic gradient and recursive least-squares adaptive algorithms are then devised for an efficient implementation of the RJIO robust beamforming technique. Simulations for a beamforming application in the presence of uncertainties show that the RJIO scheme and algorithms outperform existing algorithms in convergence and tracking performances while they require a comparable computational complexity.

2.1 Introduction

In the last decade, adaptive beamforming techniques have attracted significant interest from researchers and engineers, and found applications in radar, sonar, wireless communications, and seismology [1, 2]. The optimal linearly constrained minimum variance (LCMV) beamformer is designed in such a way that it minimizes the array output power while maintaining a constant response in the direction of a signal of interest (SoI) [1–3]. However, this technique requires the computation of the inverse of the input data covariance matrix and the knowledge of the array steering vector. Adaptive versions of the LCMV beamformer were subsequently reported with stochastic gradient (SG) [4, 5] and recursive least-squares (RLS) [6] algorithms. A key problem with adaptive beamforming techniques is the impact of uncertainties which can result in a considerable performance degradation. These mismatches are

R.C. de Lamare (✉)
Communications Research Group, Department of Electronics, University of York, York Y010 5DD, UK
e-mail: rcdl500@york.ac.uk

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_2,

caused by local scattering, imperfectly calibrated arrays, insufficient training, and imprecisely known wave field propagation conditions [2].

In the last decades, a number of robust approaches have been reported that address this problem [7–30]. These techniques can be classified according to the approach adopted to deal with the mismatches: techniques based on diagonal loading [7, 9, 12, 13], methods that estimate the mismatch or, equivalently, the actual steering vector [10, 11, 14], and techniques that exploit properties such as the constant modulus of the signals [15–17] and the low-rank of the interference subspace [8, 18–30]. Furthermore, beamforming algorithms usually have a trade-off between performance and computational complexity which depends on the designer's choice of the adaptation algorithm [3, 31]. A number of robust designs can be cast as optimization problems which end up in the so-called second-order cone (SOC) program, which can be solved with interior point methods and have a computational cost that is super-cubic in the number of parameters of the beamformer. This poses a problem for beamforming systems that have a large number of parameters and operate in time-varying scenarios, which requires the beamformer to be recomputed periodically.

A robust technique for short-data record scenarios is reduced-rank signal processing [18–30], which is very well suited for systems with a large number of parameters. These algorithms are robust against short data records, have the ability to exploit the low-rank nature of the signals encountered in beamforming applications, and can resist moderate steering vector mismatches. These methods include the computationally expensive eigen-decomposition techniques [18, 19] to alternative approaches such as the Auxiliary Vector Filter (AVF) [20, 25], the Multi-stage Wiener Filter (MSWF) [21, 23, 24] which are based on the Krylov subspace, and joint iterative optimization (JIO) approaches [22, 26–29]. The JIO techniques reported in [26, 27, 29] outperform the eigen-decomposition- and Krylov-based methods and are amenable to efficient adaptive implementations. However, robust versions of JIO methods have not been considered so far.

In this chapter, robust LCMV reduced-rank beamforming algorithms based on constrained robust joint iterative optimization (RJIO) of parameters are developed. The basic idea of the RJIO approach is to design a bank of robust adaptive beamformers which is responsible for performing dimensionality reduction, whereas the robust reduced-rank beamformer effectively forms the beam in the direction of the SoI and takes into account the uncertainty. Robust LCMV expressions for the design of the rank reduction matrix and the reduced-rank beamformer are proposed that can appropriately deal with array steering vector mismatches. SG and RLS algorithms for efficiently implementing the method are then devised. An automatic rank adaptation algorithm for determining the most adequate rank for the RJIO algorithms is described. A simulation study of the proposed RJIO algorithms and existing techniques is considered.

This chapter is organized as follows. The system and signals models are described in Sect. 2.2. The full-rank and the reduced-rank LCMV filtering problems are formulated in Sect. 2.3. Section 2.4 is dedicated to the RJIO method, whereas Sect. 2.5 is devoted to the derivation of the adaptive SG and RLS algorithms, the

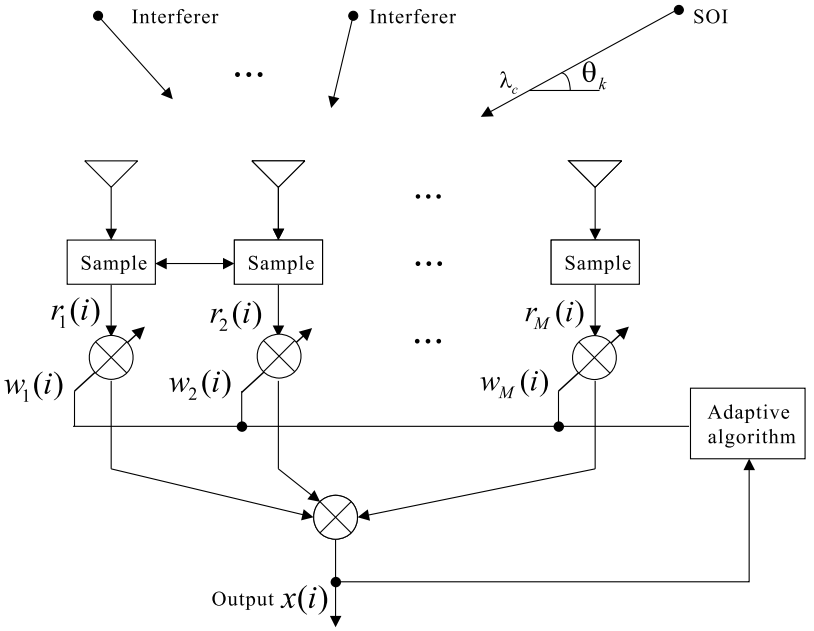


Fig. 2.1 Block diagram of a sensor-array array system with interfering signals

analysis of the computational complexity, and the rank adaptation technique. Section 2.6 presents and discusses the simulation results and Sect. 2.7 gives the concluding remarks.

2.2 System Model

Let us consider a sensor-array system equipped with a uniform linear array (ULA) of M elements, as shown in Fig. 2.1. Assuming that the sources are in the far field of the array, the signals of K narrowband sources impinge on the array ($K < M$) with unknown directions of arrival (DOA) θ_l for $l = 1, 2, \ldots, K$.

The input data from the antenna array can be organized in an $M \times 1$ vector expressed by

$$\boldsymbol{r}(i) = \boldsymbol{A}(\theta)\boldsymbol{s}(i) + \boldsymbol{n}(i), \tag{2.1}$$

where

$$\boldsymbol{A}(\theta) = \left[\boldsymbol{a}(\theta_1), \ldots, \boldsymbol{a}(\theta_K)\right]$$

is the $M \times K$ matrix of signal steering vectors. The $M \times 1$ signal steering vector is defined as

$$\boldsymbol{a}(\theta_l) = \left[1, e^{-2\pi j \frac{d_x}{\lambda_c}\cos\theta_l}, \ldots, e^{-2\pi j (M-1)\frac{d_x}{\lambda_c}\cos\theta_l}\right]^T \tag{2.2}$$

for a signal impinging at angle θ_l, $l = 1, 2, \ldots, K$, where $d_s = \lambda_c/2$ is the inter-element spacing, λ_c is the wavelength and $(\cdot)^T$ denotes the transpose operation. The vector $\boldsymbol{n}(i)$ denotes the complex vector of sensor noise, which is assumed to be zero-mean and Gaussian with covariance matrix $\sigma^2\boldsymbol{I}$.

2.3 Problem Statement and Design of Adaptive Beamformers

In this section, the problem of designing robust beamforming algorithms against steering vector mismatches is stated. The design of robust full-rank and reduced-rank LCMV beamformers is introduced along with the modeling of steering vector mismatches. The presumed array steering vector for the kth desired signal is given by $\boldsymbol{a}_p(\theta_k) = \boldsymbol{a}(\theta_k) + \boldsymbol{e}$, where $\boldsymbol{e}$ is the $M \times 1$ mismatch vector and $\boldsymbol{a}(\theta_k)$ is the actual array steering vector which is unknown for the system. By using the presumed array steering vector $\boldsymbol{a}_p(\theta_k)$, the performance of a conventional LCMV beamformer can be degraded significantly. The problem of interest is how to design a beamformer that can deal with the mismatch and minimize the performance loss due to the uncertainty.

2.3.1 Adaptive LCMV Beamformers

In order to perform beamforming with a full-rank LCMV beamformer, we linearly combine the data vector $\boldsymbol{r}(i) = [r_1^{(i)}\ r_2^{(i)}\ \ldots\ r_M^{(i)}]^T$ with the full-rank beamformer $\boldsymbol{w} = [w_1\ w_2\ \ldots\ w_M]^T$ to yield

$$x(i) = \boldsymbol{w}^H \boldsymbol{r}(i). \tag{2.3}$$

The optimal LCMV beamformer is described by the $M \times 1$ vector $\boldsymbol{w}$, which is designed to solve the following optimization problem

$$\begin{aligned} &\text{minimize} \quad E\left[\left|\boldsymbol{w}^H \mathbf{r}(i)\right|^2\right] = \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ &\text{subject to} \quad \boldsymbol{w}^H \boldsymbol{a}(\theta_k) = 1. \end{aligned} \tag{2.4}$$

The solution to the problem in (2.4) is given by [3, 4]

$$\boldsymbol{w}_{\text{opt}} = \frac{\boldsymbol{R}^{-1}\boldsymbol{a}(\theta_k)}{\boldsymbol{a}^H(\theta_k)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta_k)}, \tag{2.5}$$

where $\boldsymbol{a}(\theta_k)$ is the steering vector of the SoI, $\boldsymbol{r}(i)$ is the received data, the covariance matrix of $\boldsymbol{r}(i)$ is described by $\boldsymbol{R} = E[\boldsymbol{r}(i)\boldsymbol{r}^H(i)]$, $(\cdot)^H$ denotes Hermitian transpose and $E[\cdot]$ stands for the expected value. The beamformer $\boldsymbol{w}(i)$ can be estimated via SG or RLS algorithms [3]. However, the laws that govern their convergence and tracking behaviors imply that they depend on M and on the eigenvalue spread of $\boldsymbol{R}$.

A reduced-rank algorithm must extract the most important features of the processed data by performing dimensionality reduction. This mapping is carried out by a $M \times D$ rank-reduction matrix $\boldsymbol{S}_D$ on the received data as given by

$$\bar{\boldsymbol{r}}(i) = \boldsymbol{S}_D^H \boldsymbol{r}(i), \tag{2.6}$$

where, in what follows, all D-dimensional quantities are denoted with a "bar". The resulting projected received vector $\bar{\boldsymbol{r}}(i)$ is the input to a beamformer represented by the $D \times 1$ vector $\bar{\boldsymbol{w}} = [\bar{w}_1 \ \bar{w}_2 \ \ldots \ \bar{w}_D]^T$. The filter output is

$$\bar{x}(i) = \bar{\boldsymbol{w}}^H \bar{\boldsymbol{r}}(i). \tag{2.7}$$

In order to design a reduced-rank beamformer $\bar{\boldsymbol{w}}$ we consider the following optimization problem

$$\begin{aligned} &\text{minimize} \quad E\left[\left|\bar{\boldsymbol{w}}^H \bar{\boldsymbol{r}}(i)\right|^2\right] = \bar{\boldsymbol{w}}^H \bar{\boldsymbol{R}} \bar{\boldsymbol{w}} \\ &\text{subject to} \quad \bar{\boldsymbol{w}}^H \bar{\boldsymbol{a}}(\theta_k) = 1. \end{aligned} \tag{2.8}$$

The solution to the above problem is

$$\begin{aligned} \bar{\boldsymbol{w}}_{\text{opt}} &= \frac{\bar{\boldsymbol{R}}^{-1} \bar{\boldsymbol{a}}(\theta_k)}{\bar{\boldsymbol{a}}^H(\theta_k) \bar{\boldsymbol{R}}^{-1} \bar{\boldsymbol{a}}(\theta_k)} \\ &= \frac{(\boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D)^{-1} \boldsymbol{S}_D^H \boldsymbol{a}(\theta_k)}{\boldsymbol{a}^H \boldsymbol{S}_D(\theta_k)(\boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D)^{-1} \boldsymbol{S}_D^H \boldsymbol{a}(\theta_k)}, \end{aligned} \tag{2.9}$$

where the reduced-rank covariance matrix is $\bar{\boldsymbol{R}} = E[\bar{\boldsymbol{r}}(i)\bar{\boldsymbol{r}}^H(i)] = \boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D$ and the reduced-rank steering vector is $\bar{\boldsymbol{a}}(\theta_k) = \boldsymbol{S}_D^H \boldsymbol{a}(\theta_k)$. The above development shows that the choice of $\boldsymbol{S}_D$ to perform dimensionality reduction on $\boldsymbol{r}(i)$ is very important, and can lead to an improved convergence and tracking performance over the full-rank beamformer. A key problem with the full-rank and the reduced-rank beamformers described in (2.5) and (2.9), respectively, is that their performance is deteriorated when they employ the presumed array steering vector $\boldsymbol{a}_p(\theta_k)$. In these situations, it is fundamental to employ a robust technique that can mitigate the effects of the mismatches between the actual and the presumed steering vector.

2.3.2 Robust Adaptive LCMV Beamformers

An effective technique for robust beamforming is the use of diagonal loading strategies [7, 9, 12, 13]. In what follows, robust full-rank and reduced-rank LCMV beamforming designs are described. A general approach based on diagonal loading is employed for both full-rank and reduced-rank designs.

A robust full-rank LCMV beamformer represented by an $M \times 1$ vector $\boldsymbol{w}$ can be designed by solving the following optimization problem

$$\begin{aligned} &\text{minimize} \quad E\left[\left|\boldsymbol{w}^H \mathbf{r}(i)\right|^2\right] + \varepsilon^2 \|\boldsymbol{w}\|^2 = \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} + \varepsilon^2 \boldsymbol{w}^H \boldsymbol{w} \\ &\text{subject to} \quad \boldsymbol{w}^H \boldsymbol{a}(\theta_k) = 1, \end{aligned} \tag{2.10}$$

where ε^2 is a constant that needs to be chosen by the designer. The solution to the problem in (2.10) is given by

$$\boldsymbol{w}_{\text{opt}} = \frac{(\boldsymbol{R} + \varepsilon^2 \boldsymbol{I}_M)^{-1} \boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H(\theta_k)(\boldsymbol{R} + \varepsilon^2 \boldsymbol{I}_M)^{-1} \boldsymbol{a}_p(\theta_k)}, \tag{2.11}$$

where $\boldsymbol{a}_p(\theta_k)$ is the presumed steering vector of the SoI and $\boldsymbol{I}_D$ is an M-dimensional identity matrix. It turns out that the adjustment of ε^2 needs to be obtained numerically by an optimization algorithm.

In order to design a robust reduced-rank LCMV beamformer $\bar{\boldsymbol{w}}$, we follow a similar approach to the full-rank case and consider the following optimization problem

$$\begin{aligned} &\text{minimize} \quad E\left[\left|\bar{\boldsymbol{w}}^H \boldsymbol{S}_D^H \boldsymbol{r}(i)\right|^2\right] + \varepsilon^2 \|\boldsymbol{S}_D \bar{\boldsymbol{w}}\|^2 = \bar{\boldsymbol{w}}^H \boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D \bar{\boldsymbol{w}} \\ &\qquad\qquad\qquad\qquad\qquad\qquad + \varepsilon^2 \bar{\boldsymbol{w}}^H \boldsymbol{S}_D^H \boldsymbol{S}_D \bar{\boldsymbol{w}} \\ &\text{subject to} \quad \bar{\boldsymbol{w}}^H \boldsymbol{S}_D^H \boldsymbol{a}_p(\theta_k) = 1. \end{aligned} \tag{2.12}$$

The solution to the above problem is

$$\bar{\boldsymbol{w}}_{\text{opt}} = \frac{(\boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D + \varepsilon^2 \boldsymbol{I}_D)^{-1} \boldsymbol{S}_D^H \boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H \boldsymbol{S}_D(\theta_k)(\boldsymbol{S}_D^H \boldsymbol{R} \boldsymbol{S}_D + \varepsilon^2 \boldsymbol{I}_D)^{-1} \boldsymbol{S}_D^H \boldsymbol{a}_p(\theta_k)}, \tag{2.13}$$

where the tuning of ε^2 requires an algorithmic approach as there is no closed-form solution and $\boldsymbol{I}_D$ is a D-dimensional identity matrix.

2.4 Robust Reduced-Rank Beamforming Based on Joint Iterative Optimization of Parameters

In this section, the principles of the robust reduced-rank beamforming scheme based on joint iterative optimization of parameters, termed RJIO, are introduced. The RJIO scheme, depicted in Fig. 2.2, employs a rank-reduction matrix $\boldsymbol{S}_D(i)$ with dimensions $M \times D$ to perform dimensionality reduction on a data vector $\boldsymbol{r}(i)$ with dimensions $M \times 1$. The reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$ with dimensions $D \times 1$ processes the reduced-rank data vector $\bar{\boldsymbol{r}}(i)$ in order to yield a scalar estimate $\bar{x}(i)$. The rank-reduction matrix $\mathbf{S}_D(i)$ and the reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$ are jointly optimized in the RJIO scheme according to the MV criterion subject to a robust constraint that

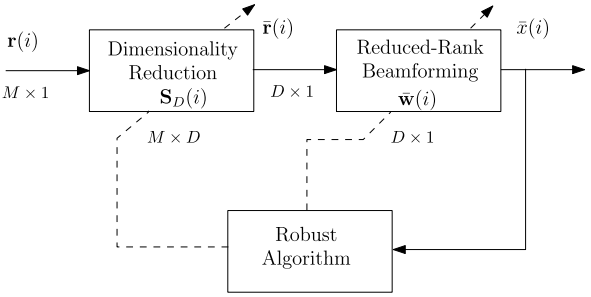


Fig. 2.2 Block diagram of the RJIO scheme

ensures that the beamforming algorithm is robust against steering vector mismatches and short data records.

In order to describe the RJIO method, let us first consider the structure of the $M \times D$ rank-reduction matrix

$$\boldsymbol{S}_D(i) = \big[\boldsymbol{s}_1(i) \mid \boldsymbol{s}_2(i) \mid \ \ldots \ \mid \boldsymbol{s}_D(i)\big], \tag{2.14}$$

where the columns $\boldsymbol{s}_d(i)$ for $d = 1, \ldots, D$ constitute a bank of D robust beamformers with dimensions $M \times 1$ as given by

$$\boldsymbol{s}_d(i) = \big[s_{1,d}(i)\ s_{2,d}(i)\ \ \ldots\ s_{M,d}(i)\big]^T.$$

The output $\bar{x}(i)$ of the RJIO scheme can be expressed as a function of the input vector $\mathbf{r}(i)$, the matrix $\mathbf{S}_D(i)$ and the reduced-rank beamformer $\bar{\mathbf{w}}(i)$:

$$\bar{x}(i) = \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i) = \bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{r}}(i). \tag{2.15}$$

It is interesting to note that for $D = 1$, the RJIO scheme becomes a robust full-rank LCMV beamforming scheme with an additional weight parameter w_D that provides an amplitude gain. For $D > 1$, the signal processing tasks are changed and the robust full-rank LCMV beamformers compute a subspace projection and the reduced-rank beamformer provides a unity gain in the direction of the SoI. This rationale is fundamental to the exploitation of the low-rank nature of signals in typical beamforming scenarios.

The robust LCMV expressions for $\boldsymbol{S}_D(i)$ and $\bar{\boldsymbol{w}}(i)$ can be computed via the following optimization problem

$$\begin{aligned} \text{minimize} \quad & E\big[\big|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)\big|^2\big] + \varepsilon^2\big\|\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\big\|^2 \\ & = \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) + \varepsilon^2\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) \\ \text{subject to} \quad & \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k) = 1. \end{aligned} \tag{2.16}$$

In order to solve the above problem, we resort to the method of Lagrange multipliers [3] and transform the constrained optimization into an unconstrained one

expressed by the Lagrangian

$$\begin{aligned}\mathscr{L}\big(\boldsymbol{S}_D(i), \bar{\boldsymbol{w}}(i), \varepsilon^2(i)\big) = E\big[\big|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)\big|^2\big] \\ + \varepsilon^2(i)\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) \\ + \big[\lambda(\bar{\boldsymbol{w}}^H(i)\mathbf{S}_D^H(i)\boldsymbol{a}_p(\theta_k) - 1\big],\end{aligned} \tag{2.17}$$

where λ is a scalar Lagrange multiplier, $*$ denotes complex conjugate. By fixing $\bar{\boldsymbol{w}}(i)$, minimizing (2.17) with respect to $\boldsymbol{S}_D(i)$, and solving for λ, we get

$$\boldsymbol{S}_D(i) = \frac{(\boldsymbol{R} + \varepsilon^2(i)\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{R}}_{\bar{w}}^{-1}}{\bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{R}}_{\bar{w}}^{-1}\bar{\boldsymbol{w}}(i)\boldsymbol{a}_p^H(\theta_k)(\boldsymbol{R}(i) + \varepsilon^2(i)\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k)}, \tag{2.18}$$

where $\boldsymbol{R} = E[\boldsymbol{r}(i)\boldsymbol{r}^H(i)]$ and $\bar{\boldsymbol{R}}_{\bar{w}} = E[\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i)]$. By fixing $\boldsymbol{S}_D(i)$, minimizing (2.17) with respect to $\bar{\boldsymbol{w}}(i)$, and solving for λ, we arrive at the expression

$$\bar{\boldsymbol{w}}(i) = \frac{(\bar{\boldsymbol{R}}(i) + \varepsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{I}_D\boldsymbol{S}_D(i))^{-1}\bar{\boldsymbol{a}}_p(\theta_k)}{\bar{\boldsymbol{a}}_p^H(\theta_k)(\bar{\boldsymbol{R}}(i) + \varepsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{I}_D\boldsymbol{S}_D(i))^{-1}\bar{\boldsymbol{a}}_p(\theta_k)}, \tag{2.19}$$

where $\bar{\boldsymbol{R}}(i) = E[\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)\boldsymbol{r}^H(i)\boldsymbol{S}_D(i)] = E[\bar{\boldsymbol{r}}(i)\bar{\boldsymbol{r}}^H(i)]$, $\bar{\boldsymbol{a}}_p(\theta_k) = \boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)$. Note that the filter expressions in (2.18) and (2.19) are not closed-form solutions for $\bar{\boldsymbol{w}}(i)$ and $\boldsymbol{S}_D(i)$ since (2.18) is a function of $\bar{\boldsymbol{w}}(i)$ and (2.19) depends on $\boldsymbol{S}_D(i)$. Thus, it is necessary to iterate (2.18) and (2.19) with initial values to obtain a solution [32]. The key strategy lies in the robust joint optimization of the beamformers. The rank D and the diagonal loading parameter $\varepsilon^2(i)$ must be adjusted by the designer to ensure appropriate performance or can be estimated via another algorithm. In the next section, iterative solutions via adaptive algorithms are sought for the robust computation of $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$, the diagonal loading $\varepsilon(i)$, and the rank adaptation.

2.5 Adaptive Algorithms

In this section, adaptive SG and RLS versions of the RJIO scheme are developed for an efficient implementation. The important issue of determining the rank of the scheme with an adaptation technique is considered. The computational complexity in arithmetic operations of the RJIO-based algorithms is then detailed.

2.5.1 Stochastic Gradient Algorithm

In this part, we present a low-complexity SG adaptive reduced-rank algorithm for an efficient implementation of the RJIO method. The basic idea is to employ an alternating optimization strategy to update $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$, and the diagonal loading $\varepsilon^2(i)$.

By computing the instantaneous gradient terms of (2.17) with respect to $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$, and $\varepsilon^2(i)$, we obtain

$$\begin{aligned}
\nabla \mathscr{L}_{MV\,\boldsymbol{S}_D^*(i)} &= \bar{x}^*(i)\boldsymbol{r}(i)\bar{\boldsymbol{w}}^H(i) + \varepsilon^2(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i) + 2\lambda^*\mathbf{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i),\\
\nabla \mathscr{L}_{MV\,\bar{\boldsymbol{w}}^*(i)} &= \bar{x}^*(i)\boldsymbol{S}_D^H(i)\mathbf{r}(i) + \varepsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) + 2\lambda^*\boldsymbol{S}_D^H(i)\mathbf{a}_p(\theta_k),\\
\nabla \mathscr{L}_{MV\,\varepsilon^2(i)} &= 2\varepsilon(i)\boldsymbol{w}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i).
\end{aligned} \tag{2.20}$$

By introducing the positive step sizes μ_s, μ_w, and μ_ε, using the gradient rules $\boldsymbol{S}_D(i+1) = \boldsymbol{S}_D(i) - \mu_s \nabla \mathscr{L}_{MV\,\boldsymbol{S}_D^*(i)}$, $\bar{\boldsymbol{w}}(i+1) = \bar{\boldsymbol{w}}(i) - \mu_w \nabla \mathscr{L}_{MV\,\bar{\boldsymbol{w}}^*(i)}$ and $\varepsilon(i+1) = \varepsilon(i) - \mu_w \nabla \mathscr{L}_{MV\,\varepsilon(i)}$, enforcing the constraint and solving the resulting equations, we obtain

$$\begin{aligned}
\boldsymbol{S}_D(i+1) = \boldsymbol{S}_D(i) - \mu_s\big[&\bar{x}^*(i)\boldsymbol{r}(i)\bar{\boldsymbol{w}}^H(i) + \varepsilon(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i)\\
&- \big(\boldsymbol{a}_p^H(\theta_k)\boldsymbol{a}_p(\theta_k)\big)^{-1}\boldsymbol{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i)\big(\bar{x}^*(i)\boldsymbol{a}_p^H(\theta_k)\boldsymbol{r}(i) + \varepsilon(i)\big)\big],
\end{aligned} \tag{2.21}$$

$$\begin{aligned}
\bar{\boldsymbol{w}}(i+1) = \bar{\boldsymbol{w}}(i) - \mu_w\big(&\bar{x}^*(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i) + \varepsilon(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\\
&+ \big(\boldsymbol{a}_p^H(\theta_k)\boldsymbol{a}_p(\theta_k)\big)^{-1}\big(\bar{x}^*(i)\boldsymbol{r}^H(i)\boldsymbol{S}_D(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)\\
&+ \varepsilon(i)\boldsymbol{w}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)\big)\big),
\end{aligned} \tag{2.22}$$

$$\varepsilon(i+1) = \varepsilon(i) - \mu_\varepsilon \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i), \tag{2.23}$$

where $\bar{x}(i) = \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)$. The RJIO scheme trades-off a full-rank beamformer against one rank-reduction matrix $\boldsymbol{S}_D(i)$, one reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$, and one adaptive loading recursion operating in an alternating fashion and exchanging information.

2.5.2 Recursive Least-Squares Algorithms

Here, an RLS algorithm is devised for an efficient implementation of the RJIO method. To this end, let us first consider the Lagrangian

$$\begin{aligned}
\mathscr{L}_{\mathrm{LS}}\big(\boldsymbol{S}_D(i), \bar{\boldsymbol{w}}(i), \varepsilon(i)\big) = &\sum_{l=1}^{i}\alpha^{i-l}\big|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(l)\big|^2\\
&+ \varepsilon^2(i)\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\\
&+ \lambda\big(\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k) - 1\big),
\end{aligned} \tag{2.24}$$

where α is the forgetting factor chosen as a positive constant close to, but less than 1.

Fixing $\bar{\boldsymbol{w}}(i)$, computing the gradient of (2.24) with respect to $\boldsymbol{S}_D(i)$, equating the gradient terms to zero, and solving for λ, we obtain

$$\boldsymbol{S}_D(i) = \frac{\boldsymbol{P}(i)\boldsymbol{a}_p(\theta_k)\boldsymbol{a}_p^H(\theta_k)\boldsymbol{S}_D(i-1)}{\boldsymbol{a}_p^H(\theta_k)\boldsymbol{P}(i)\boldsymbol{a}_p(\theta_k)}, \tag{2.25}$$

where we defined the inverse covariance matrix $\boldsymbol{P}(i) = \boldsymbol{R}^{-1}(i)$ for convenience of presentation. Employing the matrix inversion lemma [3], we obtain

$$\boldsymbol{k}(i) = \frac{\alpha^{-1}\boldsymbol{P}(i-1)\boldsymbol{r}(i)}{1+\alpha^{-1}\boldsymbol{r}^H(i)\boldsymbol{P}(i-1)\boldsymbol{r}(i)}, \tag{2.26}$$

$$\boldsymbol{P}(i) = \alpha^{-1}\boldsymbol{P}(i-1) - \alpha^{-1}\boldsymbol{k}(i)\boldsymbol{r}^H(i)\boldsymbol{P}(i-1) + \varepsilon^2(i)\boldsymbol{I}_M, \tag{2.27}$$

where $\boldsymbol{k}(i)$ is the $M \times 1$ Kalman gain vector. We set $\boldsymbol{P}(0) = \delta \boldsymbol{I}_M$ to start the recursion of (2.27), where δ is a positive constant.

Assuming $\boldsymbol{S}_D(i)$ is known and taking the gradient of (2.24) with respect to $\bar{\boldsymbol{w}}(i)$, equating the terms to a null vector, and solving for λ, we obtain the $D \times 1$ reduced-rank beamformer

$$\bar{\boldsymbol{w}}(i) = \frac{\bar{\boldsymbol{P}}(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H(\theta_k)\boldsymbol{S}_D(i)\bar{\boldsymbol{P}}(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)}, \tag{2.28}$$

where $\bar{\boldsymbol{P}}(i) = \bar{\boldsymbol{R}}^{-1}(i)$ and $\bar{\boldsymbol{R}}(i) = \sum_{l=1}^{i} \alpha^{i-l}\bar{\boldsymbol{r}}(l)\bar{\boldsymbol{r}}^H(l)$ is the reduced-rank input covariance matrix. In order to estimate $\bar{\boldsymbol{P}}(i)$, we use the matrix inversion lemma [3] as follows:

$$\bar{\boldsymbol{k}}(i) = \frac{\alpha^{-1}\bar{\boldsymbol{P}}(i-1)\bar{\boldsymbol{r}}(i)}{1+\alpha^{-1}\bar{\boldsymbol{r}}^H(i)\bar{\boldsymbol{P}}(i-1)\bar{\boldsymbol{r}}(i)}, \tag{2.29}$$

$$\bar{\boldsymbol{P}}(i) = \alpha^{-1}\bar{\boldsymbol{P}}(i-1) - \alpha^{-1}\bar{\boldsymbol{k}}(i)\bar{\boldsymbol{r}}^H(i)\bar{\boldsymbol{P}}(i-1) + \varepsilon^2(i)\boldsymbol{I}_D, \tag{2.30}$$

where $\bar{\boldsymbol{k}}(i)$ is the $D \times 1$ reduced-rank gain vector and $\bar{\boldsymbol{P}}(i) = \bar{\boldsymbol{R}}^{-1}(i)$ is referred to as the reduced-rank inverse covariance matrix. Hence, the covariance matrix inversion $\bar{\boldsymbol{R}}^{-1}(i)$ is replaced at each step by the recursive processes (2.29) and (2.30) for reducing the complexity. The recursion of (2.30) is initialized by choosing $\bar{\boldsymbol{P}}(0) = \bar{\delta}\boldsymbol{I}_D$, where $\bar{\delta}$ is a positive constant. The last recursion adjusts the diagonal loading according to the following update equation

$$\varepsilon(i+1) = \varepsilon(i) - \mu_\varepsilon \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i). \tag{2.31}$$

The RJIO-RLS algorithm trades-off a full-rank beamformer with M coefficients against one matrix recursion to compute $\boldsymbol{S}_D(i)$, given in (2.25)–(2.27), one $D \times 1$ reduced-rank adaptive beamformer $\bar{\boldsymbol{w}}(i)$, given in (2.28)–(2.30), and one recursion to adjust the diagonal loading described in (2.31) in an alternating manner and exchanging information.

Table 2.1 Computational complexity of LCMV algorithms

Algorithm	Additions	Multiplications
LCMV-SG [4]	$3M+1$	$3M+2$
LCMV-RLS [6]	$3M^2-2M+3$	$6M^2+2M+2$
RJIO-SG	$3DM+4M+2D-2$	$5DM+2M+5D+2$
RJIO-RLS	$3M^2-M+3+3D^2-7D+3$	$7M^2+3M+7D^2+10D$
SMI [23]	$2/3M^3+3M^2$	$2/3M^3+5M^2$

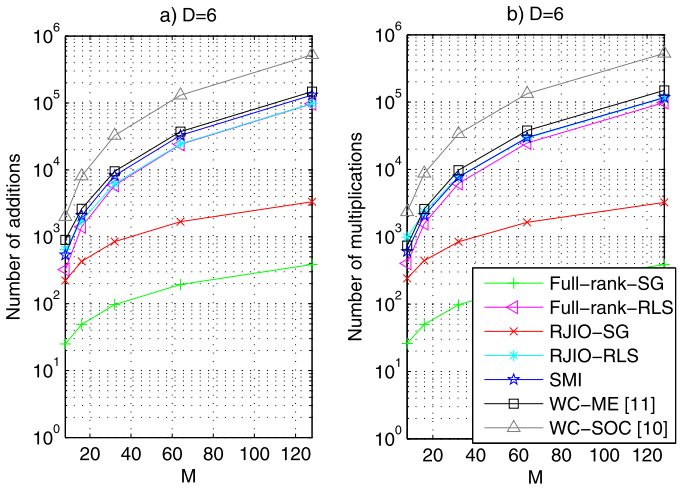


Fig. 2.3 Computational complexity in terms of arithmetic operations against M

2.5.3 Complexity of RJIO Algorithms

Here, we evaluate the computational complexity of the RJIO and analyzed LCMV algorithms. The complexity expressed in terms of additions and multiplications is depicted in Table 2.1. We can verify that the RJIO-SG algorithm has a complexity that grows linearly with DM, which is about D times higher than the full-rank LCMV-SG algorithm and significantly lower than the remaining techniques. If $D \ll M$ (as we will see later) then the additional complexity can be acceptable provided the gains in performance justify them. In the case of the RJIO-RLS algorithm, the complexity is quadratic with M^2 and D^2. This corresponds to a complexity slightly higher than the one observed for the full-rank LCMV-RLS algorithm, provided D is significantly less than M, and lower than the robust beamforming algorithms WC-SOC [9] and WC-ME [10].

In order to illustrate the main trends in what concerns the complexity of the proposed and analyzed algorithms, we show in Fig. 2.3 the complexity in terms of additions and multiplications versus the number of input samples M. The curves

indicate that the RJIO-RLS algorithm has a complexity lower than the WC-ME [10] and the WC-SOC [9], whereas it remains at the same level of the full-rank LCMV-RLS algorithm. The RJIO-SG algorithm has a complexity that is situated between the full-rank LCMV-RLS and the full-rank LCMV-SG algorithms.

2.5.4 Rank Adaptation

The performance of the algorithms described in the previous subsections depends on the rank D. This motivates the development of methods to automatically adjust D on the basis of the cost function. Different from existing methods for rank adaptation which use MSWF-based algorithms [23] or AVF-based recursions [25], we focus on an approach that jointly determines D based on the LS criterion computed by the filters $\boldsymbol{S}_D(i)$ and $\bar{\boldsymbol{w}}_D(i)$, where the subscript D denotes the rank used for the adaptation. In particular, we present a method for automatically selecting the ranks of the algorithms based on the exponentially weighted a posteriori least-squares type cost function described by

$$\mathscr{C}\big(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1)\big) = \sum_{l=1}^{i} \alpha^{i-l} \big|\bar{\boldsymbol{w}}_D^H(i-1)\boldsymbol{S}_D(i-1)\boldsymbol{r}(l)\big|^2, \tag{2.32}$$

where α is the forgetting factor and $\bar{\mathbf{w}}_D(i-1)$ is the reduced-rank beamformer with rank D. For each time interval i, we can select the rank D_{opt} which minimizes $\mathscr{C}(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1))$, and the exponential weighting factor α is required as the optimal rank varies as a function of the data record. The key quantities to be updated are the rank-reduction matrix $\boldsymbol{S}_D(i)$, the reduced-rank beamformer $\bar{\boldsymbol{w}}_D(i)$, the associated presumed reduced-rank steering vector $\bar{\boldsymbol{a}}_p(\theta_k)$, and the inverse of the reduced-rank covariance matrix $\bar{\boldsymbol{P}}(i)$ (for the RJIO-RLS algorithm). To this end, we define the following extended rank-reduction matrix $\boldsymbol{S}_D(i)$ and the extended reduced-rank beamformer weight vector $\bar{\boldsymbol{w}}_D(i)$ as follows:

$$\boldsymbol{S}_D(i) = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,D_{\min}} & \cdots & s_{1,D_{\max}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{M,1} & s_{M,2} & \cdots & s_{M,D_{\min}} & \cdots & s_{M,D_{\max}} \end{bmatrix} \quad \text{and}$$

$$\bar{\boldsymbol{w}}_D(i) = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{D_{\min}} \\ \vdots \\ w_{D_{\max}} \end{bmatrix}. \tag{2.33}$$

The extended rank-reduction matrix $\boldsymbol{S}_D(i)$ and the extended reduced-rank beamformer weight vector $\bar{\boldsymbol{w}}_D(i)$ are updated along with the associated quantities $\bar{\boldsymbol{a}}(\theta_k)$

and $\bar{\boldsymbol{P}}(i)$ (only for the RLS) for the maximum allowed rank $D_{\max}$ and then the rank adaptation algorithm determines the rank that is best for each time instant i using the cost function in (2.32). The rank adaptation algorithm is then given by

$$D_{\text{opt}} = \arg \min_{D_{\min} \leq d \leq D_{\max}} \mathscr{C}\big(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1)\big), \tag{2.34}$$

where d is an integer, $D_{\min}$ and $D_{\max}$ are the minimum and maximum ranks allowed for the reduced-rank beamformer, respectively. Note that a smaller rank may provide faster adaptation during the initial stages of the estimation procedure and a greater rank usually yields a better steady-state performance. Our studies reveal that the range for which the rank D of the proposed algorithms has a positive impact on the performance of the algorithms is limited, being from $D_{\min} = 3$ to $D_{\max} = 8$ for the reduced-rank beamformer recursions. These values are rather insensitive to the system load (number of users) and the number of array elements, and work very well for all scenarios and algorithms examined. The additional complexity of the proposed rank adaptation algorithm is that it requires the update of all involved quantities with the maximum allowed rank $D_{\max}$ and the computation of the cost function in (2.32). This procedure can significantly improve the convergence performance and can be relaxed (the rank can be made fixed) once the algorithm reaches steady state. Choosing an inadequate rank for adaptation may lead to performance degradation, which gradually increases as the adaptation rank deviates from the optimal rank.

2.6 Simulations

In this section, the performance of the RJIO and some existing beamforming algorithms is assessed using computer simulations. A sensor-array system with a ULA equipped with M sensor elements is considered for assessing the beamforming algorithms. In particular, the performance of the RJIO scheme with SG and RLS algorithms is compared with existing techniques, namely, the full-rank LCMV-SG [4] and LCMV-RLS [6], and the robust techniques reported in [9], termed WC-SOC, and [10], called Robust-ME, and the optimal linear beamformer that assumes the knowledge of the covariance matrix and the actual steering vector [2]. In particular, the algorithms are compared in terms of the signal-to-interference-plus-noise ratio (SINR), which is defined for the reduced-rank schemes as

$$\text{SINR}(i) = \frac{\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}_s\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)}{\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}_I\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)}, \tag{2.35}$$

where $\boldsymbol{R}_s$ is the covariance matrix of the desired signal and $\boldsymbol{R}_I$ is the covariance matrix of the interference and noise in the environment. Note that for the full-rank schemes the SINR(i) assumes $\boldsymbol{S}_D^H(i) = \boldsymbol{I}_M$. For each scenario, 200 runs are used to obtain the curves. In all simulations, the desired signal power is $\sigma_d^2 = 1$, and

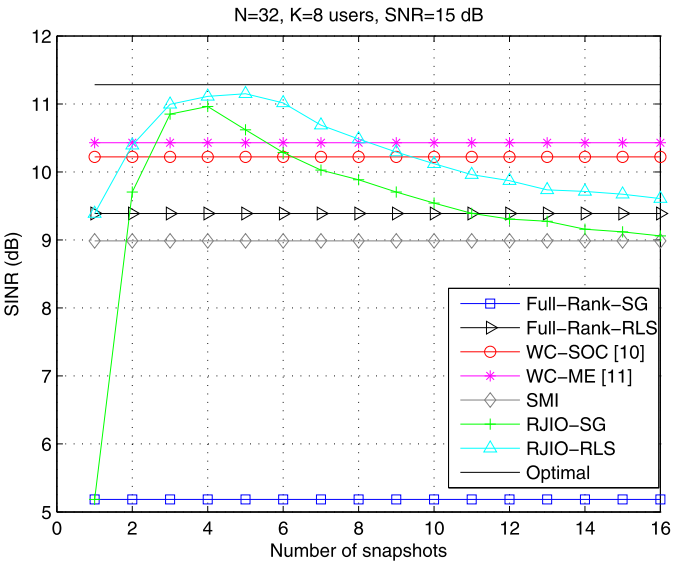


Fig. 2.4 SINR performance of LCMV algorithms against rank (D) with $M = 32$, SNR = 15 dB, $N = 250$ snapshots

the signal-to-noise ratio (SNR) is defined as SNR $= \frac{\sigma_d^2}{\sigma^2}$. The beamformers are initialized as $\bar{\boldsymbol{w}}(0) = [1\ 0\ \ldots\ 0]$ and $\boldsymbol{S}_D(0) = [\boldsymbol{I}_D^T\ \boldsymbol{0}_{D\times(M-D)}^T]$, where $\boldsymbol{0}_{D\times M-D}$ is a $D \times (M - D)$ matrix with zeros in all experiments.

In order to assess the performance of the RJIO and other existing algorithms in the presence of uncertainties, we consider that the array steering vector is corrupted by local coherent scattering

$$\boldsymbol{a}_p(\theta_k) = \boldsymbol{a}(\theta_k) + \sum_{k=1}^{4} \mathrm{e}^{j\Phi_k} \boldsymbol{a}_{\mathrm{sc}}(\theta_k), \tag{2.36}$$

where Φ_k is uniformly distributed between zero and 2π and θ_k is uniformly distributed with a standard deviation of 2 degrees with the assumed direction as the mean. The mismatch changes for every realization and is fixed over the snapshots of each simulation trial. In the first two experiments, we consider a scenario with 7 interferers at $-60°$, $-45°$, $30°$ $-15°$, $0°$, $45°$, $60°$ with powers following a log-normal distribution with associated standard deviation 3 dB around the SoI's power level. The SoI impinges on the array at $30°$. The parameters of the algorithms are optimized.

We first evaluate the SINR performance of the analyzed algorithms against the rank D using optimized parameters (μ_s, μ_w, and forgetting factors λ) for all schemes and $N = 250$ snapshots. The results in Fig. 2.4 indicate that the best rank for the RJIO scheme is $D = 4$ (which will be used in the second scenario) and it is very close to the optimal full-rank LCMV beamformer that has knowledge about the

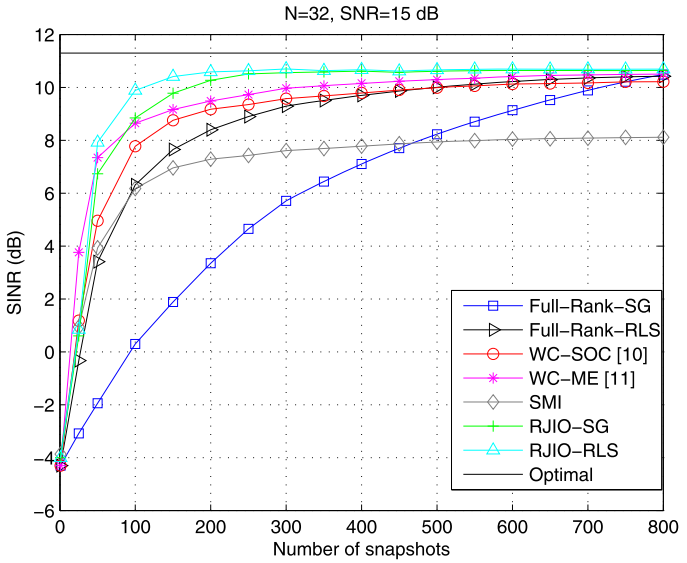


Fig. 2.5 SINR performance of robust LCMV algorithms against snapshots with $M = 32$, SNR $= 15$ dB

actual steering vector. An examination of systems with different sizes has shown that D is relatively invariant to the system size, which brings considerable computational savings. In practice, the rank D can be adapted in order to obtain fast convergence and ensure good steady-state performance and tracking after convergence.

We display another scenario in Fig. 2.5 where the robust adaptive LCMV beamformers are set to converge to the same level of SINR. The parameters used to obtain these curves are also shown. The curves show an excellent performance for the RJIO scheme which converges much faster than the full-rank-SG algorithm, and is also better than the more complex WC-SOC [9] and Robust-ME [10] schemes.

In the next example, we consider the design of the RJIO-SG and RJIO-RLS algorithms equipped with the rank adaptation method described in Sect. 2.5.4. We consider 5 interferers at $-60°$, $-30°$, $0°$, $45°$, $60°$ with equal powers to the SoI, which impinges on the array at $15°$. Specifically, we evaluate the rank adaptation algorithms against the use of fixed ranks, namely, $D = 3$ and $D = 8$ for both SG and RLS algorithms. The results show that the rank adaptation method is capable of ensuring an excellent trade-off between convergence speed and steady-state performance, as illustrated in Fig. 2.6. In particular, the algorithm can achieve a significantly faster convergence performance than the scheme with fixed rank $D = 8$, whereas it attains the same steady state performance.

In the last experiment, we consider a nonstationary scenario where the system has 6 users with equal power and the environment experiences a sudden change at time $i = 800$. The 5 interferers impinge on the ULA at $-60°$, $-30°$, $0°$, $45°$, $60°$ with equal powers to the SoI, which impinges on the array at $15°$. At time instant $i = 800$ we have 3 interferers with 5 dB above the SoI's power level entering

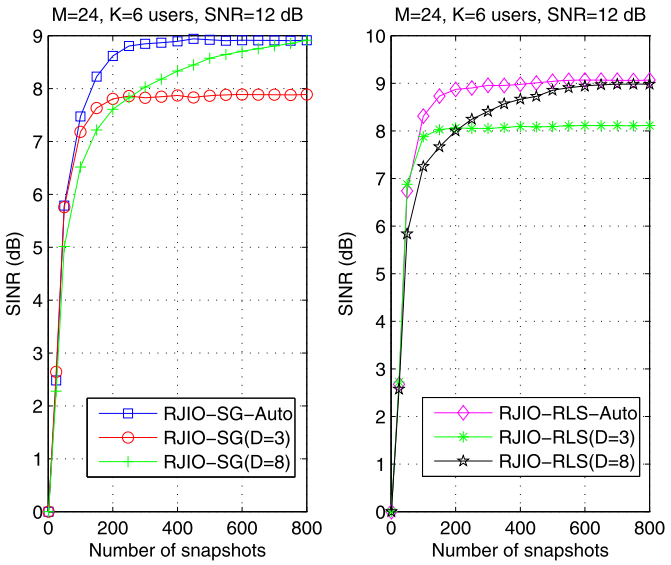


Fig. 2.6 SINR performance of RJIO-LCMV (**a**) SG and (**b**) RLS algorithms against snapshots with $M = 24$, SNR $= 12$ dB with rank adaptation

the system with DoAs $-45°$, $-15°$ and $30°$, whereas one interferer with DoA $45°$ and a power level equal to the SoI exits the system. The RJIO and other analyzed adaptive beamforming algorithms are equipped with rank adaptation techniques and have to adjust their parameters in order to suppress the interferers. We optimize the step sizes and the forgetting factors of all the algorithms in order to ensure that they converge as fast as they can to the same value of SINR. The results of this experiment are depicted in Fig. 2.7. The curves show that the RJIO algorithms have a superior performance to the existing algorithms considered in this study.

2.7 Conclusions

We have investigated robust reduced-rank LCMV beamforming algorithms based on robust joint iterative optimization of beamformers. The RJIO reduced-rank scheme is based on a robust constrained joint iterative optimization of beamformers according to the minimum variance criterion. We derived robust LCMV expressions for the design of the rank-reduction matrix and the reduced-rank beamformer and developed SG and RLS adaptive algorithms for their efficient implementation along with a rank adaptation technique. The numerical results for an adaptive beamforming application with a ULA have shown that the RJIO scheme and algorithms outperform in convergence, steady state and tracking the existing robust full-rank and reduced-rank algorithms at comparable complexity. The proposed algorithms can be extended to other array geometries and applications.

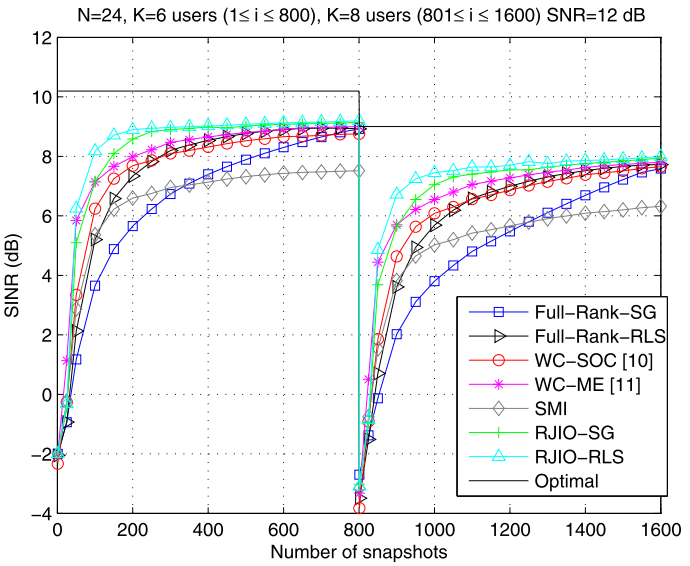


Fig. 2.7 SINR performance of robust LCMV algorithms against the number of snapshots with $M = 24$, SNR $= 12$ dB in a nonstationary scenario

References

1. Van Trees, H.L.: Detection, Estimation, and Modulation Theory, Part IV, Optimum Array Processing. Wiley, New York (2002)
2. Li, J., Stoica, P.: Robust Adaptive Beamforming. Wiley, New York (2006)
3. Haykin, S.: Adaptive Filter Theory, 4th edn. Prentice Hall, New York (2002)
4. Frost, O.L. III: An algorithm for linearly constrained adaptive array processing. Proc. IEEE **AP-30**, 27–34 (1972)
5. de Lamare, R.C., Sampaio-Neto, R.: Low-complexity variable step-size mechanisms for stochastic gradient algorithms in minimum variance CDMA receivers. IEEE Trans. Signal Process. **54**, 2302–2317 (2006)
6. Resende, L.S., Romano, J.M.T., Bellanger, M.G.: A fast least-squares algorithm for linearly constrained adaptive filtering. IEEE Trans. Signal Process. **44**, 1168–1174 (1996)
7. Cox, H., Zeskind, R.M., Owen, M.H.: Robust adaptive beamforming. IEEE Trans. Acoust. Speech Signal Process. **ASSP-35**, 1365–1376 (1987)
8. Feldman, D.D., Griffiths, L.J.: A projection approach for robust adaptive beamforming. IEEE Trans. Signal Process. **42**, 867–876 (1994)
9. Vorobyov, S.A., Gershman, A.B., Luo, Z.-Q.: Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem. IEEE Trans. Signal Process. **51**, 313–324 (2003)
10. Li, J., Stoica, P.: On robust capon beamforming and diagonal loading. IEEE Trans. Signal Process. **51**(7), 1702–1715 (2003)
11. Stoica, P., Li, J., Wang, Z.: Doubly constrained robust capon beamformer. IEEE Trans. Signal Process. **52**(9), 2407–2423 (2004)
12. Lorenz, R., Boyd, S.: Robust minimum variance beamforming. IEEE Trans. Signal Process. **53**, 1684–1696 (2005)
13. Chen, C.-Y., Vaidyanathan, P.P.: Quadratically constrained beamforming robust against direction-of-arrival mismatch. IEEE Trans. Signal Process. **55**(8) (2007)

14. Hassanien, A., Vorobyov, S.A.: Robust adaptive beamforming using sequential quadratic programming: an iterative solution to the mismatch problem. IEEE Signal Process. Lett. **15**, 733–736 (2008)
15. Wang, L., de Lamare, R.C.: Low-complexity adaptive step size constrained constant modulus SG algorithms for blind adaptive beamforming. Signal Process. **89**(12), 2503–2513 (2009)
16. Wang, L., de Lamare, R.C., Yukawa, M.: Adaptive reduced-rank constrained constant modulus algorithms based on joint iterative optimization of filters for beamforming. IEEE Trans. Signal Process. **58**(6), 2983–2997 (2010)
17. de Lamare, R.C., Sampaio-Neto, R., Haardt, M.: Blind adaptive constrained constant-modulus reduced-rank interference suppression algorithms based on interpolation and switched decimation. IEEE Trans. Signal Process. **59**(2), 681–695 (2011)
18. Scharf, L.L., Tufts, D.W.: Rank reduction for modeling stationary signals. IEEE Trans. Acoust. Speech Signal Process. **ASSP-35**, 350–355 (1987)
19. Haimovich, A.M., Bar-Ness, Y.: An eigenanalysis interference canceler. IEEE Trans. Signal Process. **39**, 76–84 (1991)
20. Pados, D.A., Batalama, S.N.: Joint space-time auxiliary vector filtering for DS/CDMA systems with antenna arrays. IEEE Trans. Commun. **47**(9), 1406–1415 (1999)
21. Goldstein, J.S., Reed, I.S., Scharf, L.L.: A multistage representation of the Wiener filter based on orthogonal projections. IEEE Trans. Inf. Theory **44**(7) (1998)
22. Hua, Y., Nikpour, M., Stoica, P.: Optimal reduced rank estimation and filtering. IEEE Trans. Signal Process. **49**(3), 457–469 (2001)
23. Honig, M.L., Goldstein, J.S.: Adaptive reduced-rank interference suppression based on the multistage Wiener filter. IEEE Trans. Commun. **50**(6) (2002)
24. Santos, E.L., Zoltowski, M.D.: On low rank MVDR beamforming using the conjugate gradient algorithm. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (2004)
25. Haoli, Q., Batalama, S.N.: Data record-based criteria for the selection of an auxiliary vector estimator of the MMSE/MVDR filter. IEEE Trans. Commun. **51**(10), 1700–1708 (2003)
26. de Lamare, R.C., Sampaio-Neto, R.: Reduced-rank adaptive filtering based on joint iterative optimization of adaptive filters. IEEE Signal Process. Lett. **14**(12) (2007)
27. de Lamare, R.C.: Adaptive reduced-rank LCMV beamforming algorithms based on joint iterative optimisation of filters. Electron. Lett. **44**(9) (2008)
28. de Lamare, R.C., Sampaio-Neto, R.: Adaptive reduced-rank processing based on joint and iterative interpolation, decimation and filtering. IEEE Trans. Signal Process. **57**(7), 2503–2514 (2009)
29. de Lamare, R.C., Wang, L., Fa, R.: Adaptive reduced-rank LCMV beamforming algorithms based on joint iterative optimization of filters: design and analysis. Signal Process. **90**(2), 640–652 (2010)
30. Fa, R., de Lamare, R.C., Wang, L.: Reduced-rank STAP schemes for airborne radar based on switched joint interpolation, decimation and filtering algorithm. IEEE Trans. Signal Process. **58**(8), 4182–4194 (2010)
31. Van Veen, B.D.: Adaptive convergence of linearly constrained beamformers based on the sample covariance matrix. IEEE Trans. Signal Process. **39**, 1470–1473 (1991)
32. Bertsekas, D.P.: Nonlinear Programming, 2nd edn. Athena Scientific, Belmont (1999)

Chapter 3
Designing OFDM Radar Waveform for Target Detection Using Multi-objective Optimization

Satyabrata Sen, Gongguo Tang, and Arye Nehorai

Abstract We propose a multi-objective optimization (MOO) technique to design an orthogonal frequency division multiplexing (OFDM) radar signal for detecting a moving target in the presence of multipath reflections. We employ an OFDM signal to increase the frequency diversity of the system, as different scattering centers of a target resonate variably at different frequencies. Moreover, the multipath propagation increases the spatial diversity by providing extra "looks" at the target. First, we develop a parametric OFDM measurement model for a particular range cell under test, and convert it to an equivalent sparse-model by considering the target returns over all the possible signal paths and target velocities. Then, we propose a constrained MOO problem to design the spectral-parameters of the transmitting OFDM waveform by simultaneously optimizing three objective functions: maximizing the Mahalanobis distance to improve the detection performance, minimizing the weighted trace of the Cramér–Rao bound matrix for the unknown parameters to increase the estimation accuracy, and minimizing the upper bound on the sparse-recovery error to improve the performance of the equivalent sparse-estimation approach.

S. Sen (✉)
Computer Science and Mathematics Division, Oak Ridge National Laboratory, 1 Bethel Valley Road, Oak Ridge, TN 37931, USA
e-mail: sens@ornl.gov

G. Tang
Department of Electrical and Computer Engineering, University of Wisconsin-Madison, 1415 Engineering Drive, Madison, WI 53706, USA
e-mail: gtang5@wisc.edu

A. Nehorai
Preston M. Green Department of Electrical & Systems Engineering, Washington University in St. Louis, 1 Brookings Drive, Saint Louis, MO 63130, USA
e-mail: nehorai@ese.wustl.edu

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_3,

3.1 Introduction

The problem of adaptive waveform design is becoming increasingly relevant and challenging to modern state-of-the-art radar systems. For many years, conventional radars have transmitted a fixed waveform on every pulse, and the research efforts have been primarily devoted to optimally process the corresponding received signals [50]. However, with the recent technological advancements in the fields of flexible waveform generators and high-speed signal processing hardware, it is now possible to generate and transmit sophisticated radar waveforms that are dynamically adapted to the sensing environments on a periodic basis (potentially on a pulse-by-pulse basis) [7, 23, 35, 55, 56]. Such adaptation can lead to a significant performance gain over the classical (non-adaptive) radar waveforms, particularly in the defense applications involving fast-changing scenarios.

A comprehensive survey on different waveform-design techniques can be found in [38, Chap. 1.2] and references therein; here we briefly discuss some of the salient research work on this topic. Earlier attempts of radar waveform design were to compute parameters of the radar waveform (amplitude, phase, etc.) and the associated receiver response in order to improve the target-detection performance in the presence of clutter and interference [19, 42, 43, 51, 53]. The target-matched illumination techniques are proposed in [22, 25, 41] to optimally design the combination of transmit-waveform and receive-filter for the identification and characterization of targets with known responses. In [5, 29, 44], the information-theoretic optimization criteria, based on the mutual information between the transmit/receive waveform and target response, are considered to design the optimal waveforms for detection, estimation, and tracking problems. Several waveform-optimization algorithms based on the properties of the Cramér–Rao bound (CRB) matrix are presented in [24, 31, 49] for the purpose of target-tracking and parameter estimation. Recently, various constrained waveform-design methodologies are studied to obtain more practical radar waveforms, such as constraining the optimal waveform to have a constant modulus [39], to have a bounded peak-to-average power ratio [14], and to be *similar* to another waveform that has the desired autocorrelation function or ambiguity profile [12, 13, 30, 40].

The waveform-design problems become further intriguing when one needs to simultaneously satisfy two or more optimality criteria, particularly in a multi-mission radar system [4, 21]. Often, the desirable optimization functions are very different and even conflicting to each other, which give rise to dissimilar parameter values for the optimal waveform. To tackle this quandary, the multi-objective optimization (MOO) procedures are employed that concurrently optimize the various objective functions in a Pareto-sense [1, 2, 15, 16, 48]. This type of optimality was originally introduced by Francis Ysidro Edgeworth in 1881 [20] and later generalized by Vilfredo Pareto in 1896 [37].

In this work (see also [47, 48]), we consider a Pareto-optimal waveform-design approach for an orthogonal frequency division multiplexing (OFDM) radar signal to detect a moving target in the presence of multipath reflections, which exist, for example, in urban environments as shown in Fig. 3.1. In [45, 46], we showed that

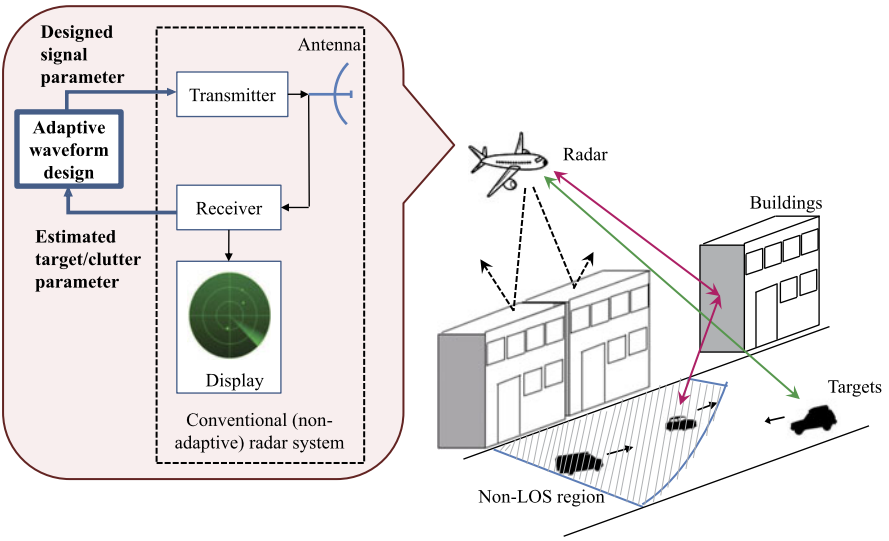


Fig. 3.1 Principle of adaptive waveform design in a radar operating in the multipath scenarios

the target-detection capability can be significantly improved by exploiting multiple Doppler shifts corresponding to the projections of the target velocity on each of the multipath components. Furthermore, the multipath propagations increase the spatial diversity of the radar system by providing extra "looks" at the target and thus enabling target detection and tracking even beyond the line-of-sight (LOS) [28]. To resolve and exploit the multipath components it is common to use short pulse, multi-carrier wideband radar signals. We consider OFDM signaling scheme [34, 36], which is one way to use several subcarriers simultaneously. The use of an OFDM signal mitigates possible fading, resolves multipath reflections, and provides additional frequency diversity as different scattering centers of a target resonate at different frequencies [27, 54].

First, we develop a parametric OFDM measurement model for a particular range cell, to detect a far-field target moving in a multipath-rich environment. We assume that the radar has the complete knowledge of the first-order (or single bounce) specularly reflected multipath signals. Using such knowledge of the geometry, we can determine all the possible paths, be they LOS or reflected, from the range cell under test. However, in practice the target responses reach the radar only via a limited number of paths depending on the position of the target within the range cell. Therefore, considering all the possible signal paths and target velocities, which represent themselves as varying Doppler shifts at the radar receiver, we convert the OFDM measurement model to an equivalent sparse model. The nonzero components of the sparse vector in this model correspond to the scattering coefficients of the target at the true signal paths and target velocity.

The formulation of a sparse-measurement model transforms a target-detection problem into a task of estimating the nonzero coefficients of a sparse signal. To

estimate the sparse vector, we propose a sparse-recovery algorithm based on the Dantzig selector (DS) approach [8]. The DS approach belongs to the class of convex relaxation methods in which the ℓ_0 norm is replaced by the ℓ_1 norm that remains a measure of sparsity while being a convex function. However, instead of using the standard DS, in this work we employ a collection of multiple small DS that exploits more prior structures of the sparse vector and provides improved performances both in terms of computational-time and estimation-accuracy [48].

Next, we propose to optimally design the spectral-parameters of the transmitting OFDM waveform for the next coherent processing interval in order to improve the system performance. We formulate and solve a constrained MOO problem [11, 17, 33, 57] that simultaneously optimizes three objective functions. The maximization of the Mahalanobis distance [3, 32] is considered as the first of the three objective functions. This is because the Mahalanobis distance provides a standard measure to quantify the distance between two distributions associated with two hypotheses of a detection problem. However, to compute the Mahalanobis distance in practice, we need to estimate the target scattering-coefficients, velocity, and noise covariance matrix. We characterize the accuracy of such estimations by calculating the CRBs for the unknown parameters, because CRB is a universal lower bound on the variance of all unbiased estimators of a set of parameters [26, Chap. 3]. Therefore, our second objective function tries to minimize a weighted trace of the resultant CRB matrix. Additionally, if we solve the equivalent sparse-estimation problem, then we analyze the reconstruction-performance by evaluating an upper bound on the sparse-estimation error in terms of the ℓ_1-constrained minimal singular value (ℓ_1-CMSV) of the sparse-measurement matrix [52]. Compared with the traditional restricted isometry constant (RIC) [9, 10], which is extremely difficult to compute for an arbitrarily given matrix, the ℓ_1-CMSV is an easily computable measure and provides more intuition on the stability of sparse-signal recovery. Hence, as the third objective function, we propose to minimize the upper bound on the sparse-estimation error for improving the performance of sparse-recovery.

To solve the MOO problem, we apply the well-known nondominated sorting genetic algorithm II (NSGA-II) [18], which belongs to the class of evolutionary algorithms (EAs) and provides a set of solutions known as Pareto-optimal solutions [17]. All the solutions residing on the Pareto-front are considered to be superior to any other solution in the search space when all objectives are considered. The idea of finding as many Pareto-optimal solutions as possible motivates the use of EAs that generate several solutions in a single run. Alternatively, we avoid using the scalarization technique, that transforms a MOO problem into a single-objective one by pre-multiplying each objective with a scalar weight (as done in [15, 16]), primarily for two reasons [17, Chaps. 2, 3]: (i) the optimal solution of a scalarization technique can be very subjective in nature, as it is heavily sensitive to the pre-defined scalar weights used in forming the single-objective function; and (ii) all the Pareto-optimal solutions may not be found by a scalarization approach for the nonlinear and nonconvex optimization problems.

We demonstrate the performance improvement due to the adaptive OFDM-waveform design with several numerical examples. We observe that the Pareto-optimal solutions provide a set of compromised solutions that varies in between

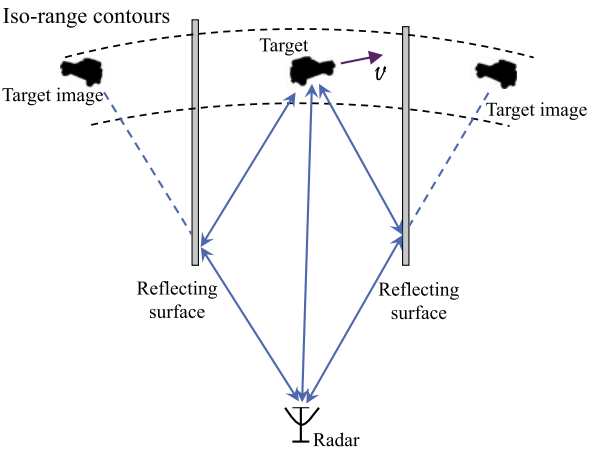


Fig. 3.2 A schematic representation of the multipath scenario

two extrema that are approximately equal to the individual solutions of the objective functions when solved separately. We found that all the Pareto-optimal solutions produce better performances than the fixed waveform, in terms of the Mahalanobis distance and weighted trace of CRB matrix; whereas only a smaller set of the Pareto-optimal solutions has improved performance with respect to the upper bound on sparse-error. Assuming that the noise powers over different subcarriers are the same, we infer that the solution of the Pareto-optimal design redistributes the energy of the transmitted signal by putting the most energy to that particular subcarrier along which the signal-to-noise ratio is the strongest.

The rest of the chapter is organized as follows. In Sect. 3.2, we first develop a parametric OFDM measurement model by incorporating the effects of first-order multipath reflections. Then, in Sect. 3.3, we convert the detection problem to one of sparse-estimation and present a decomposed DS-based sparse-recovery algorithm. In Sect. 3.4, we propose the adaptive OFDM-waveform design algorithm based on the MOO approach. Numerical examples and conclusions are presented in Sects. 3.5 and 3.6, respectively.

3.2 Problem Description and Modeling

Figure 3.2 presents a schematic representation of the problem scenario. We consider a far-field target in a multipath-rich environment, moving with a constant relative velocity **v** with respect to the radar. At the operating frequency, we assume that the reflecting surfaces produce only specular reflections of the radar signal, and for simplicity we consider only the first-order reflections. We further assume that the radar has complete knowledge of the environment under surveillance because nowadays accurate information about the city and building layouts can be obtained from lidar imaging systems, blueprints at the city hall, and even public tools such as Google

Maps. Hence, for a particular range cell (the region within the two successive iso-range contours of Fig. 3.2) the radar knows the specific paths, through which the target information reaches the radar receiver, in terms of its direction-of-arrival (DOA) unit-vectors ($\mathbf{u}_p$, $p = 1, 2, \ldots, P$). Under this scenario, our goal here is to decide whether a target is present or not in the range cell under test.

In the following, we first develop a parametric OFDM-measurement model that includes the effects of multipath reflections in between the radar and target. Then, we discuss our statistical assumptions on the clutter and noise.

3.2.1 OFDM Signal Model

We consider a wideband OFDM signaling system with L active subcarriers, a bandwidth of B Hz, and pulse width of T_p seconds. Let $\mathbf{a} = [a_0, a_1, \ldots, a_{L-1}]^T$ represent the complex weights transmitted over the L subcarriers, and satisfy $\sum_{l=0}^{L-1} |a_l|^2 = 1$. Then, the complex envelope of the transmitted signal in a single pulse can be represented as

$$s(t) = \sum_{l=0}^{L-1} a_l e^{j2\pi l \Delta f t}, \quad \text{for } 0 \le t \le T_\mathrm{p}, \tag{3.1}$$

where the subcarrier spacing is denoted by $\Delta f = B/(L+1) = 1/T_\mathrm{p}$. Let f_c be the carrier frequency of operation; then a coherent burst of N transmitted pulses is given by

$$\widetilde{s}(t) = 2\,\mathrm{Re}\left\{ \sum_{n=0}^{N-1} s(t - nT) e^{j2\pi f_\mathrm{c} t} \right\}, \tag{3.2}$$

where T is the pulse repetition interval (PRI). We point out here that in Sect. 3.4 while designing the adaptive waveform we choose the spectral-parameters of the OFDM waveform, a_ls, in order to improve the system performance.

3.2.2 Measurement Model

For a single-pulse, single-carrier transmitted signal $\widetilde{s}_l(t) = 2\,\mathrm{Re}\{a_l e^{j2\pi f_l t}\}$, where $f_l = f_\mathrm{c} + l\Delta f$ is the lth subcarrier frequency, the received signal along the pth path (represented by the DOA unit-vector $\mathbf{u}_p$) and at the same carrier-frequency f_l can be written as

$$\widetilde{y}_{lp}(t) = x_{lp} \widetilde{s}_l\big(\gamma_p (t - \tau_p)\big) + \widetilde{e}_{lp}(t), \tag{3.3}$$

where x_{lp} is a complex quantity representing the scattering coefficient of the target along the lth subchannel and pth path; $\gamma_p = 1 + \beta_p$ where $\beta_p = 2\langle \mathbf{v}, \mathbf{u}_p \rangle / c$ is the

relative Doppler shift along the pth path and c is the speed of propagation; τ_p is the roundtrip delay between the radar and target along the pth path; $\widetilde{e}_{lp}$ represents the clutter and measurement noise along the lth subcarrier and pth path. Therefore, the received signal over all P paths due to an L-carrier OFDM signal is given by

$$\begin{aligned}\widetilde{y}(t) &= \sum_{l=0}^{L-1}\sum_{p=1}^{P}\widetilde{y}_{lp}(t) = 2\,\mathrm{Re}\left\{\sum_{l=0}^{L-1}\sum_{p=1}^{P}a_l x_{lp} e^{j2\pi f_l \gamma_p (t-\tau_p)}\right\} + \widetilde{e}(t),\\ &= 2\,\mathrm{Re}\left\{\sum_{l=0}^{L-1}\sum_{p=1}^{P}a_l x_{lp} e^{-j2\pi f_l \gamma_p \tau_p} e^{j2\pi f_l \beta_p t} e^{j2\pi f_l t}\right\} + \widetilde{e}(t), \end{aligned} \tag{3.4}$$

and hence the corresponding complex envelope at the output of the lth subchannel is

$$y_l(t) = \sum_{p=1}^{P} a_l x_{lp} e^{-j2\pi f_l \gamma_p \tau_p} e^{j2\pi f_l \beta_p t} + e_l(t). \tag{3.5}$$

Next, we assume that the relative time gaps between any two multipath signals are very small in comparison to the actual roundtrip delays, i.e., $\tau_p \approx \tau_0$ for $p = 1, 2, \ldots, P$. This assumption holds true when the path lengths of multipath arrivals differ little (e.g., in a narrow urban canyon where the down-range is much greater than the width). Further, we consider that the temporal measurements from a specific range gate (denoted by the roundtrip delay τ_0) are collected at every $t = \tau_0 + nT$ instants. Therefore, corresponding to a specific range cell containing the target, the complex envelope of the received signal at the output of the lth subchannel is

$$y_l(n) = \sum_{p=1}^{P} a_l x_{lp} e^{-j2\pi f_l \gamma_p \tau_0} e^{j2\pi f_l \beta_p (\tau_0 + nT)} + e_l(n). \tag{3.6}$$

Defining

$$\phi_{lp}(n) \triangleq e^{-j2\pi f_l \tau_0} e^{j2\pi f_l \beta_p nT}, \tag{3.7}$$

we can rewrite (3.6) as

$$y_l(n) = a_l \boldsymbol{\phi}_l(n)^T \mathbf{x}_l + e_l(n), \tag{3.8}$$

where $\boldsymbol{\phi}_l(n) = [\phi_{l1}(n), \phi_{l2}(n), \ldots, \phi_{lP}(n)]^T$ and $\mathbf{x}_l = [x_{l1}, x_{l2}, \ldots, x_{lP}]^T$ are two $P \times 1$ vectors respectively containing the Doppler information and the scattering coefficients of the target at the lth subchannel over all P multipath.

Then, stacking the measurements of all L subchannels into one column vector of dimension $L \times 1$, we get

$$\mathbf{y}(n) = \mathbf{A}\boldsymbol{\Phi}(n)\mathbf{x} + \mathbf{e}(n), \tag{3.9}$$

where

- $\mathbf{y}(n) = [y_0(n), y_1(n), \ldots, y_{L-1}(n)]^T$;
- $\mathbf{A} = \text{diag}(\mathbf{a})$ is an $L \times L$ complex diagonal matrix that contains the transmitted weights $\mathbf{a}$;
- $\boldsymbol{\Phi}(n) = \text{blkdiag}(\boldsymbol{\phi}_0(n)^T, \boldsymbol{\phi}_1(n)^T, \ldots, \boldsymbol{\phi}_{L-1}(n)^T)$ is an $L \times LP$ complex rectangular block-diagonal matrix;
- $\mathbf{x} = [\mathbf{x}_0^T, \mathbf{x}_1^T, \ldots, \mathbf{x}_{L-1}^T]^T$ is an $LP \times 1$ complex vector;
- $\mathbf{e}(n) = [e_0(n), e_1(n), \ldots, e_{L-1}(n)]^T$ is an $L \times 1$ vector of clutter returns, measurement noise, and co-channel interference.

Finally, concatenating all the temporal data columnwise into an $LN \times 1$ vector, we obtain the OFDM-measurement model as follows:

$$\mathbf{y} = \boldsymbol{\Psi}\mathbf{x} + \mathbf{e}, \tag{3.10}$$

where

- $\mathbf{y} = [\mathbf{y}(0)^T, \mathbf{y}(1)^T, \ldots, \mathbf{y}(N-1)^T]^T$;
- $\boldsymbol{\Psi} = [(\mathbf{A}\boldsymbol{\Phi}(0))^T \ \ldots \ (\mathbf{A}\boldsymbol{\Phi}(N-1))^T]^T$ is an $LN \times LP$ matrix containing the Doppler information of the target;
- $\mathbf{e} = [\mathbf{e}(0)^T, \mathbf{e}(1)^T, \ldots, \mathbf{e}(N-1)^T]^T$ is an $LN \times 1$ vector comprising clutter returns, noise, and interference.

3.2.3 Statistical Assumptions

In our problem, the clutter could be the contribution of undesired reflections from the environment surrounding or behind the target, or random multipath reflections from the irregularities on the reflecting surface (e.g., windows and balconies of the buildings in an urban scenario), that cannot be modeled as specular components. In (3.9), the noise vector $\mathbf{e}(n)$ models the clutter returns, measurement noise, and co-channel interference at the output of L subchannels. We assume that $\mathbf{e}(n)$ is a temporally white and circularly symmetric zero-mean complex Gaussian vector, correlated between different subchannels with positive definite covariance matrix $\boldsymbol{\Sigma}$. This assumption implies that the OFDM measurements in (3.10) are distributed as

$$\mathbf{y} \sim \mathbb{C}\mathcal{N}_{LN}(\boldsymbol{\Psi}\mathbf{x}, \mathbf{I}_N \otimes \boldsymbol{\Sigma}). \tag{3.11}$$

3.3 Sparse-Estimation Approach

In this section, we reformulate the detection problem of the previous section to a sparsity-based estimation task. Using our knowledge of the geometry, we can determine all the possible signal paths in between the radar and the range cell under test, and subsequently can understand the possible extent of the Doppler variations at the radar receiver. In general, depending on the problem-scenario and target-velocity, a

set of such Doppler shifts could be very large. However, restricting our operation to a narrow region of interest (e.g., an urban canyon where the range is much greater than the width) and a few classes of targets that have comparable velocities (e.g., cars/trucks within a city environment), we can limit the extent of viable Doppler shifts to a smaller quantity.

In the following, we first convert the OFDM-measurement model of (3.10) to a sparse model that accounts for a set of finely discretized Doppler shifts. Then, we present an efficient sparse-recovery approach that employs a collection of multiple small DS in order to utilize more prior structures of the sparse vector.

3.3.1 Sparse Model

Suppose we discretize the extent of feasible Doppler shifts into N_β grid points as $\{\beta_i, i = 0, 1, \ldots, N_\beta - 1\}$. Then, we can remodel (3.8) as

$$y_l(n) = a_l \widetilde{\boldsymbol{\phi}}_l(n)^T \boldsymbol{\zeta}_l + e_l(n), \tag{3.12}$$

where

- $\widetilde{\boldsymbol{\phi}}_l(n) = [\phi_{l0}(n), \phi_{l1}(n), \ldots, \phi_{l(N_\beta-1)}(n)]^T$ represents an equivalent sparsity-based modeling of $\boldsymbol{\phi}_l(n)$;
- $\boldsymbol{\zeta}_l = [\zeta_{l0}, \zeta_{l1}, \ldots, \zeta_{l(N_\beta-1)}]^T$ is an $N_\beta \times 1$ sparse vector, having $P(\ll N_\beta)$ nonzero entries corresponding to the true target scattering coefficients, i.e.,

$$\zeta_{li} = \begin{cases} x_{lp} & \text{if } i = p, \\ 0 & \text{otherwise.} \end{cases} \tag{3.13}$$

Using the formulation of (3.12) and following the approach presented in Sect. 3.2.2 to obtain (3.10) from (3.8), we deduce a sparse-measurement model as

$$\mathbf{y} = \widetilde{\boldsymbol{\Psi}} \boldsymbol{\zeta} + \mathbf{e}, \tag{3.14}$$

where

- $\widetilde{\boldsymbol{\Psi}} = [(\mathbf{A}\widetilde{\boldsymbol{\Phi}}(0))^T \ \ldots \ (\mathbf{A}\widetilde{\boldsymbol{\Phi}}(N-1))^T]^T$ is an $LN \times LN_\beta$ sparse-measurement matrix containing all the viable Doppler information in terms of the $L \times LN_\beta$ dimensional matrices $\widetilde{\boldsymbol{\Phi}}(n) = \text{blkdiag}(\widetilde{\boldsymbol{\phi}}_0(n)^T, \widetilde{\boldsymbol{\phi}}_1(n)^T, \ldots, \widetilde{\boldsymbol{\phi}}_{L-1}(n)^T)$;
- $\boldsymbol{\zeta} = [\boldsymbol{\zeta}_0^T, \boldsymbol{\zeta}_1^T, \ldots, \boldsymbol{\zeta}_{L-1}^T]^T$ is an $LN_\beta \times 1$ sparse-vector that has LP nonzero entries representing the scattering coefficients of the target along all the P received paths and L subcarriers.

3.3.2 Sparse Recovery

The goal of a sparse-reconstruction algorithm is to estimate the vector $\boldsymbol{\zeta}$ from the noisy measurement $\mathbf{y}$ of (3.14) by exploiting the sparsity. One of the most popular

approaches of sparse-signal recovery is the Dantzig selector [8], which provides an estimate of $\boldsymbol{\zeta}$ as a solution to the following ℓ_1-regularization problem:

$$\min_{\mathbf{z}\in\mathbb{C}^{LN_\beta}} \quad \|\mathbf{z}\|_1 \quad \text{subject to} \quad \left\|\widetilde{\boldsymbol{\Psi}}^H(\mathbf{y}-\widetilde{\boldsymbol{\Psi}}\mathbf{z})\right\|_\infty \le \lambda\cdot\sigma, \tag{3.15}$$

where $\lambda=\sqrt{2\log(LN_\beta)}$ is a control parameter that ensures that the residual $(\mathbf{y}-\widetilde{\boldsymbol{\Psi}}\mathbf{z})$ is within the noise level and $\sigma=\sqrt{\text{tr}(\boldsymbol{\Sigma})/L}$.

However, from the construction of $\boldsymbol{\zeta}$ in (3.14), i.e., from $\boldsymbol{\zeta}=[\boldsymbol{\zeta}_0^T,\boldsymbol{\zeta}_1^T,\ldots,\boldsymbol{\zeta}_{L-1}^T]^T$ we observe that each $\boldsymbol{\zeta}_l, l=0,1,\ldots,L-1$, is sparse with sparsity level P. Furthermore, the system matrix $\widetilde{\boldsymbol{\Psi}}$ in (3.14) can be expressed as

$$\widetilde{\boldsymbol{\Psi}}=[\widetilde{\boldsymbol{\Psi}}_0 \quad \widetilde{\boldsymbol{\Psi}}_1 \quad \ldots \quad \widetilde{\boldsymbol{\Psi}}_{L-1}], \tag{3.16}$$

where each block-matrix of dimension $LN\times N_\beta$ is orthogonal to any other block-matrix, i.e., $\widetilde{\boldsymbol{\Psi}}_{l_1}^H\widetilde{\boldsymbol{\Psi}}_{l_2}=\mathbf{0}$ for $l_1\neq l_2$.

To exploit this additional structure in the sparse-recovery algorithm, we propose a concentrated estimate $\widehat{\boldsymbol{\zeta}}=[\widehat{\boldsymbol{\zeta}}_0^T,\widehat{\boldsymbol{\zeta}}_1^T,\ldots,\widehat{\boldsymbol{\zeta}}_{L-1}^T]^T$ which is obtained from the individual solutions, $\widehat{\boldsymbol{\zeta}}_l$s, of the L small Dantzig selectors:

$$\min_{\mathbf{z}_l\in\mathbb{C}^{N_\beta}} \quad \|\mathbf{z}_l\|_1 \quad \text{subject to} \quad \left\|\widetilde{\boldsymbol{\Psi}}_l^H(\mathbf{y}-\widetilde{\boldsymbol{\Psi}}_l\mathbf{z}_l)\right\|_\infty \le \lambda_l\cdot\sigma \quad \text{for } l=0,1,\ldots,L-1, \tag{3.17}$$

where $\lambda_l=\sqrt{2\log(N_\beta)}$. As (3.17) exploits more prior structures of the sparse vector, it provides improved performances over (3.15) both in terms of computational-time and estimation-accuracy [48].

3.4 Adaptive Waveform Design

In this section, we develop an adaptive waveform design technique based on a multi-objective optimization (MOO) approach. To improve the detection performance, we propose to maximize the Mahalanobis distance which quantifies the distance between two distributions involved in the detection problem. However, in practice, the computation of the Mahalanobis distance requires estimations of the target scattering-response, target velocity, and noise covariance matrix. So, in addition to maximizing the Mahalanobis distance, we intend to increase the estimation-accuracy by minimizing a weighted trace of the CRB matrix computed for the unknown parameters. Furthermore, the formulation of sparse-measurement model allows us to construct and solve another optimization problem that minimizes the upper bound on the sparse-estimation error for improving the efficiency of sparse-recovery. In the following, we first present in detail these three single-objective functions and then describe the MOO problem.

3.4.1 Maximizing the Mahalanobis Distance

To decide whether a target is present or not in the range cell under test, the standard procedure is to construct a decision problem that chooses between two possible hypotheses: the null hypothesis $\mathscr{H}_0$ (target-free hypothesis) or the alternate hypothesis $\mathscr{H}_1$ (target-present hypothesis). The problem can be expressed as

$$\begin{cases} \mathscr{H}_0 : \mathbf{y} = \mathbf{e}, \\ \mathscr{H}_1 : \mathbf{y} = \boldsymbol{\Psi}\mathbf{x} + \mathbf{e}, \end{cases} \tag{3.18}$$

and the measurement $\mathbf{y}$ is distributed as $\mathbb{C}\mathscr{N}_{LN}(\mathbf{0}, \mathbf{I}_N \otimes \boldsymbol{\Sigma})$ or $\mathbb{C}\mathscr{N}_{LN}(\boldsymbol{\Psi}\mathbf{x}, \mathbf{I}_N \otimes \boldsymbol{\Sigma})$. To distinguish between these two distributions, one standard measure is the squared Mahalanobis distance, defined as

$$\begin{aligned} d^2 &= \mathbf{x}^H \boldsymbol{\Psi}^H (\mathbf{I}_N \otimes \boldsymbol{\Sigma})^{-1} \boldsymbol{\Psi}\mathbf{x} \\ &= \sum_{n=0}^{N-1} \mathbf{x}^H \boldsymbol{\Phi}(n)^H \mathbf{A}^H \boldsymbol{\Sigma}^{-1} \mathbf{A} \boldsymbol{\Phi}(n)\mathbf{x}. \end{aligned} \tag{3.19}$$

Then, to maximize the detection performance, we can formulate an optimization problem as

$$\mathbf{a}^{(1)} = \arg\max_{\mathbf{a}\in\mathbb{C}^L} \left[\sum_{n=0}^{N-1} \mathbf{x}^H \boldsymbol{\Phi}(n)^H \mathbf{A}^H \boldsymbol{\Sigma}^{-1} \mathbf{A} \boldsymbol{\Phi}(n)\mathbf{x} \right] \quad \text{subject to} \quad \mathbf{a}^H\mathbf{a} = 1. \tag{3.20}$$

After some algebraic manipulations (see [48, App. C]) we can rewrite this problem as

$$\mathbf{a}^{(1)} = \arg\max_{\mathbf{a}\in\mathbb{C}^L} \mathbf{a}^H \left[\sum_{n=0}^{N-1} \left(\boldsymbol{\Phi}(n)\mathbf{x}\mathbf{x}^H \boldsymbol{\Phi}(n)^H \right)^T \odot \boldsymbol{\Sigma}^{-1} \right] \mathbf{a} \quad \text{subject to} \quad \mathbf{a}^H\mathbf{a} = 1. \tag{3.21}$$

Hence, the optimization problem reduces to a simple eigenvalue-eigenvector problem, and the solution of (3.21) is the eigenvector corresponding to the largest eigenvalue of

$$\left[\sum_{n=0}^{N-1} \left(\boldsymbol{\Phi}(n)\mathbf{x}\mathbf{x}^H \boldsymbol{\Phi}(n)^H \right)^T \odot \boldsymbol{\Sigma}^{-1} \right].$$

However in practical scenarios, to obtain $\mathbf{a}^{(1)}$ by solving (3.21), we need to estimate the values of $\mathbf{v}$, $\mathbf{x}$, and $\boldsymbol{\Sigma}$.

3.4.2 Minimizing the Weighted Trace of CRB Matrix

To characterize the accuracy of the estimation process, we compute the CRBs on the target velocity, $\mathbf{v}$, and scattering-parameters, $\mathbf{x}$. For mathematical simplicity, we

assume here that the noise covariance matrix, $\mathbf{\Sigma}$, is known. The motivation behind considering the CRB as the performance measure is that it represents a universal lower bound on the variance of all unbiased estimators of a set of parameters. If an estimator is unbiased and attains the CRB, then it is said to be *efficient* in using the measured data. Alternatively, even if there is no unbiased estimator that attains the CRB, finding this lower bound provides a useful theoretical benchmark against which we can compare the performance of any other unbiased estimator [26, Chap. 3].

Considering a ground-moving target with $\mathbf{v} = v_{\mathrm{x}}\hat{i} + v_{\mathrm{y}}\hat{j}$, we define two sets of vectors $\mathbf{g}_l(n)$s and $\mathbf{h}_l(n)$s, for $l = 0, \ldots, L-1,\ n = 0, \ldots, N-1$, respectively as

$$\mathbf{g}_l(n) = (j4\pi f_l nT/c)[u_{\mathrm{x},1}, u_{\mathrm{x},2}, \ldots, u_{\mathrm{x},P}]^T,$$
$$\mathbf{h}_l(n) = (j4\pi f_l nT/c)[u_{\mathrm{y},1}, u_{\mathrm{y},2}, \ldots, u_{\mathrm{y},P}]^T,$$

where $\{u_{\mathrm{x},p}, u_{\mathrm{y},p}\}$ are the components of $\mathbf{u}_p$, i.e., $\mathbf{u}_p = u_{\mathrm{x},p}\hat{i} + u_{\mathrm{y},p}\hat{j}$ for $p = 1, 2, \ldots, P$. Then, denoting the unknown parameter-vector as $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \mathbf{x}^T]^T$, where $\boldsymbol{\eta} = [v_{\mathrm{x}}, v_{\mathrm{y}}]^T$, we get the partial-derivative matrices as

$$\mathbf{D}_{\boldsymbol{\eta}} \triangleq \frac{\partial(\mathbf{\Psi}\mathbf{x})}{\partial \boldsymbol{\eta}} = \left[\frac{\partial \mathbf{\Psi}}{\partial v_{\mathrm{x}}}\mathbf{x} \ \ \frac{\partial \mathbf{\Psi}}{\partial v_{\mathrm{y}}}\mathbf{x}\right], \tag{3.22}$$

$$\mathbf{D}_{\mathbf{x}} \triangleq \frac{\partial(\mathbf{\Psi}\mathbf{x})}{\partial \mathbf{x}} = \mathbf{\Psi}, \tag{3.23}$$

where

$$\begin{aligned} \frac{\partial \mathbf{\Psi}}{\partial v_{\mathrm{x}}} &= \left[\left(\mathbf{A}\frac{\partial \mathbf{\Phi}(0)}{\partial v_{\mathrm{x}}}\right)^T \ldots \left(\mathbf{A}\frac{\partial \mathbf{\Phi}(N-1)}{\partial v_{\mathrm{x}}}\right)^T\right]^T, \\ \frac{\partial \mathbf{\Phi}(n)}{\partial v_{\mathrm{x}}} &= \mathrm{blkdiag}\big(\big(\boldsymbol{\phi}_0(n) \odot \mathbf{g}_0(n)\big)^T, \ldots, \big(\boldsymbol{\phi}_{L-1}(n) \odot \mathbf{g}_{L-1}(n)\big)^T\big), \end{aligned} \tag{3.24}$$

and

$$\begin{aligned} \frac{\partial \mathbf{\Psi}}{\partial v_{\mathrm{y}}} &= \left[\left(\mathbf{A}\frac{\partial \mathbf{\Phi}(0)}{\partial v_{\mathrm{y}}}\right)^T \ldots \left(\mathbf{A}\frac{\partial \mathbf{\Phi}(N-1)}{\partial v_{\mathrm{y}}}\right)^T\right]^T, \\ \frac{\partial \mathbf{\Phi}(n)}{\partial v_{\mathrm{y}}} &= \mathrm{blkdiag}\big(\big(\boldsymbol{\phi}_0(n) \odot \mathbf{h}_0(n)\big)^T, \ldots, \big(\boldsymbol{\phi}_{L-1}(n) \odot \mathbf{h}_{L-1}(n)\big)^T\big). \end{aligned} \tag{3.25}$$

Subsequently, we calculate the CRB on $\boldsymbol{\theta}$ as

$$\mathbf{CRB}(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{CRB}_{\boldsymbol{\eta}\boldsymbol{\eta}} & \mathbf{CRB}_{\boldsymbol{\eta}\mathbf{x}} \\ \mathbf{CRB}_{\mathbf{x}\boldsymbol{\eta}} & \mathbf{CRB}_{\mathbf{x}\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\boldsymbol{\eta}\boldsymbol{\eta}} & \mathbf{J}_{\boldsymbol{\eta}\mathbf{x}} \\ \mathbf{J}_{\mathbf{x}\boldsymbol{\eta}} & \mathbf{J}_{\mathbf{x}\mathbf{x}} \end{bmatrix}^{-1}, \tag{3.26}$$

where the elements of the Fisher information matrix (FIM) are expressed as

$$\begin{aligned}\mathbf{J}_{\boldsymbol{\eta}\boldsymbol{\eta}} &= 2\,\mathrm{Re}\left\{\mathbf{D}_{\boldsymbol{\eta}}^{H}(\mathbf{I}_N \otimes \boldsymbol{\Sigma})^{-1}\mathbf{D}_{\boldsymbol{\eta}}\right\} \\ &= \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\left[\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{x}}}\mathbf{x}\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{y}}}\mathbf{x}\right]^{H}\mathbf{A}^{H}\boldsymbol{\Sigma}^{-1}\mathbf{A}\left[\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{x}}}\mathbf{x}\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{y}}}\mathbf{x}\right]\right\}, \quad (3.27)\end{aligned}$$

$$\begin{aligned}\mathbf{J}_{\boldsymbol{\eta}\mathbf{x}} &= 2\,\mathrm{Re}\left\{\mathbf{D}_{\boldsymbol{\eta}}^{H}(\mathbf{I}_N \otimes \boldsymbol{\Sigma})^{-1}\mathbf{D}_{\mathbf{x}}\right\} \\ &= \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\left[\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{x}}}\mathbf{x}\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{y}}}\mathbf{x}\right]^{H}\mathbf{A}^{H}\boldsymbol{\Sigma}^{-1}\mathbf{A}\boldsymbol{\Phi}(n)\right\}, \quad (3.28)\end{aligned}$$

$$\begin{aligned}\mathbf{J}_{\mathbf{x}\boldsymbol{\eta}} &= 2\,\mathrm{Re}\left\{\mathbf{D}_{\mathbf{x}}^{H}(\mathbf{I}_N \otimes \boldsymbol{\Sigma})^{-1}\mathbf{D}_{\boldsymbol{\eta}}\right\} \\ &= \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\boldsymbol{\Phi}(n)^{H}\mathbf{A}^{H}\boldsymbol{\Sigma}^{-1}\mathbf{A}\left[\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{x}}}\mathbf{x}\frac{\partial \boldsymbol{\Phi}(n)}{\partial v_{\mathrm{y}}}\mathbf{x}\right]\right\}, \quad (3.29)\end{aligned}$$

$$\mathbf{J}_{\mathbf{x}\mathbf{x}} = 2\,\mathrm{Re}\left\{\mathbf{D}_{\mathbf{x}}^{H}(\mathbf{I}_N \otimes \boldsymbol{\Sigma})^{-1}\mathbf{D}_{\mathbf{x}}\right\} = \sum_{n=0}^{N-1} 2\,\mathrm{Re}\left\{\boldsymbol{\Phi}(n)^{H}\mathbf{A}^{H}\boldsymbol{\Sigma}^{-1}\mathbf{A}\boldsymbol{\Phi}(n)\right\}. \quad (3.30)$$

Now, to obtain a scalar objective function that summarizes the CRB matrix, several optimality criteria can be considered. For example, A-optimality criterion employs the trace, D-optimality uses the determinant, and E-optimality computes the maximum eigenvalue of the CRB matrix [6, Chap. 7.5.2]. However, due to the different physical characteristics of $\boldsymbol{\eta}$ and $\mathbf{x}$, the variances of their estimators may differ in several orders of magnitude and units. Therefore, we construct an objective function to design the OFDM spectral-parameters that minimizes a weighted summation of the traces of individual CRBs on $\boldsymbol{\eta}$ and $\mathbf{x}$ as

$$\mathbf{a}^{(2)} = \arg\min_{\mathbf{a}\in\mathbb{C}^{L}} c_{\boldsymbol{\eta}}\,\mathrm{tr}(\mathbf{CRB}_{\boldsymbol{\eta}\boldsymbol{\eta}}) + c_{\mathbf{x}}\,\mathrm{tr}(\mathbf{CRB}_{\mathbf{x}\mathbf{x}}) \quad \text{subject to} \quad \mathbf{a}^{H}\mathbf{a} = 1, \quad (3.31)$$

where $c_{\boldsymbol{\eta}}$ and $c_{\mathbf{x}}$ are the weighting parameters.

3.4.3 Minimizing the Upper Bound on Sparse Error

Many functions of the system matrix $\widetilde{\boldsymbol{\Psi}}$ have been proposed to analyze the performance of methods used to recover $\boldsymbol{\zeta}$ from $\mathbf{y}$, the most popular measure being the restricted isometry constant (RIC). However, for a given arbitrary matrix, the computation of RIC is extremely difficult. Therefore, in [52] we proposed a new, easily computable measure, ℓ_1-constrained minimal singular value (ℓ_1-CMSV) of $\widetilde{\boldsymbol{\Psi}}$, to assess the reconstruction performance of an ℓ_1-based algorithm. According to [52, Def. 3], we define the ℓ_1-CMSV of $\widetilde{\boldsymbol{\Psi}}$ as

$$\rho_s(\widetilde{\boldsymbol{\Psi}}) = \min_{\boldsymbol{\zeta}\neq\mathbf{0}, s_1(\boldsymbol{\zeta})\leq s} \frac{\|\widetilde{\boldsymbol{\Psi}}\boldsymbol{\zeta}\|_2}{\|\boldsymbol{\zeta}\|_2}, \quad \text{for any } s \in [1, LN_\beta], \quad (3.32)$$

and

$$s_1(\boldsymbol{\zeta}) \triangleq \frac{\|\boldsymbol{\zeta}\|_1^2}{\|\boldsymbol{\zeta}\|_2^2} \leq \|\boldsymbol{\zeta}\|_0. \tag{3.33}$$

Then, the performance of our decomposed Dantzig selector (DS) approach in (3.17) is given by the following theorem:

Theorem 3.1 [48] *Suppose $\boldsymbol{\zeta} \in \mathbb{C}^{LN_\beta}$ is an LP-sparse vector having an additional structure as presented in* (3.14)*, with each $\boldsymbol{\zeta}_l \in \mathbb{C}^{N_\beta}$ being a P-sparse vector, and* (3.14) *is the measurement model. Then, with high probability, the concentrated solution $\widehat{\boldsymbol{\zeta}} = [\widehat{\boldsymbol{\zeta}}_0^T, \widehat{\boldsymbol{\zeta}}_1^T, \ldots, \widehat{\boldsymbol{\zeta}}_{L-1}^T]^T$ of* (3.17) *satisfies*

$$\|\widehat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}\|_2 \leq 4 \sqrt{\sum_{l=0}^{L-1} \frac{\lambda_l^2 P \sigma^2}{a_l^4 \rho_{4P}^4(\widetilde{\boldsymbol{\Phi}}_l)}}, \tag{3.34}$$

where $\widetilde{\boldsymbol{\Phi}}_l$ is related with $\widetilde{\boldsymbol{\Psi}}_l$ of (3.16) *as $\widetilde{\boldsymbol{\Psi}}_l = a_l \widetilde{\boldsymbol{\Phi}}_l$. More specifically, if $\lambda_l = \sqrt{2(1+q)\log(N_\beta)}$ for each $q \geq 0$ is used in* (3.17)*, then the bound holds with probability greater than $1 - L(\sqrt{\pi(1+q)\log(N_\beta)} \cdot (N_\beta)^q)^{-1}$.*

Proof See [48]. □

To minimize the upper bound on the sparse-estimation error, we formulate an optimization problem as

$$\mathbf{a}^{(3)} = \arg \min_{\mathbf{a} \in \mathbb{C}^L} \sum_{l=0}^{L-1} \frac{\lambda_l^2 P \sigma^2}{a_l^4 \rho_{4P}^4(\widetilde{\boldsymbol{\Phi}}_l)} \quad \text{subject to} \quad \mathbf{a}^H \mathbf{a} = 1. \tag{3.35}$$

Using the Lagrange-multiplier approach, we can easily obtain the solution of (3.35) as

$$a_l^{(3)} = \sqrt{\frac{(2\alpha_l)^{1/3}}{\sum_{l=0}^{L-1} (2\alpha_l)^{1/3}}}, \quad \text{where } \alpha_l = \frac{\lambda_l^2 P \sigma^2}{\rho_{4P}^4(\widetilde{\boldsymbol{\Phi}}_l)}, \tag{3.36}$$

for $l = 0, 1, \ldots, L-1$.

However, the computation of $\rho_{4P}(\widetilde{\boldsymbol{\Phi}}_l)$ is difficult with the complex variables. Therefore, we use a computable lower bound on $\rho_{4P}(\widetilde{\boldsymbol{\Phi}}_l)$, defined as

$$\rho_{8P}(\boldsymbol{\Gamma}_l) \leq \rho_{4P}(\widetilde{\boldsymbol{\Phi}}_l), \tag{3.37}$$

where

$$\boldsymbol{\Gamma}_l^T \boldsymbol{\Gamma}_l = \begin{bmatrix} \boldsymbol{\Gamma}_1^T \boldsymbol{\Gamma}_1 + \boldsymbol{\Gamma}_2^T \boldsymbol{\Gamma}_2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma}_1^T \boldsymbol{\Gamma}_1 + \boldsymbol{\Gamma}_2^T \boldsymbol{\Gamma}_2 \end{bmatrix}, \quad \boldsymbol{\Gamma}_1 = \mathrm{Re}\, \widetilde{\boldsymbol{\Phi}}_l, \boldsymbol{\Gamma}_2 = \mathrm{Im}\, \widetilde{\boldsymbol{\Phi}}_l. \tag{3.38}$$

Then, similar to (3.36), we can obtain the optimal OFDM spectral-parameters as

$$a_l^{(3)} = \sqrt{\frac{(2\widetilde{\alpha}_l)^{1/3}}{\sum_{l=0}^{L-1}(2\widetilde{\alpha}_l)^{1/3}}}, \quad \text{where } \widetilde{\alpha}_l = \frac{\lambda_l^2 P\sigma^2}{\rho_{8P}^4(\mathbf{\Gamma}_l)}, \tag{3.39}$$

for $l = 0, 1, \ldots, L-1$.

3.4.4 Multi-objective Optimization

From the discussion of the previous subsections, we notice that if the solution of (3.21) is used we would get an improved detection performance provided that we know a-priori the values of the target velocity, scattering-response, and noise covariance matrix. Alternatively, by solving (3.31), we could improve the performances of the underlying estimation problems for the target response and velocity. Furthermore, if we were to use the solution of (3.39), we would achieve an efficient sparse-recovery result when we address the detection problem from a sparse-estimation perspective. Hence, based on these arguments, we device a constrained MOO problem to design the spectral parameters of the OFDM waveform such that simultaneously (i) the squared Mahalanobis distance of the detection problem is maximized, (ii) the weighted summation of the traces of CRB matrices for $\boldsymbol{\eta}$ and $\mathbf{x}$ is minimized, and (iii) the upper bound on the sparse-estimation error of the equivalent sparse-recovery approach is minimized. Mathematically, this is represented as

$$\mathbf{a}_{\text{opt}} = \begin{cases} \arg\max_{\mathbf{a}\in\mathbb{C}^L} \mathbf{a}^H[\sum_{n=0}^{N-1}(\widetilde{\mathbf{\Phi}}(n)\boldsymbol{\zeta}\boldsymbol{\zeta}^H\widetilde{\mathbf{\Phi}}(n)^H)^T \odot \mathbf{\Sigma}^{-1}]\mathbf{a}, \\ \arg\min_{\mathbf{a}\in\mathbb{C}^L} c_{\boldsymbol{\eta}}\,\text{tr}(\mathbf{CRB}_{\boldsymbol{\eta}\boldsymbol{\eta}}) + c_{\mathbf{x}}\,\text{tr}(\mathbf{CRB}_{\mathbf{xx}}), \\ \arg\min_{\mathbf{a}\in\mathbb{C}^L} \sum_{l=0}^{L-1} \frac{\lambda_l^2 P\sigma^2}{a_l^4 \rho_{8P}^4(\mathbf{\Gamma}_l)}, \end{cases} \tag{3.40}$$

subject to $\mathbf{a}^H\mathbf{a} = 1$.

We employ the standard nondominated sorting genetic algorithm II (NSGA-II) to solve our MOO problem, after modifying it to incorporate the constraint $\mathbf{a}^H\mathbf{a} = 1$ that needs to be imposed upon the solutions.

3.5 Numerical Results

In this section, we present the results of several numerical examples to discuss the solutions of the MOO problem and to demonstrate the performance improvement due to the adaptive OFDM-waveform design technique. For simplicity, we consider a 2D scenario, as shown in Fig. 3.3, where both the radar and target are in the same plane. Our analyses can easily be extended to 3D scenarios. First, we provide a description of the simulation setup and then discuss different numerical results.

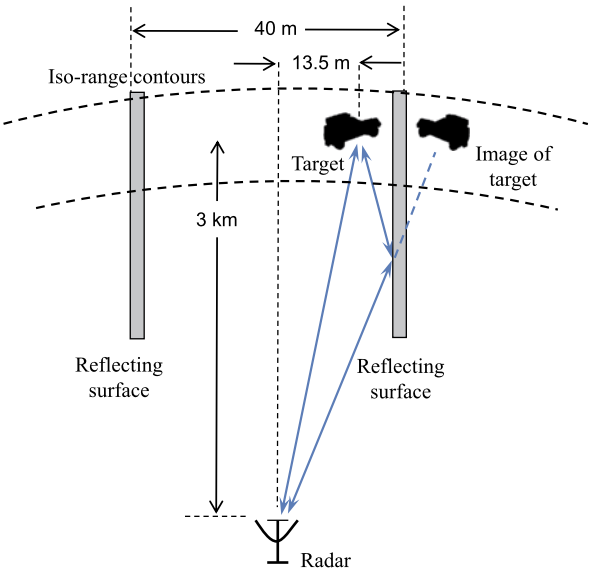


Fig. 3.3 A schematic representation of the multipath scenario considered in the numerical examples

- *Target and multipath parameters*:
  - Throughout a given coherent processing interval (CPI), the target remained within a particular range cell. We simulated the situation of a range cell that is at a distance of 3 km from the radar (positioned at the origin).
  - The target was 13.5 m east from the center line, moving with velocity $\mathbf{v} = (35/\sqrt{2})(\hat{i} + \hat{j})$ m/s.
  - There were two different paths between the target and radar: one direct and one reflected, subtending angles of 0.26° and 0.51°, respectively, with respect to the radar. Hence, the target manifests two relative speeds of $\langle \mathbf{v}, \mathbf{u}_p \rangle = 24.86$ and 24.53 m/s at the radar receiver.
  - The scattering coefficients of the target, $\mathbf{x}$, were varied to simulate two scenarios having different energy-distributions across different subchannels. For example, Scenario I had the strongest target-reflectivity on the second subcarrier with $x_{l,\mathrm{d}}^{(1)} = [2, 4, 1]^T$ and $x_{l,\mathrm{r}}^{(1)} = [1, 2, 0.5]^T$ representing the scattering coefficients along the direct and reflected paths, respectively. On the other hand, in Scenario II we considered the strongest target-reflectivity along the first subcarrier with $x_{l,\mathrm{d}}^{(1)} = [3, 1, 2]^T$ and $x_{l,\mathrm{r}}^{(1)} = [1.5, 0.5, 1]^T$.

    However, for the purpose of a fair comparison, we scaled the target-scattering coefficients to ensure a constant signal-to-noise ratio (SNR), defined as

$$\mathrm{SNR} = \frac{\|\mathbf{x}\|^2}{\mathrm{tr}(\mathbf{\Sigma})}. \tag{3.41}$$

    Hence, a stronger target-reflectivity along a certain subcarrier implied that there would be some other subcarriers with very poor target-reflectivities.

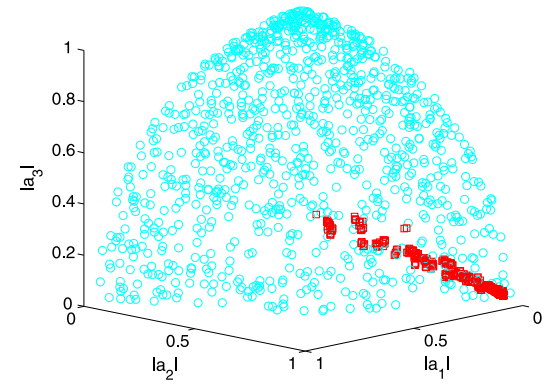
- *Radar parameters*:
  - Carrier frequency $f_c = 1$ GHz;
  - Available bandwidth $B = 10$ MHz;
  - Number of OFDM subcarriers $L = 3$;
  - Subcarrier spacing of $\Delta f = B/(L+1) = 2.5$ MHz;
  - Pulse width $T_p = 1/\Delta f = 400$ ns;
  - Pulse repetition interval $T = 4$ ms;
  - Number of coherent pulses $N = 20$;
  - All the transmit OFDM weights were equal, i.e., $a_l = 1/\sqrt{L}\ \forall l$.
- *Simulation parameters*:
  To apply a sparse estimation, we partitioned the viable relative speeds from 24.5 to 25 m/s with steps of 0.05 m/s. We generated the noise samples from a $\mathbb{C}\mathcal{N}_{LN}(\mathbf{0}, \mathbf{I}_N \otimes \mathbf{\Sigma})$ distribution with $\mathbf{\Sigma} = [1, 0.1, 0.01; 0.1, 1, 0.1; 0.01, 0.1, 1]$. Hence, for all the results presented in this section we ensured a constant noise-power distribution among all the subchannels.

  To solve the MOO problem (3.40), we employed the NSGA-II with the following parameters: population size = 1000, number of generations = 100, crossover probability = 0.9, and mutation probability = 0.1. The initial population of 1000 different values of $\mathbf{a}$ were generated randomly, but ensuring that the total-energy constraint $\mathbf{a}^H\mathbf{a} = 1$ was satisfied. Furthermore, at each generation of the NSGA-II, we imposed the total-energy constraint on the children-chromosomes by introducing an 'if-statement' in the 'genetic-operator' portion of the NSGA-II code. However, satisfying the hard-equality constraint $\mathbf{a}^H\mathbf{a} = 1$ was difficult to simulate due to the numerical precision errors. That is why we relaxed it with a softer constraint by considering $0.999 \leq \mathbf{a}^H\mathbf{a} \leq 1.001$.

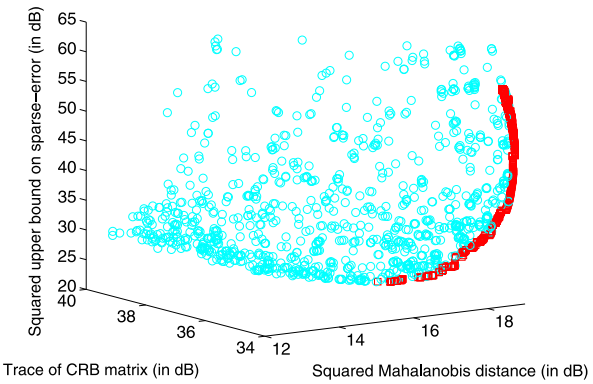
3.5.1 Results of the MOO Problem

The results of the MOO problem are depicted in Figs. 3.4, 3.5 and Figs. 3.6, 3.7 for the target Scenarios I and II, respectively. We maintained a fixed SNR of 0 dB for these simulations. The initial population of 1000 different values of $\mathbf{a}$ were generated randomly. Considering a Cartesian coordinate system with $|a_{1_{\text{opt}}}|$, $|a_{2_{\text{opt}}}|$, and $|a_{3_{\text{opt}}}|$ as the axes, the initial population is represented on the surface of a sphere restricted to the first octant, as shown by circles in Figs. 3.4(a) and 3.6(a) for the two different target scenarios. The values of the associated objective functions are also indicated by circles respectively in Figs. 3.4(b) and 3.6(b), whose coordinate systems are constructed with the three objective functions representing the axes on the logarithmic scales.

We represent the Pareto-optimal solutions by squares in Figs. 3.4(a) and 3.6(a) and the associated Pareto-optimal objective values by squares in Figs. 3.4(b) and 3.6(b), for the Scenarios I and II, respectively. In Scenario I, when the target had the strongest reflectivity along the second subcarrier, we got the optimal solutions varying from $|\mathbf{a}_{\text{opt}}| = [0.6189, 0.6691, 0.4119]^T$ to $|\mathbf{a}_{\text{opt}}| = [0, 1, 0]^T$ on an



(a)



(b)

Fig. 3.4 Results of the NSGA-II in Scenario I: (**a**) optimal solutions and (**b**) values of the objective functions at the zeroth and 100th generations are respectively represented by *circles* and *squares*

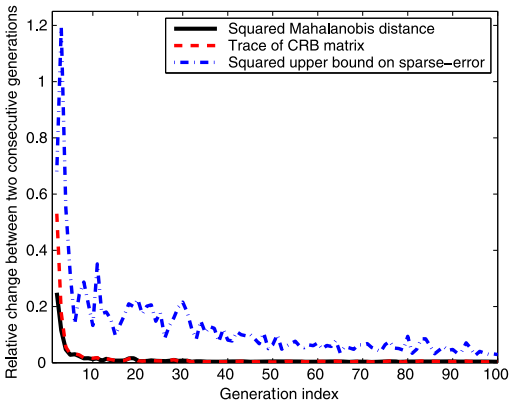


Fig. 3.5 Convergence of the objective functions to the Pareto-optimal values in Scenario I

approximately straight-line locus subtended on the surface of a sphere. It is important to note here that if we had solved only (3.39) to minimize the upper bound on the sparse-error, then the solution would have been $|\mathbf{a}^{(3)}| = [0.6261, 0.6578, 0.4187]^T$;

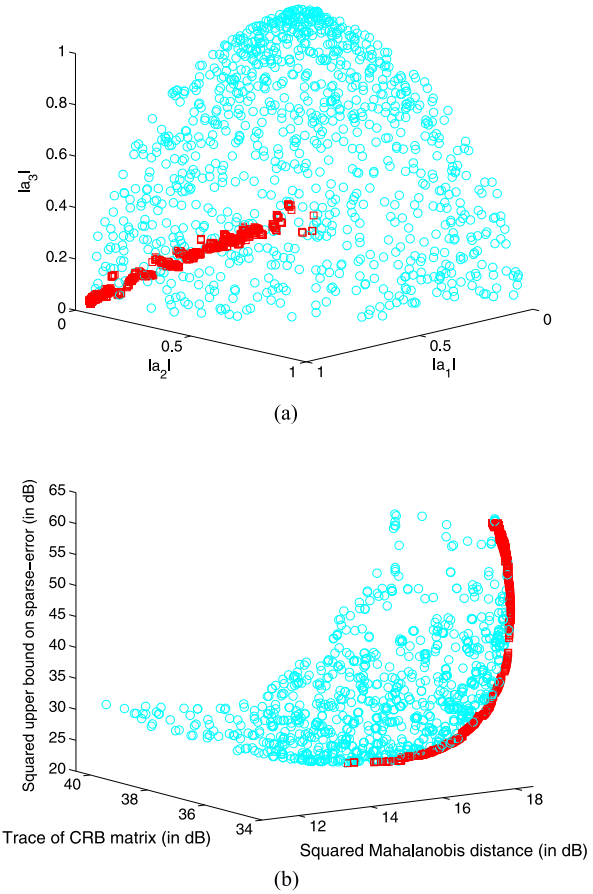


Fig. 3.6 Results of the NSGA-II in Scenario II: (**a**) optimal solutions and (**b**) values of the objective functions at the zeroth and 100th generations are respectively represented by *circles* and *squares*

whereas an effort to obtain the solution of only (3.21) would result in $|\mathbf{a}^{(1)}| = [0.0652, 0.9975, 0.0262]^T$. This implies that Pareto-optimal solutions provide a set of compromised solutions varying in between two extrema that are approximately equal to the individual solutions of the objective functions when solved separately.

Similarly, for the Scenario II which had the strongest target-reflectivity along the first subcarrier, we found that the MOO-solutions lied on an approximately straight-line locus drawn on the surface of a sphere and varied from $|\mathbf{a}_{\text{opt}}| = [0.6339, 0.6510, 0.4182]^T$ to $|\mathbf{a}_{\text{opt}}| = [1, 0, 0]^T$. Comparing with the individual solutions of the objective functions in Scenario II, we noticed that (3.21) resulted in $|\mathbf{a}^{(1)}| = [0.9992, 0.0374, 0.0015]^T$; whereas the solution of (3.39) still produced $|\mathbf{a}^{(3)}| = [0.6261, 0.6578, 0.4187]^T$ because it was a function of the system matrix $\widetilde{\mathbf{\Psi}}$ only.

In addition, to assess the speed of convergence to these Pareto-optimal solutions, in Figs. 3.5 and 3.7 we depict the relative change in values of the three objective functions at different generation indices for both the target scenarios under con-

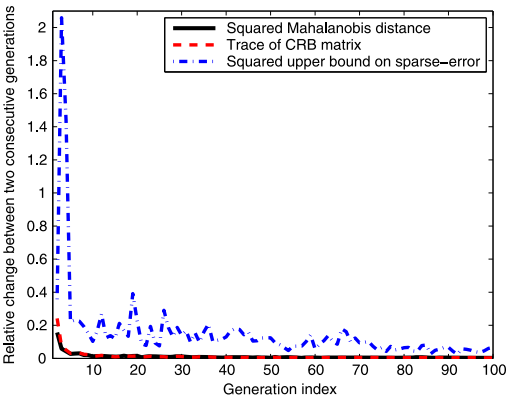


Fig. 3.7 Convergence of the objective functions to the Pareto-optimal values in Scenario II

sideration. If $\mathbf{o}_j(k-1)$ and $\mathbf{o}_j(k)$, for $j = 1, 2, 3, k = 2, 3, \ldots, 100$, denote two vectors of objective-functions respectively computed at the $(k-1)$th and kth generations over the entire population, then their relative changes were calculated as $\|\mathbf{o}_j(k) - \mathbf{o}_j(k-1)\|/\|\mathbf{o}_j(k)\|$. It is quiet evident from these plots that the Pareto-optimal solutions were reached very quickly even within the tenth generation, particularly for the first two objective functions (3.21) and (3.31).

3.5.2 Improvement in Detection and Estimation Performance

We demonstrate the performance improvement due to the adaptive waveform design at several SNR values in terms of the squared Mahalanobis distance, weighted trace of CRB matrix, and squared upper bound on sparse-error. These results are shown in Figs. 3.8 and 3.9 for the target Scenarios I and II, respectively. As we expect, the Mahalanobis-distance measure improved as we increased the SNR values, but the trace of CRB matrix decreased and the upper bound on the sparse-estimation error remained unchanged. In each figure, the red-colored lines (in total 1000 of them) represent the variations of the objective-functions, associated with the entire population of 1000 solutions; whereas the blue-colored line shows their counterparts corresponding to the fixed (nonadaptive) waveform $\mathbf{a} = 1/\sqrt{L} = [0.5774, 0.5774, 0.5774]^T$. In both the target scenarios, we found that all the Pareto-optimal solutions produced better performances, in terms of the Mahalanobis distance and trace of CRB matrix, when compared to those with the fixed waveform. However, with respect to the squared upper bound on sparse-error, only a subset of the Pareto-optimal solutions was found to show improved performance than that with the fixed waveform.

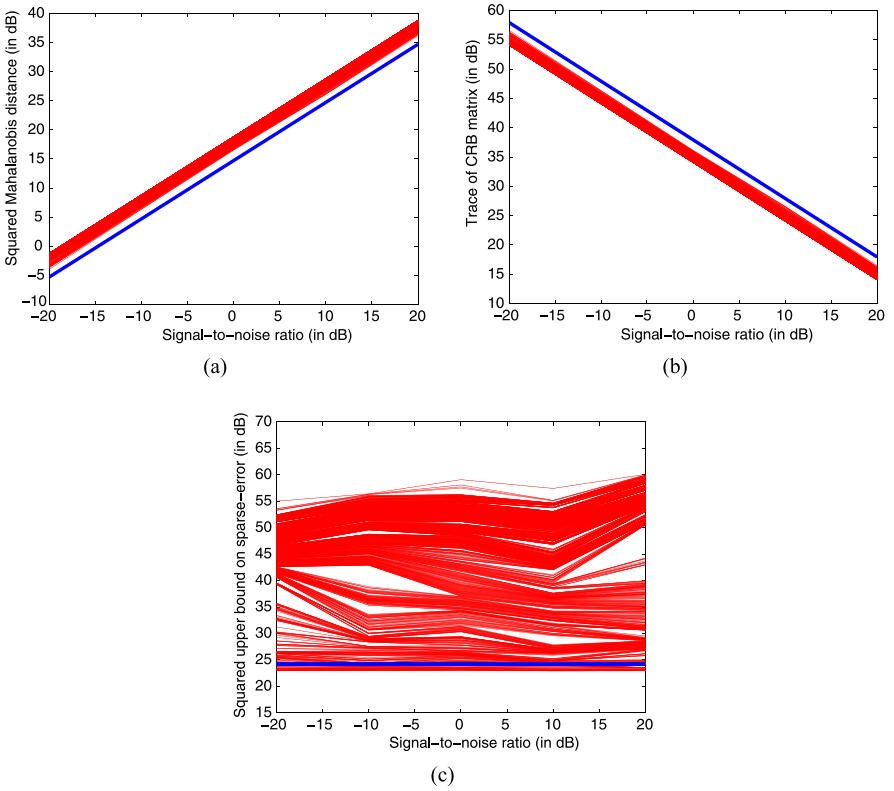


Fig. 3.8 Comparison of performances due to the fixed and adaptive waveforms in Scenario I in terms of the (**a**) squared Mahalanobis distance, (**b**) weighted trace of Cramér–Rao bound matrix, and (**c**) squared upper bound on sparse-error, respectively

3.5.3 Redistributions of Signal and Target Energies

To understand the reason behind the performance improvement due to the adaptive waveform design, we looked into the energy-distribution of the transmitted signal and effective target-return across different subchannels both before and after the waveform design. We used the subset of Pareto-optimal solutions that satisfied all the three objective functions at 0 dB to exemplify the results on energy-redistribution for both the target scenarios in Fig. 3.10.

We represent the effective transmit-signal energy at different subchannels as

$$\varepsilon_{\mathrm{S},l} = |a_l|^2, \quad \text{for } l = 0, 1, \ldots, L-1. \tag{3.42}$$

On the other hand, the effective target-returns across different subchannels are considered as

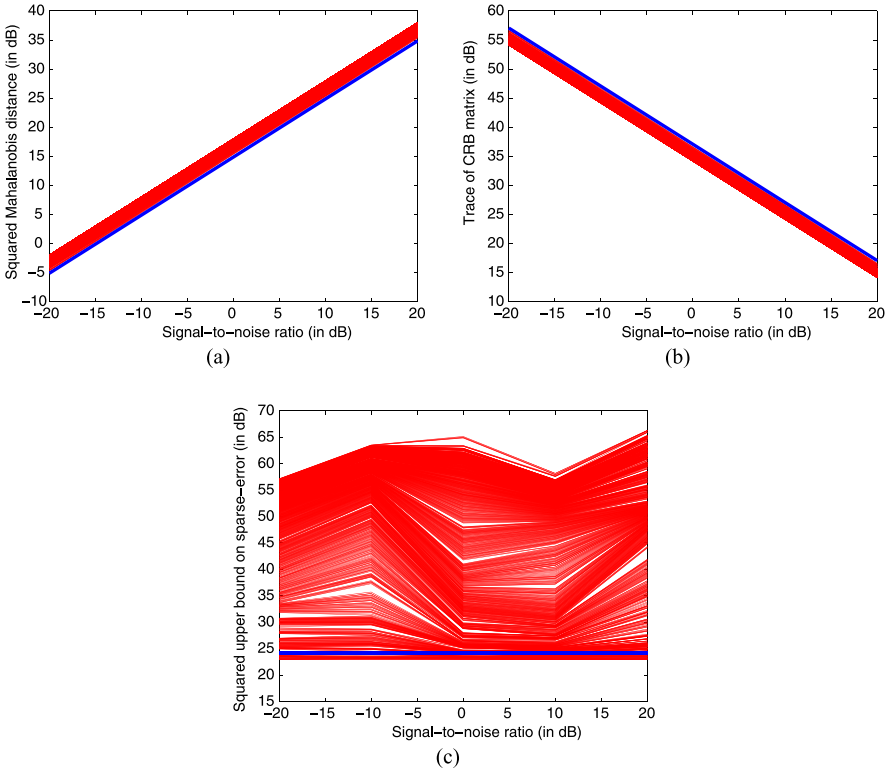


Fig. 3.9 Comparison of performances due to the fixed and adaptive waveforms in Scenario II in terms of the (**a**) squared Mahalanobis distance, (**b**) weighted trace of Cramér–Rao bound matrix, and (**c**) squared upper bound on sparse-error, respectively

$$\varepsilon_{\mathrm{T},l} = \left| \frac{1}{N} \sum_{n=0}^{N-1} a_l \boldsymbol{\phi}_l(n)^T \mathbf{x}_l \right|^2, \quad \text{for } l = 0, 1, \ldots, L-1. \tag{3.43}$$

Due to the adaptive design of a_ls, the set of values of $\{\varepsilon_{\mathrm{S},l}\}$ and $\{\varepsilon_{\mathrm{T},l}\}$ were different before and after the optimization process, as both of these quantities depend on the transmitted signal parameters.

In Fig. 3.10(a), we plot the values of $\{\varepsilon_{\mathrm{S},l}\}$ and $\{\varepsilon_{\mathrm{T},l}/(\sum_{l=0}^{L-1} \varepsilon_{\mathrm{T},l})\}$ for the target Scenario I. Noticing the two left-most vertical bars, we observe that the MOO-approach boosted up the transmitted-signal energy on the second subchannel along which the target-reflectivity was the strongest. Additionally, we found a considerable amount of energy redistribution among different subchannels for the effective target-returns, as shown in the two right-most vertical bars. With the fixed waveform, we had $\{\varepsilon_{\mathrm{T},l}\} = \{0.1527, 0.7868, 0.0605\}$ before the waveform design; whereas after we obtained the Pareto-optimal solution, the normalized values of $\{\varepsilon_{\mathrm{T},l}\}$ changed to $\{0.1075, 0.8752, 0.0173\}$. Hence, we conclude that the MOO-based optimal waveform design puts more signal-energy into that particular sub-

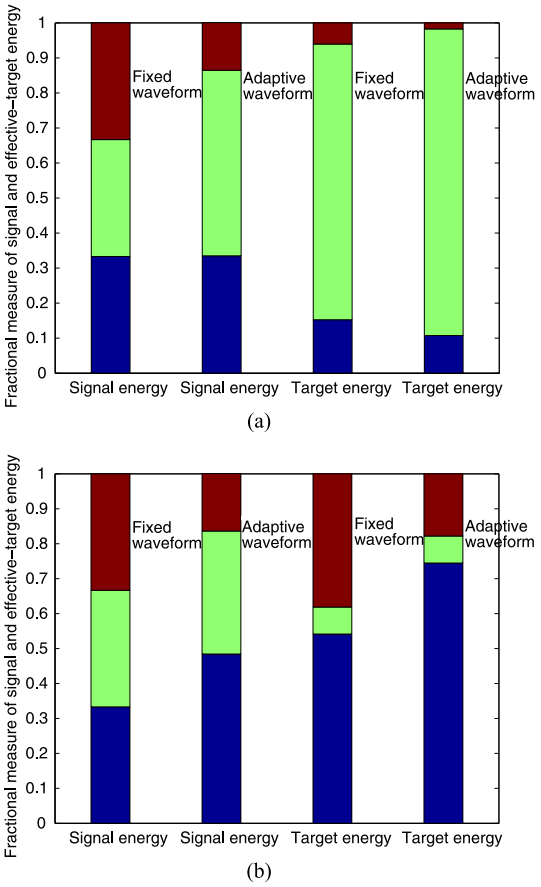


Fig. 3.10 The normalized energy distributions of the transmit-signal and effective target-returns across different subchannels in (**a**) Scenario I and (**b**) Scenario II

carrier at which the target response is stronger, and thus makes the effective target-return more prominent along that subcarrier.

As a further confirmation, we did a similar analysis with values of $\{\varepsilon_{\mathrm{S},l}\}$ and $\{\varepsilon_{\mathrm{T},l}/(\sum_{l=0}^{L-1}\varepsilon_{\mathrm{T},l})\}$ for Scenario II. Results are shown in Fig. 3.10(b). Observing the two left-most vertical bars, we again notice that the transmitted-signal energy was amplified along the first subchannel after the waveform design. The two right-most vertical bars indicate a noticeable redistribution of the effective target-energies among the different subchannels. After the adaptive waveform design, we found that the normalized values of $\{\varepsilon_{\mathrm{T},l}\}$ changed from $\{0.5414, 0.0775, 0.3811\}$ to $\{0.7447, 0.0775, 0.1779\}$. This reconfirms our conclusion that the Pareto-optimal waveform design tries to further enhance the stronger target-returns. Moreover, since we kept the noise-energy fixed and varied only the target-energy over different subchannels, we can extend our conclusion to assert that the solution of the Pareto-optimal design redistributes the energy of the transmitted signal by putting the most energy to that particular subcarrier in which the signal-to-noise ratio is the strongest.

3.6 Conclusions

In this chapter, we proposed a multi-objective optimization (MOO) technique to design the spectral parameters of an orthogonal frequency division multiplexing (OFDM) radar signal for detecting a moving target in the presence of multipath reflections. The use of an OFDM signal increased the frequency diversity of our system, as different scattering centers of a target resonate variably at different frequencies. We first developed a parametric OFDM measurement model for a particular range cell under test, and then converted it to a sparse model that accounted for the target returns over all possible signal paths and target velocities. In our model, the nonzero components of the sparse vector were equal to the scattering coefficients of the target at the true signal paths and target velocity. To estimate the sparse vector, we employed a collection of multiple small Dantzig selectors that utilized more prior structures of the sparse vector. In addition, we proposed a criterion to optimally design the OFDM spectral parameters for the next coherent processing interval based on the MOO approach. We applied the nondominated sorting genetic algorithm II (NSGA-II) to solve a constrained MOO problem that simultaneously optimizes three objective functions: maximizes the Mahalanobis distance to improve the detection performance, minimizes the weighted trace of the Cramér–Rao bound matrix for the unknown parameters to increase the estimation accuracy, and minimizes the upper bound on the sparse-error to improve the efficiency of the equivalent sparse-estimation approach.

We presented several numerical examples to discuss the solutions of the MOO problem and to demonstrate the achieved performance improvement due to the adaptive OFDM-waveform design. As expected, we noticed that the solutions residing on the Pareto-front were compromisable in nature and they varied in between two extrema that were approximately equal to the individual solutions of the objective functions when solved independently. We found that only a subset of the Pareto-optimal solutions produced better performance than a fixed waveform with respect to the all three objective functions. When the noise powers over different subcarriers were the same, we further inferred that the Pareto-optimal solutions put the most transmitted signal-energy to that particular subcarrier along which the signal-to-noise ratio is the strongest.

In our future work, we will extend our model to incorporate more realistic physical effects, such as diffractions and refractions, which exist, for example, due to sharp edges and corners of the buildings or rooftops in an urban environment. We will incorporate other waveform design criteria, e.g., ambiguity function and similarity constraint, into the MOO algorithm. In addition, we will validate the performance of our proposed adaptive waveform design technique with real data.

Acknowledgements This work was supported by the AFOSR Grant FA9550-11-1-0210 and ONR Grant N000140810849.

References

1. Amuso, V.J., Enslin, J.: The Strength Pareto Evolutionary Algorithm 2 (SPEA2) applied to simultaneous multi-mission waveform design. In: Proc. Intl. Waveform Diversity & Design Conf., Pisa, Italy, pp. 407–417 (2007)
2. Amuso, V.J., Josefiak, B.: A distributed object-oriented multi-mission radar waveform design implementation. In: Proc. Intl. Waveform Diversity & Design Conf., Kauai, HI, pp. 266–270 (2012)
3. Anderson, T.W.: An Introduction to Multivariate Statistical Analysis, 3rd edn. Wiley, Hoboken (2003)
4. Antonik, P., Wicks, M.C., Griffiths, H.D., Baker, C.J.: Multi-mission multi-mode waveform diversity. In: Proc. IEEE Radar Conf., Verona, NY, pp. 580–582 (2006)
5. Bell, M.: Information theory and radar waveform design. IEEE Trans. Inf. Theory **39**(5), 1578–1597 (1993)
6. Boyd, S.P., Vandenberghe, L.: Convex Optimization. Cambridge University Press, New York (2004)
7. Calderbank, R., Howard, S., Moran, B.: Waveform diversity in radar signal processing. IEEE Signal Process. Mag. **26**(1), 32–41 (2009)
8. Candès, E., Tao, T.: The Dantzig selector: statistical estimation when p is much larger than n. Ann. Stat. **35**(6), 2313–2351 (2007)
9. Candès, E.J.: The restricted isometry property and its implications for compressed sensing. C. R. Math. **346**(9–10), 589–592 (2008)
10. Candès, E.J., Tao, T.: Decoding by linear programming. IEEE Trans. Inf. Theory **51**(12), 4203–4215 (2005)
11. Coello, C.A.C., Lamont, G.B., Veldhuizen, D.A.V.: Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edn. Springer, New York (2007)
12. De Maio, A., De Nicola, S., Huang, Y., Zhang, S., Farina, A.: Code design to optimize radar detection performance under accuracy and similarity constraints. IEEE Trans. Signal Process. **56**(11), 5618–5629 (2008)
13. De Maio, A., De Nicola, S., Huang, Y., Palomar, D.P., Zhang, S., Farina, A.: Code design for radar STAP via optimization theory. IEEE Trans. Signal Process. **58**(2), 679–694 (2010)
14. De Maio, A., Huang, Y., Piezzo, M., Zhang, S., Farina, A.: Design of optimized radar codes with a peak to average power ratio constraint. IEEE Trans. Signal Process. **59**(6), 2683–2697 (2011)
15. De Maio, A., Piezzo, M., Farina, A., Wicks, M.: Pareto-optimal radar waveform design. IET Radar Sonar Navig. **5**(4), 473–482 (2011)
16. De Maio, A., Piezzo, M., Iommelli, S., Farina, A.: Design of Pareto-optimal radar receive filters. Int. J. Electron. Telecommun. **57**(4), 477–481 (2011)
17. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms, 1st edn. Wiley, New York (2001)
18. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
19. DeLong, D., Hofstetter, E.: On the design of optimum radar waveforms for clutter rejection. IEEE Trans. Inf. Theory **13**(3), 454–463 (1967)
20. Edgeworth, F.Y.: Mathematical Physics: An Essay on the Application of Mathematics to the Moral Sciences. C. K. Paul & Co., London (1881)
21. Enslin, J.W.: An evolutionary algorithm approach to simultaneous multi-mission radar waveform design. Master's thesis, Rochester Institute of Technology, Rochester, NY (2007)
22. Garren, D.A., Odom, A.C., Osborn, M.K., Goldstein, J.S., Pillai, S.U., Guerci, J.R.: Full-polarization matched-illumination for target detection and identification. IEEE Trans. Aerosp. Electron. Syst. **38**(3), 824–837 (2002)
23. Gini, F., De Maio, A., Patton, L.K.: Waveform design and diversity for advanced radar systems. Inst of Engineering & Technology (2011)

24. Hurtado, M., Zhao, T., Nehorai, A.: Adaptive polarized waveform design for target tracking based on sequential Bayesian inference. IEEE Trans. Signal Process. **56**(3), 1120–1133 (2008)
25. Kay, S.: Optimal signal design for detection of Gaussian point targets in stationary Gaussian clutter/reverberation. IEEE J. Sel. Top. Signal Process. **1**(1), 31–41 (2007)
26. Kay, S.M.: Fundamentals of Statistical Signal Processing: Estimation Theory. Prentice Hall PTR, Upper Saddle River (1993)
27. Knott, E.F.: Radar cross section. In: Skolnik, M.I. (ed.) Radar Handbook, 2nd edn. McGraw-Hill, Inc., New York (1990). Chap. 11
28. Krolik, J.L., Farrell, J., Steinhardt, A.: Exploiting multipath propagation for GMTI in urban environments. In: Proc. IEEE Radar Conf., pp. 65–68 (2006)
29. Leshem, A., Naparstek, O., Nehorai, A.: Information theoretic adaptive radar waveform design for multiple extended targets. IEEE J. Sel. Top. Signal Process. **1**(1), 42–55 (2007)
30. Li, J., Guerci, J.R., Xu, L.: Signal waveform's optimal-under-restriction design for active sensing. IEEE Signal Process. Lett. **13**(9), 565–568 (2006)
31. Li, J., Xu, L., Stoica, P., Forsythe, K.W., Bliss, D.W.: Range compression and waveform optimization for MIMO radar: a Cramér–Rao bound based study. IEEE Trans. Signal Process. **56**(1), 218–232 (2008)
32. Mahalanobis, P.C.: On the generalized distance in statistics. Proc. Natl. Inst. Sci. India **2**, 49–55 (1936)
33. Marler, R.T., Arora, J.S.: Survey of multi-objective optimization methods for engineering. Struct. Multidiscip. Optim. **26**(6), 369–395 (2004)
34. May, T., Rohling, H.: Orthogonal frequency division multiplexing. In: Molisch, A.F. (ed.) Wideband Wireless Digital Communications, pp. 17–25. Prentice Hall PTR, Upper Saddle River (2001)
35. Nehorai, A., Gini, F., Greco, M.S., Suppappola, A.P., Rangaswamy, M.: Introduction to the issue on adaptive waveform design for agile sensing and communication. IEEE J. Sel. Top. Signal Process. **1**(1), 2–5 (2007)
36. Pandharipande, A.: Principles of OFDM. IEEE Potentials **21**(2), 16–19 (2002)
37. Pareto, V.: Cours D'Economie Politique, vols. I and II. F. Rouge, Lausanne (1896)
38. Patton, L.K.: On the satisfaction of modulus and ambiguity function constraints in radar waveform optimization for detection. Ph.D. thesis, Wright State University, Dayton, OH (2009)
39. Patton, L.K., Rigling, B.D.: Modulus constraints in adaptive radar waveform design. In: Proc. IEEE Radar Conf., Rome, Italy, pp. 1–6 (2008)
40. Patton, L.K., Rigling, B.D.: Autocorrelation and modulus constraints in radar waveform optimization. In: Proc. Intl. Waveform Diversity & Design Conf., Kissimmee, FL, pp. 150–154 (2009)
41. Pillai, S.U., Oh, H.S., Youla, D.C., Guerci, J.R.: Optimal transmit-receiver design in the presence of signal-dependent interference and channel noise. IEEE Trans. Inf. Theory **46**(2), 577–584 (2000)
42. Rihaczek, A.W.: Optimum filters for signal detection in clutter. IEEE Trans. Aerosp. Electron. Syst. **AES-1**, 297–299 (1965)
43. Rummler, W.D.: A technique for improving the clutter performance of coherent pulse train signals. IEEE Trans. Aerosp. Electron. Syst. **AES-3**(6), 898–906 (1967)
44. Sen, S., Nehorai, A.: OFDM MIMO radar with mutual-information waveform design for low-grazing angle tracking. IEEE Trans. Signal Process. **58**(6), 3152–3162 (2010)
45. Sen, S., Nehorai, A.: Adaptive OFDM radar for target detection in multipath scenarios. IEEE Trans. Signal Process. **59**(1), 78–90 (2011)
46. Sen, S., Hurtado, M., Nehorai, A.: Adaptive OFDM radar for detecting a moving target in urban scenarios. In: Proc. Intl. Waveform Diversity & Design (WDD) Conf., Orlando, FL, pp. 268–272 (2009)
47. Sen, S., Tang, G., Nehorai, A.: Multi-objective optimized OFDM radar waveform for target detection in multipath scenarios. In: 44th Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA (2010)

48. Sen, S., Tang, G., Nehorai, A.: Multi-objective optimization of OFDM radar waveform for target detection. IEEE Trans. Signal Process. **59**(2), 639–652 (2011)
49. Sira, S.P., Papandreou-Suppappola, A., Morrell, D.: Dynamic configuration of time-varying waveforms for agile sensing and tracking in clutter. IEEE Trans. Signal Process. **55**(7), 3207–3217 (2007)
50. Skolnik, M.I.: Introduction to Radar Systems, 3rd edn. McGraw-Hill, New York (2002)
51. Spafford, L.: Optimum radar signal processing in clutter. IEEE Trans. Inf. Theory **14**(5), 734–743 (1968)
52. Tang, G., Nehorai, A.: Performance analysis of sparse recovery based on constrained minimal singular values. IEEE Trans. Signal Process. **59**(12), 5734–5745 (2011)
53. Van Trees, H.L.: Optimum signal design and processing for reverberation-limited environments. IEEE Trans. Mil. Electron. **9**(3), 212–229 (1965)
54. Weinmann, F.: Frequency dependent RCS of a generic airborne target. In: URSI Int. Symp. Electromagnetic Theory (EMTS), Berlin, Germany, pp. 977–980 (2010)
55. Wicks, M.C.: A brief history of waveform diversity. In: Proc. IEEE Radar Conf., Pasadena, CA, pp. 1–6 (2009)
56. Wicks, M.C., Mokole, E., Blunt, S., Schneible, R., Amuso, V. (eds.): Principles of Waveform Diversity and Design. SciTech Pub., Raleigh (2010)
57. Zitzler, E., Laumanns, M., Bleuler, S.: A tutorial on evolutionary multiobjective optimization. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) Metaheuristics for Multiobjective Optimisation. Springer Lecture Notes in Economics and Mathematical Systems, vol. 535, pp. 3–37. Springer, Berlin (2004)

Chapter 4
Multi-object Tracking Using Particle Swarm Optimization on Target Interactions

Bogdan Kwolek

Abstract In this work, a particle swarm optimization based algorithm for multi-target tracking is presented. At the beginning of each frame, the objects are tracked individually using highly discriminative appearance models among different targets. The task of object tracking is considered as a numerical optimization problem, where a particle swarm optimization is used to track the local mode of the similarity measure. The objective function is built on the region covariance matrix and multi-patch based object representation. The target locations and velocities that are determined in such a way are further employed in a particle swarm optimization based algorithm, which refines the trajectories extracted in the first phase. Afterwards, a conjugate method is used in the final optimization. Thus, the particle swarm algorithm is utilized to seek good local minima and the conjugate gradient is used to find the local minimum accurately. At this stage, we optimize complex energy functions which represent the presence, movement and interaction of all targets in sequence of recent frames within a temporal window. The algorithm has been evaluated on publicly available datasets. The experimental results show performance improvement over relevant algorithms.

4.1 Introduction

Visual tracking of multiple objects is a challenging problem. The aim is to infer the states of all targets in the scene and to maintain their identity over time. Despite significant progress in this area, reliable tracking of multiple targets is still a great challenge, particularly in crowded scenes. Many different methods [2, 6, 10, 17, 22, 23] have been proposed in the last decade. One solution to multiple object tracking is the use of multiple trackers, where each tracker is responsible for tracking one object. The so-called tracking-by-detection algorithms [8] gained considerable attention in this area of the research. A widely used approach to multi-target tracking consists in exploiting a joint state-space representation, which concatenates all of the targets'

B. Kwolek (✉)
Rzeszow University of Technology, Al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
e-mail: bkwolek@prz.edu.pl

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_4,

states together [23], or inferring this joint data association problem by estimating all possible associations between the targets and the observations [17, 24]. In contrast to the above mentioned approaches, in order to achieve multi-target tracking, the multiple parallel filters where a single filter per target has its own state space were proposed in [9]. However, when the interactions among the moving targets take place, difficulties in maintaining the correct object identities might arise. Therefore, modeling the interactions among targets and occlusion reasoning play an incredibly important role in multi-target tracking. Khan et al. [17] use a Markov Random Field (MRF) motion prior to modeling the interactions among targets. Andriyenko et al. [2] propose a model for global occlusion reasoning. In an approach that is based on particle swarm optimization [30], the object interactions are modeled as species competition and repulsion. Particle Swarm Optimization (PSO) is a population based stochastic optimization technique [16] which shares many similarities with evolutionary computation techniques. It has been shown to perform well on many nonlinear and multimodal optimization problems.

Visual object tracking is an important ingredient of any multi-object tracking algorithm. Particle filters [13] are one of the most efficient techniques for object tracking. They were successfully applied in many visual tracking applications [28], including multi-object tracking [8, 23]. The task of object tracking can be considered as a numerical optimization problem, where a local optimization is used to track the local mode of the similarity measure in a parameter space of translation, rotation, and scale. In [29], it was shown that, in tasks consisting in tracking a face or a human, a particle swarm optimization-based tracker outperforms a tracker built on a particle filter in terms of accuracy.

Visual object tracking using particle swarm optimization has been an active research area for several years [18, 19]. Recently, particle swarm optimization was proposed to achieve full body motion tracking [14, 20, 31]. The particle swarm optimization, which is a population-based searching technique, has high search efficiency by combining a local search (using self-experience) and a global one (using neighbor experience). In particular, a few simple rules result in high effectiveness of exploration of a high-dimensional search space. In contrast, in a particle filter, the samples do not exchange information and do not communicate with each other, and thus they have reduced capability of exploring huge search spaces.

In this work, we present a PSO based algorithm for multi-target tracking. At the beginning of each frame, the targets are tracked individually using highly discriminative appearance models among different targets. Each of them is tracked on the basis of separate particle swarm optimizations. The target locations and velocities that are determined by independent trackers are further employed in a particle swarm optimization based algorithm which refines the trajectories extracted in the first phase. Afterwards, a conjugate method is used in the final optimization. At this stage, we utilize a complex energy function which represents the presence, movement, and interaction of all targets within a temporal window consisting of the recent frames.

4.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) [16] is a global optimization algorithm to find the minimum of a numerical function. PSO is a derivative-free, stochastic and population-based computational method often used to optimize functions in rather unfriendly non-convex, non-continuous search spaces. It maintains a swarm of particles, where each one represents a candidate solution. Particles are placed in the search space and move through such a space according to rules which take into account each particle's personal knowledge and the global knowledge of the swarm. Every particle moves with its own velocity in the multidimensional search space, determines its own position, and calculates its fitness using an objective function $f(x)$. Each particle follows simple position and velocity update equations; yet, as particles interact, the collective behavior arises, and the interactions between particles lead to the emergence of global and collective search capabilities, which allow the particles to gravitate towards the global extremum.

At the beginning of the optimization, each individual is initialized with a random position and velocity. While seeking for the best fitness, every individual is attracted towards a position which is affected by the best position p_i found so far by itself and the global best position g found by the whole swarm. In every iteration k, each particle's velocity is first updated based on the particle's current velocity, the particle's local information, and global swarm information. Then, each particle's position is updated using this velocity. The position and velocity of particle i are calculated as follows:

$$v^{(i,k+1)} = \omega v^{(i,k)} + c_1 r_1 \left(p^{(i)} - x^{(i,k)}\right) + c_2 r_2 \left(g - x^{(i,k)}\right), \tag{4.1}$$

$$x^{(i,k+1)} = x^{(i,k)} + v^{(i,k+1)}, \tag{4.2}$$

where the constants c_1 and c_2 are used to balance the influence of the individual's knowledge and that of the group, respectively, r_1 and r_2 are uniformly distributed random numbers, $x^{(i)}$ is position of the ith particle, $p^{(i)}$ is the local best position of particle i, whereas g stands for the global best position, and ω is an inertia constant. The swarm stops the search when a termination criterion is met.

Particles can be attached to each other by any kind of neighborhood topology represented by a graph. In the fully connected neighborhood topology, which is represented by a fully connected graph, all particles in the swarm are connected to one another. Each particle in a swarm represents a candidate solution of the problem. With respect to a fitness function, the best location that has been visited thus far by a particle is stored in the particle's memory. The fitness values corresponding to such best positions are also stored. Additionally, the particles have access to the best location of the whole swarm, i.e., a position that yielded the highest fitness value. A particle therefore employs the best position encountered by itself and the best position of the swarm to move itself toward the optimal value of the objective function.

4.3 PSO-Based Object Tracking

The visual object tracking can be perceived as a dynamic optimization problem. In the PSO-based tracking, in each frame, the object's state is determined using a fitness function expressing the object's appearance. In order to cover possible state changes between consecutive images, the particles are propagated according to a weak transition model. In this section, we show how single object tracking can be accomplished by PSO. We present the fitness function as well as the re-diversification of the swarm to cover the object state changes between the consecutive images.

4.3.1 Multi-patch Based Object Tracking Using Region Covariance

The fitness function is based on the region covariance matrix (RC). The object is represented by an image template consisting in several non-overlapping image patches. For every pixel i in such a patch of size $M \times N$, we calculate a feature vector b_i

$$b_i = (x \quad y \quad R \quad G \quad B \quad I_x \quad I_y)^T \tag{4.3}$$

where x, y represent the Cartesian coordinates of pixel i, whereas R, G, B stand for color components, and I_x, I_y are image derivatives. The RC descriptor is given by:

$$C = \frac{1}{MN-1} \sum_{i=1}^{MN} (b_i - \overline{b})(b_i - \overline{b})^T \tag{4.4}$$

where $\overline{b}$ denotes the vector of means of the corresponding features for the pixels in the template. The region covariance descriptor has many advantages. In particular, RC indicates both spatial and statistical properties of the objects, it allows combining multiple modalities and features, and last but not least, it is capable of relating regions of different sizes. This descriptor is also robust to the variations in illumination conditions, pose, and view. Although the covariance matrices are positive semi-definite in general, in practice they should be regularized by adding a small constant multiple of the identity matrix, making them strictly positive.

In [5], a Log-Euclidean Riemannian metric has been introduced to obtain statistics on symmetric positive definite matrices. The Singular Value Decomposition (SVD) of a symmetric matrix A of size $n \times n$ is $U \Sigma U^T$, where U is an orthonormal matrix, and $\Sigma = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ is diagonal matrix with nonnegative eigenvalues. The matrix exponential $\exp(A)$ of a symmetric matrix is given by: $\exp(A) = U \cdot \mathrm{diag}(\exp(\lambda_1), \ldots, \exp(\lambda_n)) \cdot U^T$; conversely, the matrix logarithm of a symmetric positive definite matrix is calculated according to: $\log(A) = U \cdot \mathrm{diag}(\log(\lambda_1), \ldots, \log(\lambda_n)) \cdot U^T$. Each symmetric matrix is associated to a tensor by the exponential, conversely, a tensor has a unique symmetric matrix logarithm.

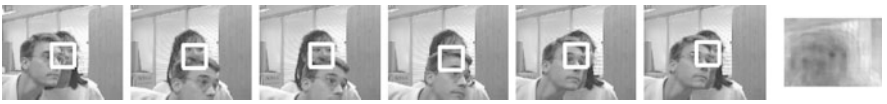


Fig. 4.1 PSO based tracking using multi-patch object representation. Frames #431, 441, 453, 455, 460, 461, and the probability image of the target in frame #431

The distance between two symmetric positive definite matrices X and Y under the Log-Euclidean Riemannian metric can be expressed as follows:

$$\text{dist}(X, Y) = \left\| \log(X) - \log(Y) \right\|_2. \tag{4.5}$$

The Riemannian mean of several elements is an arithmetic mean of matrix elements. Using the Log-Euclidean metric, the algorithm [25] for the incremental subspace update can be employed directly.

In object tracking, we should seek in each frame a location for which the covariance matrix within the object template is most similar to the covariance matrix of the model template. Hence, we should find an object location x^* for which the distance $\text{dist}(\cdot, \cdot)$ between the corresponding covariance matrix X and model covariance matrix $\overline{X}$ assumes the minimal value, i.e., we have to minimize

$$x^* = \arg\min_x \text{dist}(X_x, \overline{X}). \tag{4.6}$$

This is a nonlinear optimization problem that is solved using the PSO algorithm, which in each frame seeks for the best match.

Figure 4.1 depicts some tracking results that were obtained using the multi-patch object representation and a PSO consisting of 10 particles and executing 10 iterations. The tracking of a woman's face was done on color images of size 128×96.[1] We employed both horizontal and vertical patches. The horizontal patches were constructed through dividing vertically the object template into two adjoining patches. Then such patches were divided into 10 horizontally oriented patches, in fives in each of the two vertically oriented patches. The vertical patches were created analogously. The right most image depicts the probability image of the target in frame #431. The detection of outliers is achieved through sorting the scores of the patches and then omitting the poorest ones. The fitness function $f_g(x)$ is the average of K such best matches between the patches of the template at the location x^* and the corresponding patches of the model template.

A tracking algorithm built on the covariance score and with multi-patch object representation can recover after substantial temporal occlusions or large movements. Figure 4.2 illustrates some tracking results that were obtained on the image sequence 'S2L1_View_1' from PETS 2009 database [12], see also Fig. 4.3. As we can observe, the walking woman is successfully tracked despite considerable and multiple temporal occlusions with the static road sign and the pedestrians.

[1] Sequence obtained from http://robotics.stanford.edu/birch/headtracker.

Fig. 4.2 Sub-images with object undergoing tracking in frames #129, 130, 131, 149, 150, 151, 152, 153, 154, 155

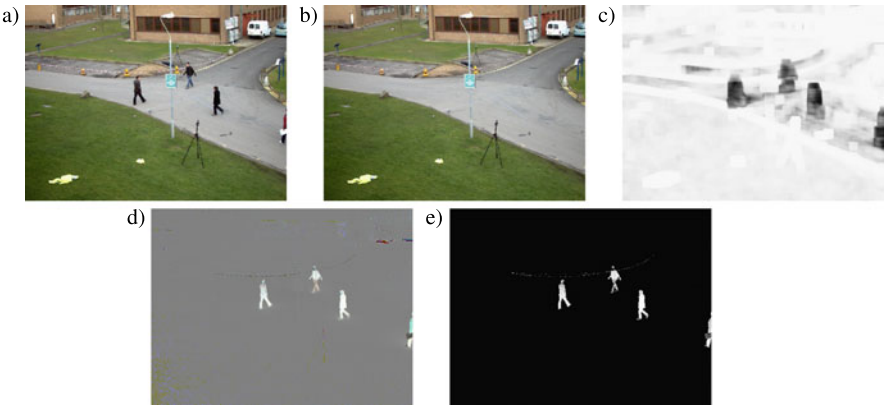


Fig. 4.3 Input image (**a**), reference image (**b**), NCC-based probability image between the reference image and the input image (**c**), color ratios between reference and current image (**d**), and image foreground (**e**)

4.3.2 Foreground Prior

In multiple object tracking, the targets usually become completely or partially occluded. This results in the lack of evidence consisting in non-observability of an occluded target in the image data. In PETS 2009 datasets, some occlusions by the road sign (see images in Fig. 4.2) are relatively long-lasting. As a consequence, the above presented tracker was unable to successfully track some targets in the whole time span, i.e., from entering the scene until exiting the tracking area. Moreover, in a few cases, after loosing the target, the tracker concentrated by mistake on some background areas. In order to cope with such undesirable effects and to decrease the probability of concentrating of the tracker on some non-target areas, we extended the feature vector b_i by a term expressing the object prior. The seventh element of the extended feature vector expresses the object probability which is determined by a foreground segmentation algorithm.

4.3.3 Foreground Segmentation

Our foreground segmentation algorithm is based on a color reference image, which is foreground-free and is extracted automatically in advance, given a sequence of

images with moving targets. Afterwards we employ both region and pixel cues which handle the illumination variations. In addition, we accommodate online the reference image against the illumination and scene changes. The reference image is extracted on the basis of the median of pixel values in some temporal widow. For the 'S2L1_View_1' sequence, the number of images that were needed to extract the foreground free images was equal to 40. Figure 4.3(b) depicts the reference image which was extracted using pixel intensities and the above mentioned number of images.

The normalized cross-correlation *NCC* was used to extract brightness and contrast invariant similarity between the reference image and the current image. It was computed very efficiently using integral images. The *NCC* was used to generate the probability images between the reference images and the current image, see Fig. 4.3(c).

We construct an image of color ratios between the reference image and the current image, where the value of each pixel at location x_1 is given by [4]:

$$\left[\arctan\left(\frac{R^c_{x1}}{R^r_{x1}}\right) \quad \arctan\left(\frac{G^c_{x1}}{G^r_{x1}}\right) \quad \arctan\left(\frac{B^c_{x1}}{B^r_{x1}}\right)\right]^T \tag{4.7}$$

where c and r denote the current and reference image, respectively, whereas R, G, B stand for color components of the *RGB* color space. Such color ratios are independent of the illumination, change in viewpoint, and object geometry. Figure 4.3(d) depicts an example image of color ratios. We can observe that for the pixels belonging to the background the color assumes gray values. This happens because the color channels in the *RGB* color space are highly correlated. Moreover, the color ratios are far smaller in comparison to the ratios between foreground and background. However, as we might observe in the color ratio image there are noisy pixels. The majority of such noisy pixels can be excluded from the image using the probability images, extracted by the normalized cross-correlation.

In our algorithm, we compute online the reference image using the running median. Afterwards, given such an image, we compute the difference image. The difference image is then employed in a simple rule-based classifier, which extracts the foreground objects and shadowed areas. In the classifier, we utilize also the probability image extracted via normalized cross-correlation, as well as the color ratios. The classifier makes decision if pixel is from the background, shadow, or foreground. For shadowed pixels the normalized cross-correlation assumes values near to one. The output of the classifier is the enhanced object probability image. Optionally, in the final stage, we employ the graph-cut optimization algorithm [7] in order to fill small holes in the foreground objects.

4.3.4 Re-diversification of the Swarm

At the beginning of each frame, in some surrounding of the swarm's best location g_t the algorithm selects possible object candidates. Such object candidates are delineated using the foreground blobs. A simple heuristics, which is based on blob

Fig. 4.4 Sub-images with object being tracked in frames #106, 107, 109, 112, 130, 140, and the binary sub-image in frame #112

areas and height-to-width ratios in connection to location of the object at the ground plane, is carried out to select the object candidates. For the videos that were recorded using the calibrated cameras, we project the person locations on the ground onto 3D world coordinates. Such 3D person's location is calculated on the basis of the center of the bottom edge belonging to the bounding box of the blob. Then we employed such information, together with the projected blob sizes, to enhance the delineation of the target candidates as well as to determine the occlusions and splits of the blobs representing the pedestrians into multiple blobs. Afterwards, the particles are initially placed in the gravity centers of the object candidates selected in such a way. The positions of the remaining particles of the swarm are initialized on the basis of the normal distribution which is concentrated around the state estimate at time $t-1$:

$$x_t^{(i)} \leftarrow \mathcal{N}(g_{t-1}, \Sigma) \tag{4.8}$$

where g_{t-1} denotes the location of the best particle that was determined in the previous frame at time $t-1$ and Σ denotes the covariance matrix of the Gaussian distribution whose diagonal elements are proportional to the predicted velocity $v_t = g_{t-1} - g_{t-2}$.

In Fig. 4.4, we can observe the behavior of the tracker with such a swarm re-diversification. As one can notice, the tracking temporally failed in frame #109. Thanks to placing the particles at both candidate objects (see the rightmost image on Fig. 4.4), the tracker correctly recovered the identity of the person in frame #112. It is worth noting that, owing to the object prior in the covariance matrix, the bounding box was placed on the person undergoing tracking and not on the background areas, see frame #109. In order to enhance the object candidate selection, we employed also a person detector [11]. Overall, the person detector found 4550 objects in the 'S2L1_ View_1' dataset. To further enhance the re-diversification of the swarm, the particles were initially placed on the locations determined by the person detector.

4.4 Multiple Object Tracking

The ordinary PSO is not well suited to achieve multiple object tracking. One possible approach to tackle such a problem might be to utilize a PSO that is built on highly discriminative appearance models among different targets, for instance, like those in [10], together with an association framework to achieve better maintaining the identities over time. However, in practice, complex interactions between targets

often lead to difficulties in resolving ambiguities between them. In general, it is relatively easy to track the distinctive objects, but it is much more difficult to achieve reliable tracking when occlusion happens, particularly when the targets have similar appearances. Another approach to this problem might be to represent the positions of feature points by individual particles and to track them using spatial constraints like the maximum distance between feature points together with the maximum distance to the best particle, as it was done in the seminal work [19] (that introduced the PSO for object tracking), and then to select the reliable trajectories on the basis of forward–backward errors [15]. Taking into account the high effectiveness of the PSO when seeking in high-dimensional spaces the problem of multi-object tracking might be formulated as optimization of an energy function, for instance, like those in [3], and estimating the joint state. Recently, the power of the PSO has been fully exploited in multi-object tracking [30], where species-based trackers are employed and each of them tracks one object. In the approach mentioned above, the object interactions are modeled as species competition and repulsion. The occlusion is implicitly inferred using the power of each species and the image observations. Our approach to multiple object tracking is also based on multiple particle swarms. Each object is tracked by a separate swarm. Given the initial tracklets that were determined by the swarms, the refinement of the object's trajectories is done by a PSO-based optimization algorithm. In contrast to [2], which starts an optimization of the energy function from relatively good initial object trajectories and then maintains the identities through the global optimization, in our approach a local optimization takes place in a moving time window. The initial tracklets, which are determined by the swarms, are further distilled in the PSO-based optimization stage that in turn resolves between-object interactions. In the energy function, all target locations belonging to the current time window are considered.

4.4.1 Multiple Object Tracking by Multiple Particle Swarms

In the first phase, the targets are tracked individually. The between-object interactions are initially determined on the basis of our foreground extraction algorithm and a blob analysis. Given the location of a blob in the image as well as the size of its bounding box in relation to the area of the connected component, we decide if a blob represents a single target. In general, a single blob may include multiple objects, while one object may split into multiple blobs. In case of occlusions, two or more swarms responsible for tracking different objects compete for the same target or cluster at the same location. After the end of the occlusion, the swarms should recognize the object identities and continue tracking the objects.

Assuming that in the considered test sequences the people walk on a known ground plane, the location of a candidate target on the ground plane is utilized in evaluation of the expected object area as well as its height and width. This information helps us to decide if the considered target is occluded or if eventually the considered blob is fragmented into several blobs. During the decision making process,

we examine also the distance between the edges of the corresponding rectangles that model the locations and sizes of the objects. Two or more objects are considered as possibly occluded if the distance between the closest edges of the boxes is below a threshold, which in turn depends on the location of the objects on the ground plane. The larger the distance of the object from the camera, the smaller the threshold. At this stage we take into account the distance between the locations of the global best particles in the previous frame, too. The information about the matching of individual patches composing the object templates with the reference templates is considered in the decision process mentioned above and helps us to decide which object or objects are occluded and which are occluding. The search space of the particle swarm with the smaller fitness value is gradually expanded to allow the recovery of the target after occlusion. In scenes with a layout like a corridor with a long vertical passage, with many pairs of pedestrians, etc., where a probability of long term occlusion and the lack of evidence in a longer period of time is considerable, we extract the targets that are close to each other and have similar motion directions. In case of such long term occlusions, we estimate the location (motion) of the occluded object on the basis of the location of the occluder.

As we already mentioned, at this stage the targets are tracked individually. A swarm responsible for tracking a single person is created at the moment of entering the tracked area. The swarm finishes the tracking if the person leaves the tracking scene. Such a scenario greatly simplifies the resolving of interactions, as in each time instant we known the number of the targets. In the presented approach, the position of the target is always defined.

The object tracking is done using the algorithm discussed in Sect. 4.3. In contrast to a typical approach for object tracking, where a model of the object appearance is accommodated over time, in our approach we maintain a pool of models expressing the object appearance at various poses or in different camera views. The object location is determined on the basis of the most similar object model from such a pool of the object models. Each target maintains a constant number of the models in the pool. If the target is not occluded, i.e., the area of the blob as well as the size of the surrounding blob is consistent with the location of the target on the ground-plane, the person detector successfully sought a person in the proximity of the considered person location, the value of the objective function is above an assumed threshold, we replace the pre-selected in advance model by a model determined at the best object location. At the end of the occlusion, or optionally when a target leaves the pre-specified area surrounding the road sign in the 'S2L1_View_1' sequence, we perform the object back-tracking using the above-mentioned pool of the object models. If the back-tracker arrives to a different object, on the basis of the pool of the object model we calculate the sum of the fitness values on both trajectories and choose the trajectory with better fitness. The size of the template modeling the object is determined with regard to its location on the ground-plane.

4.4.2 Refinement of the Tracklets by Particle Swarm Optimization

Particle swarm optimization demonstrated to be an efficient global search method for nonlinear complex systems without a priori knowledge about the system structure. Here, we employ its potential in the optimization of the complex energy function which represents the presence, movement, and interaction of all targets in a sequence of last frames within a temporal window. If the calibration data are available, the tracking is done in the world coordinates. This means that object locations at the ground-plane that were determined by individual trackers are projected to 3D.

Our energy function consists of three terms expressing the pedestrian presence, priors for the pedestrian motion, and mutual exclusion:

$$E(X) = \alpha E_l + \beta E_v + \gamma E_c. \tag{4.9}$$

The vector X consists of the ground-plane coordinates of all targets being in the scene from current time t to time $t - T$. This means that the energy is minimized in a temporal window comprising the last T frames.

The energy should be smaller for the trajectories going around regions of high pedestrian likelihood. Thus, the term expressing the pedestrian presence is given by:

$$E_l(X) = -\sum_{\tau=t}^{t-T}\sum_{id=1}^{P}\exp\left(-\sigma_l^2\sum_{h=1}^{H(t)}\left\|x_\tau^{(id)} - d_\tau^{(h)}\right\|^2\right) \tag{4.10}$$

where t stands for the current time, P is the number of the targets, whereas $H(t)$ denotes the number of the detections in frame τ, and the $d_\tau^{(h)}$ is the location of the detection h in frame τ. The term expressing the motion of the target favors movement with a constant velocity:

$$E_v(X) = \sum_{\tau=t}^{t-T}\sum_{id=1}^{P}\left\|v_\tau^{(id)} - v_{\tau-1}^{(id)}\right\|^2. \tag{4.11}$$

The term expressing the mutual exclusion should penalize the trajectory configurations if two targets approach each other. It assumes the following form:

$$E_c(X) = \sum_{\tau=t}^{t-T}\sum_{id_i\neq id_j}\frac{s_c}{\|x_\tau^{(id_i)} - x_\tau^{(id_j)}\|^2} \tag{4.12}$$

where s_c is a scale factor.

The deterministic optimization algorithms like gradient descent converge rapidly, but may get stuck in local minima of multimodal functions. In the vicinity of a local optimum, the deterministic algorithms converge faster than stochastic search algorithms because stochastic search algorithms waste computational time doing a random search. On the other hand, the PSO may avoid becoming trapped in local optima and find the global optimum. Therefore, in our algorithm the energy function

is first optimized by a PSO and then by a conjugate gradient algorithm [26]. The search area of the PSO is sufficiently large to cover promising configurations. In the PSO, we employ 40 particles, and the maximum number of the iterations is set to 300. The locations determined by the individual person trackers are employed to initialize the PSO, whereas the output of the PSO is used as starting trajectory of the conjugate gradient optimization algorithm which is responsible for the final refinement of the trajectories. Thus, the particle swarm algorithm is utilized to seek good local minima and the conjugate gradient is used to find the local minimum accurately. The optimization is done using person coordinates and velocities from a sequence of the last frames. Thus, the state vector X consists of the person locations determined in the current frame by individual trackers and the refined locations of all persons in a sequence of the last frames.

We achieved considerable improvement of the results by running the optimization on only last 20 frames. For each person entering the tracking area, the optimization starts in the seventh frame. In the eight frame, the optimization algorithm runs on the current locations determined by individual trackers and the refined locations from frames #2–7, etc. Substantial improvement of the tracking accuracy was observed in scenarios with considerable temporal occlusions. In such scenarios, the blobs representing the pedestrians are frequently fragmented, the trackers temporally loose the tracks, making uncoordinated jumps from one object to another. Owing to the energy optimization which considers the interactions of all targets in a sequence of the last frames, the trajectories are far smoother, and most importantly, they pass through regions of high pedestrian likelihood.

4.5 Experiments

The algorithm was evaluated on two publicly available video sequences. The performance of our PSO-based algorithm for multi-object tracking was compared with the performance of the available PSO-based algorithm [30] for tracking multiple objects. In this recently proposed algorithm, species-based trackers are employed and each tracking one object. The object interactions are modeled as species competition and repulsion, whereas the occlusion is implicitly inferred using the power of each species and the image observations. The discussed method has been evaluated on a video sequence from the PETS 2004 database which is an open database for research on visual surveillance, available at http://homepages.inf.ed.ac.uk/rbf/CAVIAR/. The tracking performance of our algorithm was compared with the performance of the algorithm mentioned above on an image sequence that is called 'ThreePastShop2cor', which consists of color RGB images of size 384×288, recorded with 25 frames per second. Figure 4.5 depicts some key frames, where three pedestrians are tracked through occlusion. All three persons were correctly tracked in 108 frames. Thanks to patch-based representation of the object template, the algorithm is able to select the occluding object.

The algorithm was compared with state-of-the-art algorithms for multi-object tracking by analyses carried out both through qualitative visual evaluations as well

Fig. 4.5 Tracking three persons undergoing occlusions. Frames #422, 455, 465, 480, 488, 518

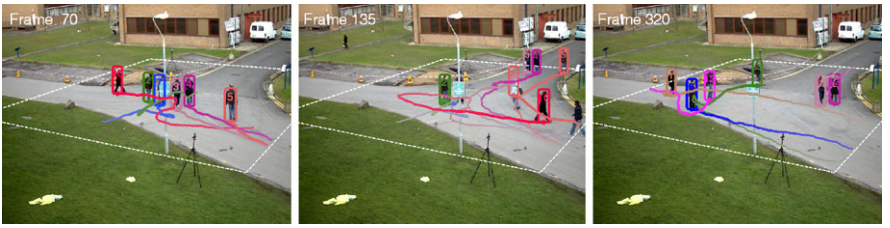


Fig. 4.6 Tracking results on the PETS 2009 *S2L*1_*View*_1 dataset with trajectory refinement using PSO. Frames #70, 130, 320

Table 4.1 Quantitative comparison of our method with state-of-the-art methods on the *S2L*1_*View*_1 sequence from PETS 2009 data set

Metric	[6]	[3]	[2]	Current work
MOTA	79.0 %	81.4 %	88.3 %	90.4 %
MOTP	59.0 %	76.1 %	75.7 %	85.2 %
MT	–	82.6 %	87.0 %	91.3 %
ML	–	0.0 %	4.4 %	4.4 %

as quantitatively using the latest VS-PETS benchmark from 2009 [12]. The experiments were carried out on the sequence 'S2L1_View_ 1', which was recorded at 7 frames per second and contains 795 color images of size 768×576.

The algorithm was evaluated using *CLEAR* metrics [27]. The Multi-Object Tracking Accuracy (*MOTA*) counts all missed targets, false positives, and identity mismatches. It is normalized to tracking all targets such that 100 % means no errors. The Multi-Object Tracking Precision (*MOTP*) expresses the normalized distance between the ground truth location and the estimated location. Mostly Tracked (*MT*) accounts for the percentage of ground-truth trajectories that are covered by the tracker for more than 80 % in length, whereas Mostly Lost (*ML*) is the percentage of the ground-truth trajectories that are covered by the tracker for less than 20 % in length [21]. Table 4.1 illustrates the accuracy and precision, as well as the number of mostly tracked and mostly lost trajectories. The accuracy is a bit higher than 90 %. When no optimization was used, the accuracy was somewhat below 75 %. The percentage of mostly tracked trajectories is nearly 4.5 % higher in comparison to the best reported results.

Figure 4.6 depicts some tracking results. It also shows ground-plane trajectories. As we can observe, the trajectories are far longer in comparison to trajectories that

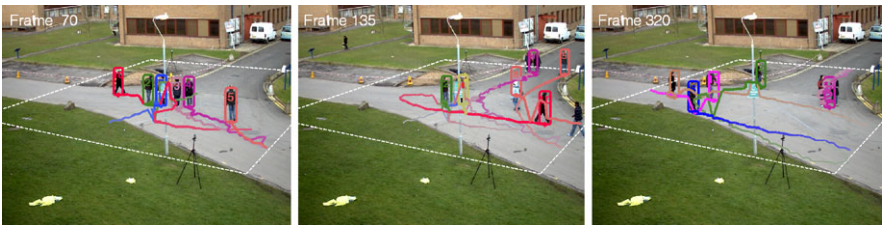


Fig. 4.7 Tracking results on the PETS 2009 *S2L1_View_1* dataset. Frames #70, 130, 330

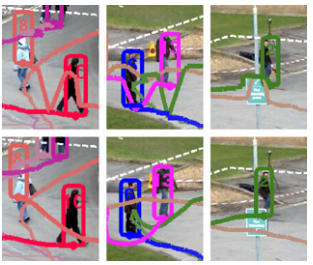


Fig. 4.8 The trajectories without optimization (*top row*) and with the optimization (*bottom row*). Sub-images from frames #135, 320, and 320

are depicted on relevant images in [2]. In almost 40 occlusions like those in frames 129–131 of Fig. 4.2, where the targets undergo temporal occlusion and then split into separate blobs, or the target is occluded by the road sign like in frames 106–112 of Fig. 4.4, the algorithm properly recognized the identities of the targets, avoided clustering on a single target, despite some temporal errors in location or identity estimation.

Figure 4.7 illustrates some tracking results that were obtained using only individual tracking. As we can observe, the trajectories are not so smooth in comparison to the trajectories obtained through the optimization. In particular, one can observe considerable jitters in the trajectories as a result of temporal switches of the identities, see, for instance, a jump close to the road sign in frame #70 of Fig. 4.7.

Our results demonstrate that in multi-object tracking, considerable improvement of the tracking accuracy can be obtained through the use of an optimization algorithm for the refinement of the results obtained by individual trackers, even if they are built on highly discriminative appearance models among different targets. Through formulating an energy function that operates on all targets that are present in a sequence of last frames within a temporal window, and thus takes into account all interactions between them, it is possible to considerably refine the trajectories obtained by individual trackers, see Fig. 4.8.

In our algorithm, in contrast to [3], the joint state is optimized only in some moving temporal widow, which moves forward as the time elapses. The state vector consists of the states determined by the individual trackers in the current frame and the states that were progressively refined in previous frames. In contrast to the algorithm mentioned above, no sophisticated initialization of the optimization algorithm in the form of the pre-calculated trajectories by an Extended Kalman Filter (EKF) or globally optimal discrete tracker based on linear programming [1] is needed. We

also demonstrated that the PSO algorithm is an effective tool for solving such nonlinear and nonconvex energy functions. Since the PSO does not rely on any gradient information, smoothness, or continuity properties, it is possible to employ in the objective functions the terms that employ information, for instance, about the nearest neighbors, identity switches, etc. The PSO-algorithm has also demonstrated great usefulness in single object tracking where swarms consisting of 20 particles and in 10 iterations are able to follow objects, even in case of considerable temporal occlusions. The discussed algorithms were implemented in MATLAB/C.

4.6 Conclusions

We demonstrated that in multi-object tracking, considerable improvement of the tracking accuracy can be obtained through the use of an optimization algorithm for the refinement of the results obtained by individual trackers, even if they are built on highly discriminative appearance models. In the presented algorithm, the joint state is optimized in some moving temporal widow. The state vector consists of the states determined by the individual trackers in the current frame and the states that were progressively refined in the previous frames. We demonstrated that the particle swarm optimization is an effective tool for solving such nonlinear and nonconvex energy functions. Individual object tracking was considered as a numerical optimization problem, where a particle swarm optimization was utilized in searching for the best local mode of the similarity measure.

References

1. Andriyenko, A., Schindler, K.: Globally optimal multi-target tracking on a hexagonal lattice. In: Proc. of the 11th European Conf. on Computer Vision: Part I, pp. 466–479 (2010)
2. Andriyenko, A., Schindler, K.: An analytical formulation of global occlusion reasoning for multi-target tracking. In: IEEE Int. Workshop on Visual Surveillance, pp. 1839–1846 (2011)
3. Andriyenko, A., Schindler, K.: Multi-target tracking by continuous energy minimization. In: IEEE Int. Conf. on CVPR, pp. 1265–1272 (2011)
4. Arsic, D., Lyutskanov, A., Rigoll, G., Kwolek, B.: Multi-camera person tracking applying a graph-cuts based foreground segmentation in a homography framework. In: IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance, pp. 30–37 (2009)
5. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**, 411–421 (2006)
6. Berclaz, E.T.J., Fleuret, F., Fua, P.: Multiple object tracking using k-shortest paths optimization. IEEE Trans. Pattern Anal. Mach. Intell. **33**(9), 1806–1819 (2011)
7. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)
8. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Van Gool, L.J.: Robust tracking-by-detection using a detector confidence particle filter. In: ICCV'09, pp. 1515–1522 (2009)
9. Cai, Y., de Freitas, N., Little, J.J.: Robust visual tracking for multiple targets. In: ECCV, vol. IV, pp. 107–118 (2006)
10. Cheng-Hao, K., Huang, C., Nevatia, R.: Multi-target tracking by on-line learned discriminative appearance models. In: IEEE Int. Conf. on CVPR, pp. 685–692 (2010)

11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Int. Conf. on CVPR, vol. 1, pp. 886–893 (2005)
12. Ferryman, J., Shahrokni, A.: PETS2009: dataset and challenge. In: IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance, pp. 1–6 (2009)
13. Isard, M., Blake, A.: Condensation—conditional density propagation for visual tracking. Int. J. Comput. Vis. **29**, 5–28 (2006)
14. John, V., Trucco, E., Ivekovic, S.: Markerless human articulated tracking using hierarchical particle swarm optimisation. Image Vis. Comput. **28**(11), 1530–1547 (2010)
15. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward–backward error: automatic detection of tracking failures. In: Int. Conf. on Pattern Rec., pp. 2756–2759 (2010)
16. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. of IEEE Int. Conf. on Neural Networks, pp. 1942–1948 (1995)
17. Khan, Z., Balch, T., Dellaert, F.: MCMC-based particle filtering for tracking a variable number of interacting targets. IEEE Trans. Pattern Anal. Mach. Intell. **27**, 1805–1918 (2005)
18. Koelsch, M., Turk, M.: Flocks of features for tracking articulated objects. In: Kisacanin, T.H.B., Pavlovic, V. (eds.) Real-Time Vision for Human–Computer Interaction. Springer, Berlin (2005). Chap. 9
19. Koelsch, M., Turk, M.: Hand tracking with flocks of features. In: IEEE Int. Conf. on CVPR, vol. 2, p. 1187 (2005)
20. Kwolek, B., Krzeszowski, T., Wojciechowski, K.: Real-time multi-view human motion tracking using 3D model and latency tolerant parallel particle swarm optimization. In: 5th Int. Conf. MIRAGE, pp. 169–180. Springer, Berlin (2011)
21. Li, Y., Huang, C., Nevatia, R.: Learning to associate: hybridboosted multi-target tracker for crowded scene. In: IEEE Int. Conf. on CVPR, pp. 2953–2960 (2009)
22. Li, Y., Huang, C., Nevatia, R.: Stable multi-target tracking in real-time surveillance video. In: CVPR, pp. 2953–2960 (2009)
23. Okuma, K., Taleghani, A., De Freitas, N., Little, J.J., Lowe, D.G.: A boosted particle filter: multi-target detection and tracking. In: ECCV, pp. 28–39 (2004)
24. Rasmussen, C., Hager, G.D.: Probabilistic data association methods for tracking complex visual objects. IEEE Trans. Pattern Anal. Mach. Intell. **23**, 560–576 (2001)
25. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. Int. J. Comput. Vis. **77**(1–3), 125–141 (2008)
26. Steihaug, T.: The conjugate gradient method and trust regions in large-scale optimization. SIAM J. Numer. Anal. **20**, 626–637 (1983)
27. Stiefelhagen, R., Bernardin, K., Bowers, R., Garofolo, J.S., Mostefa, D., Soundararajan, P.: The CLEAR 2006 evaluation. In: CLEAR. LNCS, vol. 4122, pp. 1–44. Springer, Berlin (2006)
28. Yang, H., Shao, L., Zheng, F., Wang, L., Song, Z.: Recent advances and trends in visual tracking: a review. Neurocomputing **74**(18), 3823–3831 (2011)
29. Zhang, X., Hu, W., Maybank, S., Li, X., Zhu, M.: Sequential particle swarm optimization for visual tracking. In: IEEE Int. Conf. on CVPR, pp. 1–8 (2008)
30. Zhang, X., Hu, W., Qu, W., Maybank, S.: Multiple object tracking via species-based particle swarm optimization. IEEE Trans. Circuits Syst. Video Technol. **20**(11), 1590–1602 (2010)
31. Zhang, X., Hu, W., Wang, X., Kong, Y., Xie, N., Wang, H., Ling, H., Maybank, S.: A swarm intelligence based searching strategy for articulated 3D human body tracking. In: IEEE Workshop on 3D Information Extraction for Video Analysis and Mining, pp. 45–50. IEEE, New York (2010)

Chapter 5
A Comparative Study of Modified BBO Variants and Other Metaheuristics for Optimal Power Allocation in Wireless Sensor Networks

Ilhem Boussaïd, Amitava Chatterjee, Patrick Siarry, and Mohamed Ahmed-Nacer

Abstract This chapter studies the performance of a wireless sensor network in the context of binary detection of a deterministic signal. The work considers a decentralized organization of spatially distributed sensor nodes, deployed close to the phenomena under monitoring. Each sensor receives a sequence of observations and transmits a summary of its information, over fading channel, to a data gathering node, called fusion center, where a global decision is made. Because of hard energy limitations, the objective is to develop optimal power allocation schemes that minimize the total power spent by the whole sensor network under a desired performance criterion, specified as the detection error probability. The fusion of binary decisions is studied in this chapter by considering two scenarios depending on whether the observations are independent and identically distributed (i.i.d.) or correlated. The present work aims at developing a numerical solution for the optimal power allocation scheme via variations of the biogeography-based optimization algorithm. The proposed algorithms have been tested for several case studies, and their performances are compared with constrained versions of the differential evolution algorithm, the genetic algorithm, and the particle swarm optimization algorithm.

I. Boussaïd (✉) · M. Ahmed-Nacer
University of Science and Technology Houari Boumediene (USTHB), El-Alia BP 32, Bab-Ezzouar, 16111 Algiers, Algeria
e-mail: ilhem.boussaid@univ-paris12.fr

M. Ahmed-Nacer
e-mail: anacer@cerist.dz

A. Chatterjee
Electrical Engineering Department, Jadavpur University, Kolkata, West Bengal 700 032, India
e-mail: cha_ami@yahoo.co.in

P. Siarry
LiSSi (EA 3956), Université de Paris-Est Créteil Val de Marne, 61 avenue du Général de Gaulle, 94010 Créteil, France
e-mail: siarry@univ-paris12.fr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_5,

5.1 Introduction

Wireless Sensor Network (WSN) is a system of spatially distributed sensor nodes with the abilities of sensing, computing, and communicating through wireless channels. Development of WSNs is motivated by many applications such as environment monitoring, security, and detection of remote parameters [2].

In a distributed detection system (also called decentralized detection system) [24], every sensor node performs some preliminary processing of data in a distributed manner and transmits a local decision to central node (called a sink or a fusion center). In turn, the fusion center processes the received data and selects one of a few hypotheses for the final decision-making. The main difference between this approach and the classical centralized decision system is that the fusion center has no access to the raw observation made at each sensor. Evidently, a distributed sensor system is suboptimal compared to a centralized system in which the fusion center has access to the observations from all sensors without distortion. However, the distributed schemes offer the possibility for drastic reductions in communication requirements and energy required to obtain an accurate estimate, at the expense of some performance degradation [26].

Because of strict limitations on resources such as energy, bandwidth, and computational complexity, the standard problem in decentralized detection is to optimize the performance of the system with respect to a desired performance criterion, specified as the detection error probability at the fusion center. The decision rule at the fusion center and the local sensor decision rules need to be jointly designed to optimize the specified performance criterion. So the question is: *How to combine the local sensor observations, within bandwidth and power constraints, while keeping the fusion error probability under a required threshold?*

This chapter aims at developing a numerical solution for the optimal power scheduling in WSN for correlated observations [4]. Three constrained variants of the Biogeography-based Optimization (BBO) algorithm have been proposed to address this issue. They are named as Constrained BBO (CBBO), CBBO-DE, which incorporates the mutation procedure inherited from Differential Evolution (DE) [22] to replace the BBO-based mutation, and 2-Stage-CBBO-DE where the population is updated by applying, alternately from one iteration to the next, the BBO and DE updating methods [5]. Constrained versions of DE, Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) algorithms are also developed in order to compare the result with the three algorithms mentioned above.

The rest of this chapter is organized as follows: Sect. 5.2 provides a formulation of the distributed detection problem [27]. The problem is described considering the special case of binary hypothesis testing problem, where the optimal decision rule is expressed in terms of the Likelihood Ratio (LR) statistic. In Sect. 5.3, the optimal power allocation problem is considered under the assumption of correlated and i.i.d. observations. Section 5.4 briefly describes the conventional BBO algorithm and its constrained variants designed for the problem at hand. The experimental results and detailed performance analysis are given in Sect. 5.5. Finally, the conclusions are presented in Sect. 5.6.

5.2 Problem Statement

The detection problems can usually be cast as binary or M-ary hypothesis testing problems. For example, in a radar, one has to decide whether a target is present or not, based on a noisy return signal that may or may not contain the reflection of a probing pulse. Similarly, in digital communications, over each signaling interval $[0, T]$, a pulse $s(t, X)$ is transmitted which encodes the information about a symbol X taking a discrete set of values. When X takes only two values, this decision gives rise to a binary hypothesis testing problem. But, when X takes $M = 2^k$ values with $k > 1$, the decision problem takes the form of an M-ary hypothesis testing problem [14].

In the standard decentralized problem, a set of dispersed sensor nodes receives information about the state of nature $\mathscr{H}$ (there are M hypothesis on the state of the environment). Based on its observation, sensor node ℓ selects one of D_ℓ possible messages and sends it to the fusion center via a dedicated channel.[1] Based on the received data, the fusion center solves a classical hypothesis testing problem and decides on one of the possible hypotheses [26].

For the simple binary hypothesis testing problem, the objective of the detector is to distinguish between two hypotheses $\mathscr{H}_0$ and $\mathscr{H}_1$ based on the observation of a random vector X. The Bayesian formulation of the binary hypothesis testing problem is based on the philosophy that all uncertainties are quantifiable, and that the costs and benefits of all outcomes can be measured. This means that the hypotheses $\mathscr{H}_0$ and $\mathscr{H}_1$ possess a priori probabilities $\pi_0 = \mathscr{P}(\mathscr{H}_0)$ and $\pi_1 = \mathscr{P}(\mathscr{H}_1)$, and $\pi_0 + \pi_1 = 1$. One possible performance criterion in the Bayes formulation is to find a detector that minimizes the probability of error $\mathscr{P}_e$. The performance of a test will be evaluated in terms of three quantities: the probability of detection $\mathscr{P}_D$, the probability of a miss ($\mathscr{P}_M = 1 - \mathscr{P}_D$), and the probability of false alarm $\mathscr{P}_F$.

Alternatively, one can use the Neyman–Pearson formulation of the binary hypothesis testing problem [17], where only the probability distribution of the observations under each hypothesis is known. The goal is to determine the optimum decision rules at the sensors and the fusion center that minimize the probability $\mathscr{P}_M$ of a miss (or equivalently, maximize the probability of detection $\mathscr{P}_D$) while ensuring that the probability of false alarm $\mathscr{P}_F$ is less than or equal to a number α (upper bound on the probability of false alarm).

For simplicity, we consider the Bayesian formulation based on a known prior probability distribution with the objective to minimize the probability of error at the fusion center rather than Neyman–Pearson formulation which solves the constrained optimization problem that minimizes the probability of a miss (false negative), subject to a constraint on the probability of a false alarm (false positive) [12, 18, 25].

[1] The number of sensor nodes and the number of distinct messages are fixed beforehand. This implicitly limits the amount of data available at the fusion center. The quantity of information provided to the fusion center by a network of L sensors, each sending one of D_ℓ possible messages, does not exceed $\sum_{\ell=1}^{L} \lceil \log_2(D_\ell) \rceil$ bits per channel use [23].

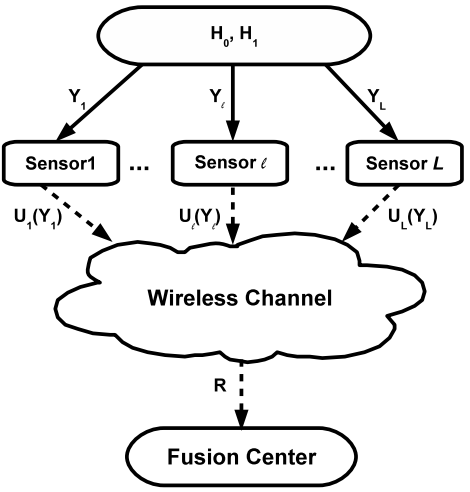


Fig. 5.1 Decentralized binary hypothesis detection problem

This section provides a precise formulation of the decision fusion problem by designing the local and the global decision rules. The local decisions are transmitted over a wireless communication channel prone to attenuation and fading, so that they may not be correctly received at the fusion center. Based on the received data, the fusion center solves a classical hypothesis testing problem and decides on one of the possible hypotheses [26]. However, a question which arises here is: *How to effectively combine, in an optimum way, the information from all sensor nodes in the network to ensure that the final decision is reached with a high degree of reliability?*

Fusion rules with i.i.d. and correlated observations have been investigated and a Bayesian approach has been adopted to address this issue. The approach for decentralized detection, proposed in this chapter, is based on the work of Wimalajeewa et al. [27].

5.2.1 Fusion Problem Formulation

Consider a network with a set $(S_1, \ldots, S_L)$ of L spatially distributed transceiver nodes and a fusion center, as illustrated in Fig. 5.1. Each sensor node observes a single component of the L-dimensional vector $\mathbf{X} = [X_1, \ldots, X_L]^T$, testing two statistical hypotheses, $\mathscr{H}_0$ (no signal) and $\mathscr{H}_1$ (signal present), with the prior probabilities denoted as $\pi_0 = \mathscr{P}(\mathscr{H}_0)$ and $\pi_1 = \mathscr{P}(\mathscr{H}_1)$, respectively.

Let us consider the problem of detecting a constant signal embedded in additive Gaussian noise. The local observation Y_ℓ, obtained at sensor ℓ, under each of the two hypotheses testing, is given by:

$$\begin{aligned} &\mathscr{H}_0 : Y_\ell = V_\ell, \quad \ell = 1, 2, \ldots, L, \\ &\mathscr{H}_1 : Y_\ell = X_\ell + V_\ell, \quad \ell = 1, 2, \ldots, L. \end{aligned} \tag{5.1}$$

The additive observation noise, denoted by V_ℓ, is assumed to be Gaussian with zero mean and variance σ_v^2. The signal to be detected X_ℓ is a known constant (i.e., $X_\ell = m$ for $\ell = 1, 2, \ldots, L$, where m (> 0) indicates the deterministic signal).

We can write the observations in vector form as $\mathbf{Y} = \mathbf{X} + \mathbf{V}$, where $\mathbf{V} = [V_1, V_2, \ldots, V_L]^T$ is a zero-mean Gaussian L-vector of noise samples with covariance matrix $\mathbf{\Sigma}_v$.[2] $\mathbf{X} = [X_1, X_2, \ldots, X_L]^T$ is the observed signal vector, T means transpose. We refer to $\gamma_0 = m^2/\sigma_v^2$ as the local observation signal-to-noise ratio (SNR).

5.2.2 Node Decision Rules

Each node processes its own observation to produce a local decision $U_\ell(Y_\ell)$ and sends it to the fusion sensor. For simplicity, we consider a special class of sensor nodes where each node retransmits an amplified version of its own observation to the fusion center. This class of sensor was shown to perform well when the observations at the sensor nodes are corrupted by additive noise [7]. In this setup, a sensor node acts as an analog relay amplifier with a transmission function given by :

$$U_\ell(Y_\ell) = G_\ell Y_\ell, \quad \ell = 1, 2, \ldots, L, \tag{5.2}$$

where G_ℓ is the amplifier gain at node ℓ.

5.2.3 Transmission of Local Decisions

Each local decision U_ℓ is transmitted over a wireless communication channel which is prone to attenuation and fading. The information R_ℓ reaching the fusion center from the ℓth sensor under each hypothesis $\mathscr{H}_j$, $j = 0, 1$ is given by:

$$\begin{aligned} \mathscr{H}_0 &: R_\ell = N_\ell, \quad \ell = 1, 2, \ldots, L, \\ \mathscr{H}_1 &: R_\ell = H_\ell G_\ell X_\ell + N_\ell, \quad \ell = 1, 2, \ldots, L, \end{aligned} \tag{5.3}$$

where H_ℓ is the channel fading coefficient and N_ℓ is the effective noise vector at the fusion center with mean zero and covariance matrix $\mathbf{\Sigma}_n = H_\ell G_\ell \mathbf{\Sigma}_v G_\ell H_\ell + \mathbf{\Sigma}_w$. Here, $\mathbf{\Sigma}_w$ is the receiver noise covariance. Moreover, $N_\ell = H_\ell G_\ell V_\ell + W_\ell$, W_ℓ is the receiver noise, assumed to be a sequence of i.i.d. Gaussian components with zero mean and variance σ_w^2. In vector notation, we write $\mathbf{R} = \mathbf{AX} + \mathbf{N}$ where $\mathbf{A} = \text{diag}(H_1 G_1, H_2 G_2, \ldots, H_L G_L)$, $\mathbf{R} = [R_1, R_2, \ldots, R_L]^T$ is the received information, $\mathbf{X} = [X_1, X_2, \ldots, X_L]^T$ is the observed signal vector and $\mathbf{N} = [N_1, N_2, \ldots, N_L]^T$ is noise. The noise covariance matrix at the fusion center $\mathbf{\Sigma}_n$ can be rewritten as

$$\mathbf{\Sigma}_n = \mathbf{A}^T \mathbf{\Sigma}_v \mathbf{A} + \Sigma_w. \tag{5.4}$$

[2] In general, $\mathbf{\Sigma}_v$ is not a diagonal matrix unless the observation noise is independent and identically distributed (i.i.d.).

5.2.4 The Decision Fusion Problem

The received observations $\mathbf{R} = [R_1, \ldots, R_L]^T$ at the fusion center are distributed as:

$$\begin{aligned} \mathscr{H}_0 &: \mathbf{R} \sim \mathscr{N}(0, \mathbf{\Sigma}_n), \\ \mathscr{H}_1 &: \mathbf{R} \sim \mathscr{N}(\mathbf{AM}, \mathbf{\Sigma}_n), \end{aligned} \tag{5.5}$$

where $\mathscr{N}$ denotes a Gaussian distribution, $\mathbf{M} = m\mathbf{e}$ and $\mathbf{e}$ is the L-length vector with all ones. The objective of the system is to decide which of the two possible signals is present. The optimal procedure for deciding between the two hypotheses is a threshold rule on the log-likelihood ratio (LLR) of the observation vector [18]. Let us consider the threshold τ (assuming minimum probability of error Bayesian fusion [27]): $\tau = \pi_0/\pi_1$. The LLR $T(\mathbf{R})$ for the detection problem can be written as:

$$T(\mathbf{R}) = m\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{R} - \frac{1}{2}m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}. \tag{5.6}$$

Then the optimum Bayesian decision rule can be rewritten as:

$$\delta(\mathbf{R}) = \begin{cases} 1 & \text{if } T(\mathbf{R}) \geq \ln \tau, \\ 0 & \text{if } T(\mathbf{R}) < \ln \tau. \end{cases} \tag{5.7}$$

The LLR has the following distribution under the two hypotheses:

$$\begin{aligned} \mathscr{H}_0 &: T(\mathbf{R}) \sim \mathscr{N}\left(-\frac{1}{2}m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}, m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}\right), \\ \mathscr{H}_1 &: T(\mathbf{R}) \sim \mathscr{N}\left(\frac{1}{2}m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}, m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}\right). \end{aligned} \tag{5.8}$$

If we further assume that the two hypotheses are equally likely, then the optimal decision threshold at the fusion center is $\tau = 1$. The performance of this threshold test on $T(\mathbf{R})$ is characterized by the probability of the fusion error P_e (i.e., the probability that the fusion center chooses hypothesis $\mathscr{H}_1$ when $\mathscr{H}_0$ is true) which is expressed as

$$\begin{aligned} P_e &= P_F\pi_0 + (1 - P_D)\pi_1 \\ &= \mathscr{Q}\left(\frac{1}{2}\sqrt{m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e}}\right), \end{aligned} \tag{5.9}$$

where π_j is the prior probability of hypothesis $\mathscr{H}_j$, P_F is the false alarm probability of the optimal detector at the fusion center if $j = 0$, P_D is the probability of detection if $j = 1$, and $\mathscr{Q}(\cdot)$ is the complementary Gaussian cumulative distribution function:

$$\mathscr{Q}(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{\frac{-\xi^2}{2}} d\xi. \tag{5.10}$$

5.3 Optimal Power Allocation

Power allocation plays a key role in improving the system performance. In this section, the optimal power allocation among the sensors in the distributed detection system is considered. The objective is to minimize the total power spent by the whole sensor network to achieve a desired detection performance. As it was discussed in the previous paragraphs, the distributed nature of observations coupled with bandwidth and power constraints requires a means of combining local sensor observations while keeping the fusion error as small as possible.

The main question that one seeks to answer is: *What is the optimal power allocation, for a given threshold, at the fusion center?* The problem of finding the optimal power allocation scheme can be posed as follows: *Find a set of sensor gains* $(G_1, \ldots, G_\ell, \ldots, G_L)$ *which solves the following constrained optimization problem*:

$$\begin{cases} \min \quad \sum_{\ell=1}^{L} G_\ell^2 \\ \text{subject to} \quad P_e = Q\left(\frac{1}{2}\sqrt{m^2 \mathbf{e}^T \mathbf{A} \boldsymbol{\Sigma}_n^{-1} \mathbf{A} \mathbf{e}}\right) \leq \varepsilon, \\ \qquad\qquad G_\ell \geq 0, \quad \ell = 1, 2, \ldots, L. \end{cases} \tag{5.11}$$

The objective is to minimize the total power while keeping the fusion error probability under a required threshold ε. We consider two situations: (i) where the local observations are i.i.d., and (ii) where the observations of different nodes are correlated. For both cases, we determine the optimum power allocation schemes that minimize the total power required to satisfy a certain performance level. In situation (i), the optimal solution to the gain allocation is analytically derived. In situation (ii), an approximate analytical solution to the power allocation problem that minimizes the fusion error probability bound in (5.24) is derived for small correlations. It is also shown that, under arbitrary correlated observations, the numerical approach becomes suitable to find the optimal power allocation, since it gets difficult to solve this problem analytically.

5.3.1 Independent Observations

In the special case when local observations and the receiver noise are both i.i.d., $\boldsymbol{\Sigma}_v = \sigma_v^2 \mathbf{I}$ and $\boldsymbol{\Sigma}_w = \sigma_w^2 \mathbf{I}$, where $\mathbf{I}$ is the $L \times L$ identity matrix, the mean-square error (MSE) based on the received signal (5.3) is given by:

$$MSE = \left(\sum_{\ell=1}^{L} \frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2} \right)^{-1}. \tag{5.12}$$

Let us assume that the fusion center is equipped with the knowledge of channel fading coefficients.[3] Then, the probability of the fusion error (5.9) is simplified to:

$$P_e = \mathcal{Q}\left(\frac{m}{2}\sqrt{\sum_{\ell=1}^{L}\frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2}}\right). \tag{5.13}$$

It is interesting to note that the probability of error at the fusion center has a performance floor of $\mathcal{Q}(\frac{\sqrt{L\gamma_0}}{2})$ when G_ℓ^2 tends to infinity ($\ell = 1, 2, \ldots, L$), i.e.,

$$\begin{aligned} &\lim_{G_\ell \to \infty} \sum_{\ell=1}^{L} \frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2} = \frac{L}{\sigma_v^2}, \\ &\lim_{G_\ell \to \infty} P_e = \mathcal{Q}\left(\frac{\sqrt{L\gamma_0}}{2}\right). \end{aligned} \tag{5.14}$$

Intuitively, (5.14) says that, for a fixed L, the performance attained is determined mainly by the observation quality at local sensor nodes regardless of the quality of the wireless channel.

Using the fusion error probability given in (5.13), when the local observations are i.i.d., the first inequality in (5.11) can be expressed as

$$\beta \leq \sqrt{\sum_{\ell=1}^{L}\frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2}} \tag{5.15}$$

where

$$\beta = \frac{2}{m} Q^{-1}(\varepsilon). \tag{5.16}$$

Then, the optimization problem can be stated as:

$$\begin{cases} \min \quad \sum_{\ell=1}^{L} G_\ell^2 \\ \text{subject to} \quad \beta^2 - \sum_{\ell=1}^{L} \frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2} \leq 0, \\ \qquad\qquad G_\ell \geq 0, \quad \ell = 1, 2, \ldots, L. \end{cases} \tag{5.17}$$

The Lagrangian cost function is given by

$$\mathfrak{L}(G, \lambda_0, \mu_k) = \sum_{k=1}^{L} G_k^2 + \lambda_0 \left[\beta^2 - \sum_{k=1}^{L} \frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^{*2} + \sigma_w^2}\right] + \sum_{k=1}^{L} \mu_k(-G_k) \tag{5.18}$$

[3]The assumption that the transmission is idealized, i.e., the information sent from local sensors is assumed to be received intact at the fusion center may be reasonable for some applications, but it may not be realistic for many WSNs where the transmitted information has to endure both channel fading and noise/interference. Acquiring channel state information may be too costly for a resource-constrained sensor network. It may also be impossible to accurately estimate the quality of a fast-changing channel. Hence, it can be argued to be reasonable to assume that this information is available at the fusion center.

where $\lambda_0 \geq 0$ and $\mu_k \geq 0$ for $k = 0, \ldots, L$ are the Lagrange multipliers associated with the inequality constraints.

Given that both the objective function and the constraints are convex, the Karush–Kuhn–Tucker (KKT) conditions are valid [13]. The optimal solution is derived as:

$$G_k^2 = \begin{cases} \frac{\sigma_w^2}{H_k^2 \sigma_v^2}\left[\frac{H_k \sum_{j=1}^{K_1} \frac{1}{H_j}}{(K_1 - \beta^2\sigma_v^2)} - 1\right] & \text{if } k \leq K_1 \text{ and } L > \beta^2\sigma_v^2, \\ 0 & \text{if } k > K_1 \text{ and } L > \beta^2\sigma_v^2, \\ \text{infeasible} & \text{if } L < \beta^2\sigma_v^2 \end{cases} \tag{5.19}$$

where K_1 is found such that $f(K_1) < 1$ and $f(K_1 + 1) \geq 1$ for $1 \leq K_1 \leq L$, $f(k) = \frac{(k - \beta^2\sigma_v^2)}{H_k \sum_{j=1}^{k} \frac{1}{H_j}}$. The proof of the uniqueness of such a K_1 and the global optimality of the solution (5.19) for the optimization problem (5.17) are given in [27].

Statistically, we model the fading coefficients H_ℓ ($\ell = 1, \ldots, L$) as unit mean Rayleigh random variables and, without loss of generality, they are assumed to be ranked in the descending order such that $H_1 \geq H_2 \geq \cdots \geq H_L$.

The solution given in (5.19) is feasible only if $L > \beta^2\sigma_v^2$, i.e., $\gamma_0 > \frac{4}{L}(\mathcal{Q}^{-1}(P_e))^2$, this implies that the probability of error P_e is lower-bounded by $\mathcal{Q}(\frac{\sqrt{L\gamma_0}}{2})$, which is consistent with (5.14).

5.3.2 Correlated Observations

While the popular assumption that the observations at the sensors are independent is convenient for analysis, it does not necessarily hold for arbitrary sensor systems. In practice, it is likely that the sensor observations are spatially correlated leading to a nondiagonal covariance matrix $\mathbf{\Sigma}_v$.

We consider here that the sensor nodes are equally spaced, along a straight line, at a distance d and correlation between observations at node i and j is proportional to $\rho^{d|i-j|}$, where $0 < |\rho| \leq 1$.[4] The observation noise covariance matrix $\mathbf{\Sigma}_v$ can be written as a symmetric Hermitian Toeplitz matrix, referred to as Kac–Murdock–Szegö matrix [11]:

$$\mathbf{\Sigma}_v = \sigma_v^2 \begin{pmatrix} 1 & \rho^d & \ldots & \rho^{d(L-2)} & \rho^{d(L-1)} \\ \rho^d & 1 & \ldots & \rho^{d(L-3)} & \rho^{d(L-2)} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho^{d(L-1)} & \rho^{d(L-2)} & \ldots & \rho^d & 1 \end{pmatrix}. \tag{5.20}$$

[4] Correlation degree $\rho = 1$ means that two observations are perfectly correlated. Correlation degree $0 < \rho < 1$ indicates that two observations are partially correlated (i.e., spatial correlation), while $\rho = 0$ implies that two observations are independent of each other.

The covariance matrix $\mathbf{\Sigma}_v$ of the observation noise is not diagonal. Consequently, it is difficult to evaluate $\mathbf{\Sigma}_n^{-1}$ in closed form in (5.9) for a general $\mathbf{\Sigma}_v$. One answer to this problem is the tridiagonal approximation of $\mathbf{\Sigma}_v$ for sufficiently small ρ, this corresponds to the case where only adjacent node observations are correlated.

Following a similar procedure as in [27], we present the upper bound on the probability of error at the fusion center using the tridiagonal approximation matrix as well as Bergstrom's inequality [1] and finally considering the case where the correlation coefficients are sufficiently small.

According to Bergstrom's inequality, for any positive definite matrices $\mathbf{P}$ and $\mathbf{Q}$:

$$\mathbf{e}^T\mathbf{P}^{-1}\mathbf{e} \geq \frac{(\mathbf{e}^T(\mathbf{P}+\mathbf{Q})^{-1}\mathbf{e})(\mathbf{e}^T\mathbf{Q}^{-1}\mathbf{e})}{\mathbf{e}^T\mathbf{Q}^{-1}\mathbf{e} - \mathbf{e}^T(\mathbf{P}+\mathbf{Q})^{-1}\mathbf{e}}. \tag{5.21}$$

From Eqs. (5.4) and (5.9), $m^2\mathbf{e}^T\mathbf{A}\mathbf{\Sigma}_n^{-1}\mathbf{A}\mathbf{e} = m^2\mathbf{e}^T(\Sigma_v + \sigma_w^2\mathbf{A}^{-2})^{-1}\mathbf{e}$. Let $\mathbf{P} = (\Sigma_v + \sigma_w^2\mathbf{A}^{-2})$ and consider the matrix $\mathbf{Q}$ given by:

$$\mathbf{Q} = \sigma_v^2 \begin{pmatrix} 1 & -\rho^d & \dots & -\rho^{d(L-2)} & -\rho^{d(L-1)} \\ -\rho^d & 1 & \dots & -\rho^{d(L-3)} & -\rho^{d(L-2)} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ -\rho^{d(L-1)} & -\rho^{d(L-2)} & \dots & -\rho^d & 1 \end{pmatrix}. \tag{5.22}$$

From (5.21) it can be shown that

$$\mathbf{e}^T\left(\mathbf{\Sigma}_v + \sigma_w^2\mathbf{A}^{-2}\right)^{-1}\mathbf{e} \geq \left(\frac{1}{\sum_{\ell=1}^{L}\frac{H_\ell^2G_\ell^2}{2\sigma_v^2H_\ell^2G_\ell^2+\sigma_w^2}} - \frac{1}{D}\right)^{-1} \tag{5.23}$$

where $D = \mathbf{e}^T\mathbf{Q}^{-1}\mathbf{e}$. Therefore, from (5.9) and (5.23), the fusion error probability can be bounded from above by:

$$P_e \leq \mathcal{Q}\left(\frac{m}{2}\left(\frac{1}{\sum_{\ell=1}^{L}\frac{H_\ell^2G_\ell^2}{2\sigma_v^2H_\ell^2G_\ell^2+\sigma_w^2}} - \frac{1}{D}\right)^{-\frac{1}{2}}\right). \tag{5.24}$$

When $\rho = 0$, $D = L/\sigma_v^2$, we get

$$\lim_{G_\ell\to\infty}\left(\frac{1}{\sum_{\ell=1}^{L}\frac{H_\ell^2G_\ell^2}{2\sigma_v^2H_\ell^2G_\ell^2+\sigma_w^2}} - \frac{1}{D}\right) = \frac{L}{\sigma_v^2}. \tag{5.25}$$

For the correlated observations, the optimization problem (5.11) can be reformulated to the following equivalent statement:

$$\begin{cases} \min \quad \sum_{\ell=1}^{L} G_\ell^2 \\ \text{subject to} \quad \beta^2 - e^T A\,\Sigma_n^{-1} Ae \leq 0, \\ \qquad\qquad G_\ell \geq 0, \quad \ell = 1, 2, \dots, L. \end{cases} \tag{5.26}$$

Since it is difficult to obtain an analytical closed form of the power allocation problem, as it was explained in Sect. 5.3.1, it is useful to have an analytical approximation of the problem that minimizes the fusion error probability bound (5.24).

To analytically approximate the optimal power allocation problem, the following optimization problem is considered:

$$\begin{cases} \min \quad \sum_{\ell=1}^{L} G_\ell^2 \\ \text{subject to} \quad q - \sum_{\ell=1}^{L} \frac{H_\ell^2 G_\ell^2}{2\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2} \leq 0, \\ \qquad\qquad G_\ell \geq 0, \quad \ell = 1, 2, \ldots, L \end{cases} \tag{5.27}$$

where $q = (\frac{1}{\beta^2} + \frac{1}{D})^{-1}$. Thus, the optimal power allocated to the sensor nodes can be derived following the same procedure as in Sect. 5.3.1. The corresponding optimal solution to the problem (5.27) is given by:

$$G_k^2 = \begin{cases} \frac{\sigma_w^2}{2H_k^2\sigma_v^2}\left[\frac{H_k \sum_{j=1}^{N1} \frac{1}{H_j}}{(N1 - 2\sigma_v^2 q)} - 1\right] & \text{if } k \leq N1 \text{ and } L > 2\sigma_v^2 q, \\ 0 & \text{if } k > N1 \text{ and } L > 2\sigma_v^2 q, \\ \text{infeasible} & \text{if } L < 2\sigma_v^2 q \end{cases} \tag{5.28}$$

where $N1$ is unique and is defined such that $\tilde{f}(N1) < 1$ and $\tilde{f}(N1 + 1) \geq 1$ for $1 \leq N1 \leq L$. $\tilde{f}(k) = \frac{(k - 2\sigma_v^2 q)}{H_k \sum_{j=1}^{k} \frac{1}{H_j}}$.

In the optimal solution, the number of active sensors should be greater than $2\sigma_v^2 q$ in order to satisfy the required fusion error probability at the fusion center. This solution suggests also that some sensors should remain inactive in order to minimize the total power consumption.

Next, we propose a numerical approach to find the optimal power allocation when local observation are arbitrary correlated. The solution proposed in this work is based on the variation of the BBO algorithm.

5.4 Constrained BBO for Optimal Power Allocation

This section details the description of the basic Biogeography-Based Optimization algorithm, adapted from [21], in a nutshell. A brief description of the constrained optimization problem and a review of several popular constraint-handling approaches are presented, followed by a detailed description of the algorithms proposed in this work.

5.4.1 Standard Unconstrained Biogeography-Based Optimization (BBO)

The Biogeography-based optimization (BBO) algorithm, developed by Dan Simon [21], was strongly influenced by the *equilibrium theory* of island biogeography [15]. The basic premise of this theory is that the rate of change in the number of species on

an island depends critically on the balance between the immigration of new species onto the island and the emigration of established species.

The BBO algorithm operates upon a population of individuals called *islands* (or *habitats*). Each island represents a possible solution to the problem in hand. The fitness of each island is determined by its *Habitat Suitability Index* (*HSI*), a metric which determines the goodness of a candidate solution, and each island feature is called a *Suitability Index Variable* (*SIV*). Good solutions may have a larger number of species, which represents an island with a low *HSI*, compared to poor solutions.

The number of species present on the island is determined by a balance between the rate at which the new species arrive and the rate at which the old species become extinct on the island. In BBO, each individual has its own immigration rate (λ) and emigration rate (μ). These parameters are affected by the number of species (S) in an island and are used to probabilistically share information between islands. Islands with smaller populations are more vulnerable to extinction (i.e., the immigration rate is high). But as more species inhabit the island, the immigration rate reduces and the emigration rate increases. In BBO, good solutions (i.e., islands with many species) tend to share their features with poor solutions (i.e., islands with few species), and poor solutions accept a lot of new features from good solutions.

But how might immigration and emigration work on an island? The migration pattern is determined by the immigration rate (λ) at which new species immigrate to the island. The rate of immigration (λ) will decline with the number of species (S) present on the island. The maximum immigration rate (I) occurs when island is empty and decreases as more species are added. Once all potential colonists are on the island, then $S = S_{\max}$ (maximum number of species the island can support) and immigration rate must be equal to zero. The immigration rate, when there are S species in the island, is given by

$$\lambda_S = I\left(1 - \frac{S}{S_{\max}}\right). \tag{5.29}$$

The emigration rate (μ), at which populations of established species emigrate, will increase with the number of species (S). The maximum emigration rate (E) occurs when all possible species are present on the island (when $S = S_{\max}$), and must be zero when no species are present. The emigration rate, when there are S species in the island, is given by

$$\mu_S = E\left(\frac{S}{S_{\max}}\right). \tag{5.30}$$

Figure 5.2 graphically represents the relationships between the number of species (S), emigration rate (μ), and immigration rate (λ). Over time, the countervailing forces of emigration and immigration result in an equilibrium level of species richness. The equilibrium value (S^*) is the point at which the rate of arrival of species (λ) is exactly matched by the rate of emigration (μ). We have assumed that μ and λ are constant linear relationships, but different mathematical models of biogeography that included more complex variables are presented in [15].

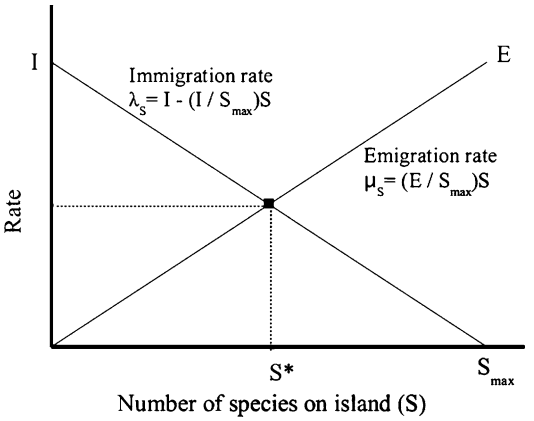


Fig. 5.2 Linear migration model—the relationship of fitness of islands (number of species), emigration rate μ and immigration rate λ

We now consider the probability P_S that the island contains exactly S species. The number of species will change from time t to time $t + \Delta t$ as follows:

$$P_S(t + \Delta t) = P_S(t)(1 - \lambda_S \Delta t - \mu_S \Delta t) + P_{S-1}\lambda_{S-1}\Delta t + P_{S+1}\mu_{S+1}\Delta t, \quad (5.31)$$

which states that the number of species on the island in one time step is based on the total number of current species on the island, the new immigrants, and the number of species which leave during the time period. We assume here that Δt is small enough so that the probability of more than one immigration or emigration can be ignored. In order to have S species at time $t + \Delta t$, one of the following conditions must hold:

- There were S species at time t, and no immigration or emigration occurred between t and $t + \Delta t$;
- One species immigrated onto an island already occupied by $S - 1$ species at time t.
- One species emigrated from an island occupied by $S + 1$ species at time t.

The limit of (5.31) as $\Delta t \to 0$ is given by Eq. (5.32):

$$\dot{P}_S = \begin{cases} -(\lambda_S + \mu_S)P_S + \mu_{S+1}P_{S+1} & \text{if } S = 0, \\ -(\lambda_S + \mu_S)P_S + \lambda_{S-1}P_{S-1} + \mu_{S+1}P_{S+1} & \text{if } 1 \le S \le S_{\max} - 1, \\ -(\lambda_S + \mu_S)P_S + \lambda_{S-1}P_{S-1} & \text{if } S = S_{\max}. \end{cases} \quad (5.32)$$

Equation (5.32) can be arranged into a single matrix form:

$$\begin{bmatrix} \dot{P}_0 \\ \dot{P}_1 \\ \vdots \\ \dot{P}_n \end{bmatrix} = \begin{bmatrix} -(\lambda_0 + \mu_0) & \mu_1 & 0 & \dots & 0 \\ \lambda_0 & -(\lambda_1 + \mu_1) & \mu_2 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \lambda_{n-2} & -(\lambda_{n-1} + \mu_{n-1}) & \mu_n \\ 0 & \dots & 0 & \lambda_{n-1} & -(\lambda_n + \mu_n) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_n \end{bmatrix}. \quad (5.33)$$

For notational brevity, we simply write $n = S_{\max}$.

The BBO algorithm can be overall described with Algorithm 1. The two basic operators which govern the working of BBO are the *migration* and *mutation*.

Migration is used to modify existing islands by mixing features within the population. The immigration rate (λ) and the emigration rate (μ) of each island are used to probabilistically share information between islands, rand(0, 1) is a uniformly distributed random number in the interval [0, 1] and $X_{i,j}$ is the jth SIV of the solution $\mathbf{X}_i$. The BBO migration strategy is similar to the global recombination approach of evolutionary strategies (ES) [3], in which many parents can contribute to a single offspring. The main difference is that recombination is used to create new solutions, while in BBO migration is used to change existing solutions.

An island's HSI can change suddenly due to apparently random events (unusually large flotsam arriving from a neighboring island, disease, natural catastrophes, etc.). BBO models this phenomena as *SIV mutation*, and uses species count probabilities to determine mutation rates. Mutation is used to enhance diversity of the population, thereby preventing the search from stagnating. The likelihood that a given solution S was expected a priori to exist as a solution for the given problem is indicated by the species count probability P_S. If an island S is selected to execute the mutation operation, then a chosen variable *SIV* is randomly modified based on its associated probability P_S. In this context, it should be remarked that very high HSI solutions and very low HSI solutions are both equally improbable. Medium HSI solutions are relatively probable. If a given solution has a low probability, then it is likely to be mutated to some other solution. Conversely, a solution with high probability is less likely to be mutated. The mutation rate $m(S)$ is inversely proportional to the solution probability:

$$m(S) = m_{\max}\left(1 - \frac{P_S}{P_{\max}}\right) \tag{5.34}$$

where $m_{\max}$ is a user-defined parameter, and $P_{\max} = \max_S P_S$, $S = 1, \ldots, S_{\max}$. If an island is selected for mutation, then a randomly chosen SIV in the island is simply replaced with a new randomly generated variable from its range.

On the other hand, elitism (copying some of the fittest individuals to the next generation) is applied.

5.4.2 Constrained Optimization

In many optimization scenarios, inequality constraints and equality constraints may be imposed in addition to the objective functions. The standard form of a constrained optimization problem is formulated as follows:

$$\begin{cases} \text{Find} \quad x \quad \text{which optimizes } f(x) \\ \text{subject to} \quad g_i(x) \leq 0, \quad i = 1, 2, \ldots, p, \\ \qquad\qquad\quad h_j(x) = 0, \quad j = 1, 2, \ldots, q \end{cases} \tag{5.35}$$

Algorithm 1: BBO

```
Initialize a set of solutions (islands) to a problem
while Termination condition not met do
    Evaluate the fitness (HSI) for each solution
    Compute S, λ and μ for each solution
    Migration:
    for i = 1 to N do
        Use λ_i to probabilistically decide whether to immigrate to X_i
        if rand(0, 1) < λ_i then
            for j = 1 to N do
                Select the emigrating island X_j with probability ∝ μ_j
                if rand(0, 1) < μ_j then
                    Replace a randomly selected decision variable (SIV) of X_i
                    with its corresponding variable in X_j
                end
            end
        end
    end
    Mutation:
    for i = 1 to N do
        Compute the probability P_i using λ_i and μ_i
        Use the probability P_i to compute the mutation rate m_i
        for j = 1 to D do
            Select a variable (SIV) X_{i,j} with probability ∝ P_i
            if rand(0, 1) < m_i then
                Replace X_{i,j} with a randomly generated variable from its
                range
            end
        end
    end
    Implement elitism to retain the best solutions in the population from one
    generation to the next.
end
```

where $f(x)$ is an objective function, x denotes the decision solution vector, p is the number of inequality constraints, and q is the number of equality constraints (in both cases, constraints could be linear or nonlinear). Feasible individuals satisfy all constraints while infeasible individuals do not satisfy at least one constraint. Then, a solution candidate x is feasible if and only if $g_i(x) \leq 0\ \forall i = 1, 2, \ldots, p$ and $h_j(x) = 0\ \forall j = 1, 2, \ldots, q$ holds. Obviously, only a feasible individual can be a solution, i.e., an optimum, for a given optimization problem.

A number of approaches have been proposed by incorporating constraint-handling techniques into evolutionary algorithms to solve constrained optimization problems. Comprehensive surveys about such approaches can be found in [8, 16]. Most of the evolutionary constraint handling methods can be broadly classified into five categories [16]: (i) methods based on preserving feasibility of solutions, (ii) methods based on penalty functions, (iii) methods making distinction between feasible and infeasible solutions, (iv) methods based on decoders, and (v) hybrid methods.

The penalty function method is widely regarded as the most popular constraint-handling technique due to its simple principle and ease of implementation. In this approach, the constraints are incorporated into the objective function so that the original constrained problem is transformed into unconstrained one by adding (or subtracting) a penalty term to (or from) the objective function for points not lying in the feasible set and thus violating some of the constraints. This method typically generates a sequence of infeasible points, approaching optimal solutions to the original problem from the outside (exterior) of the feasible set. The general formulation of the exterior penalty approach is:

$$\phi(x) = f(x) \pm \left[\sum_{i=1}^{p} r_i \times \zeta_i + \sum_{j=1}^{q} c_j \times \vartheta_j \right]. \tag{5.36}$$

Here, the constraints are combined with the objective function $f(x)$, resulting in a new (expanded) objective function $\phi(x)$ which is then actually optimized. ζ_i and ϑ_j are functions of the constraints $g_i(x)$ and $h_j(x)$, respectively, and r_i and c_j are positive constants, called "penalty factors".

The general formulation of ζ_i and ϑ_j is:

$$\begin{aligned} \zeta_i &= \left(\max\left\{0, g_i(x)\right\}\right)^{\alpha}, \\ \vartheta_j &= \left|h_j(x)\right|^{\beta} \end{aligned} \tag{5.37}$$

where α and β are normally 1 or 2.

If an inequality constraint is satisfied, then $g_i(x) \leq 0$ and $\max\{0, g_i(x)\}$ will return 0, and therefore that constraint will not contribute anything to the function ϕ. If a constraint is violated, i.e., $g_i(x) > 0$ or $h_j(x) \neq 0$, a large term will get added to ϕ such that the solution is pushed back towards the feasible region.

Another approach for handling constraints would be to consider the objective function and the constraints separately. The constraint handling method described by Deb [9] proposes using a binary tournament selection operator and applies the following rules to compare two individuals:

1. Any feasible solution is preferred to any infeasible solution.
2. Among two feasible solutions, the one having better objective function value is preferred.
3. Among two infeasible solutions, the one having smaller constraint violation is preferred.

Deb's approach does not require a penalty factor because in any of the above three scenarios, solutions are never compared in terms of both objective function value and constraint violation information. Of the three tournament cases mentioned above, in the first case, neither the objective function value nor the constraint violation information is used, simply the feasible solution is preferred. In the second case, solutions are compared in terms of the objective function values alone and, in the third case, solutions are compared in terms of the constraint violation information alone.

5.4.3 Description of the Proposed Algorithms

In this chapter, three constrained versions of BBO algorithm, which are improvements over the BBO algorithm, are proposed to numerically find the optimal power scheduling in WSN. These constrained variants, namely CBBO, CBBO-DE, and 2-Stage-CBBO-DE, implement different strategies for updating the population in subsequent generations, along with some adaptations to solve the constrained optimization problem under consideration. The other steps given in this section are the same for all proposed algorithms.

For the first constrained version, named as CBBO for Constrained BBO, the population is updated using the conventional BBO-based operators [4]. The migration process is followed by the BBO-based mutation, in an iterative fashion. The second variation, named as CBBO-DE, introduces the DE mutation [6] to replace the BBO-based mutation. Unlike CBBO, CBBO-DE first generates new parameter vectors, by using the DE mutation operation, and then the BBO-based migration operator is applied for the resultant mutant vectors. In the 2-Stage-CBBO-DE variant, the integration of DE and BBO is achieved by employing a collaboration that consists of activating each technique in an alternate fashion. The different population update strategies are described in Sect. 5.4.3.6.

5.4.3.1 Solution Representation

A solution for the optimal power allocation problem is encoded by a L-dimensional vector $\mathbf{G} = [G_1, G_2, \ldots, G_L]$. Each decision variable G_ℓ ($\ell = 1, \ldots, L$), which denotes the amplifier gain at sensor node ℓ, is directly coded as a real value. Figure 5.3 illustrates the solution representation.

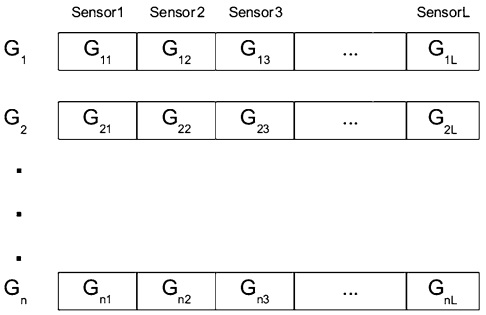


Fig. 5.3 Solution representation for the optimal power allocation problem

5.4.3.2 Objective Function

The objective is to determine the optimum power allocation where optimality is defined as minimizing the power spent by the network to achieve the desired performance at the fusion center. The problem of finding the optimal solution to the gain allocation is posed in Sect. 5.3 for both i.i.d. (Eq. (5.17)) and correlated observations (Eq. (5.27)).

5.4.3.3 Constraint Handling Approach

In the proposed algorithms, the constraints have been introduced into the objective function using a penalty function [8]. The introduction of the penalty term enables us to transform a constrained optimization problem into an unconstrained one (Eq. (5.38)). Only inequality constraints are considered in this work. The new objective function for the minimization problem is formulated as

$$F(\mathbf{G}) = \begin{cases} f(\mathbf{G}) & \text{if } \psi_j(\mathbf{G}) \leq 0, \\ f(\mathbf{G}) + \Phi(\psi^+(\mathbf{G})) & \text{otherwise,} \end{cases} \tag{5.38}$$

where $f(G) = \sum_{\ell=1}^{L} G_\ell^2$, $\psi_1(G) = \beta^2 - \sum_{\ell=1}^{L} \frac{H_\ell^2 G_\ell^2}{\sigma_v^2 H_\ell^2 G_\ell^2 + \sigma_w^2}$ when the observations are i.i.d. and $\psi_1(G) = \beta^2 - e^T A \Sigma_n^{-1} A e$ when the observations are correlated. $\psi_{\ell+1}(G) = -G_\ell$ for $\ell = 1, 2, \ldots, L$, $G = [G_1, G_2, \ldots, G_L]^T$.

In Eq. (5.38), $\Phi(\psi^+(\mathbf{G})) = \sum_{j=1}^{p} (\max\{0, \psi_j^+(\mathbf{G})\})^2$ is the penalty function and $\psi^+(\mathbf{G}) = [\psi_1^+(\mathbf{G}), \psi_2^+(\mathbf{G}), \ldots, \psi_p^+(\mathbf{G})]$ are the constraint violations (i.e., $\psi_i^+(\mathbf{G}) = \max\{0, \psi_j(\mathbf{G})\}$, $i = 1, \ldots, p$) and $p = L + 1$ is the number of inequality constraints.

5.4.3.4 Individual Initialization

A random initial population of NP L-dimensional search variable vectors (or islands) is generated. Since the islands are likely to be changed over different generations, the following notation is adopted for representing the ith island of the

Algorithm 2: Stochastic ranking algorithm

```
I_j = j  ∀j ∈ 1, ..., NP
for i = 1 to N do
    for j = 1 to NP − 1 do
        sample u ∈ rand(0, 1) (uniform random number generator)
        if (Φ(G_{I_j}) = Φ(G_{I_{j+1}}) = 0) or (u < P_f) then
            if (f(G_{I_j}) > f(G_{I_{j+1}})) then
                Swap (I_j, I_{j+1})
            else
                if (Φ(G_{I_j}) > Φ(G_{I_{j+1}})) then
                    Swap (I_j, I_{j+1})
                end
            end
        end
    end
    if no Swap done then
        break
    end
end
```

population at the current generation g: $\mathbf{G}_{i,g} = (G_{i,1,g}, G_{i,2,g}, G_{i,j,g}, \ldots, G_{i,L,g})$, where $i = 1, \ldots, NP$; $j = 1, \ldots, L$, and $G_{i,j,g}$ is the jth SIV of the island $\mathbf{G}_i$ at generation g.

5.4.3.5 Individual Stochastic Ranking

To deal with the constraints, the stochastic ranking algorithm was employed (see Algorithm 2) [20]. Thus, each island in the population is ranked based on both fitness value and constraints violation amount. Migration is based on this ranking to share information between islands. The highest ranking islands have high species emigration rates (μ) and are more likely to share information with other less fit solutions based on their immigration rates (λ).

Here N denotes the number of sweeps going through the whole population, NP stands for the size of the population. P_f is the probability of using only the objective function for comparing islands in the infeasible region of the search space and for

which a value of $0.4 < P_f < 0.5$ was reported as the most appropriate. $f(G_i)$ is the fitness value of the island G_i and $\Phi(G_i)$ its constraints violation degree, such that

$$\Phi(\mathbf{G}_i) = \sum_{j=1}^{p} \left(\max\left\{0, \psi_j(\mathbf{G}_i)\right\}\right)^2 \tag{5.39}$$

where ψ_j denotes the constraint violation for the jth constraint.

5.4.3.6 Population Update Strategy

It has already been mentioned that the proposed algorithms adopt different mechanisms for updating the population. The operating principles of the CBBO, CBBO-DE, and 2-Stage-CBBO-DE updating strategies are described as follows:

1. **CBBO**: In the CBBO algorithm, the population is updated by successively applying the migration procedure followed by the mutation procedure in an iterative fashion, similar to the philosophy employed in original BBO (see Algorithm 1).

2. **CBBO-DE**: This variant incorporates the mutation procedure inherited from DE algorithm [19, 22], to replace the existing mutation procedure in BBO. Unlike CBBO, CBBO-DE first generates new parameter vectors, by using the DE mutation operation, and then the BBO-based migration operator is applied for the resultant mutated vectors [6].

   (a) **DE Mutation**
   The mutation is performed by calculating weighted vector differences between other randomly selected individuals of the same population. A mutation scale factor F is used to control the amplification of the differential variation. The mutation operation constructs, for each population vector $\mathbf{G}_{i,g}$, a mutant vector $\mathbf{V}_{i,g}$.

   Different mutation schemes are suggested by Price et al. [19]. The general convention used to name the different DE variants is *DE/x/y/z*. Here DE stands for differential evolution, x represents a string that denotes the base vector, i.e., the vector being perturbed (whether it is randomly selected or it is the best vector in the population with respect to fitness value and constraint violation), y is the number of difference vectors, considered for perturbation of x, and z denotes the crossover scheme which may be *binomial* or *exponential*. The mutation is performed following the *DE/rand/1/bin*-variant, also known as the classical version of DE, which is the most frequently used mutation strategy. This mutation scheme uses a randomly selected base vector $\mathbf{G}_{r_1,g}$ and only one weighted difference vector $F(\mathbf{G}_{r_2,g} - \mathbf{G}_{r_3,g})$ is used to perturb it:

$$\mathbf{V}_{i,g} = \mathbf{G}_{r_1,g} + F(\mathbf{G}_{r_2,g} - \mathbf{G}_{r_3,g}), \tag{5.40}$$

where the indices r_1, r_2, r_3 are randomly chosen over the interval $[1, NP]$ and should be mutually different from the running index i and F is a real constant scaling factor within the range $[0, 2]$, usually chosen less than 1.

(b) **Migration**
The DE mutation produces new vectors $\mathbf{V}_{i,g}$ $(i = 1, \ldots, NP)$, which are updated by the BBO-based migration operator. The migration operator is the same as what was employed in original BBO, except that it is applied to the newly modified individuals $\mathbf{V}_{i,g}$. This operation produces new population vectors $\mathbf{M}_{i,g}$ as follows:

$$M_{i,j,g} = \begin{cases} V_{k,j,g} & \text{if rand}(0,1) < \lambda_i, \\ V_{i,j,g} & \text{otherwise,} \end{cases} \tag{5.41}$$

where $i = 1, 2, \ldots, N$, $j = 1, \ldots, D$ and $V_{k,j,g}$ is the jth decision variable of a randomly selected individual $\mathbf{V}_{k,g}$ among the transformed population in generation g. $\mathbf{V}_{k,g}$ is selected with a probability based on its emigration rate μ_k and λ_i is the immigration rate of the individual $\mathbf{M}_{i,g}$.

3. **2-Stage-CBBO-DE**
The population update strategy of the 2-Stage-CBBO-DE algorithm is similar to the one described in [5]. The population is updated by applying, alternately from one iteration of the algorithm to the next, the BBO and DE updating methods, as described bellow.

- **BBO updating method**
The BBO updating method consists of applying the migration and the mutation operators. The migration operator reproduces a new population vector $\mathbf{M}_{i,g}$ as follows:

$$M_{i,j,g} = \begin{cases} G_{k,j,g} & \text{if rand}(0,1) < \lambda_i, \\ G_{i,j,g} & \text{otherwise,} \end{cases} \tag{5.42}$$

where $i = 1, 2, \ldots, NP$, $j = 1, \ldots, L$ and $G_{k,j,g}$ is the jth decision variable of a randomly selected individual $\mathbf{G}_{k,g}$. $\mathbf{G}_{k,g}$ is selected with a probability based on μ_k.

The mutation is performed for the whole population by perturbing the newly migrant individuals $M_{i,g}$ as follows:

$$M_{i,j,g} = \begin{cases} \text{rand}(l_j, u_j) & \text{if rand}(0,1) < m(i), \\ M_{i,j,g} & \text{otherwise,} \end{cases} \tag{5.43}$$

where $i = 1, 2, \ldots, NP$, $j = 1, \ldots, L$, $m(i)$ is the mutation rate given by Eq. (5.34) and $\text{rand}(l_j, u_j)$ is a random number (uniformly distributed) between lower and upper bounds l_j and u_j.

- **DE updating method**
DE employs the mutation operation to produce a mutant vector with respect to each individual, the so-called target vector, in the current population. For the

proposed algorithm, the mutation is performed using the *DE/rand/1* mutation strategy as follows:

$$\mathbf{V}_{i,g} = \mathbf{G}_{r_1,g} + F(\mathbf{G}_{r_2,g} - \mathbf{G}_{r_3,g})$$

where $\mathbf{G}_{r_1,g}$, $\mathbf{G}_{r_2,g}$, and $\mathbf{G}_{r_3,g}$ are three individual vectors chosen at random and mutually different.

After the mutation phase, a crossover operation is applied to each pair of the target vector $\mathbf{G}_{i,g}$ and its corresponding mutant vector $\mathbf{V}_{i,g}$ to generate a trial vector $\mathbf{M}_{i,g}$:

$$M_{i,j,g} = \begin{cases} V_{i,j,g} & \text{if rand}(0,1) \leq CR \text{ or } j = j_{\text{rand}}, \\ G_{i,j,g} & \text{otherwise}, \end{cases} \tag{5.44}$$
$$i = 1, 2, \ldots, NP \quad \text{and} \quad j = 1, 2, \ldots, L.$$

The crossover factor CR is randomly taken from the interval $[0, 1]$ and presents the probability of creating parameters for trial vector from a mutant vector. Index j_{rand} is a randomly chosen integer within the range $[1, NP]$. It is responsible for the trial vector containing at least one parameter from the mutant vector. rand$(0, 1)$ is a uniform random number in the range $[0, 1]$.

5.4.3.7 Selection

We adopt the binary tournament selection, described in [9]. Hence, the island $\mathbf{G}_{i,g}$ will be replaced by its newly mutated and migrant island $\mathbf{M}_{i,g}$, to survive as a member for the next generation $g + 1$, under any of the following conditions:

- $\mathbf{G}_{i,g}$ is infeasible, but $\mathbf{M}_{i,g}$ is feasible,
- Both $\mathbf{G}_{i,g}$ and $\mathbf{M}_{i,g}$ are feasible, but $f(\mathbf{M}_{i,g}) < f(\mathbf{G}_{i,g})$,
- Both $\mathbf{G}_{i,g}$ and $\mathbf{M}_{i,g}$ are infeasible, but $\text{Viol}(\mathbf{M}_{i,g}) < \text{Viol}(\mathbf{G}_{i,g})$ (the island having smaller constraint violation is preferred).

In order to impose the same degree of importance to all constraints, the maximum violation value for each constraint in the whole population is used to normalize each violated constraint calculated in (5.45):

$$\text{Viol}(\mathbf{G}_{i,g}) = \sum_{j=1}^{p} \frac{\psi_j^+(\mathbf{G}_{i,g})}{\psi_{\max}(j)} \tag{5.45}$$

where $\psi_{\max}(j)$ is the greatest violation value for the constraint j.

The selection procedure is shown in Algorithm 3.

5.4.3.8 Elitism

To prevent the loss of the best islands, elitism is implemented as follows: the n_{elit} worst islands of the current generation are replaced by the n_{elit} elite islands from the previous generation (n_{elit} is the elitism parameter).

Algorithm 3: Selection

```
Compare G_{i,g} with the corresponding M_{i,g} vector
if (Viol(G_{i,g}) = Viol(M_{i,g}) = 0)) then
    if (f(M_{i,g}) < f(G_{i,g})) then
        G_{i,g+1} = M_{i,g}
    else
        G_{i,g+1} = G_{i,g}
    end
else
    if G_{i,g} is feasible and M_{i,g} is infeasible then
        G_{i,g+1} = G_{i,g}
    else
        if both G_{i,g} and M_{i,g} are infeasible then
            if (Viol(M_{i,g}) < Viol(G_{i,g})) then
                G_{i,g+1} = M_{i,g}
            else
                G_{i,g+1} = G_{i,g}
            end
        end
    end
end
```

5.5 Experimental Results and Analysis

We compared the results of our methods with a constrained version of DE, GA, and PSO algorithms, developed for the purpose of this study. The constraint handling approach is the same as that used in the constrained versions of BBO (Sect. 5.4.3.3).

The constrained DE, called CDE, proceeds exactly as the original algorithm presented in [19], but the selection operator is replaced by the binary tournament selection described in Sect. 5.4.3.7.

An adaptation of the Standard PSO (SPSO 07[5]) to the constrained optimization problem is also implemented. In the initialization phase, the positions and velocities of all individuals are randomly generated. At each iteration, particle i adjusts its position $\mathbf{G}_i$ and velocity $\mathbf{V}_i$ along each dimension ℓ of the search space, based on the best position it has encountered so far in its flight (also called the *personal best*

[5] http://www.particleswarm.info/Programs.html.

"pbest" for the particle) and the best position found by any other particle in its topological neighborhood (*global best "gbest"*). Suppose that $\mathbf{P}_{i,g}$ represents *pbest* of the ith particle at generation g and $\mathbf{G}_{i,g+1}$ represents the newly generated position of the ith particle at generation $g+1$. In the standard PSO, $\mathbf{P}_{i,g+1} = \mathbf{G}_{i,g+1}$ only if $f(\mathbf{G}_{i,g+1}) < f(\mathbf{P}_{i,g})$. While in CPSO, pair-wise solutions are compared based on feasibility rules described in [9]. That is, $\mathbf{P}_{i,g}$ will be replaced by $\mathbf{G}_{i,g+1}$ under any of the following conditions: (i) $\mathbf{P}_{i,g}$ is infeasible, but $\mathbf{G}_{i,g+1}$ is feasible, (ii) both $\mathbf{P}_{i,g}$ and $\mathbf{G}_{i,g+1}$ are feasible, but $f(\mathbf{G}_{i,g+1}) < f(\mathbf{P}_{i,g})$, (iii) both $\mathbf{P}_{i,g}$ and $\mathbf{G}_{i,g+1}$ are infeasible, but $\text{Viol}(\mathbf{G}_{i,g+1}) < \text{Viol}(\mathbf{P}_{i,g})$ where $\text{Viol}(\cdot)$ is the constraint violation value of an infeasible solution, described in Eq. (5.45).The global best *gbest* is updated, in a similar way.

The constrained real-coded GA (CGA) creates new offspring from the members of the population using the genetic operators (crossover and mutation) and places these individuals in a new population. The replacement strategy used takes into consideration both the fitness and the constraint violation, as described in Sect. 5.4.3.7.

5.5.1 Parameter Configuration

A population size of 100 individuals was used for each of the algorithms on each of the test problems, except for the CPSO algorithm, where the population size was set to $(10 + 2 * L^2)$, as for the standard PSO (SPSO 07), where L is the problem dimension (number of sensors).

For the CBBO-DE, 2-Stage-CBBO-DE, and CDE algorithms, the mutation scale factor $F = 0.5$ and the crossover control parameter $CR = 0.9$ were chosen, as recommended in [22]. These algorithms use the *DE/rand/1/bin* mutation schema. For CBBO algorithm, we used the same parameter setting as in [21], the only exception being the population size set at 100 and the mutation applied with a probability of 0.01. For the CGA, the Simulated Binary Crossover Operator (SBX) variant [10] was used with a crossover rate of 0.9. Individuals for producing offspring were chosen using a binary tournament selection strategy after evaluating the fitness value and the constraint violation of each individual in the selection pool (q.v. Section 5.4.3.7). For the mutation, a rate of 0.05 was used.

Each algorithm was run 30 times. The algorithms stopped when the maximum number of evaluations, fixed at 25,000, had been exhausted. For the stochastic ranking algorithm, the recommended parameter settings [20] was retained.

Simulations have been carried out for various values of parameters: fusion error probability (ε), correlation degree (ρ), and number of sensors (L), and the performances of the different algorithms are shown for different combinations: $\rho = \{0, 0.01, 0.1, 0.5\}$. $\rho = 0$ represents the uncorrelated case. The fusion error threshold ε takes its values in $\{0.1, 0.05, 0.01, 0.001\}$. The observation signal-to-noise ratio (SNR) γ_0 was set at 10 dB. The channel fading coefficients H_i followed an exponential distribution (i.e., Rayleigh fading) with a unit mean. Without loss of generality, the channel fading coefficients were ranked in the descending order such that $H_1 \geq H_2 \geq \cdots \geq H_L$.

5.5.2 Numerical Results

The statistical features (best, mean, and standard deviation values) of the best feasible solutions obtained after 30 independent runs for each case study are used to evaluate the performance of the competing algorithms.

Table 5.1 shows a comparison of the performances of the competing algorithms for different values of ε and L in the uncorrelated case ($\rho = 0$). We can see that the CBBO-DE algorithm emerged the best candidate algorithm for $L = 10$ and $L = 50$ sensors in terms of the best "mean" results. For the 20 sensors case, the 2-Stage-CBBO-DE produces the best "mean" results. In terms of the "best" fitness function values, it can be inferred that the CBBO-DE algorithm outperformed the other competing algorithms in the case of a large number of sensors for the different values of error probability at the fusion center. For $L = 20$, better "best" results were found by CPSO in two cases and by CGA in one case.

Table 5.2 shows a comparison of the numerical results of the competing algorithms when the observations are correlated in the case of $L = 10$ sensors for different values of the fusion error probability ε and the degree of correlation ρ. As illustrated in this table, CBBO-DE is found to be the best performing algorithm since it produced better "mean" results in 7 cases out of a total of 12. In terms of the "best" fitness function value, out of these 12 test cases, the CGA algorithm could achieve the best results in 5 cases, while the CBBO-DE, 2-Stage-CBBO-DE, and CPSO algorithms obtained the "best" fitness function value in 2 cases and the CBBO-DE in 3 cases. In the case of small probability of error at the fusion center ($\varepsilon = 0.001$), the CGA algorithm emerged as the best performer for the different values of ρ.

For $L = 20$ sensors, as shown in Table 5.3, CBBO-DE emerged as the best performer in terms of the best "mean" and the "best" fitness function values.

The results for $L = 50$ sensors, in the case of correlated observations, are presented in Table 5.4. The CBBO algorithm has emerged as the best performer since it obtained the best "mean" results in 10 cases out of a total of 12.

From these sets of performance evaluations, it can be generally concluded that, when the observations are correlated (Tables 5.2, 5.3, and 5.4), the performance improvement for the CBBO-DE algorithm, compared to the other competing algorithms, was larger for $L = 10$ and $L = 20$ sensors than for $L = 50$ sensors, where the solution quality of CBBO is superior.

Table 5.5 shows the amplifier gain allocated to each sensor for $L = 10$ sensors. The second column of this table gives the analytical optimal schedule for i.i.d. observations ($\rho = 0$), where 0 means a node should remain inactive in order to provide significant system power savings. These analytical results are obtained using the method of Lagrange multipliers [27]. From these results, one can observe that more power is distributed to sensors with good channel fading coefficients and less power is allocated to sensors with poor channels. Consequently, one can decide whether a sensor is in transmit mode or in silent mode.

Table 5.1 Numerical results when the observations are i.i.d. ($\rho = 0$): $L = \{10, 20, 50\}$ sensors, $\gamma_0 = 10$ dB, and $\varepsilon = \{0.1, 0.05, 0.01, 0.001\}$

ε		CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$L = 10$							
0.1	Mean	3.17935E+00	**3.17263E+00**	3.17271E+00	3.17320E+00	3.21051E+00	3.24232E+00
	Std.	4.57E–03	2.00E–04	2.65E–04	6.51E–04	2.84E–02	1.14E–01
	Best	3.17249E+00	3.17233E+00	3.17239E+00	3.17241E+00	3.17971E+00	**3.17230E+00**
0.05	Mean	5.98759E+00	**5.97219E+00**	5.97222E+00	5.97237E+00	5.99619E+00	6.04440E+00
	Std.	9.55E–03	1.21E–05	3.35E–05	1.06E–04	5.88E–02	1.57E–01
	Best	5.97339E+00	5.97218E+00	5.97218E+00	5.97221E+00	**5.69023E+00**	5.97221E+00
0.01	Mean	1.51470E+01	**1.51303E+01**	1.51303E+01	1.51304E+01	1.51400E+01	1.53088E+01
	Std.	7.95E–03	3.97E–05	2.93E–05	8.73E–05	1.14E–01	2.33E–01
	Best	1.51315E+01	**1.51303E+01**	1.51303E+01	1.51303E+01	1.45632E+01	1.51303E+01
0.001	Mean	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.99942E+01**	4.00000E+01
	Std.	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.09E–02	0.00E+00
	Best	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.98277E+01**	4.00000E+01
$L = 20$							
0.1	Mean	1.94366E+00	1.93989E+00	**1.93780E+00**	1.93974E+00	1.96652E+00	2.30705E+00
	Std.	5.71E–03	8.72E–03	3.51E–03	3.19E–03	1.65E–02	6.06E–01
	Best	1.93454E+00	1.93265E+00	1.93325E+00	1.93487E+00	1.94606E+00	**1.93249E+00**
0.05	Mean	3.65694E+00	3.65123E+00	**3.64836E+00**	3.65344E+00	3.68518E+00	3.88634E+00
	Std.	1.03E–02	9.77E–03	3.00E–03	5.18E–03	1.16E–02	2.44E–01
	Best	3.64379E+00	**3.64179E+00**	3.64406E+00	3.64553E+00	3.66486E+00	3.64469E+00
0.01	Mean	9.12602E+00	9.12339E+00	**9.11030E+00**	9.12452E+00	9.18365E+00	9.24245E+00
	Std.	1.35E–02	1.55E–02	6.17E–03	1.01E–02	3.74E–02	2.11E–01
	Best	9.10347E+00	9.10605E+00	9.10173E+00	9.10742E+00	**9.09650E+00**	9.09706E+00
0.001	Mean	2.16507E+01	2.16480E+01	**2.16406E+01**	2.16622E+01	2.17666E+01	2.25538E+01
	Std.	1.87E–02	2.95E–02	1.37E–02	2.11E–02	4.37E–02	1.27E+00
	Best	2.16205E+01	2.16116E+01	2.16250E+01	2.16340E+01	2.17037E+01	**2.15973E+01**
$L = 50$							
0.1	Mean	9.05946E–01	**8.73121E–01**	1.00626E+00	1.05196E+00	9.40606E–01	1.67135E+00
	Std.	1.30E–02	8.70E–03	3.97E–02	5.69E–02	1.83E–02	1.40E+00
	Best	8.84219E–01	**8.67229E–01**	9.38443E–01	9.68506E–01	9.15533E–01	8.80470E–01
0.05	Mean	1.71838E+00	**1.67661E+00**	1.84009E+00	1.91955E+00	1.77156E+00	3.12401E+00
	Std.	1.38E–02	6.22E–03	3.58E–02	7.73E–02	2.51E–02	1.65E+00
	Best	1.69353E+00	**1.66688E+00**	1.76722E+00	1.79779E+00	1.72290E+00	1.85115E+00
0.01	Mean	4.41536E+00	**4.38484E+00**	4.63694E+00	4.74516E+00	4.53181E+00	5.69718E+00
	Std.	2.80E–02	4.76E–02	6.48E–02	9.81E–02	4.15E–02	1.05E+00
	Best	4.37703E+00	**4.34752E+00**	4.53116E+00	4.58753E+00	4.46010E+00	4.46465E+00
0.001	Mean	1.00421E+01	**1.00120E+01**	1.04841E+01	1.06984E+01	1.03002E+01	1.11533E+01
	Std.	3.72E–02	4.17E–02	1.04E–01	1.11E–01	6.43E–02	1.25E+00
	Best	9.96900E+00	**9.91934E+00**	1.02832E+01	1.05173E+01	1.02028E+01	9.97487E+00

Table 5.2 Comparison of numerical results when the observations are correlated: $\rho = \{0.01, 0.1, 0.5\}$, $L = 10$ sensors, $\gamma_0 = 10$ dB, and $\varepsilon = \{0.1, 0.05, 0.01, 0.001\}$

ε		CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$\rho = 0.01$							
0.1	Mean	3.19130E+00	**3.18336E+00**	3.18356E+00	3.18470E+00	3.21675E+00	3.26267E+00
	Std.	5.02E–03	5.02E–03	3.94E–04	6.74E–04	1.72E–02	1.28E–01
	Best	3.18434E+00	3.18305E+00	3.18307E+00	3.18322E+00	3.18874E+00	**3.18300E+00**
0.05	Mean	6.01119E+00	**5.99738E+00**	5.99740E+00	5.99758E+00	6.00321E+00	6.10566E+00
	Std.	1.06E–02	1.89E–05	4.78E–05	1.40E–04	1.18E–01	2.23E–01
	Best	5.99912E+00	5.99736E+00	5.99758E+00	5.99742E+00	**5.37139E+00**	5.99748E+00
0.01	Mean	1.52741E+01	**1.52553E+01**	1.52553E+01	1.52554E+01	1.53046E+01	1.54702E+01
	Std.	8.98E–03	9.41E–05	4.00E–05	7.17E–05	2.89E–02	3.62E–01
	Best	1.52588E+01	**1.52553E+01**	1.52553E+01	1.52553E+01	1.52680E+01	1.52553E+01
0.001	Mean	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.99859E+01**	4.00000E+01
	Std.	0.00E+00	0.00E+00	0.00E+00	0.00E+00	7.00E–02	0.00E+00
	Best	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.96102E+01**	4.00000E+01
$\rho = 0.1$							
0.1	Mean	3.29369E+00	**3.28340E+00**	3.28385E+00	3.28434E+00	3.31745E+00	3.32138E+00
	Std.	6.64E–03	1.12E–04	2.99E–04	7.08E–04	2.33E–02	8.00E–02
	Best	3.28524E+00	3.28325E+00	3.28329E+00	3.28352E+00	3.29679E+00	**3.28321E+00**
0.05	Mean	6.25118E+00	6.23910E+00	**6.23901E+00**	6.23930E+00	6.27467E+00	6.28582E+00
	Std.	5.71E–03	3.85E–05	1.76E–05	3.13E–04	2.15E–02	5.08E–02
	Best	6.24318E+00	6.23900E+00	**6.23898E+00**	6.23910E+00	6.24694E+00	6.23913E+00
0.01	Mean	1.65806E+01	**1.65620E+01**	1.65624E+01	1.65625E+01	1.65821E+01	1.68063E+01
	Std.	1.03E–02	3.56E–09	1.08E–04	1.99E–04	6.88E–02	4.75E–01
	Best	1.65677E+01	1.65620E+01	1.65623E+01	1.65623E+01	**1.62204E+01**	1.65623E+01
0.001	Mean	4.00000E+01	4.90770E+01	4.00000E+01	4.00000E+01	**3.99607E+01**	4.00000E+01
	Std.	4.77E–07	7.81E–04	0.00E+00	0.00E+00	2.08E–01	0.00E+00
	Best	4.00000E+01	4.90210E+01	4.00000E+01	4.00000E+01	**3.88388E+01**	4.00000E+01
$\rho = 0.5$							
0.1	Mean	3.87091E+00	**3.85830E+00**	3.85885E+00	3.85995E+00	3.90135E+00	3.90053E+00
	Std.	6.49E–03	2.32E–04	4.53E–04	9.29E–04	2.64E–02	4.11E–02
	Best	3.86102E+00	**3.85800E+00**	3.85827E+00	3.85840E+00	3.86301E+00	3.85817E+00
0.05	Mean	8.16317E+00	8.13610E+00	**8.13608E+00**	8.13644E+00	8.18067E+00	8.18731E+00
	Std.	1.81E–02	8.04E–05	9.41E–05	3.28E–04	2.16E–02	9.29E–02
	Best	8.13971E+00	8.13600E+00	**8.13596E+00**	8.13604E+00	8.14816E+00	8.13605E+00
0.01	Mean	3.49787E+01	**3.43490E+01**	3.49497E+01	3.49508E+01	3.50209E+01	3.51627E+01
	Std.	1.89E–02	8.08E–07	4.45E–04	9.72E–04	4.67E–02	1.43E–01
	Best	3.49545E+01	**3.43480E+01**	3.49490E+01	3.49492E+01	3.48397E+01	3.49499E+01
0.001	Mean	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.99640E+01**	4.00000E+01
	Std.	0.00E+00	0.00E+00	0.00E+00	8.26E–07	1.32E–01	0.00E+00
	Best	4.00000E+01	4.00000E+01	4.00000E+01	4.00000E+01	**3.92740E+01**	4.00000E+01

Table 5.3 Comparison of numerical results when the observations are correlated: $\rho = \{0.01, 0.1, 0.5\}$, $L = 20$ sensors, $\gamma_0 = 10$ dB, and $\varepsilon = \{0.1, 0.05, 0.01, 0.001\}$

ε		CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$\rho = 0.01$							
0.1	Mean	1.94926E+00	**1.93960E+00**	1.94247E+00	2.01270E+00	1.96992E+00	2.19232E+00
	Std.	9.26E-03	2.14E–03	2.14E–03	1.51E–02	1.24E–02	2.45E–01
	Best	1.94131E+00	1.93760E+00	1.93938E+00	1.98330E+00	1.95234E+00	**1.93730E+00**
0.05	Mean	3.67103E+00	**3.65590E+00**	3.66186E+00	3.68830E+00	3.70072E+00	3.88189E+00
	Std.	7.16E–03	9.74E–04	3.08E–03	3.05E–02	1.70E–02	2.47E–01
	Best	3.66106E+00	**3.65480E+00**	3.65729E+00	3.65620E+00	3.67640E+00	3.65888E+00
0.01	Mean	9.19023E+00	**9.16070E+00**	9.17459E+00	9.18835E+00	9.22620E+00	9.41124E+00
	Std.	1.18E–02	8.98E–04	4.59E–03	9.41E–03	1.25E–01	4.45E–01
	Best	9.17263E+00	9.15980E+00	9.16595E+00	9.17181E+00	**8.57142E+00**	9.15913E+00
0.001	Mean	2.18956E+01	**2.18420E+01**	2.18784E+01	2.19148E+01	2.20025E+01	2.22904E+01
	Std.	2.17E–02	3.99E–03	9.98E–03	2.30E–02	4.00E–02	8.45E–01
	Best	2.18605E+01	**2.18400E+01**	2.18591E+01	2.18677E+01	2.19378E+01	2.18406E+01
$\rho = 0.1$							
0.1	Mean	2.00287E+00	**1.99050E+00**	1.99530E+00	1.99980E+00	2.02247E+00	2.26079E+00
	Std.	1.03E–02	1.32E–03	2.06E–03	5.18E–03	1.52E–02	2.22E–01
	Best	1.99094E+00	**1.98930E+00**	1.99199E+00	1.99325E+00	1.99851E+00	1.99635E+00
0.05	Mean	3.81559E+00	**3.79580E+00**	3.80405E+00	3.80747E+00	3.84429E+00	4.07500E+00
	Std.	8.52E–03	1.89E–03	3.07E–03	6.88E–03	1.62E–02	2.79E–01
	Best	3.80046E+00	3.79400E+00	3.79833E+00	3.79842E+00	3.82339E+00	**3.79367E+00**
0.01	Mean	9.82097E+00	**9.78940E+00**	9.80599E+00	9.81877E+00	9.87166E+00	9.93990E+00
	Std.	1.74E–02	9.19E–04	6.07E–03	9.16E–03	2.32E–02	2.00E–01
	Best	9.79300E+00	9.78840E+00	9.79575E+00	9.80198E+00	9.82768E+00	**9.78741E+00**
0.001	Mean	2.43796E+01	**2.43240E+01**	2.43596E+01	2.43919E+01	2.45121E+01	2.51887E+01
	Std.	2.30E–02	1.96E–03	1.02E–02	1.77E–02	5.04E–02	1.13E+00
	Best	2.43481E+01	**2.43230E+01**	2.43418E+01	2.43649E+01	2.43882E+01	2.43241E+01
$\rho = 0.5$							
0.1	Mean	2.31219E+00	**2.30260E+00**	2.31333E+00	2.31651E+00	2.33887E+00	2.49607E+00
	Std.	6.19E–03	2.19E–03	4.87E–03	6.61E–03	1.41E–02	2.50E–01
	Best	2.30151E+00	**2.30070E+00**	2.30616E+00	2.30666E+00	2.31180E+00	2.30100E+00
0.05	Mean	4.86493E+00	**4.83570E+00**	4.86189E+00	4.87636E+00	4.91358E+00	5.13300E+00
	Std.	1.74E–02	2.71E–03	1.27E–02	1.04E–02	1.91E–02	3.81E–01
	Best	4.84294E+00	**4.83300E+00**	4.84199E+00	4.85172E+00	4.87008E+00	4.83300E+00
0.01	Mean	1.59311E+01	**1.58650E+01**	1.59416E+01	1.59902E+01	1.59949E+01	1.60754E+01
	Std.	2.17E–02	3.27E–03	2.47E–02	3.43E–02	1.41E–01	1.82E–01
	Best	1.59008E+01	1.58620E+01	1.59021E+01	1.59305E+01	**1.55192E+01**	1.58611E+01
0.001	Mean	6.16194E+01	**6.06850E+01**	6.16165E+01	6.16119E+01	6.16867E+01	6.59741E+01
	Std.	1.47E–02	6.51E–02	1.77E–02	8.82E–03	1.98E–01	2.93E+00
	Best	6.15896E+01	**6.05660E+01**	6.15852E+01	6.15966E+01	6.06504E+01	6.17442E+01

Table 5.4 Comparison of numerical results when the observations are correlated: $\rho = \{0.01, 0.1, 0.5\}$, $L = 50$ sensors, $\gamma_0 = 10$ dB, and $\varepsilon = \{0.1, 0.05, 0.01, 0.001\}$

ε		CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$\rho = 0.01$							
0.1	Mean	**9.02118E–01**	1.48094E+00	9.96784E–01	1.04841E+00	9.44415E–01	1.87671E+00
	Std.	7.89E–03	4.68E–01	3.22E–02	4.42E–02	1.88E–02	1.36E+00
	Best	8.88871E–01	8.79589E–01	9.42663E–01	9.62202E–01	9.14747E–01	**8.76651E–01**
0.05	Mean	**1.72662E+00**	2.86366E+00	1.84437E+00	1.92668E+00	1.78391E+00	3.44436E+00
	Std.	1.63E–02	8.37E–01	3.30E–02	7.91E–02	2.36E–02	1.75E+00
	Best	**1.69629E+00**	1.69376E+00	1.77337E+00	1.78682E+00	1.74011E+00	1.68829E+00
0.01	Mean	**4.44819E+00**	6.05320E+00	4.67165E+00	4.79533E+00	4.57978E+00	5.69284E+00
	Std.	2.27E–02	7.98E–01	5.41E–02	1.04E–01	3.94E–02	9.10E–01
	Best	4.40527E+00	**4.39060E+00**	4.55601E+00	4.63914E+00	4.46567E+00	4.39809E+00
0.001	Mean	**1.01404E+01**	1.07460E+01	1.05199E+01	1.07251E+01	1.03588E+01	1.12213E+01
	Std.	3.91E–02	3.31E–01	6.85E–02	1.35E–01	6.17E–02	1.41E+00
	Best	1.00660E+01	1.01680E+01	1.03465E+01	1.04958E+01	1.02185E+01	**1.00242E+01**
$\rho = 0.1$							
0.1	Mean	**9.34620E–01**	3.08130E+00	1.03799E+00	1.08992E+00	9.80039E-01	1.74470E+00
	Std.	1.14E–02	4.30E–01	3.06E–02	5.93E–02	1.76E–02	1.08E+00
	Best	**9.15381E–01**	2.43080E+00	1.11310E+00	9.97765E–01	9.35467E–01	9.80038E–01
0.05	Mean	**1.80867E+00**	4.37700E+00	1.94203E+00	2.01247E+00	1.88165E+00	2.88210E+00
	Std.	1.06E–02	6.28E–01	3.55E–02	7.45E–02	2.94E–02	1.15E+00
	Best	**1.78997E+00**	3.37890E+00	1.86992E+00	1.88792E+00	1.83228E+00	1.79232E+00
0.01	Mean	**4.75915E+00**	6.65320E+00	4.98549E+00	5.10913E+00	4.89356E+00	5.96645E+00
	Std.	2.04E–02	1.01E+00	6.12E–02	7.77E–02	4.51E–02	1.17E+00
	Best	**4.71623E+00**	5.49050E+00	4.82053E+00	4.98578E+00	4.80740E+00	4.89258E+00
0.001	Mean	**1.09317E+01**	1.16690E+01	1.14322E+01	1.16480E+01	1.12121E+01	1.16309E+01
	Std.	3.99E–02	5.81E–01	1.15E–01	1.56E–01	7.61E–02	8.29E–01
	Best	1.08731E+01	1.10910E+01	1.12102E+01	1.14085E+01	1.09675E+01	**1.07873E+01**
$\rho = 0.5$							
0.1	Mean	1.62230E+00	1.67224E+00	**1.23836E+00**	1.60562E+00	1.55067E+00	2.00899E+00
	Std.	1.80E–01	3.81E–01	3.62E–02	3.35E–01	4.21E–01	1.57E+00
	Best	1.38950E+00	1.11521E+00	1.19164E+00	1.17440E+00	1.10691E+00	**1.10187E+00**
0.05	Mean	2.77030E+00	3.24208E+00	**2.52436E+00**	3.12709E+00	3.03643E+00	3.18564E+00
	Std.	1.51E–01	6.00E–01	3.95E–02	5.28E–01	6.66E–01	1.33E+00
	Best	2.56400E+00	2.36221E+00	2.45759E+00	2.48189E+00	**2.35095E+00**	2.38701E+00
0.01	Mean	**7.19150E+00**	7.42100E+00	7.22594E+00	8.20864E+00	8.03350E+00	7.80258E+00
	Std.	1.50E–01	3.24E–01	6.71E–02	9.17E–01	1.20E+00	1.04E+00
	Best	6.96500E+00	6.91000E+00	7.02768E+00	7.17023E+00	6.78723E+00	**6.77276E+00**
0.001	Mean	**1.88410E+01**	1.84490E+01	1.92334E+01	2.05440E+01	2.02199E+01	1.91415E+01
	Std.	1.20E–01	3.68E–02	1.68E–01	1.12E+00	1.73E+00	4.94E–01
	Best	1.86090E+01	**1.83800E+01**	1.89643E+01	1.92448E+01	1.84280E+01	1.84879E+01

Table 5.5 The analytical and numerical optimal solutions to the gain allocation when the observations are i.i.d. ($\rho = 0$) and when the observations are correlated: $\rho = \{0.01.0.1.0.5\}$, $L = 10$ sensors, $\gamma_0 = 10$ dB, and $\varepsilon = 0.1$

Sensors	Analytical	CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$\rho = 0$							
S1	1.0362	1.0461	1.0379	1.0362	1.0376	0.9988	1.036
S2	0.9972	0.9995	0.9967	0.9992	0.9982	0.9612	0.9976
S3	0.8834	0.8833	0.8842	0.8879	0.8819	0.9542	0.8829
S4	0.4823	0.4784	0.4781	0.4783	0.4819	0.4735	0.4805
S5	0.3021	0.2641	0.3009	0.2866	0.2968	0.3077	0.3053
S6	0	0.0196	0.0252	0.0258	0.0301	0.1352	0.0033
S7	0	0.0044	0.0131	0	0.0085	0.0985	0.0127
S8	0	1.11E–08	0.0035	0.0025	0	0.009	0
S9	0	9.11E–04	3.90E–05	0	0.0096	5.22E–04	4.45E–04
S10	0	0.0035	0.0017	0.0036	0.0019	0.023	3.25E–04
Poptimal	3.1723	3.1725	3.1723	3.1723	3.1724	3.1797	3.1723
$\rho = 0.01$							
S1		1.0170	1.0387	1.0355	1.0389	1.0047	1.0381
S2		0.9884	0.9959	0.9916	0.9894	1.0346	0.9905
S3		0.9079	0.8833	0.8908	0.8824	0.8842	0.8843
S4		0.4842	0.4816	0.4877	0.4732	0.5196	0.4842
S5		0.3191	0.3146	0.3093	0.3483	0.1860	0.3283
S6		0.0368	0.0359	0.0082	0.0270	0.0154	0.0026
S7		0.1062	0.0038	0.0169	0.0206	0.1470	0.0058
S8		1.46E–08	0.0043	4.52E–04	0.0088	0.0114	2.04E–04
S9		4.02E–10	0.0030	0	9.91E–04	0.0216	1.73E–04
S10		1.04E–08	0.0018	0.0021	0	0.0041	0.0019
Poptimal		3.1843	3.1830	3.1831	3.1832	3.1887	3.1830
$\rho = 0.1$							
S1		1.0470	1.0422	1.0439	1.0460	1.0909	1.0460
S2		0.9783	1.0422	0.9635	0.9566	0.9850	0.9625
S3		0.8558	0.8781	0.8784	0.8774	0.7900	0.8754
S4		0.4833	0.5156	0.5090	0.5207	0.4292	0.5088
S5		0.4087	0.4474	0.4564	0.4583	0.4104	0.4494
S6		0.1606	0.1652	0.1589	0.1207	0.3136	0.1883
S7		0.2704	0.0155	0.0297	0.0933	0.2444	9.12E–05
S8		1.46E–08	1.95E–07	5.75E–04	0	0.0328	0
S9		3.34E–09	2.92E–09	4.69E–04	0	0.0086	3.56E–05
S10		2.83E–09	0	0.0051	0	0.0199	0
Poptimal		3.2852	3.2832	3.2833	3.2835	3.2968	3.2832

Table 5.5 (Continued)

Sensors	Analytical	CBBO	CBBO-DE	2-Stage-CBBO-DE	CDE	CGA	CPSO
$\rho = 0.5$							
S1		1.1575	1.1331	1.1346	1.1309	1.0956	1.1332
S2		0.8641	0.8202	0.8218	0.8375	0.7977	0.8334
S3		0.7925	0.8655	0.8579	0.8633	0.8290	0.8456
S4		0.7110	0.1630	0.2273	0.1443	0.0462	0.2466
S5		0.6426	0.6246	0.6178	0.6078	0.8519	0.5773
S6		0.4774	0.4436	0.4932	0.4826	0.2903	0.5087
S7		8.99E–09	0.7289	0.6948	0.7138	0.7070	0.7152
S8		1.53E–09	0	0.0093	0.0057	0.0027	2.32E–05
S9		4.51E–09	4.98E–04	0.0062	0.0064	0.0874	6.46E–04
S10		3.95E–09	0	0.0172	0	0.1389	0.0018
Poptimal		3.8610	3.8580	3.8583	3.8584	3.8630	3.8582

5.6 Conclusion

The present work has considered the problem of optimal power scheduling for the decentralized detection of a deterministic signal in a WSN with power and bandwidth constrained distributed nodes. An efficient optimal power allocation scheme has the potential of suitably turning off the nodes with poor channels and providing significant system power savings. In this work, three variants of the BBO algorithm have been proposed for the optimal power allocation in WSNs. These algorithms have been compared with three other competing algorithms, i.e., three separately developed constrained versions of the DE, GA, and PSO algorithms. It has been shown that the CBBO-DE algorithm has outperformed the other competing algorithms for several types of simulation case studies, including both independent local observation cases and correlated observation cases. It has also been observed that, in the case of a large number of sensors, CBBO emerged as the best performer.

Finally, in this work, the fusion center is given the information on channel condition, assumed to obey Rayleigh fading. For the situation where the channel condition is changing rapidly, the power allocation needs to be updated dynamically to ensure an optimum performance. However, if the changes are reasonably slow, there will be enough time to properly update the power allocation. It can be argued that for applications operating in a dynamic environment, there are other potential alternatives using methods for dynamic optimization.

References

1. Abadir, K., Magnus, J.: Matrix Algebra. Econometric Exercises 1. Cambridge University Press, Cambridge (2005)
2. Akyildiz, I.F., Su, W., Sankasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**, 393–422 (2002)

3. Back, T.: Evolutionary Algorithms in Theory and Practice. Oxford Univ. Press, Oxford (1996)
4. Boussaïd, I., Chatterjee, A., Siarry, P., Ahmed-Nacer, M.: Hybridizing biogeography-based optimization with differential evolution for optimal power allocation in wireless sensor networks. IEEE Trans. Veh. Technol. **60**(5), 28–39 (2011). doi:10.1109/TVT.2011.2151215
5. Boussaïd, I., Chatterjee, A., Siarry, P., Ahmed-Nacer, M.: Two-stage update biogeography-based optimization using differential evolution algorithm (DBBO). Comput. Oper. Res. **38**, 1188–1198 (2011)
6. Boussaïd, I., Chatterjee, A., Siarry, P., Ahmed-Nacer, M.: Biogeography-based optimization for constrained optimization problems. Comput. Oper. Res. **39**, 3293–3304 (2012)
7. Chamberland, J.F., Veeravalli, V.V.: Decentralized detection in wireless sensor systems with dependent observations. In: International Conference on Computing, Communications and Control Technologies, Austin, TX (2004)
8. Coello, C.: Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput. Methods Appl. Mech. Eng. **191**(11–12), 1245–1287 (2002)
9. Deb, K.: An efficient constraint handling method for genetic algorithm. Comput. Methods Appl. Mech. Eng. **186**, 311–338 (2000)
10. Deb, K., Agrawal, R.: Simulated binary crossover for continuous search space. Complex Syst. **9**, 115–148 (1995)
11. Dow, M.: Explicit inverses of Toeplitz and associated matrices. ANZIAM J. **44**(E), E185–E215 (2003)
12. Kay, S.: Fundamentals of Statistical Signal Processing, Vol. 2: Detection Theory. Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs (1998)
13. Kuhn, H.W.: Nonlinear programming: a historical view. SIGMAP Bull. **31**, 6–18 (1982). doi:10.1145/1111278.1111279
14. Levy, B.C.: Principles of Signal Detection and Parameter Estimation. Springer, Berlin (2008)
15. MacArthur, R., Wilson, E.: The Theory of Biogeography. Princeton University Press, Princeton (1967)
16. Michalewicz, Z., Schoenauer, M.: Evolutionary algorithms for constrained parameter optimization problems. Evol. Comput. **4**(1), 1–32 (1996)
17. Neyman, J., Pearson, E.S.: On the problem of the most efficient tests of statistical hypotheses. Philos. Trans. R. Soc. Lond. Ser. A **231**, 289–337 (1933)
18. Poor, H.: An Introduction to Signal Detection and Estimation, 2nd edn. Springer Texts in Electrical Engineering. Springer, New York (1998)
19. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Natural Computing Series. Springer, Berlin (2005)
20. Runarsson, T.P., Yao, X.: Stochastic ranking for constrained evolutionary optimization. IEEE Trans. Evol. Comput. **4**, 284–294 (2000)
21. Simon, D.: Biogeography-based optimization. IEEE Trans. Evol. Comput. **12**, 702–713 (2008)
22. Storn, R.M., Price, K.V.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**, 341–359 (1997)
23. Swami, A., Zhao, Q., Hong, Y.: Wireless Sensor Networks: Signal Processing and Communications. Wiley, New York (2007)
24. Tenny, R.R., Sandell, N.R.: Detection with distributed sensors. IEEE Trans. Aerosp. Electron. Syst. **17**, 501–510 (1981)
25. Trees, H.: Detection, Estimation, and Modulation Theory. Wiley-Interscience, New York (1998)
26. Tsitsiklis, J.: Decentralized detection. Adv. Stat. Signal Process. **2**, 297–344 (1993)
27. Wimalajeewa, T., Jayaweera, S.K.: Optimal power scheduling for correlated data fusion in wireless sensor networks via constrained PSO. Trans. Wirel. Commun. **7**(9), 3608–3618 (2008). doi:10.1109/TWC.2008.070386

Chapter 6
Joint Optimization of Detection and Tracking in Adaptive Radar Systems

Murat Şamil Aslan and Afşar Saranlı

Abstract A promising line of research attempts to bridge the gap between a detector and a tracker by means of considering jointly optimal parameter settings for both of these subsystems. Along this fruitful path, this chapter focuses on the problem of detection threshold optimization in a *tracker-aware* manner so that a feedback from the tracker to the detector is established to maximize the overall system performance. Special emphasis is given to the optimization schemes based on two non-simulation performance prediction techniques for the probabilistic data association filter, namely, the modified Riccati equation (MRE) and the hybrid conditional averaging algorithm. The possible improvements are presented in non-maneuvering target tracking where a number of algorithmic and experimental evaluation gaps are identified and newly proposed methods are compared with the existing ones in a unified theoretical and experimental framework. Furthermore, for the MRE-based dynamic threshold optimization problem, a closed-form solution is proposed. This solution brings a theoretical lower bound on the operating signal-to-noise ratio concerning when the tracking system should be switched to the track-before-detect mode.

6.1 Introduction

Radar systems are one of the most important remote sensing equipments available today. They are used everywhere including civilian, military, and space applications, and being all-weather they are indispensable for long-range surveillance.

Radar systems typically radiate a pulse of electromagnetic energy and capture the returning echo for the purpose of determining the location, velocity, and other state information of a "target" of interest. To achieve this goal, the captured electro-

M.Ş. Aslan (✉)
İLTAREN, Advanced Technologies Research Institute, TÜBİTAK BİLGEM, Ümitköy 06800, Ankara, Turkey
e-mail: msamil.aslan@tubitak.gov.tr

A. Saranlı
Dep. of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey
e-mail: afsars@metu.edu.tr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_6,

magnetic echo is first converted to an electrical signal and passed through a signal processing stage which includes *signal conditioning*[1] [57] and *detection*. This is usually followed by a *tracking* (also called *information processing* [6] or *data processing* [12, 57]) stage as illustrated in Fig. 6.1.

With this traditional treatment of viewing the overall radar system as a concatenation of two subsystems, the radar research has been conducted along two distinct paths, namely "detection theory" and "tracking theory" with not much interaction between them. The tracking literature mostly assumed that the detection (or signal processing) stage is a prior and isolated process, providing *measurements* for the tracking stage. Given a set of such measurements, most of the studies aim to optimize the tracking filter based on either the *minimum mean square error* (MMSE) or the *maximum a posteriori* (MAP) criterion [9]. Similarly, researchers in radar signal processing literature usually assumed no incoming information from the downstream tracking algorithms. Their common optimization approach in the detection phase is first to specify a desired (or acceptable) false alarm probability (P_{FA}^d) for the detector and then maximize the probability of detection (P_D) with this constraint [65]. The value of P_{FA}^d is usually selected in view of the radar processor's computational capacity in handling the maximum number of false alarms. Although this seems a reasonable criterion, it is only a heuristic one. It neither accounts for the properties of the downstream tracker, nor it cares for an overall performance objective.

A reasonable and challenging question is whether parameter decisions made for the detector and tracker subsystems are optimal for the combined performance of the overall radar system. Intuitively, one can easily see that thresholding in the detection phase might have significant influence on downstream tracking performance. In one extreme case where no thresholding is applied, targets are detected perfectly but together with lots of false alarms. In the other extreme where the threshold is set very high, false alarms are greatly reduced but together with a high probability of missing the targets.

Another equally important question is whether these subsystem level parameters have to be statically optimized or should they rather be adaptive in space and time. One strongly feels that some adaptation is necessary since the motion of the target changes both the spatial context and the Signal-to-Noise Ratio (SNR). Yet another important concern in adaptive optimization of these subsystem level parameters might be the operating regime (i.e., transient or steady-state) of the tracking filter. Depending on how far from its steady-state operating region it is, the filter could, for example, be fed with more or less false alarms and missed detections.

In this chapter, we focus on answers to these exciting questions. In particular, we consider the interaction between the detector and the tracker subsystems and focus on how we can optimally select the operating P_{FA} value[2] of the detector in

[1] This includes the processing blocks prior to detection such as analog-to-digital (A/D) conversion, beamforming, pulse compression, clutter filtering, and Doppler processing [57].

[2] This in turn determines the detector operating point (P_{FA}, P_D) for a given SNR and hence the detection threshold.

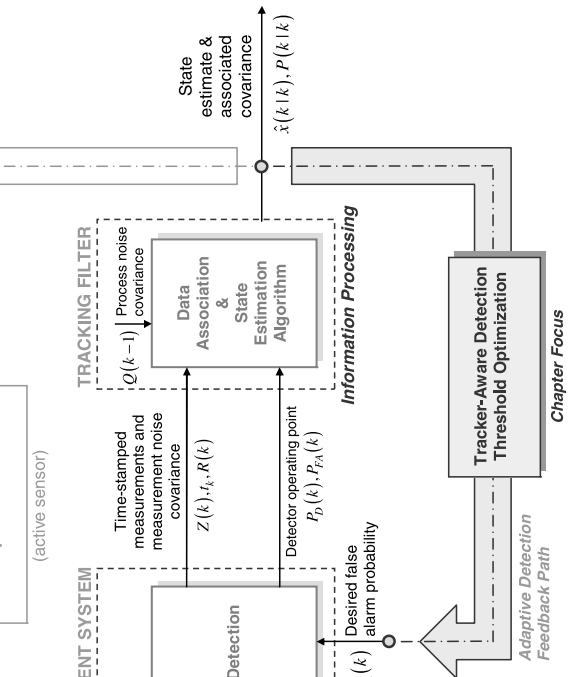


Fig. 6.1 Block diagram of a conventional tracking system with a decoupled feed-forward structure. This block diagram is rather generic in the sense that the sensor can be either an active one such as an active sonar, radar, or ladar (laser detection and ranging), or a passive one such as a passive sonar, electro-optical (EO) or infrared (IR) sensor. In the active case, the energy flow in the channel is *bidirectional* where the energy is transmitted towards the target via a transmitter (Tx) and the returned echo is received with a receiver (Rx). The type of energy can be either *electromagnetic* as in the case of a radar, ladar, EO and IR sensors, or *acoustic* as in the case of an active/passive sonar. The focus of the chapter, the *adaptive detection feedback path*, is to a large extend independent of the sensor type used except when a specific detector is used as an example. However, to be more specific, the presentation in this chapter is in the context of a radar sensor

a *tracker-aware* manner so that the performance of the downstream tracker, hence of the combined target state estimation system, is maximized. This structure constitutes a form of feedback from the downstream tracker to the upstream detector as illustrated in Fig. 6.1. We strongly believe that this is an important subarea of the research consisting of steps that are necessary for the ultimate goal of *cognitive radar* [25] which also includes the upper feedback path in Fig. 6.1, i.e., *adaptive waveform optimization*.

6.1.1 Related Work

The covariance of the filter's estimation error is one of the most important performance measures for the downstream tracking. So, as a good starting point, researchers first attempted to form an objective function which links their *optimization variables*[3] to a scalar function of this covariance, e.g., its trace or determinant. However, the problem with the tracking filters operating under clutter (or *measurement origin uncertainty* [10]) is that their covariance expressions depend on the measurements they received, hence are related to the optimization variables through *stochastic* recursions. This makes finding a solution to the concerned optimization problem difficult in two aspects. First, it makes difficult to apply classical optimization tools to the problem due to the stochastic nature of the objective function. Second, from the causality point of view, optimizing the variables through measurement-dependent objective functions makes the resulting solution impractical as this optimal setting of parameters are supposed to produce these measurements again. These two reasons motivated researchers to seek possibly approximate but *measurement-independent* (i.e., deterministic) covariance recursions for these filters. Finding a deterministic covariance recursion can also be viewed as making a *non-simulation performance prediction* (NSPP) for the filter under concern, as such a recursion helps evaluating (approximately) the performance of the filter without recourse to time-consuming Monte Carlo runs.

The line of study devoted to the optimization of the radar detector by minimizing a cost function based on such a *deterministic* covariance recursion has been pioneered by Fortmann et al. [20] where they considered the Probabilistic Data Association Filter (PDAF) [10] as a tracking filter. The main contributions of [20] are the so-called *Modified Riccati Equation* (MRE), which provides an approximate *deterministic* covariance recursion for the PDAF, and *tracker operating characteristic* (TOC) curves, which are by-product of the steady-state solution of the MRE. Fortmann et al. have shown that for a given SNR, one can determine the optimal detector operating point by finding the tangential intersection point between TOC curves of the tracking filter and the corresponding receiver operating characteristic

[3] In this chapter, we consider only the detection threshold as our optimization variable. A more general set including also the transmitting waveform as in the case of *cognitive radar* [25] is out of the scope of the present chapter.

		OPTIMIZATION APPROACH	
		STATIC	DYNAMIC
NSPP	MRE	***STATIC-MRE-TOC*** [Fortmann85]	***DYNAMIC-MRE-LS*** [Gelfand96] ***DYNAMIC-MRE-CF*** [Aslan10]
	HYCA	***STATIC-HYCA-LUT*** [Li91] ***STATIC-HYCA-TOC*** [Aslan11]	***DYNAMIC-HYCA-LS*** [Aslan11]

Fig. 6.2 The algorithmic space of NSPP-based tracker-aware detector threshold optimization schemes for tracking a *nonmaneuvering* target with the PDAF. Here, the abbreviations TOC, LUT, LS, and CF correspond to the "tracker operating characteristic", "look-up table", "line search", and "closed-form", respectively [2]

(ROC) curve of the detector. The introduced MRE provides a *steady-state* NSPP for the PDAF in clutter, very similar to how the standard Riccati equation (SRE) does for the Kalman filter in clutter-free environments. We call this approach to tracker-aware detector threshold optimization as *NSPP-based static* [2] approach (called **STATIC-MRE-TOC** in Fig. 6.2). Belonging to the same category is a set of studies by Li et al. [46] where they have improved the idea in [20] by introducing a different deterministic covariance recursion (an NSPP tool) for the PDAF, called the Hybrid Conditional Averaging (HYCA) algorithm. Interestingly its application to optimization of detection thresholds using a look-up-table procedure (called **STATIC-HYCA-LUT** in Fig. 6.2) has made limited use of the results [48]. A neat comparison of MRE-based and HYCA-based static threshold optimization using TOC curves has been recently made in [2].

When the steady-state analysis is inappropriate, such as in time-varying or nonlinear systems, a suggested solution is to apply the same methodology by iterating the MRE or HYCA not to their steady-state but for n steps into the future. In the case of $n = 1$, this leads to a *dynamic* [2] (also called *adaptive* [22]) threshold optimization scheme. In [22], Gelfand et al. proposed two such problems, namely, *prior* and *posterior* threshold optimization, where they minimize the mean-square state estimation error over detection thresholds, based on the measurements up to the previous (prior) and current (posterior) time steps, respectively. It was further shown that for the *prior* case the problem reduces into a single line search [22] (**DYNAMIC-MRE-LS** in Fig. 6.2). Due to the claimed mathematical intractability of obtaining a full closed-form solution, in [22] this problem was solved using iterative numerical optimization techniques, such as the Golden-Section and Fibonacci Search methods [11]. In [5], the same problem is solved in an approximate closed-form (**DYNAMIC-MRE-CF** in Fig. 6.2). The solution was applicable for a special case of Neyman–Pearson (NP) detector and based on a functional approximation introduced by [40]. It was shown that this approximate closed-form solution leads to considerable reduction in computational complexity without any notable loss in performance [4, 5]. A comparison of aforementioned NSPP-based detector threshold

optimization schemes given in Fig. 6.2 has been presented in [2]. Apart from a comprehensive experimental survey, the primary contribution of [2] was the establishment of a unified experimental and theoretical framework to categorize and compare these schemes as *static* or *dynamic* threshold optimization as given in Fig. 6.2.

It is worth noting that another fundamentally different approach to the detection threshold optimization problem has been considered in the literature by Willett et al. [68]. This method, while being a perfectly valid alternative approach, differs from all NSPP-based approaches in that it is based on an optimal Bayesian detector framework where the prior hypothesis probabilities required by the detector are fed back from the posterior *information state*[4] of the PDAF.

In all aforementioned studies, which are the NSPP-based methods in Fig. 6.2, i.e., [2, 5, 20, 22, 48], and the one presented in [68], it is implicitly assumed that the model describing the target motion dynamics is fairly well known to the filter (*non-maneuvering target* assumption). This assumption has been relaxed in [3] where a threshold optimization problem is formulated and solved for tracking maneuvering targets by extending the previous ideas applicable to the PDAF to multiple model filtering structures which use PDAFs as modules. A recent study on the same line has been presented by Wang et al. in [66] where instead of a Gaussian mixture as in [3], a moment-matched single Gaussian has been used in the cost function.

A line of recent articles [17, 23, 26, 27] show the growing interest into the concept of *cognitive radar* [24, 25], which aims to make a radar system smarter and more adaptive by dynamically optimizing the "transmitter" as well. We should note, however, that steps towards this goal are not entirely new. In the context of overall system optimization, the optimization of transmitter waveforms was first introduced in [39] and applied to the PDAF in [41]. Another study was [32] where the design of the waveform and detection threshold for range and range-rate tracking in clutter is formulated and numerically solved as a finite horizon optimization problem. A good summary of waveform optimization for tracking is given in [67].

6.1.2 Chapter Outline

In the present chapter, we consider these exciting theoretical and experimental steps towards the goal of spatially and temporally adaptive radar. In particular, we focus on the interaction between the detector and the tracker subsystems and consider the problem of *tracker-aware* optimization of detector threshold per target track and per resolution cell. We build on two important NSPP techniques for the PDAF, namely, MRE of Fortmann [20] and HYCA of Li [46]. There are two common properties of the optimization problems that we consider: First, they all aim at maximizing the performance of a tracking filter over detection thresholds.[5] Second, the cost

[4] See, e.g., [9, pp. 373].

[5] This in essence results in a feedback from the tracker to the detector as illustrated in Fig. 6.1.

(or objective) functions of the optimization problems are all based on an *offline* approximation of the filter's covariance, which is obtained by either MRE or HYCA. Therefore, in Sect. 6.2, we briefly present these two important NSPP techniques for the PDAF. Based on that, in Sect. 6.3, tracker-aware detector threshold optimization problems are defined and their solution methodologies are explained. In Sect. 6.4, these algorithms are compared through a number of simulation experiments. Finally, in Sect. 6.5, the important results of the chapter are summarized and possible future studies from the present point are discussed.

6.2 Offline Performance Evaluation of Tracking Algorithms

Involving both *continuous* and *discrete* uncertainties,[6] real-world tracking is a *hybrid* [45] estimation problem. Tracking algorithms which operate under these uncertainties are necessarily stochastic. As the performance of these algorithms cannot be evaluated *confidently* with a single run, the common practice for performance evaluation is to run an extensive number of Monte Carlo simulations and take the ensemble average of a performance measure over the runs.

Although this methodology is very simple and straightforward, it might be very time-consuming and costly in some cases. More importantly, if a design, an *optimization*,[7] or sensitivity analysis of a tracking algorithm is of interest, Monte Carlo simulations based approach does not give much insight into the problem. In that case, analytical expressions and deterministic tools are much more useful. So, the techniques for performance evaluation that do not require expensive stochastic simulations are needed. There are numerous works done in this context in the literature. However, the available tools for offline evaluation of the performance can roughly be classified into three categories [49]:

- **Error Bounding Techniques:** These techniques are the most popular offline performance evaluation tools. They provide Cramér–Rao like bounds on the performance. There are lots of works done under different titles (possibly having much in common), such as nonlinear filtering [19, 38, 63, 64], filtering with intermittent observations [14, 15, 19, 28, 56, 61], tracking in clutter [29, 34, 70], bearing-only tracking [16, 34, 69], multitarget tracking [18, 33, 43, 58], and maneuvering target tracking [30, 69]. Rather than predicting the filter performance, these techniques put some *best-achievable* borders for the problem at hand. The tightness of such

[6] Examples of continuous uncertainties are the inaccuracy in the measurements and "small" perturbations in the target motion which are usually modeled as an additive measurement noise and process noise, respectively. These types of uncertainties are well-understood and solved in the literature over the past four decades under the title of *classical state estimation* [1, 9, 42]. However, major challenges of tracking arise from two discrete-valued uncertainties: measurement origin uncertainty, which is, in the words of Li and Bar-Shalom [46], the *crux* of tracking, and target maneuver which appears as an *abrupt* and "large" deviation in the target motion.

[7] In this chapter, we consider this aspect, i.e., *tracker-aware optimization of detection thresholds*.

bounds is usually not known and questionable. In this aspect, they can provide at most semi-quantitative measures for offline performance evaluation of a tracking filter.

- **Analytic Model Approach:** The second class of tools is referred to as the analytic model approach [49]. In this methodology, the aim is to establish some (possibly approximate) analytic relationships between the performance measure and some "key" parameters of the algorithm (see, e.g., [8, 51, 59]). Although these techniques provide analytically useful expressions, they are obtained under several assumptions and approximations due to complexity of the big picture. Therefore, their accuracy is still not as good as of the performance prediction approach.
- **Performance Prediction Approach:** This is an algorithmic approach. It aims at obtaining an offline (or a deterministic) algorithm for calculation of one of the performance measures of the tracking filter, usually the estimation error covariance. Developing such a deterministic algorithm for the covariance propagation is in general a hard task. However, this methodology is proven to produce much more accurate results compared to the previous two techniques mentioned above (see, for example, [20, 46], and [47]).

In this section (and also in the entire chapter), our focus will be on the performance prediction category which we refer to as *non-simulation performance prediction* (NSPP) techniques. The key point in NSPP techniques is to obtain a *deterministic* recursion for the *estimation error covariance*, which then can be used to quantify the filter's performance offline. In the simplest case, when there is no clutter and no variation in target dynamics (i.e., no "target maneuver" in tracking terminology), the *Kalman filter* [37] already has a deterministic covariance recursion in the form of a (matrix) *Riccati equation* [1]. However, for the more complex situations in which there is clutter or the target dynamics is time-varying, the error covariance calculation of the filter under concern is no longer deterministic. This is due to the presence of discrete type uncertainties introduced into the problem, which makes the covariance calculation dependent on the measurements received, hence stochastic.

To be able to make an NSPP for the filters in these situations, there are two main methodologies proposed so far. The first methodology, which also pioneered the NSPP topic, is the work of Fortmann et al. [20] where both types of uncertainties (discrete and continuous) in the problem are *globally* averaged out. In this pioneering work [20], the authors applied this methodology for the *Probabilistic Data Association Filter* (PDAF) [7] and obtained a Riccati-like recursion for the deterministic calculation of its covariance. This recursion was named as the *Modified Riccati Equation* (MRE) [20]. The MRE approach is further extended to multi-sensor case (Multisensor PDAF—MSPDAF) by Frei [21], and recently studied in the context of NSPP for Kalman filtering with intermittent observations [14, 15, 61].

Inspired by the work of Fortmann et al., the second methodology was proposed by Li et al. in [46] where only the continuous uncertainties are averaged, while the discrete uncertainties are retained in the propagation of the covariance. Similar to Fortmann et al, in the proposal paper of their algorithm [46], they first derive it for the PDAF and name it as the *Hybrid Conditional Averaging* (HYCA) algorithm.

However, they also note and show that rather than being applicable only for the PDAF, HYCA is actually a methodology that can be applied for NSPP of various hybrid filters, such as the Interacting Multiple Model (IMM) filter [47], MSPDAF [21], the Nearest Neighbor Filter (NNF) [50], and the Strongest Neighbor Filter (SNF) [44].

In the following subsections, we will first construct our state-space representation of target dynamics and measurement system, then we briefly explain the two NSPP approaches for the PDAF, namely, MRE [20] and HYCA [46].

6.2.1 Target and Measurement Models

We assume the following models for the target motion and measurement process:

- The state of the target of interest, of dimension n_x, is assumed to make its transition in time according to the equation

$$x(k+1) = F(k)x(k) + G(k)v(k), \quad k = 0, 1, \ldots, \tag{6.1}$$

  where $\{v(k)\}$, called *process noise*, is a white sequence with $v(k) \sim \mathcal{N}(0, Q(k))$.[8] The linear system dynamics represented by the state transition matrix $F(k)$ and process noise gain matrix $G(k)$ are assumed to be known for all k (non-maneuvering target assumption). The initial state $x(0)$, which is generally unknown, is modeled as $x(0) \sim \mathcal{N}(\hat{x}(0|0), P(0|0))$ where the mean $\hat{x}(0|0)$ and the covariance $P(0|0)$ are assumed to be known.
- The true (i.e., target originated) measurement, of dimension n_z, is given by

$$z(k) = Hx(k) + w(k), \quad k = 1, 2, \ldots, \tag{6.2}$$

  where $\{w(k)\}$, called the *measurement noise*, is a white sequence with $w(k) \sim \mathcal{N}(0, R(k))$ and H, assumed constant, is the measurement matrix linking the state and the measurement vectors.
- The two noise sequences $\{v(k)\}$ and $\{w(k)\}$ and the initial state $x(0)$ are assumed to be *mutually uncorrelated* for all k.
- At each time step k, the true measurement defined in (6.2) is available with a known *detection probability* possibly less than unity, i.e., $P_D(k) \leq 1$.
- *False alarm* or *clutter* lead to false measurements. The locations of these measurements are modeled as random variables which are *independent identically distributed* (i.i.d.) with uniform spatial distribution over the *validation gate*, which is a hyper-ellipsoid in the measurement space, defined by [6, pp. 95]

$$V_G(k, \gamma_G) \triangleq \left\{ z(k) : \left[z(k) - \hat{z}(k|k-1) \right]^T S^{-1}(k) \left[z(k) - \hat{z}(k|k-1) \right] \leq \gamma_G \right\} \tag{6.3}$$

[8] The common notation $x \sim \mathcal{N}(\bar{x}, \Sigma)$ means that "the random variable x is normally (Gaussian) distributed with mean $\bar{x}$ and covariance Σ."

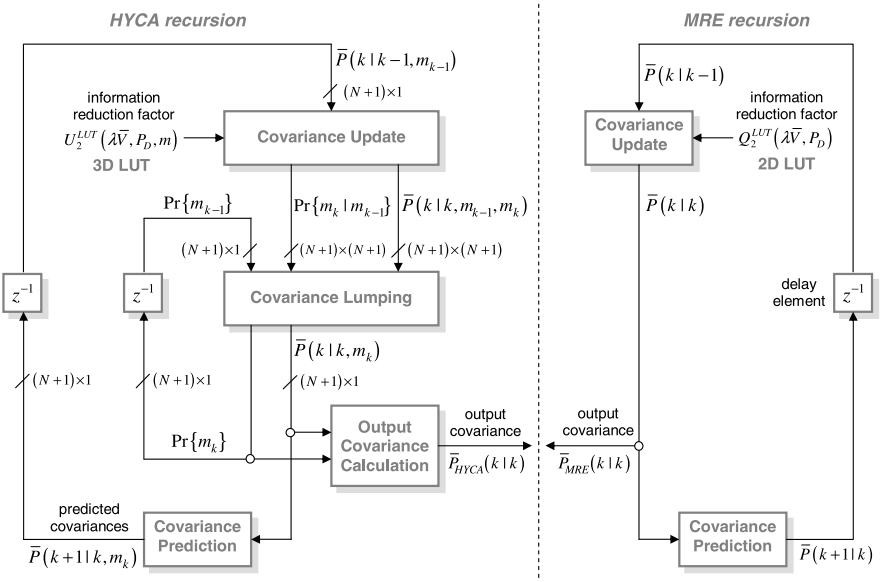


Fig. 6.3 Block diagrams of two offline covariance recursion algorithms for the PDAF: HYCA [46] and MRE [20]. The corresponding output of each algorithm, denoted by $\bar{P}_{\mathrm{HYCA}}(k|k)$ and $\bar{P}_{\mathrm{MRE}}(k|k)$, is a deterministic approximation to the filter calculated covariance, $P(k|k)$ of the PDAF

where γ_G is the *gate threshold*. Here, $\hat{z}(k|k-1)$ is the measurement predicted by the tracking filtering algorithm, which is in our case the PDAF, and $S(k)$ is the covariance associated with the difference $z(k) - \hat{z}(k|k-1)$, which is the *innovation* corresponding to the target-originated measurement.

- Assuming that $\lambda(k)$ and $V(k)$ are the spatial clutter density and the volume of the validation gate at time step k, respectively, the number of false measurements at any time step k, denoted by m_k^F, is modeled as a random variable with probability mass function (pmf) $\mu_F(m_k^F; \lambda(k)V(k))$ where $\mu_F(m; \bar{m})$ denotes the Poisson pmf for the dummy variable m with mean $\bar{m}$, i.e.,

$$\mu_F(m; \bar{m}) \triangleq \frac{e^{-\bar{m}} \bar{m}^m}{m!}. \tag{6.4}$$

6.2.2 NSPP Techniques for the PDAF

There exist two NSPP algorithms proposed so far for the PDAF, namely, MRE [20] and HYCA [46]. The block diagrams of these offline covariance recursion algorithms are given in Fig. 6.3. The steps of each algorithm are briefly summarized in the following subsections.

6.2.3 The Modified Riccati Equation (MRE)

Given $\bar{P}(k|k-1)$ at time step $k-1$, one step recursion of MRE algorithm produces $\bar{P}(k+1|k)$ at time step k. The recursion is initialized with $\bar{P}(1|0) \triangleq F(0)P(0|0)F^T(0) + G(0)Q(0)G^T(0)$. The algorithm consists of two main parts, whose derivation details can be found in [20]:

(i) *Covariance Update:*

$$\bar{S}(k) = H\bar{P}(k|k-1)H^T + R(k), \tag{6.5}$$

$$\bar{W}(k) = \bar{P}(k|k-1)H^T\bar{S}^{-1}(k), \tag{6.6}$$

$$\bar{V}(k) = c_{n_z} g^{n_z} \left|\bar{S}(k)\right|^{1/2}, \tag{6.7}$$

$$q_2(k) = Q_2^{\mathrm{LUT}}\big(\lambda\bar{V}(k), P_D\big), \tag{6.8}$$

$$\bar{P}(k|k) = \bar{P}(k|k-1) - q_2(k)\bar{W}(k)\bar{S}(k)\bar{W}^T(k) \triangleq \bar{P}_{\mathrm{MRE}}(k|k), \tag{6.9}$$

(ii) *Covariance Prediction:*

$$\bar{P}(k+1|k) = F(k)\bar{P}(k|k)F^T(k) + G(k)Q(k)G^T(k). \tag{6.10}$$

Here, the output of the algorithm $\bar{P}(k|k) \triangleq \bar{P}_{\mathrm{MRE}}(k|k)$ is a deterministic approximation to the filter calculated covariance $P(k|k)$ of the PDAF, i.e., $\bar{P}_{\mathrm{MRE}}(k|k) \approx \mathrm{E}[P(k|k)|Z^{k-1}]$. At each time step k, the value of $q_2(k)$ can be obtained from a two dimensional (2D) LUT, $Q_2^{\mathrm{LUT}}(\cdot,\cdot)$ via interpolation where $Q_2^{\mathrm{LUT}}(\cdot,\cdot)$ is prepared offline and only once from the *Information Reduction Factor* (IRF) given in (6.11):

$$q_2\big(\lambda\bar{V}(k), P_D\big) \triangleq P_D \frac{c_{n_z}}{(2\pi)^{n_z/2}} \sum_{m_k=1}^{\infty} \frac{e^{-\lambda\bar{V}(k)}(\lambda\bar{V}(k))^{m_k-1}}{(m_k-1)!} \left(\frac{n_z}{g^{n_z}}\right)^{m_k-1} \times I_2\big(\lambda\bar{V}(k), P_D, m_k\big) \tag{6.11}$$

with

$$I_2\big(\lambda\bar{V}(k), P_D, m_k\big) \triangleq \int_0^g \cdots \int_0^g \frac{\exp(-r_1^2)r_1^2}{b(\lambda\bar{V}(k), P_D) + \sum_{j=1}^{m_k}\exp(-r_j^2/2)} \times (r_1 r_2 \cdots r_{m_k})^{n_z-1}\, dr_1\, dr_2 \cdots dr_{m_k}, \tag{6.12}$$

$$b\big(\lambda\bar{V}(k), P_D\big) \triangleq (2\pi)^{n_z/2} \frac{\lambda\bar{V}(k)}{c_{n_z} g^{n_z}} \frac{(1 - P_D P_G)}{P_D} \tag{6.13}$$

where $c_{n_z} \triangleq \pi^{n_z/2}/\Gamma(n_z/2+1)$, with $\Gamma(\cdot)$ being the gamma function, is the volume of the n_z-dimensional unit hypersphere ($c_1 = 2, c_2 = \pi, c_3 = 4\pi/3$, etc.), and $g \triangleq \sqrt{\gamma_G}$ is referred to as "number of sigmas" (standard deviations) of the gate and

linked to the *gate probability*[9] via chi-square tables. The relationship between λ and the probability of false alarm (P_{FA}) is given by $\lambda \triangleq P_{\text{FA}}/V_C$ where V_C is the *resolution* (*or detection*) *cell volume*.

6.2.4 The Hybrid Conditional Averaging (HYCA) Algorithm

The block diagram of the HYCA (see Fig. 6.3) contains two delay elements corresponding to two main recursions in the algorithm: the recursion of the prediction covariances $\{\bar{P}(k|k-1, m_{k-1})\}_{m_{k-1}=0}^{N}$ and that of the marginal probabilities $\{\Pr\{m_{k-1}\}\}_{m_{k-1}=0}^{N}$. Therefore, given these two at time step $k-1$, one step recursion of HYCA produces $\{\bar{P}(k+1|k, m_k)\}_{m_k=0}^{N}$ and $\{\Pr\{m_k\}\}_{m_k=0}^{N}$ at time step k. The recursions are initialized with $\bar{P}(1|0, m_0) \triangleq F(0)P(0|0)F^T(0) + G(0)Q(0)G^T(0)$ and $\Pr\{m_0\} \triangleq 1/(N+1)$ for $m_0 = 0, 1, \ldots, N$. The algorithm consists of four main parts, whose derivation details can be found in [46]:

(i) *Covariance Update:* For each $m_{k-1} = 0, 1, \ldots, N$, and for each $m_k = 0, 1, \ldots, N$,

$$\bar{S}(k, m_{k-1}) = H\bar{P}(k|k-1, m_{k-1})H^T + R(k), \tag{6.14}$$

$$\bar{W}(k, m_{k-1}) = \bar{P}(k|k-1, m_{k-1})H^T\bar{S}^{-1}(k, m_{k-1}), \tag{6.15}$$

$$\bar{V}(k, m_{k-1}) = c_{n_z} g^{n_z} \left|\bar{S}(k, m_{k-1})\right|^{1/2}, \tag{6.16}$$

$$u_2(k, m_k) = U_2^{\text{LUT}}\left(\lambda\bar{V}(k, m_{k-1}),\, P_D, m_k\right), \tag{6.17}$$

$$\begin{aligned}\bar{P}(k|k, m_{k-1}, m_k) = \big(&\bar{P}(k|k-1, m_{k-1}) \\ &- u_2(k, m_k)\bar{W}(k, m_{k-1})\bar{S}(k, m_{k-1})\bar{W}^T(k, m_{k-1})\big),\end{aligned} \tag{6.18}$$

$$\begin{aligned}\Pr\{m_k|m_{k-1}\} = &\left[1 + P_D P_G\left(\frac{m_k}{\lambda\bar{V}(k, m_{k-1})} - 1\right)\right] \\ &\times \mu_F\left(m_k; \lambda\bar{V}(k, m_{k-1})\right),\end{aligned} \tag{6.19}$$

where $\mu_F(\cdot;\cdot)$ is the Poisson pmf defined previously in (6.4). At each time step k, the value of $u_2(k, m_k)$ can be obtained from a three dimensional (3D) LUT, $U_2^{\text{LUT}}(\cdot,\cdot,\cdot)$ via interpolation where $U_2^{\text{LUT}}(\cdot,\cdot,\cdot)$ is needed to be prepared offline and only once from the IRF given in (6.20) (which is given in [31] after

[9] The gate probability (P_G) is defined as the probability that the target-originated measurement falls inside the validation gate given that the target is detected.

some manipulations on the original form introduced in [46])

$$u_2\big(\lambda\bar{V}(k,m_{k-1}),\,P_D,m_k\big)=\frac{m_k}{b(\lambda\bar{V}(k,m_{k-1}),\,P_D)+\frac{1}{c_{n_z}}(\frac{2\pi}{g^2})^{n_z/2}P_G m_k}\times\frac{1}{n_z}\left(\frac{n_z}{g^{n_z}}\right)^{m_k}I_2\big(\lambda\bar{V}(k,m_{k-1}),\,P_D,m_k\big)\tag{6.20}$$

where $I_2(\cdot,\cdot,\cdot)$ and $b(\cdot,\cdot)$ are defined in (6.12) and (6.13), respectively.

(ii) *Covariance Lumping:*

$$\Pr\{m_k\}=\sum_{m_{k-1}=0}^{N}\Pr\{m_k|m_{k-1}\}\Pr\{m_{k-1}\},\tag{6.21}$$

$$\Pr\{m_{k-1}|m_k\}=\frac{\Pr\{m_{k-1}\}\Pr\{m_k|m_{k-1}\}}{\Pr\{m_k\}},\tag{6.22}$$

$$\bar{P}(k|k,m_k)=\sum_{m_{k-1}=0}^{N}\bar{P}(k|k,m_{k-1},m_k)\Pr\{m_{k-1}|m_k\}.\tag{6.23}$$

(iii) *Covariance Prediction:*

$$\bar{P}(k+1|k,m_k)=F(k)\bar{P}(k|k,m_k)F^T(k)+G(k)Q(k)G^T(k).\tag{6.24}$$

(iv) *Output Covariance Calculation:* This is an optional part in the sense that it is only for output purposes—it is not a part of the algorithm recursions:

$$\bar{P}_{\mathrm{HYCA}}(k|k)=\sum_{m_k=0}^{N}\bar{P}(k|k,m_k)\Pr\{m_k\}.\tag{6.25}$$

Similar to the MRE case, here, the output of the algorithm $\bar{P}_{\mathrm{HYCA}}(k|k)$ is a deterministic approximation to the filter calculated covariance $P(k|k)$ of the PDAF.

6.3 NSPP-Based Detector Threshold Optimization

The NSPP techniques mentioned in the previous section have found several important application areas in the literature such as detector threshold optimization [20, 22, 48], waveform optimization [32, 39, 41, 53, 62], multisensor tracking (as a sensor selection criterion) [52, 54, 55], multitarget tracking (for the occlusion problem) [36], and multifunction radar resource allocation [35]. In this section, we focus on the area of detector threshold optimization. We consider specifically the PDAF

as a tracking filter. The NSPP-based detector threshold optimization for the PDAF case appeared in the works [5, 20, 22, 48], and [2] which are all given in Fig. 6.2. In this section, we define the underlying optimization problems of these approaches. The presentation is given as in Fig. 6.2, i.e., in two parts, static and dynamic optimization schemes. The section ends with a comparison of these approaches via a simulation scenario.

6.3.1 Static Threshold Optimization (STOP)

The NSPP-based static threshold optimization problem is [2] to determine the optimal P_{FA}^{*} value such that

$$\begin{aligned} P_{\text{FA}}^{*} &= \arg\min_{P_{\text{FA}}} \left\{ f_S[\bar{P}_{\text{NSPP}}] \right\} \\ \text{subject to} \quad & P_D = f_{\text{ROC}}(P_{\text{FA}}, \zeta) \quad \text{and} \quad 0 \leq P_{\text{FA}} \leq 1, \end{aligned} \tag{6.26}$$

where $f_S : \mathbb{R}^{n_x \times n_x} \to \mathbb{R}$ is an appropriate scalar measure deduced from a matrix (such as trace, determinant, or a matrix norm) and $\bar{P}_{\text{NSPP}}$ is the steady-state covariance matrix obtained by propagating one of the NSPP recursions to its steady-state, i.e.,

$$\bar{P}_{\text{NSPP}} \triangleq \lim_{k \to \infty} \bar{P}_{\text{NSPP}}(k|k), \tag{6.27}$$

where $\bar{P}_{\text{NSPP}}(k|k)$ corresponds to the output of either the HYCA or the MRE algorithm at time step k. The equality constraint of the optimization problem is nothing but an ROC curve relation which links P_D to P_{FA}, or vice-versa, through current SNR (ζ), and the inequality constraint ensures that the resultant operating false alarm value is a valid probability.

Remark 6.1 Note that this optimization is performed *offline*. The MRE and HYCA recursions are initialized as explained in the Sects. 6.2.3 and 6.2.4. In the practical implementation, one cannot iterate the recursion given in (6.27) indefinitely. One should check whether the value of a suitably chosen norm of the difference matrix between two consecutive covariance matrices is below a chosen threshold to conclude that the recursion is converged, or whether a maximum number of iterations is reached to conclude on its divergence.

The optimization problem given in (6.26) is a line search. Provided that the cost function is *unimodal*, the global optimum point can be found directly applying well-known numerical techniques, such as the Golden-Section or Fibonacci Search methods [11]. For each function evaluation at an arbitrary point P_{FA}^{i}, one needs to obtain

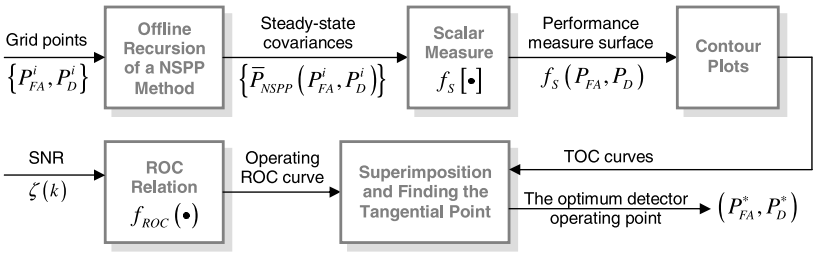


Fig. 6.4 The flow of static threshold optimization (STOP) for graphical (TOC-curve) approach. The procedure given in this figure is repeated for different SNR values to obtain the optimum operating curve in the P_{FA}–P_D plane. Then, this optimal operating curve is used together with ROC curve relation to find the STOP curve which is the ultimate goal of STOP. The STOP curve provides an SNR-dependent optimum P_{FA} setting which makes the threshold optimization online possible under varying SNR conditions. A numerical example is given in Sect. 6.4.1

the steady-state covariance matrix $\bar{P}_{\text{NSPP}}$ from (6.27) for the (P_{FA}^i, P_D^i) pair where $P_D^i \triangleq f_{\text{ROC}}(P_{\text{FA}}^i, \zeta)$.

Another alternative is to utilize a graphical (i.e., TOC-curve) approach. In this case, we first construct the scalar performance measure surface $f_S(P_{\text{FA}}, P_D)$ by evaluating the cost function at each point of a sufficiently fine mesh grid on the P_{FA}–P_D plane. Then, we obtain the contours of this surface, which constitute the TOC curves [20]. Finally, for the current SNR value, we find the tangential point of the corresponding operating ROC curve to the TOC curves. This point is the optimal (P_{FA}^*, P_D^*) pair satisfying the ROC curve relation, hence the solution to the constraint optimization problem defined in (6.26). The procedure is summarized in Fig. 6.4.

Although this graphical technique is computationally more expensive, from the practical applicability point of view, it is not a problem since we make the optimization offline and only once. Furthermore, the graphical approach is more preferable compared to the direct utilization of the line search algorithms, as it allows easier interpretation and better insight into the problem. For both approaches, however, at some points in the P_{FA}–P_D plane, cost function evaluation may be problematic due to non-existence of the limit given in (6.27). This causes an *instability region* [20] in the P_{FA}–P_D plane.

The TOC-curve approach was first used in [20] for solving the static threshold optimization based on MRE, leading to the threshold optimization scheme **STATIC-MRE-TOC** in Fig. 6.2. The same approach is applied to the HYCA case in [2] which results in the optimization scheme **STATIC-HYCA-TOC**. A numerical example comparing these two approaches is given in Sect. 6.4.1.

6.3.2 Dynamic Threshold Optimization (DTOP)

A general form of NSPP-based dynamic threshold optimization problem is defined as [2]

$$\begin{aligned} P_{\mathrm{FA}}^{*}(k) &= \arg\min_{P_{\mathrm{FA}}} \left\{ f_S\left[\bar{P}_{\mathrm{NSPP}}(k|k) \right] \right\} \\ \text{subject to} \quad P_D(k) &= f_{\mathrm{ROC}}\left(P_{\mathrm{FA}}(k), \zeta(k) \right) \quad \text{and} \quad 0 \le P_{\mathrm{FA}}(k) \le 1. \end{aligned} \tag{6.28}$$

Note that the dynamic threshold optimization differs only from its static counterpart in making the optimization not in the steady-state, but at every time step k. This optimization is performed *online*. Different from the static threshold optimization, now, the MRE and HYCA recursions are initialized at every time step with the *online-calculated* covariance of the PDAF.

6.3.2.1 MRE-Based Dynamic Threshold Optimization

The problem formulation based on the MRE approach was first proposed in [22] as the so-called *prior detector threshold optimization*:

$$\begin{aligned} P_{\mathrm{FA}}^{*}(k) &= \arg\min_{P_{\mathrm{FA}}} \left\{ \mathrm{E}\left[\left\| x(k) - \hat{x}(k|k) \right\|^2 \middle| Z^{k-1} \right] \right\} \\ \text{subject to} \quad P_D(k) &= f_{\mathrm{ROC}}\left(P_{\mathrm{FA}}(k), \zeta(k) \right) \quad \text{and} \quad 0 \le P_{\mathrm{FA}}(k) \le 1, \end{aligned} \tag{6.29}$$

where $\hat{x}(k|k)$ is the state estimated by the PDAF at time step k and Z^{k-1} is the cumulative set of *validated measurements*[10] up to $k-1$ (i.e., *prior* to k).

Lemma 6.1 *The optimization problem given in* (6.29) *is equivalent to the one given in* (6.30) *with the choices of* $f_S[\cdot] = \mathrm{tr}\{\cdot\}$ *and* $\bar{P}_{\mathrm{NSPP}}(k|k) = \bar{P}_{\mathrm{MRE}}(k|k)$ *where* $\mathrm{tr}\{\cdot\}$ *is the* trace *operator. Furthermore, it can be reduced to the equivalent optimization problem of the form*:

$$\begin{aligned} P_{\mathrm{FA}}^{*}(k) &= \arg\max_{P_{\mathrm{FA}}} q_2\left(\lambda(k) V(k), P_D(k) \right) \\ \textit{subject to} \quad P_D(k) &= f_{\mathrm{ROC}}\left(P_{\mathrm{FA}}(k), \zeta(k) \right) \quad \textit{and} \quad 0 \le P_{\mathrm{FA}}(k) \le 1, \end{aligned} \tag{6.30}$$

where $q_2(\cdot,\cdot)$ *is the IRF given in* (6.11), $\lambda(k) \triangleq P_{\mathrm{FA}}(k)/V_C$ *is the clutter density, and* $V(k) \triangleq c_{n_z} g^{n_z} |S(k)|^{1/2}$ *is the* online-calculated (*validation*) *gate volume of the PDAF at time step* k.

[10] The measurement $z(k)$ is said to be a *validated measurement*, if it is inside a validation gate defined in (6.3).

Proof The cost function given in (6.29) can be rewritten as

$$\begin{aligned}
\mathscr{J}(k, P_{\mathrm{FA}}) &= \mathrm{E}\big[\mathrm{tr}\big\{\|x(k) - \hat{x}(k|k)\|^2\big\}\big|Z^{k-1}\big] \\
&= \mathrm{E}\big[\mathrm{tr}\big\{\big(x(k) - \hat{x}(k|k)\big)^T\big(x(k) - \hat{x}(k|k)\big)\big\}\big|Z^{k-1}\big] \\
&= \mathrm{E}\big[\mathrm{tr}\big\{\big(x(k) - \hat{x}(k|k)\big)\big(x(k) - \hat{x}(k|k)\big)^T\big\}\big|Z^{k-1}\big] \\
&= \mathrm{tr}\big\{\mathrm{E}\big[\big(x(k) - \hat{x}(k|k)\big)\big(x(k) - \hat{x}(k|k)\big)^T\big|Z^{k-1}\big]\big\} \\
&= \mathrm{tr}\big\{\mathrm{E}\big[\mathrm{E}\big[\big(x(k) - \hat{x}(k|k)\big)\big(x(k) - \hat{x}(k|k)\big)^T\big|Z^{k}\big]\big|Z^{k-1}\big]\big\} \\
&= \mathrm{tr}\big\{\mathrm{E}\big[P(k|k)|Z^{k-1}\big]\big\} \\
&\approx \mathrm{tr}\big\{\bar{P}_{\mathrm{MRE}}(k|k)\big\} \\
&= \mathrm{tr}\big\{P(k|k-1)\big\} - q_2\big(\lambda(k)V(k), P_D\big)\mathrm{tr}\big\{W(k)S(k)W^T(k)\big\},
\end{aligned}$$

where the first equality is due to the property that the trace of a scalar is itself, the third one is due the property that $\mathrm{tr}\{AB\} = \mathrm{tr}\{BA\}$, the fourth one is due to linearity of $\mathrm{tr}\{\cdot\}$ and $\mathrm{E}[\cdot]$ operators, and the fifth one follows from the *smoothing property* [9] of expectations. Note that $W(k)S(k)W^T(k) \geq 0$ implies $\mathrm{tr}\{W(k)S(k)W^T(k)\} \geq 0$, and $q_2(\lambda(k)V(k), P_D)$ is the only term that depends on P_{FA}. Hence the minimization of $\mathscr{J}(k, P_{\mathrm{FA}})$ can be achieved by maximizing $q_2(\lambda(k)V(k), P_D)$ over P_{FA}, which completes the proof. □

Remark 6.2 We experimentally observe that choosing any other scalar measures for the function $f_S[\cdot]$ from the set $\{|\cdot|, \|\cdot\|_1, \|\cdot\|_2, \|\cdot\|_\infty, \|\cdot\|_F\}$ results in the same optimization problem given in (6.30) where the elements of the set are the determinant, 1-norm (the largest column sum), 2-norm (the largest singular value), ∞-norm (the largest row sum) and Frobenius-norm of a matrix, respectively.

Due to mathematical intractability, the problem given in (6.30) was solved by utilizing some line search algorithms that require only the evaluation of the cost function (e.g., Golden-Section or Fibonacci Search methods) [22]. We call this scheme as **DYNAMIC-MRE-LS** in Fig. 6.2.

Lemma 6.2 (A Closed-Form Solution [5]) *An approximate closed-form solution for the MRE-based dynamic threshold optimization can be found for a special type*

of Neyman–Pearson detector under $\text{HOG}_{\text{I}}^{\text{SQL}}$ *assumption*[11] *as*

$$P_{\text{FA}}^{*}(k) = \begin{cases} [0.37 N_C(k)(\zeta - 1.57)]^{(1+\zeta)/(0.57-\zeta)} & \text{if } \zeta \geq 1.57 + 1/[0.37 N_C(k)], \\ 1 & \text{otherwise}, \end{cases} \tag{6.31}$$

where $N_C(k) \triangleq V(k)/V_C$ *is the number of resolution cells enclosed by the validation gate at time k of the PDAF.*

Proof A functional approximation for the IRF in (6.11) was proposed in [40] as

$$q_2\big(\lambda \bar{V}(k), P_D\big) \approx \hat{q}_2\big(\lambda \bar{V}(k), P_D\big) = \frac{0.997 P_D}{1 + 0.37 P_D^{-1.57} \lambda \bar{V}(k)}. \tag{6.32}$$

The ROC curve relation for the Neyman–Pearson (NP) detector under $\text{HOG}_{\text{I}}^{\text{SQL}}$ is given by [65]

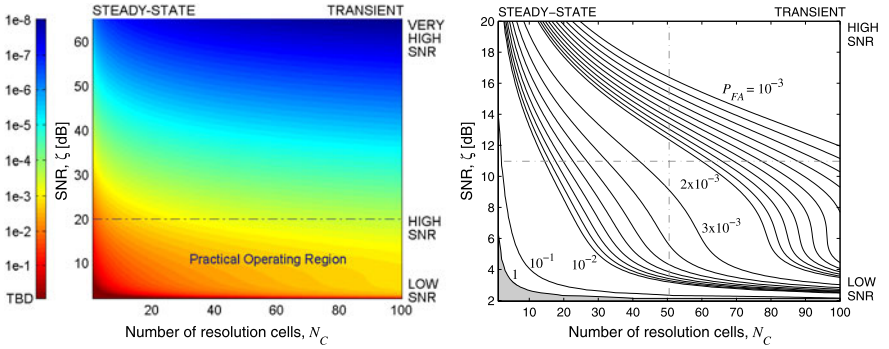
$$P_D = P_{\text{FA}}^{1/(1+\zeta)}. \tag{6.33}$$

Using (6.32) and (6.33) in solving the constraint optimization problem defined in (6.30), after some elaboration, results in the closed-form solution given in (6.31). A more detailed explanation can be found in [5]. □

We refer to this closed-form solution given in (6.31) as **DYNAMIC-MRE-CF** in Fig. 6.2. This expression gives some useful insights into dynamic detection threshold optimization. Consider, e.g., the plot of the optimal P_{FA} surface as a function of ζ and N_C which is illustrated in Fig. 6.5(a) where the third data dimension (optimal P_{FA} values) are represented by colors. Note that the optimization consistently suggests increasing P_{FA} when the SNR decreases or the filter goes from its transient operation to its steady-state operation.[12] Note also that, considering a practical operating region, where SNR values are below 20 dB, the optimization suggests considerably higher P_{FA} values than the ones used commonly in practice (i.e., between 10^{-8} and 10^{-4} [57]). Similar values like 10^{-8} are only suggested when the SNR is very high (>60 dB) and the gate volume is large, i.e., in the transient phase of the filter. This clearly shows that the practically chosen P_{FA} values are far from an optimal setting in terms of the overall radar system tracking performance. The main

[11] This covers *homogeneous* and *Gaussian* background detector noise, a Swerling-I target fluctuation and *square-law* detection scheme. In the radar detection theory, such assumptions are made frequently when obtaining the ROC curves for a specific detector [65]. We refer to this joint assumption shortly as $\text{HOG}_{\text{I}}^{\text{SQL}}$.

[12] It can be argued that a decreasing value of N_C, namely decreasing the number of resolution cells falling inside a validation gate, suggests that the gate volume (hence the Gaussian hyperellipse suggested by the filter covariance) is diminishing. This in turn suggests the convergence of the filter to its steady-state although this may not be guaranteed to be the correct state estimate. Conversely, by the same argument, a large value of N_C suggests a large gate volume, which in turn suggests that the filter is comparatively in its transient phase.



(a) (b)

Fig. 6.5 (**a**) The surface of optimal P_{FA} values as a function of ζ and N_C, (**b**) P_{FA} contours in the practical operating region. Note that the shaded area in (**b**) corresponds to applying no thresholding at all, where the whole system operates in *track-before-detect* (TBD) mode

reason for choosing considerably low P_{FA} values in practice is the computational limitations of the radar data processor in handling large numbers of false alarms. Although this may still be of considerable concern today, it is our belief that diminishing silicon prices and increasing computational power will enable performance optimal algorithms to be the choice over heuristic approaches. At low SNR values, the tracking performance gain of operating at these high but optimal false alarm values may be substantial as illustrated by the experimental results that follow.

A very interesting result of the proposed closed-form expression can also be observed from Fig. 6.5. Namely, the solution can be viewed as a generalization of the *track-before-detect* (TBD) approach suggested in the literature for very low SNR scenarios [13, 60]. Note that in some portion of the N_C–ζ plane P_{FA} is set to 1. This means that the optimal solution applies no thresholding on the raw radar signals, effectively making a seamless and automatic transition to the TBD approach, which is a degenerate case of the optimal setting given in (6.31).

6.3.2.2 HYCA-Based Dynamic Threshold Optimization

Inspired from the formulation for the MRE case by Gelfand [22], the dynamic threshold optimization problem for HYCA case is formulated in [2] as

$$\begin{aligned} P^*_{\text{FA}}(k) &= \arg\min_{P_{\text{FA}}} \left\{ \text{tr}\left\{ \bar{P}_{\text{HYCA}}(k|k) \right\} \right\} \\ \text{subject to} \quad P_D(k) &= f_{\text{ROC}}\left(P_{\text{FA}}(k), \zeta(k) \right) \quad \text{and} \quad 0 \le P_{\text{FA}}(k) \le 1. \end{aligned} \tag{6.34}$$

Lemma 6.3 *The optimization problem given in* (6.34)*, which is equivalent to the one given in* (6.28) *with the choices of* $f_S[\cdot] = \text{tr}\{\cdot\}$ *and* $\bar{P}_{\text{NSPP}}(k|k) = \bar{P}_{\text{HYCA}}(k|k)$,

can be reduced to

$$P_{\mathrm{FA}}^{*}(k) = \arg\max_{P_{\mathrm{FA}}} \left\{ \sum_{m_k=0}^{N} u_2\big(\lambda(k)V(k),\, P_D(k), m_k\big)\pi(m_k) \right\} \tag{6.35}$$

$$\textit{subject to} \quad P_D(k) = f_{\mathrm{ROC}}\big(P_{\mathrm{FA}}(k), \zeta(k)\big) \quad \textit{and} \quad 0 \le P_{\mathrm{FA}}(k) \le 1,$$

i.e., maximization of a weighted sum of information reduction factors for each possible values of m_k with weights

$$\pi(m_k) \triangleq \left[1 + P_D(k) P_G \left(\frac{m_k}{\lambda(k)V(k)} - 1 \right)\right] \mu_F\big(m_k; \lambda(k)V(k)\big) \tag{6.36}$$

where $u_2(\cdot,\cdot,\cdot)$ and $\mu_F(\cdot;\cdot)$ are the IRF given in (6.20) and the Poisson pmf defined in (6.4), respectively.

Proof The proof is skipped. □

The optimization problem given in (6.35) is solved using line search algorithms, e.g., the Fibonacci Search method in [2], which results in the scheme **DYNAMIC-HYCA-LS** in Fig. 6.2.

6.4 Simulations

We consider the problem of tracking a single target in clutter using a 2D radar. The target state vector is composed of the position and velocity components in East (ξ) and North (η) directions:

$$x(k) \triangleq \begin{bmatrix} \xi(k) & \dot{\xi}(k) & \eta(k) & \dot{\eta}(k) \end{bmatrix}^T. \tag{6.37}$$

The target performs a *coordinated turn* [9, pp. 467] with a constant and *known* turn rate:

$$F = \begin{bmatrix} 1 & \frac{\sin(\Omega T)}{\Omega} & 0 & -\frac{1-\cos(\Omega T)}{\Omega} \\ 0 & \cos(\Omega T) & 0 & -\sin(\Omega T) \\ 0 & \frac{1-\cos(\Omega T)}{\Omega} & 1 & \frac{\sin(\Omega T)}{\Omega} \\ 0 & \sin(\Omega T) & 0 & \cos(\Omega T) \end{bmatrix}, \qquad G = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix}, \tag{6.38}$$

where the turn rate is selected as $\Omega = 1$ deg/s and the sampling period is $T = 1$ s. Since we do not estimate the turn rate in the state vector, the state dynamics is linear. This is adopted to decouple the maneuver problem from the clutter problem on which our focus is. The process noise $v(k) \triangleq [\, v_\xi(k) \;\; v_\eta(k) \,]^T$ is a zero-mean white Gaussian random vector sequence with covariance matrix $Q = I_{2\times 2} q^2$ where $q = 0.1$ m/s^2 for all k and $I_{2\times 2}$ denotes the 2×2 identity matrix.

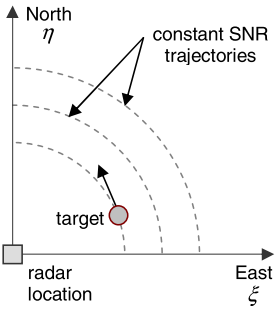


Fig. 6.6 Scenario geometry: Assumed constant SNR target trajectories and location of the radar. Measurements are taken from a radar located at the center of the target motion and assumed to consist of the position values in East and North directions, i.e., $z(k) \triangleq [\xi^m(k), \eta^m(k)]^T$. Note that such a geometry ensures the range of the target to be constant during the simulation

Scenario geometry is shown in Fig. 6.6. Such an artificial scenario is selected to have a constant SNR during the simulation, assuming that the SNR depends only on range as $\zeta(r) = C_\zeta / r^4$ where C_ζ is a constant representing all the other factors in the SNR equation and r is the range to the target. To determine C_ζ, ζ is assumed to be 50 for 5 km, which yields $C_\zeta = 3.125 \times 10^{16}$ m^4.

Measurements are assumed to be in *rectangular*[13] coordinates. The measurement noise $w(k) \triangleq [\, w_\xi(k) \; w_\eta(k) \,]^T$ is a zero-mean white Gaussian random vector sequence with covariance matrix

$$R = \begin{bmatrix} (\Delta r_\xi / \sqrt{12})^2 & 0 \\ 0 & (\Delta r_\eta / \sqrt{12})^2 \end{bmatrix} \tag{6.39}$$

for all k, where Δr_ξ and Δr_η are the range resolutions in East and North directions, respectively and taken as $\Delta r_\xi = \Delta r_\eta = 50$ m, which results in a resolution cell volume of $V_C = 2500$ m^2. Note that the covariance matrix given in (6.39) is the result of the assumption that the true measurement is uniformly distributed in the resolution cell [6, pp. 472]. The a priori information about the state, i.e., the mean $\hat{x}(0|0)$ and the covariance $P(0|0)$ of the initial state $x(0)$, is obtained by *two point differencing* [9, pp. 247].

6.4.1 Static Threshold Optimization Based on MRE and HYCA Algorithms

To apply static threshold optimization, TOC curves are first obtained for both MRE and HYCA algorithms, as illustrated in the flow diagram in Fig. 6.4. The parameter

[13]Normally, a 2D radar provides polar measurements. A rectangular resolution cell is adopted to have a linear measurement model.

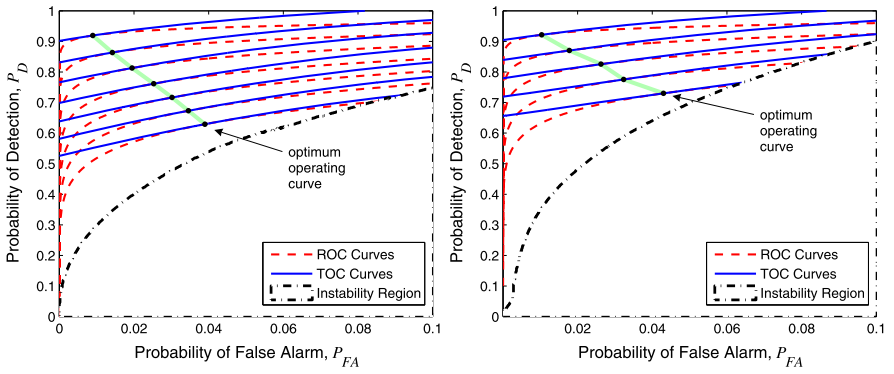


Fig. 6.7 STOP using TOC curves: (**a**) MRE case and (**b**) HYCA case. Note that the instability region for the HYCA case has a slightly larger area than that of MRE. One can fit a line equation for the optimum operating curves in both approaches. Then this equation, together with the ROC curve relation, determines the STOP curve which can be used to find the optimum operating point for an arbitrary SNR value

N of the HYCA algorithm is taken as 15. We run both recursions on a 500×500 regular P_{FA}–P_D grid. The borders of the grid are from 0 to 0.1 for P_{FA} and from 0 to 1 for P_D. Both recursions are run over each point in this grid until convergence. As mentioned before, the recursions do not converge to a steady-state covariance for some of the grid points, due to non-existence of the limit given in (6.27). This causes an *instability region* [20] as illustrated in Fig. 6.7.

We define the scalar performance function ($f_S[\cdot]$) as the steady-state position estimation error, i.e.,

$$f_S[\bar{P}_{\text{NSPP}}] \triangleq \sigma^{ss}_{\text{POS}} = \sqrt{\bar{P}^{11}_{\text{NSPP}} + \bar{P}^{33}_{\text{NSPP}}} \tag{6.40}$$

where $\bar{P}^{ii}_{\text{NSPP}}$ is the ith diagonal element of $\bar{P}_{\text{NSPP}}$. The TOC curves are obtained as the contours of the corresponding performance measure surface. The superimposition of these curves onto the ROC curves is shown in Fig. 6.7 where the functional form of the ROC curves is given by[14]

$$P_D = P_{\text{FA}}^{1/(1+\zeta)}. \tag{6.41}$$

Note that the collection of optimum operating points for different SNR values, consisting of the tangential points of TOC and ROC curves, are well-behaved. A line fitting works quite well and results in approximations

$$P_D = -9.523\,P_{\text{FA}} + 1.002 \quad \text{for MRE}, \tag{6.42}$$

$$P_D = -5.943\,P_{\text{FA}} + 0.979 \quad \text{for HYCA}. \tag{6.43}$$

[14]This is valid for a special case of a NP detector under $\text{HOG}^{\text{SQL}}_{\text{I}}$ assumption.

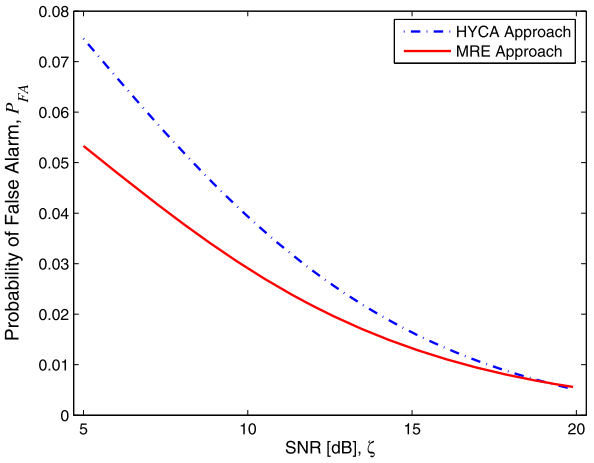


Fig. 6.8 STOP curves for MRE and HYCA approaches. Note that for a practical range of operating SNR values, the HYCA based approach consistently suggests a higher P_{FA}. Note that the STOP curves suggest that the desired false alarm probability of the detector should be readjusted according to SNR variations. This is different from the conventional approach in which the desired false alarm probabilities are fixed

Each of these optimum operating line equations can be combined with the ROC curve relation given in (6.41) to obtain a relation between P_{FA} and ζ as

$$P_{\text{FA}}^{1/(1+\zeta)} + 9.523\,P_{\text{FA}} - 1.002 = 0 \quad \text{for MRE,} \tag{6.44}$$

$$P_{\text{FA}}^{1/(1+\zeta)} + 5.943\,P_{\text{FA}} - 0.979 = 0 \quad \text{for HYCA.} \tag{6.45}$$

The numerical solution of these equations constitutes the *static threshold optimization* (STOP) curves given in Fig. 6.8. The resulting overall system target tracking performance corresponding to the two static methods is also of interest and is investigated in detail in subsequent experimental sections. At this point, however, it can be observed that for a practical range of operating SNR values, HYCA-based optimization consistently suggests a higher P_{FA}, and therefore results in more false detections for the radar processor to handle.[15]

6.4.2 *Experiment 1: Comparison with Heuristic Approaches*

In this experiment, we compare the tracking systems given in Table 6.1, each consisting of a PDA tracking filter and a Neyman–Pearson (NP) front-end detector.

In the first three systems, the detectors use conventional (i.e., heuristically selected) constant desired false alarm probabilities of $P_{\text{FA}} = 10^{-8}$, $P_{\text{FA}} = 10^{-6}$, and $P_{\text{FA}} = 10^{-4}$, which are the typical values used in practice [57]. On the other hand, the other systems utilize tracker-aware detectors for which the desired false alarm

[15]This is an important practical problem in the radar. Under excessive number of false detections, the radar may initiate lots of false tracks. This causes the radar to allocate its resources, e.g., dwell time, transmission power, unnecessarily and inefficiently.

Table 6.1 Compared tracking systems

System Name	Desired False Alarm Probability, P_{FA}^d
PDAKF-HEURISTIC-E8	$P_{\text{FA}}(k) = 10^{-8}$
PDAKF-HEURISTIC-E6	$P_{\text{FA}}(k) = 10^{-6}$
PDAKF-HEURISTIC-E4	$P_{\text{FA}}(k) = 10^{-4}$
PDAKF-STATIC-MRE [20]	$P_{\text{FA}}(k)$ is set as given in Eq. (6.44)
PDAKF-STATIC-HYCA [2]	$P_{\text{FA}}(k)$ is set as given in Eq. (6.45)
PDAKF-DYNAMIC-MRE [22]	$P_{\text{FA}}(k)$ is set as given in Eq. (6.30)
PDAKF-DYNAMIC-HYCA [2]	$P_{\text{FA}}(k)$ is set as given in Eq. (6.35)

probabilities are determined using MRE/HYCA-based static/dynamic threshold optimization. The dynamic optimizations given in (6.30) and (6.35) are solved using Fibonacci Search where we take the initial interval of uncertainty for P_{FA} as $\mathscr{I}_{P_{\text{FA}}} \triangleq [10^{-6}, 10^{-1}]$ and the maximum error tolerance[16] as $\Delta P_{\text{FA}} \triangleq 10^{-7}$.

We choose four different constant SNR scenarios of 5, 10, 15 and 20 dB. In each scenario, the target follows the corresponding constant SNR trajectory for 200 time steps as illustrated in Fig. 6.6. That is, SNR is time-invariant for each scenario, but from scenario to scenario we considered different constant SNR values. We have conducted 500 Monte Carlo runs for each scenario and compared the algorithms on a special performance plane where we consider two measures: the percentage of lost tracks (a transient performance indicator) and steady-state RMS position error (a steady-state performance indicator). The Track Loss Percentage (TLP) measure is defined as $TLP \triangleq N_{\text{TL}}/N_{\text{MC}} \times 100$ where N_{TL} is the number of Monte Carlo runs that result in *track loss*[17] and N_{MC} is the total number of Monte Carlo runs performed. The other measure, steady-state RMS position error, is obtained by ensemble averaging over only the "track-loss free" runs. The algorithm performances on this plane are given in Fig. 6.9 for each SNR scenario considered. In these plots, the lower left corner represents the ultimate performance, i.e., low TLP and low steady-state RMS position error. Note that the points (algorithm performances) get closer and eventually converge to the performance of the Kalman filter with perfect data association, when SNR increases. We may conclude that threshold optimization is less critical when the SNR is high, e.g., between 15 and 20 dB. On the other hand,

[16]Given an initial interval of uncertainty, $[a, b]$ and the number of function evaluations, N, the Fibonacci Search algorithm reduces the length of the uncertainty interval to $(b-a)/F_{N+1}$, where F_{N+1} is the $(N+1)$th number in the Fibonacci sequence $\{1, 1, 2, \ldots\}$. Therefore, given the number N, the length of the final uncertainty interval, so the maximum error in finding the extremum point, is determined. Here, we do the other way around. That is, we specify the maximum error tolerance that we are required to have at the end of the algorithm which in turn determines the minimum required number of function evaluations, N.

[17]We accept that the track is lost for the ith Monte Carlo run if $\varepsilon_{\text{POS}}^i > \rho$ where $\rho \triangleq \sqrt{\text{tr}\{R\}}$ is the measurement error level and $\varepsilon_{\text{POS}}^i$ is the average position estimation error for the ith run.

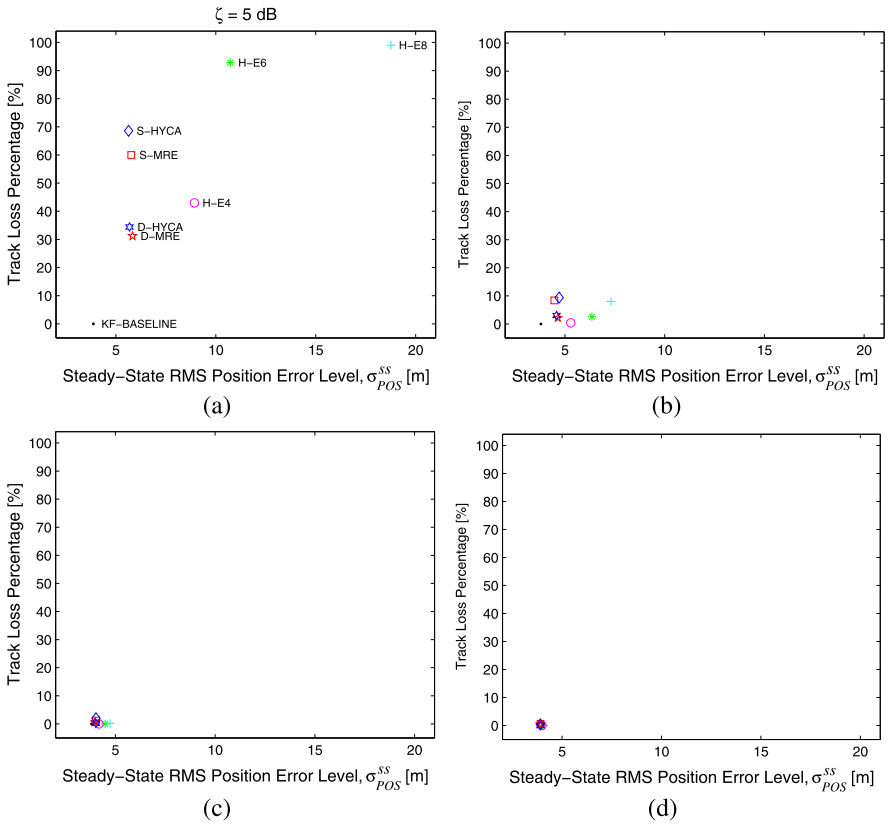


Fig. 6.9 Algorithms on the performance plane: The steady-state RMS position error versus TLP. Here, the prefixes "H", "S", and "D" correspond to "heuristic", "static", and "dynamic" approaches, respectively. Note that, as SNR increases, the performance of the algorithms gets closer to the best achievable performance point, the Kalman filter with perfect data association

in the lowest SNR case (see Fig. 6.9(a)), threshold optimization greatly improves the performance.

If we zoom in this most critical lowest SNR case and consider other possible heuristic approaches whose desired false alarm probabilities are ranging from $P_{\text{FA}} = 10^{-8}$ to $P_{\text{FA}} = 10^{-1}$, we get the whole performance trajectory shown in Fig. 6.10. As shown, the DTOP schemes are the only algorithms whose performances are located nearly at the lower left corner of the trade-off plane. Although the static schemes have low steady-state RMS position error level, they may not provide low TLP as shown in the figure. The dynamic optimization schemes have better transient characteristic as compared to static ones. This is an important aspect of DTOP schemes and leads to improved *track loss performance*. From the practical point of view, it can be argued that having a lower low track loss percentage is more critical than having a lower steady-state position error. So in that respect, DTOP schemes seem to be more viable solutions in practice.

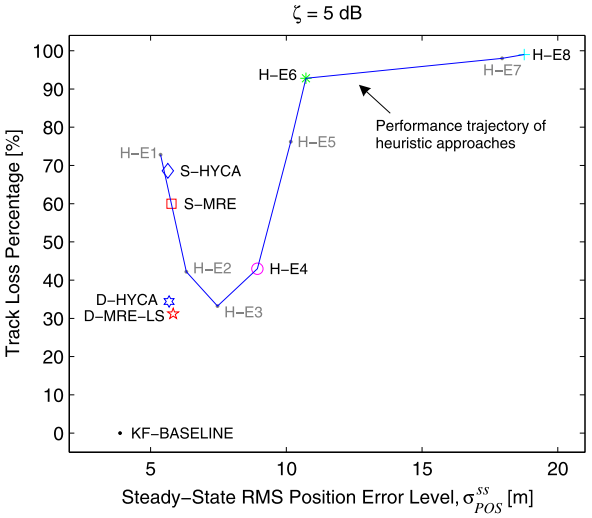


Fig. 6.10 The performance of the algorithms in 5 dB case. At some portions of the performance trajectory of heuristic approaches, there are sharp bends, resulting in a significant change in the performance. Therefore, in general one cannot guarantee a reliable performance with a heuristic approach

Initial motivation to formulate the static and dynamic optimization methods based on HYCA is due to its promise in modeling and therefore improving transient behavior of the overall system better than MRE. In experimental results, however, we could not observe this improvement to the extent hoped for. Formulating and solving the detector threshold optimization problem based on either MRE or HYCA does not result in a big difference in system performance.

The variation of the average operating P_{FA} values suggested by the algorithms over the considered SNR range is shown in Fig. 6.11. A common observation is that, on the average, optimization algorithm suggested P_{FA} values are all decreased when SNR increases. This is consistent with the expectation and with the *track-before-*

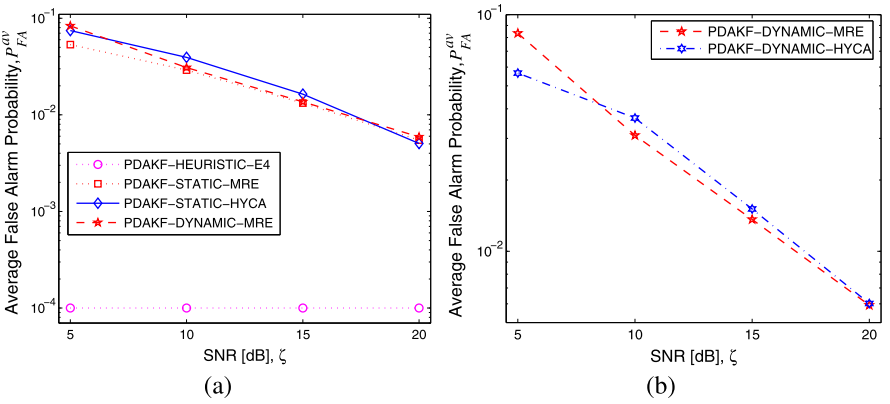


Fig. 6.11 (**a**) The average operating P_{FA} values suggested by the static and dynamic optimization schemes and (**b**) those of the DTOP schemes as a function of SNR

Table 6.2 Compared tracking systems

System Name	Desired False Alarm Probability, P_{FA}^{d}
PDAF-OP-GOL [22]	$P_{\text{FA}}(k) = \arg\max_{P_{\text{FA}}} q_2(\lambda V(k), P_D)$ (Golden-Section Search)
PDAF-OP-FIB [22]	$P_{\text{FA}}(k) = \arg\max_{P_{\text{FA}}} q_2(\lambda V(k), P_D)$ (Fibonacci Search)
PDAF-OP	$P_{\text{FA}}(k)$ is set as given in Eq. (6.31)

detect (TBD) literature which proposes using no thresholding under very low SNR [13, 60].

An important aspect of practical applicability of DTOP schemes is their computational complexities. The dynamic approaches are computationally much more expensive than STOP approaches. This is mainly due to the iterative line search algorithm involved. In the next experiment, we consider, in particular, MRE-based dynamic threshold optimization problem and compare the line search based algorithms proposed in [22] with the approximate closed-form solution of [5].

6.4.3 Experiment 2: Comparison of MRE-Based DTOP Schemes

The objective of this second experiment is to now make a comparison between online optimal threshold selection methods only, in particular between the iterative line-search based methods used in [22] and the approximate closed-form solution proposed in [5].

We compare both the overall tracking performance and the computational complexity of three optimal tracking systems given in Table 6.2. Each tracking system consists of a PDA tracking filter and an NP front-end detector. In each system, the optimal P_{FA} value found by threshold optimization is fed to the detector at every time step. The main differences between these tracking systems are their solution methodology in solving the optimization problem defined in (6.30). For example, PDAF-OP-GOL [22] and PDAF-OP-FIB [22] solve this problem using the Golden-Section and Fibonacci Search methods, respectively. On the other hand, PDAF-OP [5] solves the problem approximately in closed-form as given in (6.31). An example comparative variation of the true cost function $q_2(\lambda V(k), P_D)$, where $\lambda \triangleq P_{\text{FA}}/V_C$, and its functional approximation $\hat{q}_2(\lambda V(k), P_D)$ with respect to P_{FA} is illustrated in Fig. 6.12. Here, both cost functions are evaluated on the NP detector ROC curve given in (6.41) and for $V = 10V_C$ and $\zeta = 10$ dB values. Note that the true cost function $q_2(\lambda V(k), P_D)$ is *unimodal* in the P_{FA} range shown in Fig. 6.12. Therefore, both line search algorithms converge to the global optimum of this function. Note also that the global optimum found by the closed-form solution slightly differs from the one of the actual function. At this point, we seek answers to the following two questions:

- Is there any notable loss of tracking performance by solving the approximate optimization problem rather than the original one?

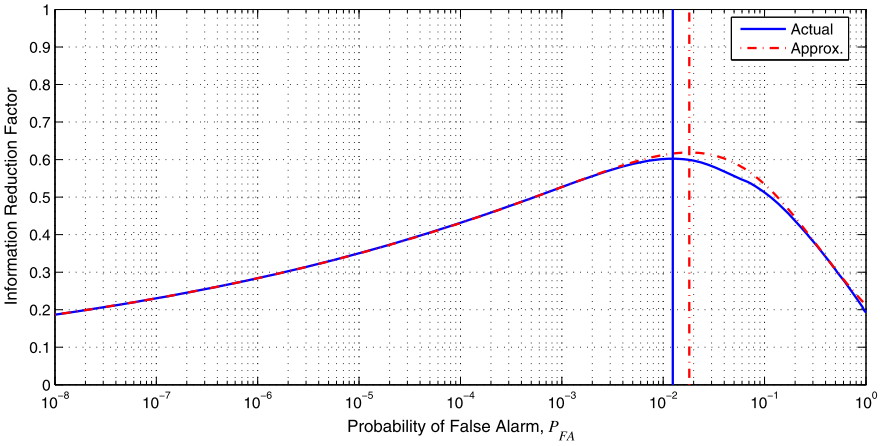


Fig. 6.12 Variation of the IRF $q_2(P_{\text{FA}}N_C(k), P_D)$ and its functional approximation $\hat{q}_2(P_{\text{FA}}N_C(k), P_D)$ over a NP detector ROC curve for $N_C = 10$ and $\zeta = 10$ dB case

- If no such loss is observed, what is the amount of computational gain obtained by using a closed-form solution to the approximated problem as compared to iterative solution of the original problem?

To answer these questions, we configure an experiment that is described in this section. First, we note that the variation of both cost functions given in Fig. 6.12 is best viewed in log-scale for the P_{FA} axis. The use of a linear scale in P_{FA} squeezes the global optimum peaks in a very small interval so that they cannot be observed. The same effect would also slow down the line search algorithms, hence resulting in an unfair evaluation. This observation leads us to operate the two line search algorithms not in linear but in logarithmic scale. That is, it is much more efficient to search for the global optimum over the exponent term of P_{FA}. Therefore, for the iterative methods based on the Golden Section and Fibonacci Search, we take the initial interval of uncertainty for the exponent of P_{FA} as $\mathscr{I}_e = [-8, 0]$ and the tolerance of the search stop condition on the exponent as $\Delta_e = 0.01$. We consider 5 constant SNR scenarios of 5, 8, 11, 14, 17 dB and perform 500 Monte Carlo runs for each scenario. The simulation results are given in Fig. 6.13. Considering RMS position errors, which are obtained from the track-loss free Monte Carlo runs, all the filters exhibit similar performance. In terms of TLP, the performances are again very close. Hence, as an answer to the first question mentioned above, we conclude that the proposed closed-form solution does not imply a tracking performance penalty. The superiority of PDAF-OP becomes obvious when we consider the execution times (all simulations being run on the same hardware with all auxiliary processes killed) given in Fig. 6.13(d). Note that PDAF-OP, which uses closed-form adaptation scheme in detector threshold optimization, clearly outperforms PDAF-OP-GOL and PDAF-OP-FIB approaches, which use one dimensional search algorithms for the same task. The computational gains are significant.

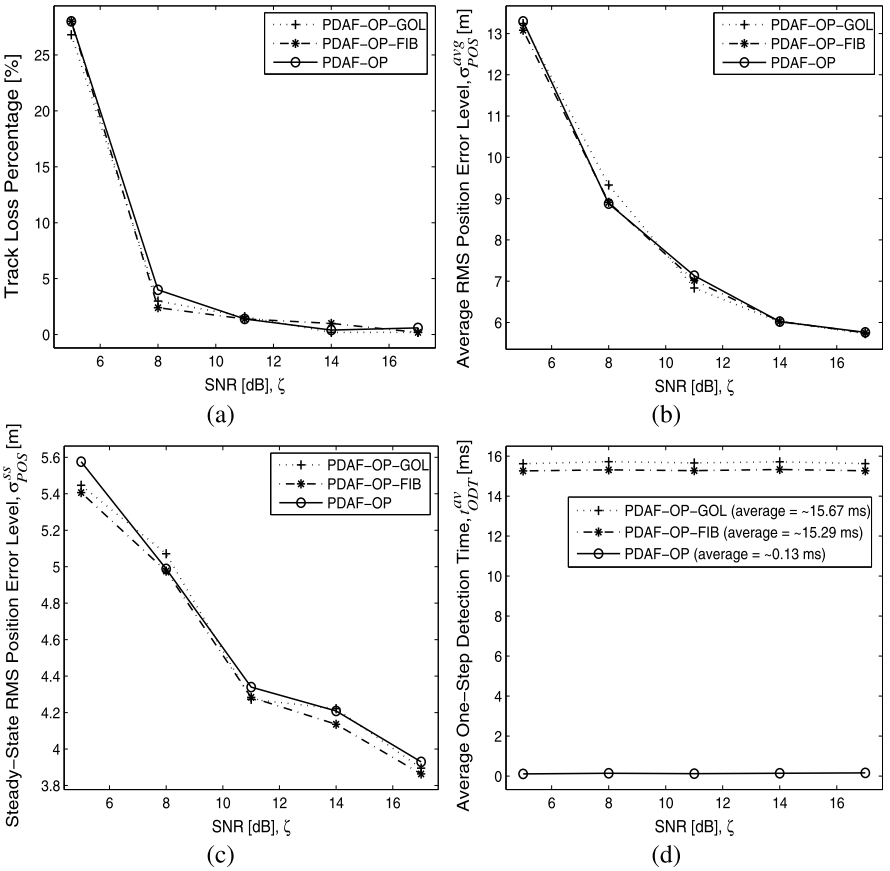


Fig. 6.13 Comparison of the tracking systems for $\mathscr{I}_e = [-8, 0]$ and $\Delta_e = 0.01$. Although there is no notable difference in terms of RMS position error and TLP, the closed-form approach (PDAF-OP) is computationally much more efficient than the iterative approaches

For the iterative search methods, the computation time critically depends on number of function evaluations which is determined by $\mathscr{I}_e$ and Δ_e. For this particular selection of these parameters, the gain in terms of computational power is (approximately) as much as 115 times. To obtain a fairer comparison, we attempted to observe the trade-off between the tracking performance and computation times by changing the stopping tolerance parameter. Since we have already observed that the proper setting of P_{FA} becomes more crucial for low SNR conditions, a very low SNR scenario of (5 dB) is chosen for this comparison and the same experiment is repeated for $\mathscr{I}_e = [-8, 0]$ and stopping tolerance values of $\Delta_e = \{0.1, 1, 2, 3, 4\}$. The corresponding computational gains are approximately $\{85, 66, 64\}$ times for $\Delta_e = \{0.1, 1, 2\}$. For the other tolerance values $\Delta_e = \{3, 4\}$, the line search algorithms produce unacceptable results and the optimization fails. Therefore, we conclude that even for the crude tolerance case of $\Delta_e = 2$, the closed-form solution is

approximately 64 times more efficient than the iterative search algorithms. Note that a very crude stopping tolerance usually obtains an arbitrary point near the mid-point of the initial search interval rather than the true maximum point. This behavior can be expected to impact performance especially when the maximum lies close to the interval boundaries, such as the one illustrated in Fig. 6.12.

6.5 Conclusion and Future Directions

In this chapter, a theoretical and experimental framework has been presented for joint optimization of detector and tracker subsystems. This exciting problem, which can be considered within the context of a more general topic of *cognitive radar*, is called *tracker-aware detection threshold optimization* by the authors.

The problem and possible improvements are presented in non-maneuvering target tracking domain, particularly for the PDAF case. There were some prior attempts [20, 22, 48] to this problem, but a comparison of these solution schemes in a unified framework was not available in the literature until [2]. After categorizing these attempts as *static* and *dynamic* optimization schemes, a comprehensive comparison of these schemes is presented in a unified experimental and theoretical framework. Contrary to expectations, the results concluded that only marginal gains can be achieved by HYCA-based approaches as compared to MRE-based ones. Moreover, it is observed that there exists a trade-off between having low track loss percentage (TLP) and having low steady-state tracking error.[18] The dynamic schemes are found to be well-located in this trade-off by providing considerably low TLP and low level of steady-state estimation error. The cost paid for this achievement is the computational complexity. An approximate closed-form solution proposed in [5] partially overcomes this issue for the MRE-based dynamic optimization scheme. Although the solution is given only for the Neyman–Pearson detector case, in authors' opinion, it can be applied for other practically used detection systems, which mimic asymptotically the NP detector, such as the Cell Averaging Constant False Alarm Rate (CA-CFAR) system. Apart from its computational efficiency, the proposed closed-form solution also gives some useful insights into the problem. The most important implication is that it provides a theoretical lower bound on the detection SNR concerning when the whole tracking system should be switched to the track before detect (TBD) mode.

For the future research directions, the NSPP algorithms for other tracking filters, such as NNF or SNF, which are already available in the literature can be applied to the detection threshold optimization problem. Furthermore, new NSPP algorithms can also be proposed. Especially, the one for the IMM-PDAF for tracking maneuvering targets deserve some attention.

An interesting and also a challenging research direction is for the case of tracking multiple targets. When two tracks corresponding to two targets overlap, optimal determination of the detection threshold seems to be a challenging problem.

[18] These measures can be seen also as transient vs. steady-state performance, respectively.

Another important point to note is the problem of an unknown SNR situation. In all the detection optimization schemes, SNR is assumed to be known, but this is clearly not the case in practice. Therefore, SNR should be estimated. In this case, the threshold optimization problem is coupled with the online SNR estimation which brings extra challenges to the problem.

References

1. Anderson, B.D.O., Moore, J.B.: Optimal Filtering. Prentice-Hall, Englewood Cliffs (1979)
2. Aslan, M., Saranlı, A.: Threshold optimization for tracking a nonmaneuvering target. IEEE Trans. Aerosp. Electron. Syst. **47**(4), 2844–2859 (2011)
3. Aslan, M., Saranlı, A.: A tracker-aware detector threshold optimization formulation for tracking maneuvering targets in clutter. Signal Process. **91**(9), 2213–2221 (2011)
4. Aslan, M., Saranlı, A., Baykal, B.: Optimal tracker-aware radar detector threshold adaptation: a closed-form solution. In: International Conference on Information Fusion, pp. 470–477 (2008)
5. Aslan, M., Saranlı, A., Baykal, B.: Tracker-aware adaptive detection: an efficient closed-form solution for the Neyman–Pearson case. Digit. Signal Process. **20**(5), 1468–1481 (2010)
6. Bar-Shalom, Y., Li, X.R.: Multitarget–Multisensor Tracking: Principles and Techniques. YBS Publishing, Storrs (1995)
7. Bar-Shalom, Y., Tse, E.: Tracking in a cluttered environment with probabilistic data association. Automatica **11**, 451–460 (1975)
8. Bar-Shalom, Y., Campo, L.J., Luh, P.B.: From receiver operating characteristic to system operating characteristic: evaluation of a track formation system. IEEE Trans. Autom. Control **35**(2), 172–179 (1990)
9. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. Willey, New York (2001)
10. Bar-Shalom, Y., Willett, P.K., Tian, X.: Tracking and Data Fusion: A Handbook of Algorithms. YBS Publishing, Storrs (2011)
11. Bazaraa, M., Sherali, H., Shetty, C.: Nonlinear Programming Theory and Algorithms. Wiley, New York (2006)
12. Blackman, S.S., Popoli, R.: Design and Analysis of Modern Tracking Systems. Artech House, Norwood (1999)
13. Boers, Y., Driessen, J.N.: Particle filter based detection for tracking. In: Proceedings of the American Control Conference, Arlington, VA, vol. 6, pp. 4393–4397 (2001)
14. Boers, Y., Driessen, H.: Modified Riccati equation and its application to target tracking. IEE Proc. Radar Sonar Navig. **153**, 7–12 (2006)
15. Boers, Y., Driessen, H.: Results on the modified Riccati equation: target tracking applications. IEEE Trans. Aerosp. Electron. Syst. **42**, 379–384 (2006)
16. Bréhard, T., Cadre, J.P.L.: Closed-form posterior Cramér–Rao bounds for bearings-only tracking. IEEE Trans. Aerosp. Electron. Syst. **42**(4), 1198–1223 (2006)
17. Chavali, P., Nehorai, A.: Scheduling and power allocation in a cognitive radar network for multiple-target tracking. IEEE Trans. Signal Process. **60**(2), 715–729 (2012)
18. Daum, F.E.: Bounds on performance for multiple target tracking. IEEE Trans. Autom. Control **35**(4), 443–446 (1990)
19. Farina, A., Ristic, B., Timmoneri, L.: Cramér–Rao bound for nonlinear filtering with $P_d < 1$ and its application to target tracking. IEEE Trans. Signal Process. **50**, 1916–1924 (2002)
20. Fortmann, T.E., Bar-Shalom, Y., Scheffe, M., Gelfand, S.: Detection thresholds for tracking in clutter—a connection between estimation and signal processing. IEEE Trans. Autom. Control **AC-30**(3), 221–229 (1985)

21. Frei, C.W.: A comparison of parallel and sequential implementations of a multisensor multitarget tracking algorithm. Master's thesis, Northwestern University, Evanston, IL (1995)
22. Gelfand, S.B., Fortmann, T.E., Bar-Shalom, Y.: Adaptive detection threshold optimization for tracking in clutter. IEEE Trans. Aerosp. Electron. Syst. **32**, 514–523 (1996)
23. Goodman, N.A., Venkata, P.R., Neifeld, M.A.: Adaptive waveform design and sequential hypothesis testing for target recognition with active sensors. IEEE J. Sel. Top. Signal Process. **1**, 105–113 (2007)
24. Haykin, S.: Cognitive radar networks. In: 1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, pp. 1–3 (2005)
25. Haykin, S.: Cognitive radar: a way of the future. IEEE Signal Process. Mag. **23**, 30–40 (2006)
26. Haykin, S.: Optimal waveform design for cognitive radar. In: 42nd Asilomar Conference on Signals, Systems and Computers, pp. 3–7 (2008)
27. Haykin, S.: Cognitive tracking radar. In: IEEE Radar Conference, pp. 1467–1470 (2010)
28. Hernandez, M., Ristic, B., Farina, A., Timmoneri, L.: A comparison of two Cramér–Rao bounds for nonlinear filtering with $P_d < 1$. IEEE Trans. Signal Process. **52**(9), 2361–2370 (2004)
29. Hernandez, M.L., Farina, A., Ristic, B.: PCRLB for tracking in cluttered environments: measurement sequence conditioning approach. IEEE Trans. Aerosp. Electron. Syst. **42**(2), 680–704 (2006)
30. Hernandez, M.L., Ristic, B., Farina, A., Sathyan, T., Kirubarajan, T.: Performance measure for Markovian switching systems using best-fitting Gaussian distributions. IEEE Trans. Aerosp. Electron. Syst. **44**(2), 724–747 (2008)
31. Hong, S.M., Shin, H.S.: An analytic approximation of information reduction factor for performance prediction of PDA tracking. In: Proceedings of the 41st SICE Annual Conference, vol. 3, pp. 1447–1449 (2002)
32. Hong, S.M., Evans, R.J., Shin, H.S.: Optimization of waveform and detection threshold for range and range-rate tracking in clutter. IEEE Trans. Aerosp. Electron. Syst. **41**, 17–33 (2005)
33. Hue, C., Cadre, J.P.L., Pérez, P.: Posterior Cramer–Rao bounds for multi-target tracking. IEEE Trans. Aerosp. Electron. Syst. **42**(1), 37–49 (2006)
34. Jauffret, C., Bar-Shalom, Y.: Track formation with bearing and frequency measurements in clutter. IEEE Trans. Aerosp. Electron. Syst. **26**(6), 999–1010 (1990)
35. Jung, Y.H., Hong, S.M.: Modeling and parameter optimization of agile beam radar tracking. IEEE Trans. Aerosp. Electron. Syst. **39**(1), 13–33 (2003)
36. Kalandros, M., Pao, L.Y.: Multisensor covariance control strategies for reducing bias effects in interacting target scenarios. IEEE Trans. Aerosp. Electron. Syst. **41**(1), 153–173 (2005)
37. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. ASME J. Basic Eng. **82**, 34–45 (1960)
38. Kerr, T.H.: Status of CR-like lower bounds for nonlinear filtering. IEEE Trans. Aerosp. Electron. Syst. **25**(5), 590–601 (1989)
39. Kershaw, D.J., Evans, R.J.: Optimal waveform selection for tracking systems. IEEE Trans. Inf. Theory **40**(5), 1536–1550 (1994)
40. Kershaw, D.J., Evans, R.J.: A contribution to performance prediction for probabilistic data association tracking filters. IEEE Trans. Aerosp. Electron. Syst. **32**(3), 1143–1148 (1996)
41. Kershaw, D.J., Evans, R.J.: Waveform selective probabilistic data association. IEEE Trans. Aerosp. Electron. Syst. **33**, 1180–1188 (1997)
42. Kumar, P.R., Varaiya, P.: Stochastic Systems: Estimation, Identification and Adaptive Control. Prentice-Hall, Inc., Englewood Cliffs (1986)
43. Leven, W.F.: Approximate Cramér–Rao bounds for multiple target tracking. Ph.D. thesis, Georgia Institute of Technology (2006)
44. Li, X.R.: Tracking in clutter with strongest neighbor measurements—Part I: theoretical analysis. IEEE Trans. Aerosp. Electron. Syst. **43**(11), 1560–1578 (1998)
45. Li, X.R.: Multitarget–Multisensor Tracking: Applications and Advances, vol. III, pp. 499–567. Artech House, Inc., Norwood (2000). Chap. 10

46. Li, X.R., Bar-Shalom, Y.: Stability evaluation and track life of the PDAF for tracking in clutter. IEEE Trans. Autom. Control **36**, 588–602 (1991)
47. Li, X.R., Bar-Shalom, Y.: Performance prediction of the interacting multiple model algorithm. IEEE Trans. Aerosp. Electron. Syst. **29**(3), 755–771 (1993)
48. Li, X.R., Bar-Shalom, Y.: Detection threshold selection for tracking performance optimization. IEEE Trans. Aerosp. Electron. Syst. **30**, 742–749 (1994)
49. Li, X.R., Bar-Shalom, Y.: A hybrid conditional averaging technique for performance prediction of algorithms with continuous and discrete uncertainties. In: Proceedings of the American Control Conference, Maryland, pp. 1530–1534 (1994)
50. Li, X.R., Bar-Shalom, Y.: Tracking in clutter with nearest neighbor filters: analysis and performance. IEEE Trans. Aerosp. Electron. Syst. **32**(3), 995–1010 (1996)
51. Mori, S., Chang, K.C., Chong, C.Y., Dunn, K.P.: Prediction of track purity and track accuracy in dense target environments. IEEE Trans. Autom. Control **40**(5), 953–959 (1995)
52. Pao, L.Y., Trailović, L.: The optimal order of processing sensor information in sequential multisensor fusion algorithms. IEEE Trans. Autom. Control **45**(8), 1532–1536 (2000)
53. Rago, C., Willett, P., Bar-Shalom, Y.: Detection-tracking performance with combined waveforms. IEEE Trans. Aerosp. Electron. Syst. **34**(2), 612–624 (1998)
54. Ramdaras, U.D., Absill, F.G.J.: Sensor selection: the modified Riccati equation approach compared with other selection schemes. In: International Conference on Information Fusion, pp. 1–6 (2007)
55. Ramdaras, U.D., Absill, F.G.J.: Target tracking in sensor networks: criteria for sensor selection. In: IEEE Radar Conference, pp. 192–196 (2007)
56. Rapoport, I., Oshman, Y.: A Cramér–Rao-type estimation lower bound for systems with measurement faults. IEEE Trans. Autom. Control **50**(50), 1234–1245 (2005)
57. Richards, M.A.: Fundamentals of Radar Signal Processing. McGraw-Hill, New York (2005)
58. Ristic, B., Farina, A., Hernandez, M.: Cramér–Rao lower bound for tracking multiple targets. IEE Proc. Radar Sonar Navig. **151**, 129–134 (2004)
59. Rogers, S.R.: Diffusion analysis of track loss in clutter. IEEE Trans. Aerosp. Electron. Syst. **27**(2), 380–387 (1991)
60. Salmond, D.J., Birch, H.: A particle filter for track-before-detect. In: Proceedings of the American Control Conference, Arlington, VA, vol. 5, pp. 3755–3760 (2001)
61. Sinopoli, B., Schenato, L., Franceschetti, M., Poola, K., Jordan, M.I., Sastry, S.S.: Kalman filtering with intermittent observations. IEEE Trans. Autom. Control **49**, 1453–1464 (2004)
62. Sira, S.P., Papandreou-Suppappola, A., Morrell, D.: Dynamic configuration of time-varying waveforms for agile sensing and tracking in clutter. IEEE Trans. Signal Process. **55**(7), 3207–3217 (2007)
63. Taylor, J.H.: The Cramér–Rao estimation error lower bound computation for deterministic nonlinear systems. IEEE Trans. Autom. Control **AC-24**(2), 343–344 (1979)
64. Tichavský, P., Muravchik, C.H., Nehorai, A.: Posterior Cramér–Rao bounds for discrete-time nonlinear filtering. IEEE Trans. Signal Process. **46**(5), 1386–1396 (1998)
65. Varshney, P.K.: Distributed Detection and Data Fusion. Springer, New York (1997)
66. Wang, J.T., Wang, H.Q., Qin, Y.L., Zhuang, Z.W.: Efficient adaptive detection threshold optimization for tracking maneuvering targets in clutter. Prog. Electromagn. Res. B **41**, 357–375 (2012)
67. Willett, P., Niu, R., Bar-Shalom, Y.: Multitarget–Multisensor Tracking: Applications and Advances, vol. III, pp. 459–498. Artech House, Inc., Norwood (2000). Chap. 9
68. Willett, P., Niu, R., Bar-Shalom, Y.: Integration of Bayes detection with target tracking. IEEE Trans. Signal Process. **49**(1), 17–29 (2001)
69. Xu, B., Wu, Z., Wang, Z.: On the Cramér–Rao lower bound for biased bearings-only maneuvering target tracking. Signal Process. **87**(12), 3175–3189 (2007)
70. Zhang, X., Willet, P., Bar-Shalom, Y.: Dynamic Cramér–Rao bound for target tracking in clutter. IEEE Trans. Aerosp. Electron. Syst. **41**(4), 1154–1167 (2005)

Chapter 7
Iterative Design of FIR Filters

Bipul Luitel and Ganesh Kumar Venayagamoorthy

Abstract This chapter presents the iterative design of finite impulse response (FIR) filters using particle swarm optimization with a quantum infusion (PSO-QI) algorithm. Filter design, in this work, is formulated as a parameter optimization problem using population-based stochastic methods; and hence, it is iterative. PSO-QI is a hybrid algorithm combining PSO and quantum-behaved PSO. PSO-QI combines the best features of these individual algorithms. Therefore, the design specification for FIR filters can be satisfied more accurately. Two methods of evaluating the performance of the algorithm (cost function) are implemented. Minimizing the mean squared error between the actual and the ideal filter response is one approach to performance evaluation. The second approach involves minimizing the mean squared error between the ripples in the passband and the stopband of the designed filter and the desired filter specification. The results presented show that filters designed using PSO-QI most closely match the design specification, and their performance is more consistent when compared with other evolutionary algorithms. The results are compared with the constrained least squares method of filter design.

7.1 Introduction

Digital filters suppress the unwanted parts of a signal, such as noise, and extract the important parts, such as underlying components within a frequency range. Consequently, they have been applied in communication for noise reduction, audio/video signal enhancement, etc. Digital filters are an important component of digital signal processing and are used in a wide range of modern applications including, but not limited to, telecommunications, acoustics, biometrics, biomedical science, and speech and image processing.

Traditionally, techniques such as the tables method [7], windowing method [1], frequency sampling method, and best uniform approximation method have been

B. Luitel (✉) · G.K. Venayagamoorthy
Real-Time Power and Intelligent Systems (RTPIS) Lab, Clemson University, Clemson, SC, USA
e-mail: iambipul@ieee.org

G.K. Venayagamoorthy
e-mail: gkumar@ieee.org

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_7,

used for the design of digital filters [6]. The simplest of these is the windowing method [1]. In this method, the ideal impulse response is multiplied with a window function. Various kinds of window functions (Butterworth, Chebyshev, Kaiser, etc.) can be used depending on the requirements of the ripples on the passband and stopband, the stopband attenuation and the transition width. These various windows limit the infinite length impulse response of the ideal filter into a finite window to design an actual response. However, windowing methods do not allow sufficient control of the frequency response in the various frequency bands and other filter parameters, such as transition width. The designer always has to compromise on one or the other of the design specifications. In [20], a mixed integer linear programming (MILP)-based approach for designing linear phase FIR filters is described. However, the solution time in MILP-based algorithms increases exponentially as the order of the filter increases. A branch-and-bound approach for designing hardware platform-efficient FIR filters is described in [2]. Traditional design techniques have design time and/or design parameter limitations. Many recent studies have investigated different techniques for designing digital filters [3, 11, 12, 16, 17]. Because population-based stochastic search methods have proved effective in multidimensional nonlinear environments, computational intelligence techniques, such as neural networks [8], genetic algorithms (GAs) [1, 7], immune algorithms [7], differential evolution (DE) [9, 18], and Particle Swarm Optimization (PSO) [7, 10], have been applied in the design of digital filters. Hybrid algorithms, which combine features of different algorithms, or modified and mutation-based PSO algorithms, which perform better than classical PSO, such as Quantum-behaved PSO (QPSO) [5, 6], Differential Evolution PSO (DEPSO) [13], and craziness-based PSO [15], have also been applied for better parameter control and better approximation to the ideal filter [6]. These algorithms, because they are multidimensional optimization methods, can effectively consider the different constraints during filter design. Finite Impulse Response (FIR) filters do not have feedback as do Infinite Impulse Response (IIR) filters and hence are inherently stable. They can be easily designed as linear phase filters. However, they require more memory and computational complexity than IIR filters to achieve the same performance.

Many modern applications already demand high computational speed and robust solutions. Hence, traditional techniques and many computational intelligence algorithms will also fail to meet future design requirements, which will prove even more stringent. It is important to reduce the number of coefficients and still try to meet other design requirements when it comes to implementing the digital filter in hardware. Therefore, algorithms that have better convergence, that can perform more consistently, and that can design filters with better frequency responses for fewer coefficients are more likely to be applied in resource-constrained and performance-critical applications. This chapter presents the application of PSO-QI [14], which shows such a potential, for digital FIR filter design. Population based optimization algorithms such as PSO-QI are probabilistic and not deterministic, so they require multiple iterations for convergence. Digital filter design, as explained in this chapter, is also a form of a parameter (coefficients) optimization process and hence is iterative. The major contributions of this work are as follows:

- Iterative design of lowpass, highpass, bandpass, and bandstop FIR filters using swarm, evolutionary, and quantum-infused hybrid algorithms.
- Comparison of the performance of PSO-QI against other methods using two types of cost functions.

7.2 Particle Swarm Optimization with Quantum Infusion

PSO-QI is a hybrid algorithm that uses the quantum principle from QPSO to create a new offspring in PSO. After the positions and velocities of the particles are updated using standard PSO equations, a randomly-chosen particle from PSO's *pbest* (the previous particle position giving the best fitness value) population is utilized to carry out the quantum operation, thus creating offspring by mutating the *gbest* (the best particle among all the particles in the swarm). The fitness of the offspring is evaluated, and the offspring replaces the *gbest* only if its fitness is better. This ensures that the fitness of the *gbest* is equal to or better than its fitness in the previous iteration. Thus, it is improved and nears the best solution over iterations.

According to the uncertainty principle, the position and velocity of a particle in the quantum world cannot be determined simultaneously. Thus, QPSO differs from standard PSO mainly because exact values of x and v cannot be determined. Hence, the probability of finding a particle at a particular position in the quantum search space is mapped into its actual position in the solution space by a technique called "collapsing." In Quantum Delta-Potential-Well-based PSO (QDPSO) [19], a delta potential well-based probability density function is used to avoid explosion and help the particles converge. By using Monte Carlo Simulation [19], the position equation in QDPSO is given by (7.1):

$$x(k) = J(k) \pm \frac{L(k)}{2} \ln(1/u) \tag{7.1}$$

where u is a uniform random number in the interval [0, 1]. The particle's local attractor point J has coordinates given by the following equation:

$$J_d(k) = \alpha_1 P_{gd}(k) + \alpha_2 P_{id}(k) \tag{7.2}$$

where P_{id} is the ith *pbest* particle in the dth dimension, and P_{gd} is the dth dimension of the *gbest* particle obtained from PSO. L is length of the potential field given by:

$$L(k) = 2\beta \big| J(k) - x(k) \big|. \tag{7.3}$$

The parameter β is the only parameter of the algorithm. It is called the creativity coefficient and is responsible for the convergence speed of the particle.

The mean best position, *mbest*, is defined as:

$$mbest(k) = \frac{1}{S} \sum_{i=0}^{S} P_i(k) = \left(\frac{1}{S} \sum_{i=0}^{S} P_{i1}(k), \ldots, \frac{1}{S} \sum_{i=0}^{S} P_{iD}(k) \right) \tag{7.4}$$

where S is the size of the population, D is the number of dimensions, and P_i is the *pbest* position of each particle. In QPSO, J in (7.3) is replaced by *mbest* to form (7.5).

$$L(k) = 2\beta \left| mbest(k) - x(k) \right|. \tag{7.5}$$

Using (7.2), this can also be written as follows to show the mutation of *gbest*, where the addition or subtraction is carried out with 50 % probability:

$$x(k+1) = \alpha_1 P_{gd}(k) + \alpha_2 P_{id}(k) \pm \beta \left| mbest(k) - x(k) \right| \ln(1/u). \tag{7.6}$$

In PSO-QI, the position update equation (7.6) has been used to mutate the *gbest* particle obtained from PSO. The pseudocode for the PSO-QI algorithm is as follows:

Initialize position x, velocity v, and let $pbest = x$
repeat
  for $i = 1$ to *populationsize* **do**
    Evaluate fitness
    if fitness (i) < fitness $(pbest)$ **then**
      $pbest = x$ and $gbest = \min(pbest)$
    end if
    Update v and x using standard PSO equations
  end for
  Calculate *mbest* using (7.4)
  Select a random particle r
  for d from 1 to *dimensionsize* **do**
    $\alpha_1, \alpha_2 = rand(0, 1)$
    $J = (\alpha_1 * P_{rd} + \alpha_2 * P_{gd})/(\alpha_1 + \alpha_2)$
    $L = 2\beta * |mbest - x_{rd}|$ using (7.5)
    if $rand(0, 1) > 0.5$ **then**
      using (7.1)
      $offspring = J - \frac{L}{2} * \ln(1/u)$
    else
      $offspring = J + \frac{L}{2} * \ln(1/u)$
    end if
    if fitness $(offspring)$ < fitness$(gbest)$ **then**
      $gbest = offspring$
    end if
  end for
until termination criterion is met.

7.3 Digital FIR Filter

When the output of the filter at any given time depends only on the current inputs, that filter is called a non-recursive or FIR filter. FIR filters have only zeros in their

transfer function. Because the poles of FIR filters are located at the origin and lie within the unit circle, they are inherently stable. Also, FIR filters can be designed as linear phase filters, which makes them a better choice in phase-sensitive applications. FIR filters whose phase response is linear with respect to frequency are said to be linear phase. An FIR filter is linear phase if its coefficients are symmetric around the center coefficient. Delay through such filters is constant at all frequencies; hence, they do not cause phase distortion.

An FIR filter can be described by the transfer function:

$$H(z) = \sum_{i=0}^{N} a_i z^{-i}. \tag{7.7}$$

The parameters $a_0, a_1, a_2, \ldots, a_N$ appearing in (7.7) are called filter coefficients, and they determine the characteristics of a filter. Filter specifications, which are important in a filter design process, include the passband and stopband normalized frequencies (ω_p, ω_s), passband and stopband ripple (δ_p) and (δ_s), stopband attenuation and transition width. These specifications are satisfied by the filter coefficients in (7.7). In any filter design problem, some of these specifications are fixed while others are determined. In this chapter, swarm, evolutionary, quantum, and hybrid optimization algorithms are applied in order to obtain an actual filter response that comes as close as possible to the ideal response.

7.4 FIR Filter Design Using PSO-QI

From (7.7), the transfer function of the FIR filter can also be represented as:

$$H(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_N z^{-N}. \tag{7.8}$$

For (7.8), the numerator coefficient vector $a_0, a_1, a_2, \ldots, a_N$ is represented in N dimensions. In PSO-like algorithms, each particle is distributed in a D-dimensional search space, where $D = N$ for the FIR filter. The position of each particle in this D-dimensional search space represents the coefficients of the FIR filter's transfer function. During each iteration, these particles find a new position, which is the new set of coefficients. Using the new values of the coefficients, the performance of each particle is evaluated based on some predefined fitness function. The fitness is then used to improve the search during each iteration, and the result obtained after a certain number of iterations or after the error falls below a certain limit is considered the final result. The error between the filter response of the desired and approximate filters is given by (7.9):

$$E(\omega) = G(\omega)\left[H_d\left(e^{j\omega}\right) - H\left(e^{j\omega}\right)\right] \tag{7.9}$$

where $G(\omega)$ is the weighting function used to provide different weights for the approximate errors in different frequency bands, $H_d(e^{j\omega})$ is the frequency response of

the desired filter, and $H(e^{j\omega})$ is the frequency response of the approximate filter [7]. In this chapter, the first fitness function considers an ideal filter as the desired filter, and hence, $H_d(e^{j\omega}) = 1$ for the passband and $H_d(e^{j\omega}) = 0$ for the stopband. Now, the Mean Squared Error (MSE) between the desired (ideal) filter and the approximate filter is defined as I_1 (7.10):

$$I_1 = \frac{1}{T}\left[\sum_{k=1}^{T}\left(1 - \left|H\left(e^{j\omega_k}\right)\right|\right)^2_{\omega\in F_p} + \sum_{k=1}^{T}\left(\left|H\left(e^{j\omega_k}\right)\right|\right)^2_{\omega\in F_s}\right] \tag{7.10}$$

where T is the number of samples used to calculate the error, F_p is the set of passband frequencies, and F_s is the set of stopband frequencies. For a lowpass filter, $0 < F_p < \omega_p$ and $\omega_s < F_s < 1$; for a highpass filter, $0 < F_s < \omega_p$ and $\omega_s < F_p < 1$; for a bandpass filter, $\omega_l < F_p < \omega_u$ and $F_s = (0 < F < \omega_l) \cup (\omega_u < F < 1)$; and for a bandstop filter, $\omega_l < F_s < \omega_u$ and $F_p = (0 < F < \omega_l) \cup (\omega_u < F < 1)$, where ω_p and ω_s are passband and stopband normalized cutoff frequencies for the lowpass and highpass filters, and ω_l and ω_u are the lower and upper normalized cutoff frequencies for the bandpass and bandstop filters.

In another case study, a desired filter with a specified magnitude of ripples on the passband and the stopband is considered. For this case, the MSE between the difference in the desired and the approximate frequency response is considered and is defined as (7.11):

$$I_2 = \frac{1}{T}\left[\sum_{k=1}^{T}\left(\left|E(\omega_k)_{\omega\in F_p}\right| - \delta_p\right)^2 + \sum_{k=1}^{T}\left(\left|E(\omega_k)_{\omega\in F_s}\right| - \delta_s\right)\right] \tag{7.11}$$

where δ_p and δ_s are the ripples in the passband and stopband, respectively. The algorithms try to minimize this error and thus increase the fitness of the filter designed using the swarm, evolutionary and quantum algorithms. A flowchart showing the design of the FIR filter using PSO-QI is shown in Fig. 7.1.

7.5 Studies and Results

Two main cases have been studied for FIR filter design. Cases I and II represent the design of FIR filters using fitness functions given by (7.10) and (7.11), respectively. In each case, lowpass (LP), highpass (HP), bandpass (BP), and bandstop (BS) FIR filters are designed. The results are compared with the constrained least squares (CLS) method of filter design (using the 'fircls' function of MATLAB). Various parameters of the algorithms are shown in Table 7.1. The specifications of the filters are provided in Table 7.2. In these studies, the performance of PSO-QI is compared with PSO and DEPSO. The PSO parameters used here are based on the best parameters as reported in the literature [4]. Alternative values of PSO parameters are also tested for research purposes. However, an analysis of these results with respect to the PSO parameters is beyond the scope of this chapter.

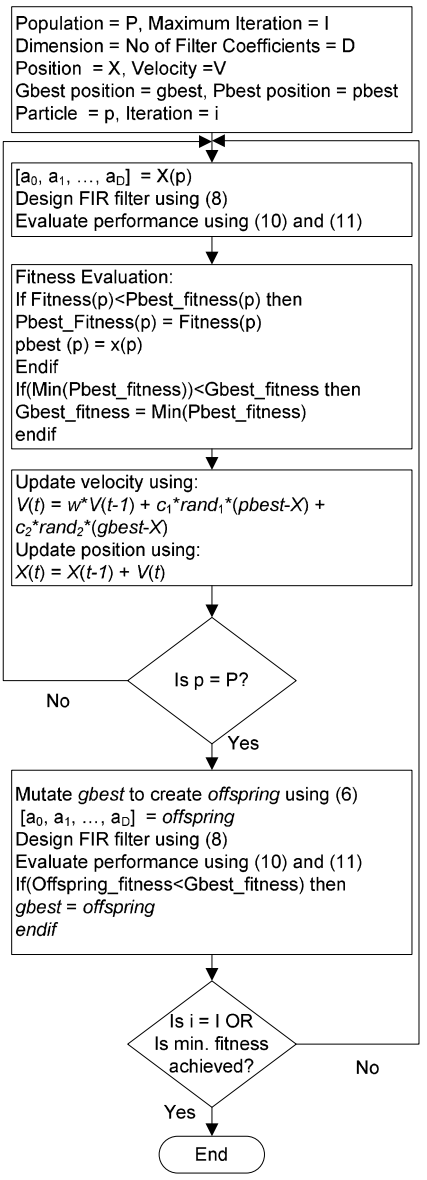


Fig. 7.1 Flowchart showing the design of an FIR filter using PSO-QI

Different kinds of FIR filters of order 20 are designed. Each algorithm is run 50 times with 5000 iterations. The minimum (Min), maximum (Max), average (Avg), and standard deviation (Std) of the MSEs observed at 200, 500, 2000, and 5000 iterations are presented in Tables 7.3, 7.4, 7.5, 7.6. The MSE values for Cases I and II are different because of different fitness functions, so the filter designed using I_2 in Case II is also evaluated using I_1 from Case I in order to perform a one-one compar-

Table 7.1 Algorithm specifications

Parameter	Symbol	Value
Population size	S	25
Number of iterations		5000
Number of trials		50
Inertia	w	linearly decreasing from 0.9 to 0.4
Cognitive constant	c_1	2
Social constant	c_2	2
Creativity coefficient	β	linearly increasing from 0.5 to 1
Crossover rate		0.5

Table 7.2 Filter specifications

Filter	Parameter	Symbol	Value
LP, HP	Passband normalized cutoff frequency	ω_p	0.3
LP, HP	Stopband normalized cutoff frequency	ω_s	0.35
BP, BS	Normalized lower cutoff frequency	ω_l	0.3
BP, BS	Normalized upper cutoff frequency	ω_u	0.7
LP, HP, BP, BS	Passband ripple	δ_p	0.1
LP, HP, BP, BS	Stopband ripple	δ_p	0.01
LP, HP, BP, BS	Number of coefficients	N	21
LP, HP, BP, BS	Number of samples	T	256
LP, HP, BP, BS	Weighting vector	G	1

ison of the MSEs obtained by the two approaches. This is shown in the row labeled "Avg Case I" (italicized) under the columns for Case II. This comparison of the two cases shows that lower values of MSE are obtained when using I_1. This suggests that trying to approximate a filter to the ideal filter results is better than trying to approximate it to a given design specification. The filters designed by the CLS method are also evaluated using fitness functions of both cases (shown in columns 'I_1' and 'I_2'). The comparison of these MSEs with those obtained with CLS also confirms that using PSO-QI in Case I achieves better results. The lowest MSE values for Case I, and the lowest standard deviations values for Cases I and II, are shown in bold for different iterations. The consistent performance of the PSO-QI algorithm is demonstrated in the results by the lower standard deviation values obtained in the final iteration. The maximum ripples in the passband and stopband obtained for different filters are also presented.

These show that none of the design approaches could meet the specification exactly. This is mainly because the algorithms compromised on transition width in order to best meet the other requirements for designing an FIR filter of a given order. Evolutionary algorithms are still able to design filters with narrower transition

Table 7.3 Comparison of results for lowpass FIR filters

	Case I			Case II			CLS	
	PSO	DEPSO	PSO-QI	PSO	DEPSO	PSO-QI	I_1	I_2
Avg Time (s)	61.00296	63.51265	63.84031	63.80519	66.25460	67.14100	<0.05	
Iterations: 200								
Avg	2.309090	**1.367769**	1.696142	2.217716	1.290407	1.582336		
Min	0.918869	0.542448	0.839119	0.828234	0.398866	0.780351		
Max	5.857412	3.868421	2.652464	4.590857	2.214809	3.848432		
Std	0.885853	0.516622	0.447363	0.732349	**0.420632**	0.536772		
Iterations: 500								
Avg	1.353529	**0.647617**	0.950669	1.203629	0.647285	0.957092		
Min	0.491010	0.196000	0.447505	0.493261	0.291929	0.536816		
Max	4.034052	3.035559	1.509080	3.417326	1.074832	3.396342		
Std	0.698160	0.408098	0.233942	0.494053	**0.182853**	0.409753		
Iterations: 2000								
Avg	0.172948	0.073314	**0.019102**	0.064034	0.015196	0.015353		
Min	0.007040	0.004527	0.004621	0.003499	0.002368	0.002583		
Max	2.572305	2.569120	0.056903	2.458580	0.047153	0.048897		
Std	0.610465	0.360470	0.011947	0.345662	**0.009839**	0.011033		
Iterations: 5000								
Avg	0.154615	0.052767	0.001628	0.049444	0.000559	0.000508		0.001790
Avg Case I[a]	*0.154615*	*0.052767*	**0.001628**	*0.054134*	*0.002973*	*0.002867*	*0.003577*	
Min	0.001465	0.001465	0.001466	0.000338	0.000338	0.000338		
Max	2.556164	2.558192	0.002145	2.447605	0.000939	0.000991		
Std	0.611835	0.361552	0.000171	0.346073	0.000130	**0.000137**		
Maximum passband ripple (PBR)								
Avg	0.352552	0.248751	**0.207626**	0.290292	0.245822	0.240912	0.280047	
Min	0.171267	0.157795	0.158447	0.214734	0.206672	0.210183		
Max	2.752903	2.512941	0.248144	2.609865	0.288430	0.273903		
Std	0.584106	0.327469	0.023396	0.334947	0.015451	**0.011410**		
Maximum stopband ripple (SBR)								
Avg	0.400418	0.287675	0.228088	0.224103	0.172175	**0.166285**	0.294372	
Min	0.208338	0.208427	0.208663	0.141762	0.141850	0.141765		
Max	3.152688	3.087697	0.265057	3.089060	0.206332	0.215424		
Std	0.697131	0.404287	**0.011508**	0.413732	0.016193	0.016907		

[a] Case II evaluated using I_1 for comparison

Table 7.4 Comparison of results for highpass FIR filters

	Case I			Case II			CLS	
	PSO	DEPSO	PSO-QI	PSO	DEPSO	PSO-QI	I_1	I_2
Avg Time (s)	62.24904	64.89828	65.58133	63.55418	66.17452	67.029559	<0.05	
Iterations: 200								
Avg	1.581663	**0.808356**	1.076798	1.537441	0.841425	0.916861		
Min	0.772375	0.360826	0.338438	0.610639	0.193175	0.279725		
Max	3.203546	2.202218	1.841334	3.703361	2.972754	1.561778		
Std	0.533546	0.312968	0.312228	0.653536	0.473965	**0.267402**		
Iterations: 500								
Avg	0.840090	**0.357080**	0.560789	0.793403	0.328740	0.451516		
Min	0.291398	0.201776	0.232625	0.297373	0.104778	0.118738		
Max	2.661553	0.600713	1.066141	3.052292	1.479729	0.860819		
Std	0.435271	**0.101233**	0.164376	0.557034	0.207237	0.177360		
Iterations: 2000								
Avg	0.052417	0.021758	**0.020838**	0.042570	0.013654	0.009659		
Min	0.006056	0.006115	0.005003	0.001894	0.001562	0.001810		
Max	1.599131	0.058461	0.058322	1.441952	0.038430	0.037950		
Std	0.223447	0.011439	0.012320	0.202313	0.009904	**0.006886**		
Iterations: 5000								
Avg	0.033264	0.001372	0.001392	0.028917	0.000395	0.000390		0.001191
Avg Case I[a]	*0.033264*	**0.001372**	*0.001392*	*0.035859*	*0.004065*	*0.004125*	*0.003010*	
Min	0.001139	0.001111	0.001098	0.000283	0.000277	0.000283		
Max	1.594575	0.002066	0.001932	1.426712	0.001024	0.000590		
Std	0.225309	0.000206	0.000212	0.201712	0.000106	**0.000068**		
PBR								
Avg	0.243249	0.204076	**0.208505**	0.262383	0.224378	0.227248	0.282123	
Min	0.169534	0.160837	0.171776	0.206175	0.193702	0.202260		
Max	2.009299	0.242780	0.236678	2.035581	0.252572	0.262238		
Std	0.255295	0.020422	0.016280	0.256075	0.012594	**0.011951**		
SBR								
Avg	0.258005	0.209504	0.209503	0.199337	0.155278	**0.153901**	0.250586	
Min	0.187816	0.189222	0.192309	0.135176	0.136969	0.135751		
Max	2.594075	0.253945	0.243862	2.391242	0.227605	0.186739		
Std	0.337364	0.013087	0.012443	0.316451	0.013554	**0.010365**		

[a] Case II evaluated using I_1 for comparison

Table 7.5 Comparison of results for bandpass FIR filters

	Case I			Case II			CLS	
	PSO	DEPSO	PSO-QI	PSO	DEPSO	PSO-QI	I_1	I_2
Avg Time (s)	63.58043	66.26156	66.67736	65.81591	68.68277	69.082577	<0.05	
Iterations: 200								
Avg	2.313546	**1.179951**	1.554501	1.982181	1.170335	1.586532		
Min	1.172316	0.596113	0.734473	1.062480	0.426690	0.777035		
Max	5.305936	2.400848	3.782131	5.406325	4.640706	3.857878		
Std	0.818909	**0.389749**	0.553299	0.705846	0.592620	0.522986		
Iterations: 500								
Avg	1.279520	**0.569743**	0.946342	1.123554	0.577790	0.893920		
Min	0.531468	0.305807	0.496780	0.339226	0.259503	0.346511		
Max	3.662125	1.199284	3.478458	4.336395	3.292193	3.598533		
Std	0.481833	**0.198388**	0.433324	0.564460	0.417246	0.457655		
Iterations: 2000								
Avg	**0.028687**	0.031538	0.036194	0.073863	0.026110	0.026231		
Min	0.018695	0.018111	0.020022	0.014262	0.012300	0.012841		
Max	0.043332	0.074883	0.095605	2.440620	0.058284	0.071384		
Std	**0.006733**	0.011413	0.014509	0.341638	0.009765	0.013427		
Iterations: 5000								
Avg	0.016428	0.016461	0.016384	0.058796	0.010273	0.010180		0.014942
Avg Case I[a]	*0.016428*	*0.016461*	**0.016384**	*0.069030*	*0.017737*	*0.017593*	*0.022408*	
Min	0.016131	0.016137	0.016126	0.009753	0.009745	0.009748		
Max	0.018635	0.018114	0.017252	2.434750	0.012572	0.011388		
Std	0.000368	0.000382	**0.000262**	0.342869	0.000616	0.000456		
Maximum passband ripple (PBR)								
Avg	0.479268	0.482948	**0.479261**	0.547783	0.523475	0.524352	0.522534	
Min	0.439810	0.442510	0.430079	0.503902	0.501465	0.504393		
Max	0.501762	0.504366	0.500726	1.756379	0.537421	0.537750		
Std	0.017339	0.017040	0.017466	0.174703	0.010566	**0.010480**		
Maximum stopband ripple (SBR)								
Avg	0.481411	0.478034	0.480510	0.465866	0.433212	**0.432359**	0.444855	
Min	0.451668	0.455327	0.455083	0.412330	0.411028	0.411814		
Max	0.518466	0.518381	0.524113	2.060793	0.460651	0.462116		
Std	0.019935	0.019943	0.020896	0.230634	0.014247	**0.013644**		

[a] Case II evaluated using I_1 for comparison

Table 7.6 Comparison of results for bandstop FIR filters

	Case I			Case II			CLS	
	PSO	DEPSO	PSO-QI	PSO	DEPSO	PSO-QI	I_1	I_2
Avg Time (s)	64.15355	66.95550	67.30056	65.97979	68.81981	69.174664	<0.05	
Iterations: 200								
Avg	1.764650	**1.069331**	1.230915	1.581259	0.884914	1.095112		
Min	0.755631	0.392187	0.492116	0.394064	0.299558	0.377232		
Max	4.355703	3.311367	2.881854	2.732023	1.561214	2.018252		
Std	0.630130	0.615877	0.447678	0.531293	**0.301292**	0.392611		
Iterations: 500								
Avg	0.938503	**0.501911**	0.689372	0.835579	0.385321	0.593301		
Min	0.334030	0.269786	0.316557	0.192995	0.167570	0.273839		
Max	3.677116	2.991282	2.748155	1.613370	0.673929	0.986046		
Std	0.495277	0.475637	0.340402	0.310986	**0.143453**	0.176029		
Iterations: 2000								
Avg	0.075912	**0.075032**	0.037216	0.022549	0.022663	0.023310		
Min	0.021493	0.018992	0.019132	0.013481	0.010508	0.011030		
Max	2.083948	1.997408	0.100823	0.044540	0.043413	0.077686		
Std	0.290045	0.277730	0.013211	**0.006871**	0.008525	0.011572		
Iterations: 5000								
Avg	0.056783	0.055084	0.015453	0.009954	0.010155	0.010056		0.015682
Avg Case I[a]	*0.056783*	*0.055084*	**0.015453**	*0.018135*	*0.018273*	*0.018130*	*0.023026*	
Min	0.015190	0.015191	0.015190	0.009181	0.009214	0.009183		
Max	2.073480	1.990370	0.016199	0.010547	0.011016	0.010629		
Std	0.291025	0.279277	**0.000242**	0.000404	0.000357	0.000356		
Maximum passband ripple (PBR)								
Avg	0.482101	0.485566	**0.466093**	0.514832	0.513667	0.512387	0.489694	
Min	0.429418	0.443209	0.438536	0.497235	0.493062	0.495210		
Max	1.313270	1.509979	0.496024	0.534771	0.546941	0.541161		
Std	0.120814	0.148465	0.013660	0.011329	0.012043	**0.010907**		
Maximum stopband ripple (SBR)								
Avg	0.522783	0.522197	0.490307	**0.436772**	0.440931	0.440415	0.478088	
Min	0.469555	0.463607	0.461629	0.411035	0.414722	0.411434		
Max	1.925307	1.897104	0.522759	0.456460	0.460227	0.456801		
Std	0.202776	0.198756	0.013233	0.012244	**0.009701**	0.009885		

[a] Case II evaluated using I_1 for comparison

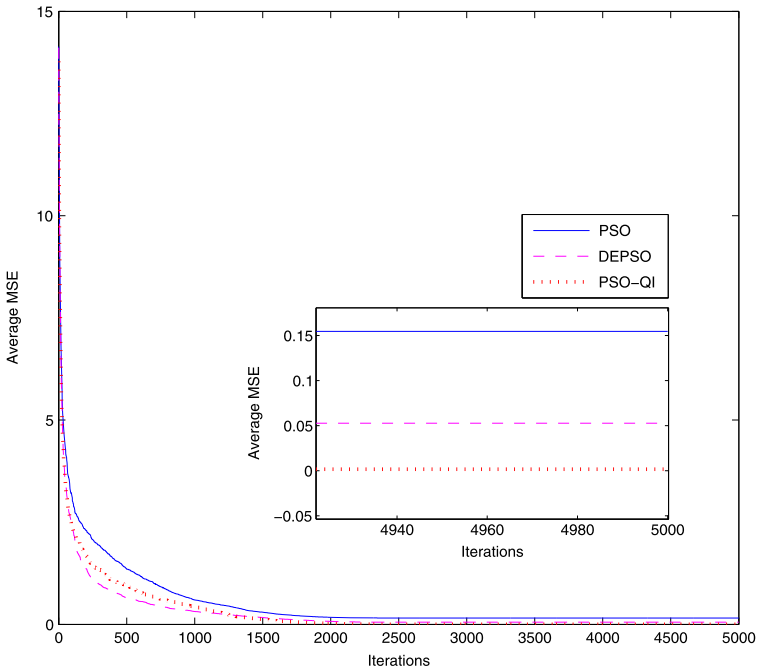


Fig. 7.2 Error plot for the design of LP FIR filter in Case I

widths than those designed using the CLS method. The average design times for 5000 iterations, as observed on an Intel Core 2 CPU 2.13 GHz desktop computer with 2 GB of RAM, are also listed. DEPSO and PSO-QI take slightly longer than PSO, as is evident from the fact that for a population size of P, PSO takes P while DEPSO and PSO-QI take $P + 1$ fitness evaluations per iteration.

Figures 7.2, 7.3, 7.4, 7.5 show the convergence curves for the different algorithms used to design the LP, HP, BP, and BS FIR filters, respectively, using the fitness function from Case I. These graphs show the quick convergence behavior of DEPSO over several iterations.

The magnitude plots of the LP FIR filters designed in Case I are shown in Fig. 7.6. Similar plots for Case II are shown in Fig. 7.7. Similarly, the magnitude plots of the HP FIR filter for Cases I and II are shown in Figs. 7.8 and 7.9. In Figs. 7.10, 7.11, 7.12, 7.13, similar plots are shown for the BP FIR and BS FIR filters, respectively.

All of these frequency response diagrams show the ability of the evolutionary algorithms to design different kinds of FIR filters successfully. The figures show that the CLS method of filter design is able to meet the passband and stopband ripple requirement for most frequency bands, except around the cutoff frequencies (i.e., with a wider transition width). These results suggest that based on the given filter specification, deviations in one or the other design parameters is unavoidable. The filter designed using evolutionary algorithms is hence the most "optimal" in

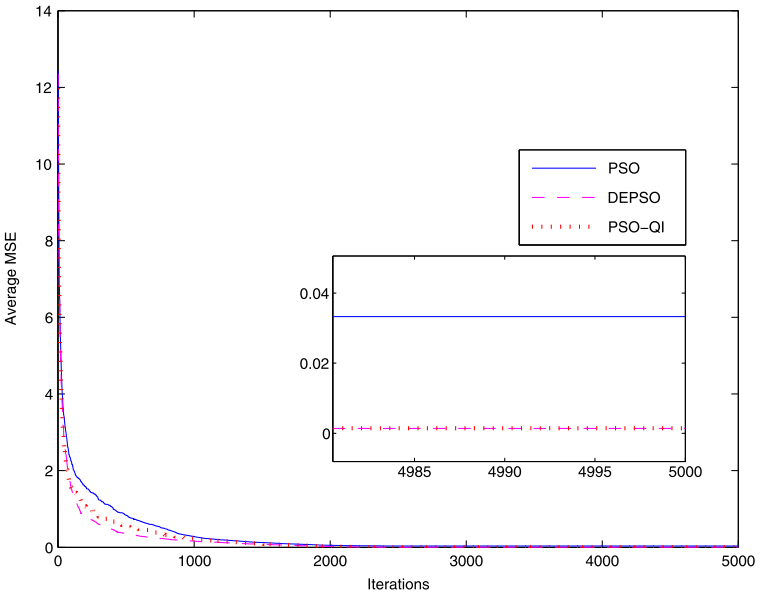


Fig. 7.3 Error plot for the design of HP FIR filter in Case I

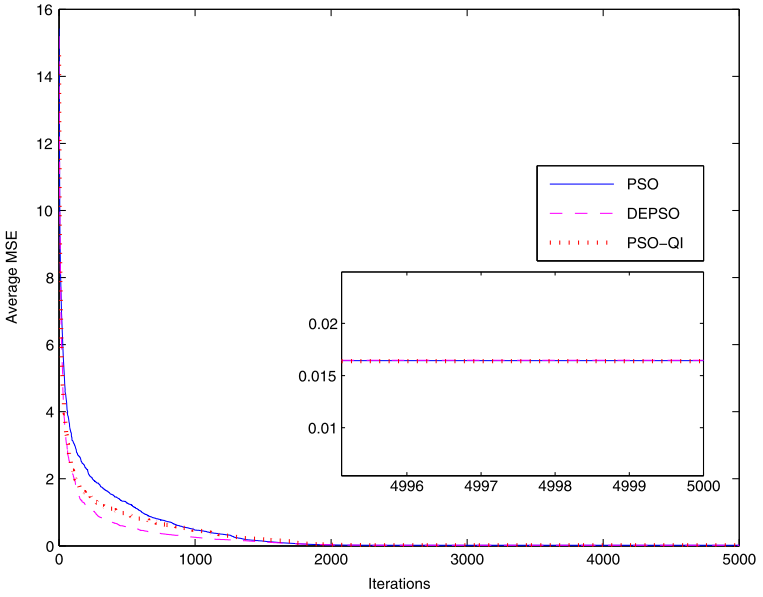


Fig. 7.4 Error plot for the design of BP FIR filter in Case I

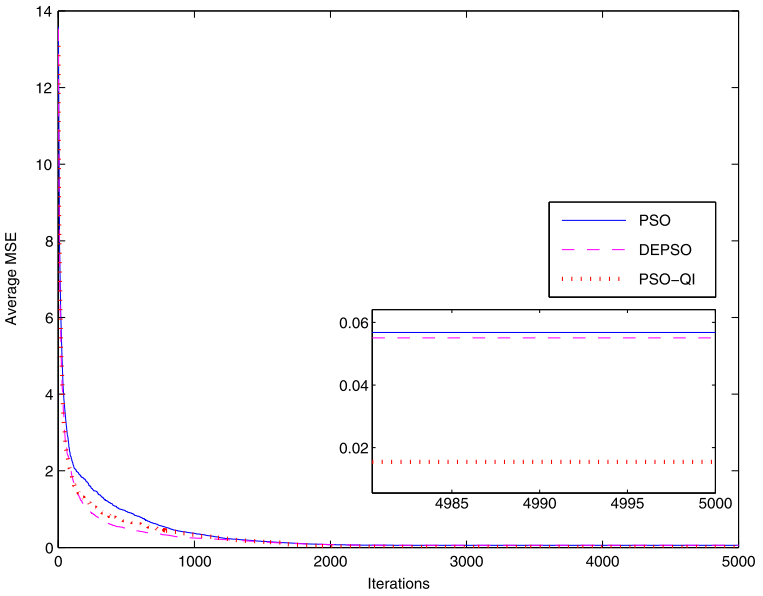


Fig. 7.5 Error plot for the design of BS FIR filter in Case I

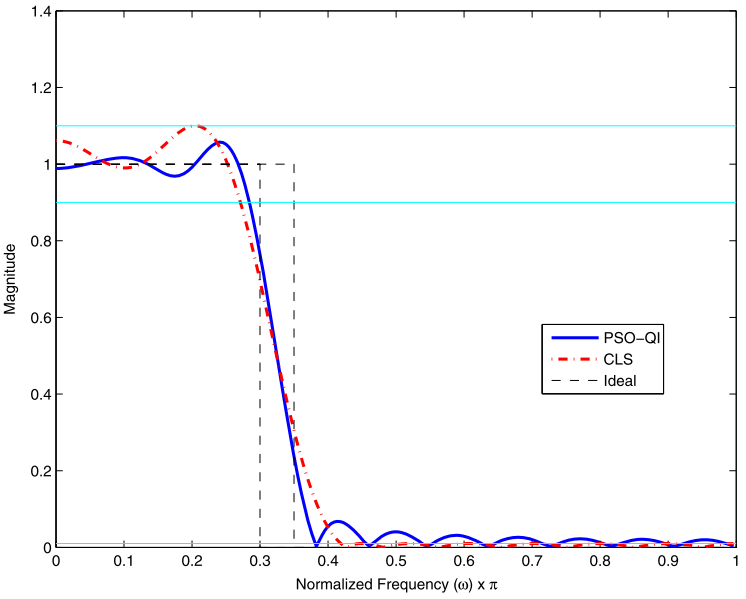


Fig. 7.6 Magnitude plot of the LP FIR filter designed in Case I

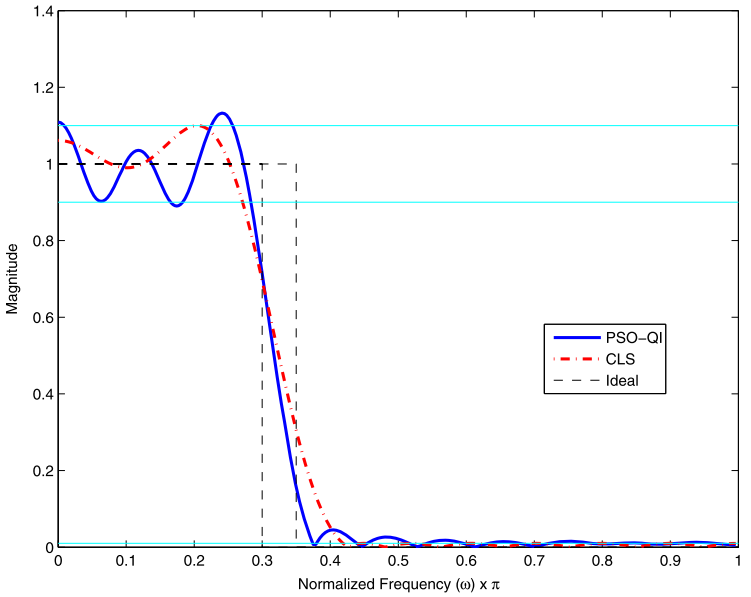


Fig. 7.7 Magnitude plot of the LP FIR filter designed in Case II

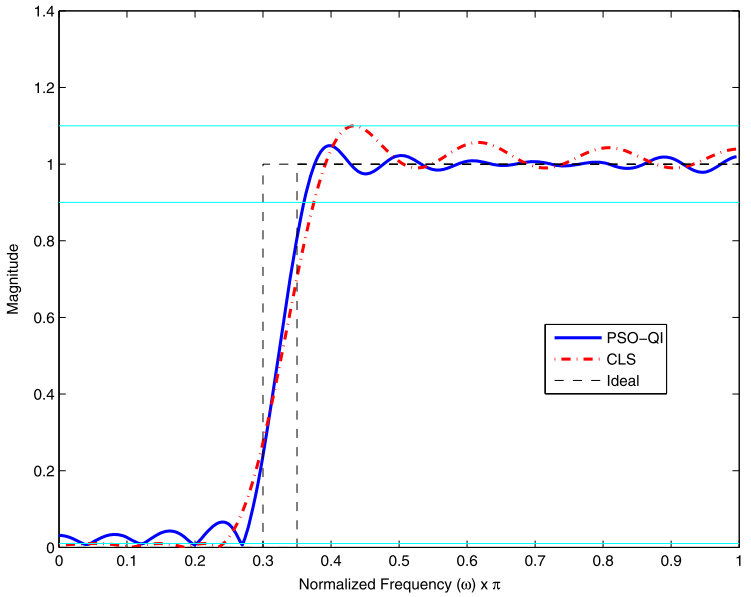


Fig. 7.8 Magnitude plot of the HP FIR filter designed in Case I

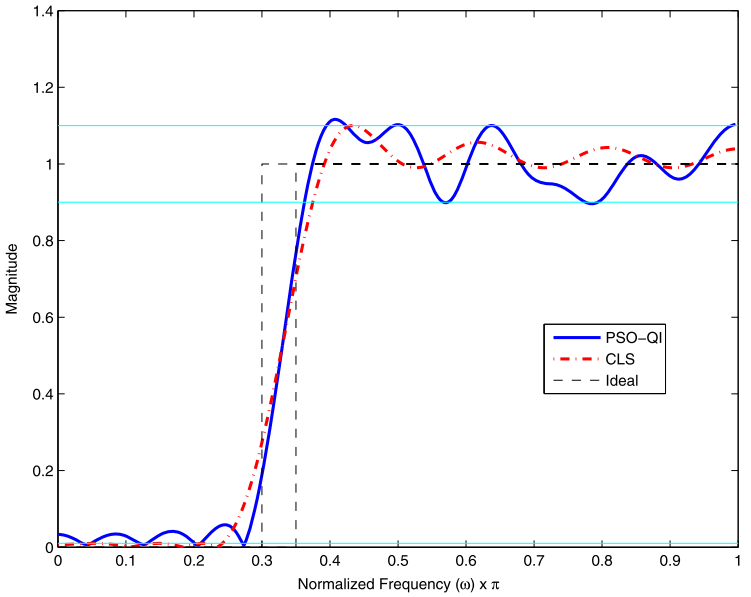


Fig. 7.9 Magnitude plot of the HP FIR filter designed in Case II

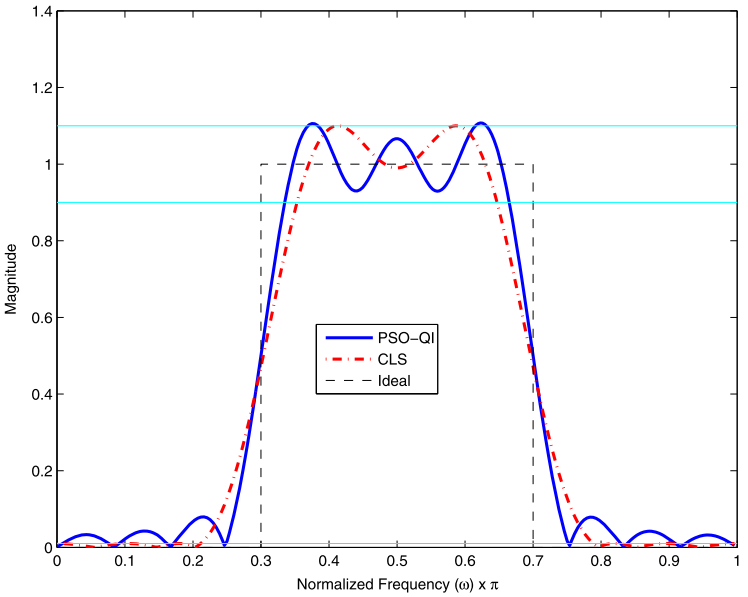


Fig. 7.10 Magnitude plot of the BP FIR filter designed in Case I

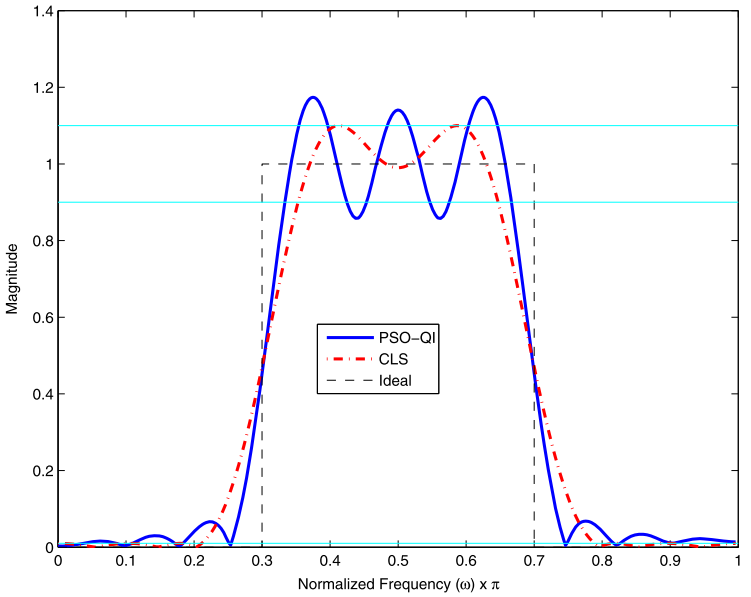


Fig. 7.11 Magnitude plot of the BP FIR filter designed in Case II

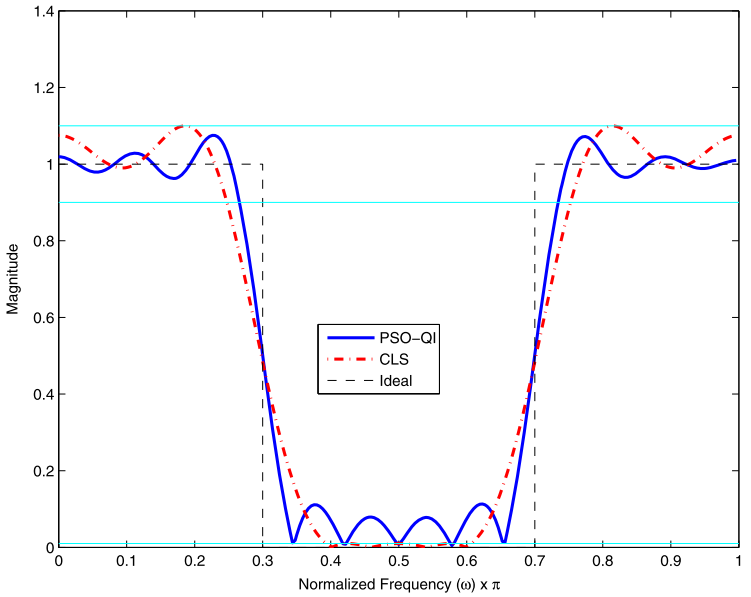


Fig. 7.12 Magnitude plot of the BS FIR filter designed in Case I

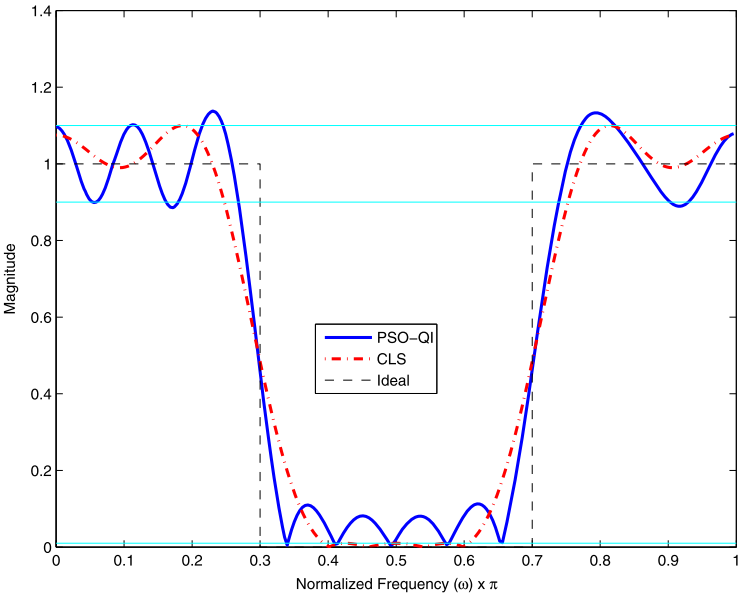


Fig. 7.13 Magnitude plot of the BS FIR filter designed in Case II

the sense that its response is closer to the ideal filter, and its deviation from the design specification is minimal. The proposed PSO-QI algorithm is the most suitable design approach among the three algorithms because of its minimum average error and consistent performance, as evidenced by the lower standard deviation. The following observations are made based on the results:

- Maximum ripples in the passband and stopband are obtained around the cutoff frequencies.
- Filters designed using fitness function in Case I deviate less towards the passband cutoff frequency, and hence, have lower maximum passband ripple values.
- Filters designed using the fitness function in Case II deviate less towards the stopband cutoff frequency, and hence, have lower maximum stopband ripple values.
- However, the passband ripples are higher throughout the frequency bands (as seen in Figs. 7.6 through 7.13) using fitness function in Case II than in Case I.
- For most of the filters designed, lower standard deviations are found for Case II. This implies that the fitness function in Case II has more consistent performance.
- DEPSO has better convergence for fewer iterations (up to 2000), while PSO-QI outperforms other algorithms for 5000 iterations.

Table 7.7 shows the coefficients for the different types of FIR filters obtained using the PSO-QI algorithm.

Table 7.7 Coefficients of the various FIR filters designed using PSO-QI

	Lowpass		Highpass		Bandpass		Bandstop	
	I_1	I_2	I_1	I_2	I_1	I_2	I_1	I_2
a_0	0.0362	−0.0232	0.0893	0.1036	0.0168	0.1080	−0.1794	0.1395
a_1	0.0442	−0.0732	−0.0704	−0.0880	0.1245	−0.0201	−0.2121	0.0667
a_2	0.0106	−0.1134	−0.0030	−0.0623	−0.0600	−0.2810	−0.2064	0.1124
a_3	−0.0554	−0.1292	−0.0604	0.0890	−0.2996	0.0373	−0.4590	−0.0157
a_4	−0.1098	−0.1058	0.0901	−0.0749	0.0965	0.3819	0.0970	−0.0391
a_5	−0.0982	−0.0731	−0.0530	0.1438	0.3772	−0.0408	−0.3437	−0.3051
a_6	4.8e–4	−0.0635	0.0943	−0.2070	−0.0923	−0.3069	0.2825	0.0315
a_7	0.1513	−0.1088	−0.1030	0.3764	−0.2702	0.0220	−0.0054	−0.3814
a_8	0.2776	−0.1771	0.2299	−0.4208	0.0455	0.0921	0.0745	0.2527
a_9	0.3141	−0.2272	−0.5143	−0.0578	0.0497	0.0036	0.0817	−0.0332
a_{10}	0.2479	−0.1976	0.2716	0.3628	0.0097	0.1000	−0.1100	0.2147
a_{11}	0.1267	−0.0850	0.3287	0.0402	0.1146	−0.0178	−0.0510	0.2834
a_{12}	0.0233	0.0623	−0.2660	−0.2403	−0.0334	−0.1378	0.0011	−2.1e–4
a_{13}	−0.0160	0.1751	−0.0971	−0.0390	−0.1168	0.0110	−0.0493	0.2135
a_{14}	0.0036	0.1910	−0.0346	0.0033	0.0177	0.0325	0.0805	0.0222
a_{15}	0.0391	0.1200	0.1323	0.1498	0.0067	0.0044	0.0471	0.0450
a_{16}	0.0482	0.0091	0.0644	0.0647	0.0105	0.0847	−0.0233	0.1907
a_{17}	0.0227	−0.0784	−0.0230	−0.1411	0.0788	−0.0139	0.0184	0.0546
a_{18}	−0.0150	−0.1030	−0.0885	−0.0706	−0.0206	−0.1079	−0.0520	0.1564
a_{19}	−0.0354	−0.0774	−0.0343	0.0953	−0.0643	0.0090	−0.0414	0.0810
a_{20}	−0.0276	−0.0309	0.0783	0.0063	0.0082	0.0428	0.0308	0.0076

7.6 Conclusion

This chapter presented the iterative design of FIR filters using population-based swarm (PSO), hybrid swarm-evolutionary (DEPSO), and swarm-quantum (PSO-QI) stochastic algorithms. Different types of FIR filters were designed using the mean squared error as the fitness function formulated in two ways. All three of the algorithms were able to approximate the filter coefficients over a number of iterations, but PSO-QI always performed the best, given enough iterations. The results show that PSO-QI is more consistent in its performance. In comparison to the constrained least squares method of FIR filter design, filters designed iteratively using evolutionary algorithms had better frequency response. However, none of the approaches confirmed to the given design specification. Hence, there is more potential for research in this area. A multi-objective optimization problem may be formulated with relative weighting placed on different objectives in order to simultaneously optimize various filter parameters. The results indicated that different algorithms perform differently in various frequency bands, and while one criterion is satisfied, another is violated. Hence, the best advantage of iterative-designed digital filters

over traditional techniques comes from the algorithms' ability to find multiple solutions so that designers can choose a solution based on their requirements. This can be achieved using a Pareto optimization technique, which is a potential area of research in the iterative design of digital filters.

Acknowledgements The financial support provided by NSF EFRI (#1238097) and NSF CAREER (#1231820) is gratefully acknowledged.

References

1. Ababneh, J., Bataineh, M.: Linear phase FIR filter design using particle swarm optimization and genetic algorithms. Digit. Signal Process. **18**(4), 657–668 (2008)
2. Aktan, M., Yurdakul, A., Dundar, G.: An algorithm for the design of low-power hardware-efficient FIR filters. IEEE Trans. Circuits Syst. I, Regul. Pap. **55**(6), 1536–1545 (2008)
3. Chang, C.H., Chen, J., Vinod, A.: Information theoretic approach to complexity reduction of FIR filter design. IEEE Trans. Circuits Syst. I, Regul. Pap. **55**(8), 2310–2321 (2008)
4. Del Valle, Y., Venayagamoorthy, G., Mohagheghi, S., Hernandez, J., Harley, R.: Particle swarm optimization: basic concepts, variants and applications in power systems. IEEE Trans. Evol. Comput. **12**(2), 171–195 (2008)
5. Fang, W., Sun, J., Xu, W.: Analysis of adaptive IIR filter design based on quantum-behaved particle swarm optimization. In: The Sixth World Congress on Intelligent Control and Automation (WCICA 2006), vol. 1 (2006)
6. Fang, W., Sun, J., Xu, W., Liu, J.: FIR digital filters design based on quantum-behaved particle swarm optimization. In: First International Conference on Innovative Computing, Information and Control (ICICIC'06), vol. 1 (2006)
7. Huang, W., Zhou, L., Qian, J.: FIR filter design: frequency sampling filters by particle swarm optimization algorithm. In: Proceedings of 2004 International Conference on Machine Learning and Cybernetics, vol. 4 (2004)
8. Jou, Y., Chen, F.: Least-squares design of FIR filters based on a compacted feedback neural network. IEEE Trans. Circuits Syst. II, Express Briefs **54**(5), 427–431 (2007)
9. Karaboga, N.: Digital IIR filter design using differential evolution algorithm. EURASIP J. Appl. Signal Process. **2005**(8), 1269–1276 (2005)
10. Krusienski, D., Jenkins, W.: Particle swarm optimization for adaptive IIR filter structures. In: Proc. Congress on Evolutionary Computation (CEC2004), vol. 1, pp. 965–970 (2004)
11. Kwan, H.K., Jiang, A.: FIR, Allpass, and IIR variable fractional delay digital filter design. IEEE Trans. Circuits Syst. I, Regul. Pap. **56**(9), 2064–2074 (2009)
12. Liu, Y., Lin, Z.: Optimal design of frequency-response masking filters with reduced group delays. IEEE Trans. Circuits Syst. I, Regul. Pap. **55**(6), 1560–1570 (2008)
13. Luitel, B., Venayagamoorthy, G.: Differential evolution particle swarm optimization for digital filter design. In: Proc. IEEE World Congress on Computational Intelligence (WCCI'08), pp. 3954–3961 (2008)
14. Luitel, B., Venayagamoorthy, G.K.: Particle swarm optimization with quantum infusion for system identification. Eng. Appl. Artif. Intell. **23**(5), 635–649 (2010)
15. Mandal, S., Ghoshal, S.P., Kar, R., Mandal, D.: Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique. J. King Saud Univ, Comput. Inf. Sci. **24**(1), 83–92 (2012)
16. Meher, P.K.: New approach to look-up-table design and memory-based realization of FIR digital filter. IEEE Trans. Circuits Syst. I, Regul. Pap. **57**(3), 592–603 (2010). doi:10.1109/TCSI.2009.2026683
17. Shyu, J.J., Pei, S.C., Huang, Y.D.: Design of variable two-dimensional FIR digital filters by McClellan transformation. IEEE Trans. Circuits Syst. I, Regul. Pap. **56**(3), 574–582 (2009)

18. Storn, R.: Designing nonstandard filters with differential evolution. IEEE Signal Process. Mag. **22**(1), 103–106 (2005)
19. Sun, J., Feng, B., Xu, W.: Particle swarm optimization with particles having quantum behavior. In: Proc. Congress on Evolutionary Computation (CEC2004), vol. 1, pp. 325–331 (2004)
20. Yu, Y., Lim, Y.: Design of linear phase FIR filters in subexpression space using mixed integer linear programming. IEEE Trans. Circuits Syst. I, Regul. Pap. **54**(10), 2330–2338 (2007)

Chapter 8
A Metaheuristic Approach to Two Dimensional Recursive Digital Filter Design

Abhronil Sengupta, Tathagata Chakraborti, and Amit Konar

Abstract The two dimensional IIR digital filter design problem has received increased attention over the past few years. Recently, several metaheuristic algorithms have been employed in this domain and have produced promising results. Invasive Weed Optimization is one of the latest population-based metaheuristic algorithms that mimics the colonizing action of weeds. In this chapter, an improvement to the classical weed optimization algorithm has been proposed by introducing a constriction factor in the seed dispersal phase. Temporal Difference Q-Learning has been employed to adapt this parameter for different population members through the successive generations. Such hybridization falls under a special class of adaptive Memetic Algorithms. The proposed memetic realization, called Intelligent Invasive Weed Optimization (IIWO), has been applied to the two-dimensional recursive digital filter design problem and it has outperformed several competitive algorithms that have been applied in this research field in the past.

8.1 Introduction

Digital filters can be classified into two broad categories, namely, finite impulse response (FIR) filters and infinite impulse response (IIR) filters. FIR filters are easier to implement as they are non-recursive and are always stable. However, IIR filters are much more efficient in comparison to FIR filters as they are capable of producing very sharp and accurate frequency responses [4]. Since the design of IIR filters is more challenging than its FIR counterpart, we have selected 2D IIR filter design as the fundamental problem of this chapter.

A. Sengupta (✉) · T. Chakraborti · A. Konar
Dept. of Electronics and Telecommunication Eng., Jadavpur University, Kolkata, India
e-mail: senguptaabhronil@gmail.com

T. Chakraborti
e-mail: tathagata.net@live.com

A. Konar
e-mail: konaramit@yahoo.co.in

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_8,

The most popular design procedures of 2D IIR filters fall under two main categories [4, 8, 16, 18]:

– Those based on appropriate transformation of one dimensional (1D) filters [14, 26];
– Those based on appropriate optimization techniques such as linear programming, Remez exchange algorithm, etc. [3, 10, 13, 15, 23, 29].

However, the majority of these algorithms result in an unstable filter. Although various methods have been proposed to tackle the instability problem, yet their practical implementation suffers from a very small stability margin [18]. The applications of evolutionary computation techniques to the design of digital IIR filters have been investigated in [4, 8, 16, 18]. The results reported in [4, 8, 16, 18] suggest that modern search heuristics are more efficient in the filter design problem. In [18] and [16], the authors propose a Neural Network (NN) and Genetic Algorithm (GA) approach to the recursive filter design. However, the computer language GENETICA [8] was able to outperform the above mentioned algorithms. Recently Das et al. [4] formulated a new variant of Particle Swarm Optimization (MEPSO) for the purpose. In this chapter, we focus on the application of an improved version of a recently proposed metaheuristic, namely, Invasive Weed Optimization (IWO).

IWO [17] is a derivative-free optimization technique that mimics the ecological behavior of weeds. This metaheuristic algorithm has attracted researchers because of its reduced computational cost and efficiency in tackling real world optimization problems. However, it is not free from the problems of stagnation and preconvergence. Here, we attempt to improve the performance of the traditional IWO algorithm by incorporating a learning strategy in the weed population to efficiently disperse seeds throughout the problem space during the reproduction phase. Such a memetic learning technique helps in balancing the exploration and exploitation capabilities of the weeds which is necessary for providing precise solutions to global optimization problems.

The concept of Memetic Algorithm (MA) [5] falls in a broad category of population based metaheuristics that incorporate strategies for individual learning. Evolutionary Algorithms (EAs) determine the global optima in a given search landscape in ways inspired by natural evolution and the Darwinian principles of the struggle for existence and survival of the fittest. Traditional EAs fail to exploit local information and generally become impractical due to excessively large time required to locate a more or less accurate solution. However, cultural evolution is capable of local refinement. Thus, MA captures the power of global search by its evolutionary component and local search by its cultural component, and has successfully outperformed conventional EAs in several fields of science and engineering [12, 19, 20].

The earliest research regarding Memetic Algorithms can be traced back to the work of Moscato [19]. Our research falls in the domain of Adaptive Memetic Algorithms (AMAs) which involve adaptive selection of memes from the meme pool. This adaptive selection is controlled by the ability of the meme to perform improvement in fitness value. Several variants of AMAs are found in the literature [2].

The AMA to be proposed, named Intelligent Invasive Weed Optimization (IIWO), includes an Invasive Weed Optimization (IWO) algorithm for global search

and a Temporal Difference Q-Learning (TDQL) [27, 28] for local refinement. A constriction factor has been included in the expression for standard deviation for dispersal of seeds. It is important to mention here that the constriction factors for all members of the population should not be equal for the best performance. A member with a good fitness should search in the local neighborhood, whereas a poor performing member should participate in the global search. A good member thus should have small constriction factors, while worse members should have relatively large constriction factors. The selection of constriction factors from the meme pool is governed by the TDQL learning policy.

8.2 The Design Problem Formulation

The general prototype transfer function of an N dimensional recursive IIR digital filter is represented by the following expression:

$$H(z_1, z_2) = H_0 \frac{\sum_{i=0}^{N} \sum_{j=0}^{N} p_{ij} z_1^i z_2^j}{\prod_{k=1}^{N} (1 + q_k z_1 + r_k z_2 + s_k z_1 z_2)}, \quad p_{00} = 1. \tag{8.1}$$

The variables z_1 and z_2 represent the complex indeterminants in the discrete Laplace Transform and are related to the Fourier domain frequency terms ω_1 and ω_2 by the relationship $z_1 = e^{-j\omega_1}$ and $z_2 = e^{-j\omega_2}$ (where $\omega_1, \omega_2 \in [-\pi, \pi]$).

Let us assume that the user-specified amplitude response of the filter is designated by $M_d(\omega_1, \omega_2)$. The design task reduces to finding an appropriate transfer function $H(z_1, z_2)$ such that $M(\omega_1, \omega_2) = H(e^{-j\omega_1}, e^{-j\omega_2})$ follows the desired response $M_d(\omega_1, \omega_2)$ as closely as possible. The approximation can be achieved by minimizing [4, 8, 16, 18],

$$J(p_{ij}, q_k, r_k, s_k, H_0) = \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \left[\left| M(\omega_1, \omega_2) \right| - M_d(\omega_1, \omega_2) \right]^b \tag{8.2}$$

where $\omega_1 = (\pi/N_1)n_1$, $\omega_2 = (\pi/N_2)n_2$, and b is a positive integer (usually $b = 1, 2, 4,$ or 8).

Here the prime objective is to reduce the error between the desired and actual amplitude responses of the filter at $N_1 N_2$ points. Since the denominator consists of only first degree factors, we assert the stability conditions following [14, 26] as:

$$|q_k + r_k| - 1 < s_k < 1 - |q_k - r_k|, \tag{8.3}$$

where $k = 1, 2, \ldots, N$.

Thus the design of a 2D recursive filter is equivalent to the following constrained optimization problem:

Minimize

$$J = \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \left[\left| M\left(\frac{\pi n_1}{N_1}, \frac{\pi n_2}{N_2}\right) \right| - M_d\left(\frac{\pi n_1}{N_1}, \frac{\pi n_2}{N_2}\right) \right]^b \tag{8.4}$$

subject to the constraints

$$|q_k + r_k| - 1 < s_k, \quad k = 1, 2, \ldots, N, \tag{8.5}$$

$$s_k < 1 - |q_k - r_k|, \quad k = 1, 2, \ldots, N \tag{8.6}$$

where N_1, N_2, and N are positive integers.

Without loss of generality, we consider the case of $N = 2$. Thus $H(z_1, z_2)$ in (8.1) can be simplified as shown in (8.7):

$$\begin{aligned} H(z_1, z_2) = H_0\big(&p_{00} + p_{01}z_2 + p_{02}z_2^2 + p_{10}z_1 + p_{20}z_1^2 + p_{11}z_1z_2 + p_{12}z_1z_2^2 \\ &+ p_{21}z_1^2z_2 + p_{22}z_1^2z_2^2\big)/\big[(1 + q_1z_1 + r_1z_2 + s_1z_1z_2)(1 + q_2z_1 \\ &+ r_2z_2 + s_2z_1z_2)\big]. \end{aligned} \tag{8.7}$$

Now, transforming the variables z_1 and z_2 to the frequency domain terms ω_1 and ω_2, we obtain the following expression for $M(\omega_1, \omega_2)$ as shown in (8.8):

$$\begin{aligned} M(\omega_1, \omega_2) = H_0\big[&\{p_{00} + p_{01}f_{01} + p_{02}f_{02} + p_{10}f_{10} + p_{20}f_{20} + p_{11}f_{11} \\ &+ p_{12}f_{12} + p_{21}f_{21} + p_{22}f_{22}\}/D - j\{p_{01}g_{01} + p_{02}g_{02} + p_{10}g_{10} \\ &+ p_{20}g_{20} + p_{11}g_{11} + p_{12}g_{12} + p_{21}g_{21} + p_{22}g_{22}\}/D\big] \end{aligned} \tag{8.8}$$

where

$$\begin{aligned} D = \big[&(1 + q_1f_{10} + r_1f_{01} + s_1f_{11}) - j(q_1g_{10} + r_1g_{01} + s_1g_{11})\big] \cdot \big[(1 + q_2f_{10} \\ &+ r_2f_{01} + s_2f_{11}) - j(q_2g_{10} + r_2g_{01} + s_2g_{11})\big], \end{aligned} \tag{8.9}$$

$$f_{xy}(\omega_1, \omega_2) = \cos(x\omega_1 + y\omega_2), \tag{8.10}$$

$$g_{xy}(\omega_1, \omega_2) = \sin(x\omega_1 + y\omega_2), \quad x, y = 0, 1, 2. \tag{8.11}$$

In a more compact form, $M(\omega_1, \omega_2)$ can be expressed as follows:

$$M(\omega_1, \omega_2) = H_0 \frac{A_R - jA_I}{(B_{1R} - jB_{1I})(B_{2R} - jB_{2I})} \tag{8.12}$$

where

$$\begin{aligned} A_R = &p_{00} + p_{01}f_{01} + p_{02}f_{02} + p_{10}f_{10} + p_{20}f_{20} + p_{11}f_{11} + p_{12}f_{12} + p_{21}f_{21} \\ &+ p_{22}f_{22}, \end{aligned} \tag{8.13}$$

$$A_I = p_{01}g_{01} + p_{02}g_{02} + p_{10}g_{10} + p_{20}g_{20} + p_{11}g_{11} + p_{12}g_{12} + p_{21}g_{21} + p_{22}g_{22}, \tag{8.14}$$

$$B_{1R} = 1 + q_1 f_{10} + r_1 f_{01} + s_1 f_{11}, \tag{8.15}$$

$$B_{1I} = q_1 g_{10} + r_1 g_{01} + s_1 g_{11}, \tag{8.16}$$

$$B_{2R} = 1 + q_2 f_{10} + r_2 f_{01} + s_2 f_{11}, \tag{8.17}$$

$$B_{2I} = q_2 g_{10} + r_2 g_{01} + s_2 g_{11}. \tag{8.18}$$

Thus, the constrained minimization task becomes:
Minimize

$$J = \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \left[\left| M\left(\frac{\pi n_1}{N_1}, \frac{\pi n_2}{N_2}\right) \right| - M_d\left(\frac{\pi n_1}{N_1}, \frac{\pi n_2}{N_2}\right) \right]^b \tag{8.19}$$

subject to the constraints imposed by (8.5)–(8.6) with $k = 1, 2$. The corresponding fitness function is defined as $f = 1/(J + eps)$ such that maximization of f results in minimization of J; eps is a small bias term having value 0.001.

8.3 An Outline of IWO Algorithm

Invasive Weed Optimization is a metaheuristic that is inspired by the colonizing actions of weeds. The biological processes used to model the ecological behavior of the weed population are mainly divided into the Initialization, Reproduction, Seed Dispersal, and Competitive Exclusion phases. They are briefly described in the following subsections.

8.3.1 Generation of an Initial Population

IWO starts with a population of NP D-dimensional parameter vectors, or weeds, representing the candidate solutions. We shall denote subsequent generations in IWO by $G = 0, 1, \ldots, G_{\max}$. We represent the ith vector of the population at the current generation as $\mathbf{X}_{i,G} = [x_{1,i,G}, x_{2,i,G}, \ldots, x_{D,i,G}]$.

The population members are initialized according to a uniform random distribution along every dimension, subject to the minimum and maximum bounds:

$$\begin{aligned} \mathbf{X}_{\min} &= \{x_{1,\min}, x_{2,\min}, \ldots, x_{D,\min}\} \quad \text{and} \\ \mathbf{X}_{\max} &= \{x_{1,\max}, x_{2,\max}, \ldots, x_{D,\max}\}. \end{aligned} \tag{8.20}$$

This ensures that for a reasonable number of vectors, the initial population at $G = 0$ covers the entire search space uniformly. Hence we may initialize the jth

component of the ith vector as

$$x_{j,i,0} = x_{j,\min} + \text{rand}_{i,j}(0, 1)(x_{j,\max} - x_{j,\min}) \tag{8.21}$$

where $\text{rand}_{i,j}(0, 1)$ is a uniformly distributed random number lying between 0 and 1 and is instantiated independently for each component of the ith vector. The following steps are taken next: Reproduction, Seed Dispersal, and Competitive Exclusion (in that order), which are explained in the following subsections.

8.3.2 Reproduction

The plants will produce seeds depending on their relative fitness which will be spread out over the problem space. Each seed, in turn, will grow into a flowering plant. Thus, if $S_{\max}$ and $S_{\min}$ denote the number of seeds produced by plants with the best and the worst fitness, respectively, then seed count of plants will increase linearly from $S_{\min}$ to $S_{\max}$ depending on their corresponding fitness values. The number of seeds produced by the ith weed $\mathbf{X}_{i,G}$ is therefore given by

$$s_{i,G} = \left\lfloor \frac{F_{\max,G} - f(\mathbf{X}_{i,G})}{F_{\max,G} - F_{\min,G}}(S_{\max} - S_{\min}) \right\rfloor \tag{8.22}$$

where $F_{\max,G}$ and $F_{\min,G}$ are the maximum and minimum fitness values at the Gth generation of the weed colony.

8.3.3 Dispersal of Seeds Through the Search Space

The produced seeds are randomly distributed over the D-dimensional search space by random numbers drawn from a normal distribution with zero mean but with a varying variance. However, the standard deviation (SD), σ, of the normal distribution decreases over the generations from an initial value, $\sigma_{\max}$, to a value, $\sigma_{\min}$, and is determined by the following equation,

$$\sigma = \left(\frac{G_{\max} - G}{G_{\max}}\right)^n (\sigma_{\max} - \sigma_{\min}) + \sigma_{\min} \tag{8.23}$$

where σ is the SD at the current generation and $G_{\max}$ is the maximum number of iterations while n is the nonlinear modulation index. This is the adaptation property of the algorithm.

8.3.4 Competitive Exclusion

If a plant does not reproduce, it will become extinct. Hence this leads to the requirement of a competitive exclusion in order to eliminate plants with low fitness values.

This is done to limit the maximum number of plants in the colony. Initially, fast reproduction of plants take place, and all the plants are included in the colony. The fitter plants reproduce more than the undesirable ones. The elimination mechanism is activated when the population exceeds a stipulated $NP_{\max}$. The plants and produced seeds are ranked together as a colony and plants with lower fitness values are eliminated to limit the population count to $NP_{\max}$. This is the selection property of the algorithm. The above steps are repeated until maximum number of iterations is reached.

8.4 Differential Q-Learning

Let us consider a given agent A. Let $S_1, S_2, \ldots, S_n$ be n possible states that can be exhibited by agent A. Each possible state is characterized by m possible actions $a_1, a_2, \ldots, a_m$. At a particular state–action pair, the specific reward that the agent can achieve is denoted by $r(S_i, a_j)$ and is referred to as "immediate reward" that the agent receives for executing action a_j at state S_i. The basic goal of classical Q-Learning is to choose the next action by a learning policy such that the cumulative reward that may be acquired by the agent during subsequent transition of states from its next state is maximized. The learning policy is achieved by updating Q-values at each state–action pair. The higher the Q-value, the higher will be the probability of selection of a particular action for an agent at a specified state.

Let the agent be in state S_i and is executing action a_j. Then the Q-value at state S_i due to action of a_j is updated by

$$Q(S_i, a_j) = r(S_i, a_j) + \gamma \max_{a'} Q\big(\delta(S_i, a_j), a'\big) \tag{8.24}$$

where $0 < \gamma < 1$ and $\delta(S_i, a_j)$ denotes the next state due to the selection of action a_j at state S_i. Let the next state selected be S_k.

However, several improvements to the classical Q-Learning algorithm have been proposed. Differential Q-learning is a modified version of Q-learning [27, 28]. In this approach, the Q-table update policy has the ability to remember the effect of past Q value of a particular state–action pair while updating the corresponding Q value. The modified Q update equation is given by

$$Q(S_i, a_j) \leftarrow (1-\alpha) Q(S_i, a_j) + \alpha\Big(r(S_i, a_j) + \gamma \max_{a'} Q\big(\delta(S_k, a_j), a'\big)\Big). \tag{8.25}$$

In this case, the Q-value $Q(S_i, a_j)$ is incremented when the action a_j led to a state $\delta(S_i, a_j)$ in which there exists an action a', such that the best possible Q-value $Q(\delta(S_i, a_j), a')$ in the next time step plus the achieved reward $r(S_i, a_j)$ is greater than the current value of $Q(S_i, a_j)$. α is called the "learning rate". A setting of $\alpha = 0$ would result in a trivial scenario where no learning behavior is exhibited by the agent, while $\alpha = 1$ would make the agent extremely greedy in terms of learning behavior, thereby providing emphasis only on the most recent information. The importance of future rewards is determined by the discount factor γ. Smaller values of

γ make the agent "opportunistic" while larger values make it strive for a long-term high reward.

8.5 IIWO: The Proposed Approach

The modified algorithm is based on the concept that fitter individuals should be involved in local search while the remaining plants should search the problem space globally at a particular generation. The classical IWO algorithm neglects this fact by assuming the same standard deviation σ for all the weeds in the seed dispersal stage. Although σ is made to decay through the successive generations, there is yet no provision for σ to attain low values for fitter individuals at a particular generation to enable the local search procedure. Local search is initiated only when the generation count has increased to a large value to ensure a low value of σ. Thus in classical IWO, all the weeds undergo a gradual behavioral transformation from an explorative to an exploitive one. In our proposed algorithm, we state that fitter individuals should behave in an exploitative manner through successive generations from the initialization of the weed colony and not wait for the standard deviation to reduce to low values. Following this concept, we introduce a constriction factor, η, in Eq. (8.23) as follows:

$$\sigma = \eta\left(\left(\frac{G_{\max} - G}{G_{\max}}\right)^n (\sigma_{\max} - \sigma_{\min}) + \sigma_{\min}\right) \tag{8.26}$$

where $\eta \in (0, 1]$. The proper choice of parameter η for different population members will help balance the explorative and exploitive capabilities of the individuals resulting in local refinement.

We employ a synergy of IWO and TDQL to realize an Adaptive Memetic Algorithm (AMA) for achieving superior performance in the filter design problem. Each evolutionary step is followed by a performance-based evaluation of the members. The individual population members receive reward/penalty based on their fitness and the Q-table is updated using the TDQL learning rule. A meme pool for parameter η is maintained from where the control parameters for individual members of the population are selected. The adaptive selection of memes is performed by a Roulette-Wheel selection from the meme pool. Selection of η from the meme pool, followed by one step of IWO and updating of the Q-table, is continued until the condition for convergence of the AMA is satisfied.

The row indices of the Q-table represent states of the population obtained from the last iteration of the IWO algorithm, where a member is allocated to a particular state using a fitness function based rank evaluation. The column indices correspond to the actions performed by the members at a particular state. They represent uniform quantized values of the control parameter in the range $(0, 1]$. For example, if the parameter under consideration be η with possible quantized values $\{\eta_1, \eta_2, \ldots, \eta_{10}\}$. Then $Q(S_i, \eta_j)$ represents the total reward given to a member at state S_i for selecting $\eta = \eta_j$. The Roulette-Choice strategy is used

to select a particular value of η from the meme pool $\{\eta_1, \eta_2, \ldots, \eta_{10}\}$ using the $Q(S_i, \eta_j)$, $j = 1, 2, \ldots, 10$ for the individual member located at state S_i.

The adaptation of $Q(S_i, \eta_j)$ is done through the reward/penalty mechanism of classical TDQL. If a member of the population at state S_i on selecting $\eta = \eta_j$ moves to a new state S_k causing an improvement in its fitness measure, then $Q(S_i, \eta_j)$ is given a positive reward following the TDQL algorithm. Otherwise a penalty is given to $Q(S_i, \eta_j)$ by introducing a decrease in Q-value.

The basic algorithm is outlined in the following sections.

8.5.1 Initialization

The algorithm employs a population of NP D-dimensional parameter vectors representing the candidate solutions. Thus the jth component of the ith population member is initialized according to (8.21) as mentioned in Sect. 8.3. The entries for the Q-table are initialized as small values. For instance, if the maximum Q-value attainable is 100, then we initialize the Q-values of all entries in the Q-table as 1.

8.5.2 Adaptive Selection of Memes

The proper choice of the Reinforcement Learning Scheme facilitates the adaptive selection of memes from the meme pool. We employ Fitness Proportional selection, also known as the Roulette-Wheel selection, for the selection of potentially useful memes. A basic advantage of this selection mechanism is that diversity of the meme population can be maintained. Although fitter memes would enjoy much higher probability of selection, the memes with poorer fitness do manage to survive and may contribute some components as evolution continues. Mathematically, the selection commences by the selection of a random number in the range [0, 1] for each population member. Let us consider the selection from the η meme pool for a member of state S_i. The next step involves the selection of η_j such that the cumulative probability of selection of $\eta = \eta_1$ through η_{j-1} is greater than r. Symbolically,

$$\sum_{m=1}^{j-1} p(S_i, \eta = \eta_m) < r < \sum_{m=j}^{10} p(S_i, \eta = \eta_m). \tag{8.27}$$

The probability of selection of $\eta = \eta_j$ from the meme pool $\{\eta_1, \eta_2, \ldots, \eta_{10}\}$ is given by

$$p(S_i, \eta = \eta_j) = \frac{Q(S_i, \eta_j)}{\sum_{k=1}^{10} Q(S_i, \eta_k)}. \tag{8.28}$$

8.5.3 Invasive Weed Optimization

The IWO algorithm used here employs reproduction, seed dispersal, and competitive exclusion as introduced in Sect. 8.3. The basic difference of the current realization is the selection of constriction factor η from the meme pool adaptively by Step 8.5.2 before invoking the IWO process.

8.5.4 State Assignment

The population members are now ranked in increasing order of fitness and assigned corresponding states. For example, a member of rank k is assigned the state S_k.

8.5.5 Updating the Q-table

Let a member at state S_i on selection of η_j move to a new state S_k. The update equation for $Q(S_i, \eta_j)$ is given by

$$Q(S_i, \eta_j) \leftarrow (1-\alpha)Q(S_i, \eta_j) + \alpha\Big(r(S_i, \eta_j) + \gamma \max_{\eta'} Q\big(\delta(S_k, \eta_j), \eta'\big)\Big). \quad (8.29)$$

The choice of the reward function is critical to the proper operation of the Q-learning mechanism. In case the seeds produced by a particular weed experience greater fitness in comparison to the parent weed, $r(S_i, \eta_j)$ is set to the absolute difference of fitness of the parent weed and the fittest seed. Otherwise a penalty of $-K$ is applied, however small.

The next step involves the determination of the factor $\max_{\eta'} Q(\delta(S_k, \eta_j), \eta')$. A particular weed may enter the next generation along with multiple seeds or it may be completely eliminated. In case of multiple state acquisition in the next generation, the factor is set equal to the maximum of $\max_{\eta'} Q(\delta(S_k, \eta_j), \eta')$ for all S_k's. Otherwise it is set equal to 0 in case of plant exclusion.

Sections 8.5.2–8.5.5 are repeated till the maximum number of iterations is reached.

8.6 The Filter Design Experiments and Results

Let us consider a specific example of the design problem [4], where the user-specification for the desired circular symmetric low-pass filter response is given

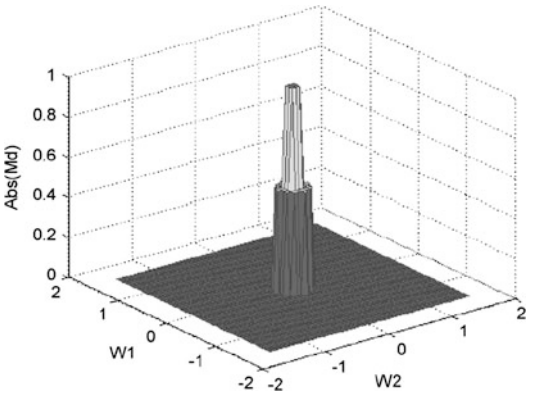


Fig. 8.1 Desired amplitude response of the 2D filter

by

$$\begin{aligned} M_d = 1 \quad & \text{if } \sqrt{\omega_1^2 + \omega_2^2} < 0.04\pi, \\ 0.5 \quad & \text{if } 0.04\pi < \sqrt{\omega_1^2 + \omega_2^2} < 0.08\pi, \\ 0 \quad & \text{otherwise.} \end{aligned} \tag{8.30}$$

The desired amplitude response of the 2D filter is shown in Fig. 8.1. For our experiment we choose $N_1 = 100$ and $N_2 = 100$. Results have been reported for $b = 1, 2, 4$, and 8. The efficiency of our approach to the filter design problem is demonstrated through comparisons of our results with state-of-the-art methodologies, namely MEPSO, G3 with PCX, and DE reported in [4]. The parameter settings for our proposed IIWO algorithm have been tabulated below in Table 8.1. They were set after a set of tuning experiments and were left unaltered for the entire simulation. The amplitude responses of the filters obtained by the above mentioned algorithms are shown in Fig. 8.2.

The problem constraints are handled using the method outlined in [6] as follows:

- Any feasible solution is preferred to any infeasible solution;
- Between two feasible solutions, the one with better fitness is preferred;
- Between two infeasible solutions, the one with a smaller constraint violation is preferred.

In order to test the accuracy of the IIWO algorithm, we ran it along with the competitor algorithms for 50,000 function evaluations. Each algorithm was executed for 30 independent runs. The mean of the objective function values J_b (where b denotes the value of the exponent) of the 30 independent runs are reported in Table 8.2. The best objective filter coefficients obtained with exponent $b = 2$ after 50,000 function evaluations for all the competitor algorithms have been presented in Table 8.3.

Finally, as an illustration of the performance of the designed low pass filters we demonstrate an image denoising experiment on the 256×256 gray scale image of Lenna. Denoising of digital images is one generic application of the lowpass

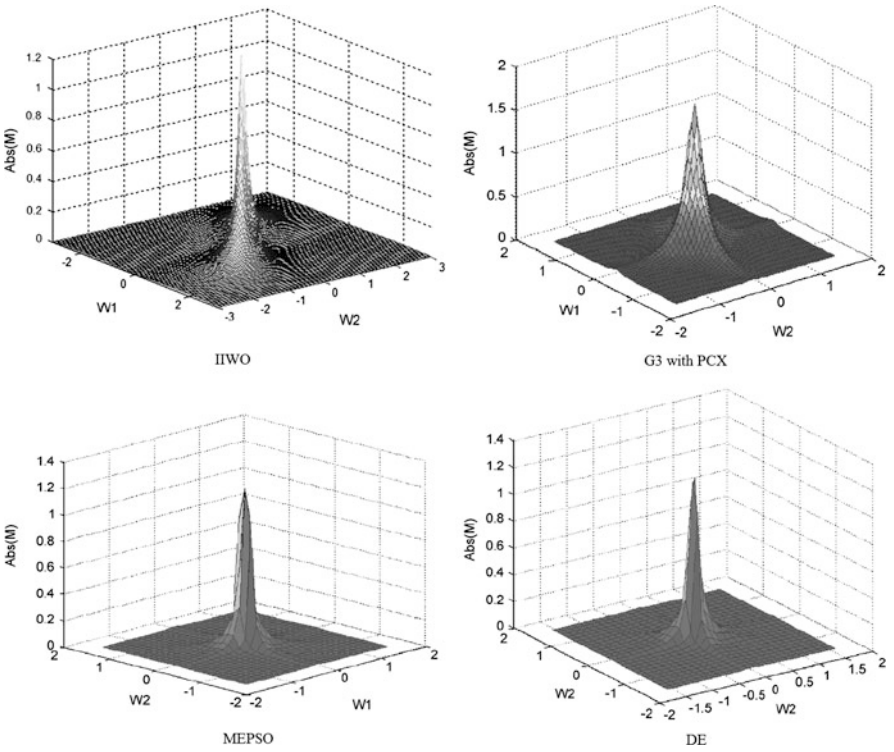


Fig. 8.2 Amplitude responses of the filters obtained by the various competitive algorithms

Table 8.1 Parameter settings for the IIWO algorithm

Parameters	Value
Initial Population	40
Maximum population	40
Maximum no. of seeds	10
Minimum no. of seeds	0
Std. dev. for seed dispersal	0.1–0.001
Reward parameter, α	0.25
Penalty parameter, γ	0.8

2D filters [9, 21, 22]. The original image is first corrupted with Gaussian noise of mean zero and variance 0.005. Then the image is transformed into the frequency domain using Fast Fourier Transform (FFT) and multiplied with the filter transfer function. The filtered image is then obtained by taking the inverse FFT (IFFT). The image processing techniques are performed using the MATLAB image processing toolbox. The results of image denoising by the filters designed by the competitive algorithms are shown in Fig. 8.3.

Fig. 8.3 Result of filtering with the designed filter: (**A**) original image "Lenna", (**B**) image corrupted with Gaussian noise, (**C**) filtering with G3 with PCX-based method, (**D**) filtering with DE based method, (**E**) filtering with the MEPSO-based method, (**F**) filtering with IIWO method

Table 8.2 Mean value of J with exponent $p = 1, 2, 4, 8$ after 50,000 FEs

Values of J for different exponents	MEPSO	G3 with PCX	DE	IIWO
J_1	60.3923	95.7113	98.5513	**42.5227**
J_2	9.0005	10.4252	11.9078	**6.4371**
J_3	0.5039	0.5732	2.9613	**0.3786**
J_4	0.0058	0.0178	0.2903	**0.0034**

Table 8.3 Filter coefficients obtained with exponent $p = 2$ after 50,000 FEs

Filter Coefficients	MEPSO	G3 with PCX	DE	IIWO
p_{01}	0.3061	−0.3016	−0.2426	−0.1652
p_{02}	0.9949	2.9023	2.4827	−0.7623
p_{10}	0.3935	−0.3435	−0.3484	2.7677
p_{11}	−0.0338	−2.0490	−2.0898	2.9921
p_{12}	0.6481	0.0387	0.0323	−1.6530
p_{20}	1.2345	2.4932	2.4915	−1.1049
p_{21}	0.5030	0.1975	0.1613	1.7266
p_{22}	0.4481	0.7493	0.7563	1.4750
q_1	−1.0239	−0.4738	−0.9113	−0.3247
q_2	0.0342	−0.0843	−0.0255	−0.3656
r_1	−0.9605	2.9493	2.9613	−0.3232
r_2	−0.0371	−0.0376	−0.0344	−0.1629
s_1	0.9523	0.8874	0.8674	−0.2719
s_2	−0.9056	−0.8476	−0.8075	−0.3672
H_0	0.00034	0.0784	0.0012	−0.0017

8.7 Conclusions

The chapter proposes a new method to the design problem of a zero-phase IIR digital filter. The proposed algorithm uses a synergy of Temporal Difference Q-Learning and Invasive Weed Optimization to realize an Adaptive Memetic Algorithm (AMA) that statistically outperformed the most recent and popular methods outlined in the existing literature [4, 8, 16, 18] in terms of performance accuracy and solution quality. Further, the superior quality of the recovered image as compared to the other competitive algorithms demonstrates how the designed 2D filter lends itself to well-known 2D IIR filter applications like image denoising.

Integration of such reinforcement learning schemes in the evolutionary platform is a completely new field of research [1, 24, 25], and further studies will involve comparative analysis of performance of the proposed memetic algorithm using other popular reinforcement learning schemes.

References

1. Bhowmik, P., Rakshit, P., Konar, A., Kim, E., Nagar, A.K.: DE-TDQL: an adaptive memetic algorithm. In: 2012 IEEE Congress on Evolutionary Computation (CEC) (2012)
2. Chen, X., Ong, Y.-S., Lim, M.-H., Tan, K.C.: A multi-facet survey on memetic computation. IEEE Trans. Evol. Comput. **15**(5), 591–607 (2011)
3. Daniel, M., Willsky, A.: Efficient implementations of 2-D noncausal IIR filters. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **44**(7), 549–563 (1997)
4. Das, S., Konar, A.: A swarm intelligence approach to the synthesis of two-dimensional IIR filters. Eng. Appl. Artif. Intell. **20**, 1086–1096 (2007)
5. Dawkins, R.: The Selfish Gene. Oxford University Press, Oxford (1976)
6. Deb, K.: An efficient constraint handling method for genetic algorithms. Comput. Methods Appl. Mech. Eng. **186**, 311–338 (1998)
7. Dumitrescu, B.: Optimization of two-dimensional IIR filters with non-separable and separable denominator. IEEE Trans. Signal Process. **53**(5), 1768–1777 (2005)
8. Gonos, I.F., Virirakis, L.I., Mastorakis, N.E., Swamy, M.N.S.: Evolutionary design of 2-dimensional recursive filters via the computer language GENETICA. IEEE Trans. Circuits Syst. II, Express Briefs **53**(4), 254–258 (2006)
9. Gonzales, R.C., Woods, R.E.: Digital Image Processing, 3rd edn. Addison-Wesley, Reading (1992)
10. Hsieh, C.-H., Kuo, C.-M., Jou, Y.-D., Han, Y.-L.: Design of two-dimensional FIR digital filters by a two-dimensional WLS technique. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **44**(5), 348–412 (1997)
11. Kaczorek, T.: Two-Dimensional Linear Systems. Springer, Berlin (1985)
12. Kendall, G., Cowling, P., Soubeiga, E.: Choice function and random hyperheuristics. In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, pp. 667–671 (2002)
13. Laasko, T., Ovaska, S.: Design and implementation of efficient IIR notch filters with quantization error feedback. IEEE Trans. Instrum. Meas. **43**(3), 449–456 (1994)
14. Lu, W.-S., Antoniou, A.: Two-Dimensional Digital Filters. Marcel Dekker, New York (1992)
15. Maria, G.A., Fahmy, M.M.: An LP design technique for two-dimensional digital recursive filters. IEEE Trans. Acoust. Speech Signal Process. **ASSP-22**(1), 15–21 (1974)
16. Mastorakis, N.E., Gonos, I.F., Swamy, M.N.S.: Design of 2-dimensional recursive filters using genetic algorithms. IEEE Trans. Circuits Syst. I, Fundam. Theory Appl. **50**(5), 634–639 (2003)
17. Mehrabian, A.R., Lucas, C.: A novel numerical optimization algorithm inspired from weed colonization. Ecol. Inform. **1**, 355–366 (2006)
18. Mladenov, V., Mastorakis, N.: Design of two-dimensional recursive filters by using neural networks. IEEE Trans. Neural Netw. **12**(3), 585–590 (2001)
19. Moscato, P.: On evolution, search, optimization, genetic algorithms and martial arts: towards memetic algorithms. In: Caltech Concurrent Computation Program (report 826)
20. Ong, Y.-S., Lim, M.H., Zhu, N., Wong, K.-W.: Classification of adaptive memetic algorithms: a comparative study. IEEE Trans. Syst. Man Cybern. **36**, 1 (2006)
21. Oppenheim, A.V., Schafer, R.W., Buck, J.R.: Discrete-Time Signal Processing. Prentice-Hall, Englewood Cliffs (1999)
22. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing. Prentice-Hall, Englewood Cliffs (1996)
23. Rajan, P.K., Swamy, M.N.S.: Quadrantal symmetry associated with two-dimensional digital transfer functions. IEEE Trans. Circuits Syst. **CAS-29**(6), 340–343 (1983)
24. Sengupta, A., Chakraborti, T., Konar, A., Kim, E., Nagar, A.K.: An adaptive memetic algorithm using a synergy of differential evolution and learning automata. In: 2012 IEEE Congress on Evolutionary Computation (CEC) (2012)

25. Sengupta, A., Chakraborti, T., Konar, A., Nagar, A.K.: A multi-objective memetic optimization approach to the circular antenna array design problem. Prog. Electromagn. Res. B **42**, 363–380 (2012)
26. Tzafestas, S.G. (ed.): Multidimensional Systems, Techniques and Applications. Marcel Dekker, New York (1986)
27. Watkins, C.: Learning from delayed rewards. PhD dissertation, King's College, Cambridge, England (1989)
28. Watkins, C., Dayan, P.: Q-learning. Mach. Learn. **8**, 279–292 (1992)
29. Zhu, W.-P., Ahmad, M.O., Swamy, M.N.S.: A closed-form solution to the least-square design problem of 2-D linear-phase FIR filters. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **44**(12), 1032–1039 (1997)

Chapter 9
A Survey of Kurtosis Optimization Schemes for MISO Source Separation and Equalization

Marc Castella, Eric Moreau, and Vicente Zarzoso

Abstract Blind source separation and equalization aim at recovering a set of unknown source signals from their linearly distorted mixtures observed at a sensor array output, with little or no prior knowledge about the sources or the distorting channel. This fundamental signal processing problem arises in a broad range of applications such as multiuser digital communications, biomedical data analysis, and seismic exploration. Put forward over three decades ago, the normalized fourth-order cumulant, also known as kurtosis, has arguably become one of the most popular blind source separation and equalization criteria. Using multiple-input single-output (MISO) filter structures for single source extraction combined with suitable deflation procedures, the kurtosis contrast yields separation algorithms free of spurious extrema in ideal system conditions. The lack of closed-form solutions, however, calls for numerical optimization schemes. The present chapter reviews some of the iterative algorithms most widely used for MISO source separation and equalization based on kurtosis. These include gradient and Newton search methods, algorithms with optimal step-size selection, as well as techniques based on reference signals. Their main features are briefly summarized and their performance is illustrated by some numerical experiments in digital communications and biomedical signal processing.

M. Castella (✉)
Institut Mines-Télécom/Télécom SudParis, CNRS UMR 5157 SAMOVAR, 9 Rue Charles Fourier, 91011 Evry Cedex, France
e-mail: marc.castella@telecom-sudparis.eu

E. Moreau
University of the South Toulon-Var, ISITV, LSIS UMR-CNRS 7296, av. G. Pompidou, BP 56, 83162 La Valette du Var Cedex, France
e-mail: moreau@univ-tln.fr

V. Zarzoso
I3S Laboratory, University of Nice Sophia Antipolis, CNRS, Les Algorithmes, Euclide-B, 2000 route des Lucioles, BP 121, 06903 Sophia Antipolis Cedex, France
e-mail: zarzoso@i3s.unice.fr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_9,

9.1 Introduction

In this section, we first present the blind channel equalization and source separation problems and make a point about the important practical use of kurtosis as a source extraction criterion. A brief historical overview follows together with the chapter outline and notations.

9.1.1 Channel Equalization and Source Separation

Numerous signal processing applications involve extracting signals of interest from their observed mixtures, possibly corrupted by propagation effects and additive noise. Instances of this problem abound in array signal processing, where a receiver sensor array may capture contributions from multiple sources of simultaneous activity originating from different space locations. The source signals are generally unknown and so is the mixing physical system. The estimation of the source signals with no prior knowledge of the mixing system parameters or the source signals themselves (e.g., pilot sequences in communications) is the so-called *blind source separation* (*BSS*) problem [22].

From a signals and systems perspective, the observations can be considered as the output of a *multiple-input multiple-output* (*MIMO*) filter, which is assumed to be linear throughout this chapter. Two important cases can be distinguished. In instantaneous mixtures, propagation effects mainly reduce to amplitude attenuation, but otherwise the sources do not suffer any significant distortion prior to mixing. This scenario occurs in narrowband propagation conditions, where time delays are negligible or can just be approximated by phase shifts. In convolutive mixtures, the sources undergo temporal distortions that can often be represented by the convolution with the channel impulse response, assumed to behave as a linear time invariant (LTI) filter. Convolution can be regarded as a mixture of the source with time-delayed replicas of itself. This case models more severe effects such as multipath propagation and limited channel bandwidth in wideband source environments. Disentangling convolutive mixtures requires the time *equalization* of the source contributions in addition to their spatial separation. In either case, instantaneous or convolutive, the goal of BSS is inverting the MIMO channel to recover the sources. The related objective of identifying the channel is sometimes useful in certain applications, but will not be addressed in the sequel.

BSS is an important problem in wireless digital communications, where signals emitted from different mobile users have to be equalized at the base station both temporally and spatially in order to eliminate inter-symbol interference and co-channel user interference. Blind processing spares the transmission of training sequences, with the consequent benefits in spectral efficiency. Source separation is also of relevance in biomedical applications, where sources of physiological information (e.g., firing patterns) appear mixed at the electrode output in electromyogram and electroencephalogram recordings. Likewise, the separation of cardiac activity sources

that mix together in the electrocardiogram can provide useful information about a patient's heart condition. Applications related to source separation extend far beyond signal processing to encompass domains as diverse as factor analysis, mechanical system diagnosis, or financial forecast, among others.

The separation process can be mainly carried out in two fashions. In joint or symmetric separation schemes, the sources are estimated simultaneously with a MIMO separation filter having the observed mixtures as inputs and the estimated sources as outputs. This approach usually leads to rather elaborate separation algorithms, whose convergence properties are difficult to study. In deflationary separation, by contrast, one source is extracted at a time using a *multiple-input single-output* (*MISO*) filter. This separation scheme requires an additional processing stage called *deflation* to avoid extracting the same source several times. The deflation approach, however, leads to simpler separating algorithms easier to analyze theoretically and presenting nice convergence properties. Such is the case of the source extraction methods studied in this chapter.

Source signals can be classified as random (stochastic) or deterministic, and are characterized accordingly using different tools. Random sources require statistical characterizations, often by means of moments or cumulants, the latter corresponding to specific nonlinear combinations of the former [60]. Based on these statistical properties, a large diversity of separation principles or *contrast functions* have been proposed in the literature, whereby the separation is achieved by the filters maximizing such contrasts [10, 12, 15–17, 22, 31]. The case of mutually independent sources has drawn considerable interest, since statistical independence is a plausible assumption in many practical applications. The normalized fourth-order cumulant, or *kurtosis*, constitutes one of the most widespread contrasts for the extraction of independent sources in linear mixtures, as reviewed in the remaining of this chapter.

9.1.2 Why Kurtosis?

Higher-order statistics (i.e., moments and cumulants with orders higher than two) have extensively been used for the separation of mutually independent non-Gaussian sources. Their success in performing BSS in this scenario hinges heavily on the concepts of statistical independence and non-Gaussianity, linked by the Central Limit Theorem, and their ability to measure both. The Central Limit Theorem states that a linear mixture of independent non-Gaussian random variables is more Gaussian than the original variables; accordingly, one should proceed in the opposite direction, i.e., increasing non-Gaussianity, to undo the mixture and separate the variables. Since the higher-order marginal cumulants of a Gaussian variable are null, maximizing their absolute value leads to sensible principles for separating spatial and/or temporal mixtures, of which kurtosis is a leading example. Indeed, MISO extraction filters can be found by optimizing the kurtosis contrast function.

Despite its potential lack of robustness to outliers [34], kurtosis has been widely employed for source separation and equalization for many years. Though also proposed as part of joint separation criteria [19, 23, 40, 42–47, 51, 61], this contrast

becomes particularly interesting when used for MISO processing. Its most attractive feature in this context—which may probably explain the interest it has aroused over more than two decades—lies in the absence of spurious local extrema under ideal model conditions. As a result, *global* convergence to right source extraction solutions is guaranteed even by means of local optimization algorithms. This highly desirable property was first proven in [58] in the context of blind single-channel equalization, and later in [26] for BSS in real-valued instantaneous mixtures after prewhitening. The proof was finally extended to the more general convolutive complex mixture case in [59, 64]; see also [27] for the instantaneous scenario. The good convergence properties of later algorithms such as the multiuser kurtosis optimization of [50] are actually inherited from the decoupling of the MIMO criterion into a set of MISO extraction criteria through a deflation approach. As reviewed in the chapter, a good number of cost-effective iterative algorithms are available for kurtosis maximization.

Besides its mathematical tractability and computational convenience, kurtosis proves more robust to finite sample effects than related criteria such as the fourth-order moment or the fourth-order cumulant [5, 6]. This interesting property is especially useful when processing short data records.

9.1.3 Historical Overview

Originally proposed by Wiggins [66] and Donoho [28] for single-channel deconvolution in the context of seismic exploration, the use of kurtosis for interference cancelation and signal recovery quickly spread to other application domains such as digital communications, biomedical signal processing, image denoising, and speech enhancement, as well as more involved signal models. Its application in digital channel equalization dates back to the work of Shalvi and Weinstein [58], who proved its validity as a blind deconvolution criterion for the non-Gaussian distributions typically encountered in communications and proposed gradient algorithms for kurtosis maximization based on spectral prewhitening. Extensions to multichannel models soon followed. The criterion was proposed for BSS by Delfosse–Loubaton [26] and Papadias [50] using second-order sphering or prewhitening, and by Tugnait [64] even without prewhitening. Connections with the popular constant modulus criterion for blind equalization, which had been developed a few years earlier in [33, 57, 63], were realized by Comon [20, 21] and Regalia [53]; see also [77]. In its original definition, Hyvärinen's popular FastICA algorithm for BSS based on independent component analysis also relied on the kurtosis criterion [36]; the algorithm was independently developed by Moreau in [40]. More recent developments include monotonically convergent algorithms optimizing quadratic contrasts based on reference signals [10, 12, 13, 16] as well as parameter-free iterative algorithms with algebraic optimal step-size selection [74].

9.1.4 Chapter Outline

This chapter summarizes the basic concepts behind the use of kurtosis as an MISO source separation and equalization criterion, and reviews some practical numerical algorithms proposed in the literature for kurtosis optimization. Our focus is on the deflationary separation of statistically independent sources in linear mixtures, which are assumed noiseless for the sake of simplicity. Although adaptive (online, recursive, sample-by-sample) algorithms have also been devised, our interest lies primarily in batch (offline, windowed, block) algorithms that reuse a whole set of observed signal samples at each iteration. As stated in [1], batch processing leads to statistically more efficient implementations, from which adaptive versions can often be obtained with simple changes.

After presenting the BSS signal model and assumptions in Sect. 9.2, the general deflation procedure based on the kurtosis criterion is introduced in Sect. 9.3. Section 9.4, the core of the chapter, reviews a number of iterative algorithms for kurtosis contrast maximization. A few experimental results illustrating the performance of such methods are reported in Sect. 9.5. Finally, Sect. 9.6 summarizes the main results of the chapter and points out some possible avenues of further research.

9.1.5 Mathematical Notations

Before beginning the exposition, defining some mathematical notations will be useful. Throughout the chapter, unless otherwise stated, signals can be complex- or real-valued. The letter n stands for a generic integer, $n \in \mathbb{Z}$. Lightface (x, X), boldface lowercase ($\mathbf{x}$), and boldface uppercase ($\mathbf{X}$) characters denote scalar, vector, and matrix quantities, respectively. The transpose and Hermitian (conjugate-transpose) operators are denoted by superscripts $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$. The cumulant of a set of random variables is represented by $\mathrm{Cum}\{\cdot\}$. In particular, the fourth-order marginal cumulant of a zero-mean random variable y is given by:

$$\mathrm{C}\{y\} \triangleq \mathrm{Cum}\{y, y^*, y, y^*\} = \mathrm{E}\{|y|^4\} - 2\mathrm{E}\{|y|^2\}^2 - |\mathrm{E}\{y^2\}|^2 \tag{9.1}$$

where $\mathrm{E}\{\cdot\}$ represents the mathematical expectation. Finally, symbols $\star$ and $\cdot$ stand for, respectively, the convolution operator and the scalar product.

9.2 Blind Source Separation: Model and Assumptions

In this section, we introduce the different mixing models considered in this chapter. The assumptions used to perform BSS are also given.

9.2.1 Convolutive Mixtures

We consider an observed Q-dimensional ($Q \in \mathbb{N}$, $Q \geq 2$) discrete-time signal $\mathbf{x}(n)$ following the linear model

$$\mathbf{x}(n) \triangleq \mathbf{M}(n) \star \mathbf{s}(n) = \sum_{p \in \mathbb{Z}} \mathbf{M}(p)\mathbf{s}(n-p). \tag{9.2}$$

Hence, the observed data $\mathbf{x}(n)$ are the output of the convolutive MIMO channel represented by the $(Q \times N)$ matrix impulse response $\mathbf{M}(n)$ excited by the unknown N-dimensional ($N \in \mathbb{N}$, $N \geq 2$, $N \leq Q$) source input $\mathbf{s}(n)$. The (i, j)th entry of the matrix $\mathbf{M}(n)$, denoted $m_{ij}(n)$, represents the scalar channel transforming source $s_j(n)$ before adding its contribution to mixture $y_i(n)$. The above model is assumed noise-free. The objective of BSS is to restore the sources by exploiting the observations alone, without any knowledge of the MIMO channel $\mathbf{M}(n)$ and the sources $\mathbf{s}(n)$. Clearly, some further assumptions are necessary to prevent this problem from being ill-posed.

A first type of assumptions concerns the convolutive mixing system $\mathbf{M}(n)$. We assume that it admits a left inverse or MIMO separating filter $\mathbf{W}(n)$ such that

$$\mathbf{y}(n) \triangleq \mathbf{W}(n) \star \mathbf{x}(n) = \sum_{p \in \mathbb{Z}} \mathbf{W}(p)\mathbf{x}(n-p) \tag{9.3}$$

recovers all sources in the separator output $\mathbf{y}(n)$. Since the source ordering and spectral profiles cannot be identified by using criteria based on statistical independence only, the separation is considered successful whenever the global system

$$\mathbf{G}(n) \triangleq \mathbf{W}(n) \star \mathbf{M}(n) = \sum_{p \in \mathbb{Z}} \mathbf{W}(p)\mathbf{M}(n-p) \tag{9.4}$$

is of the form

$$\mathbf{G}(n) = \mathbf{D}(n)\mathbf{P} \tag{9.5}$$

where $\mathbf{D}(n)$ is an invertible diagonal convolutive MIMO system modeling the source spectral ambiguity, and $\mathbf{P}$ a permutation matrix modeling the source ordering ambiguity. Remark that these ambiguities are inherent to blind processing and are acceptable in most applications. When the sources are independent and identically distributed (i.i.d.), the scalar filtering represented by the diagonal entries of $\mathbf{D}(n)$ reduces to a simple delay $n_i \in \mathbb{Z}$ with a possible scale factor $d_{ii} \in \mathbb{C}$, and so $d_{ii}(n) = d_{ii}\delta_{n-n_i}$, where δ_n denotes Dirac's discrete delta function; see, e.g., [16, 17, 22, 59] for more details.

Another set of assumptions concerns the source signals:

A1. For all i, the source sequence $s_i(n)$ is stationary and zero-mean. In addition, the fourth-order cumulants, $\mathrm{C}\{s_i\}$, exist and are assumed to be nonzero.

A2. The source processes $s_i(n)$, $i \in \{1, \ldots, N\}$, are mutually statistically independent.

These assumptions allow the separation of independent non-Gaussian sources using kurtosis.

In all the following, we will focus on the MISO approach to BSS, which consists in extracting one source after another. This is done by estimating one row of $\mathbf{W}(n)$, denoted by $\mathbf{w}(n)$, in such a way that the extractor output $y(n) = \mathbf{w}(n) \star \mathbf{x}(n)$ corresponds to one source up to a possible scalar filtering; this step is detailed in Sect. 9.3.1. If a full separation is to be accomplished, a deflation stage has to be applied before searching for a new source, as will be described in Sect. 9.3.2.

9.2.2 Instantaneous Mixtures

In the instantaneous mixture case, channel effects reduce to scale factors without time delays, so that the MIMO channel is given by $\mathbf{M}(n) = \mathbf{M}\delta_n$. As a result, the LTI systems in Eqs. (9.2)–(9.4) reduce to constant matrices, and the respective convolution operations become matrix products:

$$\mathbf{x}(n) = \mathbf{M}\mathbf{s}(n), \qquad \mathbf{y}(n) = \mathbf{W}\mathbf{x}(n), \qquad \mathbf{G} = \mathbf{W}\mathbf{M}.$$

Similarly, when dealing with MISO separation, the extracted source output reads $y(n) = \mathbf{w}\mathbf{x}(n)$, where $\mathbf{w}$ is a constant row vector corresponding to one row of matrix $\mathbf{W}$.

A similar matrix model holds when considering finite impulse response (FIR) equalizers, a practical setting to deal with convolutive mixtures. If the separating filter $\mathbf{W}(n)$ is represented by an order-R causal FIR MIMO filter, the summation index in separation equation (9.3) extends from 0 to R. Hence, the convolution can be expressed as the matrix product:

$$\mathbf{y}(n) = \big[\mathbf{W}(0), \mathbf{W}(1), \ldots, \mathbf{W}(R)\big]\mathbf{x}_{R+1}(n)$$

where vector $\mathbf{x}_{R+1}(n)$ is obtained by stacking $(R+1)$ consecutive delays of the observed vector $\mathbf{x}(n)$: $\mathbf{x}_{R+1}(n) \triangleq [\mathbf{x}(n)^{\mathrm{T}}, \mathbf{x}(n-1)^{\mathrm{T}}, \ldots, \mathbf{x}(n-R)^{\mathrm{T}}]^{\mathrm{T}}$. Thanks to the equivalent matrix model enabled by the stacking device, results presented later in the chapter for the instantaneous case can readily be extended to the convolutive case with FIR equalizers. More details can be found, e.g., in [13, 14, 16].

9.3 Deflationary Source Separation

This section introduces the general methodology of the deflation-based BSS approach considered in this chapter. The approach iterates between the following two fundamental steps:

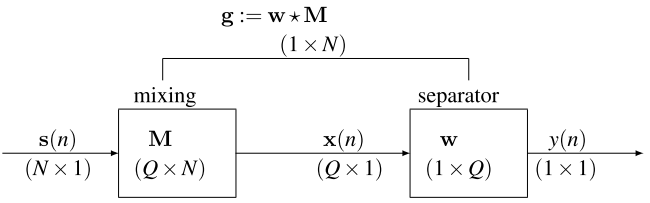


Fig. 9.1 Schematic diagram of the signal model considered in this chapter

Extraction aims at estimating one source from the observed mixture by using a MISO filter maximizing a suitable contrast function, as detailed in Sect. 9.3.1.

Deflation aims at canceling out the contribution of the source estimated in the previous step, so that the number of sources contributing to the mixture decreases by one. Section 9.3.2 explains how to perform this step.

Using the deflated mixture obtained in the second step, the algorithm goes back to the first step to search for another source, and so forth. The procedure is repeated until all sources have been estimated (full separation) or the source of interest has been recovered.

9.3.1 Source Extraction with MISO Contrast Functions

The MISO source extraction problem consists in estimating one row of the separating filter $\mathbf{W}(n)$ in Eq. (9.3). The entries of this row vector, called vector equalizer and denoted $\mathbf{w}(n)$, represent a bank of Q scalar LTI filters with impulse responses $\{w_i(n)\}_{i=1}^{Q}$. The output of the separation procedure is the scalar signal

$$y(n) = \mathbf{w}(n) \star \mathbf{x}(n) = \sum_{i=1}^{Q} w_i(n) \star x_i(n). \tag{9.6}$$

Defining the global LTI vector filter $\mathbf{g}(n) \triangleq \mathbf{w}(n) \star \mathbf{M}(n)$, with impulse response $\mathbf{g}(n) = \sum_{p\in\mathbb{Z}} \mathbf{w}(p)\mathbf{M}(n-p)$, we then have

$$y(n) = \mathbf{g}(n) \star \mathbf{s}(n) = \sum_{i=1}^{N} g_i(n) \star s_i(n). \tag{9.7}$$

Vector $\mathbf{g}(n)$ is a row of the global matrix defined in Eq. (9.4). These notations are summed up in Fig. 9.1. According to the ambiguities described in Sect. 9.2.1, a successful extraction restores one of the source components $s_i(n), i \in \{1, \ldots, N\}$, possibly up to scalar filtering: $y(n) = d_{ii}(n) \star s_i(n)$. The global filter $\mathbf{g}(n)$ thus contains a single nonzero entry, $g_i(n) = d_{ii}(n)$, $g_j(n) = 0$, $j \neq i$, and is actually a row

of global matrix (9.5) modeling ideal separation conditions. Vectors $\mathbf{g}(n)$ with such a structure are referred to as *trivial filters*.

The MISO separating filter $\mathbf{w}(n)$ can be obtained by maximizing specific criteria called *contrast functions*. By definition, the contrast attains its maximum value when $\mathbf{w}(n)$ is a separating filter or, equivalently, when the global vector $\mathbf{g}(n)$ is a trivial filter. More detailed explanations on the properties that define contrast functions can be found in [41]. The extractor output *kurtosis* is defined as the normalized fourth-order cumulant

$$\mathscr{J}(\mathbf{w}) \triangleq \frac{\mathrm{C}\{y(n)\}}{\mathrm{E}\{|y(n)|^2\}^2} \tag{9.8}$$

where the dependence of $\mathscr{J}$ on $\mathbf{w}(n)$ is through Eq. (9.6). Note that we have used the shorthand notation $\mathbf{w}$ to represent the coefficients of vector filter $\mathbf{w}(n)$. Then we have the following fundamental results:

Proposition 1 (Kurtosis contrast) *Under assumptions* A1–A2, *the absolute kurtosis function*

$$\mathscr{J}_{\kappa}(\mathbf{w}) \triangleq \left|\mathscr{J}(\mathbf{w})\right| \tag{9.9}$$

is a contrast for MISO source extraction in the model defined by Eqs. (9.2) *and* (9.6). *Similarly, under the same conditions, the function*

$$\mathscr{J}_{\varepsilon}(\mathbf{w}) \triangleq \varepsilon\, \mathscr{J}(\mathbf{w}) \tag{9.10}$$

is a contrast for the extraction of a source with kurtosis sign ε.

The proof of these results is omitted here for reasons of space, as our focus is on the optimization algorithms of Sect. 9.4. Suffice it to say that the validity of this contrast has been proved in several scenarios, including the i.i.d. context [64] and the non-i.i.d. context [59]; the instantaneous case has specifically been addressed in [26, 27, 50]; see [22] for other references. As introduced in Sect. 9.1.2, one of the major interests of MISO contrasts as the above is that they are free of local spurious maxima. More precisely, any local maximum corresponds to a valid separating filter.

Remark 1 Due to the normalizing term in Eq. (9.8), the kurtosis contrast is scale invariant, i.e., $\mathscr{J}(\alpha\mathbf{w}) = \mathscr{J}(\mathbf{w})$, $\forall\alpha \in \mathbb{C}\backslash\{0\}$. In other words, the function $\mathscr{J}$ is homogeneous of degree zero. This means that $\mathscr{J}$ is unaffected by a possible amplitude scaling of $y(n)$. However, due to the source scale ambiguity (Sect. 9.2), we are free to impose the unit power constraint $\mathrm{E}\{|y(n)|^2\} = 1$, which is equivalent to a unit norm constraint on the global filter $\mathbf{g}(n)$: $\sum_i \sum_n |g_i(n)|^2 = 1$.

Remark 2 Alternatively to the unconstrained optimization of the kurtosis contrast, one could maximize the fourth-order cumulant in the numerator of (9.8) under the constraint $\mathrm{E}\{|y(n)|^2\} = 1$ (see, e.g., [14]; also [58] in the single-channel case). However, as will be explained in Sect. 9.4, optimizing the normalized contrast with simultaneous unit power scaling is actually a requirement in practical numerical implementations.

9.3.2 Deflation Procedure

This section briefly describes the deflation method proposed in [39]; the alternative deflation scheme of [26] will be presented in Sect. 9.4.3. We assume that a possibly filtered version of source $s_p(n)$ has been recovered in extractor output $y(n)$ by means of a suitable MISO approach (Sect. 9.3.1). We show here that one can then subtract the contribution of $s_p(n)$ to the observations and, in doing so, the original mixture of N sources simplifies to a smaller mixture of $(N-1)$ sources. More precisely, we have the following result:

Proposition 2 (Deflation criterion) *Consider the linear mixture model* (9.2), *which, with evident definitions, can be expressed as*

$$\mathbf{x}(n) = \sum_{i=1}^{N} \mathbf{m}_i(n) \star s_i(n).$$

For a given $p \in \{1, \ldots, N\}$, *let* $y(n) = g_p(n) \star s_p(n)$ *be a scalar filtering of source* $s_p(n)$. *Define the following adjusting filter*:

$$\mathbf{t}^\star(n) = \arg\min_{\mathbf{t}(n)} \mathrm{E}\big\{\big\|\mathbf{x}(n) - \mathbf{t}(n) \star y(n)\big\|^2\big\}. \tag{9.11}$$

Then

$$\mathbf{x}(n) - \mathbf{t}^\star(n) \star y(n) = \sum_{i \neq p} \mathbf{m}_i(n) \star s_i(n). \tag{9.12}$$

Proof We can write

$$\begin{aligned}\mathbf{x}(n) - \mathbf{t}(n) \star y(n) &= \sum_{i=1}^{N} \mathbf{m}_i(n) \star s_i(n) - \mathbf{t}(n) \star y(n) \\ &= \big[\mathbf{m}_p(n) - \mathbf{t}(n) \star g_p(n)\big] \star s_p(n) + \sum_{i \neq p} \mathbf{m}_i(n) \star s_i(n).\end{aligned}$$

Exploiting the source independence assumption A2 and the linearity of the expectation operator, we have

$$\begin{aligned}&\mathrm{E}\big\{\big\|\mathbf{x}(n) - \mathbf{t}(n) \star y(n)\big\|^2\big\} \\ &\quad = \mathrm{E}\big\{\big\|\big[\mathbf{m}_p(n) - \mathbf{t}(n) \star g_p(n)\big] \star s_p(n)\big\|^2\big\} + \sum_{i \neq p} \mathrm{E}\big\{\big\|\mathbf{m}_i(n) \star s_i(n)\big\|^2\big\}.\end{aligned}$$

The filter $\mathbf{t}(n)$ minimizing the above expression cancels out $[\mathbf{m}_p(n) - \mathbf{t}(n) \star g_p(n)]$, and result (9.12) readily follows. □

According to the above proposition, replacing the original observations with $\mathbf{x}(n) - \mathbf{t}^{\star}(n) \star y(n)$ reduces the mixture of N sources to a mixture of $(N-1)$ sources only. For this result to hold, $y(n)$ is required to contain a filtered version of a source signal, which can be obtained by maximizing a MISO contrast as shown in the previous section. Combining these two ideas yields the following generic deflationary source separation algorithm:

General algorithm for deflationary source separation

- Set $\mathbf{x}^{(1)}(n) = \mathbf{x}(n)$.
- For $p = 1, 2, \ldots, (N-1)$, do:
  1. *Extraction:* From the observations $\mathbf{x}^{(p)}(n)$, determine an estimate $y_p(n)$ of one source signal up to admissible ambiguities using a suitable MISO contrast such as (9.9)–(9.10).
  2. *Deflation:* Deflate the observations by removing the contribution of the estimated source:
     (a) Find a column vector filter $\mathbf{t}_p^{\star}(n)$ satisfying

$$\mathbf{t}_p^{\star}(n) = \arg\min_{\mathbf{t}(n)} \mathrm{E}\left\{ \left\| \mathbf{x}^{(p)}(n) - \mathbf{t}(n) \star y_p(n) \right\|^2 \right\}.$$

     (b) Define the deflated observations $\mathbf{x}^{(p+1)}(n)$ as follows:

$$\mathbf{x}^{(p+1)}(n) = \mathbf{x}^{(p)}(n) - \mathbf{t}_p^{\star}(n) \star y_p(n).$$

- Estimate the last source as an arbitrary MISO filtered version of $\mathbf{x}^{(N)}(n)$.

In practice, one often deals with FIR filters and the above problem amounts to the least squares solution of a linear system. Remark that in practical settings such as noisy or short sample size scenarios, the sources can only be estimated with some inaccuracies, and then the error term to be minimized in step (9.11) cannot be perfectly canceled. As a result, estimation errors accumulate through successive deflation iterations [14]. This error propagation is probably the main drawback of the deflation approach. In the remaining of the chapter, we turn our attention to practical algorithms for optimizing the kurtosis contrast in Step 1 of the above general algorithm.

9.4 Optimization Methods

As seen in Sect. 9.3.1, MISO source extraction can be accomplished by finding the extraction filters maximizing the kurtosis contrast. This problem lacks closed-form

solutions and, as a consequence, it requires iterative numerical algorithms. This section provides a survey of iterative techniques for kurtosis maximization proposed in the literature. These include gradient-based algorithms (Sect. 9.4.1) possibly including some form of projection (Sect. 9.4.2) or parametrization of the separation system (Sect. 9.4.3). Newton search is also considered, the popular FastICA algorithm with cubic nonlinearity being arguably the most popular example (Sect. 9.4.4). The kurtosis-based FastICA can actually be recast as a gradient algorithm with constant step size, which motivates the development of more elaborate algorithms with optimal selection of the step-size parameter (Sect. 9.4.5). Our review concludes with techniques based on reference signals leading to quadratic criteria that can be optimized by algorithms with monotonic convergence (Sect. 9.4.6).

The contrasts under study depend on the MISO filter $\mathbf{w}(n)$, but can also be considered as functions of the global MISO filter $\mathbf{g}(n) = \mathbf{w}(n) \star \mathbf{M}(n)$. For ease of notation, the corresponding filters will just be denoted in the sequel without reference to time index n. We focus on the maximization of the absolute kurtosis contrast $\mathscr{J}_\kappa$ (Eq. (9.9)), the treatment of $\mathscr{J}_\varepsilon$ (Eq. (9.10)) being totally analogous.

9.4.1 Gradient Search Algorithms

Since the seminal works establishing kurtosis as a deconvolution criterion [28, 58, 66], a wide variety of blind separation and equalization methods based on this contrast have been put forward using gradient optimization [40, 50, 53, 58, 64, 74]. The idea consists in taking small steps in the direction of the gradient:

$$\mathbf{w}^+ = \mathbf{w} + \mu \nabla \mathscr{J}_\kappa(\mathbf{w}) \tag{9.13}$$

where $\mathbf{w}^+$ is the updated extracting vector and symbol ∇ denotes the nabla, or gradient vector, operator of first-order partial derivatives with entries $[\nabla \mathscr{J}_\kappa(\mathbf{w})]_i = \partial \mathscr{J}_\kappa(\mathbf{w})/\partial w_i$. From the first-order Taylor expansion of $\mathscr{J}_\kappa$ around $\mathbf{w}$, and taking into account update (9.13), we have:

$$\mathscr{J}_\kappa\big(\mathbf{w}^+\big) \approx \mathscr{J}_\kappa(\mathbf{w}) + \nabla \mathscr{J}_\kappa(\mathbf{w})^{\mathrm{T}}\big(\mathbf{w}^+ - \mathbf{w}\big) = \mathscr{J}_\kappa(\mathbf{w}) + \mu \big\| \nabla \mathscr{J}_\kappa(\mathbf{w}) \big\|^2.$$

Hence, a finite sufficiently small positive value of μ guarantees $\mathscr{J}_\kappa(\mathbf{w}^+) \geq \mathscr{J}_\kappa(\mathbf{w})$, with equality if and only if $\nabla \mathscr{J}_\kappa(\mathbf{w}) = \mathbf{0}$, i.e., when the algorithm has reached a stationary point of $\mathscr{J}_\kappa$. Likewise, a negative μ would allow the local minimization of the contrast. In the instantaneous case, the absolute kurtosis gradient is given by

$$\begin{aligned} \nabla \mathscr{J}_\kappa(\mathbf{w}) = \frac{4\,\mathrm{sign}(\mathscr{J}(\mathbf{w}))}{\mathrm{E}^2\{|y|^2\}} \Big\{ & \mathrm{E}\big\{|y|^2 y^* \mathbf{x}\big\} - \mathrm{E}\{y\mathbf{x}\}\mathrm{E}\big\{y^{*2}\big\} \\ & - \frac{(\mathrm{E}\{|y|^4\} - |\mathrm{E}\{y^2\}|^2)\mathrm{E}\{y^*\mathbf{x}\}}{\mathrm{E}\{|y|^2\}} \Big\}. \end{aligned} \tag{9.14}$$

This leads to the following algorithm:

Gradient algorithm for kurtosis optimization

- Set an initial value $\mathbf{w}^{(0)}$ for the extracting vector.
- For $k = 1, 2, \ldots, k_{\max}$, do:
  1. Compute the gradient direction $\mathbf{d}^{(k-1)} = \nabla \mathscr{J}_\kappa(\mathbf{w}^{(k-1)})$ from Eq. (9.14).
  2. Compute an appropriate step size μ and update $\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} + \mu \mathbf{d}^{(k-1)}$.

The real-valued parameter μ is known as the *learning rate*, *step size,* or *adaptation coefficient*, and is assumed to be constant in classical gradient algorithms. The optimal selection of this parameter will be the subject of Sect. 9.4.5.

9.4.2 Projected Gradient Search

Let us consider the kurtosis contrast (9.9) as a function of the global filter $\mathbf{g}$, which we can express as

$$\mathscr{J}_\kappa(\mathbf{g}) = \frac{\mathscr{C}(\mathbf{g})}{\|\mathbf{g}\|^4}$$

where $\mathscr{C}(\mathbf{g}) = |\mathrm{C}\{y(n)\}|$ is the absolute fourth-order cumulant of the extractor output in the numerator of (9.8), considered as a function of $\mathbf{g}$. The function $\mathscr{C}$ is homogeneous of degree four, i.e., $\mathscr{C}(\alpha\mathbf{g}) = \alpha^4\mathscr{C}(\mathbf{g})$, $\forall \alpha > 0$. From Euler's homogeneous function theorem, it follows that $\mathbf{g} \cdot \nabla\mathscr{C}(\mathbf{g}) = 4\mathscr{C}(\mathbf{g})$, $\forall \mathbf{g}$, which proves that the gradient of $\mathscr{C}$ has a nonzero radial component. In the vicinity of a local maximum of $\mathscr{C}$, according to the Karush–Kuhn–Tucker conditions, the tangent component of $\nabla\mathscr{C}(\mathbf{g})$ tends to zero and the ratio of the radial to the tangent part of $\nabla\mathscr{C}(\mathbf{g})$ tends to infinity. For numerical reasons, this is not acceptable in an iterative algorithm. Recall indeed that, due to the scale ambiguity (Sect. 9.2), one can impose a unit norm constraint on the global filter $\mathbf{g}$, and hence only the tangent part of $\nabla\mathscr{C}(\mathbf{g})$ is of importance. If this tangent part is too small with respect to the radial part, it appears as a numerical error. This justifies that, although the maximization of $\mathscr{J}_\kappa$ is theoretically equivalent to the constrained maximization of $\mathscr{C}$ (as noted in Remark 2), the former must be used in practice (see, e.g., [14, 16, 17, 59]).

The optimization of $\mathscr{J}_\kappa$, however, must be carried out with some care. As noted in Remark 1, the contrast $\mathscr{J}_\kappa$ is scale invariant, or homogeneous of degree zero. Using Euler's homogeneous function theorem again, we have $\mathbf{w}^{\mathrm{T}}\nabla\mathscr{J}_\kappa(\mathbf{w}) = 0$, i.e., the gradient of $\mathscr{J}_\kappa$ at $\mathbf{w}$ is orthogonal to $\mathbf{w}$. Since the unit ball $\|\mathbf{w}\| \leq 1$ is a convex set, a step in direction of the gradient vector will always yield a point outside the unit ball. As a result, the extracting vector norm will monotonically increase at each iteration of any gradient algorithm: $\|\mathbf{w}^+\| \geq \|\mathbf{w}\|$, where the equality will hold at

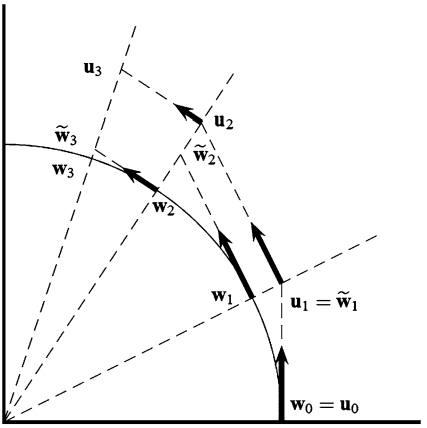


Fig. 9.2 Schematic comparison of the successive points obtained with gradient and projected gradient algorithms. For $k = 0, 1, 2, 3$, the sequence $\mathbf{u}_k$ is generated by a gradient algorithm, whereas $\widetilde{\mathbf{w}}_k$ is generated by a projected gradient update before renormalization. Sequence $\mathbf{w}_k$ is generated by a projected gradient iteration

convergence. This drift is illustrated by Fig. 9.2, and may become unacceptable due to numerical overflow whenever a great number of gradient iterations are required. This undesired phenomenon can be prevented by a normalizing step after the gradient update, e.g., by projecting the extracting vector on the unit sphere, thus yielding the so-called projected gradient algorithm summarized below:

Projected gradient algorithm for kurtosis optimization

- Set an initial value $\mathbf{w}^{(0)}$ for the extracting vector.
- For $k = 1, 2, \ldots, k_{\max}$, do:
  1. Compute the gradient direction $\mathbf{d}^{(k-1)} = \nabla \mathscr{J}(\mathbf{w}^{(k-1)})$ from Eq. (9.14).
  2. Compute an appropriate step size μ and update $\widetilde{\mathbf{w}} = \mathbf{w}^{(k-1)} + \mu \mathbf{d}^{(k-1)}$.
  3. Project the update as $\mathbf{w}^{(k)} = \widetilde{\mathbf{w}}/\|\widetilde{\mathbf{w}}\|$, or any other suitable form of normalization.

It is important to remark that, thanks to the contrast's scale invariance, the normalization step does not affect the contrast function value attained at the gradient update step.

9.4.3 Gradient Algorithm with Filter Parametrization

In the instantaneous case, the rank of the observation covariance matrix decreases by one after each deflation step and, consequently, the dimension of the observation space can be reduced without losing information. By performing dimensionality reduction, the search for the next source can be carried out in a lower-dimensional

parameter space, thus leading to computational savings and faster convergence. One of the early kurtosis maximization algorithms for instantaneous BSS in real-valued mixtures is based on an ingenious parametrization of the separating matrix allowing dimensionality reduction at the deflation step [26], and can be summarized as follows.

The method relies on a preliminary prewhitening step leading to linearly transformed observations $\mathbf{z}(n) \in \mathbb{R}^N$ with identity covariance matrix. Under the source independence and unit-variance assumption, one can easily see that the whitened observations are linked to the sources through an unknown orthogonal transformation $\mathbf{Q} \in \mathbb{R}^{N\times N}$, resulting in the observation model

$$\mathbf{z} = \mathbf{Q}\mathbf{s}. \tag{9.15}$$

Source separation is then achieved from the whitened observations through a particular deflation approach. This approach relies on the decomposition of matrix $\mathbf{Q}$ in terms of Givens planar rotations $\bar{\mathbf{Q}}_{i,j}(\theta)$, defined as an identity matrix except for entries (i,i), (i,j), (j,i), and (j,j), $1 \le i < j \le N$, which are given by

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}.$$

More precisely, $\mathbf{Q}$ is decomposed as

$$\mathbf{Q}(\boldsymbol{\theta}) = \mathbf{Q}_{N-1}(\theta_{N-1})\mathbf{Q}_{N-2}(\theta_{N-2})\cdots\mathbf{Q}_1(\theta_1)$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_{N-1}]^{\mathrm{T}}$, with $\theta_i \in]-\pi/2, \pi/2[$, $1 \le i \le (N-1)$, and $\mathbf{Q}_i(\theta_i) = \bar{\mathbf{Q}}_{i,N}(\theta_i)$. Matrix $\mathbf{Q}$ can be further split into two terms:

$$\mathbf{Q}(\boldsymbol{\theta}) = \begin{bmatrix} \widetilde{\mathbf{Q}}(\boldsymbol{\theta}) & \mathbf{q}(\boldsymbol{\theta}) \end{bmatrix}$$

in which $\widetilde{\mathbf{Q}}(\boldsymbol{\theta}) \in \mathbb{R}^{N\times(N-1)}$ and

$$\mathbf{q}(\boldsymbol{\theta}) = [\sin\theta_1, \cos\theta_1\sin\theta_2, \ldots, \cos\theta_1\cdots\cos\theta_{N-2}\sin\theta_{N-1}, \cos\theta_1\cdots\cos\theta_{N-2}\cos\theta_{N-1}]^{\mathrm{T}} \in \mathbb{R}^N$$

represents the extracting vector for the source currently targeted as

$$y = \mathbf{q}^{\mathrm{T}}\mathbf{z}. \tag{9.16}$$

Angular parameters $\boldsymbol{\theta}$ are estimated through a gradient update, much like those summarized in the previous sections. More importantly, by the structure of the mixing matrix after prewhitening, the extracting vector $\mathbf{q}(\boldsymbol{\theta})$ lies orthogonal to all columns of matrix $\widetilde{\mathbf{Q}}(\boldsymbol{\theta})$, $\forall\boldsymbol{\theta}$. As a result, the vector $\tilde{\mathbf{z}} \triangleq \widetilde{\mathbf{Q}}^{\mathrm{T}}(\boldsymbol{\theta})\mathbf{z} \in \mathbb{R}^{N-1}$ is uncorrelated with y, the source extracted by $\mathbf{q}(\boldsymbol{\theta})$. Hence, to extract the next source, the algorithm can be repeated using $\tilde{\mathbf{z}}$ instead of $\mathbf{z}$ and reducing the dimensions of $\boldsymbol{\theta}$ accordingly. The uncorrelation of $\tilde{\mathbf{z}}$ and y prevents the same source from being extracted again. This dimensionality reduction, achieved by the particular parametrization of the orthogonal mixing matrix after prewhitening in the real-valued case, reduces the computational cost after each deflation stage.

9.4.4 Approximate Newton Search: The FastICA Algorithm

One of the most popular methods for kurtosis optimization is the FastICA algorithm [35, 36], see also [40], which is based on Newton rather than gradient updates. After deriving the algorithm in Sect. 9.4.4.1, we provide an interpretation as a gradient-like method in Sect. 9.4.4.2.

9.4.4.1 Derivation of the Algorithm

Newton methods are based on the second-order Taylor approximation of the contrast around the current point $\mathbf{w}$:

$$\mathscr{J}_\kappa\big(\mathbf{w}^+\big) \approx \mathscr{J}_\kappa(\mathbf{w}) + \nabla \mathscr{J}_\kappa(\mathbf{w})^{\mathrm{T}}\big(\mathbf{w}^+ - \mathbf{w}\big) + \frac{1}{2}\big(\mathbf{w}^+ - \mathbf{w}\big)^{\mathrm{T}}\mathbf{H}(\mathbf{w})\big(\mathbf{w}^+ - \mathbf{w}\big) \quad (9.17)$$

where $\mathbf{H}(\mathbf{w})$ represents the Hessian matrix of the second-order derivatives, with elements $[\mathbf{H}(\mathbf{w})]_{ij} = \partial^2 \mathscr{J}_\kappa(\mathbf{w})/\partial w_i \partial w_j$. The Newton update selects the vector $\mathbf{w}^+$ that cancels out the gradient of the second-order approximation on the left-hand side of Eq. (9.17), yielding

$$\mathbf{w}^+ = \mathbf{w} - \mathbf{H}(\mathbf{w})^{-1}\nabla \mathscr{J}_\kappa(\mathbf{w}). \quad (9.18)$$

As compared to gradient update (9.13), no parameter needs to be fine-tuned here but, in exchange, the Hessian matrix needs to be inverted at each iteration, which can be costly and may introduce numerical instabilities. Hessian inversion is probably the main drawback of Newton methods.

Matrix inversion can sometimes be avoided, and Newton methods consequently simplified, by approximating the Hessian matrix, as is the case with FastICA. The algorithm considers the real-valued mixture scenario after prewhitening, with observation model (9.15) and extraction equation (9.16), as in the method described in the previous section. By constraining the extracting vector to lie on the unit sphere, $\|\mathbf{q}\| = 1$, the extractor output is guaranteed to fulfill the unit-variance normalization convention, $\mathrm{E}\{y^2\} = 1$. Under such assumptions, the absolute kurtosis contrast (9.9) simplifies into $\tilde{\mathscr{J}}_\kappa(\mathbf{q}) = |\mathscr{J}_f(\mathbf{q}) - 3|$, where

$$\mathscr{J}_f(\mathbf{q}) = \mathrm{E}\big\{y^4\big\} \quad (9.19)$$

is the fourth-order moment of the extractor output; similar contrasts based on the fourth-order moment had also been proposed in [9, 26]. The gradient and Hessian of this simplified contrast are given, respectively, by the expressions:

$$\nabla \tilde{\mathscr{J}}_\kappa(\mathbf{q}) = 4\,\mathrm{sign}\big(\mathscr{J}_f(\mathbf{q})\big)\mathrm{E}\big\{y^3\mathbf{z}\big\}, \quad (9.20)$$

$$\tilde{\mathbf{H}}(\mathbf{q}) = 12\,\mathrm{sign}\big(\mathscr{J}_f(\mathbf{q})\big)\mathrm{E}\big\{y^2\mathbf{z}\mathbf{z}^{\mathrm{T}}\big\}. \quad (9.21)$$

The sign terms in the above equations can be omitted since they cancel out when combined in a Newton update like (9.18). The Hessian is further approximated as follows. At a valid extraction solution, say $\mathbf{q}^*$, the extractor output equals a source component, $y \approx s_i$. The Hessian at that point is then $\tilde{\mathbf{H}}(\mathbf{q}^*) = 12\mathrm{E}\{s_i^2\mathbf{z}\mathbf{z}^\mathrm{T}\}$ (up to an irrelevant sign) and, by virtue of the prewhitening assumption, $\mathrm{E}\{s_i^2\mathbf{z}\mathbf{z}^\mathrm{T}\} = \mathbf{Q}\mathrm{E}\{s_i^2\mathbf{s}\mathbf{s}^\mathrm{T}\}\mathbf{Q}^\mathrm{T}$, where $\mathbf{Q}$ is the unitary transformation linking the sources to the whitened observations in Eq. (9.15). The final simplification assumes that

$$\mathrm{E}\{s_i^2\mathbf{s}\mathbf{s}^\mathrm{T}\} \approx \mathrm{E}\{s_i^2\}\mathrm{E}\{\mathbf{s}\mathbf{s}^\mathrm{T}\} = \mathbf{I}. \tag{9.22}$$

As a result, the Hessian reduces to an identity matrix. In combination with Eq. (9.18), these simplifications lead to the approximate Newton update

$$\mathbf{q}^+ = \mathbf{q} - \frac{1}{3}\mathrm{E}\{y^3\mathbf{z}\}. \tag{9.23}$$

This is followed by normalization of the extracting vector (projection on the unit sphere):

$$\mathbf{q}^+ \leftarrow \mathbf{q}^+ / \|\mathbf{q}^+\| \tag{9.24}$$

which is necessary to fulfill the assumption $\|\mathbf{q}\| = 1$ imposed by prewhitening. The orthogonality of the extracting vectors in the whitened observation subspace (see Eq. (9.15)) enables a simplified deflation method, the so-called *deflationary orthogonalization* [35, 40]. In this alternative procedure, the updated extracting vector is projected onto the orthogonal complement of the subspace spanned by the already estimated extracting vectors, as in Gram–Schmidt orthogonalization. In that case, the deflation method described in Sect. 9.3.2 is no longer necessary. The kurtosis-based FastICA algorithm is summarized in the table below.

FastICA algorithm for kurtosis optimization

- Sphere the observed signals to obtain the whitened observations $\mathbf{z}$.
- Set an initial value $\mathbf{q}^{(0)}$ for the extracting vector in the whitened space.
- For $k = 1, 2, \ldots, k_{\max}$, do:
  1. Update: $\widetilde{\mathbf{q}} = \mathrm{E}\{(y^{(k-1)})^3\mathbf{z}\} - 3\mathbf{q}^{(k-1)}$, with $y^{(k-1)} = (\mathbf{q}^{(k-1)})^\mathrm{T}\mathbf{z}$.
  2. Normalize: $\mathbf{q}^{(k)} = \widetilde{\mathbf{q}}/\|\widetilde{\mathbf{q}}\|$.

Whitening can be performed by a number of techniques, including the singular value decomposition (SVD) of the observed data matrix or the eigenvalue decomposition (EVD) of the covariance matrix. Note that the update used in the above algorithm description is equivalent to Eq. (9.23), due to the source sign indeterminacy and the normalization step.

A very attractive property of FastICA is its cubic global convergence under ideal system conditions (noiseless infinite sample observations perfectly fulfilling the instantaneous linear model) [36, 40]. This desirable feature, which helps to explain the method's success, is revisited next.

9.4.4.2 FastICA as a Constant Step-Size Gradient Algorithm

As a parameter-free technique, the Newton approach has the potential to avoid the convergence problems that may result from an unfortunate choice of step-size in gradient algorithms. In fact, comparing Eq. (9.13), (9.20), and (9.23), we realize that the approximate Newton update can be regarded as a gradient-descent iteration to minimize the fourth-order moment $\mathscr{J}_f(\mathbf{q})$ (Eq. (9.19)). Moreover, this implicit gradient iteration uses a fixed step size $\mu = -1/12$.

Although this specific value for the step-size parameter may seem arbitrary, it is actually instrumental in endowing FastICA with the desirable cubic global convergence property under ideal system conditions. To prove this claim, let us consider an update of the form (9.23) with a generic but otherwise constant step size, say ν, rather than the value $-1/3$ used in FastICA's iteration. First recall that the global filter $\mathbf{g} = \mathbf{Q}^{\mathrm{T}}\mathbf{q}$ links the extractor output with the sources as $y = \mathbf{g}^{\mathrm{T}}\mathbf{s}$, where $\|\mathbf{g}\| = 1$, since $\|\mathbf{q}\| = 1$ due to prewhitening (Sect. 9.4.4.1). In terms of $\mathbf{g}$, the resulting update would read:

$$\mathbf{g}^{+} = \mathbf{g} + \nu \mathrm{E}\{y^3\mathbf{s}\} = \left[\mathbf{I} + \nu \mathrm{E}\left\{\left(\mathbf{g}^{\mathrm{T}}\mathbf{s}\right)^2 \mathbf{s}\mathbf{s}^{\mathrm{T}}\right\}\right]\mathbf{g}. \tag{9.25}$$

The second equality stems from the fact that $\mathrm{E}\{y^3\mathbf{s}\} = \mathrm{E}\{y^2\mathbf{s}(\mathbf{s}^{\mathrm{T}}\mathbf{g})\} = \mathrm{E}\{y^2\mathbf{s}\mathbf{s}^{\mathrm{T}}\}\mathbf{g}$. The crucial step to prove FastICA's cubic global convergence is showing that the updates induced in the entries of $\mathbf{g}$ by the above equation are uncoupled from each other and have a cubic-only dependence, i.e., that

$$g_i^{+} = \alpha_i g_i^3 \tag{9.26}$$

for all entries $i = 1, 2, \ldots, N$ of vector $\mathbf{g}$, with at least one of the coefficients α_i different from zero. Focusing on a generic entry g_i, Eq. (9.25) yields:

$$g_i^{+} = g_i + \nu \sum_{j=1}^{N} h_{ij} g_j \tag{9.27}$$

where $h_{ij} \triangleq \mathrm{E}\{y^2 s_i s_j\}$ is the (i, j)th element of matrix $\mathrm{E}\{y^2\mathbf{s}\mathbf{s}^{\mathrm{T}}\}$, which can be expressed as the quadratic form $h_{ij} = \mathbf{g}^{\mathrm{T}}\mathrm{E}\{s_i s_j \mathbf{s}\mathbf{s}^{\mathrm{T}}\}\mathbf{g}$. Under the source statistical independence assumption A2, we can easily show that

$$h_{ij} = \begin{cases} \sum_{k \neq i} g_k^2 + (\kappa_i + 3) g_i^2, & i = j, \\ 2 g_i g_j, & i \neq j, \end{cases} \tag{9.28}$$

where κ_i represents the kurtosis of the ith source. Inserting this expression into Eq. (9.27), and after some algebraic manipulations keeping in mind that $\|\mathbf{g}\| = 1$, we easily arrive at

$$g_i^+ = (1 + 3\nu)g_i + \nu\kappa_i g_i^3. \tag{9.29}$$

Hence, a relationship of the form (9.26), and thus FastICA's cubic convergence, can be achieved only if $\nu = -1/3$, which corresponds precisely to the coefficient used in update rule (9.23) obtained by the simplified Newton iteration with identity Hessian matrix. Besides the specific step size value used in its update equation, another key ingredient of FastICA's excellent convergence properties is the apparently trivial normalization step (9.24). It is indeed this step that allows the global filter to fulfill $\|\mathbf{g}\| = 1$ after each iteration, thus allowing the simplification of Eq. (9.27) through Eq. (9.28) into Eq. (9.29). Remark, however, that the objective function (9.19) implicitly optimized by FastICA is a valid contrast for real-valued sources and mixtures under prewhitening. The use of prewhitening imposes some performance bounds on further higher-order processing, as analyzed in [8].

The desirable convergence properties of FastICA only hold asymptotically, i.e., under infinite sample size conditions, when the noiseless observation model is perfectly satisfied [29, 36]. Nevertheless, when processing short observation windows the fourth-order contrast presents higher sample variance than the kurtosis for a range of source distributions (including sub- and super-Gaussian), leading to separation estimates farther from the optimal solution [5, 6]. A simple numerical example helps to illustrate this limitation. Figure 9.3 plots the contrast function values against angle $\Delta\theta$ parametrizing the global filter $\mathbf{g} = [\cos(\Delta\theta), \sin(\Delta\theta)]^{\mathrm{T}}$ for an instantaneous orthogonal mixture of two zero-mean unit-variance uniformly distributed sources, with an observation window of just $T = 50$ samples. Separation solutions are defined by integer multiples of $\pi/2$ rad, recovering the sources up to the sign and permutation ambiguities inherent to blind processing (Sect. 9.2). Clearly, the local minima of the sample fourth-order moment (dashed line) lie farther away from the separation solutions than the maxima of the sample absolute kurtosis (solid line); in addition, the minima near 0 and $\pm\pi$ rad become saddle points, which tend to slow down the algorithm's convergence. Comparing panels (a) and (b) shows that FastICA converges to different solutions depending on the initial value of the extracting vector, requiring in each case nearly 30 iterations [74]. The theoretical large-sample performance of FastICA has been analyzed in [62], including a solution to prevent the detrimental effects of saddle points. This solution, however, is only valid in the version of the algorithm designed for joint or simultaneous source separation rather than single source extraction.

The version of the FastICA algorithm reviewed above is designed for real-valued sources and mixtures only. An extension to complex signals was carried out in [7], and was later shown to inherit the cubic global convergence property of its real-valued counterpart [56]. Such an extension, however, is valid only for sources satisfying the second-order circularity condition $\mathrm{E}\{s_i^2\} = 0$. The non-circular source scenario is specifically addressed in [30, 38, 48, 49], all under the prewhitening assumption. Interestingly, the alternative version of the algorithm independently developed a decade earlier in [40] comprised the complex non-circular case, too.

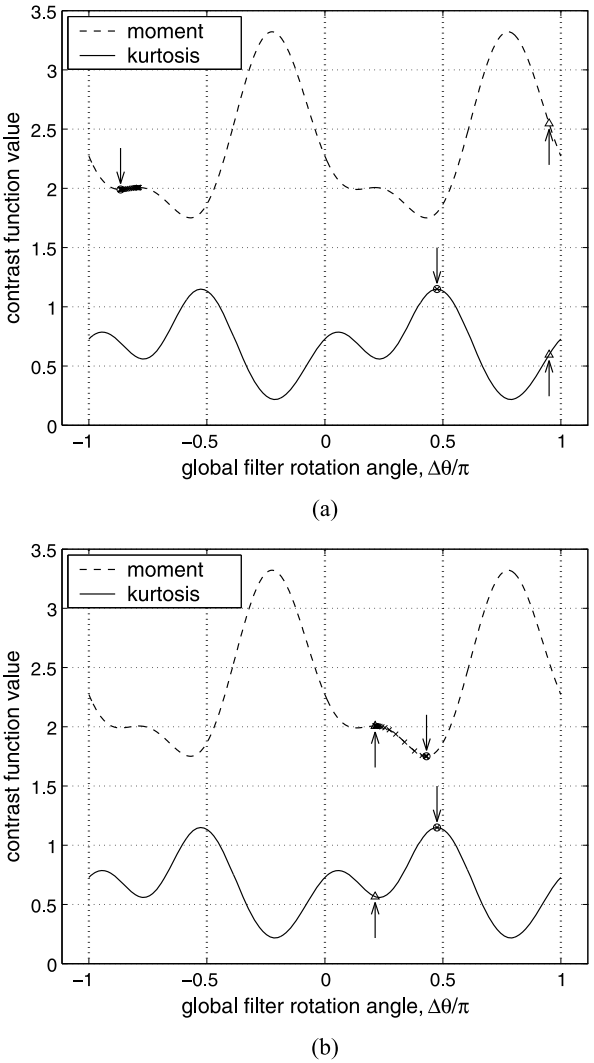


Fig. 9.3 Kurtosis-based MISO contrast function values and iterative algorithm trajectories. Instantaneous orthogonal mixture realization of two uniformly distributed sources composed of $T = 50$ samples. *Dashed lines*: fourth-order moment contrast (9.19) implicitly minimized by FastICA (Sect. 9.4.4.1). *Solid lines*: absolute kurtosis contrast (9.9) maximized by RobustICA (Sect. 9.4.5.2). *Triangle markers* and *upward arrows*: initial positions. *Cross markers*: algorithms' solutions after each iteration. *Round markers* and *downward arrows*: final solutions. *Vertical dotted lines*: satisfactory separation solutions up to sign and permutation. Panels (**a**)–(**b**) correspond to two different extracting vector initializations over the same mixture realization

9.4.5 Algorithms with Optimal Step-Size Selection

In this section, we present how the step size for the above gradient algorithms can be derived in an optimal way.

9.4.5.1 Step-Size Optimization

The step-size parameter μ used in gradient updates (see, e.g., Eq. (9.13)) sets a difficult trade-off between the convergence speed and accuracy of the resulting iterative

algorithm. A very small absolute value of μ theoretically guarantees monotonic convergence to a local stationary point of the contrast, but convergence may be too slow since the update also takes very small steps in the gradient direction. To speed up the algorithm, μ may be increased, but then the algorithm may oscillate around a local extremum without settling down, a phenomenon known as *misadjustment* [37], or even risk divergence. This speed–quality trade-off is common to both batch (operating over a signal block) and adaptive (stochastic, recursive, sample-by-sample) implementations of gradient-based optimization algorithms [1, 2, 4].

Many works in the literature have been devoted to making an optimal, or at least judicious, choice of the step size, aiming at fast convergence with low misadjustment. A classical approach consists in starting the iterations with a large value of μ and then decreasing it progressively as the algorithm converges. However, detecting whether the algorithm approaches the right solution is a problem far from trivial, which often hinders the performance of this simple idea. Newton updates are in theory parameter-free, yet, as seen earlier in the chapter for the FastICA algorithm, they can actually reduce to constant step-size gradient iterations. A fact long unnoticed but pointed out in [20, 21] was recently developed in [71, 74, 76]: the shape of the kurtosis contrast enables the closed-form computation of the optimal step-size value at each extracting vector update. The resulting algorithm is described next.

9.4.5.2 Algebraic Exact Line Search: The RobustICA Algorithm

A simple approach to addressing the trade-off set by the learning coefficient in gradient algorithms is exact line search. This optimization technique aims at the step size leading to the *global* optimum of the contrast along the current search direction:

$$\mu_{\text{opt}} = \arg\max_{\mu} \mathscr{J}_{\kappa}(\mathbf{w} + \mu\mathbf{d}) \tag{9.30}$$

where $\mathbf{d}$ typically represents the gradient vector at $\mathbf{w}$, or any other suitably chosen direction. In most cases, this one-dimensional optimization is costly, as it usually requires iterative numerical methods [52]. However, when the contrast is a rational function or a polynomial, the search for the optimal step size can be notably simplified [20, 21].

Although this property is satisfied by most functions based on fourth-order statistics, using the full version of the kurtosis contrast (9.9) presents some attractive advantages relative to simplified versions such as the fourth-order moment [74]:

- The kurtosis is a valid source extraction contrast even if prewhitening is not performed [64]. Avoiding prewhitening prevents the performance limitations imposed by this second-order processing step [8].
- The kurtosis is a valid contrast in both real- and complex-valued mixture scenarios, so that both cases can be treated without any modification. Both types of sources can appear simultaneously in a given mixture, and the mixing matrix entries can also be real or complex. Complex sources do need not to be circularly distributed.

- The kurtosis shows a reduced sample variance in comparison with other contrasts of the same order [5, 6]. This property translates into an increased numerical stability in short sample scenarios.

In addition to these properties intrinsic to the kurtosis contrast, the optimal step-size iterative maximization technique described below presents the following computational advantages:

- By construction, the algorithm offers the possibility of avoiding saddle points and spurious local extrema that may arise when processing short observation windows.
- The algorithm presents a high convergence speed, as measured in terms of source extraction quality achieved for a given number of operations. In the basic real-valued two-source case, the algorithm converges in just a single iteration, even without prewhitening.
- With simple modifications allowing the maximization of contrast $\mathscr{J}_\varepsilon$ (9.10), the method can be designed to target sub-Gaussian ($\varepsilon < 1$) or super-Gaussian ($\varepsilon > 1$) sources, as defined by the kurtosis sign. This feature may spare the cost of a full separation in scenarios where only a specific source or set of sources is actually of interest.

To derive the optimal step-size iterative maximization of the absolute kurtosis contrast (9.9), we first notice that its stationary points are the same as those of $\mathscr{J}$ in Eq. (9.8). Also, $\mathscr{J}$ evaluated at $\mathbf{w} + \mu\mathbf{d}$, with fixed $\mathbf{w}$ and $\mathbf{d}$, depends on μ only, and is given by the rational function:

$$\mathscr{J}(\mu) = \frac{\mathrm{E}\{|y^+|^4\} - |\mathrm{E}\{(y^+)^2\}|^2}{\mathrm{E}^2\{|y^+|^2\}} - 2 = \frac{P(\mu)}{Q^2(\mu)} - 2 \tag{9.31}$$

where $y^+ = y + \mu d$, $y = \mathbf{w}^{\mathrm{H}}\mathbf{x}$, $d = \mathbf{d}^{\mathrm{H}}\mathbf{x}$, $P(\mu) = P_1(\mu) - |P_2(\mu)|^2$, $P_1(\mu) = \mathrm{E}\{|y^+|^4\}$, $P_2(\mu) = \mathrm{E}\{(y^+)^2\}$, and $Q(\mu) = \mathrm{E}\{|y^+|^2\}$. For convenience in the following development, we denote

$$a = y^2, \qquad b = d^2, \qquad c = yd, \qquad e = \mathbb{R}\mathrm{e}\big(yd^*\big). \tag{9.32}$$

After some manipulations, the above polynomials can be expressed as:

$$P(\mu) = \sum_{k=0}^{4} h_k \mu^k, \qquad Q(\mu) = \sum_{k=0}^{2} i_k \mu^k \tag{9.33}$$

where

$$\begin{aligned}
&h_0 = \mathrm{E}\big\{|a|^2\big\} - \big|\mathrm{E}\{a\}\big|^2, \qquad h_1 = 4\mathrm{E}\big\{|a|e\big\} - 4\mathbb{R}\mathrm{e}\big(\mathrm{E}\{a\}\mathrm{E}\big\{c^*\big\}\big),\\
&h_2 = 4\mathrm{E}\big\{e^2\big\} + 2\mathrm{E}\big\{|a||b|\big\} - 4\big|\mathrm{E}\{c\}\big|^2 - 2\mathbb{R}\mathrm{e}\big(\mathrm{E}\{a\}\mathrm{E}\big\{b^*\big\}\big),\\
&h_3 = 4\mathrm{E}\big\{|b|e\big\} - 4\mathbb{R}\mathrm{e}\big(\mathrm{E}\{b\}\mathrm{E}\big\{c^*\big\}\big), \qquad h_4 = \mathrm{E}\big\{|b|^2\big\} - \big|\mathrm{E}\{b\}\big|^2,
\end{aligned} \tag{9.34}$$

$$i_0 = \mathrm{E}\{|a|\}, \qquad i_1 = 2\mathrm{E}\{e\}, \qquad i_2 = \mathrm{E}\{|b|\}. \tag{9.35}$$

Hence, the derivative of $\mathscr{J}(\mathbf{w}+\mu\mathbf{g})$ with respect to μ is given by

$$\mathscr{J}'(\mu) = \frac{P'(\mu)Q(\mu) - 2P(\mu)Q'(\mu)}{Q^3(\mu)} = \frac{p(\mu)}{Q^3(\mu)}. \tag{9.36}$$

Combining Eqs. (9.33)–(9.36), $p(\mu)$ is given by the fourth-degree polynomial (quartic)

$$p(\mu) = \sum_{n=0}^{4} a_n \mu^n \tag{9.37}$$

with

$$\begin{aligned} a_0 &= -2h_0 i_1 + h_1 i_0, & a_1 &= -4h_0 i_2 - h_1 i_1 + 2h_2 i_0, \\ a_2 &= -3h_1 i_2 + 3h_3 i_0, & a_3 &= -2h_2 i_2 + h_3 i_1 + 4h_4 i_0, \\ a_4 &= -h_3 i_2 + 2h_4 i_1. \end{aligned}$$

The real parts of the roots of this polynomial are the step-size candidates. To determine the optimal step size μ_{opt}, the roots are plugged back into Eqs. (9.31)–(9.33) to check which candidate maximizes $\mathscr{J}_\kappa(\mathbf{w}+\mu\mathbf{d}) = |\mathscr{J}(\mathbf{w}+\mu\mathbf{d})|$; if aiming at a source with kurtosis sign ε, one should check $\mathscr{J}_\varepsilon(\mathbf{w}+\mu\mathbf{d}) = \varepsilon\,\mathscr{J}(\mathbf{w}+\mu\mathbf{d})$ instead. The extracting vector is then updated as $\mathbf{w}^+ = \mathbf{w} + \mu_{\mathrm{opt}}\mathbf{d}$. Since the kurtosis is scale invariant, the extracting vector can be normalized after updating, as in Eq. (9.24). It should be remarked that this normalization is not forced by prewhitening—an optional step when using kurtosis, as we saw before—but just performed by numerical convenience (Sect. 9.4.2). As a suitable search direction, one can use the gradient of the full version of kurtosis, given by Eq. (9.14), which can be normalized for increased numerical stability. This optimal step-size algorithm for iterative kurtosis maximization is referred to as *RobustICA* [71, 74, 76]. In the context of blind single-channel equalization, the method had been suggested without details a few years earlier under the name of *optimal step-size kurtosis maximization algorithm* (*OS-KMA*) [69]. A very similar optimization idea holds for other source separation and equalization principles, both in blind and semi-blind operating modes, such as the constant power [68] and the constant modulus criterion [69, 70, 72]

RobustICA algorithm for kurtosis optimization with optimal step size

- Set an initial value $\mathbf{w}^{(0)}$ for the extracting vector.
- For $k = 1, 2, \ldots, k_{\max}$, do:

1. Compute the gradient direction $\mathbf{d}^{(k-1)} = \nabla \mathscr{J}_\kappa(\mathbf{w}^{(k-1)})$ from Eq. (9.14).
2. Obtain the optimal step size μ_{opt} as described above (Eqs. (9.31)–(9.37)).
3. Update: $\widetilde{\mathbf{w}} = \mathbf{w}^{(k-1)} + \mu_{\text{opt}}\mathbf{d}^{(k-1)}$.
4. Normalize: $\mathbf{w}^{(k)} = \widetilde{\mathbf{w}}/\|\widetilde{\mathbf{w}}\|$.

To quickly illustrate the benefits of RobustICA, we take up the simulation example introduced at the end of Sect. 9.4.4.2. Recall that the solid lines in Fig. 9.3 plot the kurtosis contrast as a function of the global filter angle $\Delta\theta$ for the observed instantaneous orthogonal mixture of two sources. Despite the short observation window (just 50 samples), the kurtosis local maxima lie quite close to the valid extraction solutions, thus yielding improved source estimates. Moreover, the algorithm converges in just a single iteration whatever the initialization employed (panels (a)–(b)).

Finally, let us point out that the optimal step-size cencept can also be used in the context of monotonically convergent algorithms presented in the next section. The basic idea is similar to RobustICA's, and the reader is referred to [13] for details.

9.4.6 Algorithms Based on Reference Signals

As justified in Sect. 9.1.2, many contrast functions are based on higher-order cumulants. A common example studied throughout this chapter is the kurtosis (9.8) giving rise to contrasts (9.9)–(9.10), which is a normalized *marginal* cumulant of the extractor output. In this section, we introduce contrast functions based on *cross-cumulants*. The advantage of these alternative contrasts is that they can be expressed as quadratic functions (Sect. 9.4.6.1) and, as such, their optimization is highly facilitated. Indeed, we will see that they can be considered as a starting point for developing monotonically convergent algorithms (Sect. 9.4.6.2).

9.4.6.1 Quadratic Contrast Functions

Crucial to the development of quadratic contrasts is the assumption that a *reference* signal, denoted by $z(n)$, is available. The reference signal is defined as the output of a MISO reference filter $\mathbf{v}(n)$:

$$z(n) = \mathbf{v}(n) \star \mathbf{x}(n) = \mathbf{t}(n) \star \mathbf{s}(n)$$

with $\mathbf{t}(n) = \mathbf{v}(n) \star \mathbf{M}(n)$. At first sight, one could think of $z(n)$ as a kind of prior information about the source being targeted by the extraction procedure. In this

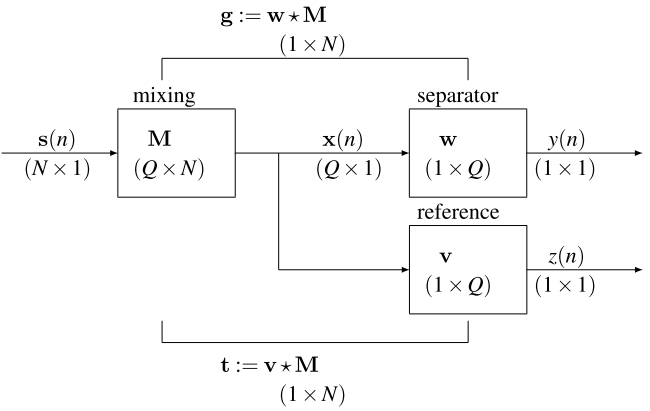


Fig. 9.4 Schematic diagram of the system setup and notations used by algorithms based on reference signals

respect, the resulting method can be considered as *semi-blind* [12]. The above notations are graphically summarized in Fig. 9.4.

Now, given a reference signal $z(n)$ we can define the following criterion:

$$\mathscr{J}_r(\mathbf{w}, \mathbf{v}) \triangleq \frac{|\mathscr{C}_z\{y\}|}{\mathrm{E}\{|y(n)|^2\}\mathrm{E}\{|z(n)|^2\}} \tag{9.38}$$

where

$$\mathscr{C}_z\{y\} \triangleq \mathrm{Cum}\big\{y(n), y(n)^*, z(n), z(n)^*\big\}$$

is a fourth-order cross-cumulant defined for any jointly stationary signals $z(n)$ and $y(n)$. The first interesting property of criterion (9.38) is that it is a contrast function for almost any fixed reference signal $z(n)$. Precise conditions for the validity of such a contrast are derived in [10, 12, 16]. Essentially, the reference $z(n)$ should be 'close' enough to one particular source signal so as not to contain identical power contributions from two (or more) sources. This assumption is generally satisfied in practice even if the reference filter is chosen randomly. More interestingly, function (9.38) can be expressed as

$$\mathscr{J}_r(\mathbf{w}, \mathbf{v}) = \frac{|\mathbf{w}^{\mathrm{H}}\mathbf{C}_{\mathbf{v}}\mathbf{w}|}{(\mathbf{w}^{\mathrm{H}}\mathbf{R}\mathbf{w})(\mathbf{v}^{\mathrm{H}}\mathbf{R}\mathbf{v})}. \tag{9.39}$$

In the above equation, $\mathbf{R}$ and $\mathbf{C}_{\mathbf{v}}$ are a covariance and a cumulant matrix which, in the case of an instantaneous mixture, are given by:

$$\mathbf{R} = \mathrm{E}\big\{\mathbf{x}(n)\mathbf{x}(n)^{\mathrm{H}}\big\}, \qquad [\mathbf{C}_{\mathbf{v}}]_{ij} = \mathrm{Cum}\big\{x_i(n), x_j(n)^*, z(n), z(n)^*\big\}.$$

Replacing $\mathbf{x}(n)$ by a vector stacking the consecutive delayed values of the observation as in Sect. 9.2.2, a similar definition holds in the convolutive case (see [13, 16] for details).

For fixed $\mathbf{v}$, the term $\mathbf{v}^{\mathrm{H}}\mathbf{R}\mathbf{v}$ is just an irrelevant constant factor, and Eq. (9.39) becomes essentially a Rayleigh quotient in the extracting vector $\mathbf{w}$. The maximization of this quotient is a well-known problem in array signal processing and matrix algebra that can be solved, e.g., via the generalized EVD of matrix pencil $(\mathbf{C}_{\mathbf{v}}, \mathbf{R})$ and accepts an SVD-based solution [16]. Despite these interesting features, it has been observed that using this contrast function within a deflation procedure is not robust, since the rank of $\mathbf{R}$ decreases when performing deflation, as noted in Sect. 9.4.3. As a consequence of the unknown rank of $\mathbf{R}$, the performance of the SVD-based optimization seriously degrades as more sources are recovered. The following section details an alternative method avoiding this drawback.

9.4.6.2 Monotonically Convergent Algorithms Based on Quadratic Contrasts

As mentioned above, the SVD-based optimization of the quadratic contrast (9.38)–(9.39) is not robust and not recommended in the case where $\mathbf{R}$ is of unknown and non maximal rank, which always occurs in a deflation scenario. As a first alternative, maximizing $\mathscr{J}_r(\mathbf{w}, \mathbf{v})$ by a gradient algorithm has been proposed in [11] at the cost of an increased computational burden. In this section, we show that an intermediate approach is possible.

One can note that criteria (9.9) and (9.38) are linked by $\mathscr{J}_\kappa(\mathbf{w}) = \mathscr{J}_r(\mathbf{w}, \mathbf{w})$. Based on this fact and on the symmetry property $\mathscr{J}_r(\mathbf{w}, \mathbf{v}) = \mathscr{J}_r(\mathbf{v}, \mathbf{w})$, an alternative algorithm has recently been proposed in [13] for the optimization of $\mathscr{J}_\kappa(\mathbf{w})$. The idea is to perform the iterative maximization of $\mathscr{J}_r$ with respect to the extracting filter after initializing this latter with a given reference filter. The reference filter is then updated with the extracting filter obtained after maximization, and so forth. In the summary given below, the gradient operator with respect to the first argument of $\mathscr{J}_r$ is denoted by $\nabla_1 \mathscr{J}_r$.

Algorithm for kurtosis maximization based on reference signals

- Initialize the reference filter $\mathbf{v}_0(n)$ and compute the corresponding reference signal $z_0(n) = \mathbf{v}_0(n) \star \mathbf{x}(n)$.
- For $k = 0, 1, \ldots, (k_{\max} - 1)$, initialize the extracting filter as $\mathbf{w}_0 = \mathbf{v}_k$ and do:
  - For $\ell = 0, 1, 2, \ldots, (\ell_{\max} - 1)$, do exact line search along $\mathbf{w}$-dimension:
    1. Compute gradient direction $\mathbf{d}_\ell = \nabla_1 \mathscr{J}_r(\mathbf{w}_\ell, \mathbf{v}_k)$
    2. Compute the optimal step size $\mu_{\mathrm{opt}} = \arg\max_\mu \mathscr{J}_r(\mathbf{w}_\ell + \mu \mathbf{d}_\ell, \mathbf{v}_k)$.
    3. Update: $\widetilde{\mathbf{w}} = \mathbf{w}_\ell + \mu_{\mathrm{opt}} \mathbf{d}_\ell$.
    4. Normalize: $\mathbf{w}_{\ell+1} = \frac{\widetilde{\mathbf{w}}}{(\mathrm{E}\{|\widetilde{\mathbf{w}}(n) \star \mathbf{x}(n)|^2\})^{1/2}}$.
  - Update the reference filter: $\mathbf{v}_{k+1} = \mathbf{w}_{\ell_{\max}}$.

The convergence of a simpler version of the above algorithm has been proved in [13] when the sources have identical cumulant sign. More precisely, with $k_{\max}$ infinite and for any initialization point, the algorithm converges to a stationary point of the criterion $\mathscr{J}_\kappa$, which in practice can only be a maximum, hence corresponding to a separating filter.

Let us add a few remarks about the above algorithm, which are all detailed in [13]:

- The method can be considered as a hybrid approach between kurtosis and reference-based contrast function maximization.
- Depending on $k_{\max}$ and $\ell_{\max}$, a compromise can be made between computational time and performance. Generally speaking, an appropriate choice can significantly reduce the computational load compared to a gradient optimization of contrast $\mathscr{J}_\kappa$.
- A link can be established with the Expectation Maximization (EM) method and generalizations of it referred to as Minimization–Maximization (MM) algorithms. In particular, it can be shown that the above algorithm maximizes at each step a lower-bound of the kurtosis.
- Similar to RobustICA (Sect. 9.4.5.2), the optimal step size in the exact line search step of the algorithm can be obtained algebraically by finding the roots of a polynomial of degree two.

9.5 Illustrative Results

This section presents some illustrative experimental results of the kurtosis-based techniques reviewed in this chapter. For the sake of conciseness, our attention is restricted to the areas of digital communications (Sect. 9.5.1) and biomedical signal processing (Sect. 9.5.2).

9.5.1 Source Separation and Equalization in Digital Communications

As mentioned in Sect. 9.1.1, digital communications is one of the application domains where blind source separation and equalization techniques have proven most useful. In this area, the source signals and other system parameters can be specifically designed according to the features of the propagation channel and separation criteria being employed. The MIMO signal model of Sect. 9.2 is indeed a key ingredient of recent developments such as space-time coding for increasing the performance of wireless communication systems [32]. Also, most digital modulations verify non-Gaussianity assumption A1, and thus naturally lend themselves to the kurtosis-based blind source extraction algorithms considered in the chapter.

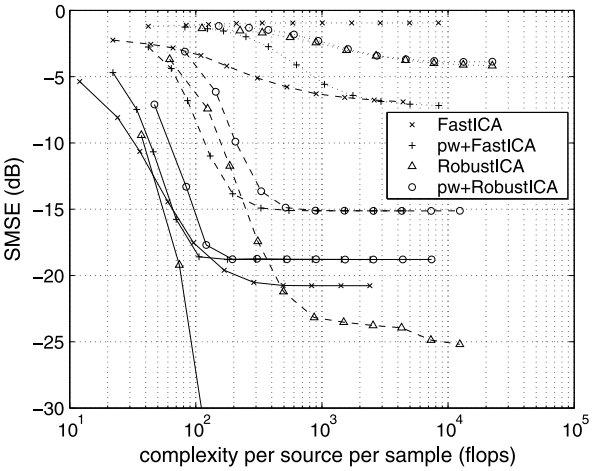


Fig. 9.5 Kurtosis-based blind source separation in a simulated MIMO wireless digital communication system transmitting BPSK sources. Source extraction quality against computational cost (measured in floating point operations, flops) for different mixture sizes N. Signal blocks composed of $T = 150$ samples. *Solid lines*: $N = 5$. *Dashed lines*: $N = 10$. *Dotted lines*: $N = 20$. In the legend, 'pw' denotes prewhitening. Reproduced from [74] with permission

A flat-fading (frequency-nonselective) MIMO wireless channel accepts the instantaneous linear mixture model of Sect. 9.2.2. A simulation scenario composed of N independent BPSK-modulated sources transmitted through one such channel will serve to illustrate the improved convergence and short-sample robustness of kurtosis compared with related fourth-order contrasts. Figure 9.5 plots the average reconstructed signal mean square error (SMSE) in random orthogonal mixtures for the RobustICA algorithm directly optimizing the kurtosis contrast (Sect. 9.4.5.2) and the FastICA algorithm based on the simplified fourth-order moment (Sect. 9.4.4.1). For the kind of sources considered in this experiment, FastICA's Hessian simplification in Eq. (9.22) is not an approximation but actually holds true. Yet the benefits—enabled by the use of kurtosis—of avoiding prewhitening can be remarked, as RobustICA without this second-order processing step presents the best quality–cost trade-off, overcoming the performance flooring shown by the other implementations except for a high number of sources. Extensive experimental results comparing RobustICA with other kurtosis-maximization methods in both real- and complex-valued scenarios are reported in [71, 74, 76]. Optimal step-size algorithms for iterative optimization of other source separation and equalization principles are evaluated in [68–70, 72]. A comparison between algorithms based on reference signals (Sect. 9.4.6) and the gradient optimization of kurtosis can be found in [13, 16].

9.5.2 Artifact Rejection in Biomedical Recordings

Another application domain where the kurtosis and related fourth-order criteria have proven their interest is artifact rejection in signals of biomedical origin. The biomedical domain poses hard separation problems, as the sources of physiological activity are often difficult to model, may show strong inter- and intra-patient variability and,

unlike digital communications, cannot be designed to suit the specific properties of the propagation environment and source separation criteria being considered.

In cardiac signal processing, the classical problem of fetal electrocardiogram (ECG) extraction from maternal abdominal recordings was first approached from the perspective of BSS based on higher-order statistics in [24, 25]. In [75], this approach was later shown to outperform traditional techniques for array processing such as Widrow's multireference adaptive noise canceling [65].

More recently, kurtosis-based methods have been applied to the analysis of Atrial Fibrillation (AF), the most common sustained cardiac arrhythmia encountered in clinical practice, yet still not fully understood by cardiologists. This disease is manifested by a disorganized propagation of electrical activity across the atria, the heart's upper chambers. As a result, atrial activity becomes uncoupled from the ventricular beats and results in an inefficient atrial contraction; in turn, impaired mechanical function increases the risk of blood-clot formation and stroke. In the ECG, the electrical activity originating in the atria during AF manifests as the absence of the P wave, which is replaced by rapid oscillations, called f-waves. This can be observed in the top plot of Fig. 9.6, which displays the ECG signal recorded in chest lead V1 from an AF patient.[1] Ventricular activity corresponds to the large amplitude peaks (the so-called QRST complex) occurring quasi-periodically, whereas atrial activity is visible between consecutive ventricular beats. The frequency spectrum of atrial fibrillatory activity, typically narrowband with a dominant peak between 3 and 9 Hz, is also masked by the broader ventricular spectrum, as observed in the top panel of Fig. 9.7.

To accurately analyze AF from noninvasive recordings, one first needs to cancel out the ventricular artifact that interferes with the continuous atrial activity signal. Atrial source extraction techniques based on higher-order statistics were first proposed for this purpose in [54, 55] based on the statistical independence between atrial and ventricular signal components. Unfortunately, the signal of interest tends to become Gaussian as the disease evolves into chronic forms, rendering these techniques ineffective because assumption `A1` no longer holds. To surmount this difficulty, the time coherence or narrowband spectrum of the atrial signal can be exploited through a refinement based on the second-order blind identification (SOBI) technique of [3], giving rise to the FastICA-SOBI method of [18]. Interestingly, kurtosis maximization in the frequency domain [73, 74] allows the simultaneous exploitation of both the statistical independence between atrial and ventricular activities and the atrial signal's time coherence in a single processing stage, while achieving improved extraction performance. This is illustrated in the middle and bottom plots of Figs. 9.6–9.7, which show the atrial activity reconstructed in lead V1 by processing the AF patient's 12-lead ECG. RobustICA applied in the frequency domain (*RobustICA-f*) achieves a neater atrial signal estimate than the hybrid FastICA-SOBI technique of [18], as perceived by visual inspection especially

[1] Recording kindly provided by the Hemodynamics Department, Clinical University Hospital, University of Valencia, Spain, and ITACA-Bioingenieria, Polytechnic University of Valencia, Spain.

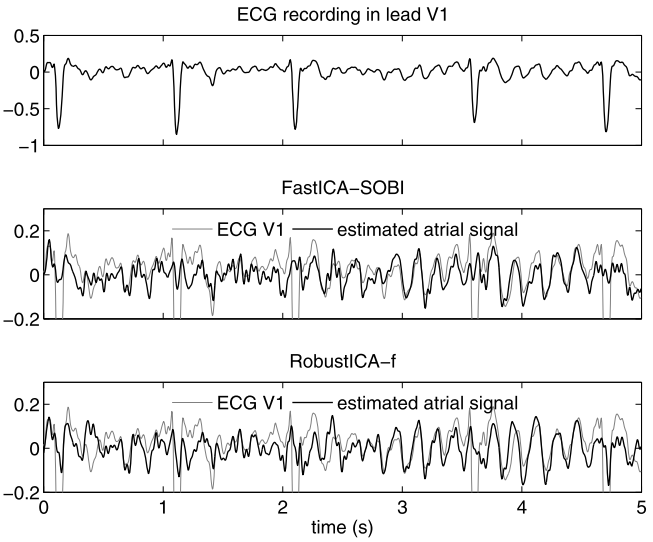


Fig. 9.6 Atrial activity extraction in atrial fibrillation ECGs based on kurtosis optimization. *Top*: a 5-second segment of ECG lead V1 from an atrial fibrillation patient. *Middle*: atrial signal contribution to lead V1 estimated by the combined FastICA-SOBI technique [18] from the 12-lead ECG. *Bottom*: atrial signal contribution to lead V1 estimated by RobustICA-f from the 12-lead ECG. For reference, *gray lines* in the background show the original signal recorded in lead V1. Amplitudes are in mV. The corresponding frequency spectra are shown in Fig. 9.7

in the frequency domain (Fig. 9.7). Visual results are validated by quantitative performance measures such as spectral concentration, an atrial signal quality index proposed in [18]. Results on a whole AF ECG database confirm RobustICA-f's improved atrial signal estimation performance [73, 74]. Source extraction techniques for artifact suppression in ECG recordings are discussed at length in [67].

9.6 Conclusions

Over the last two decades, kurtosis has become one of the most popular contrasts for blind source separation and equalization in linear channels. In combination with MISO filtering structures for single-source extraction and suitable deflation schemes, local optimizers of kurtosis lead to satisfactory source estimation in ideal model conditions. Despite the lack of closed-form solutions, its mathematical tractability and computational convenience have spurred the development of a rich variety of cost-efficient iterative methods for optimizing this contrast, mostly based on gradient and Newton updates. Some of the most representative of these algorithms have been reviewed in this chapter. A selection of ready-to-use MATLAB™ implementations can be found in the "Online Material" section below.

Space limitations have precluded the treatment of important issues such as filter-order selection, more elaborate deflation strategies, in-depth analysis of finite sam-

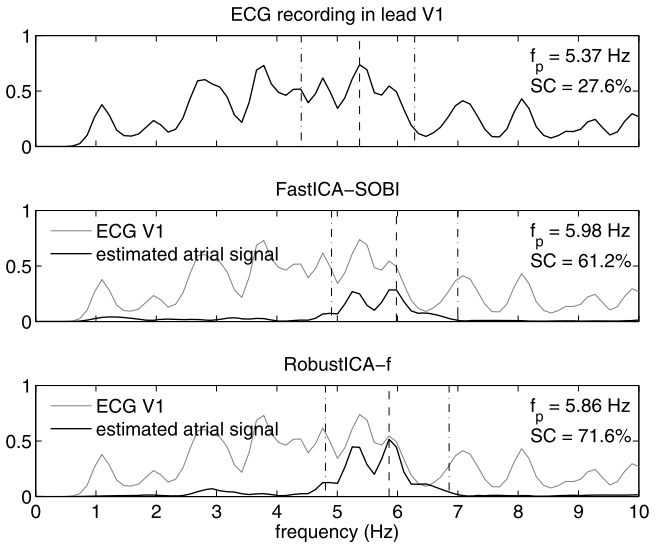


Fig. 9.7 Atrial activity extraction in atrial fibrillation ECGs based on kurtosis optimization. Power spectral densities of the time courses shown in Fig. 9.6. *Top*: power spectral density of ECG lead V1 from an atrial fibrillation patient. *Middle*: power spectral density of atrial signal contribution to lead V1 estimated by the combined FastICA-SOBI technique [18] from the 12-lead ECG. *Bottom*: power spectral density of atrial signal contribution to lead V1 estimated by RobustICA-f from the 12-lead ECG. For reference, the frequency spectrum of lead V1 is plotted in *gray lines*. f_p denotes the estimated dominant peak frequency. SC denotes spectral concentration, an objective performance index quantifying the atrial signal relative power around f_p. *Dashed lines*: dominant frequency location. *Dash-dotted lines*: bounds used in the computation of spectral concentration. In the y-axes, only relative amplitudes are important

ple effects, or the separation of mixtures composed of fewer sensors than sources (underdetermined case), a scenario of great practical impact. These open questions are currently very active research topics in signal processing.

Online Material

Most methods described in this chapter are available online at `http://www-public.it-sudparis.eu/~castella/toolbox/`.

The RobustICA technique described in Sect. 9.4.5.2 can be found at `http://www.i3s.unice.fr/~zarzoso/robustica.html`.

References

1. Amari, S.: Natural gradient works efficiently in learning. Neural Comput. **10**(2), 251–276 (1998)

2. Amari, S., Cichocki, A.: Adaptive blind signal processing—neural network approaches. Proc. IEEE **86**(10), 2026–2048 (1998)
3. Belouchrani, A., Abed-Meraim, K., Cardoso, J.F., Moulines, E.: A blind source separation technique using second-order statistics. IEEE Trans. Signal Process. **45**(2), 434–444 (1997)
4. Benveniste, A., Métivier, M., Priouret, P.: Algorithmes Adaptatifs et Approximations Stochastiques. Masson, Paris (1987). English translation by S.S. Wilson, Adaptive Algorithms and Stochastic Approximations. Springer, Berlin (1990)
5. Bermejo, S.: Finite sample effects in higher order statistics contrast functions for sequential blind source separation. IEEE Signal Process. Lett. **12**(6), 481–484 (2005)
6. Bermejo, S.: Finite sample effects of the fast ICA algorithm. Neurocomputing **71**(1–3), 392–399 (2007)
7. Bingham, E., Hyvärinen, A.: A fast fixed-point algorithm for independent component analysis of complex valued signals. Int. J. Neural Syst. **10**(1), 1–8 (2000)
8. Cardoso, J.F.: On the performance of orthogonal source separation algorithms. In: Proc. EUSIPCO-94, VII European Signal Processing Conference, Edinburgh, UK, pp. 776–779 (1994)
9. Cardoso, J.F., Laheld, B.H.: Equivariant adaptive source separation. IEEE Trans. Signal Process. **44**(12), 3017–3030 (1996)
10. Castella, M., Moreau, E.: Generalized identifiability conditions for blind convolutive MIMO separation. IEEE Trans. Signal Process. **57**(7), 2846–2852 (2009)
11. Castella, M., Moreau, E.: A new optimization method for reference-based quadratic contrast functions in a deflation scenario. In: Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Taipei, Taiwan, R.O.C, pp. 3161–3164 (2009)
12. Castella, M., Moreau, E.: Reference based contrast functions in a semi-blind context. In: Proc. of ICA'09, 8th International Conference on Independent Component Analysis and Signal Separation, Paraty-RJ, Brazil. LNCS, vol. 5441, pp. 9–16 (2009)
13. Castella, M., Moreau, E.: New kurtosis optimization schemes for MISO equalization. IEEE Trans. Signal Process. **60**(3), 1319–1330 (2012)
14. Castella, M., Bianchi, P., Chevreuil, A., Pesquet, J.C.: A blind source separation framework for detecting CPM sources mixed by a convolutive MIMO filter. Signal Process. **86**(8), 1950–1967 (2006)
15. Castella, M., Chevreuil, A., Pesquet, J.C.: Mélanges convolutifs. In: Comon, P., Jutten, C. (eds.) Séparation de Sources, Tome 1: Concepts de Base et Analyse en Composantes Indépendantes, pp. 231–272. Hermès, Paris (2007). Chap. 7
16. Castella, M., Rhioui, S., Moreau, E., Pesquet, J.C.: Quadratic higher-order criteria for iterative blind separation of a MIMO convolutive mixture of sources. IEEE Trans. Signal Process. **55**(1), 218–232 (2007)
17. Castella, M., Chevreuil, A., Pesquet, J.C.: Convolutive mixtures. In: Comon, P., Jutten, C. (eds.) Handbook of Blind Source Separation, Independent Component Analysis and Applications, pp. 281–324. Academic Press, New York (2010). Chap. 8
18. Castells, F., Rieta, J.J., Millet, J., Zarzoso, V.: Spatiotemporal blind source separation approach to atrial activity estimation in atrial tachyarrhythmias. IEEE Trans. Biomed. Eng. **52**(2), 258–267 (2005)
19. Comon, P.: Independent component analysis, a new concept? Signal Process. **36**(3), 287–314 (1994). Special Issue on Higher-Order Statistics
20. Comon, P.: Independent component analysis, contrasts, and convolutive mixtures. In: Proc. 2nd IMA International Conference on Mathematics in Communications, Lancaster, UK, pp. 10–17 (2002)
21. Comon, P.: Contrasts, independent component analysis, and blind deconvolution. Int. J. Adapt. Control Signal Process. **18**(3), 225–243 (2004). Special Issue on Blind Signal Separation
22. Comon, P., Jutten, C. (eds.): Handbook of Blind Source Separation, Independent Component Analysis and Applications. Academic Press, Oxford (2010)

23. Comon, P., Moreau, E.: Improved contrast dedicated to blind separation in communications. In: Proc. ICASSP-97, 22nd IEEE International Conference on Acoustics, Speech and Signal Processing, Munich, Germany, pp. 3453–3456 (1997)
24. De Lathauwer, L., Callaerts, D., De Moor, B., Vandewalle, J.: Fetal electrocardiogram extraction by source subspace separation. In: Proc. IEEE/ATHOS Signal Processing Conference on Higher-Order Statistics, Girona, Spain, pp. 134–138 (1995)
25. De Lathauwer, L., De Moor, B., Vandewalle, J.: Fetal electrocardiogram extraction by blind source subspace separation. IEEE Trans. Biomed. Eng. **47**(5), 567–572 (2000). Special Topic Section on Advances in Statistical Signal Processing for Biomedicine
26. Delfosse, N., Loubaton, P.: Adaptive blind separation of independent sources: a deflation approach. Signal Process. **45**(1), 59–83 (1995)
27. Ding, Z., Nguyen, T.: Stationary points of a kurtosis maximization algorithm for blind signal separation and antenna beamforming. IEEE Trans. Signal Process. **48**(6), 1587–1596 (2000)
28. Donoho, D.: On minimum entropy deconvolution. In: Proc. 2nd Applied Time Series Analysis Symposium, Tulsa, OK, pp. 565–608 (1980)
29. Douglas, S.C.: On the convergence behavior of the FastICA algorithm. In: Proc. ICA-2003, 4th International Symposium on Independent Component Analysis and Blind Signal Separation, Nara, Japan, pp. 409–414 (2003)
30. Douglas, S.C.: Fixed-point algorithms for the blind separation of arbitrary complex-valued non-Gaussian signal mixtures. EURASIP J. Adv. Signal Process. (2007). doi: 10.1155/2007/36525
31. Dubroca, R., De Luigi, C., Castella, M., Moreau, E.: A general algebraic algorithm for blind extraction of one source in a MIMO convolutive mixture. IEEE Trans. Signal Process. **58**(5), 2484–2493 (2010)
32. Gesbert, D., Shafi, M., Shan-Shiu, D., Smith, P.J., Naguib, A.: From theory to practice: an overview of MIMO space-time coded wireless systems. IEEE J. Sel. Areas Commun. **21**(3), 281–302 (2003)
33. Godard, D.N.: Self-recovering equalization and carrier tracking in two-dimensional data communication systems. IEEE Trans. Commun. **28**(11), 1867–1875 (1980)
34. Hyvärinen, A.: One-unit contrast functions for independent component analysis: a statistical analysis. In: Proc. IEEE Neural Networks for Signal Processing Workshop, Amelia Island, FL, pp. 388–397 (1997)
35. Hyvärinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999)
36. Hyvärinen, A., Oja, E.: A fast fixed-point algorithm for independent component analysis. Neural Comput. **9**(7), 1483–1492 (1997)
37. Johnson, C.R., Schniter, P., Fijalkow, I., Tong, L., Behm, J.D., et al.: The core of FSE-CMA behavior theory. In: Haykin, S.S. (ed.) Unsupervised Adaptive Filtering, Vol. II: Blind Deconvolution, pp. 13–112. Wiley, New York (2000). Chap. 2
38. Li, H., Adali, T.: A class of complex ICA algorithms based on the kurtosis cost function. IEEE Trans. Neural Netw. **19**(3), 408–420 (2008)
39. Loubaton, P., Regalia, P.: Blind deconvolution of multivariate signals: a deflation approach. In: Proceedings of ICC, Geneva, Switzerland, pp. 1160–1164 (1993)
40. Moreau, E.: Criteria for complex sources separation. In: Proc. EUSIPCO-96, VII European Signal Processing Conference, Trieste, Italy, vol. II, pp. 931–934 (1996)
41. Moreau, E., Comon, P.: Contrasts. In: Comon, P., Jutten, C. (eds.) Handbook of Blind Source Separation, Independent Component Analysis and Applications, pp. 65–105. Academic Press, Oxford (2010). Chap. 3
42. Moreau, E., Macchi, O.: Complex self-adaptive algorithms for source separation based on high order contrasts. In: Proc. EUSIPCO-94, VII European Signal Processing Conference, Edinburgh, UK, pp. 1157–1160 (1994)
43. Moreau, E., Macchi, O.: A one stage self-adaptive algorithm for source separation. In: Proc. ICASSP-94, 19th IEEE International Conference on Acoustics, Speech and Signal Processing, Adelaide, Australia, vol. 3, pp. 49–52 (1994)

44. Moreau, E., Macchi, O.: High order contrasts for self-adaptive source separation. Int. J. Adapt. Control Signal Process. **10**(1), 19–46 (1996)
45. Moreau, E., Pesquet, J.C.: Generalized contrasts for multichannel blind deconvolution of linear systems. IEEE Signal Process. Lett. **4**(6), 182–183 (1997)
46. Moreau, E., Thirion-Moreau, N.: Non symmetrical contrasts for sources separation. IEEE Trans. Signal Process. **47**(8), 2241–2252 (1999)
47. Moreau, E., Pesquet, J.C., Thirion-Moreau, N.: Convolutive blind signal separation based on asymmetrical contrast functions. IEEE Trans. Signal Process. **55**(1), 356–371 (2007)
48. Novey, M., Adali, T.: Complex ICA by negentropy maximization. IEEE Trans. Neural Netw. **19**(4), 596–609 (2008)
49. Novey, M., Adali, T.: On extending the complex FastICA algorithm to noncircular sources. IEEE Trans. Signal Process. **56**(5), 2148–2154 (2008)
50. Papadias, C.B.: Globally convergent blind source separation based on a multiuser kurtosis maximization criterion. IEEE Trans. Signal Process. **48**(12), 3508–3519 (2000)
51. Pesquet, J.C., Moreau, E.: Cumulant based independence measures for linear mixtures. IEEE Trans. Inf. Theory **47**(5), 1947–1956 (2001)
52. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C. The Art of Scientific Computing, 2nd edn. Cambridge University Press, Cambridge (1992)
53. Regalia, P.A.: A finite-interval constant modulus algorithm. In: Proc. ICASSP-2002, 27th International Conference on Acoustics, Speech and Signal Processing, Vol. III, Orlando, FL, pp. 2285–2288 (2002)
54. Rieta, J.J., Zarzoso, V., Millet-Roig, J., García-Civera, R., Ruiz-Granell, R.: Atrial activity extraction based on blind source separation as an alternative to QRST cancellation for atrial fibrillation analysis. In: Proc. Computers in Cardiology, Boston, MA, vol. 27, pp. 69–72 (2000)
55. Rieta, J.J., Castells, F., Sánchez, C., Zarzoso, V., Millet, J.: Atrial activity extraction for atrial fibrillation analysis using blind source separation. IEEE Trans. Biomed. Eng. **51**(7), 1176–1186 (2004)
56. Ristaniemi, T., Joutsensalo, J.: Advanced ICA-based receivers for block fading DS-CDMA channels. Signal Process. **82**(3), 417–431 (2002)
57. Sato, Y.: A method of self-recovering equalization for multi-level amplitude modulation. IEEE Trans. Commun. **23**, 679–682 (1975)
58. Shalvi, O., Weinstein, E.: New criteria for blind deconvolution of nonminimum phase systems (channels). IEEE Trans. Inf. Theory **36**(2), 312–321 (1990)
59. Simon, C., Loubaton, P., Jutten, C.: Separation of a class of convolutive mixtures: a contrast function approach. Signal Process. **4**(81), 883–887 (2001)
60. Stuart, A., Ord, K.: Kendall's Advanced Theory of Statistics, vol. 1, 6th edn. Hodder Arnold, Sevenoaks (1994)
61. Thirion-Moreau, N., Moreau, E.: Generalized criteria for blind multivariate signal equalization. IEEE Signal Process. Lett. **9**(2), 72–74 (2002)
62. Tichavský, P., Koldovský, Z., Oja, E.: Performance analysis of the FastICA algorithm and Cramér-Rao bounds for linear independent component analysis. IEEE Trans. Signal Process. **54**(4), 1189–1203 (2006)
63. Treichler, J.R., Agee, B.G.: A new approach to multipath correction of constant modulus signals. IEEE Trans. Acoust. Speech Signal Process. **31**(2), 459–472 (1983)
64. Tugnait, J.K.: Identification and deconvolution of multichannel linear non-Gaussian processes using higher order statistics and inverse filter criteria. IEEE Trans. Signal Process. **45**(3), 658–672 (1997)
65. Widrow, B., Glover, J.R., McCool, J.M., et al.: Adaptive noise cancelling: principles and applications. Proc. IEEE **63**(12), 1692–1716 (1975)
66. Wiggins, R.A.: Minimum entropy deconvolution. Geoexploration **16**, 21–35 (1978)
67. Zarzoso, V.: Extraction of ECG characteristics using source separation techniques: exploiting statistical independence and beyond. In: Naït-Ali, A. (ed.) Advanced Biosignal Processing, pp. 15–47. Springer, Berlin (2009). Chap. 2

68. Zarzoso, V., Comon, P.: Blind and semi-blind equalization based on the constant power criterion. IEEE Trans. Signal Process. **53**(11), 4363–4375 (2005)
69. Zarzoso, V., Comon, P.: Blind channel equalization with algebraic optimal step size. In: Proc. EUSIPCO-2005, XIII European Signal Processing Conference, Antalya, Turkey (2005)
70. Zarzoso, V., Comon, P.: Semi-blind constant modulus equalization with optimal step size. In: Proc. ICASSP-2005, 30th International Conference on Acoustics, Speech and Signal Processing, Vol. III, Philadelphia, PA, pp. 577–580 (2005)
71. Zarzoso, V., Comon, P.: Comparative speed analysis of FastICA. In: Proc. ICA-2007, 7th International Conference on Independent Component Analysis and Signal Separation, London, UK, pp. 293–300 (2007)
72. Zarzoso, V., Comon, P.: Optimal step-size constant modulus algorithm. IEEE Trans. Commun. **56**(1), 10–13 (2008)
73. Zarzoso, V., Comon, P.: Automated extraction of atrial fibrillation activity from the surface ECG using independent component analysis in the frequency domain. In: Proc. Medical Physics and Biomedical Engineering World Congress, pp. 395–398 (2009). Invited
74. Zarzoso, V., Comon, P.: Robust independent component analysis by iterative maximization of the kurtosis contrast with algebraic optimal step size. IEEE Trans. Neural Netw. **21**(2), 248–261 (2010)
75. Zarzoso, V., Nandi, A.K.: Noninvasive fetal electrocardiogram extraction: blind separation versus adaptive noise cancellation. IEEE Trans. Biomed. Eng. **48**(1), 12–18 (2001)
76. Zarzoso, V., Comon, P., Kallel, M.: How fast is FastICA? In: Proc. EUSIPCO-2006, XIV European Signal Processing Conference, Florence, Italy (2006)
77. Zarzoso, V., Comon, P., Slock, D.: Semi-blind methods for communications. In: Comon, P., Jutten, C. (eds.) Handbook of Blind Source Separation, Independent Component Analysis and Applications, pp. 593–638. Academic Press, Oxford (2010). Chap. 15

Chapter 10
Swarm Intelligence Techniques Applied to Nonlinear Systems State Estimation

Hadi Nobahari, Alireza Sharifi, and Hamed MohammadKarimi

Abstract In this chapter, a new class of filters based on swarm intelligence is introduced for nonlinear systems state estimation. As a subset of heuristic filters, swarm filters formulate a nonlinear system state estimation problem as a stochastic dynamic optimization problem and utilize swarm intelligence techniques such as particle swarm optimization and ant colony optimization to find and track the best estimate. As a subset of nonlinear filters, swarm filters can successfully compete with well-known nonlinear filters such as unscented Kalman filter, etc.

10.1 Introduction

In many engineering applications, one needs to estimate the states of a dynamic system. A state estimation problem is defined as follows: given the mathematical model of a dynamic system, it is desired to estimate the time-varying states using a noisy measurement. Estimation problems are often categorized as prediction, filtering, and smoothing, depending on intended objectives and the available observations [1]. Here, the domain of focus is filtering, which is usually referred to as the extraction of true signal from the observations. Filters are usually minimizing a given objective function, while they are working. Such filters are called optimal filters [2].

Optimal filters are categorized to recursive and batch filters [1, 3]. A batch filter, e.g., least squares filter, uses the complete history of measurements to estimate unknown states. A recursive filter, in comparison, has the ability to receive and process the measurements sequentially. Recursive filters consist of two essentially stages: prediction and update [3]. Prediction uses the estimated states of the previous time

H. Nobahari (✉) · A. Sharifi · H. MohammadKarimi
Guidance and Control Research Center, Sharif University of Technology, P.O. Box: 11365-11155, Tehran, Iran
e-mail: nobahari@sharif.edu

A. Sharifi
e-mail: alirezasharifi@ae.sharif.ir

H. MohammadKarimi
e-mail: mohammadkarim@ae.sharif.ir

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_10,

step to produce an initial estimate of the current step. This stage is also known as the prior state estimation because it does not use the observations. In the update stage, the prior state estimate is combined with the current observation to refine the state estimate. This improved estimation is also termed as the posterior state estimation. The dynamic states can be estimated using the posterior Probability Distribution Function (PDF), obtained based on the received measurement. If either the system or measurement model is nonlinear, the posterior PDF will not be Gaussian, even if the measurement and the process noises are assumed to be Gaussian.

Several recursive filters can be found within the literature, the most well-known of which are the Kalman Filter (KF) [4], Extended Kalman Filter (EKF) [5], Unscented Kalman Filter (UKF) [6], Particle Filter (PF) [7], etc.

Recursive filters can also be categorized to linear and nonlinear filters [1, 3]. In a linear filter, such as KF, both system and measurement models are linear. KF assumes the posterior PDF to be Gaussian, which is characterized by a mean and a covariance. In the opposite, a nonlinear filter, such as EKF, UKF and PF, is used to estimate the states of a nonlinear dynamic system when either the system or the measurement model is nonlinear.

Analytical approximation and states sampling are two common approaches in nonlinear filtering. In the first approach, the nonlinear functions of the mathematical model are linearized and then a linear filter such as KF is utilized as well. EKF is an example of filters working based on an analytical approximation. Unlike to EKF, UKF is a sample based filter. It does not approximate the nonlinear mathematical model. Instead, it approximates the posterior PDF by a set of deterministically chosen samples. UKF is also referred to as a linear regression Kalman filter because it is based on statistical linearization rather than analytical ones [3].

The sample based filters can be categorized to mathematical and heuristic approaches [8]. UKF can be taken as a mathematical sample based filter since it uses a deterministic sampling process, the general estimation mathematics, and the mathematical operators such as unscented transform. In comparison, there are several sample based filters that utilize heuristic algorithms to sample the particles and to improve the position of them. These filters can be called heuristic filters [8]. PF is an example of heuristic filters. It works based on point mass (or particle) representation of the probability densities [9]. Unlike the UKF, PF represents the required posterior PDF by a set of random samples instead of deterministic ones. Also, it uses a resampling procedure to reduce the degeneracy of particle set. In another work, Genetic Algorithm (GA) has been combined with PF to increase the diversity of samples after resampling [10, 11]. Simulated Annealing (SA) has also been introduced into PF to improve its performance [12]. Moreover, a local search method has been inserted into PF to reduce the sample size and improve the efficiency [13].

The state estimation problem can be formulated as a stochastic dynamic optimization problem. Therefore, different ideas of heuristic optimization can be extended and modified to solve this problem. A new class of heuristic filters utilizes swarm intelligence techniques to solve the state estimation problem [8, 14–17]. The authors call these filters as swarm filters. These filters are introduced below.

This chapter is organized as follows: A state estimation problem is formulated in Sect. 10.2. PF is introduced in Sect. 10.3, where the limitations of this filter are also

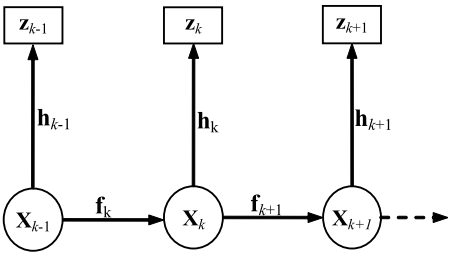


Fig. 10.1 Process and measurement models of a dynamic system

addressed. Section 10.4 is devoted to a detailed review of swarm filters. Finally, a conclusion is made in Sect. 10.5.

10.2 Estimation Problem Formulation

The problem is to estimate the states of a nonlinear dynamic system. Discrete-time state space approach is utilized to model the evolution of the system and the noisy measurements. The states are assumed to be evolving according to the following stochastic model:

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \omega_{k-1}) \tag{10.1}$$

where $\mathbf{f}_k$ is a known, possibly nonlinear function of the state vector $\mathbf{x}_{k-1}$, ω_{k-1} represents the process noise, and k is the time counter. The objective of a nonlinear filter is to recursively estimate $\mathbf{x}_k$ from the available measurements, $\mathbf{z}_k$. In a state estimation problem, the measurements are related to the states via the measurement equation:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k, \nu_k) \tag{10.2}$$

where $\mathbf{h}_k$ is a known, possibly nonlinear function and ν_k is the measurement noise. The noise sequences, ω_k and ν_k, are mutually independent and are assumed to have uniform or multimodal distribution with known covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively. A graphical illustration of the evolution and the measurement models is depicted in Fig. 10.1. The evolution and measurement models of a dynamic system can also be represented by the prior and likelihood probability densities $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and $p(\mathbf{z}_k|\mathbf{x}_k)$, respectively.

The prior density uses the states of the previous time step, $\mathbf{x}_{k-1}$, to obtain a prior estimate of the current states, $\mathbf{x}_k$. Also, the likelihood density uses the states at an instant k to estimate the current observation, $\mathbf{z}_k$. The initial state, $\mathbf{x}_0$, is assumed to have a known PDF, $p(\mathbf{x}_0)$, and to be independent of the noise sequences. Moreover, the estimation problem computes the posterior density, $p(\mathbf{x}_k|\mathbf{z}_k)$.

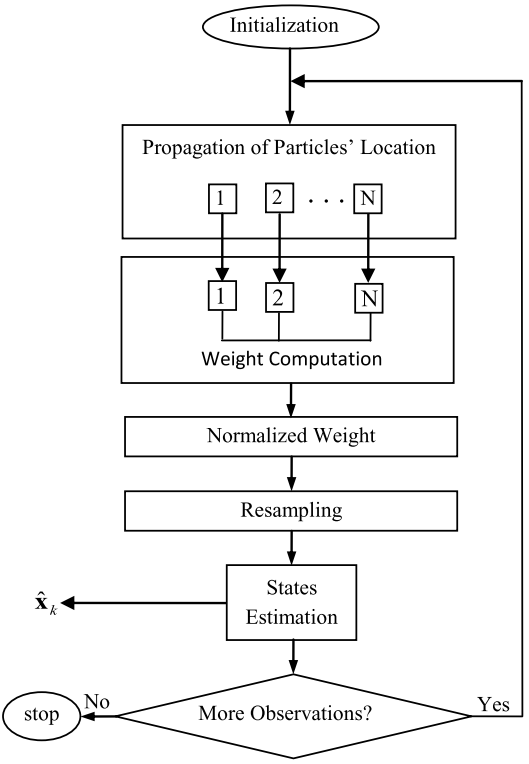


Fig. 10.2 Generic Particle Filter (GPF) algorithm

10.3 Generic Particle Filter and Limitations

The Particle Filter is a sequential Monte Carlo method for Bayesian state estimation in nonlinear systems [9]. The basic idea of PF is to approximate a posterior distribution based on a set of random particles with associated weights. PF has several variants with different sampling and resampling procedures. All sampling procedures utilized in PF can be derived from the Sequential Importance Sampling (SIS) algorithm. When the SIS is associated with a resampling procedure, it will be called Generic PF (GPF) [3]. Figure 10.2 shows the general iterative structure of GPF. A high level description of the sequential steps is shown in this figure. A more extensive introduction to PF can also be found in [9, 18, 19].

PF has a main loop. At first, the position of particles is propagated with their initial distribution. Then the outputs, estimated using each particle, are made. Later, each particle is weighted and the weights are normalized. In a resampling procedure, the particles with small and large weights are eliminated and replicated, respectively. Finally, the current state is estimated using some statistical properties. Figure 10.3 shows the process of PF. In the following subsections, these steps are discussed in detail.

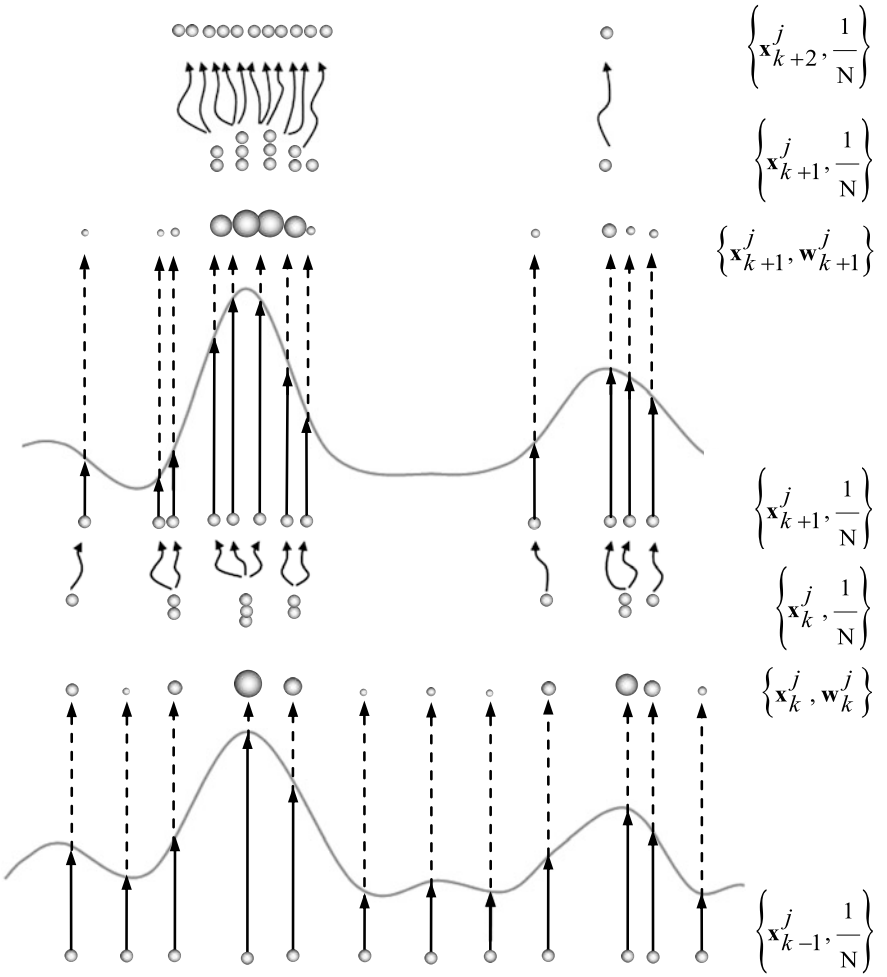


Fig. 10.3 A graphical illustration of particle filter

10.3.1 Initialization

PF has some control parameters that must be set before the execution of the algorithm. Moreover, the initial position of particles (i.e., $\mathbf{x}_0^j$ for $j = 1, \ldots, N$) is initialized using a uniform random generator.

10.3.2 Propagation of Particles' Location

The position of particle j at time $k-1$, defined as $\mathbf{x}_{k-1}^j$, is propagated by importance sampling as follows:

$$\mathbf{x}_k^j = \mathbf{f}_k\left(\mathbf{x}_{k-1}^j, \omega_{k-1}^j\right). \tag{10.3}$$

The prior probability density of the current states is expressed as $p(\mathbf{x}_k^j|\mathbf{x}_{k-1}^j, \mathbf{z}_k)$.

10.3.3 Weight Computation and Normalization

The current output, $\mathbf{z}_k^j$, estimated by particle j at time k, is calculated as follows:

$$\mathbf{z}_k^j = \mathbf{h}_k\left(\mathbf{x}_k^j\right). \tag{10.4}$$

The observation likelihood for each particle is expressed as $p(\mathbf{z}_k^j|\mathbf{x}_k^j)$. After termination of the measurement update, the weight of particle j at time k will be assigned recursively as follows [9]:

$$\mathbf{w}_k^j = \mathbf{w}_{k-1}^j \frac{p(\mathbf{x}_k^j|\mathbf{x}_{k-1}^j)\, p(\mathbf{z}_k^j|\mathbf{x}_k^j)}{q(\mathbf{x}_k^j|\mathbf{x}_{k-1}^j, \mathbf{z}_k)}. \tag{10.5}$$

The PDF $q(\mathbf{x}_k^j|\mathbf{x}_{k-1}^j, \mathbf{z}_k)$ is referred to as the importance, or proposal, density. The choice of the importance density is one of the most critical issues in the design of GPF. The optimal importance density, such as Gaussian distribution, minimizes the variance of weights [20]. Also, the weight of particle j at time k is normalized as follows:

$$\bar{\mathbf{w}}_k^j = \frac{\mathbf{w}_k^j}{\sum_{j=1}^N \mathbf{w}_k^j}. \tag{10.6}$$

10.3.4 Resampling

After a few iterations, most particles will have negligible weights. Computational effort for updating particles with small weight is bulky. This problem is called the degeneracy phenomenon. To avoid the degeneration of particles, a resampling procedure is necessary. The degeneration can be measured in terms of the effective sample size which can be estimated via [21]:

$$N_{\text{eff}} = \frac{1}{\sum_{j=1}^N (\bar{\mathbf{w}}_k^j)^2}. \tag{10.7}$$

It is straightforward to verify that $1 \leq N_{\text{eff}} \leq N$ with the following two extreme cases: (i) if the weights are distributed uniformly (i.e., $\bar{\mathbf{w}}_k^j = 1/N$ for $j = 1, \ldots, N$)

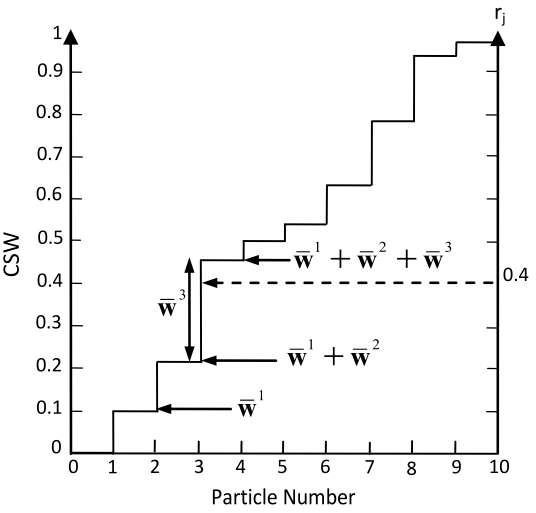


Fig. 10.4 Illustration of a resampling process in PF [3]

then $N_{\text{eff}} = N$, and (ii) if severe degeneracy occurs, then $N_{\text{eff}} = 1$ (see [3]). Resampling is a strategy to overcome degeneracy of samples in SIS. The idea of resampling is to eliminate particles with small weights and copy those with large weights. During this procedure, samples $\{\mathbf{x}_k^j, \bar{\mathbf{w}}_k^j\}$ are replaced with samples $\{\tilde{\mathbf{x}}_k^j, 1/N\}$.

Resampling was first proposed by Gordon, Salmond, and Smith [22], which is illustrated in Fig. 10.4 [3], where CSW stands for the cumulative sum of particles weight, $\sum_{j=1}^{N} \bar{\mathbf{w}}_k^j$, and the random variable, r_j, is uniformly distributed within the interval [0, 1]. For example, if $r_j = 0.4$, then the first particle for which $\sum_{j=1}^{N} \bar{\mathbf{w}}_k^j \geq r_j$ is the third particle. Therefore, a particle with large weight will have a good chance of being resampled several times.

10.3.5 State Estimation

In the final step, the posterior density at time k will be approximated as a discrete density given by [9]:

$$p\left(\hat{\mathbf{x}}_k | z_k\right) \approx \sum_{j=1}^{N} \bar{\mathbf{w}}_k^j \delta\left(\mathbf{x}_k - \tilde{\mathbf{x}}_k^j\right) \tag{10.8}$$

where the normalized weights $\bar{\mathbf{w}}_k^j$ are updated according to Eq. (10.6) and δ is the Dirac Delta-function. Therefore, the approximation of the posterior density can be formulated using some statistical properties (mean, median, confidence intervals, etc.), based on the weight of particles. For example, the states can be estimated

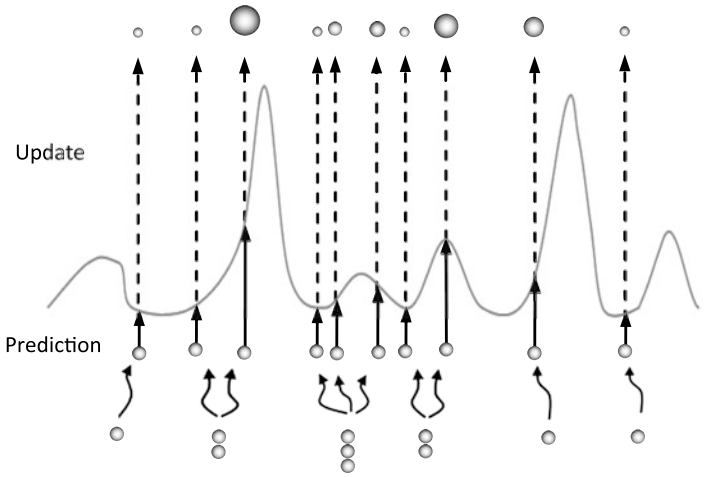


Fig. 10.5 Particle impoverishment due to narrowness of the maximum-likelihood region [14]

based on the average position of particles as follows:

$$\hat{\mathbf{x}}_k = \frac{1}{N}\sum_{j=1}^{N} \tilde{\mathbf{x}}_k^j. \tag{10.9}$$

It can also be shown that as $N \to \infty$ the above approximation approaches the true posterior density [9].

10.3.6 Limitations

Some versions of PF adapt the SIS algorithm to calculate the posterior distribution using the importance sampling density such as Sampling Importance Resampling (SIR) filter [22, 23] and Auxiliary Sampling Importance Resampling (ASIR) filter [24]. Since PF algorithms are suboptimal estimators, they have some accuracy problems. In the following subsections, these problems are discussed in detail.

10.3.6.1 Particle Impoverishment

Particle impoverishment happens when the likelihood is so narrow that the overlapping region of likelihood and prior distribution is quite small [3, 25] and no particle lies within the region of likelihood probability. Thus, many particles are wasted in the low likelihood region, as depicted in Fig. 10.5, and few particles are located in the high likelihood region. Therefore, the weights of most particles become relatively small and their efficiency is decreased; the result of this is the degradation of the estimation accuracy.

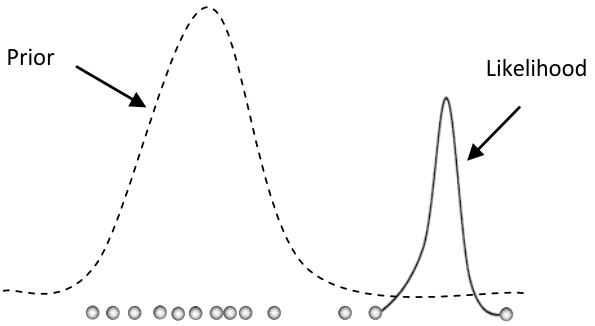


Fig. 10.6 Particle impoverishment when the maximum-likelihood region lies in the tail of the prior distribution [14]

Another reason that causes particle impoverishment is that the new measurements (i.e., the likelihood) appear in the tail of the prior distribution [9, 26] as depicted in Fig. 10.6. This problem happens when the prior distribution is not accurate. In such a case, particles may be distributed far from the maximum-likelihood region, and therefore most particles may have small weights.

Duo to particle impoverishment, only a few particles would have significant weights. Thus, the sample set will contain a few dissimilar particles and sometimes it will drop to a single sample after several iterations. As a result, important samples may be lost.

10.3.6.2 Sample Size Dependency

The sample size has a great effect on the performance of PF. If the sample size is relatively small, then the proper distribution of particles around the true states may not occur. If the sample size is large enough, the whole state space will be covered and the true states will be estimated successfully, but the computational cost is massively increased and the real-time implementation may be impossible.

10.4 Swarm Filters

There have been some systematic approaches proposed recently to solve the limitations of PF. The first approach improves the resampling, similar to binary search [22], systematic resampling [27], and residual resampling [9]. However, these methods are not ideal because the particles with large weights are statistically selected many times. This gradually leads to many repeated points, and consequently the diversity among the particles is lost [3]. The second approach improves the prior distribution with modified PF algorithms such as Extended Kalman Particle Filters (EKPF) [27] and Unscented Particle Filters (UPF) [28]. Another approach, adopted recently, uses swarm intelligence techniques to improve the sampling process. Several swarm filters are found in the literature, such as Particle Swarm Optimized Particle Filter (PSOPF) [14], Ant Colony Optimization Assisted Particle

Filter (PF_{ACO}) [15], Particle Filter with Ant Colony for Continuous Domains [17], and Continuous Ant Colony Filter (CACF) [8]. A detailed description of the swarm filters is discussed in the following subsections.

10.4.1 Particle Swarm Optimized Particle Filter (PSOPF)

The Particle Swarm Optimization (PSO) is a robust stochastic optimization technique based on the movement and intelligence of swarms. It was developed in 1995 by James Kennedy and Russell Eberhart [29]. Individuals interact with each other while they are learning from the swarm experiences and gradually move towards the goal. PSOPF merges PSO into PF to optimize the sampling step of GPF. Figure 10.7 shows the general iterative structure of PSOPF. A high level description of the sequential steps is shown in this figure.

PSOPF has two loops. The main outer loop iterates every time a new measurement is entered. The inner loop iterates to find the best estimates of the current states, corresponding to the entered measurement. At first, the inner loop propagates the initial distribution of particles. Then the output, estimated using each particle, is made. The estimated outputs are compared with the real measurement and the cost of each particle is evaluated using a Gaussian function. Particles use local and global experiences to update their position and velocity in the state space. The inner loop is terminated after the cost function reaches a certain threshold. Then, particles are weighted and normalized. To eliminate the particles with small weights and replicate the ones with large weights, a resampling step is executed. Finally, the current state is estimated. In the following subsections, these steps are discussed in detail.

10.4.1.1 Computation of Cost Function

The cost function of each particle is evaluated using a Gaussian distribution of the difference between the estimated output, $\mathbf{z}_k^j$, and the real measurement, $\mathbf{z}_k$. Therefore, the cost assigned to particle j at time k, is calculated as follows:

$$f_k^j = \exp\left[\frac{-1}{2}\left(\mathbf{z}_k - \mathbf{z}_k^j\right)^T \mathbf{R}_k^{-1}\left(\mathbf{z}_k - \mathbf{z}_k^j\right)\right] \tag{10.10}$$

where $\mathbf{R}_k$ is the observation covariance.

10.4.1.2 Local Best and Global Best Update

PSOPF utilizes a set of moving particles to perform an intelligent search in the state space, looking for the best estimate. Each particle keeps track of its coordinates

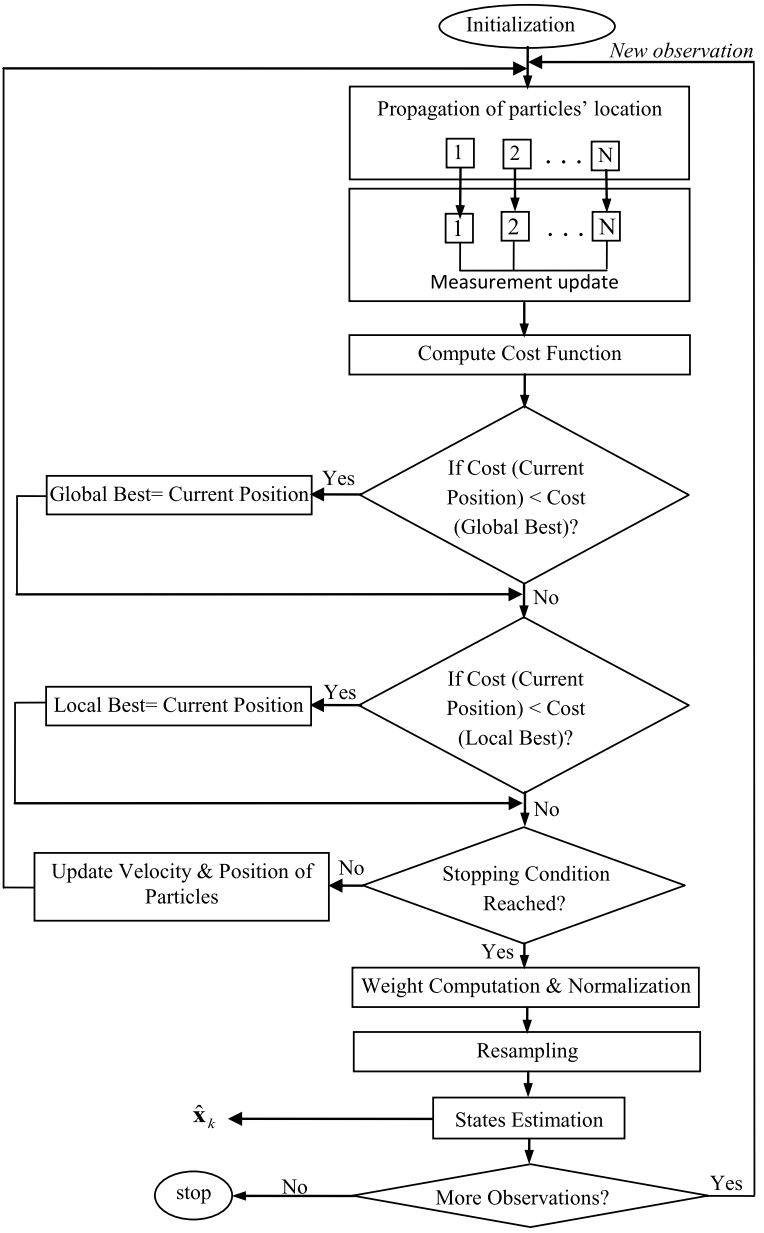


Fig. 10.7 Particle Swarm Optimized Particle Filter (PSOPF) algorithm

corresponding to the best fitness achieved so far. This state is shown by $\mathbf{p}_{\text{pbest}}$. It is the best value met by particle j from entering the current measurement up to now. Another interesting state for the particles is the best value met by all particles in

the population. This state is shown by $\mathbf{p}_{\text{gbest}}$. The logic of PSOPF is to change the velocity of each particle toward the local best $\mathbf{p}_{\text{pbest}}$ and the global best $\mathbf{p}_{\text{gbest}}$ at each time step. It will be discussed in the next section.

10.4.1.3 Velocity and Position Update Rules

In PSOPF, the velocity and position of each particle is updated continuously, as in PSO. The new velocity of particles is calculated using $\mathbf{p}_{\text{pbest}}$ and $\mathbf{p}_{\text{gbest}}$ as follows:

$$\mathbf{v}_k^j = \mathbf{r}_1 \left(\mathbf{p}_{\text{pbest}} - \mathbf{x}_k^j\right) + \mathbf{r}_2 \left(\mathbf{p}_{\text{gbest}} - \mathbf{x}_k^j\right) \tag{10.11}$$

where $\mathbf{r}_1$ and $\mathbf{r}_2$ are positive random numbers with Gaussian probability distribution, i.e., abs[$\mathbf{N}(0, 1)$]. The position vector will simply be updated as follows:

$$\mathbf{x}_{k+1}^j = \mathbf{x}_k^j + \mathbf{v}_k^j. \tag{10.12}$$

In PSOPF, the velocity may become very large and the performance may be degraded. So, the velocity should be limited to an interval $[-\mathbf{v}_{\max}, \mathbf{v}_{\max}]$.

10.4.1.4 Weight Computation and Normalization

After the termination of the inner loop, each particle is weighted according to Eq. (10.5). In this equation, the most popular suboptimal choice of the proposal density is the transitional prior as follows:

$$q\left(\mathbf{x}_k^j \middle| \mathbf{x}_{k-1}^j, \mathbf{z}_k^j\right) = p\left(\mathbf{x}_k^j \middle| \mathbf{x}_{k-1}^j\right). \tag{10.13}$$

Therefore, the weight of particle j at time k will be assigned recursively as follows:

$$\mathbf{w}_k^j = \mathbf{w}_{k-1}^j \, p\left(\mathbf{z}_k^j \middle| \mathbf{x}_k^j\right). \tag{10.14}$$

Finally, the weighted particles are normalized according to Eq. (10.6).

10.4.1.5 Stopping Condition

PSOPF has two loops, each with its own specific stopping condition. The inner loop stops when the cost of the global best estimation ($\mathbf{p}_{\text{gbest}}$) reaches a certain threshold (ε). The outer loop terminates when the measurements are ended.

10.4.2 Ant Colony Optimization Assisted Particle Filter ($\mathbf{PF_{ACO}}$)

The Ant Colony Optimization (ACO) is a stochastic optimization algorithm based on the swarm intelligence of ant colonies. Each ant navigates from the nest to the food sources to find a solution and then communicates with other ants by leaving a pheromone trail within the environment. PF_{ACO} incorporates ACO into PF to optimize the sampling step of GPF. Figure 10.8 shows the general iterative structure of PF_{ACO}.

PF_{ACO} has two loops. The main outer loop iterates every time a new measurement is entered. At first, the initial distribution of ants is propagated. Then, the outputs are estimated by the ants, and consequently each ant is weighted. The inner loop iterates to find the best estimates of the states, corresponding to the entered measurement. In this loop, the threshold parameter, explained in Sect. 10.4.2.4, is computed for each ant. The threshold is used to define a neighborhood around the ant. The next movement of each ant is selected based on a probability function such that it coincides with one of its elite (low cost function) neighbors. The probability function is utilized to model the pheromone distribution over the discrete search space. Each ant is assigned a weight which is proportional to its cost. Moreover, ants use their experience to update the pheromone distribution. The inner loop is stopped when the estimation error reaches the predefined threshold. Then, ants are weighted again, normalized, and resampled. Finally, the current state is estimated. In the following subsections, these steps are discussed in detail.

10.4.2.1 Computation of Probability Function

Each ant chooses its direction using a probability function, defined on the basis of the quality of other ants within the neighborhood. The probability that the ant i selects ant j is expressed as follows:

$$p_{ij}(t) = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum_{s\in N(i)}[\tau_{is}(t)]^{\alpha}[\eta_{is}(t)]^{\beta}} \tag{10.15}$$

where $N(i)$ is the set of all ants which are in the neighborhood of ant i, $\tau_{ij}(t)$ is the pheromone density of the link between ants i and j, as introduced in Sect. 10.4.2.3, and η is a heuristic function, defined as:

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \tag{10.16}$$

where d_{ij} is the distance between ants i and j. If $p_{ij} = 1$, then ant i moves toward ant j. When convergence occurs, it means that most of the ants have moved to a high likelihood region. The parameters α and β determine the relative influence of pheromone trails and the heuristic information, respectively.

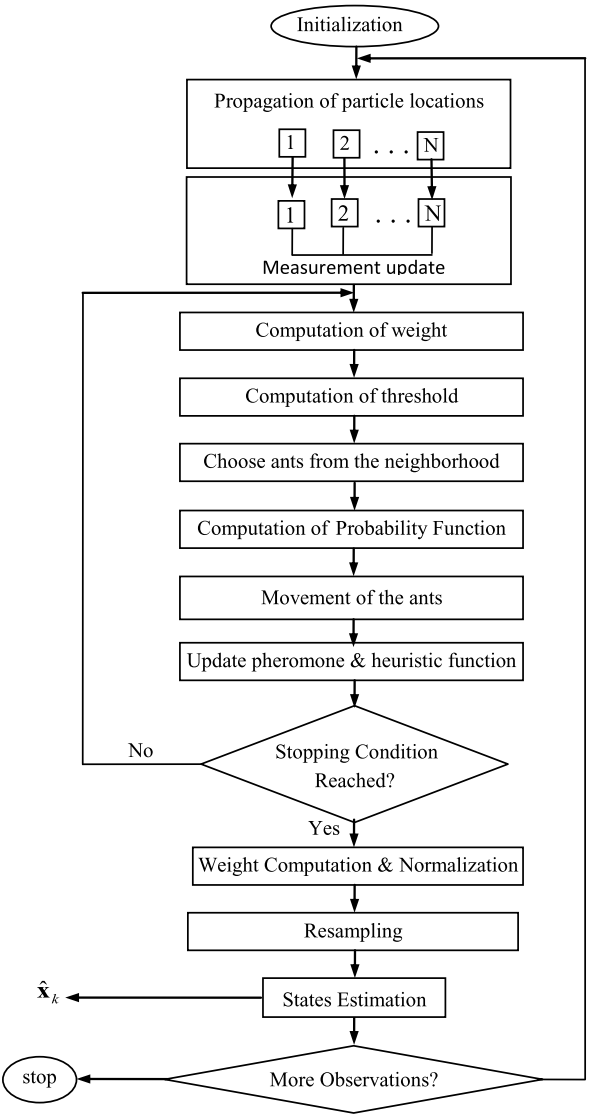


Fig. 10.8 Ant Colony Optimization Assisted Particle Filter (PF_{ACO}) algorithm

10.4.2.2 Movement of the Ants

During the iterations, ants use the current pheromone distribution to move from their current positions to their destinations. The velocity of ant i toward ant j is defined as a random number between zero and d_{ij}.

10.4.2.3 Update Pheromone Distribution

When ant i selects ant j, $\tau_{ij}(t)$ is updated as follows:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{10.17}$$

where $0 < \rho \leq 1$ is the pheromone evaporation rate and $\Delta\tau$ is a constant value that simulates the pheromone deposition over the visited links. Also, when ant j is not chosen by ant i, $\tau_{ij}(t)$ is evaporated as follows:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t). \tag{10.18}$$

The initial pheromone distribution, $\tau_{ij}(0)$, has been proposed to be a function of ants weight [15].

10.4.2.4 Stopping Condition

PF_{ACO} has two loops, each with its own specific stopping conditions. The inner loop stops when the distance between ants i and j becomes less than a certain threshold:

$$\varepsilon^j = c|r|\left(1 - \bar{w}_k^j\right) \tag{10.19}$$

where $\bar{w}_k^j$ is the normalized weight of ant j, r is a normal random number, and c is a constant value; or the number of iterations exceeds a maximum value. The other loop is terminated when the measurements are finished.

10.4.3 Particle Filter with Ant Colony for Continuous Domains

The fundamental idea in continuous ant colony algorithms is to define a continuous pheromone model [30]. The Continuous Ant Colony System (CACS) utilizes a Gaussian PDF to model pheromone distribution over the continuous search space [30]. The Ant Colony Optimization for continuous domains (ACO_R) [31, 32] utilizes a weighted sum of several Gaussian PDF instead of a single one. The Particle Filter with Ant Colony for Continuous Domains incorporates ACO_R into PF to optimize the sampling process of PF. Figure 10.9 shows the general iterative structure of this method.

The Particle Filter with Ant Colony for Continuous Domains has two loops. The outer loop iterates every time a new measurement is entered. Here, the distribution of samples is propagated. The inner loop iterates to find the best estimation. In this loop, the output, estimated using each sample, is made. The estimated outputs are compared with the real measurement and the cost of each sample is evaluated. In this algorithm, the propagated samples and the corresponding cost functions are stored in an external archive to represent the pheromone, as in ACO_R. The Gaussian PDFs

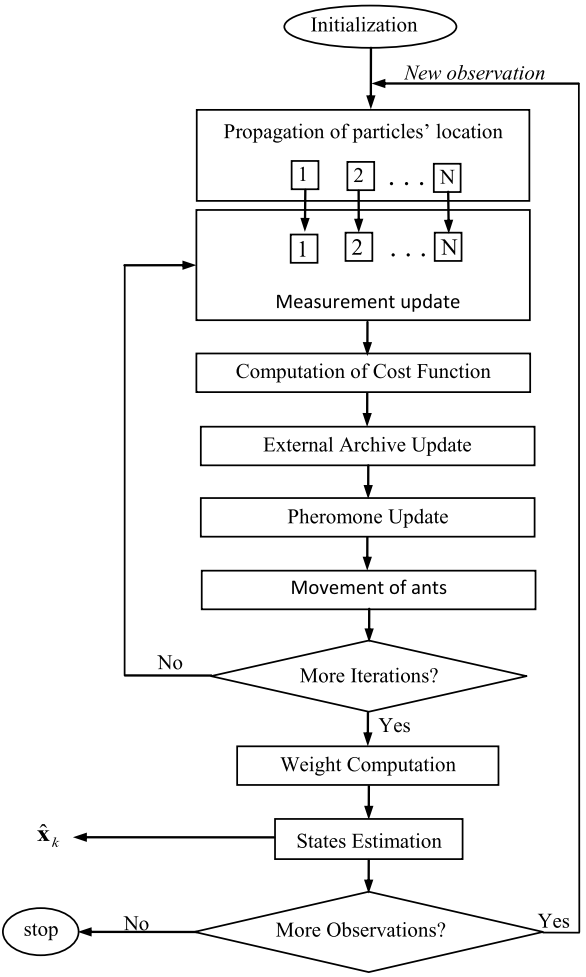


Fig. 10.9 Particle filter with ant colony for continuous domains

are utilized to model the pheromone distribution over the continuous state space. Ants use this pheromone distribution to move from their current position toward the minimum cost destinations. This loop is terminated after a predefined number of iterations. Finally, the remained samples are weighted and the current state is estimated based on the weighted mean. In the following subsections, these steps are discussed in detail.

10.4.3.1 Initialization

At the start of the algorithm, the position of N samples within the archive is initialized by uniform random sampling.

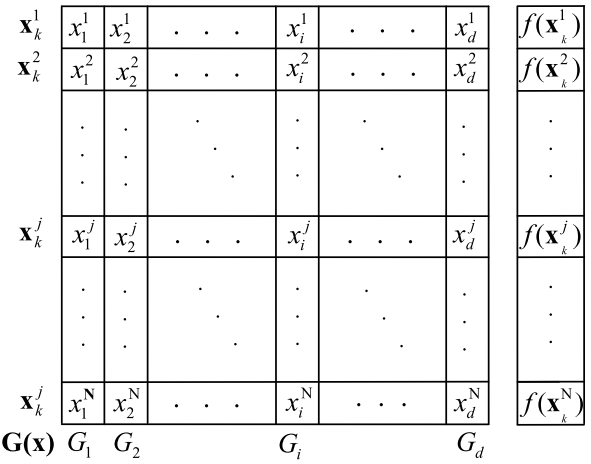


Fig. 10.10 The structure of the external archive [32]

10.4.3.2 Computation of Cost Function

After the propagation and measurement update, the cost function, f_k^j, is calculated as the square error between the real measurement, $\mathbf{z}_k$, and the jth estimated output, $\mathbf{z}_k^j$. Therefore, the cost assigned to sample j at time k is defined as follows:

$$f_k^j = \left|\mathbf{z}_k - \mathbf{z}_k^j\right|^T \left|\mathbf{z}_k - \mathbf{z}_k^j\right|. \tag{10.20}$$

The propagated samples in the archive are sorted according to their cost in ascending order, i.e., $f_k^1 \leq f_k^2 \leq \cdots \leq f_k^N$.

10.4.3.3 External Archive Update

In this method, N best solutions are stored in an external archive. The structure of this archive is shown in Fig. 10.10. During any iteration of the inner loop at time k, M new solutions (i.e., $\mathbf{x}_k^j$, $j = 1, \ldots, M$) found by ants will be added to the archive. Therefore, there will be $M + N$ ants within the archive. To limit the archive length, M worst solutions from the total $M + N$ solutions are then removed and N top solutions are retained.

10.4.3.4 Pheromone Update

Each ant uses d PDFs to perform d selections (corresponding to dimensions 1 to d) to make a complete solution $\mathbf{x}_k^m$. Each PDF is defined using a weighted sum of several Gaussian PDFs, defined as follows [32]:

$$G_i(x) = \sum_{j=1}^{N} \xi_j \frac{1}{\sigma_i^j \sqrt{2\pi}} \exp\left(-\frac{(x-\mu_i^j)^2}{2(\sigma_i^j)^2}\right), \quad -\infty \leq x \leq +\infty \tag{10.21}$$

where ξ_j is the weight of particle j and is calculated according to:

$$\xi_j = \frac{1}{\gamma N \sqrt{2\pi}} \exp\left(-\frac{(j-1)^2}{2\gamma^2 N^2}\right). \tag{10.22}$$

In the above equation, γ is a parameter of the algorithm. When γ is small, the best solutions are strongly preferred, and when it is large, the PDF becomes more uniform. Moreover, μ_i^j denotes the ith position component of sample j within the archive and σ_i^j is the average distance of other solutions from sample j, defined as:

$$\sigma_i^j = \frac{\lambda}{N-1} \sum_{n=1}^{N} \left| x_i^n - x_i^j \right| \tag{10.23}$$

where λ is a parameter of the algorithm; the lower it is, the higher convergence speed of the algorithm is achieved.

10.4.3.5 Movement of the Ants

During any iteration of the inner loop, ants choose their destinations according to a state transition strategy similar to ACO_R.

10.4.3.6 Computation of Weights

After termination of the inner loop, sample j of the archive is assigned a weight as follows:

$$w_k^j = \frac{1}{(f_k^j - \min_{j=1}^{N} f_k^j + \varepsilon)^\beta} \tag{10.24}$$

where $0 \leq \varepsilon \leq 1$ is a threshold, $\beta > 1$ is a parameter of the algorithm, and f_k^j is the cost function of sample j at time k.

10.4.3.7 State Estimation

In this algorithm, the states are estimated based on the weighted mean of archived ants as follows:

$$\hat{\mathbf{x}}_k = \sum_{j=1}^{N} w_k^j \mathbf{x}_k^j. \tag{10.25}$$

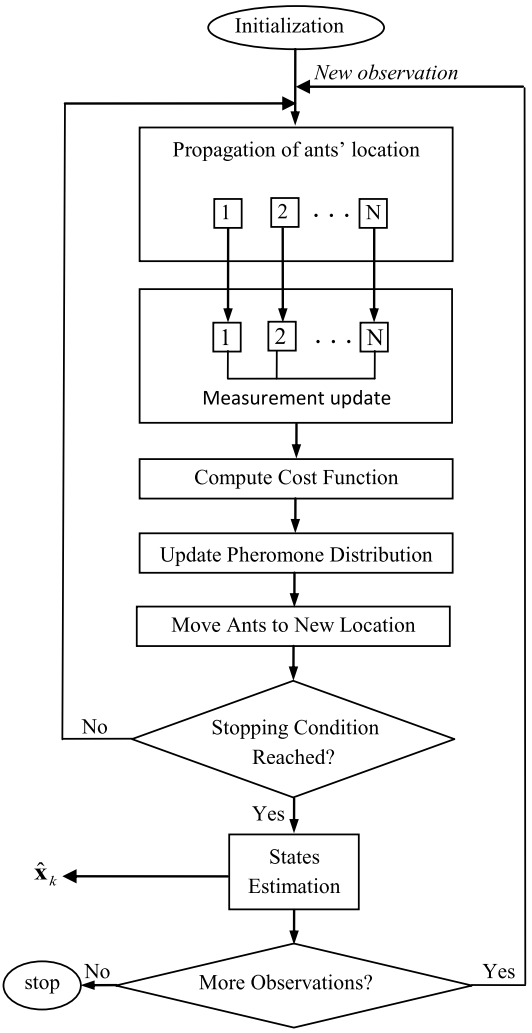


Fig. 10.11 Continuous Ant Colony Filter (CACF) algorithm

10.4.4 Continuous Ant Colony Filter (CACF)

CACF is a heuristic filter, based on the previously developed metaheuristic algorithm known as CACS [30]. It utilizes a colony of moving ants, the average positions of which is returned as the current estimation. Figure 10.11 shows the general iterative structure of CACF. A high level description of the sequential steps is shown in this figure.[1]

[1]CACF code is available at http://ae.sharif.ir/Faculty-Resume/Nobahari.php.

CACF has two loops: a main outer loop, iterating every time a new measurement is entered, and an inner loop, which iterates to find the best estimates of the current states corresponding to the entered measurement. The inner loop propagates the initial distribution of ants, at first. Then, the output, estimated using each ant, is made. The estimated outputs are compared with the real measurement and each ant is assigned a cost based on the quality of its position. Ants use their experience to update the state space pheromone distribution. As in CACS [30], a Gaussian PDF is utilized to model the pheromone distribution over the continuous state space. Ants use this pheromone distribution to move from their current position toward the minimum cost destinations. The destinations are chosen using a normal PDF. The inner loop is terminated after a predefined number of iterations. Finally, the current state estimation is made using a mean operator. In the following subsections, these steps are discussed in detail.

10.4.4.1 Initialization

This algorithm has some control parameters that should be set before the execution of the algorithm. Moreover, the initial position of ants is initialized using a uniform random generator.

10.4.4.2 Computation of Cost Function

Each ant is assigned a cost based on the quality of its current position. The cost function is defined as the square error between the estimated output, $\mathbf{z}_k^j$, and the real measurement, $\mathbf{z}_k$. Therefore, the cost, assigned to ant j at time k, is calculated as follows:

$$f_k^j = \left|\mathbf{z}_k^j - \mathbf{z}_k\right|^T \left|\mathbf{z}_k^j - \mathbf{z}_k\right|. \tag{10.26}$$

In this way, the cost function is calculated at different points of the state space and some knowledge about the problem is acquired, which will be used to update the pheromone distribution.

10.4.4.3 Updating Pheromone Distribution

CACF utilizes the same pheromone model and pheromone updating rule as in CACS [30]. During any iteration, pheromone distribution will be updated using the acquired knowledge from the evaluated points by the ants. Pheromone updating rule of CACF can be stated as follows: during any iteration, the cost is calculated for the new points, explored by the ants. Then, the best point , at time $k-1$, is assigned to $\mathbf{x}_{k-1,\min}$.

Also, the standard deviation of the pheromone distribution, σ_{k-1}, is updated based on the cost of the evaluated points and the aggregation of those points around

$\mathbf{x}_{k-1,\min}$. To simultaneously satisfy the fitness and aggregation criteria, the concept of weighted variance, proposed in [30], is defined for each dimension as follows:

$$\sigma_{k-1}^2 = \frac{\sum_{j=1}^{N} \frac{1}{f_{k-1}^j - f_{k-1,\min}} (x_{k-1}^j - x_{k-1,\min})^2}{\sum_{j=1}^{N} \frac{1}{f_{k-1}^j - f_{k-1,\min}}}. \tag{10.27}$$

Here m is the number of ants. This strategy means that the center of region, discovered during the subsequent iterations, is the last best point and the narrowness of its width depends on the aggregation of the other competitors around the best point [30]. It should be noted that after the termination of the inner loop, the standard deviation of the pheromone distribution is increased by an Expansion Factor (EF) to increase the exploration of the filter when the new measurement is entered.

10.4.4.4 Movement of Ants

During any iteration, ants move from their current position to their destination using the current pheromone distribution. Pheromone distribution is modeled using a normal PDF, the center of which is the best point ($\mathbf{x}_{k-1,\min}$) found from the first iteration and its variance depends on the aggregation of other ants around the best one. Normal PDF permits all points of the continuous state space to be chosen, either close to or far from the best point. As stated in Sect. 10.4.4.1, in the first iteration, the position of ants is initialized using a uniform random generator, whereas for all subsequent iterations, ants choose their destination using the updated pheromone distribution, based on Eq. (10.27).

10.4.4.5 State Estimation

After the termination of the inner loop, the states are estimated based on the average position of top ants:

$$\hat{\mathbf{x}}_k = \frac{1}{N_t} \sum_{j=1}^{N_t} \mathbf{x}_k^j \tag{10.28}$$

where N_t denotes the number of top ants.

10.5 Conclusion

In this chapter, a new class of filters was introduced for nonlinear system state estimation. The presented filters, called swarm filters, model the state estimation problem as a stochastic dynamic optimization problem and utilize swarm intelligence

techniques such as ACO and PSO to solve this problem. Swarm filters can be considered as a subset of a more general class of filters, called heuristic filters, where the heuristic optimization algorithms are utilized to dynamically solve the state estimation problem. Although many heuristic optimization algorithms have been developed by now, the field of heuristic filtering is still in its first days of development and a huge amount of work is left to be performed.

References

1. Siouris, G.M.: An Engineering Approach to Optimal Control and Estimation Theory. Air Force Institute of Technology, New York (1995)
2. Brayson, A.E., Ho, Y.C.: Applied Optimal Control. Blaisdell Publishing Company, Waltham (1969)
3. Ristic, B., Arulampalam, S., Gordon, N.: Beyond the Kalman Filter: Particle Filters for Tracking Applications. Blaisdell Publishing Company, Artech House, London (2004)
4. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. ASME J. Basic Eng. **82**(Series D), 35–45 (1960)
5. Jazwinski, A.H.: Stochastic Processes and Filtering Theory. Academic Press, New York (1970)
6. Julier, S.J., Uhlmann, J.K.: A new extension of the Kalman filter to nonlinear systems. In: AeroSense 11th International Symposium Aerospace Defense Sensing, Simulation and Controls, pp. 182–193 (1960)
7. Carpentier, J., Clifford, P., Fernhead, P.: An improved particle filter for non-linear problems. IEE Proc. Radar Sonar Navig. **146**(1), 2–7 (1999)
8. Nobahari, H., Sharifi, A.: A novel heuristic filter based on ant colony optimization for nonlinear systems state estimation. In: Computational Intelligence and Intelligent Systems, 6th International Symposium, CCIS, Wuhan, China, vol. 316, pp. 20–29 (2012)
9. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. J. Stat. Comput. Simul. **50**(1), 1–23 (1997)
10. Higuchi, T.: Monte Carlo filter using the genetic algorithm operators. J. Stat. Comput. Simul. **59**(1), 1–23 (1997)
11. Park, S., Hwang, J.P., Kim, E., Kang, H.: A new evolutionary particle filter for the prevention of sample impoverishment. IEEE Trans. Evol. Comput. **13**(4), 801–809 (2009)
12. Clapp, T.: Statistical Methods for the Processing of Communication Data. University of Cambridge, Cambridge (2000)
13. Troma, P., Szepesvári, C.: LS-N-IPS: an improvement of particle filters by means of local search. In: Proc. Non-Linear Control Systems (NOLCOS 2001), St. Petersburg, Russia (2001)
14. Tong, G., Fang, Z., Xu, X.: A particle swarm optimized particle filter for nonlinear system state estimation. In: IEEE Congress on Evolutionary Computation, pp. 438–442 (2006)
15. Zhong, J.P., Fung, Y.F., Dai, M.: A biologically inspired improvement strategy for particle filter: ant colony optimization assisted particle filter. Int. J. Control. Autom. Syst. **8**(3), 519–526 (2010)
16. Hao, Z., Zhang, X., Yu, P., Li, H.: Video object tracing based on particle filter with ant colony optimization. In: 2nd IEEE International Conference, Advance Computer Control, Automation and Systems, vol. 3, pp. 232–236 (2010)
17. Yu, Y., Zheng, X.: Particle filter with ant colony optimization for frequency offset estimation in OFDM systems with unknown noise distribution. J. Signal Process. **91**, 1339–1342 (2011)
18. Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. Stat. Comput. **10**, 197–208 (2000)
19. Cappe, O., Godsill, S.J., Moulines, E.: An overview of existing methods and recent advances in sequential Monte Carlo. Proc. IEEE **95**(5), 899–924 (2007)

20. Doucet, A., Godsill, S.J., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. J. Stat. Comput. **10**, 197–208 (2000)
21. Kong, A., Liu, J.S., Wong, W.H.: Sequential imputations and Bayesian missing data problems. J. Am. Stat. Assoc. **89**(425), 278–288 (1994)
22. Gordon, N.J., Salmond, D.J., Smith, A.F.M.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEEE Proc., F, Radar Signal Process. **140**(2), 107–113 (1993)
23. McGinnity, S., Irwin, G.W.: Multiple model bootstrap filter for maneuvering target tracking. IEEE Trans. Aerosp. Electron. Syst. **36**(3), 1006–1012 (2000)
24. Pitt, M., Shephard, N.: Auxiliary particle filters. J. Am. Stat. Assoc. **94**(446), 590–599 (1999)
25. Zang, W., Shi, Z.G., Du, S.C., Chen, K.S.: Novel roughening method for reentry vehicle tracking using particle filter. J. Electromagn. Waves Appl. **21**(14), 1969–1981 (2007)
26. Bruno, M.G.S., Pavlov, A.: Improved particle filters for ballistic target tracking. In: Proc. 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 2(14), pp. 705–708 (2004)
27. Doucet, A., Freitas, D., Gordon, N.J.: Sequential Monte Carlo Methods in Practice. Springer Series in Statistics for Engineering and Information Science. Springer, New York (2001)
28. van der Merwe, R., de Freitas, N., Doucet, A., Wan, E.: The unscented particle filter. In: Dietterich, T.G., Leen, T.K., Tresp, V. (eds.) Advances in Neural Information Processing Systems. NIPS13, vol. 13, pp. 548–590 (2001)
29. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proc. of IEEE Int. Conf. on Neural Networks, Piscataway, NJ, Perth, Australia, vol. 4, pp. 1942–1948 (1995)
30. Pourtakdoust, S.H., Nobahari, H.: An extension of ant colony system to continuous optimization problems. In: Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stutzle, T. (eds.) ANTS 2004. LNCS, vol. 3172, pp. 294–301. Springer, Heidelberg (2004)
31. Socha, K.: ACO for continuous and mixed-variable. In: Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stutzle, T. (eds.) ANTS 2004. LNCS, vol. 3172, pp. 25–36. Springer, Heidelberg (2004)
32. Socha, K., Dorigo, M.: Ant colony optimization for continuous domains. Eur. J. Oper. Res. **185**, 1155–1173 (2008)

Chapter 11
Heuristic Optimal Design of Multiplier-less Digital Filter

Shing-Tai Pan and Cheng-Yuan Chang

Abstract This chapter introduces the design of multiplier-less digital filter based on Canonic Signed Digit (CSD) code. The well-known genetic algorithm (GA) is used to optimal design of the digital filter. Through the CSD coding of the filter parameters, the times of multiplication in the filtering process of a signal can be significantly reduced and then the calculation speed is increased. Among the existing heuristic algorithms, such as Particle Swarm Optimization (PSO), Differential Evolution (DE), Simulated Annealing (SA), etc., GA is the most suitable to CSD design due to its gene-wise crossover property. However, the CSD structure cannot be guaranteed by a general GA after the evolution of chromosomes. Thus in this chapter, a CSD-coded GA is introduced. The CSD-coded GA can reduce the time wasted by trials and errors during the evolution and then accelerate the training speed. Besides, a new hybrid code for the filter coefficients is proposed to improve the precision of the coefficients of a digital filter. Moreover, the design of both finite-impulse response (FIR) filter and infinite-impulse response (IIR) filter are examined. For the IIR filter, the stability problem is very important. Hence, a robust stability criterion is introduced in this chapter for the design of IIR filter.

11.1 Introduction

Recently, the Canonic Signed Digit (CSD) code has been applied to the circuit design [1–5]. This is because the CSD coded design can achieve the reduction of adders/subtractors and shift registers in the circuits, and it can also accelerate the operation of the designed circuit. In implementation of a digital filter, the number

S.-T. Pan (✉)
Department of Computer Science and Information Engineering, National University of Kaohsiung, No. 700, Kaohsiung University Rd., Nan Dis., Kaohsiung 811, Taiwan, R.O.C.
e-mail: stpan@nuk.edu.tw

C.-Y. Chang
Department of Electrical Engineering, Chung Yuan Christian University, Jhongli 320, Taiwan, R.O.C.
e-mail: ccy@cycu.edu.tw

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_11,

of shift registers and adders must be reduced in order to reduce the complexity and to accelerate the operation speed of the filter. This goal can be achieved by designing the parameters of the circuit with CSD. In recent years, many researches have adopted the CSD code to design the systems' parameters. CSD is coded on the binary numeral system. Traditionally, the design of CSD coded systems is performed by designing the real-number codes for the system parameters and then transforming these codes to CSD counterpart. However, this method will lose the accuracy of the designed systems, since an approximation of the designed parameters will be obtained during the transformation. Hence, in this chapter, we will introduce a heuristic method by using the genetic algorithm (GA) to get a better design of CSD coded systems.

The GA was proposed by John Holland in 1960. GA is based on Darwin's theory of evolution: "Survival of The Fittest". It is an evolutionary algorithm which is most widely applied for solving optimization problems. The algorithm claims that the nature of biological evolution is in the genes. Biological characteristics of each species are passed down through gene sequencing from previous generations. "Survival of the Fittest" means that the current generation genes are superior compared to its previous generation, and it is more likely for the current generation to survive in the environment. GA solves problems through gene encoding using a set of parameters while simulating the natural evolution process: selection, crossover, and mutation to find an optimal solution. Based on the characteristic of GA which is similar to natural human evolution, natural crossover and mutation, we can create a new generation in which the performance of each offspring is better than that of the preceding generation. Hence, the optimization and convergence performance of the solution can be obtained [6, 7].

There are many evolutionary algorithms for the heuristic design of the digital filter, for example, Particle Swarm Optimization (PSO), Differential Evolution (DE), and Simulated Annealing (SE), etc. GA is adopted in this chapter for the design of CSD coded digital filters due to the advantage that each gene in the chromosome of GA can be dealt with individually and can be designed to keep the CSD structure. The other algorithms, such as Particle Swarm Optimization (PSO), Differential Evolution (DE), and Steepest Descent Method (SDM), will find it hard to keep the CSD structure during the evolution of each generation. So far, the research on CSD coded filters has focused on the design of FIR filters. Examples can be found in [8–11]. In those papers, the GA based on the CSD code structure was studied for the design of FIR filters. There were two types of design methods. The first method was to check the CSD structure after each evolution of GA. The second method was to transform all filter coefficients to the binary code first, and then transform the designed binary code into a CSD structure code. However, since the CSD structure code can only be an approximation of the binary code, errors are then unavoidable during the transformation.

In this chapter, GA was used to search for the optimum digital filter coefficient with the CSD code. Since the CSD code structure may be destroyed during the GA evolution process, this chapter introduces a CSD-based evolution which completely follows the CSD rule during the evolution and concurrently searches for the optimum filter coefficient with the CSD structure. Besides, since CSD is coded by the

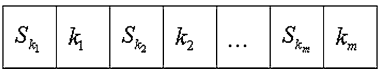


Fig. 11.1 Structure of the hybrid code

binary numerical system, in order to implement the filter in a digital hardware platform, such as FPGA, the hybrid coded method [9–11] is adopted. In this chapter, a new hybrid coded method, called Accumulated Hybrid Code (AHC), was introduced to improve the precision of the optimal design.

This chapter explores the CSD coded design for FIR and IIR filters. The organization of this chapter is as follows: Sect. 11.2 introduces the AHC. Then, the CSD-coded GA is introduced in Sect. 11.3. Based on the methods in Sect. 11.2 and Sect. 11.3, Sect. 11.4 shows the design process and numerical example for an FIR filter. Subsequently, Sect. 11.5 introduces the design process and numerical example for an IIR filter.

11.2 The Accumulated Hybrid Code (AHC)

In this chapter, based on the structure of a power-of-two code [9–11], a hybrid code is used for the coding of the coefficients in digital filters. In order to reduce design error and obtain a solution which is closer to the optimal solution, a new hybrid coded method with better precision is proposed. In this section, the traditional hybrid code is first introduced, and then a new hybrid code, named Accumulation Hybrid Code (AHC), is revealed.

11.2.1 The Traditional Hybrid Code Method

The hybrid code is a coding method which improves the signed binary code. It is similar to the signed binary algorithm. For the signed binary code, the most significant bit (MSB) is a sign bit. A positive number is represented with a sign bit '0' and a negative number is with a sign bit '1'. The structure of the hybrid code is different to that of a signed binary code. A hybrid code is a composition of several signed binary codes. Figure 11.1 shows the structure of the hybrid code which comprises m signed binary codes [9].

In the hybrid code, $k_i, i = 1, 2, \ldots, m$, are binary codes with n bits; $b_{ij}, j = 0, 1, 2, \ldots, n$, denotes the jth bit of k_i, and S_{k_i} is the sign bit of k_i. The magnitude of k_i is calculated as:

$$k_i = \sum_{j=0}^{n} b_{ij} \times 2^j, \quad i \in 1, 2, \ldots, m. \tag{11.1}$$

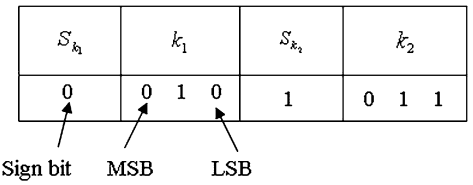


Fig. 11.2 Hybrid code of the real number 0.125

The value of a real number with the structure of the hybrid code in Fig. 11.1 can then be derived as:

$$C = \sum_{i=1}^{m} S_{k_i} \times 2^{-k_i} \tag{11.2}$$

where C represents the value of the hybrid-coded coefficients of a system. A simple example of the hybrid code is shown as follows. A real number 0.125 is coded by the hybrid code and is decoded as $0.125 = 2^{-2} - 2^{-3}$. Figure 11.2 shows the hybrid code of this example.

11.2.2 New Hybrid Code Method

The disadvantage of a binary coding parameter is that the truncation error exists between the original value and the binary coding value. This will cause the design parameters to be far from the optimal solution. This problem, of course, can be improved by increasing the bit length of the binary code. However, this will waste more memory space. In this section, a more precise hybrid code method, Accumulation Hybrid Code (AHC), is introduced without increasing the bit length of the binary code. Hence, a closer to the optimal solution for the designed parameters can be obtained without occupying more memory space. The main feature of AHC is that the exponent in (11.2) is calculated by accumulating the prior k_i and is derived by the following Eq. (11.3) [4]:

$$a_i = \sum_{l=1}^{i} k_l, \tag{11.3}$$

$$C = \sum_{j=1}^{m} S_{k_j} \times 2^{-a_j}. \tag{11.4}$$

According to the example in Fig. 11.2, the value of C can be calculated by using AHC as $2^{-2} - 2^{-5} = 0.21875$. Comparing this value to the value obtained by the traditional hybrid code method, we can see that the precision of the proposed AHC is about 10^{-5} while that of the traditional hybrid code method is about 10^{-3}. So, a more precise result can be expected when AHC is used for designing the system parameters. As a result, a solution closer to the optimal solution can be found.

11.3 CSD-Based Genetic Algorithm

In CSD format, two consecutive digits of a binary coded number cannot be '1' simultaneously. In other words, the product of every two consecutive digits must be zero, i.e., $b_n \times b_{n+1}$. With this feature, fewer bits of nonzero values will appear in the CSD-coded parameters of a system, and hence fewer shifting and adding operations are required for the computation of the system output. Therefore, the CSD coding of system parameters can accelerate the operation speed [9]. Since the CSD format of a parameter is destroyed when it is evolved by GA, in the past research, examining after each evolution whether the CSD format remained or not was necessary. A new evolution process will be active, if the CSD format is destroyed by the previous evolution. However, this method will waste much evolution time. Therefore, in this section, in order to decrease the evolution time of GA, a CSD-based crossover and a CSD-based mutation are introduced to keep the CSD code structure during the evolution process of GA.

11.3.1 Definition of the Fitness Function

In order to identify the quality of a chromosome in GA, the fitness function is usually used to evaluate each chromosome. Different fitness functions are used for different environments. It is important to define the fitness function for a chromosome. In this chapter, the fitness function for the digital filter design is defined as follows. First, the error function is defined as:

$$E_p = \frac{\sum_{\Omega} |H(\Omega) - H_I(\Omega)|^2}{N_s}, \quad 0 \le \Omega \le \pi \tag{11.5}$$

in which $H_I(\Omega)$ is the desired frequency response and N_S is the sampling point, $\Omega = 2\pi \frac{f}{f_s}$ is the digital frequency, f is the analog frequency, f_s is the sampling frequency. The fitness function for a chromosome p is then defined as:

$$\mathit{fitness}(p) = \frac{1}{E_p^2 + 1}. \tag{11.6}$$

11.3.2 CSD-Based Crossover

Concerning the crossover of the chromosomes with the CSD code format, the CSD structure should be retained after the crossover. Usually, crossover will result in a non-CSD structure and then a renewal of the chromosome is required. In the past research, one had to examine the structure of the chromosome after crossover. If a non-CSD structure had been found, the chromosome was renewed. If the bit length

of a chromosome was long, it took a lot of time to examine the chromosome format. Consequently, it wasted enormous computation time to renew a chromosome. Therefore, this chapter introduces a new method for the crossover which does not waste time to renew a chromosome. However, the introduced method must be performed only when the parent chromosomes are already in the CSD format. Then, the offspring will be permanently maintained in the CSD format. The CSD-based crossover is introduced as follows.

1. If $C_{N_1}^{P_1} \& C_{N_1-1}^{P_2} = 1$ then $C_{N_1-1}^{P_2} = \overline{C_{N_1-1}^{P_2}}$.
2. If $C_{N_2}^{P_1} \& C_{N_2+1}^{P_2} = 1$ then $C_{N_2}^{P_1} = \overline{C_{N_2}^{P_1}}$.
3. If $C_{N_1}^{P_2} \& C_{N_1-1}^{P_1} = 1$ then $C_{N_1-1}^{P_1} = \overline{C_{N_1-1}^{P_1}}$.
4. If $C_{N_2}^{P_2} \& C_{N_2+1}^{P_1} = 1$ then $C_{N_2}^{P_2} = \overline{C_{N_2}^{P_2}}$.
5. Proceed to the crossover process of the two points P_1 and P_2 with N_1 and N_2.

The notations $C_{N_i}^{P_i}$, $i = 1, 2$, represent the N_ith bit of the P_ith chromosome, and $\bar{C}$ represents the complement number of C. The symbol "&" represents the logical `and` operation.

11.3.3 CSD-Based Mutation

A mutation for a binary coded chromosome always simply changes 0 to 1, or vice versa. However, for a CSD structure, a mutation will destroy the CSD format. In this chapter, we introduce a method for the mutation so that the CSD format is maintained after mutation.

The CSD structure will be affected only when a mutation for a bit from 0 to 1 occurs. The CSD structure is not affected by the transform for a bit from 1 to 0. That is, the CSD structure might be destroyed only when the new value of the mutation point is 1. Consequently, for retaining the CSD format, we may only focus on the bit of mutation and its two neighbor bits. Based on this idea, the following steps are performed for the CSD-based mutation.

Step 1: Inspect the value of the point of mutation. As mentioned before, if the value of the bit selected for mutation is '1', the structure will not be influenced after mutation. Nothing should be done except for mutation. However, if it is '0', the mutation will transform the bit from '0' to '1' and hence may destroy the CSD structure.

Step 2: Examine the value of the bits adjacent to the mutation bit. If the value of any of them is '1', suitable changes are required. If the values are both '0', we don't need to change them.

Step 3: If any change is necessary in Step 2, find the last significant bit in a series of '1's and change it to zero. Repeat this step until the CSD format is retained.

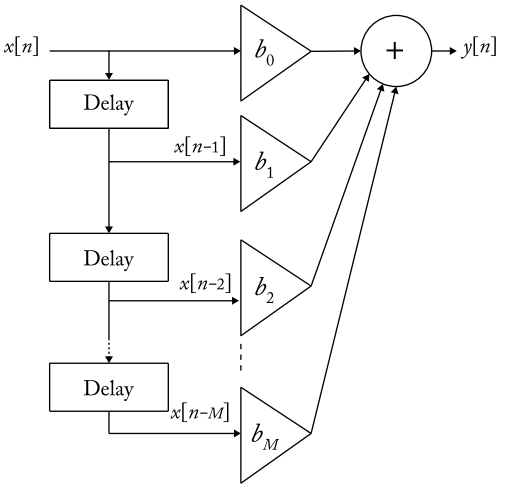


Fig. 11.3 The structure of an FIR

By following the above three steps for mutation, we can retain the CSD structure for the chromosomes. Therefore, the CSD format will be maintained after crossover and mutation. Consequently, re-inspection or renewal of the chromosomes in the offspring is unnecessary. With this method, the evolution time of GA can be reduced.

11.4 CSD-Based Design of FIR Filter

In this section, an FIR filter is designed by using CSD-based GA. As mentioned before, the designed FIR filter with CSD-coded coefficients saves multipliers and adders during the filtering process. The organization of this section is as follows. We will first review the structure of an FIR filter. And then, a method for calculating the order of an FIR filter will be introduced. A design example will be shown at the end of this section.

11.4.1 Overview of Finite-Impulse Response (FIR) Filter

An FIR filter is a non-recursive structural filter with the block diagram shown in Fig. 11.3 [12]. The governing equation of the FIR filter is described as follows:

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + \cdots + b_M x[n-M] \tag{11.7}$$

in which $b_i, i = 0, 1, 2, \ldots, M$ are the coefficients to be designed. The above equation can be written compactly as:

$$y[n] = \sum_{k=0}^{M} b_k x[n-k]. \tag{11.8}$$

If the input signal is an impulse function, $\delta(t)$, Eq. (11.8) can be rewritten as a pulse response equation shown below:

$$h[n] = \sum_{k=0}^{M} b_k \delta[n-k]. \tag{11.9}$$

Using the z transform of $Z\{x[n-k]\} = z^{-k}X(z)$, we can get the z transformation of $y[n]$ from Eq. (11.8) as follows:

$$Y[z] = \sum_{k=0}^{M} b_k z^{-k} X[z]. \tag{11.10}$$

Therefore, the transfer function $H(z)$ is derived as:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{M} b_k z^{-k} = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{M-1} z^{-(M-1)} + b_M z^{-M}. \tag{11.11}$$

Multiplying both sides of Eq. (11.11) by z^M, we can then obtain the following transfer function:

$$H(z) = \frac{b_0 z^M + b_1 z^{M-1} + b_2 z^{M-2} + \cdots + b_{M-1} z + b_M}{z^M}. \tag{11.12}$$

There are $M+1$ coefficients, $b_0, b_1, \ldots, b_M$, which are going to be designed. From Eq. (11.12) it can be seen that there are multiple poles $z = 0$ in the FIR filter. Hence, FIR is an absolutely stable system. Moreover, by letting $z = e^{j\Omega}$, the frequency response of an FIR filter can be expressed as:

$$H(\Omega) = \sum_{k=0}^{M} b_k e^{-jk\Omega}. \tag{11.13}$$

11.4.2 Estimation of the Order

Once the conditions of the to-be-designed filter are set, the order of the filter must be determined. The equation below is an order N_c estimation for the design of a low-pass FIR filter [13]

$$N_c \approx \frac{D_\infty(\delta_p, \delta_s)}{\Delta F} - f(\delta_p, \delta_s)\Delta F + 1 \tag{11.14}$$

where ΔF is the selected frequency band of the filter and

$$D_\infty(\delta_p, \delta_s) = \log_{10}\delta_s\left[a_1(\log_{10}\delta_p)^2 + a_2\log_{10}\delta_p + a_3\right] + \left[a_4(\log_{10}\delta_p)^2 + a_5\log_{10}\delta_p + a_6\right], \tag{11.15}$$

$$f(\delta_p, \delta_s) = 11.01217 + 0.51244[\log_{10}\delta_p - \log_{10}\delta_s] \tag{11.16}$$

Table 11.1 Parameters of the example

Parameter	Value
order	28
number of k_i	3
bit length of k_i	5
δ_p	0.05
δ_s	00001
passband width	0.425π
stopband width	0.425π
transfer frequency band	0.15π

where

$$a_1 = 5.309 \times 10^{-3}, \quad a_2 = 7.114 \times 10^{-2},$$
$$a_3 = -4.761 \times 10^{-1}, \quad a_4 = -2.66 \times 10^{-3},$$
$$a_5 = -5.941 \times 10^{-1}, \quad a_6 = -4.278 \times 10^{-1}$$

in which δ_p is the magnitude of the ripple in the passband of a target filter, and δ_s is the magnitude of the ripple in the stopband of a target filter. The order of the FIR filter N is the smallest integer which is greater than N_c.

11.4.3 Design Example of an FIR Filter [4]

Assume the parameters are $\delta_p = 0.05$ and $\delta_s = 0.001$, and that the frequency band ΔF is 0.15π. By using Eq. (11.14), we have $N_c = 27.024775$. Thus, we set $N = 28$ as the order of the designed filter. For the design of the lowpass filter, we assume the passband width is 0.425π and the stopband width is 0.425π. The parameters used for this design are listed in Table 11.1. According to the earlier described procedures, in this section the verification of the proposed method and an efficiency test are attempted on a symmetrical finite impulse digital lowpass filter of order 28. In addition, the simulated results are also presented.

The parameters for GA in this example are listed in Table 11.2. It is noted that the bit length for each chromosome is 270 because there are 15 coefficients in a chromosome and each coefficient contains three fields k_1, k_2, k_3 with a bit length of 6 for each field (including the sign bit).

The initial generation was simulated randomly with a CSD code. Thereafter, the CSD-based GA introduced in Sect. 11.3 is then used to design the coefficients of the filter. The flow chart for this example is as shown in Fig. 11.4.

The mean square errors (MSEs) between the target filter and the designed filter during the GA evolution from 100 to 100000 iterations are listed in Table 11.3. The designed coefficients with the CSD structure for the FIR filter are listed in

Table 11.2 Parameters of GA

Parameter	Method/Value
selection method	roulette wheel selection
number of chromosomes in each generation	40
bit length in each chromosome	270
crossover probability	0.9
mutation probability	0.01
number of iterations	100000

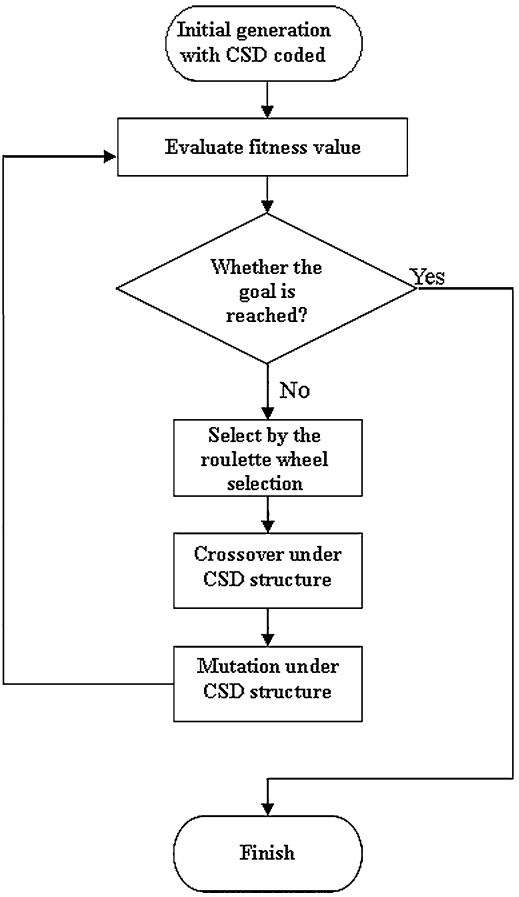


Fig. 11.4 Flow chart for this example

Table 11.4. The comparison of the frequency response between the target filter and the designed filter is depicted in Fig. 11.5. The frequency response in dB scale of designed filter is shown in Fig. 11.6.

Table 11.3 Mean square errors (MSEs) between the target and the designed filter

No. of iterations	MSE	No. of iterations	MSE
100	0.169926069135081	4000	0.000130310610534345
200	0.0377396821132693	5000	0.000129315464946452
300	0.00347155998172241	10000	9.35633135984915e-005
400	0.00261887149888509	20000	8.06307730288536e-005
500	0.00260161871871188	30000	6.35793095232774e-005
1000	0.00251135715569537	40000	6.35752031291314e-005
2000	0.000251245587124766	50000	6.35731617230672e-005
3000	0.000130830139548433	100000	6.34650359562179e-005

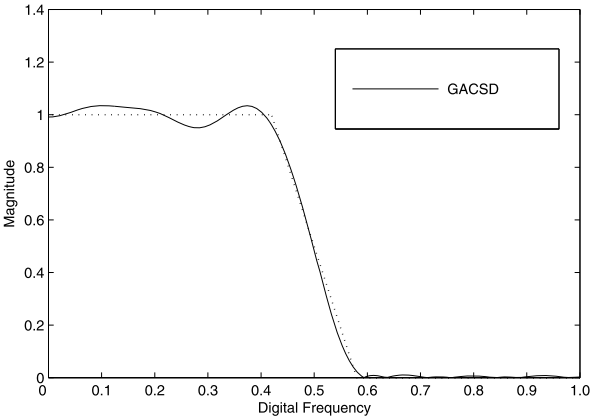


Fig. 11.5 Comparison of the frequency response between the target filter and the designed filter

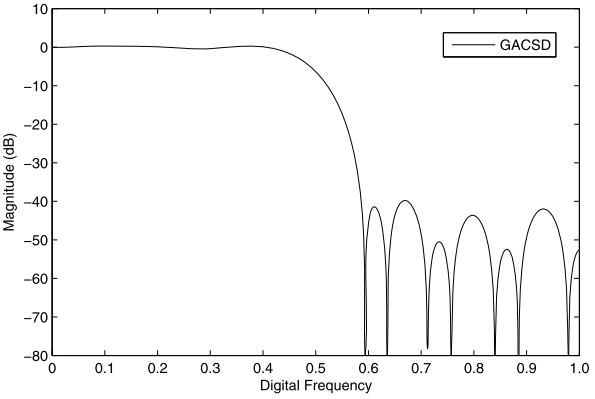


Fig. 11.6 Frequency response (dB) of the designed filter

Table 11.4 Designed filter coefficients with CSD structure

Coefficients	CSD code (a sign bit in the left-end position for each k_i)	Value
C(0)	000000 100000 101000	−0.0039063
C(1)	110000 000010 100000	−1.5259e-005
C(2)	001010 100010 000001	0.00085449
C(3)	101000 000100 000010	−0.0036011
C(4)	001001 001000 000000	0.0019684
C(5)	001000 000000 000000	0.011719
C(6)	101000 100000 101000	−0.0078278
C(7)	100100 000001 100100	−0.033203
C(8)	000100 100000 001000	0.00024414
C(9)	000101 000001 000010	0.050781
C(10)	110000 001010 010101	−1.5244e-005
C(11)	100100 100010 100000	−0.09375
C(12)	001000 000001 000010	0.0063477
C(13)	000010 000010 000100	0.31641
C(14)	000010 000000 101000	0.49902
C(15)	000010 000010 000100	0.31641
C(16)	001000 000001 000010	0.0063477
C(17)	100100 100010 100000	−0.09375
C(18)	110000 001010 010101	−1.5244e-005
C(19)	000101 000001 000010	0.050781
C(20)	000100 100000 001000	0.00024414
C(21)	100100 000001 100100	−0.033203
C(22)	101000 100000 101000	−0.0078278
C(23)	001000 000000 000000	0.011719
C(24)	001001 001000 000000	0.0019684
C(25)	101000 000100 000010	−0.0036011
C(26)	001010 100010 000001	0.00085449
C(27)	110000 000010 100000	−1.5259e-005
C(28)	000000 100000 101000	−0.0039063

11.5 CSD-Based Design of IIR Filter

This section introduces the IIR filter design. Similarly to Sect. 11.4, the CSD-based GA is used for designing the filter. However, the stability problem which did not need to be considered in the FIR filter design becomes an important issue and should be addressed in the IIR filter design. Hence, the organization of this section is as follows. We will first review the structure of an IIR filter. Subsequently, the stability problem for the IIR filter is examined. Finally, a design example is presented at the end of this section.

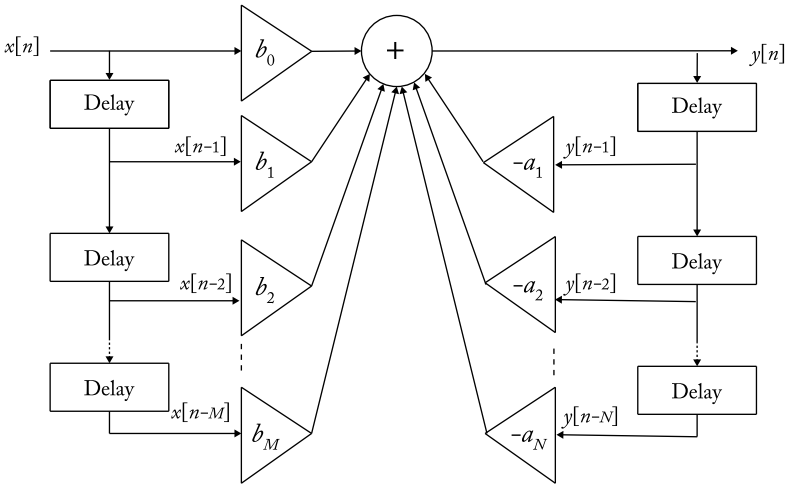


Fig. 11.7 Architecture of IIR filters [5]

11.5.1 Overview of Infinite Impulse Response (IIR) Filters

An IIR filter is a filter with the architecture of the output feedback. An architecture which is used most often is depicted in Fig. 11.7 [14]. The governing equation of an IIR filter is then described as follows:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{11.17}$$

in which $a_k, k = 0, 1, \ldots, N$, and $b_k, k = 0, 1, \ldots, M$, are the coefficients of the IIR filter to be designed. It is obvious that the output of the filter depends on not only the current and past inputs but also on the past outputs, which leads to a fact that the output of the filter will depend on the infinitely many past inputs due to the iteration process in Eq. (11.17). Hence, the filter is called the Infinite Impulse Response (IIR) filter. According to Eq. (11.17), the transfer function of the IIR filter can be derived as:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + \cdots + b_M z^{-M}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + \cdots + a_N z^{-N}} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}}. \tag{11.18}$$

However, without loss of generality, we adopt the function

$$H(z) = \frac{\sum_{k=0}^{m} n_k z^{-k}}{1 + \sum_{k=1}^{n} a_k z^{-k}} \tag{11.19}$$

as the transfer function of the designed IIR filters [14]. In the following paragraphs, the parameters a_k and b_k will be found, under the CSD coded format, via the proposed strategy so that the error between the frequency response of the designed IIR

filters and the desired frequency response is minimal. Moreover, all the poles of $H(z)$ will be placed in the disk $D(\alpha, r)$, i.e., the IIR filters are $D(\alpha, r)$-stable.

11.5.2 Stability Criterion for an IIR Filter

The main difference between the design of an FIR filter and an IIR filter is that the stability problem should be considered in the design of an IIR filter while it is unnecessary in the design of an FIR filter, since an FIR filter is always stable as we have mentioned in Sect. 11.4. Hence, this section will introduce a stability criterion for the IIR filter design. Before proceeding, the following definitions are introduced to help the description of this section.

Definition 11.1 A polynomial $d(z)$ is P-stable if all solutions of the equation $d(z) = 0$ lie inside the unit circle [15].

Definition 11.2 A polynomial $d(z)$ is $PD(\alpha, r)$-stable if all solutions of the equation $d(z) = 0$ are within the disk $D(\alpha, r)$ centered at α with radius r, in which $r > 0$ and $|\alpha| + r < 1$ [15].

Definition 11.3 Let the transfer function of an IIR filter be described as $H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}}$. If the denominator of the $H(z)$, $a(z) = \sum_{k=0}^{N} a_k z^{-k}$, is P-stable, then the IIR filter is stable. Moreover, if $a(z)$ is $PD(\alpha, r)$-stable, then the IIR filter is $D(\alpha, r)$-stable [15].

In the following paragraph, a theorem which is useful in the evolution of GA for design of a robust stable IIR filter is introduced. First, we introduce a useful theorem as follows:

Theorem 11.1 *If $f(z)$ is analytic in a bounded domain ψ and continuous in the closure of ψ, then $|f(z)|$ takes its maximum on the boundary of ψ* [16].

The following theorem will provide a boundary test criterion for the $PD(\alpha, r)$-stability of a polynomial.

Theorem 11.2 *Consider the polynomial $d(z) = \sum_{k=0}^{n} a_k z^{-k}$. If the following inequality (11.20) is satisfied, then all the solutions of $d(z) = 0$ will lie inside a disk $D(\alpha, r)$, i.e., the polynomial $d(z)$ will be $PD(\alpha, r)$-stable with $|\alpha| + r < 1$ and $|\alpha| \leq r$* [15]

$$\left| \sum_{k=1}^{n} \frac{a_k}{a_0} \left(r e^{-j\theta} + \alpha \right)^{-k} \right| < 1, \quad \forall \theta \in [0, 2\pi]. \tag{11.20}$$

11.5.3 Design of an IIR Filter

Theorem 11.2 is used to check whether the poles of an IIR filter lie inside the disk $D(\alpha, r)$. This stability check should be performed in each generation of an evolutionary algorithm used to design an IIR filter. Before proceeding, we first define the vector of the coefficients in the denominator of $H(z)$, $a = (1\ a_1\ a_2\ \ldots\ a_n)$, as a chromosome of GA. The following definition is helpful to describe the introduced method.

Definition 11.4 A chromosome $a = (1\ a_1\ a_2\ \ldots\ a_n)$ is stable if the corresponding polynomial $a(z) = \sum_{k=0}^{n} a_k z^{-k}$ is P-stable. Moreover, it is $D(\alpha, r)$-stable if the corresponding polynomial is $PD(\alpha, r)$-stable [15].

After the derivation of the stability criterion for the design of an IIR filter, based on GA, a design strategy of a robust CSD coded stable IIR filter is then proposed as follows. Moreover, the flowchart of the design procedure is depicted in Fig. 11.8 for clarification.

Step 1. Initial generation. Define the coefficients of the denominator of $H(z)$, $a(z) = \sum_{k=0}^{N} a_k z^{-k}$, as a chromosome $a = (a_0\ a_1\ a_2\ \ldots\ a_n)$. Generate the initial generation of the chromosomes based on the CSD code format.

Step 2. Check the stability property. Check the stability of the chromosome according to Theorem 11.2 and Definition 11.4. If a chromosome does not satisfy the criterion in Theorem 11.2, then regenerate a new chromosome based on the CSD code format.

Step 3. Evaluate the fitness value of the chromosomes. Evaluate the fitness value of the chromosomes according to (11.6).

Step 4. Check whether the result is acceptable. If the result is acceptable or the number of iterations is larger than an assigned maximum number, go to the end of this procedure (Step 7), otherwise go to the next step.

Step 5. Generate offspring. Generate new chromosomes by the crossover and mutation based on the CSD format which are proposed in Sect. 11.3.

Step 6. Check the stability criterion. Check the new chromosomes generated from Step 5 to see whether they satisfy the stability criterion in Theorem 11.2. Go to Step 3 if they do, or go back to Step 5.

Step 7. End of this procedure.

11.5.4 Design Example of an IIR Filter [5]

Suppose that the transfer function of a CSD coded IIR filter is described as $H(z) = \frac{b_0 + b_1 z^{-1}}{a_0 + a_1 z^{-1} + a_2 z^{-2}}$, in which the coefficients $a_i, i = 0, 1, 2$, and $b_i, i = 0, 1$, are all designed in the CSD format. The target frequency response is $H_I(\Omega)$ as depicted in

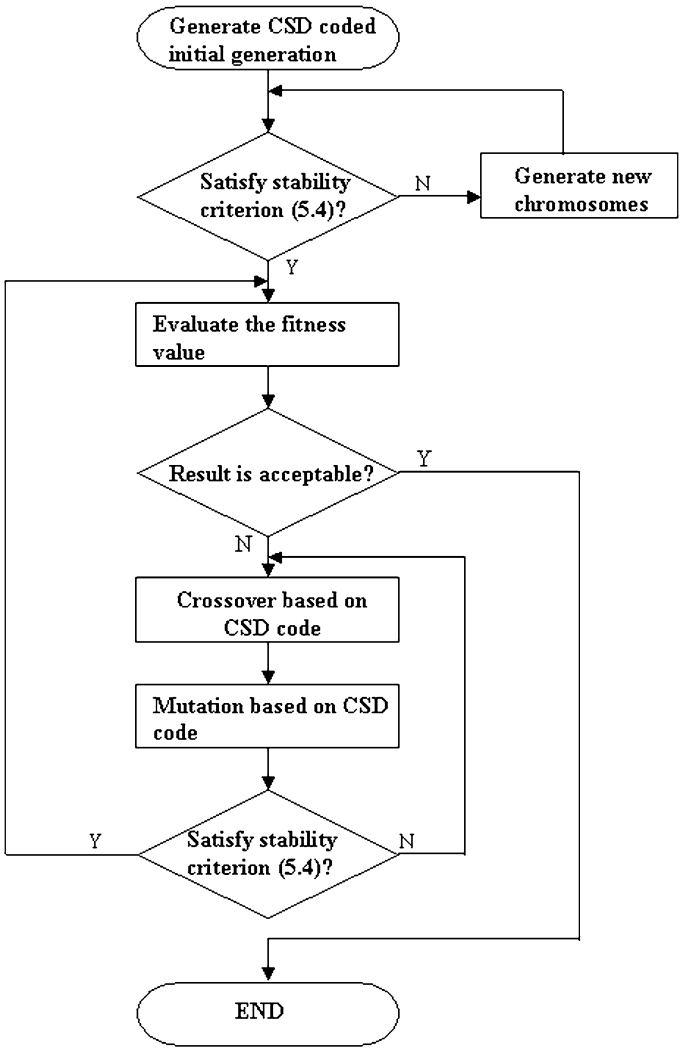


Fig. 11.8 The design procedure of the proposed strategy

Fig. 11.9. The sampling rate, passband edge frequency and stopband edge frequency of $H_I(\Omega)$ are set to be 9600, 1650, and 2800 Hz, respectively. The CSD coded IIR filter is designed to minimize the mean square error between $H(\Omega)$ and $H_I(\Omega)$, and to have the poles and zeros of $H(z)$ lie inside the disk $D(0.3, 0.7)$. That is, the IIR filter will be robustly $D(0.3, 0.7)$-stable.

According to the steps of the design strategy proposed in Sect. 11.5.3, in which the parameters of GA are listed in Table 11.5, the coefficients of the IIR filter are designed after 35000 generations of GA in the CSD coded format and are listed in Table 11.6.

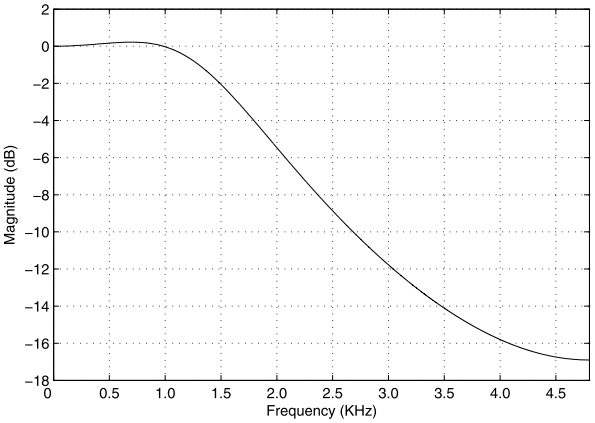


Fig. 11.9 The desired frequency response

The transfer function of designed CSD coded IIR filter is then described as:

$$H(z) = \frac{-0.49902 - 0.125z^{-1}}{-1.2505 + 1z^{-1} - 0.375z^{-2}}. \tag{11.21}$$

The frequency response of the target IIR filter and the designed IIR filter are depicted in Fig. 11.10. It can be seen that the designed frequency response is very close to that of the desired filter. Moreover, the poles of the transfer function of the IIR filter are depicted in Fig. 11.11. It is also obvious that all the poles lie inside the disk $D(0.3, 0.7)$. That is, the CSD coded IIR filter is robustly $D(0.3, 0.7)$-stable.

Table 11.5 Parameters of GA

Parameter	Value
selection method	roulette wheel selection
number of chromosomes in each generation	40
bit length for each chromosome	90
crossover probability	0.9
mutation probability	0.08
No. of generations	35000

Table 11.6 CSD code of the coefficients of the designed IIR filter

Coefficients	Real number code	CSD code (a sign bit in the left-end position for each k_i)
a_o	−1.2505	100000 100010 101001
a_1	1	000000 010010 100000
a_2	−0.375	100010 100010 100000
a_3	−0.49902	100001 001000 100001
a_4	−0.125	100010 0000001 00001

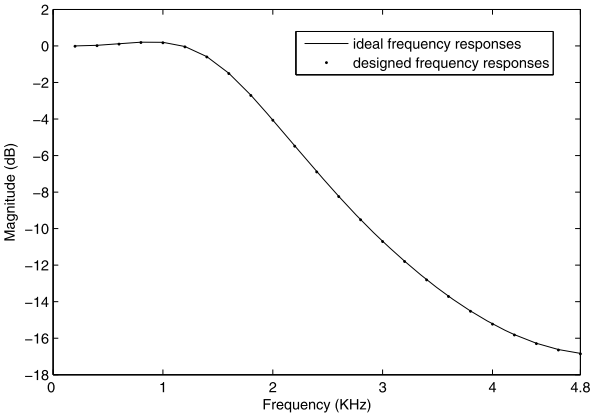


Fig. 11.10 Comparison between the frequency responses of the ideal filter and the designed filter

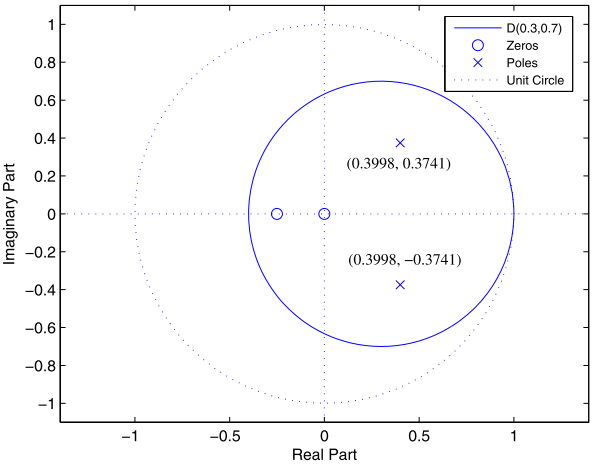


Fig. 11.11 Poles of the designed IIR filter

References

1. Hewlitt, R.M., Swartzlantler, E.S. Jr.: Canonical signed digit representation for FIR digital filters. In: IEEE Workshop on Signal Processing Systems, 11–13 Oct., pp. 416–426 (2000)
2. Cabal-Yepez, E., Carozzi, T.D., de Romero-Troncoso, R.J., Goughc, M.P., Huberc, N.: FPGA-based system for frequency detection of the main periodic component in time series information. Digit. Signal Process. **18**, 1029–1044 (2008)
3. Tang, Z., Zhang, J., Min, H.: A high-speed, programmable, CSD coefficient FIR filter. IEEE Trans. Consum. Electron. **48**(4), 834–837 (2002)
4. Pan, S.-T.: A canonic-signed-digit coded genetic algorithm for designing finite impulse response digital filter. Digit. Signal Process. **20**(2), 314–327 (2010)
5. Pan, S.-T.: CSD-coded genetic algorithm on robustly stable multiplier-less IIR filter design. Math. Probl. Eng. **2012**, 560650 (2012), 15 pages. doi:10.1155/2012/560650
6. Haupt, R.L., Haupt, S.E.: Practical Genetic Algorithms. Wiley, New York (1998)
7. Davis, L.: Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York (1991)
8. Zhao, Q., Tadokoro, Y.: A simple design of FIR filters with powers-of-two coefficients. IEEE Trans. Circuits Syst. **35**(5), 566–570 (1988)

9. Khoo, K.Y., Kwentus, A., Willson, A.N. Jr.: A programmable FIR digital filter using CSD coefficients. IEEE J. Solid-State Circuits **31**, 869–874 (1996)
10. Lee, H.R., Jen, C.W., Liu, C.M.: A new hardware-efficient architecture for programmable FIR filters. IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process. **43**, 637–644 (1996)
11. Park, I.C., Kang, H.J.: Digital filter synthesis based on an algorithm to generate all minimal signed digit representations. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **21**, 1525–1529 (2002)
12. de Vegtea, J.V.: Fundamentals of Digital Signal Processing. Prentice Hall, Englewood Cliffs (2001)
13. Ifeachor, E.C., Jervis, B.W.: Digital Signal Processing: A Practical Approach. Addison-Wesley, London (1993)
14. Uesaka, K., Kawamata, M.: Evolutionary synthesis of digital filter structures using genetic algorithm. IEEE Trans. Circuits Syst. II **50**, 977–983 (2003)
15. Pan, S.-T.: Design of robust D-stable IIR filters using genetic algorithms with embedded stability criterion. IEEE Trans. Signal Process. **57**(8), 3008–3016 (2009)
16. John, W.D.: Applied Complex Variables. Macmillan, New York (1967)

Chapter 12
Hybrid Correlation-Neural Network Synergy for Gait Signal Classification

Saibal Dutta, Amitava Chatterjee, and Sugata Munshi

Abstract This chapter presents a thorough discussion on the development of a robust algorithm for pathological classification of human gait signals. The technique involves the extraction of time and frequency domain features of the correlograms obtained by cross-correlating the gait signals with a reference, and subsequently employing a pre-trained Elman's recurrent neural network (ERNN) for automatic identification of healthy subjects and those with neurological disorder, and also the type of disorder. To assess the performance of the algorithm, stance, swing, and double support intervals (expressed as percentages of stride) of 63 subjects, either healthy, or suffering from Parkinson's disease (PD), Huntington's disease (HD), or Amyotrophic Lateral Sclerosis (ALS), have been processed by the proposed algorithm for a period of approximately 300 s. The performances of ERNNs are also compared with those already reported for back propagation neural network (BPNN), learning vector quantization (LVQ), and least-square support vector machine (LS-SVM) based classification algorithms. With time-domain features, the proposed modular ERNNs outshined the other classifiers by attaining 90.3–98.5 % classification accuracy for binary classification jobs, and an accuracy as high as 87.1 % for the four-class classification problem. With frequency-domain features, classification into healthy and pathological subjects has been studied, and in this case also, the best performance of 81.6 % mean accuracy was achieved employing ERNN.

S. Dutta (✉)
Electrical Engineering Department, Heritage Institute of Technology, Chowbaga Road, Anandapur, Kolkata, West Bengal 700107, India
e-mail: saibal_dutta2001@yahoo.com

A. Chatterjee · S. Munshi
Electrical Engineering Department, Jadavpur University, Kolkata 700032, India

A. Chatterjee
e-mail: cha_ami@yahoo.co.in

S. Munshi
e-mail: sugatamunshi@yahoo.com

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_12,

12.1 Introduction

Going by the lexicon, the term 'gait' represents the manner of moving on foot. Movement is an important routine activity of all human beings. Any impediment in movement, substantially downgrades the quality of our life. On the other hand, human gait signal has the potential to serve as an important biometric trait primarily due to its inconspicuousness, as it can be acquired from a distance without the prior knowledge of the subject [5]. The investigations related to gait as a distinguishing feature were first attempted a few decades back, from a medical/behavioral viewpoint [10, 20]. This was followed by several attempts to investigate the problem of gait recognition, in the context of capturing and analyzing gait signals [9, 19, 27, 30, 33].

So far as medical applications are concerned, the domain of gait and human movement science has stolen much of the limelight with the appreciation of the fact that locomotor dysfunctions demand considerable medical attention, involve high costs of treatment, and may also turn out to be fatal in certain cases [25]. At a certain point of time, statistics revealed that 90 % of the adults with cerebral palsy (CP) in the U.S.A. lacked access to periodic health checks-up [31], although more than 50 % of the CP hemiplegics were required to have constant personal assistance [43]. The elderly people face progressive gait disorder, which enhances the possibility of death due to falls and bone fractures [46]. Half of these victims of fall, if left unattended for more than two hours, are exposed to the danger of succumbing to dehydration, hypothermia, pneumonia, pulmonary embolism (which accounts for 38 % of deaths in hip fracture falls), rhabdomyolysis (which is a toxic breakdown of muscle fibres), and pressure ulcers [2].

A large number of research workers have examined gait signals by different methods. The methods put forward so far are primarily aimed at clustering gait signals into young and old categories [1, 32, 34, 47]. Reported works on analysis of gait signals for identifying neurological disorders in subjects, and also distinguishing them from healthy subjects, have been few and far between. The use of artificial neural networks (ANNs) for automated identification of gait patterns has already been reported [1, 3]. The work referred to considered eight subjects under three gait conditions, namely, normal gait, a simulation of leg length difference, and a simulation of leg-length difference [1]. The features from hip–knee joint angle diagrams were utilized to train the ANN, which was subsequently used for identifying the type of gait. A three-class classification problem solution yielded a classification ratio of 83.3 %.

Young–elderly classification of gait signals plays a significant role in identifying the onset of gait related disorder in aged people, so that preventive measures can be taken against fall [14, 32]. In an investigation [3], statistical features were processed by support vector machine based classifier for binary classification of gait signals. Twenty-four such features were derived from the minimum foot clearance (MFC) data of 58 subjects to obtain classification into young and elderly gaits. A mean classification accuracy of 83.3 % was achieved.

Automated determination of whether a subject is suffering from neurological diseases, and also the type of disease, is another object of interest in investigations on

identification of gait pattern. However, as compared to a young–elderly differentiation method, development of a disease identification scheme is a more complicated problem, as it is a multi-class (more than two) classification problem.

The present chapter presents the study on the development of an automated gait identification tool which can automatically determine whether or not the subject under consideration is a healthy one, and if not, then whether the source of neurological disorder in the pathological subject is due to Parkinson's disease (PD), Huntington's disease (HD), or Amyotrophic Lateral Sclerosis (ALS). Thus, the overall purpose of the proposed method is to predict whether an unknown subject under consideration is healthy or suffering from one of the three major neurological diseases.

The basic problem of designing such gait identification tools can be divided into two subworks:

(i) Suitable feature extraction from input gait signals;
(ii) Designing a suitable classification algorithm to utilize those extracted features.

Feature extraction can be conventionally carried out from input signals by using various mathematical tools like statistical methods, Fourier transform, wavelet transform based methods, etc. This chapter attempts to develop efficient feature extraction algorithms employing correlation techniques, instead of the above mentioned methods. The chapter explores cross-correlation as a potential tool for feature extraction of gait signals. Both time and frequency domain based features from correlations are analyzed to develop powerful gait signal classification algorithms. The cross-correlation technique has so far been conveniently utilized in several engineering fields, e.g., in instrumentation, robotics, and remote sensing applications. The cross-correlation technique has also been successfully used in sonar and radar systems for range and position detection. The chapter also aims at investigating the usefulness of employing Elman's recurrent neural network (ERNN) based classifiers, on the basis of the features extracted employing cross-correlation techniques. ERNN has found successful applications in the domains of function approximation, prediction, and pattern recognition. Hence, the goal of this chapter is to develop computer-based highly reliable automatic classification algorithms which can effectively classify gait signals, utilizing cross-correlation based feature extraction and neural network based classification techniques.

12.2 The Acquisition of Gait Signals

Several researchers have, over a period of time, employed different methodologies for the analysis of gait signals. Most of these schemes employ different types of acquisition procedures for recording various signals that are in some way related to gait and posture of a human body. Subsequently, a variety of mathematical techniques have been utilized for extracting meaningful features from these acquired signals. Many of these methodologies are based on the recording of [11]:

- Step frequency or cadence,

- Step length or length of one step,
- Stride length or distance between two steps,
- Stride interval (stance, swing, and double support interval),
- Reaction force or force exerted by a person on the floor while walking,
- Orthopaedic angles or orientation of limb segments,
- Electromyographic (EMG) activity of the involved musculature during walking,
- Minimum foot clearance (MFC) during walking, during the mid swing phase of the gait cycle.

Under the scope of the present work, benchmark gait signals that are freely available from the physionet database [35] have been utilized. The database contains real-life gait signals acquired from both healthy and pathological subjects having neurological disorders due to Parkinson's disease (PD), Huntington's disease (HD), and Amyotrophic Lateral Sclerosis (ALS). The database was contributed by Hausdorff et al. [16, 17], which includes 16 healthy subjects (2 men and 14 women) aged 20–74 years, 15 PD subjects (10 men and 5 women) aged 44–80 years, 19 HD subjects (6 men and 13 women) aged 29–71 years, and 13 ALS subjects (10 men and 3 women) aged 36–70 years. Height and weight of the pathological subjects recorded in the database were not significantly different from those of the healthy subjects. This database also maintains a measure of disease severity or duration, to indicate the extent to which a subject of the database is affected by PD, HD, or ALS. The database uses Hohn and Yahr score for the subjects suffering from PD. A higher value of this score indicates a more advanced condition of the disease. The score varies from 1.5 to 4, for the PD subjects under consideration in this chapter. For 60 % of these patients, the score is 3 or more, signifying a more advanced state of the disease. The database uses total functional capacity measure for HD subjects. Here, a lower score indicates more advanced functional impairment. The score varies from 1 to 12 for HD subjects under consideration. Here, for almost 50 % of these patients, the score is 5 or less (signifying more severe state of the disease), and, for the rest 50 % patients, the score is more than 5. For the subjects suffering from ALS, the severity measure maintained by the database is the time since the onset of the disease. Here, for almost 80 % of the ALS patients, the severity of the disease is moderate. The subjects were instructed to walk at their normal speed along a 77 m long hallway. To measure gait rhythm and the timing of gait cycle, force-sensitive resistors were placed as insoles in each subject's shoe. The gait time sequences were obtained using these resistive sensors with output approximately proportional to the force under the foot. Stride-to-stride measures of footfall contact times were derived from these signals, and the stride time (i.e., the time from the initial contact of one foot to the subsequent contact of the same foot) along with swing and stance times was determined for each stride. For each subject, stride-to-stride measurements of footfall contact times were acquired for approximately 300 s. In the present study, the time sequences corresponding to the left and right stance intervals, the left and right swing intervals, and the double support interval, each expressed as a percentage of the stride time for each subject, are considered. Figures 12.1, 12.2, 12.3 demonstrate the time sequence plots of the left swing interval, right stance interval, and double support interval for some sample subjects. A close inspection of these plots

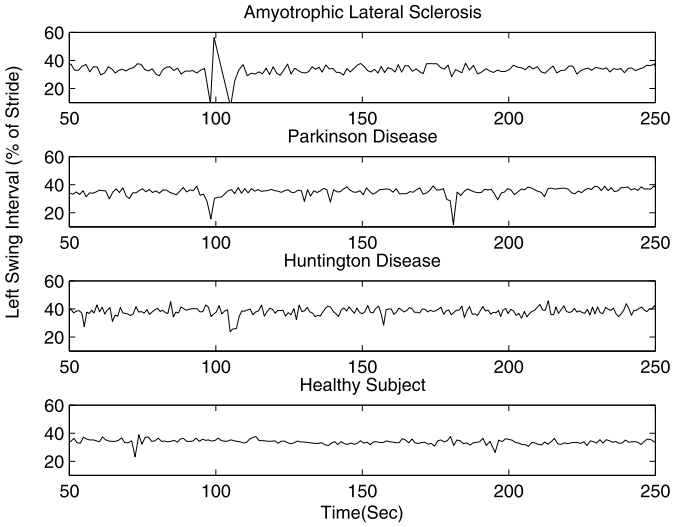


Fig. 12.1 Plot of left-swing interval (% of stride) vs. time for sample healthy and pathological subjects [13]

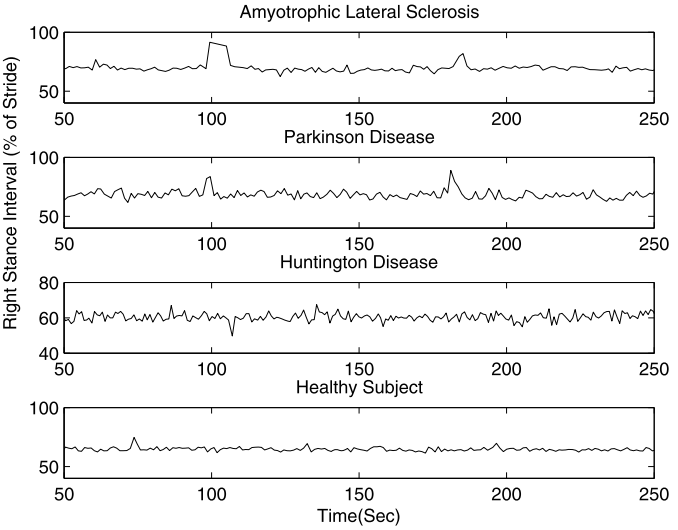


Fig. 12.2 Plot of right-stance interval (% of stride) vs. time for sample healthy and pathological subjects [13]

reveals that it is impossible to differentiate between healthy and pathological subjects without any ambiguity. This indicates the need for an intelligent system that can automatically classify pathological and healthy subjects.

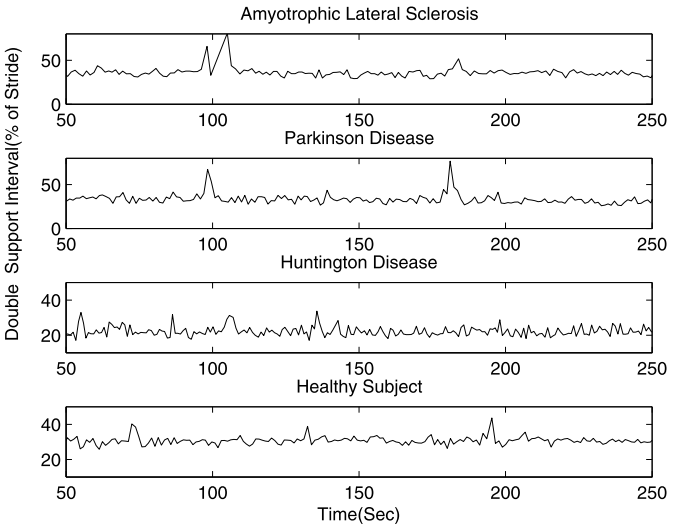


Fig. 12.3 Plot of double-support interval (% of stride) vs. time for sample healthy and pathological subjects [13]

12.3 Cross-Correlation Based Feature Extraction Methodology

Cross-correlation is a mathematical operation that can be suitably utilized to find the extent of similarities between two signals. The cross-correlation technique has been successfully used in many applications like robotics and remote-sensing, sonar and radar systems for range and position detection, recovery of signals buried in noise, signal processing [7, 28, 29], and in several other domains [8, 24, 28, 29, 37, 39]. One of the novelties of the present work lies in applying the cross-correlation technique judiciously, as a feature extraction tool, for the classification of gait signals. The cross-correlation of two finite duration causal sequences $x[k]$ and $y[k]$, each sample having length N, is given by [6, 38, 41]:

$$r_{xy}[m] = \sum_{k=0}^{N-|m|-1} x[k]y[k-m] \tag{12.1}$$

where $m = -(N-1), -(N-2), \ldots, 0, 1, \ldots, (N-2), (N-1)$. The index m represents the time shift parameter, also known as lag, and subscript xy represents sequences being correlated. The length of the cross-correlation sequence $r_{xy}[m]$ is $(2N-1)$ samples. The plot of the cross-correlation function $r_{xy}[m]$ versus m is known as the cross-correlogram. Each $(2N-1)$ length cross-correlation sequence can be fed directly to the classifier to classify bioelectric signals, but it involves massive computational burden. In order to reduce the computational load, one can extract meaningful information either directly from the cross-correlation sequence or from the transformation of the cross-correlation sequence into frequency domain using Fourier transform and utilizing the cross-spectral density information.

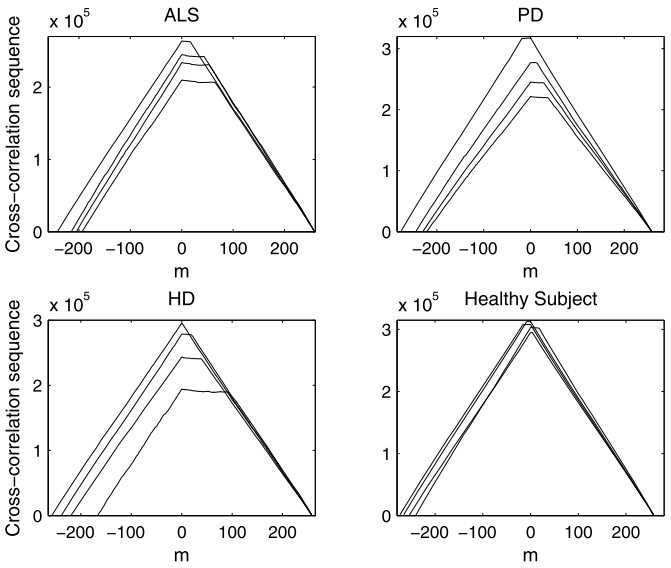


Fig. 12.4 Representative cross-correlograms of the left-stance interval sequences for the subjects belonging to different classes [13]

In this work, one of the healthy subjects from the gait database has been chosen as reference. For time-domain feature based analysis of gait signals, the stance interval (left and right), swing interval (left and right), and double-support interval sequences $y[k]$ of each of the other subjects (belonging to this database) are cross-correlated with the corresponding sequences $x[k]$ of the reference. This yields five cross-correlograms for each subject. However, for frequency domain feature based analysis, only left stance interval sequence has been considered. Some representative sets of cross-correlation sequences of gait signals are depicted in Fig. 12.4.

12.3.1 Time Domain Features

Features extracted directly from cross-correlograms are termed as time-domain features. The three common traits of the cross-correlograms, expressed quantitatively by the centroid (*cent*), the mean-square abscissa (*msa*), and the variance of abscissa (*va*) [6], are found to serve as important parameters to classify bioelectric signals.

They are defined as

$$\text{centroid} = cent = \langle m \rangle = \frac{\sum_{m=-(N-1)}^{N-1} m\, r_{xy}[m]}{\sum_{m=-(N-1)}^{N-1} r_{xy}[m]}, \tag{12.2}$$

$$\text{mean-square abscissa} = msa = \left\langle m^2 \right\rangle = \frac{\sum_{m=-(N-1)}^{N-1} m^2\, r_{xy}[m]}{\sum_{m=-(N-1)}^{N-1} r_{xy}[m]}, \tag{12.3}$$

and

$$\text{variance of abscissa} = va = \left\langle \left(m - \langle m \rangle\right)^2 \right\rangle = \frac{\sum_{m=-(N-1)}^{N-1} (m - \langle m \rangle)^2 r_{xy}[m]}{\sum_{m=-(N-1)}^{N-1} r_{xy}[m]}. \tag{12.4}$$

In the present analysis, above three quantitative time-domain descriptors of the cross-correlograms are used for the classification of gait signals.

12.3.2 Frequency Domain Features

As discussed earlier, one can analyze cross-correlation in frequency domain and extract meaningful features from it by computing Fourier transform of each cross-correlation sequence. The Fourier transform of cross-correlation sequence $r_{xy}[m]$ is called the cross-spectral density (S_{xy}), which is defined as [6, 38, 41]:

$$S_{xy}(f) = \sum_{m=-\infty}^{\infty} r_{xy}[m] e^{-j2\pi fm}. \tag{12.5}$$

The features extracted from the cross-spectral density (S_{xy}) are called frequency-domain features. These features should be ideally well-suited for characterizing a bioelectric signal, but with a reduced dimension. From the cross-spectral density information, one can create the corresponding magnitude and phase cross-spectral density, i.e., $|S_{xy}(f)|$ and $\angle S_{xy}(f)$ feature vectors. Then the features extracted from $|S_{xy}(f)|$ and $\angle S_{xy}(f)$ can be given as:

$$\mathit{fl\_mag}(n) = \left|S_{xy}(f)\right|\Big|_{f=nf_0}, \quad n = 1, 2, 3, \ldots, \tag{12.6}$$

$$\mathit{fl\_phase}(n) = \angle S_{xy}(f)|_{f=nf_0}, \quad n = 1, 2, 3, \ldots, \tag{12.7}$$

$$\begin{aligned} \mathit{fl\_composite} = \big[&\mathit{fl\_mag}(1), \mathit{fl\_mag}(2), \ldots, \mathit{fl\_mag}(n), \ldots, \\ &\mathit{fl\_phase}(1), \mathit{fl\_phase}(2), \ldots, \mathit{fl\_phase}(n), \ldots\big]. \end{aligned} \tag{12.8}$$

Here, $\mathit{fl\_mag}(n)$ denotes the magnitude of the cross-spectral density at the nth frequency sample. Similarly, $\mathit{fl\_phase}(n)$ denotes the phase of the cross-spectral density at the nth frequency sample. Then the composite feature vector *fl_composite* can be formed, considering all *fl_mag* and *fl_phase* coefficients. Figures 12.5 and 12.6 show the plots of the sample $|S_{xy}(f)|$ and $\angle S_{xy}(f)$ curves for the cross-correlation sequences of the left stance interval up to the 30th frequency sample.

12.4 Elman's Recurrent Neural Network Based Classification

Recurrent neural networks (RNNs) are particularly useful for learning both temporal and spatial patterns. As opposed to a multilayer perceptron (MLP), which employs

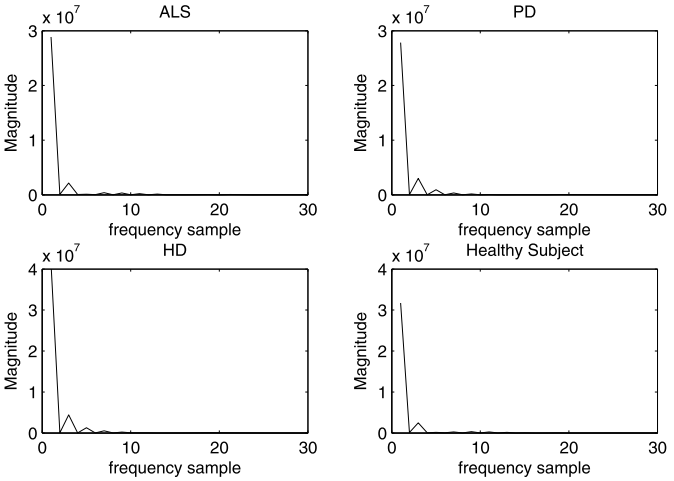


Fig. 12.5 Magnitude cross-spectral density of the left stance interval belonging to healthy and various pathological subjects

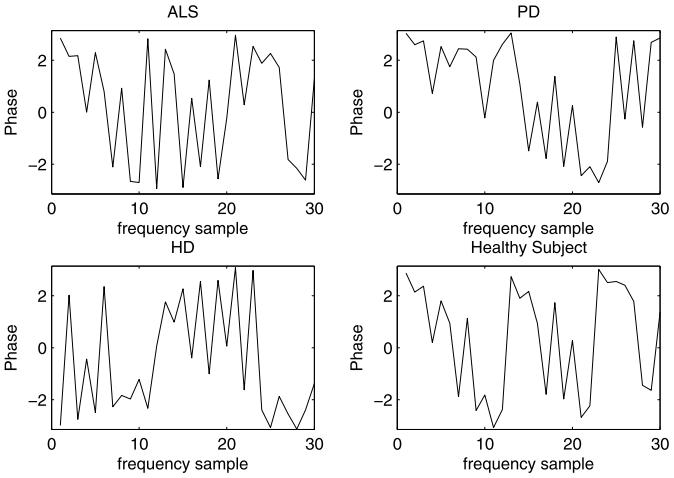


Fig. 12.6 Phase cross-spectral density of the left stance interval belonging to healthy and various pathological subjects

only feedforward connection, RNNs are more complicated as they employ a combination of feedback and feedforward connections, exhibiting the property of memory [4, 18]. RNNs are useful for two types of application: as associated memories and for input–output mapping [12, 40, 44, 48]. In the present application, RNNs have been implemented for input–output mapping. There are several architecture layouts available for different relevant RNNs. Some popular variants of such RNNs include Jordan's network (which employs feedback connection from the output of the output

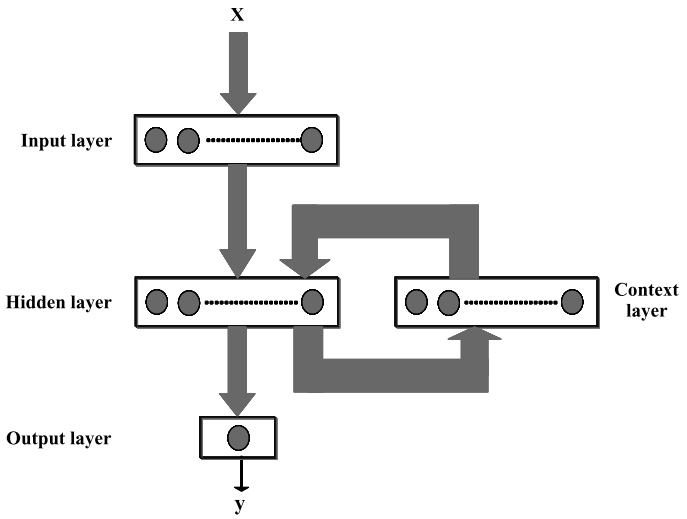


Fig. 12.7 The architecture of the Elman's RNN

layer to the input of the input layer) [21], Elman's network (which employs feedback connection from the output of the hidden layer to the input of the input layer) [26], Pollack sequential cascade network [36], higher order recurrent neural network of Giles [15], Lee and Song's network (in which each output node is connected to itself) [26], etc. In this proposed system, Elman's RNN has been employed, which is a popular RNN in the category of dynamically driven neural networks. Like several other RNNs, Elman's network incorporates static MLP architecture as its basic building block and is trained by some popular learning algorithm, employed for training MLPs. Figure 12.7 shows the generic architecture of an Elman's RNN utilized in this work. This is a three-layer architecture where layer 2 contains context units in addition to the hidden neurons. The context units comprise a bank of unit time delays and they store the outputs of the hidden neurons for one time step, and then these are fed back to the input of the input layer. Hence, the context units depict short-term memory of the RNN. However, as the output of the hidden layer of the RNN, at any time step, is a nonlinear function of both the output of the input layer at that given time step and the output of the hidden layer in the previous time step, the network continues to recycle information over multiple time steps, which is useful for efficient discovery of temporal patterns [18]. Mathematically speaking, the output from the hidden layer, at the kth time step is given as:

$$z_j^2(k) = f_1\left(\sum_{i=1}^{N} z_i^1(k) w_{ij}^{12} + \sum_{j=1}^{P} z_j^2(k-1) c_{jj}^{22} + b_j^2\right) \tag{12.9}$$

where

- $z_j^2(k)$ = output of the jth neuron of layer 2 at the kth time step,

- $z_i^1(k)$ = output of the ith neuron of layer 1 at the kth time step = $x_i(k)$,
- w_{ij}^{12} =weight connecting the ith neuron of layer 1 and the jth neuron of layer 2,
- c_{jj}^{22} = weight connecting the jth neuron of context units and the jth hidden layer neuron of layer 2,
- $z_j^2(k-1)$ = output of the jth neuron of layer 2, delayed by one time step,
- b_j^2 = bias associated with the jth neuron of layer 2,
- N = number of inputs,
- P = number of hidden layer neurons.

Hence the output of the Elman's RNN can be given as:

$$z^3(k) = f_2\left(\sum_{j=1}^{P} z_j^2(k) w_j^{23}\right) \tag{12.10}$$

where

- $z^3(k)$ = output of the only neuron in output layer, at time step k,
- w_j^{23} = weight connecting the jth neuron of layer 2 to the only neuron in layer 3,
- $f_1(\bullet)$ represents a nonlinear function, usually chosen as tansigmoidal or logsigmoidal function,
- $f_2(\bullet)$ can be either a linear or a nonlinear mapping.

A generalized Elman's RNN can employ multiple neurons in the output layer also. In the training phase, for a multiclass problem, the output, for each exemplar input to an ERNN, is chosen for the system as $y \in \{1, 2, \ldots, c, \ldots, C\}$. Here C is the total number of classes in which each RNN is designated to classify its inputs. In the implementation phase, the output of the ERNN is classified as:

$$y_{\text{class}} = c \quad \text{if } (c - 0.5) < y \leq (c + 0.5) \tag{12.11}$$

except for the two terminal classes where $y_{\text{class}} = 1$ if $y < 1.5$ and $y_{\text{class}} = C$ if $y > (C - 0.5)$.

12.5 Time Domain Cross-Correlation Based Scheme for Gait Signal Classification

For the classification of gait signals using time-domain features, benchmark signals available from the physionet database [35] have been utilized. As mentioned earlier, the database contains real-life gait signals of 16 healthy subjects, as well as 15, 19, and 13 pathological subjects having neurological disorders due to PD, HD, and ALS, respectively. The procedure for obtaining the cross-correlograms has already been explained. From the cross-correlation sequences, three quantitative descriptors [13], namely, the centroid (*cent*), the mean-square abscissa (*msa*), and the variance of abscissa (*va*), are evaluated for several subjects with known neurological states of health and these values are subsequently used to train ERNNs. Once

Table 12.1 The range or universe of discourse for the selected features

Features	Range or discourse
cent_l_st	[−60, 35]
msa_l_st	[8524, 14100]
va_l_st	[6130, 12878]
cent_r_st	[−59, 35]
msa_r_st	[8500, 14500]
va_r_st	[6122, 12900]
cent_l_sw	[−60, 35]
msa_l_sw	[8442, 14000]
va_r_sw	[6052,12829]
cent_r_sw	[−60, 35]
msa_r_sw	[8440, 14040]
va_r_sw	[6050, 12830]
cent_ds	[−59, 35]
msa_ds	[7574, 13920]
va_ds	[5990, 12745]

this process is complete, it is expected that, for a new subject, if the above quantities are calculated and fed to the ERNNs, the system can determine whether the subject is healthy or not, and also the type of illness, where the subject is found to be ill. The three features extracted from the left stance interval sequence of a subject are named as *cent_l_st*, *msa_l_st*, and *va_l_st*. Similarly, the three features extracted from the right stance interval sequence are named as *cent_r_st*, *msa_r_st*, and *va_l_st*, the three features extracted from the left swing interval sequence are named as *cent_l_sw*, *msa_l_sw*, and *va_l_sw*, and the three features extracted from the right swing interval sequence are named as *cent_r_sw*, *msa_r_sw*, and *va_r_sw*. The three features extracted from the double support sequence are named as *cent_ds*, *msa_ds*, and *va_ds*. Hence, for each subject under consideration, 15 features are extracted from five cross-correlograms. Table 12.1 lists these features, used as the inputs of the ERNN, with their range of values, obtained for the specific problem under consideration.

The system is configured as a four-class classification system (i.e., $C = 4$) where the four classes correspond to healthy subjects, pathological subjects suffering from PD, pathological subjects suffering from HD, and those suffering from ALS. Two schemes are discussed in this chapter for solving the composite problem, utilizing time-domain features, where each scheme utilizes more than one ERNN in modular form [13]. Each modular ERNN is designed to solve a sub-problem, and these ERNNs are arranged in a hierarchical fashion where the output of one ERNN determines whether another (or more than one) ERNN should be activated or not. Each ERNN is activated as a 15-input–1-output system where the 15 inputs are determined from the features extracted from cross-correlograms, as discussed in Section 12.4. For Scheme 1, ERNN1 is trained to solve a binary classification prob-

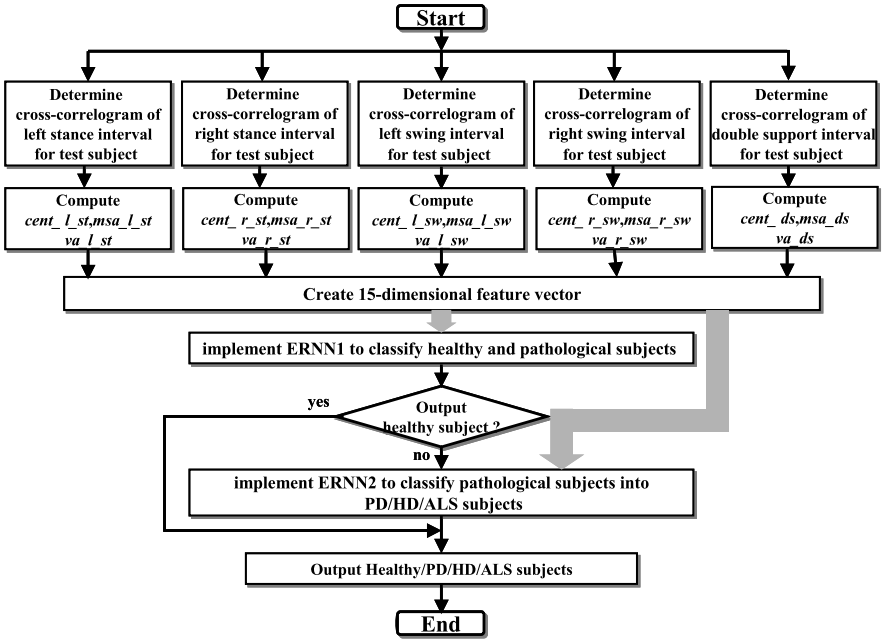


Fig. 12.8 Flowchart representation of scheme [13]

lem where we determine whether the subject tested is healthy/pathological. If the subject is diagnosed as pathological, then ERNN2 is activated with the same set of feature vectors, determined for that specific subject. ERNN2 is specifically trained with the training data set determined from pathological subjects only, and it is designed to solve a three-class problem, segregating pathological subjects into PD, HD, and ALS classes. Figure 12.8 shows Scheme 1 in a flowchart form. Finally, outputs from both ERNN1 and ERNN2 are utilized to suggest the ultimate diagnosis which classifies the subject under consideration into one among the four classes, i.e., healthy/PD/HD/ALS.

The same problem can also be solved by employing Scheme 2, shown in flowchart form in Fig. 12.9. The feature extraction part remains identical with that of Scheme 1, but the classification module now employs three modular ERNNs, namely ERNN1, ERNN3, and ERNN4, each trained to perform specific binary classification jobs. ERNN1 is implemented in an identical manner with that of Scheme 1. But if it diagnoses the subject as pathological, then ERNN3 is activated to determine whether the pathological subject is suffering from ALS or not. If the answer is negative, then ERNN4 is activated to determine whether the subject is suffering from PD or HD. The final outcome of the automated tool discussed in Scheme 2 is determined by considering outputs from all three ERNNs.

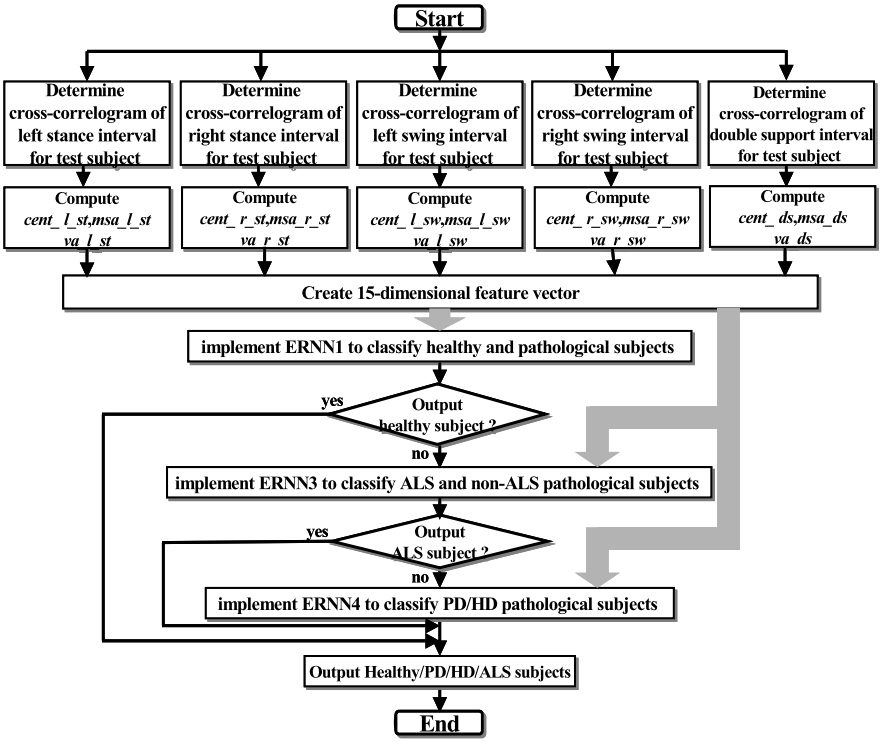


Fig. 12.9 Flowchart representation of scheme [13]

12.5.1 Performance Evaluation

The performances of the presented schemes have been tested with the signals available from [35]. As discussed earlier, five time-domain gait signals have been considered for each subject. By constructing the cross-correlation sequences with reference to the corresponding signals acquired from the reference subject, a total of 15 features for each subject (extracting three features from each of the five cross-correlograms) were determined. Table 12.1 shows the universe of discourse for all these features extracted from the entire signal database. Once the feature extraction phase is over, each of the four modular ERNNs is trained based on its corresponding training dataset. The composite training and testing datasets are created by putting 50 % data in each dataset. ERNN1 is trained utilizing the entire training dataset, ERNN2 and ERNN3 are trained utilizing those exemplars in the training dataset that belong to pathological subjects, and ERNN4 is trained utilizing those exemplars in the training dataset that belong to PD or HD diseases. On successful completion of training, each modular ERNN is tested independently. ERNN1 is tested utilizing the entire testing dataset, each of ERNN2 and ERNN3 is tested using those cases in the testing dataset that belong to pathological subjects, and ERNN4 is tested utilizing

Table 12.2 Classification accuracy of ERNN1

Algorithm	Healthy subject			Pathological subject			Overall		
	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)
BPNN	61.5	9.9	50.0–75.0	88.4	2.8	82.6–91.3	81.2	2.7	77.4–83.9
LVQ	36.6	12.9	14.3–62.5	75.1	5.5	60.9–87.5	65.9	2.5	58.1–71.0
LS-SVM	56.2	13.6	37.5–100.0	87.6	11.9	69.6–100.0	80.1	11.8	61.3–100.0
ERNN1	90	5.1	87.5–100.0	90.9	3.1	87.1–93.6	90.6	1.8	87.1–93.6

only PD and HD subjects from the testing database. Tables 12.2, 12.3, 12.4, 12.5 present those performance results for each ERNN separately. Here, each ERNN was run 20 times and the classification results in percentages are mentioned in each case with the corresponding mean, standard deviation, and range of results obtained employing 20 such runs for each ERNN. Each system was also implemented in accordance with N-fold cross-validation theory with $N = 2$, i.e., the whole set of experiments was carried out a second time by swapping the composite training and testing datasets and the result of a particular run is considered as the mean of the two runs conducted by swapping training and testing data sets. From Tables 12.2 to 12.5, it can be seen that whenever the cross-correlation aided modular ERNNs were put into operation for binary classification purposes, they could comfortably attain 90.0 % or higher classification accuracy. ERNN1 reported a mean accuracy of 90.6 %, ERNN3 reported the corresponding result as high as 97.8 %, and ERNN4 reported the mean accuracy of 94.1 %. For ERNN2, which employed a three-class classification algorithm, accuracy was understandably a little lower, i.e., 89.8 %. The performances of these ERNNs are compared with backpropagation neural network (BPNN) [42], Learning Vector Quantization (LVQ) [23], and Least Square Support Vector Machine (LS-SVM) [22, 45] based classification algorithms. Computation using BPNN, LVQ, and LS-SVM were preceded by the identical cross-correlation based feature extraction procedure described earlier, and hence it was implemented utilizing 15-dimensional feature vectors. BPNN, LVQ, and LS-SVM, like ERNNs, also employ supervised learning based training procedures. They are also employed in an identical manner as ERNNs. Hence all of the BPNN, LVQ, and LS-SVM performances were reported on the basis of 20 runs, with their corresponding mean, standard deviation, and range values. In each case, it can be easily seen that the performance of ERNN based systems is much superior compared to BPNN, LVQ, and LS-SVM based systems. BPNN, LVQ, and LS-SVM based supervised learning procedures produced much inferior results with mean classification accuracy in the range 52.1–81.2 %, 36.7–78.8 %, and 32.2–80.1 %, respectively.

Once these encouraging results were obtained with each modular ERNN, the composite hierarchical schemes, Scheme 1 and Scheme 2, have been employed to obtain the automated gait identification tools as four-class classification systems. Table 12.6 reports these results which show that Scheme 1 could produce 83.8 % and Scheme 2 could produce 87.1 % overall accuracy. These results of Scheme 1

Table 12.3 Classification accuracy of ERNN2

Algorithm	ALS			PD			HD			Overall		
	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)
BPNN	25	8.7	16.7–33.0	63.8	3.9	62.5–75.0	60	5.8	55.5–66.7	52.1	2.9	47.8–56.5
LVQ	37.5	9.5	28.6–50.0	12.9	14.6	0.0–42.9	56.1	19.2	33.3–88.9	36.7	6.1	30.4–47.8
LS-SVM	45.6	23.5	14.3–100.0	0.7	3.2	0.0–14.3	47.8	23.1	11.1–88.9	32.2	6.5	21.7–43.5
ERNN2	89.1	8.2	83.3–100.0	80	7.5	75.0–100.0	97.2	4.9	88.9–100	89.8	3.2	87.0–100.0

Table 12.4 Classification accuracy of ERNN3

Algorithm	ALS			Non-ALS			Overall		
	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)
BPNN	50	0	50.0–50.0	83.5	5.4	76.5–88.7	74.8	4	69.6–78.3
LVQ	39.3	11	28.6–50.0	94.1	0	94.1–94.1	78.8	3.9	75.0–82.6
LS-SVM	24.7	15.6	14.3–50.0	92.9	3.1	82.4–94.1	73.9	6.1	62.5–82.6
ERNN3	91.7	8.6	83.3–100.0	99.4	1.8	94.1–100	97.8	2.6	91.3–100.0

Table 12.5 Classification accuracy of ERNN4

Algorithm	PD			HD			Overall		
	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)
BPNN	51.3	13.7	37.5–75.0	68.9	6.8	56.6–77.7	60.6	5.6	53.0–70.6
LVQ	21.4	22	0.0–42.9	75	25.7	50.0–100.0	50	3	47.1–52.9
LS-SVM	41.3	14.9	25.0–62.5	47	8.1	22.2–66.7	46	5.2	35.3–52.9
ERNN4	87.5	4.1	75.0–100	99.4	2.5	88.9–100.0	93.8	1.3	88.2–94.1

Table 12.6 Composite classification accuracy for Scheme 1 and Scheme 2

Scheme	ALS (%)	PD (%)	HD (%)	Healthy (%)	Overall (%)
Scheme 1	83.3	87.5	77.8	87.5	83.9
Scheme 2	83.3	87.5	88.9	87.5	87.1

Table 12.7 Confusion matrix for Scheme 1

	Predicted Class			
Actual Class	ALS	PD	HD	Healthy
ALS	5	0	1	0
PD	0	7	1	0
HD	0	1	7	1
Healthy	0	0	1	7

and Scheme 2 are reported with the best performing modular ERNNs, taken as their building blocks. These accuracies are a little less than individual accuracies of modular ERNNs, when implemented in stand-alone form. This is understandable, as each composite scheme employs two or three modular ERNNs in hierarchical form. Tables 12.7 and 12.8 present the confusion matrices corresponding to Scheme 1 and Scheme 2 results, respectively, presented in Table 12.6.

Table 12.8 Confusion matrix for Scheme 2

	Predicted Class			
Actual Class	ALS	PD	HD	Healthy
ALS	5	1	0	0
PD	1	7	0	0
HD	0	0	8	1
Healthy	0	0	1	7

To have a realistic understanding of how strong or weak the presented schemes are, these results are compared with some of the other results reported utilizing gait signals. The presented schemes are based on signals acquired from 63 subjects, and the results are reported on the basis of 62 subjects (with one subject taken as the reference). In [3], an SVM based procedure could solve the binary classification problem into young/elderly gaits, utilizing 24 statistical features extracted from minimum foot clearance (MFC) data of 58 patients, with a mean classification accuracy of 83.3 %. Compared to this scheme, each of the binary-classification modular ERNN systems mentioned here could comfortably produce more than 90 % accuracy and with fewer input features (i.e., 15 chosen in our works). In [3], even after introduction of a hill-climbing algorithm for relevant feature selection (which introduces significant additional computational burden), the binary classification result could not improve to more than 90 %. In [1], another neural network based gait classification scheme was proposed using features from hip–knee joint angle measures. The problem was configured as a three-class classification problem, and utilized data from 8 subjects only. This scheme also reported a classification ratio of 83.3 % only. In the light of these discussions, our modular ERNNs reporting 90.3–98.5 % classification accuracy for binary classification jobs, 87.0 % accuracy for three-class classification jobs, and an accuracy as high as 87.1 % for the composite scheme (considering the complete problem as a four-class classification problem) should be considered as very promising and encouraging solutions for analyzing gait signals to segregate healthy subjects from pathological subjects and to identify the source of neurological disorder in pathological subjects [13].

12.6 Frequency Domain Cross-Correlation Based Scheme for Gait Signal Classification

In the frequency-domain based methodology, gait signals are automatically classified into healthy and pathological subjects. For the classification of gait signals, one of the healthy subjects is chosen as reference. For each of the other subjects, the left stance interval sequence was cross-correlated with the corresponding sequence of the reference subject. As explained earlier, a cross-correlogram was thus obtained for each subject. Representative sets of cross-correlograms of healthy subjects, pathological subjects suffering from ALS, PD, and HD neurological disorders

are shown in Fig. 12.4. Each cross-correlogram is then transformed to the frequency domain using Fourier transform to obtain its magnitude and phase cross-spectral densities, as shown in Fig. 12.5 and Fig. 12.6, respectively.

From magnitude and phase cross-spectral density curves, five different feature set vectors of size 20, 30, 40, 50, and 60, considering the magnitude and phase quantities up to $n = 10$, 15, 20, 25, and 30 frequency samples, have been created. These feature vectors were utilized to train separate ERNN based classifiers, and each trained classifier was subsequently tested. Here ERNNs are trained to solve a binary classification problem to determine whether the subject tested is healthy or pathological. In this scheme, for each classifier developed, the number of hidden layer neurons was set equal to the number of features to be examined. This means, for feature vectors of size $n = 20$, 30, 40, 50 and 60, the corresponding classifier was developed using 20, 30, 40, 50, and 60 neurons, respectively.

12.6.1 Performance Evaluation

Like time domain based classification of gait signals, the classification performance of the ERNN for the frequency domain based classification was also evaluated using signals available from physionet database [35]. As discussed earlier, the left stance interval sequence of reference subject is cross-correlated with the corresponding sequence of other subjects. This yields a cross-correlogram for each subject. The cross-correlogram is then transformed to the frequency domain using Fourier transform to obtain its magnitude and phase cross-spectral densities. Hence five different size (i.e., 20, 30, 40, 50 and 60) feature set vectors, considering the magnitude and phase quantities up to $n = 10$, 15, 20, 25, and 30 harmonics, have been created. Once the feature extraction phase is over, ERNNs are trained based on training datasets. The training and testing datasets are created by putting 50 % data in each dataset. On successful completion of training, ERNNs are tested independently. ERNNs are tested utilizing the testing dataset. Table 12.9 presents the performance of ERNN for different feature sets. Here each ERNN was run 20 times and the classification results in percentages are mentioned, in each case, with the corresponding mean, standard deviation, and range of results obtained employing 20 such runs for each ERNN. Each system was also implemented in accordance with the N-fold cross-validation theory with $N = 2$.

From Table 12.9, it can be seen that the ERNN reported overall mean classification accuracy in the range from 63.6 to 81.6 % for five feature sets, and the best classification results were obtained with 20 feature set vectors. The performances of these ERNNs are then compared with BPNN, LVQ, and LS-SVM based classification algorithms. BPNN, LVQ, and LS-SVM are also employed in an identical manner as ERNNs. Hence each BPNN result is reported on the basis of 20 runs, with their corresponding mean, standard deviation, and range values. It can be seen from Table 12.9 that out of the five cases, LS-SVM produced best results in four cases compared to BPNN, LVQ, and ERNN based classification algorithms. However, the best performance of 81.6 % mean accuracy was obtained using

Table 12.9 Classification performance of ERNN for frequency domain classification of gait signals

Feature Set	Algorithm	Healthy subject			Pathological Subject			Overall		
		Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)	Mean (%)	Std.dev. (%)	Range (%)
20	BPNN	59.8	23.8	12.5–87.5	75.3	7.5	65.2–87.5	71.6	7.2	61.3–83.9
	LVQ	28.6	29.3	0.0–57.1	89.6	10.7	79.2–100	74.2	0	74.2–74.2
	LS-SVM	17.1	19.5	0.0–42.9	95.8	4.7	87.5–100	76.5	2.9	74.2–80.7
	ERNN	64.8	16.1	42.9–100	86.9	10.3	73.9–100	**81.6**	7.1	71.0–93.6
30	BPNN	73.4	10.4	57.1–87.5	79.6	8.8	65.2–95.7	78.1	5.5	71.0–87.1
	LVQ	31.4	32.6	0.0–71.4	89.6	10.7	79.2–100	74.8	1.3	74.2–77.4
	LS-SVM	20.2	23.8	0.0–62.5	100	0	100–100	**80.6**	6.1	74.2–90.3
	ERNN	54.3	16.8	28.6–87.5	79.4	9.8	65.2–95.7	73.6	7.1	64.5–87.1
40	BPNN	60.4	20	28.6–100	69.4	9.2	60.9–87.0	67.1	10.8	54.8–87.1
	LVQ	32.9	34.1	0.0–71.4	89.6	10.7	79.2–100	75.2	1.5	74.2–77.4
	LS-SVM	8.4	11.7	0.0–28.6	100	0	100–100	**77.7**	3.8	74.2–83.9
	ERNN	53.6	18.7	25.0–85.7	75.3	9	60.9–87.5	70	8.8	54.8–83.9
50	BPNN	60.6	20.4	25.0–87.5	65.6	12.1	45.8–87.0	64.4	9.4	45.2–83.9
	LVQ	28.6	29.3	0.0–57.1	89.6	10.7	79.2–100	74.2	0	74.2–74.2
	LS-SVM	0	0	0.0–0.0	100	0	100–100	**75.8**	1.7	74.2–77.4
	ERNN	66.6	13.2	42.9–85.7	72.8	7.8	60.9–91.3	71.3	6.4	61.3–83.9
60	BPNN	74.6	13.1	57.1–100	66.5	8.8	43.5–78.3	66.8	9.4	35.5–77.4
	LVQ	32.9	34.1	0.0–71.4	89.6	10.7	79.2–100	75.2	1.5	74.2–77.4
	LS-SVM	13	25	0.0–75.0	97	8	73.9–100	**76.8**	3.9	74.2–87.1
	ERNN	67.7	15.3	28.6–85.7	62.1	8.8	52.2–75.0	63.6	6.3	58.1–74.2

ERNN, employing 20 feature vector sets. As far as a comparison of development of such ERNN based gait classification schemes utilizing time-domain and frequency-domain based cross-correlation features is concerned, a detailed comparison of Tables 12.6 and 12.9 reveals that both Scheme 1 and Scheme 2 for time-domain feature based systems produced better overall classification accuracy compared to the frequency-domain feature based systems.

12.7 Conclusions

In this chapter, an attempt has been made to develop robust algorithms for automatic classification of gait signals. Cross-correlation has been utilized as an efficient feature extraction tool and Elman's recurrent neural network has been employed as an

automated gait signal pattern classifier utilizing these extracted features. Classification methodologies and their performance evaluations, both involving time and frequency domain based cross-correlation features, were discussed in Sections 12.5 and 12.6.

In this work, a detailed study pertaining to gait signal classification was presented. A systematic method of choosing features utilizing both time-domain and frequency-domain cross-correlation information was described, coupled with different variants of ERNNs and relative comparisons of their performances. These performance analyses were carried out utilizing well known quantitative measures or performance indices, popularly employed for evaluating similar systems. In this context, two time-domain correlation based gait identification schemes have been presented in Section 12.6, utilizing several modular ERNNs in hierarchical form. Each of these schemes has been successfully implemented as a multiclass classification tool where one can segregate the input gait signals of healthy subjects and those of pathological subjects suffering from specific neurological disorders, e.g., Parkinson's disease (PD), Huntington's disease (HD), and Amyotrophic Lateral Sclerosis (ALS). The performances of the presented schemes have been evaluated by considering some benchmark signals, and very encouraging results have been reported, compared to other contemporary algorithms available in practice. The presented methods show how modular recurrent neural networks can be effectively utilized for the specific problem under consideration. This work is expected to encourage future researchers to make an in-depth study of the feasibility of implementing several candidate RNN algorithms available in the literature (e.g., Jordan's network, Pollack's network, FIR networks, Laguerre models, etc.) for classification of bioelectric signals and to compare their performances.

References

1. Barton, J.G., Lees, A.: An application of neural networks for distinguishing gait patterns on the basis of hip-knee joint angle diagrams. Gait Posture **5**, 28–33 (1997)
2. Beers, H.E.: Gait Disorders. The Merck Manual of Geriatrics, 3rd edn. Merck Res. Lab, Rahway (2001). Chap. 21
3. Begg, R.K., Palaniswami, M., Owen, B.: Support vector machines for automated gait classification. IEEE Trans. Biomed. Eng. **52**(5), 828–838 (2005)
4. Bose, N.K., Liang, P.: Neural Network Fundamentals with Graphs, Algorithms and Applications. Tata McGraw-Hill Publishing Company limited, New Delhi (1998)
5. Boulgouris, N.V., Hatzinakos, D., Plataniotis, K.N.: Gait recognition: a challenging signal processing technology for biometric identification. IEEE Signal Process. Mag., 78–90 (2005)
6. Bracewell, R.N.: The Fourier Transform and Its Applications, 3rd edn. McGraw-Hill, New York (2000)
7. Chandaka, S., Chatterjee, A., Munshi, S.: Cross-correlation aided support vector machine classifier for classification of EEG signals. Expert Syst. Appl. **36**, 1329–1336 (2009)
8. Chandaka, S., Chatterjee, A., Munshi, S.: Support vector machines employing cross-correlation for emotional speech recognition. Measurement **42**(4), 611–618 (2009)
9. Cunado, D., Nixon, M.S., Carter, J.N.: Using gait as a biometric, via phase-weighted magnitude spectra. In: Proc. Int. Conf. Audio- and Video-Based Biometric Person Authentication, Crans-Montana, Switzerland. LNCS, vol. 1206, pp. 95–102 (1997)

10. Cutting, J., Kozlowski, L.: Recognizing friends by their walk: gait perception without familiarity cues. Bull. Psychon. Soc. **9**(5), 353–356 (1977)
11. Davis, R.B.: Clinical gait analysis. IEEE Eng. Med. Biol. Mag., 35–40 (1988)
12. Delgado, M., Pegalajar, M., Cuéllar, M.: Mimetic evolutionary training for recurrent neural networks: an application to time-series prediction. Expert Syst. **23**(2), 99–115 (2006)
13. Dutta, S., Chatterjee, A., Munshi, S.: An automated hierarchical gait pattern identification tool employing cross-correlation-based feature extraction and recurrent neural network based classification. Expert Syst. **26**(2), 202–217 (2009)
14. Fildes, B.: Injuries Among Older People: Falls at Home and Pedestrian Accidents. Dove Publications, Melbourne (1994)
15. Giles, C., Sun, G., Chen, H., Lee, Y., Chen, D.: Higher order recurrent network and grammatical inference. Adv. Neural Inf. Process. Syst. **2**, 380–387 (1990)
16. Hausdorff, J.M., Cudkowicz, M.E., Firtion, R., Wei, J.Y., Goldberger, L.A.: Gait variability and basal ganglia disorders: stride-to-stride variations of gait cycle timing in Parkinson's disease and Huntington's disease. Mov. Disord. **13**(3), 428–437 (1998)
17. Hausdorff, J.M., Lertratanakul, A., Cudkowicz, M.E., Peterson, A.L., Kaliton, D., Goldberger, A.L.: Dynamic markers of altered gait rhythm in amyotrophic lateral sclerosis. J. Appl. Physiol. **88**(6), 2045–2053 (2000)
18. Haykin, S.: Neural Networks: A Comprehensive Foundation, 6th Indian Reprint edn. Pearson Education, Upper Saddle River (2004)
19. Huang, P.S., Harris, C.J., Nixon, M.S.: Visual surveillance and tracking of humans by face and gait recognition. In: Proc. 7th IFAC Symp. Artificial Intelligence in Real-Time Control, Grand Canyon National Park, AZ, pp. 43–44 (1998)
20. Johansson, G.: Visual perception of biological motion and a model for its analysis. Percept. Psychophys. **14**(2), 201–211 (1973)
21. Jordon, M.: Serial order: a parallel distributed processing approach. Inst. Cognitive Sci. ICS Rep. 8604, University of California, San Diego (1986)
22. Kecman, V.: Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models. MIT Press, Cambridge (2002)
23. Kohonen, T.: The self-organizing map. Proc. IEEE **78**, 1464–1480 (1990)
24. Kuppusamy, K., Lin, W., Haacke, E.M.: Statistical assessment of cross-correlation and variance methods and the importance of electrocardiogram gating in functional magnetic resonance imaging. Magn. Reson. Imaging **15**(2), 169–181 (1997)
25. Lai, D.T.H., Begg, R.K., Palaniswami, M.: Computational intelligence in gait research: a perspective on current applications and future challenges. IEEE Trans. Inf. Technol. Biomed. **13**(5), 682–702 (2009)
26. Lee, S.W., Song, H.H.: A new recurrent neural network architecture for visual pattern recognition. IEEE Trans. Neural Netw. **8**(2), 331–340 (1997)
27. Littlem, J., Boyd, J.: Recognizing people by their gait: the shape of motion. Int. J. Comput. Vis. **14**(6), 83–105 (1998)
28. Mizuno-Matsu, Y., Motamedi, M.K., Webber, W.R., Lesser, R.P.: Wavelet-crosscorrelation analysis can help predict whether bursts of pulse stimulation will terminate after discharges. Clin. Neurophysiol. **113**(1), 33–42 (2002)
29. Mizuno-Matsumoto, Y., Motamedi, G.K., Webber, W.R.S., Ishii, R., Ukai, S., Kaishima, T., Shinosaki, K., Lesser, R.: Wavelet-crosscorrelation analysis of electrocorticography recordings from epilepsy. In: International Congress Series, vol. 1278, pp. 411–414. Elsevier Science, Amsterdam (2005)
30. Murase, H., Sakai, R.: Moving object recognition in eigenspace representation: gait analysis and lip reading. Pattern Recognit. Lett. **17**(2), 155–162 (1996)
31. Murphy, K.: Medical and functional status of adults with cerebral palsy. IEEE Trans. Inf. Theory **14**(5), 734–743 (1968)
32. Nigg, B.M., Fisher, V., Ronsky, J.L.: Gait characteristics as a function of age and gender. Gait Posture **2**, 213–220 (1994)

33. Niyogi, S.A., Adelson, E.H.: Analyzing and recognizing walking figures in xyt. In: Proc. Computer Vision and Pattern Recognition, Seattle, WA, vol. 2, pp. 469–474 (1994)
34. Ostrosky, K.M., VanSwearingen, J.M., Burdett, R.G., Gee, Z.: A comparison of gait characteristics in young and old subjects. Phys. Ther. **74**, 637–646 (1994)
35. Physionet database. http://physionet.fri.uni-lj.si/physiobank/database/gaitndd/
36. Pollack, J.B.: The induction of dynamical recognizers. Mach. Learn. **7**, 227–252 (1991)
37. Roberts, M.: Signals and Systems—Analysis Using Transform Methods and MATLAB. Tata –McGraw-Hill Publishing Company limited, New Delhi (2003)
38. Signal Processing Toolbox for Use with MATLAB, User Guide, 2nd edn. Pearson Education (2001)
39. Suljagic, S., Rajsic, N., Ivanus, J., Bozovic, Z., Kalauzi, A., Rapajic, D., Nedovic, G.: P197 the predictive role of t-histograms of crosscorrelation r-coefficients in the analysis of ictal EEG activity. Electroencephalogr. Clin. Neurophysiol. **99**(4), 320 (1996)
40. Süt, N., Şenocak, M.: Assessment of the performances of multilayer perceptron neural networks in comparison with recurrent neural networks and two statistical methods for diagnosing coronary artery disease. Expert Syst. **24**(3), 131–142 (2007)
41. The Mathworks, Natwick, MA: Signal Processing Toolbox for Use with MATLAB, User Guide (2001)
42. The Mathworks, Natwick, MA: Neural Network Toolbox for Use with MATLAB (2002)
43. Therapy and equipment needs of people with cerebral palsy and like disabilities in Australia (disability series). Tech. rep., Australian Institute of Health and Welfare, Canberra, A.C.T., Australia (2006)
44. Übeyli, E.: Comparison of different classification algorithms in clinical decision-making. Expert Syst. **24**(1), 17–31 (2007)
45. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
46. Wilson, R.S., Schneider, J.A., Beckett, L.A., Evans, D.A., Bennett, D.A.: Progression of gait disorder and rigidity and risk of death in older persons. Neurology **58**(12), 1815–1819 (2002)
47. Winter, D.: The Biomechanics and Motor Control of Human Gait: Normal, Elderly, and Pathological. Univ. Waterloo Press, Waterloo (1991)
48. Zhou, S., Xu, L.: Dynamic recurrent neural networks for a hybrid intelligent decision support system for the metallurgical industry. Expert Syst. **16**(4), 240–247 (1999)

Chapter 13
Image Denoising Using Wavelets: Application in Medical Imaging

Abdeldjalil Ouahabi

Abstract Medical images obtained from MRI are the most common tool for diagnosis in Medicine. These images are often affected by random noise arising in the image acquisition process. Hence, noise removal is essential in medical imaging applications in order to enhance and recover fine details that may be hidden in the data.

A common approach for image denoising is to convert a noisy image into a transform domain such as the wavelet and contourlet domain, and then compare the transform coefficients with a fixed or adapted threshold. The underlying idea is that the useful signal can be described by a small number of coefficients of high-amplitude wavelets, and that the noise is spread across all coefficients. In fact, the wavelet representation naturally compresses the essential information in a signal into relatively few, large coefficients, which represent image details at different resolution scales.

In this chapter, we review recent wavelet denoising techniques for medical ultrasound and for magnetic resonance images and discuss their performances in terms of SNR (or PSNR) and visual aspects of image quality. However, image denoising using wavelet-based multiresolution analysis requires a delicate compromise between noise reduction and preserving significant image details. Hence, in practical applications, we will often simplify the theory using heuristics, when this leads to algorithms with lower complexity or higher flexibility.

13.1 Introduction to Multiresolution Analysis

This introductory section recalls the theory of wavelets and multiresolution analysis based on the discrete wavelet transform. The Mallat algorithm for 1D and 2D signals as a tool for concrete implementation of these concepts is shown at the end of this section.

A. Ouahabi (✉)
Polytech Tours, Tours University, 7 Avenue Marcel Dassault, 37200 Tours, France
e-mail: ouahabi@univ-tours.fr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_13,

13.1.1 Discovery and Contributions of Wavelets

The astonishing discovery of wavelets in 1975 by Jean Morlet [1] has not only been exploited commercially, but has also been royally ignored by the petrol company Elf Aquitaine, now called Total, Morlet's former employer. A likely explanation is that this then-public company has been traumatized by the scam of the century: the infamous "sniffer planes", supposedly able to miraculously "sniff out" the presence of petrol!

This politico-financial case has cost the taxpayers over a billion, and 100 millions of Francs between 1975 and 1979, and also cost the job of the polytechnical engineer Jean Morlet. Indeed, the chief casualty of this debacle is without doubt the father of wavelets who, by way of thanks, was forced into early retirement by Elf.

In practice, on the one hand, the field has had to wait for the work of Mallat [2, 3], and Daubechies [4, 5] which focused on implementation adapted to the pyramidal algorithms of Burt and Adelson [6] for a concrete exploitation of these wavelets to be born, thanks to the fast wavelet transform. On the other hand, multiresolution analysis was also contingent of the benefits of subband coding, introduced in 1977 by Esteban and Galand [7].

13.1.2 Continuous Wavelet Transforms

The transient universe is more complex but much more exciting than the peaceful garden of the stationary world.

The analysis of transients of different durations requires a transform capable of acting simultaneously on a range of temporal resolutions: wavelet transforms perform this function by decomposing a signal via a family of translated and dilated wavelets.

Called a wavelet (or mother wavelet), a finite energy function[1] ψ contains n vanishing moments (where $n \in \mathbb{N}$), that is, satisfies

$$\int_{\mathbb{R}} t^p \psi(t)\, dt = 0 \quad \forall 0 \leq p < n. \tag{13.1}$$

The relationship (13.1) indicates that wavelet ψ analyzes a signal with the following qualities:

- Oscillation (by taking positive and negative values), that is, the number n controls the oscillations of ψ; the larger the n, the more ψ oscillates,
- A zero mean (for $p = 0$),
- By disregarding the continuous component, for $p = 0$, and in general ($p > 0$) by being orthogonal to polynomial components of degrees less than n. The wavelet "kills" polynomials [8].

[1] $L^2(\mathbb{R})$ represents the set of finite energy signals.

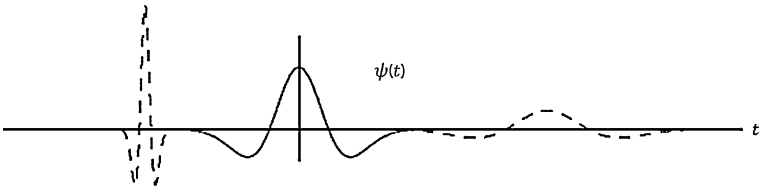


Fig. 13.1 Example of an expanded and translated wavelet

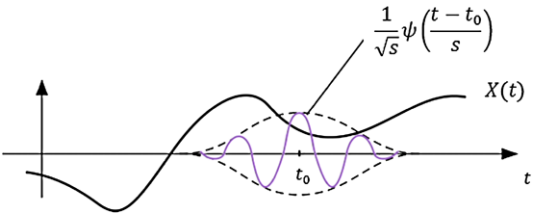


Fig. 13.2 The process of a wavelet transform

Further, the wavelet is normalized $\|\psi\| = 1$, and is centered in the neighborhood of $t = 0$. By expanding the wavelet ψ by a factor s and then translating it by u (see the example in Fig. 13.1), we obtain the family of wavelets $\psi_{u,s}$ associated with ψ, with the same standard unit (that is, $\|\psi_{u,s}\| = 1$):

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right). \tag{13.2}$$

The continuous wavelet transform of a signal $X \in L^2(\mathbb{R})$ at time u and scale s is defined by:

$$W_X(u,s) = \langle X, \psi_{u,s}\rangle = \int_{+\infty}^{-\infty} X(t)\frac{1}{\sqrt{s}}\psi^*\left(\frac{t-u}{s}\right)dt \tag{13.3}$$

where W refers to the wavelet and ψ^* denotes the complex conjugate of ψ.

The relationship (13.3) represents the scalar product of X and the set of wavelets $\psi_{u,s}$ associated with ψ. $W_X(u,s)$ characterizes the "fluctuations" of the signal $X(t)$ in the neighborhood of position u at scale s (see Fig. 13.2; here u takes the specific value t_0).

Examining expressions (13.1) and (13.3), it is clear that $W_X(u,s)$ will be insensitive to the signal's most regular behaviors and more flexible than polynomials of degree strictly less than n (the number of vanishing moments of ψ). Conversely, $W_X(u,s)$ takes into account the irregular behavior of polynomial trends. This important property plays a role in the detection of signal singularities.

$W_X(u,s)$ can also be interpreted as a linear filter operation:

$$W_X(u,s) = X * \bar{\psi}_s(u) \tag{13.4}$$

where $*$ denotes the product of convolution, with $\bar{\psi}_s(t) = \frac{1}{\sqrt{s}}\psi^*(\frac{-t}{s})$, in which the Fourier transform, $\widehat{\bar{\psi}}_s(\omega) = \sqrt{s}\widehat{\psi}^*(s\omega)$, is identified as a transfer function of a bandpass filter [2].

The relationship (13.4) demonstrates that wavelet transforms can be calculated by expanded bandpass filters (with variable s). The inverse of a continuous wavelet transform in L^2 is provided by the wavelet admissibility condition:

$$K_\psi = \int_0^{-\infty} \frac{|\widehat{\psi}(\omega)|^2}{|\omega|}\, d\omega = \int_{+\infty}^{0} \frac{|\widehat{\psi}(\omega)|^2}{|\omega|}\, d\omega < +\infty. \tag{13.5}$$

In order for this integral to be finite, it is necessary to ensure that $\widehat{\psi}(0) = 0$, which is why wavelets must have a mean of zero ($\widehat{\psi}(0) = \int_{\mathbb{R}} \psi(t)\, dt = 0$). This condition is almost sufficient. If $\widehat{\psi}(0) = 0$ with continuously differentiable $\widehat{\psi}(\omega)$, the admissibility condition is satisfied.

In practice, choosing a wavelet with a zero mean (and highly localized in time and in frequency) is sufficient. In this case, it is possibly to synthesize or reconstruct signal $X(t)$ by inverting the wavelet transform as follows:

$$X(t) = \frac{1}{K_\psi} \int_{+\infty}^{0} \int_{+\infty}^{-\infty} W_X(u,s) \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) du \frac{ds}{s^2}, \quad t \in \mathbb{R}. \tag{13.6}$$

This reconstruction uses all scales, and as such is highly redundant. Continuous wavelet transform is calculated based on the scale factor s and the time u in the set of real numbers (the time-scale plane is therefore continuously traversed), which renders it extremely redundant. In reconstruction of a signal by a continuous inverse transform, this redundancy is extreme in the sense that all the expanded and translated wavelets are employed such that they are linearly dependent, therefore reflecting existing signal information without adding new information.

13.1.3 Discrete Wavelet Transforms

As previously noted, continuous wavelet transforms are highly redundant: $W_X(u,s)$ is the 2D ((u,s)-plane) representation of a signal $X(t)$ in 1D! This redundancy can be reduced using one of the countably infinite family of wavelets $\{\psi_{j,k}\}$, where $(j,k) \in \mathbb{Z}^2$ and $\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k)$. The time-scale (u,s)-plane is converted to a "dyadic mesh" (or in base 2), as shown in Fig. 13.3:

$$\left(u \to 2^j k, s \to 2^j, (j,k) \in \mathbb{Z} \times \mathbb{Z}\right).$$

Clearly, to reduce or eliminate redundancy, the family $\{\psi_{j,k}\}_{(j,k)\in\mathbb{Z}^2}$ must constitute an orthonormal basis of $L^2(\mathbb{R})$. The conditions under which this basis becomes orthonormal and thus provides a "highly economical" wavelet transform are related to the concept of multiresolution analysis (abbreviated MRA).

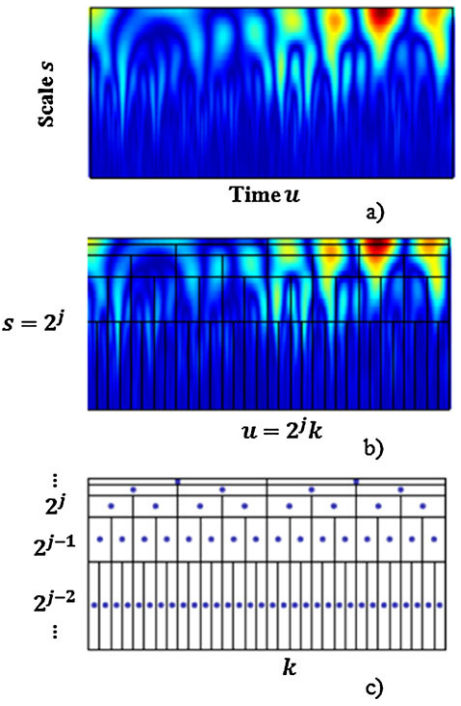


Fig. 13.3 Discretization process: (**a**) continuous wavelet transform, (**b**) discretization of the (u, s)-plane, (**c**) discrete wavelet transform

To profit from a non-redundant signal representation while ensuring a perfect reconstruction from its decomposition, an extremely effective tool, i.e., MRA, was defined by Stéphane Mallat [2] and Yves Meyer [9]. This powerful concept allows numerical implementation of wavelet decomposition; the definition of the discrete wavelet transform thus necessarily underpins that of the MRA.

13.1.4 The Concept of MRA

The idea of multiresolution analysis of a signal consists in its representation as a limit of its successive approximations, where each approximation is a smoothed version of the preceding approximation. Successive approximations are presented at different resolutions, hence the term multiresolution analysis.

When resolution increases, successive images approximate the signal increasingly better, and in contrast, when resolution decreases, the amount of information contained in an image also decreases, eventually to zero. The wavelet coefficients encode the difference in information between two successive images, that is, the details acquired by an image when its resolution doubles.

Readers interested in the theoretical aspects will find the axiomatic formulation of MRA in references [2] and [10].

Clearly, the idea of multiresolution analysis can be summarized as follows: It concerns the representation of a signal in the form of a coarse approximation and a series of "corrections" of decreasing amplitude. A true multiresolution analysis provides a seductive algorithmic element (see the Mallat algorithm in the next section) that paves the way for some impressive applications, notably in compression, denoising, image restoration, smoothing, computer graphics, vision, etc.

From a formal perspective, multiresolution analysis of a signal X consists of the realization of successive orthogonal projections of the signal on the spaces V_j, which leads to increasingly coarse approximations of X as j increases. The difference between two successive approximations represents detail information lost during the transition from one resolution to another. This detail information is contained in the subspace W_j orthogonal to V_j.

Thus the signal X belonging to a space V_j is projected on a subspace V_{j+1} and a subspace W_{j+1} with the aim of reducing the resolution by half. There is therefore:

- A scaling function $\phi(t)$ which generates an orthonormal basis of V_{j+1} via expansion and translation, and
- A wavelet function $\psi(t)$ which generates an orthonormal basis of W_{j+1} via dilation and translation.

The projection of the signal X on the space V_{j+1} is denoted as

$$A^j X = \sum_k \langle X, \phi_{j,k} \rangle \phi_{j,k} = \sum_k a_{j,k} \phi_{j,k} \tag{13.7}$$

where the scalar products[2] $a_{j,k} = \langle X, \phi_{j,k} \rangle$ provide an approximation to scale 2^j. The projection of the signal X on the space W_{j+1} is denoted as

$$D^j X = \sum_k \langle X, \psi_{j,k} \rangle \psi_{j,k} = \sum_k d_{j,k} \psi_{j,k} \tag{13.8}$$

where $d_{j,k} = \langle X, \psi_{j,k} \rangle$ are the coefficients of the wavelet transform of the signal X.

13.1.5 Implementation of MRA: Mallat Algorithm

According to relationships (13.7) and (13.8), the decomposition of a signal into wavelet bases involves a succession of discrete convolutions with the impulse response lowpass filter $\bar{h}$ and the highpass impulse response filter $\bar{g}$ as shown in Fig. 13.4.

The coefficients a_{j+1} and d_{j+1} are calculated by taking every other sample from the convolutions of a_j with $\bar{h}$ and $\bar{g}$, respectively, and so on.

[2] $a_{j,k} = \langle X, \phi_{j,k} \rangle = \int_{-\infty}^{+\infty} X(t) 2^{-j/2} \phi(2^{-j}t - k)\, dt$.

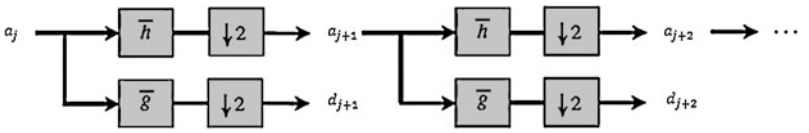


Fig. 13.4 Decomposition of a signal on multiple levels

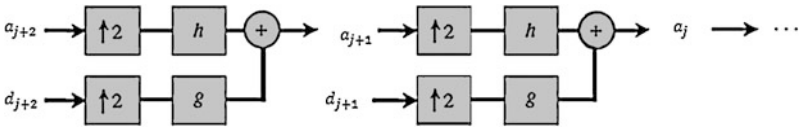


Fig. 13.5 Reconstruction or synthesis

The $\downarrow 2$ symbol represents a decimation of factor 2, that is, accounting for the use of every other (rather than every) sample.

Fast wavelet transforms are therefore calculated by the cascade of filterings by $\bar{h}$ and $\bar{g}$ followed by subsampling (or decimation) by a factor of 2. Initialization of the algorithm can present certain difficulties; however, it is possible to assimilate the sampled values of the signal X with the coefficients a_0. The complexity of this algorithm is of the order of N when the signal X is of size N.

Reconstruction or synthesis consists of an interpolation which inserts zeros in the sequences a_{j+1} and d_{j+1} to double their length, followed by a filter, as shown in Fig. 13.5.

The $\uparrow 2$ symbol represents an interpolation which inserts zeros between the samples of a_{j+1} and d_{j+1}.

Strictly speaking, the sequences a_j and d_j are respectively $a_{j,k} = a_j(k)$ and $d_{j,k} = d_j(k)$; where k is the time. There is a relationship between $h(k)$ and $g(k)$:

$$g(k) = (-1)^k h(1-k), \tag{13.9}$$

and the following notation will be adopted: $\bar{h}(k) = h(-k)$ and $\bar{g}(k) = g(-k)$. In two dimensions, decomposition into a separable wavelet basis is realized by an extension of the Mallat algorithm. Therefore, for an image, the 1D algorithm is applied first, to each row and then to each column, as shown in Fig. 13.6.

13.2 Redundant Multiresolution Analysis

When high accuracy is required in image analysis, redundant multiresolution analysis is applied, such as the undecimated discrete wavelet transform, the wavelet packet transform, and the contourlet transform. This section introduces these transforms and shows their benefits.

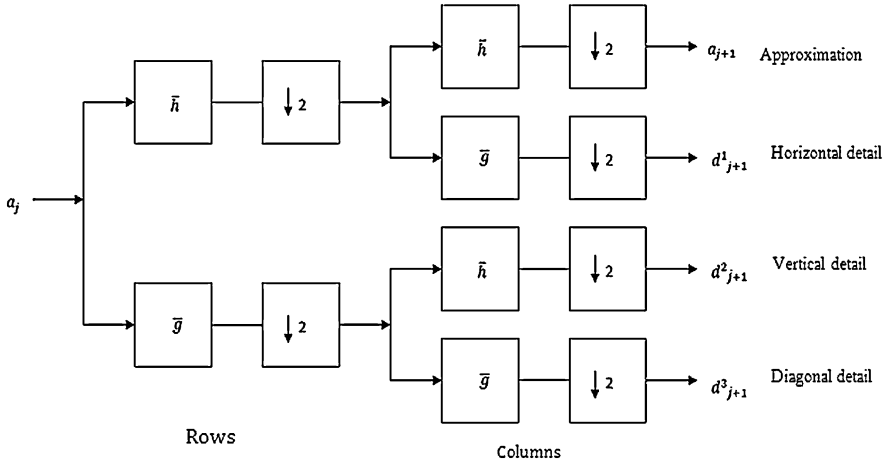


Fig. 13.6 The Mallat algorithm: decomposition of an image

13.2.1 Undecimated Discrete Wavelet Transform

The undecimated discrete wavelet transform (UDWT), also known as the stationary wavelet transform, consists of keeping the filter bank construction which provides a fast and dyadic algorithm, e.g., Mallat algorithm, but eliminating the decimation step.

Due to the absence of downsamplers (decimation step) in the UDWT's implementation, each coefficient sequence from any level of decomposition has the same length as the original: if the original signal has N samples, the UDWT J-level representation $\{a_J(k), d_j(k)\}_{0<j<J}$ is of size $N(J+1)$, making from the UDWT J-level a highly redundant representation.

The implementation of the UDWT was initially performed by an algorithm called the *à trous* algorithm (*à trous*, a French term, meaning *with holes*).

13.2.2 Wavelet Packets

Wavelet packets provide a finer analysis by decomposing, at each level, not only the approximation spaces but also the detail spaces (see Fig. 13.7). Wavelet packets, defined by Coifman, Meyer, and Wickerhauser [11], therefore represent a generalization of multiresolution decomposition.

The wavelet packets transform is redundant and should only be used in cases where an extremely fine analysis is required. The choice of the best decomposition basis depends on the principle of minimal entropy.

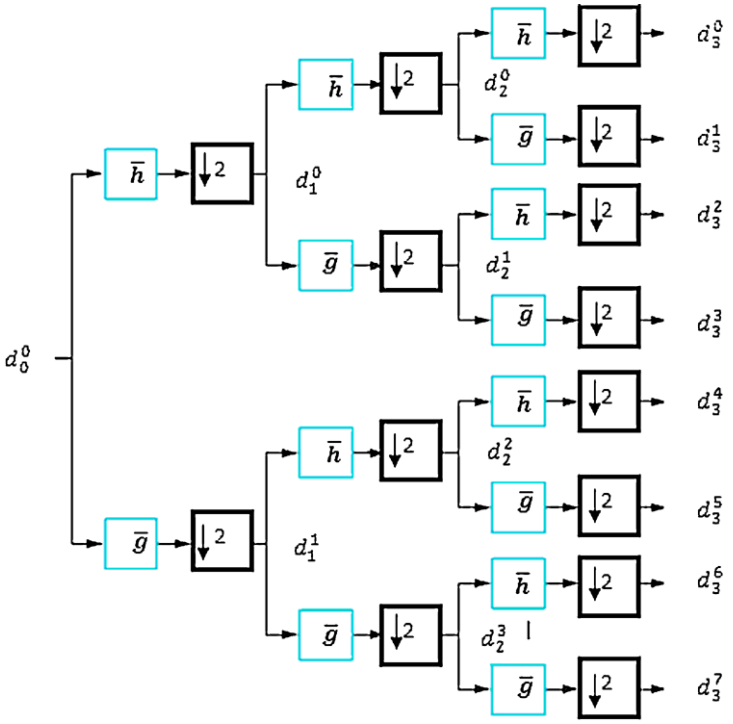


Fig. 13.7 Example of decomposition (at three levels) into wavelet packets by cascading filtering and decimation

13.2.3 Contourlet Transform

Do and Vetterli [12] have proposed the contourlet transform (CT), which is one of several transforms developed in recent years, aimed at improving image multiresolution analysis based on discrete wavelet transform. The main feature of these transforms is the potential to efficiently handle 2D singularities, i.e., edges, unlike wavelets which can deal with point singularities exclusively. This difference is caused by the two main properties that the CT possess:

- Directionality, i.e., the representation should contain basis functions in many directions, as opposed to only 3 directions (horizontal, vertical, and diagonal) of wavelets,
- Anisotropy, i.e., the representation should capture smooth contours. It should contain basis functions using a variety of elongated shapes with different aspect ratios.

The main advantage of the CT over other geometrically-driven representations, e.g., curvelets and bandelets, is its relatively simple and efficient wavelet-like implementation using iterative filter banks. Consequently, the contourlet transform is a

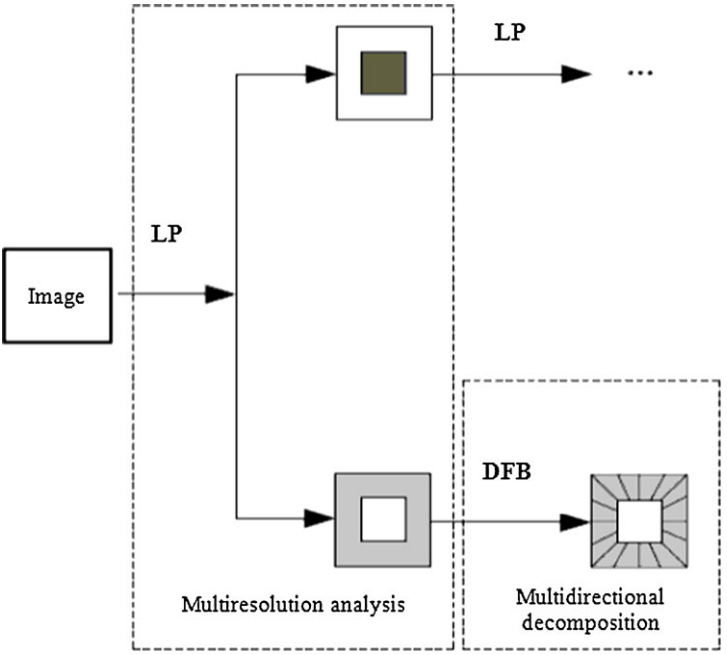


Fig. 13.8 First level decomposition of the Contourlet Transform

true multiresolution and multidirectional image representation which can effectively capture image edges and contour information in all directions; therefore, it is very suited for image processing, namely in image denoising.

The Contourlet Transform is constructed by the Laplacian pyramid (LP) and directional filter banks (DFB) as illustrated in Fig. 13.8. The LP decomposes images into subbands and DFB analyzes each detail image.

13.3 Denoising or Noise Reduction

The challenge is to restore a useful signal when only a noisy version is available. The idea consists simply in adequate modification of the coefficients (of the wavelet transform of the observed signal) taking advantage of their local properties, then inverting the transformation to obtain a noise-free version of the signal!

For a 1D signal, wavelet shrinkage denoising attempts to remove whatever noise is present and retain whatever signal is present regardless of the frequency (or scale) content of the signal. It is not a smoothing (averaging) of data. Smoothing removes high frequencies and retains low frequencies. Consequently, for a 2D signal, smoothing introduces a blurring and loss of information.

Wavelet shrinkage denoising consists of three steps: a linear DWT, a nonlinear shrinkage denoising, and a linear inverse DWT. This heuristic procedure is consid-

ered a nonparametric method, i.e., it makes no a priori assumption. It is distinct from parametric methods in which parameters must be estimated for a particular model that is assumed a priori. For example, the most popular parametric method is that of using least squares estimation.

13.3.1 Additive Gaussian White Noise Model

The following is a measurement model:

$$Y(k) = X(k) + B(k) \tag{13.10}$$

where $Y(k)$ is the measurement signal of size N, $X(k)$ is the a priori unknown useful signal, and $B(k)$ is a random noise perturbation (usually assumed to be white Gaussian with variance σ^2).

The elimination or reduction of this additive noise can be achieved nonlinearly by using multiresolution analysis under the assumption that the appropriate choice of a decomposition basis allows discrimination of the useful signal from noise. The underlying idea is that the useful signal can be described by a small number of coefficients of high-amplitude wavelets, and that the noise is spread across all coefficients. This hypothesis justifies, in part, the traditional use of denoising by thresholding.

If $d_j^Y(k)$ represent the wavelet coefficients of the measured signal, the estimation of the wavelet coefficients of the useful signal, denoted $d_j^{\widehat{X}}(k)$, is generated by two types of thresholding:

- **Hard thresholding**

$$d_j^{\widehat{X}}(k) = \begin{cases} d_j^Y(k) & \text{if } |d_j^Y(k)| > S, \\ 0 & \text{otherwise;} \end{cases} \tag{13.11}$$

- **Soft thresholding**

$$d_j^{\widehat{X}}(k) = \begin{cases} d_j^Y(k) - S & \text{if } d_j^Y(k) > S, \\ d_j^Y(k) + S & \text{if } d_j^Y(k) < -S, \\ 0 & \text{otherwise} \end{cases} \tag{13.12}$$

with $S = \sigma\sqrt{2\ln N}$ where N is the size of the measured signal, and σ represents the noise standard deviation.

A robust estimator of σ is given by:

$$\widehat{\sigma} = 1.4826 \times \text{Med}\left|d_1^Y(k)\right| \tag{13.13}$$

where $\text{Med}|d_1^Y(k)|$ designates the median value of the wavelet coefficients, for $j = 1$, in increasing order $\{|d_1^Y(k)|, 0 \le k \le N/2\}$.

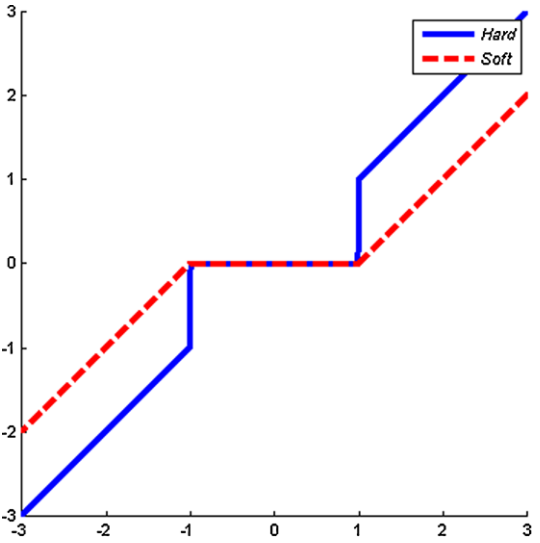


Fig. 13.9 Hard thresholding and soft thresholding

Donoho and Johnstone [13] have shown that the choice of S is near-optimal for $N \geq 4$.

Fig. 13.9 represents the thresholding procedure. Note that hard thresholding creates discontinuities at $k \pm 1$.

Heuristic denoising is based on the hard and soft thresholding functions (see Fig. 13.9), the non-negative garrote (NNG) function and the smoothly clipped absolute deviation (SCAD) function, both illustrated in Fig. 13.10.

Although these standard thresholding functions are close to optimal, they raise some limitations:

- The hard thresholding function is not everywhere continuous and its discontinuities at $k \pm 1$ generate a high variance in the estimated signal;

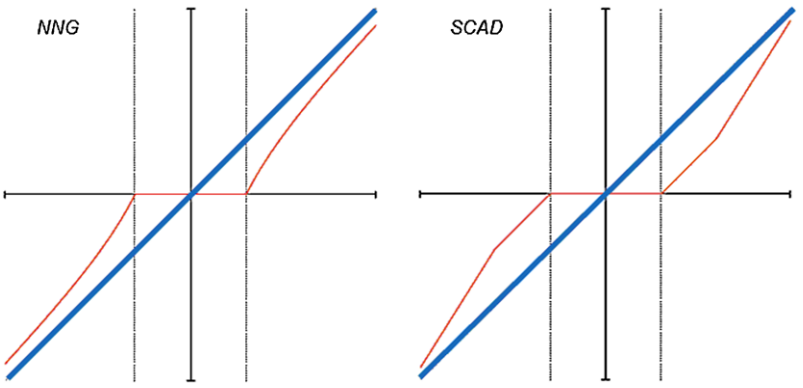


Fig. 13.10 NNG and SCAD shrinkage functions

Fig. 13.11 Denoising of the Lena image corrupted by Gaussian white noise

- The soft thresholding function is continuous, but creates an attenuation on large coefficients, which results in an oversmoothing and an important bias for the estimated signal;
- The NNG and SCAD functions achieve a certain compromise between the hard and the soft thresholding functions.

However, all the standard Wavelet Shrinkage functions presented above include zero-forcing. This zero-forcing induces singularities of the thresholding function. As a consequence, it results in a significant variation of the estimation due to the sensitivity of the inverse wavelet transform. In addition, thresholding rules assume that the wavelet representation is sparse. Note that smooth signals yield sparse wavelet representations in the sense given by: For such signals, large coefficients are very few in number.

In contrast, wavelet representations of natural images, which tend to be piecewise regular (where discontinuities are along smooth curves) rather than smooth, fail to be sparse enough since large coefficients are not very few. This justifies the introduction of more flexible wavelet shrinkage methods for correcting the drawbacks of thresholding rules.

Figure 13.11 depicts the original Lena image (top left) and the same image made artificially noisy (to the right) using Gaussian noise (randn function). This image has been denoised (see Fig. 13.11, bottom panel) using the Matlab function wdencmp with a fixed soft threshold. The Matlab code for this application is as follows:

```
% Load Lena image and add Gaussian noise
load  lena;
init = 2055615866; randn('seed',init);
x = X + 15*randn(size(X));
%Specification of fixed threshold and type of %thresholding
[thr,sorh,keepapp] = ddencmp('den','wv',x);
%Denoising using a symlet wavelet of order 4
```

```
%Soft thresholding has been chosen
xd = wdencmp('gbl',x,'sym4',2,thr,sorh,keepapp);
% Graphical representation
figure('color','white')
colormap(pink(255)), sm = size(map,1);
image(wcodemat(X,sm)), title('Original Image')
figure('color','white')
colormap(pink(255))
image(wcodemat(x,sm)), title('Noisy Image')
image(wcodemat(xd,sm)), title('Denoised Image')
```

This denoising procedure is achieved in three stages:

- Two-level wavelet decomposition,
- Thresholding, in the three directions of the detail coefficients (horizontal, vertical, and diagonal),
- Reconstruction of the image from the level 2 approximation coefficients (unchanged) and the modified detail coefficients.

The wavelet used is a symlet of order 4. The result of this simulation is very satisfying. However, in a real-life situation applying a denoising procedure is not as simple because it requires adaptive processing of image zones affected by different types of local degradation.

This area still attracts significant research interest, via the development of fine models of noise and the concept of parsimony structured by block models [14] and [15]. Such a statistical procedure has properties of theoretical optimality remarkable practicality.

From an operational perspective, it consists of regrouping unknown coefficient estimations (of wavelets or contourlets) into several disjoint blocks and selecting these groups via the so-called James–Stein rule. This procedure breathes new life into image denoising, from the point of view of both quasi-optimality and the reduction of calculation time by a factor of 6!

13.3.2 Sigmoidal Wavelet Shrinkage

The sigmoidal wavelet shrinkage [16] performs an adjustable wavelet shrinkage based on sigmoid function thanks to parameters that control the attenuation degree imposed to the wavelet coefficients. Consequently, this denoising method allows for a very flexible shrinkage. This shrinkage function is defined by:

$$\delta_{\tau,\lambda}(x) = \frac{x}{1 + e^{-\tau(|x|-\lambda)}} \tag{13.14}$$

for $x \in \mathbb{R}$ and $(\tau, \lambda) \in \mathbb{R}_+^* \times \mathbb{R}_+$. Each $\delta_{\tau,\lambda}$ is the product of the identity function with a sigmoid-like function. From Fig. 13.12, we can see that when τ tends to infinity, $\delta_{\tau,\lambda}$ tends to a hard thresholding function.

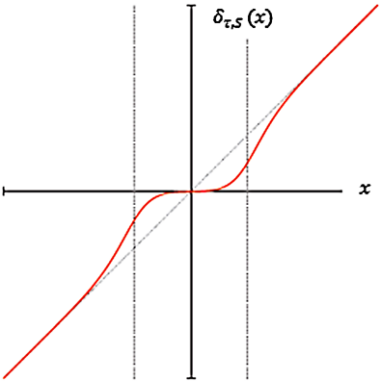


Fig. 13.12 Sigmoidal wavelet shrinkage

Fig. 13.13 *Right*: Noisy Lena image (Noise is white and Gaussian with $\sigma = 35$) and PSNR = 17 dB. *Left*: Denoised Lena using 4-level-sigmoidal wavelet shrinkage, PSNR output = 30 dB

Figure 13.13 gives a denoising example using the example of Lena image, corrupted by additive white Gaussian noise with standard deviation $\sigma = 35$. The Haar wavelet and 4 decomposition levels are used for the undecimated wavelet representation.

It is necessary to establish quantitative/heuristic measure to compare the effects of image denoising algorithms on image quality. For every test, the PSNR is calculated for the original image and the denoised image. The PSNR (in decibel unit, dB), is given by

$$\text{PSNR}_{\text{dB}} = 10 \log_{10} \left(\frac{d^2}{\varepsilon} \right) \tag{13.15}$$

where d represents the dynamics of the image (for example, for a 8-bit encoded images, $d = 255$), and

$$\varepsilon = \frac{1}{MN} \sum_{k=1}^{M} \sum_{l=1}^{N} \| I_1(k,l) - I_2(k,l) \| \tag{13.16}$$

is the mean square error between image I_1 and its noisy version I_2, each being of size $M \times N$. Typical PSNR values in real-world images range between 20 and 40 dB.

Figure 13.13 gives the noisy Lena with a PSNR = 17 dB as well as the denoised image using 4-level-sigmoidal wavelet shrinkage. The output PSNR is then equal to 30 dB. This figure highlights that the contrast of the image can be smoothly adjusted by sigmoidal wavelet shrinkage introducing artifacts.

13.3.3 Parametric Denoising: Wiener Filtering

The Wiener filter reduces or removes noise affecting the signal by comparing the observed signal with an estimation of the noiseless signal and minimizing the mean square error, in this sense the Wiener filter is optimal.

However, this technique assumes that the observed signals are stationary and/or their second order statistics are known. However, in the real world, this assumption is not always true, which severely limits the performance of Wiener filtering.

It is well known that the most important technique for removal of blur in images due to linear motion or unfocussed optics is the Wiener filtering.

Let $y = x + b$ be the observed sequence applied at the input of the Wiener filter with impulse response h to remove the noise perturbation b. The denoised signal given by the filter will be:

$$\hat{x} = y * h = \sum_{i=0}^{N} h_i\, y(k-i) \tag{13.17}$$

and the error of estimation is:

$$e(k) = \hat{x} - x(k) = \sum_{i=0}^{N} h_i\, y(k-i) - x(k) \tag{13.18}$$

where $x(k)$ is the noiseless ideal signal. The filter coefficients are the solutions of the equation:

$$h_i = \arg\min \left\{ \mathrm{E}\left\{ e^2 \right\} \right\} \tag{13.19}$$

where E is the mathematical expectation (taking the average). The solution of relation (13.19) is obtained by solving the following equation:

$$\begin{pmatrix} R_y(0) & R_y(1) & \cdots & R_y(N) \\ R_y(1) & R_y(0) & \cdots & R_y(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ R_y(N) & R_y(N-1) & \cdots & R_y(0) \end{pmatrix} \times \begin{pmatrix} h_0 \\ h_1 \\ \vdots \\ h_N \end{pmatrix} = \begin{pmatrix} R_{yx}(0) \\ R_{yx}(1) \\ \vdots \\ R_{yx}(N) \end{pmatrix} \tag{13.20}$$

where R_y is a positive definite Toeplitz matrix representing the autocorrelation of the observed signal, h is the Wiener filter's vector of coefficients, and R_{yx} is the intercorrelation between the observed and the noiseless signal.

The working hypotheses from an operational point of view are:

- The signal x and the noise b are uncorrelated, namely, $R_{bx} = 0, \forall k$;
- The noise, b, is an additive white Gaussian process, with zero-mean and variance σ_b^2;
- The useful signal, x, is a random Gaussian process with zero-mean and variance σ_x^2.

Consequently, the observed signal is a zero-mean process with variance $\sigma_y^2 = \sigma_x^2 + \sigma_b^2$. Its autocorrelation is reduced to $R_y(k) = \sigma_y^2 \mathbf{I}$ where $\mathbf{I}$ denotes the identity matrix. The intercorrelation between the observed and the noiseless signal is then $R_{yx}(k) = \sigma_y^2 \delta(k), \forall k$ where $\delta(k)$ is defined by

$$\delta(k) = \begin{cases} 1 & \text{if } k = 0, \\ 0 & \text{otherwise.} \end{cases}$$

In these conditions, the solution of Eq. (13.20) is then:

$$h_0 = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_b^2}, \quad \text{and} \quad h_1 = h_2 = \cdots = h_N = 0.$$

The denoised signal can be put in the following form:

$$\widehat{x} = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_b^2} y = \frac{\text{SNR}}{1 + \text{SNR}} y \tag{13.21}$$

where $\text{SNR} = \frac{\sigma_X^2}{\sigma_b^2}$; it represents the signal-to-noise ratio of the observed signal. In the case of an image, the variance of the useful signal, σ_X^2, can vary in space and must be estimated locally.

The relations deduced for time (or space)-domain signals are also available in the wavelet domain, under the same hypotheses, the estimated coefficients being computed with:

$$d_j^{\widehat{X}} = \frac{\sigma_X^2}{\sigma_X^2 + \sigma_b^2} d_j^Y. \tag{13.22}$$

In the context of medical imaging (see Sect. 13.4), we compare the performance of this denoising method to shrinkage wavelet denoising both visually an in terms of PSNR.

13.3.4 Suppression of Correlated Noise

Images captured by digital devices often contain noise. Various methods of wavelet-based image exist, but their performance is limited in the presence of correlated noise in the image or signal of interest.

Fig. 13.14 Denoising based on estimation of the probability of the presence of useful information: 3-band Lena (RGB)

In this context, only advanced heuristics are suitable to properly denoise images corrupted by correlated noise. These methods combine parametric methods based on a priori knowledge (or statistical modeling) and nonparametric methods such as thresholding in the wavelet domain. Among the methods which demonstrate some effectiveness is the work of A. Pizurica [17]. The heart of this approach is to estimate the probability that a given wavelet coefficient contains a significant noise-free component. Heuristically, the signal of interest is identified and extracted from the noisy image. Figure 13.14 shows the application of this method to a three band image (RGB), specifically to the color Lena image.

A few years later, another method was proposed by the same team [18] to extract the useful signal. This is characterized by measuring and quantifying the relevant information in a noisy image taking into account the structure of the correlation of neighboring wavelet coefficients. The approach consists of combining an intra-scale model with a hidden Markov type model to capture these dependencies between wavelet coefficients.

Figure 13.15 shows an example of application of the denoising method based on a Markov model and a slightly redundant discrete wavelet transform. This method is compared with SNR: it appears that the denoising presented in Fig. 13.15(d) offers a slight improvement.

13.4 Applications in Medical Imaging

In recent years, medical imagery and diagnostic techniques have seen spectacular developments and heavy investment, and research hospitals in particular have established neuroimaging centers equipped with functional PET (positron emission tomography) and MRI (magnetic resonance imagery) equipment.

These new technologies supplement the more classic techniques, which are also perpetually evolving, for example, ultrasound, X-ray tomography (or classical scanning), magnetoencephalography (or MEG, offering access to the spatiotemporal

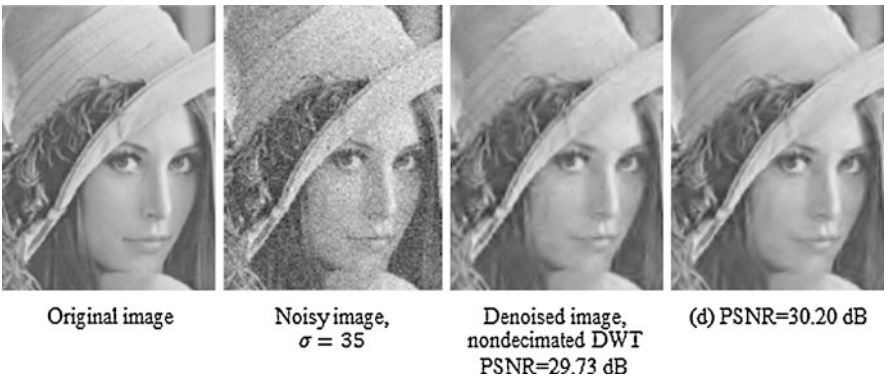


Fig. 13.15 Denoising of the Lena image by modeling with a hidden Markov model combined with a slightly redundant wavelet transform (**d**)

timecourse of cerebral activity), electroencephalograpy (EEG), and electrocardiography (ECG). However, image and signal processing techniques, commonly used in these domains, remain rudimentary despite the complexity of the signals to be analyzed (presence of anomalies of outliers, signal mixing, combination of associated modalities, inverse problems, and so on) and the demands in terms of extracting relevant information are increasingly greatly, for example, in terms of the size of a data set, data modeling, image registration and fusion imaging (modalities), as well taking into account any current major methodological issues in the field.

The challenge, then, is to establish how wavelet-based multiresolution analysis (decomposition–reconstruction, feature extraction, segmentation, contour detection, compression, denoising, progressive transmission, and so on) can be combined with existing classification techniques to meet present and future scientific and technological challenges.

13.4.1 Medical Imaging Methods and Techniques

Medical imaging includes the following techniques and methodologies:

- Acquisition, restoration, and image processing of the human body,
- Interpretation and exploitation of these images for therapeutic purposes.

The process of formation or generation of these images is based on principles from physics, such as the absorption of X-rays (radiography, mammography, scanner, or tomodensitometry, or computerized tomography), the magnetic field (spectroscopy and magnetic resonance imagery + and functional magnetoencephalography), propagation and reflection of ultrasonic waves (echography, Doppler, elastography, photoacoustics, thermoacoustics, fUltrasound, or functional brain ultrasound), radioactivity (gammagraphy or scintigraphy or single photon emission computed tomography-SPECT-or positron emission tomography, PET)

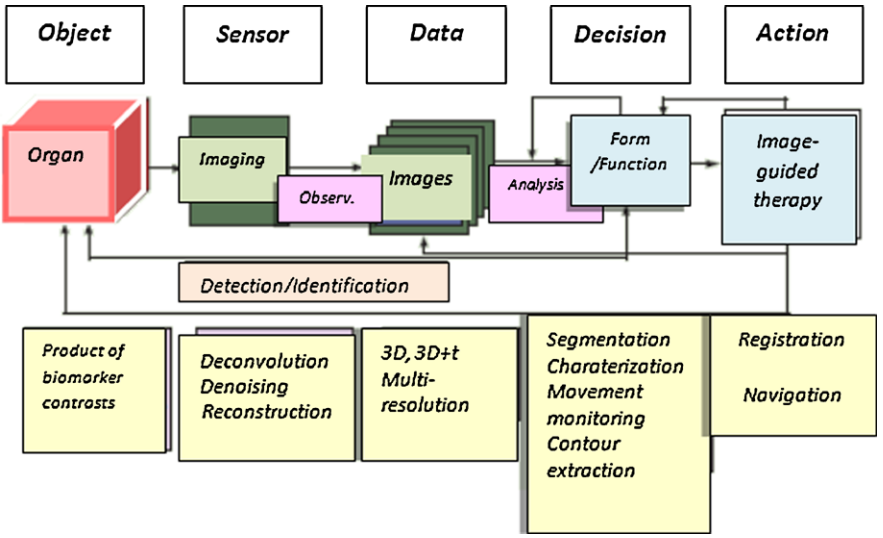


Fig. 13.16 Representation of the interrelated components of the imaging sequence

and optics (coherent optical tomography, diffuse optical imaging). These various image sources provide usually complementary insights, whether it be in the morphological (structure and anatomy) or functional (physiology, metabolism) plane.

Figure 13.16 gives an overview of medical image formation and exploitation:

- An object corresponds to an organ or a lesion (bone fracture, tumor, atheromatous plaque). It may be static or moving, a superficial tissue element or a large, important structure;
- The Sensor section corresponds to the technique of observation or measure concerned which allows the exploration of this object in its context, or separately, at a macroscopic level just as at the microscopic level, its shape, its architecture, or while functioning;
- Data (2D, 3D, or time sequence images) are only obtained after using signal processing techniques and reconstruction algorithms allowing the correction of defects intrinsic to the sensors (noise, distortion) and to multiple wave–matter interactions. In terms of information management of these data, the DICOM norm is generally adopted. Methods of signal and image processing, such as noise reduction (or denoising) by multiresolution analysis, are indispensable for efficient processing. However, artifacts frequently remain, and are linked to the reconstruction methods themselves, as well as to movements of the objects in the course of acquisition. The final spatiotemporal resolution as well as the contrast between objects depends on all these factors. Access to an image is often accompanied by the injection of agent or contrast products (enhancement of vascular structures in X-ray imaging or even in echography), biomarkers for highlighting lesions or specific biological processes, and radio-pharmaceutical compounds (tracers) that

form the basis of nuclear medical imaging (single photon emission tomography, or, using positrons, PET);

- Decision is based on the extraction of relevant information (of an object from others, of a movement, etc.) included in the images for clinical interpretation. Segmentation and contour extraction methods using wavelet multiresolution analysis play a central role here. Similarly, algorithms for estimation and movement tracking, to characterize cardiac kinetics, for example, or to improve reconstruction by correcting for breathing, have seen considerable advances. This is also the case for algorithms dealing with image registration techniques for comparing images of the same type but taken at different times, and for merging different (i.e., multimodal images) image modalities;
- Action consists of making a diagnosis and also of devising a therapeutic response to that diagnosis. In this phase, images play an essential role in both preparation (the planning phase) and application (guidance) of the therapy. This may in the first instance be to simulate the deformations introduced by the actions to be carried out (instrument–tissue interaction such as during the introduction of an endovascular probe) and in the second instance, to locate in real time the instruments relative to the organs (to reach a target, or follow a precalculated trajectory) in real time.

The proliferation of techniques and the complementary nature of these methodologies push progress in the direction of a multimodal imagery, in which data generated by several technologies, whether acquired simultaneously or not, are recalibrated, that is, mapped onto the same image. Thus, the fusion of techniques revealing the anatomy, function, and activity of a structure provides increasingly focused information. For example, the joint use of magnetoencephalography (MEG, a technique derived from electroencephalography in which the magnetic field generated by neuronal activation is measured using highly sensitive sensors) and of functional magnetic resonance imaging (fMRI) can identify the most complex neural processes, such as object and face recognition. Another example is that of superimposing on the same image the morphology of the contours of the heart obtained using MRI with information on the mobility of the heart walls obtained using Doppler echography. Some recent imaging devices can produce multimodal images in a single examination (for example, hybrid CT-SPECT systems).

Electromagnetic MEG imaging reveals a network of regions in which low-frequency activity is synchronized with the speed of the hand during manipulation of a classic ball mouse. The traces show the hand speed in green (also lighter) recorded during a 3-second period, and the corresponding cerebral activity in blue (also darker) in the principal region implicated in hand motility.

To access the temporal resolution of cognitive functions, magnetoencephalography (MEG) can be used in addition to MRI. This imaging method, especially in terms of sensors, is making advances which are leading to wider applications. Indeed, magnetoencephalography (MEG) offers better resolution, measuring the magnetic field generated by neurons with 150 or even 300 sensors placed on a cap (without direct contact).

Recent studies reveal that it is possible to demonstrate in humans, in a comprehensive and noninvasive manner, the interaction of brain activities in a network, and whose characteristic frequencies are directly related to basic behavioral parameters, in the case of limb movement. This discovery may have major implications in the evaluation of pathologies giving rise to changes in motor control, and in the implementation of prosthesis control techniques, when the motor act can be imagined but not performed.

13.4.2 Wavelet-Based Denoising in fMRI, MRI, and Echography

In this section, we will present some practical applications of wavelet domain denoising in MRI and ultrasound images.

13.4.2.1 MRI Illustration

In MRI, the practical limits of the duration of acquisition necessitate a compromise between the signal-to-noise ratio and the resolution of the image. The MR image is commonly reconstructed by calculating the discrete inverse Fourier transform of the data noise. Noise in (the modulus of) the MR image is Rician, the mean of which depends on the signal (that is, it is signal-dependent).

Denoising by multiresolution analysis is generally achieved via uniform or adaptive (hard or soft) thresholding. The algorithm described here offers an additional functionality in adapting to the local spatial context. Because the mean of the Rician noise depends on the signal, the wavelet coefficients and the scaling function (details and approximations) of a noisy MR image are biased estimations. To avoid a signal-dependent bias, the proposed algorithm is applied to the squared modulus of the MR image; the constant bias is therefore subtracted from the scaling coefficients (approximations), and the square root of the denoised image is then calculated.

K. Bartusek et al. [19] have focused on techniques of improving the quality of the image obtained by magnetic resonance (MRI) by applying the wavelet multiresolution analysis. This improvement has been evaluated according to three criteria, of which the most relevant are the signal-to-noise ratio in decibels ($\mathrm{SNR_{dB}}$) and the relative contrast of the image.

Image quality is improved by wavelet-based noise reduction. The two evaluation parameters are defined as follows: The signal-to-noise ratio of an image obtained by magnetic resonance is defined by

$$\mathrm{SNR_{dB}} = 10\log_{10}\left(\frac{I_{\mathrm{mean}}^2}{\sigma_N^2}\right) \tag{13.23}$$

where I_{mean} is the mean of the intensity values of I in a homogenous region of interest (ROI) in the image (signal), and σ_N is the standard deviation of the ROI without

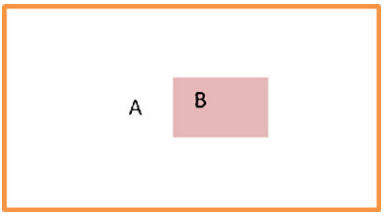


Fig. 13.17 MR image of a phantom for contrast definition

the signal (assumed to be Gaussian noise). For N_{acq} acquisitions, the standard deviation of the noise is

$$\sigma_{\text{eff}} = \frac{\sigma_N}{\sqrt{N_{\text{acq}}}}.$$

The contrast of the image (for an example, see Fig. 13.17) is defined by

$$C_{AB} = |I_A - I_B|.$$

The relative contrast C_{ref} is defined by the contrast with respect to reference intensity image I_{ref}:

$$C_{\text{rel}} = \frac{C_{AB}}{I_{\text{ref}}} = 2\frac{|I_A - I_B|}{I_A + I_B} \tag{13.24}$$

where I_A and I_B are the respective average images (intensities) of zones A and B, as shown in Fig. 13.17.

In this method, the real and imaginary parts of the RM image are filtered separately and the evaluation of the efficiency of the filtering is performed on the resulting complex image. The influence of the chosen mother wavelet (Haar, Daubechies, biorthogonal, symlets, coiflets, Meyer), as well as the type of filtering adopted (hard denoising, soft denoising, Wiener-type using wavelet coefficients), is realized in phantom images and in experimental RM images. The following paragraph summarizes the principal results obtained:

- Format: It is advisable to process both the amplitude and the phase of the image in order to suppress any bias in the resulting image (whilst conserving a high contrast). However, the phase is not always available.
- Quality: It is advisable to use a hard thresholding for denoising images with a high SNR, whilst soft thresholding is preferable for poor quality images.
- Structure: Haar or Daubechies wavelets should be used for images with a simple structure (for example, the phantom test image), and Meyer or symlet wavelets should be reserved for images with a more complex structure (e.g., brain images).
- Resolution quality: If maximum preservation of image information is required, it is advisable to use hard thresholding with a Meyer mother wavelet, or a high-order Daubechies mother wavelet.

In the case of semi-automatic image improvement, a Wiener filter with optimized parameter settings should be used.

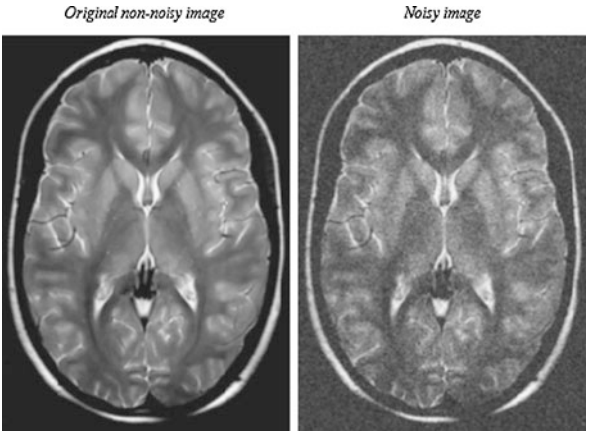


Fig. 13.18 *Left*: original MR image. *Right*: Image noised by Rician noise ($\sigma = 30$, SNR = 5.9 dB)

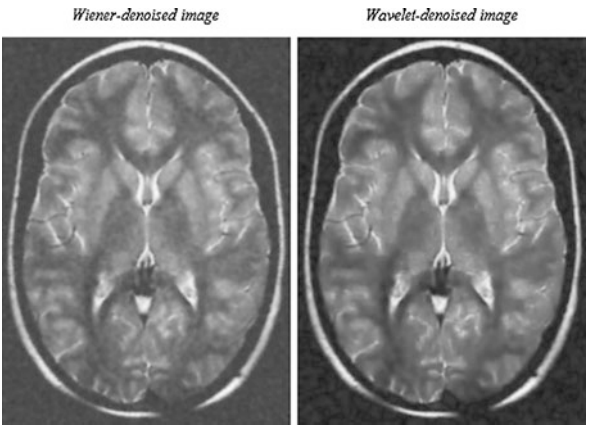


Fig. 13.19 *Left*: Image denoised by a spatially-adaptive Wiener filter (SNR = 10.1 dB). *Right*: Image denoised by the algorithm proposed for $K = 2$ and a window of analysis of size 3×3 (SNR = 12.9 dB)

Performances of the proposed method are illustrated on an MR image with artificially added Rician noise (see Fig. 13.18), and compared to spatially-adaptive Wiener filtering. Figure 13.19 shows that, qualitatively, denoising using the proposed method clearly surpasses spatially-adaptive Wiener filtering. MR images were provided by the Gand (Belgium) university hospital. Suppression of noise in these images facilitates subsequent automatic processing such as segmentation, for example.

The optimal level of wavelet decomposition is $J = 4$. The tuning parameter value K, which appears optimal, is 3 in echography (see Fig. 13.21) and 2 in MRI (see Fig. 13.19). An increment in this parameter can smooth the image, and consequently can lead to a loss of information. The spatial activity indicator $e(k)$ is calculated by locally averaging neighboring coefficients. A 3×3 window is interesting in terms of SNR in echography as well as in MRI.

This method, proposed by Aleksandra Pizurica [17] is of low complexity, both in terms of implementation and execution time, and adapts to unknown noise and to the local context of the image. The results produced have been demonstrably useful

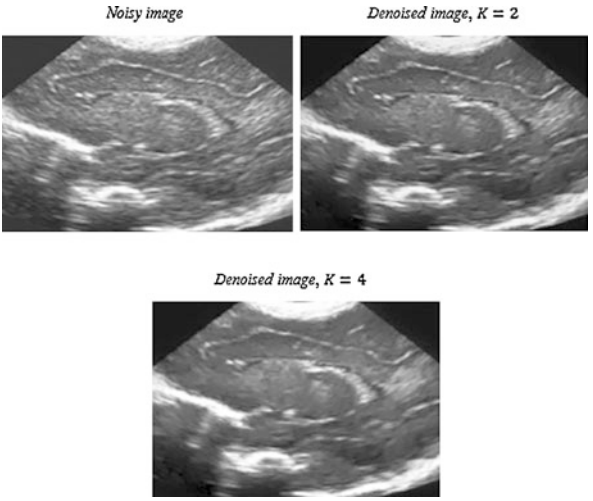


Fig. 13.20 Denoising of an echographic image for $K = 2$ and for $K = 4$

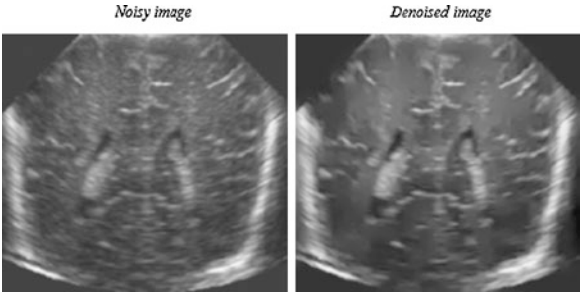


Fig. 13.21 Denoising of an echographic image for $K = 3$

in noise suppressing in medical echography and in magnetic resonance imaging. In these applications, the proposed method clearly surpasses spatially-adaptive algorithm, in terms of quantities measures of performance (SNR, for example) as well as in terms of visual quality of images.

Matlab and CC codes used in denoising are available from: http://telin.ugent.be/~sanja/.

13.4.2.2 Echography Illustration

Figure 13.20 illustrates an example of gradual speckle-type noise suppression in an echographic image. This figure depicts the results of processing with a 5×5 window for various values of tuning parameter K. These results show that the increase in K produces strong suppression of background noise or texture by raising the intensity levels of the image. Although, from Fig. 13.21, for $K = 3$ and a 3×3 window of analysis, denoising seems optimal, the fact remains that only the trained eye of the medical expert can make a decision as to the quality of the echographic image, based on the extracted and interpreted information.

13.5 Conclusion

In this chapter, denoising of medical images using heuristics based on multiresolution analysis has been presented, and some practical applications of wavelet domain denoising in ultrasound and in MRI were revisited.

The presented results demonstrate the usefulness of wavelet denoising for visual enhancement of images as well as for improving PSNR or SNR. Indeed, wavelet denoising methods confirmed their interest for noise suppression in ultrasound and in MRI images.

In the case of echography, the interactive noise reduction scheme, taking into account prior information as well as local regional statistics, led to a more natural ultrasound image, in which anatomical features were better kept intact.

In MRI, undecimated discrete wavelet transform as contourlet transform facilitates the edge features to be preserved better.

The proposed methods are adapted to medical image denoising since they account for the preference of the medical expert: a single parameter can be used to balance the preservation of (expert-dependent) relevant details against the degree of noise reduction.

These advanced heuristics are of low-complexity, both in their implementation and execution time. Moreover, they adapt themselves to unknown noise distributions and to the local spatial image context.

References

1. Grossman, P., Morlet, J.: Decomposition of Hardy functions into square integrable wavelets of constant shape. SIAM J. Math. Anal. **15**, 723–736 (1984)
2. Mallat, S.: Multiresolution approximations and wavelet orthogonal bases of L2(R). Trans. Am. Math. Soc. **315**, 69–87 (1989)
3. Mallat, S.: A Wavelet Tour of Signal Processing: The Sparse Way, 3rd edn. Academic Press, New York (2009)
4. Daubechies, I.: Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. **41**, 909–996 (1988)
5. Daubechies, I.: Ten Lectures on Wavelets. SIAM, Philadelphia (1992)
6. Burt, P., Adelson, E.: The Laplacian pyramid as a compact image code. IEEE Trans. Commun. **31**, 482–540 (1983)
7. Esteban, D., Galland, C.: Application of quadrature mirror filters to splitband voice coding schemes. In: IEEE International Conference of Acoustics, Signal and Speech Processing, Hartford, USA, pp. 191–195 (1977)
8. Unser, M.: Wavelet demystified. Ecole Multiresolution pour l'image, Lyon (2007)
9. Meyer, Y.: Ondelettes et Operateurs. Hermann, Paris (1990)
10. Ouahabi, A.: Introduction to multiresolution analysis. In: Ouahabi, A. (ed.) Signal and Image Multiresolution Analysis, pp. 1–133. Iste-Wiley, London (2012)
11. Coifman, R., Meyer, Y., Wickerhauser, V.: Wavelet analysis and signal processing. In: Ruskai, B. et al. (eds.) Wavelets and Their Applications, pp. 153–178. Jones and Barlett, Boston (1992)
12. Do, M.N., Vitterli, M.: The contourlet transform an efficient directional multiresolution image representation. IEEE Trans. Image Process. **14**, 2091–2106 (2005)

13. Donoho, D.L., Johnstone, I.M.: Ideal spatial adaptation by wavelet shrinkage. Biometrika **81**, 425–455 (1994)
14. Chesneau, C., Fadili, M.J., Starck, J.L.: Stein block thresholding for image denoising. Appl. Comput. Harmon. Anal. **28**, 67–88 (2009)
15. Dupe, F.X., Fadili, M.J., Starck, J.L.: A proximal iteration for deconvolving Poisson noisy images using sparse representations. IEEE Trans. Image Process. **18**, 310–321 (2009)
16. Atto, A.M., Pastor, D., Mercier, G.: Smooth sigmoid wavelet shrinkage for non-parametric estimation. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, Las Vegas (2008)
17. Pizurica, A., Philips, W.: Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising. IEEE Trans. Image Process. **15**, 654–665 (2006)
18. Goossens, B., Pizurica, A., Philips, W.: Removal of correlated noise by modeling the signal of interest in the wavelet domain. IEEE Trans. Image Process. **18**, 1153–1165 (2009)
19. Bartusek, K., Prinosil, J., Smekal, Z.: Wavelet-based de-noising techniques in MRI. Comput. Methods Programs Biomed. **104**, 480–488 (2011)

Chapter 14
Signal Separation with A Priori Knowledge Using Sparse Representation

Yu Guo and Su Ruan

Abstract This chapter presents a sparse representation method for single-channel signal separation with a priori knowledge. In this method, it is assumed that different source signals can be represented with different subsets of a dictionary constructed based on some a priori knowledge about these sources. Then, by estimating the sparse representation of the observed signal over this dictionary, we can finally recover the source signals. The two keys of this method are dictionary constructions and pursuit algorithms for finding sparse representations. An overview of commonly used schemes or algorithms for the two keys is given. In our work, this method is used to separate MRS data in order to achieve accurate MRS quantitation. Simulation results show the good performance of this method in separating the overlapping resonances and baseline. Quantitations of in vivo 1H MRS data of human brain tissues and prostate tissues demonstrate the effectiveness of this method.

14.1 Introduction

Signal separation, which consists of separating a set of signals from mixed signals, remains one of the most challenging and compelling problems in the domain of signal processing. In the literature, a lot of methods are proposed to solve this signal processing problem, such as all kinds of filter-based methods [13, 17], independent component analysis (ICA) [15, 19], empirical mode decomposition (EMD) [2, 9], sparsity-based methods [5, 21]. However, as all these methods are developed in some specific contexts and with certain constraints, none of them can be used to solve all signal separation problems.

Y. Guo (✉)
Department of Biomedical Engineering, Tianjin University, 92 Weijin Road, 300072 Tianjin, China
e-mail: guoyu@tju.edu.cn

S. Ruan
Laboratoire LITIS(EA 4108), Equipe Quantif, Université de Rouen, 22, Boulevard Gambetta, 76183 Rouen, France
e-mail: su.ruan@univ-rouen.fr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_14,

We focus here on the single-channel signal separation problem in which a single observed signal is available and it is composed by a linear combination of several source signals and a noise signal. For decomposing the observed signal into its subcomponents, two kinds of methods are proposed in the literature. In one kind of methods, these subcomponents can be modeled as some mathematical functions and then the signal separation problem can be converted to a parameter estimation problem. Some linear or nonlinear parameter estimation algorithms can be used to estimate these subcomponents. In the area of magnetic resonance spectroscopy (MRS) data analysis, this kind of methods is commonly used to divide an observed spectrum into several resonances associated respectively to different metabolites [10, 23]. In the other kind of methods, it is assumed that source signals are disjoint in the time domain, frequency domain or time–frequency domain, and then source separation can be achieved in one specified domain. For example, a frequency domain filtering method can separate sources which are in different frequency bands. However, the assumption that source signals are disjoint in certain domains is rather restrictive. The sources are usually non-disjoint in all these domains and even time–frequency filtering methods cannot separate them.

Recently, some researchers propose sparse representation-based methods which relax the disjoint condition by allowing the sources to be non-disjoint in the time–frequency domain, such as the methods proposed in [4–6, 12, 21]. In these methods, it is assumed that the sources can be sparsely represented over different dictionaries. Then they are finally recovered by estimating the sparse representation of the observed signal over these dictionaries. The construction of proper dictionaries, which is one of the keys for this kind of methods, is usually based on some a priori knowledge about the characters of source signals. For example, the method of morphological component analysis proposed in [21] separates textures from the piecewise smooth components by estimating the sparse representations of mixed images with respect to a wavelet dictionary for cartoon source images and Gabor dictionary for textures.

In this chapter, we introduce in detail how to achieve a single-channel signal separation based on a priori knowledge and using sparse representations. The chapter is organized as follows. We first introduce the basic scheme of this kind of signal separation methods. The two key steps are then presented: dictionary constructions and pursuit algorithms employed to find sparse representations. Finally, applications to separate MRS data are presented.

14.2 Signal Separation Using Sparse Representation

The research of sparse representation has attracted more and more interest in signal processing domain in recent years. It is widely used for denoising [4], signal separation [5, 21], direction-of-arrival estimation (DOA) [11], and so on. A basic signal representation model can be described as

$$\mathbf{y} = \mathbf{D}\mathbf{w} \tag{14.1}$$

where $\mathbf{y}$ is an $N \times 1$ signal vector, which can be represented as a linear combination of the columns (often called basis vectors or atoms) of a dictionary matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M]$, $\mathbf{w}$ is the representation coefficient vector. When $\mathbf{w}$ has only a small number of nonzero elements, this representation is called a sparse representation. Considering also that the signal vector $\mathbf{y}$ is the sum of K source vectors $\mathbf{s}_i$ $(i = 1, 2 \ldots, K)$, one has

$$\mathbf{y} = \sum_{i=1}^{K} \mathbf{s}_i . \tag{14.2}$$

To recover $\mathbf{s}_i$ $(i = 1, 2 \ldots, K)$ from $\mathbf{y}$, methods using sparse representations assume that these sources can be sparsely represented by different dictionaries $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M]$ which can be considered as the subsets of $\mathbf{D}$. Then, the sparse representation $\mathbf{w}$ of $\mathbf{y}$ over $\mathbf{D}$ is estimated and $\mathbf{s}_i$ is finally approximated by $\mathbf{w}_i \mathbf{D}_i$, where $\mathbf{w}_i$ is composed of the part of representation coefficients corresponding to $\mathbf{s}_i$.

The two keys of this kind of methods are the construction of proper dictionaries and the estimation of sparse representations over the designed dictionaries. Generally, dictionaries are designed according to the characters of source signals to be separated and are overcomplete, meaning that the number of atoms in a constructed dictionary is much bigger than the number of sample points of signal vectors ($M > N$). As a result, Eq. (14.1) is underdetermined and has infinitely many solutions. Theoretically, the sparsest representation which has the fewest nonzero elements is the solution of

$$\min_{\mathbf{w}} \quad \|\mathbf{w}\|_0 \quad \text{subject to} \quad \mathbf{y} = \mathbf{D}\mathbf{w}, \tag{14.3}$$

where $\|\cdot\|_0$ is the l_0-norm, to count the nonzero entries of a vector. The exact determination of the sparsest representation proves to be an NP-hard problem [7] which means that the time required to solve the problem using any currently known algorithm increases very quickly as the size of the problem grows. Thus, approximate solutions are considered instead. In the two following sections, we will describe in detail dictionary construction approaches and some pursuit algorithms commonly used for estimating sparse representations.

14.2.1 Dictionary Construction

Dictionaries can be constructed by either selecting one from a prespecified set of linear transforms or adapting the dictionary to a set of training signals.

Choosing a prespecified transform matrix is usually simpler and in many cases it also leads to simple and fast algorithms for the evaluation of the sparse representation. Short-time discrete Fourier transform (DFT), short-time discrete cosine transform (DCT), and discrete wavelet transform are three kinds of commonly used prespecified dictionary matrixes.

The decomposition into dictionaries associated to short-time DFT and DCT is conducive to the harmonic analysis of periodic signals. An overcomplete Fourier or cosine dictionary can be constructed by making a finer sampling of frequencies. Compared with DFT dictionaries, DCT dictionaries give decompositions in the real space and have the advantage of adapting perfectly to the algorithms developed in the context of sparse representation. The pure frequency analysis has some limitations. Although it allows detecting the dominant frequencies of a signal, it does not take into account its temporal characteristics. So DFT and DCT dictionaries are not suitable for representing a nonstationary or temporally discontinuous signal.

Time–frequency analysis of a signal is an interesting approach when the frequencies are varying. The transforms used for time–frequency analysis, such as wavelet transforms, can model a nonstationary phenomenon well over a very short duration. Therefore, dictionaries associated to time–frequency transforms are commonly constructed to represent nonstationary signals. Gabor dictionary is, for example, a kind of time–frequency dictionary and its atoms are created by sampling the following Gabor basis function: $a_{s,\tau,f}(t) = \frac{1}{s} w(\frac{t-\tau}{s}) e^{2i\pi f(t-\tau)}$, which can be described as a window function w modulated by an oscillating frequency. τ represents the temporal location of the window w, and s defines its width in the frequency domain. There is no constraint for the choice of the window function. In general, we choose the window functions so that they satisfy certain properties; one example is Hamming window.

Unlike the prespecified dictionaries, adaptive dictionaries are constructed from a set of given signals and based on learning methods, such as the dictionaries studied in [1]. The advantage of these dictionaries is that they have the adaptability to represent any signal. However, it needs enough data sources for training the desired dictionaries. In addition, the methods for constructing this kind of dictionaries usually suffer from low computational efficiency. Choosing a dictionary method depends on the application. It is therefore difficult to say which method is the best. Once the dictionary is built, let us now present solutions to obtain the sparsest representation of a given signal.

14.2.2 Pursuit Algorithms

In the past decade or so, several efficient pursuit algorithms have been proposed to approximate the solution in Eq. (14.1). Greedy algorithms and algorithms based on convex relaxations are two kinds of pursuit algorithms commonly used to find a sparse representation.

14.2.2.1 Greedy Algorithms

For the problem of sparse representation, dictionaries can be supposed to be redundant, which means signals can be well represented with only a small number of basis

functions in the dictionaries. Greedy algorithms estimate sparse representations in two steps. The first step consists of the selection of basis functions with nonzero representation coefficients and the second step is to estimate these nonzero coefficients. However, it is impossible to select all the expected basis functions at the same time. The greedy algorithms do the selection in an iterative manner.

Let Γ be the set of basis functions corresponding to nonzero representation coefficients. It is initialized as an empty set $\Gamma^0 = \{\phi\}$. In the kth iteration, a single basis function in the complement of Γ^k is added into Γ^k. In this way, Γ^k can be updated in each iteration. The difference between the algorithms using greedy strategies is in how they select a new basis function to update Γ^k.

(A) Matching pursuit

The matching pursuit (MP) algorithm proposed in [16] sequentially selects the basis functions by involving the computation of inner products between the signal and basis functions. In each iteration, matching pursuit calculates a new signal approximation. The approximation error is then used in the next iteration to determine which new basis functions are to be selected. Let $\mathbf{y}$ be the input signal vector, $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M]$ be the normalized dictionary matrix, and $\mathbf{d}_i$ be a basis vector of $\mathbf{D}$. Then, the MP algorithm can be summarized as:

Initialize: $\hat{\mathbf{y}}^0 = 0, \mathbf{r}^0 = \mathbf{y}, \Gamma^0 = \{\phi\}, \omega^0 = [], k = 1$. If $\|\mathbf{r}^k\|^2 \geq \xi$, repeat
– compute $\alpha_i = \langle \mathbf{r}^{k-1}, \mathbf{d}_i \rangle$,
– find $i_{\max} = \arg_i \max \alpha_i$,
– update:
1. $\Gamma^k = [\Gamma^k, \mathbf{d}_{i_{\max}}]$;
2. $\omega^k = [\omega^{k-1}, \alpha_{i_{\max}}]^{\mathrm{T}}$;
3. $\hat{\mathbf{y}}^k = \hat{\mathbf{y}}^{k-1} + \alpha_{i_{\max}} \mathbf{d}_{i_{\max}}$;
4. $\mathbf{r}^k = \mathbf{r}^{k-1} - \alpha_{i_{\max}} \mathbf{d}_{i_{\max}}$;
5. $k = k + 1$.

The proof of convergence of the MP algorithm relies essentially on the fact that $\langle \mathbf{r}^k, \mathbf{d}_{i_{\max}} \rangle = 0$. For the MP algorithm, there is no need to compute any inverse matrix, so it is very simple to be implemented. However, it has also the shortcoming that although asymptotic convergence is guaranteed, the resulting approximation after any finite number of iterations will in general be suboptimal and the approximation error may still be quite large, especially for a nonorthogonal dictionary.

(B) Orthogonal least squares

Another commonly used greedy algorithm is the orthogonal least squares (OLS) algorithm [3]. In the implementation of OLS, it selects a new basis function that will lead to the minimum residual error after orthogonalization. The selection procedure of OLS can be written as:

Initialize: $\mathbf{r}^0 = \mathbf{y}, \Gamma^0 = [], k = 0$. If $\|\mathbf{r}^k\|^2 \geq \xi$, repeat
– find $i_{\max} = \arg_i \min \|\Gamma_i'(\Gamma_i')^{\dagger}\mathbf{y} - \mathbf{y}\|^2$ where $\Gamma_i' = \Gamma^{k-1} \cup \{\mathbf{d}_i\}$ for all $\mathbf{d}_i \notin \Gamma^{k-1}$,
– update
1. $\Gamma^k = [\Gamma^{k-1}, \mathbf{d}_{i_{\max}}]$;

2. $\mathbf{w}^k = (\Gamma^k)^\dagger \mathbf{y}$;
3. $\mathbf{r}^k = \mathbf{y} - \Gamma^k \mathbf{w}^k$;
4. $k = k + 1$.

These greedy algorithms (MP and OLS) are proposed to build up iteratively a signal representation by selecting the atom that maximally improves the representation at each iteration. They are easily implemented, converge quickly, and have good approximation properties. However, there is no guarantee that they compute sparse representations. Only under some conditions, they can be used to compute sparse (or nearly sparse) representations [22]. A drawback of these algorithms applied to sparse representation is their greediness. It is possible to construct signal representation problems where, because of the greediness, an atom that is not part of the optimal sparse representation is selected; as a result, many of the subsequent atoms selected simply compensate for the poor initial selection [14]. This shortcoming motivated the development of basis pursuit algorithm, which succeeds on these problems.

14.2.2.2 Algorithms Based on Norm Minimization

As searching for the minimum l_0-norm is an intractable problem, the researchers consider using methods based on other norm minimizations to find approximate solutions to Eq. (14.3). The l_1-norm minimization and iteratively reweighted norm minimization are two commonly used alternatives to the l_0-norm minimization.

(A) Basis pursuit

The principle of basis pursuit (BP) proposed in [4] is to find signal representations whose coefficients have minimal l_1-norm. The resulting representations are sparse in the l_0-norm sense under certain conditions [8]. Formally, BP solves the following problem

$$\min_{\mathbf{w}} \quad \|\mathbf{w}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{D}\mathbf{w}. \tag{14.4}$$

The basis pursuit problem of Eq. (14.4) can be equivalently reformulated as a linear program (LP) which is the problem of finding a vector $\mathbf{q}$ that minimizes a linear function $\mathbf{f}^{\mathrm{T}}\mathbf{q}$ subject to linear constraints such that one or more of the following hold: $\mathbf{A}\mathbf{q} < \mathbf{b}$, $\mathbf{A}_{\mathrm{eq}}.\mathbf{q} = \mathbf{b}_{\mathrm{eq}}$, $l \le q \le u$. A tremendous amount of work has been done on the solution of linear programs. For solving a BP optimization problem, the algorithm from the LP literature can be considered as a candidate. In [4], the BP-simplex and BP-interior, which are respectively based on the simplex and interior-point algorithms, were described to solve a BP optimization problem.

In [4], the BP is also adapted to the case of noisy data. Take data of the form

$$\mathbf{x} = \mathbf{y} + \mathbf{e} \tag{14.5}$$

into consideration, where $\mathbf{x}$ is the observed signal vector, $\mathbf{y}$ is the clean signal vector, and $\mathbf{e}$ is a Gaussian noise. For finding the sparse representation of the clean signal

y, the BP principle of Eq. (14.4) cannot be directly used. The alternative principle Basis Pursuit Denoising (BPDN) is proposed. It refers to a solution of

$$\min_{\mathbf{w}} \quad \frac{1}{2}\|\mathbf{x}-\mathbf{D}\mathbf{w}\|_2^2+\lambda\|\mathbf{w}\|_1. \tag{14.6}$$

The minimization is composed of two parts: the residual and the l_1-norm of the representation vector. The parameter λ decides which part plays a more important role in the minimization. The choice of λ controls the trade-off between the quality of fitting and the degree of sparsity. A large value of λ leads to sparser solutions, and small value leads to a better fit. The BPDN of Eq. (14.6) is equivalent to a quadratic programming problem. Thus, the algorithms for a quadratic programming problem can also be used to solve the BPDN problem.

(B) FOCUSS (FOCal Undetermined System Solver)

The FOCUSS algorithm proposed in [11] relies on the principle of iteratively reweighted least squares minimization (IRLS). The diversity measure $\mathrm{E}^{(p)}(\mathbf{w})$ defined in Eq. (14.7) is used to replace the l_0-norm and ensure the sparsity of a solution.

$$\mathrm{E}^{(p)}(\mathbf{w})=\sum_{i=1}^{M}\mathrm{sgn}(p)\big|w(i)\big|^p. \tag{14.7}$$

The basic FOCUSS algorithm produces a sparse solution

$$\mathbf{w}_{k+1}=\mathbf{W}_{k+1}(\mathbf{D}\mathbf{W}_{k+1})^{\dagger}\mathbf{y}, \tag{14.8}$$

where $\mathbf{W}_{k+1}=\mathrm{diag}(|w_k(1)|^{1-(p/2)},\ldots,|w_k(M)|^{1-(p/2)})$ is the weighting matrix. To deal with noise in the measurements, a Bayesian framework is used in [18]. $\mathbf{w}$ is estimated by using a maximum a posteriori (MAP) estimator defined as follows:

$$\mathbf{w}=\arg\min_{\mathbf{w}} J(\mathbf{w}) \quad \text{where } J(\mathbf{w})=\left[\|\mathbf{D}\mathbf{w}-\mathbf{x}\|^2+\gamma\mathrm{E}^{(p)}(\mathbf{w})\right]. \tag{14.9}$$

The parameter γ controls the trade-off between the quality of fitting $\mathbf{x}$ and the degree of sparsity. A large value of γ leads to sparser solutions, and a small value leads to a better fit. The iterative algorithm derived to find the solution to Eq. (14.8) is as follows:

$$\mathbf{w}_{k+1}=\mathbf{W}_{k+1}\mathbf{D}_{k+1}^{\mathrm{T}}\left(\mathbf{D}_{k+1}\mathbf{D}_{k+1}^{\mathrm{T}}+\lambda\mathbf{I}\right)^{-1}\mathbf{x} \tag{14.10}$$

where $\mathbf{D}_{k+1}=\mathbf{D}\mathbf{W}_{k+1}$. The parameter λ is proportional to γ and should increase with the level of noise. The details about the choice of λ can be found in [18].

14.3 MRS Spectra Separation with A Priori Knowledge Using Sparse Representation

In the domain of in vivo MRS study, an observed MRS spectrum is usually the combination of several resonances corresponding respectively to different metabolites,

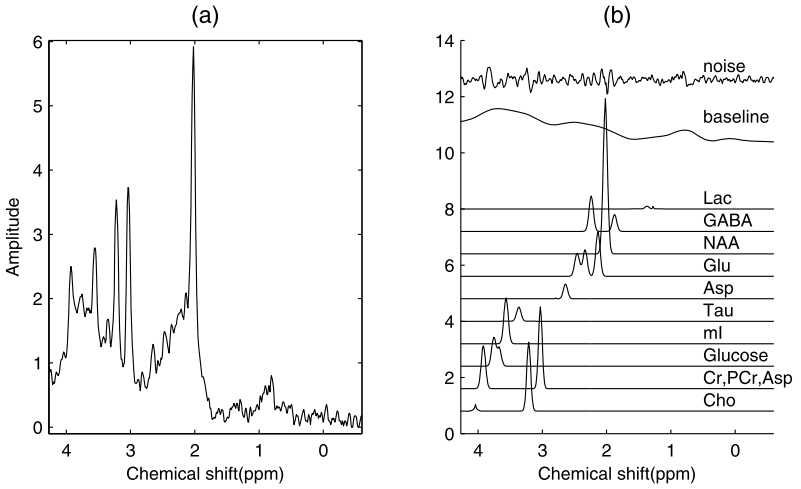


Fig. 14.1 Observed 1H MRS spectra (TE = 35 ms) of normal human brain tissue (**a**) which is composed of several resonances associated to different metabolites of interest, a baseline component and a noise (**b**)

a baseline and a noise, as shown in Fig. 14.1. It is necessary to recover these resonances for accurately quantitating the corresponding metabolites. In our previous work, we have proposed a method using sparse representation and wavelet filter for separating different components in observed MRS spectra. Here, we introduce this method for specifying signal separation with a priori knowledge using sparse representation.

14.3.1 Signal Models

Generally, an observed MRS spectrum **x** can be modeled as

$$\mathbf{x} = \mathbf{S} + \mathbf{B} + \mathbf{e} = \sum_{k=1}^{K} \mathbf{s}_k + \mathbf{B} + \mathbf{e}, \tag{14.11}$$

where **S** represents the mixed spectrum of interest which is the linear combination of several resonances $\mathbf{s}_k$ $(k = 1, \ldots, K)$, **B** a baseline contribution, **e** a Gaussian noise. Each resonance can be modeled as a lineshape. A Lorentzian lineshape in Eq. (14.12), or a Gaussian lineshape in Eq. (14.13), or a combination of Lorentzian and Gaussian lineshapes is usually used:

$$L_k(f) = \frac{a_k}{1 + [(f - f_k)/d_k]^2}, \tag{14.12}$$

$$G_k(f) = a_k \exp\left[-\left(\frac{f - f_k}{d_k}\right)^2\right], \tag{14.13}$$

where f is the frequency of each data point, a_k is the amplitude, d_k is the linewidth, and f_k is the peak frequency of the resonance $\mathbf{s}_k$. With the above signal model, each resonance is characterized by its linear parameters a_k and its nonlinear parameters f_k and d_k.

For recovering the resonances of interest in an observed MRS spectrum, the method using sparse representation [12] firstly constructs a dictionary, which is composed of several subdictionaries. Each subdictionary can only represent sparsely one of these resonances of interest. Then, a pursuit algorithm is used to estimate the sparse representation of the observed spectrum over the constructed dictionary, based on which the representation coefficients of each resonance over the corresponding subdictionary can be computed.

14.3.2 Dictionary Construction Based on the A Priori Knowledge

The basis of signal separation methods using sparse representation is that source signals can be sparsely represented by different dictionaries. Due to the fact that the form of each resonance can be uniquely characterized by its lineshape model and nonlinear parameters, we construct a dictionary which is composed of a set of normalized Gaussian and Lorentzian functions, as shown in Eq. (14.12) and Eq. (14.13). The a priori knowledge about peak frequencies and the range of possible linewidths of each resonance is used to fix parameters of these basis functions. In this way, the constructed dictionary will contain as few basis functions as possible while still representing the resonances of interest very well.

For each basis function in the dictionary, its central frequency is set as the known peak frequency of a certain resonance f_k $(k = 1, \ldots, K)$. For the basis functions with the same central frequency, their linewidths d_k change in a given range with a certain sampling interval. According to central frequencies, the dictionary is divided into K groups $\{\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_k, \ldots, \mathbf{D}_K\}$, where K is the number of possible resonances in a mixed MRS spectrum. The basis functions $\{\mathbf{d}_{k1}, \ldots, \mathbf{d}_{kj}, \ldots, \mathbf{d}_{kL}\}$ in the group $\mathbf{D}_k$ have the same central frequency f_k and different linewidths denoted as $d_{kj} = d_{k1} + (j-1)\Delta d_k$ $(j = 1, \ldots, L)$, where L is the number of basis functions in this group, $[d_{k1}, d_{kL}]$ is the range of possible linewidths of the resonance $\mathbf{s}_k$ and Δd_k is the sampling interval.

Different groups can have different numbers of basis functions. Here for convenience, we assume the same number of basis functions in different groups. When the range of the possible linewidths of the resonance $\mathbf{s}_k$ is determined, the choice of Δd_k will decide the number of basis functions L in the group $\mathbf{D}_k$. The relatively small value of Δd_k will lead to more accurate representations of resonances, but stronger correlations between basis functions, which will make it difficult to accurately estimate sparse representation in the next step. For the choice of the sample step Δd_k, there has to be a compromise between robustness and precision.

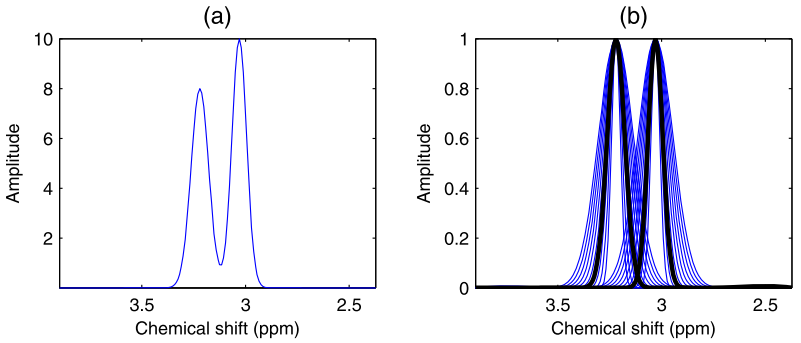


Fig. 14.2 Illustration of dictionary construction: (**a**) a simulated MRS spectrum with two peaks (simulated with Gaussian functions); (**b**) normalized Gaussian basis functions in the corresponding dictionary (the *black lines* represent the best approximations of the two spectral peaks in (**a**))

Figure 14.2 shows a simulated MRS spectrum with three spectrum peaks and the corresponding basis functions in a dictionary to represent the spectrum. The black lines in Fig. 14.2(b) correspond to the basis functions which can best approximate the two peaks in Fig. 14.2(a). The objective of our method is to find the two basis functions.

The constructed dictionary is denoted as a matrix with M basis vectors $[\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_i, \ldots, \mathbf{d}_M]$, where $M = K \times L$. If the sparse representation vector of the mixed spectrum $\mathbf{S}$ is denoted as $\mathbf{w}$, then

$$\mathbf{S} = \mathbf{D}\mathbf{w} = \sum_{i=1}^{M} \mathbf{d}_i w_i = \sum_{k=1}^{K} \mathbf{D}_k \mathbf{w}_k, \tag{14.14}$$

where $\mathbf{w} = [{\mathbf{w}_1}^{\mathrm{T}}, {\mathbf{w}_2}^{\mathrm{T}}, \ldots, {\mathbf{w}_k}^{\mathrm{T}}, \ldots {\mathbf{w}_K}^{\mathrm{T}}]^{\mathrm{T}}$. Theoretically, only the basis functions which best approximate the resonances of interest correspond to nonzero representation coefficients. In the case, where different resonances have different peak frequencies, the basis vectors in group $\mathbf{D}_k$ can only represent the resonance $\mathbf{s}_k$. Therefore, $\mathbf{s}_k = \mathbf{D}_k \mathbf{w}_k$. Let $\mathbf{w}^{\wedge}$ ($[{\mathbf{w}_1}^{\wedge\mathrm{T}}, {\mathbf{w}_2}^{\wedge\mathrm{T}}, \ldots, {\mathbf{w}_k}^{\wedge\mathrm{T}}, \ldots, {\mathbf{w}_K}^{\wedge\mathrm{T}}]^{\mathrm{T}}$) denote the estimated sparse representation of a mixed spectrum, the resonance $\mathbf{s}_k$ can then be estimated as $\mathbf{s}_k{}^{\wedge} = \mathbf{D}_k {\mathbf{w}_k}^{\wedge}$.

However, because of the presence of baseline component in an observed spectrum, the mixed spectrum $\mathbf{S}$ is unavailable. For dealing with this problem, a strategy using a wavelet filter is exploited. In the frequency domain, baselines are commonly assumed to be smooth and broad compared to the resonance signals. Therefore, a wavelet filter is used to remove the smooth components of an observed spectrum. Because of the overlapping of the baseline and the resonances of interest, the removed components contain not only the baseline, but also a portion of the useful signal. The signal remaining after the filtering consists of only a component of mixed resonances of interest which does not overlap with the baseline. Our idea is to carry out the estimation of sparse representation on the remaining signal to finally reconstruct the resonances of interest in their entirety.

A wavelet filter is denoted as $g(\bullet)$. $\mathbf{x}_h = g(\mathbf{x})$ is the result of filtering an observed spectrum $\mathbf{x}$ with $g(\bullet)$. $\mathbf{x}_l = \mathbf{x} - \mathbf{x}_h$ is the smooth component removed by the filter. Because of the overlapping of $\mathbf{B}$ and $\mathbf{s}_k$ $(k = 1, \ldots, K)$, when the baseline $\mathbf{B}$ is eliminated completely by the wavelet filter, the smooth and broad components of metabolite spectra will also be lost at the same time. This can be denoted as: $\mathbf{x}_h \approx \mathbf{S}_h$ and $\mathbf{x}_l \approx \mathbf{B} + \mathbf{S}_l$, where $\mathbf{S}_h = g(\mathbf{S})$ and $\mathbf{S}_l = \mathbf{S} - \mathbf{S}_h$. To represent $\mathbf{S}_h$, all the basis vectors $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M\}$ in the dictionary $\mathbf{D}$ analyzed above are also processed by the same wavelet filter. A new dictionary $\mathbf{D}_h$ using the remaining components $g(\mathbf{d}_i)$ $(i = 1, \ldots, M)$ is then constructed. The signal $\mathbf{x}_h$ remaining after filtering can be written as:

$$\mathbf{x}_h = \mathbf{S}_h + \xi_{\mathbf{B}} = \mathbf{D}_h\mathbf{w} + \xi_{\mathbf{B}} \approx \mathbf{D}_h\mathbf{w}, \tag{14.15}$$

where $\xi_{\mathbf{B}}$ is the remaining component of the baseline after wavelet filtering, which should be as small as possible. Finally, the representation coefficient vector $\mathbf{w}$ of mixed resonances $\mathbf{S}$ with respect to the dictionary matrix $\mathbf{D}$ in Eq. (14.14) can be estimated by computing the sparsest solution of Eq. (14.15).

14.3.3 Resonance Estimation with FOCUSS Algorithm

For estimating the sparse representation in Eq. (14.15), an algorithm based on FOCUSS algorithm was developed with the consideration of the following particularities in this application. Firstly, basis functions in the same group have the same central frequency, so strong correlations exist between them. Pursuit algorithms such as greedy pursuit and basis pursuit, which have severe restriction on the correlations between basis functions, perform poorly here. Secondly, the expected representation coefficients are nonnegative, and the non-negativity constraint can be added.

With the non-negativity constraint, the optimization function of regularized FOCUSS algorithm can be modified as

$$\mathbf{w} = \arg\min_{\mathbf{w}} J(\mathbf{w}), \quad \text{where } J(\mathbf{w}) = \left[\|\mathbf{D}_h\mathbf{w} - \mathbf{x}_h\|^2 + \gamma \mathrm{E}^{(p)}(\mathbf{w})\right] \text{ and } \forall i : w_i \geq 0. \tag{14.16}$$

By reference to the iterative form of regularized FOCUSS algorithm, the iterative form of the optimization in (14.16) is then developed as follows:

(a) $\mathbf{w}_{k+1} = \mathbf{W}_{k+1}\mathbf{D}_{k+1}{}^{\mathrm{T}}(\mathbf{D}_{k+1}\mathbf{D}_{k+1}{}^{\mathrm{T}} + \lambda\mathbf{I})^{-1}\mathbf{x}_h$;

(b) $w_{k+1}(i) = \begin{cases} 0 & \text{if } w_{k+1}(i) < 0, \\ w_{k+1}(i) & \text{if } w_{k+1}(i) \geq 0, \end{cases}$

where $\mathbf{W}_{k+1} = \mathrm{diag}(|w_k(1)|^{1-(p/2)}, \ldots, |w_k(M)|^{1-(p/2)})$ and $\mathbf{D}_{k+1} = \mathbf{D}_h\mathbf{W}_{k+1}$. At each iteration step, the negative values of the updated solution $\mathbf{w}_{k+1}$ are set to zero to ensure the non-negativity of $\mathbf{w}$. Actually, the non-negativity constraint also increases the sparsity of a solution to a certain degree. The regularization parameter λ is the function of the level of noise. In [24], three different criteria for choosing λ are investigated. They are (i) quality of fit; (ii) a sparsity criterion; (iii) L-curve.

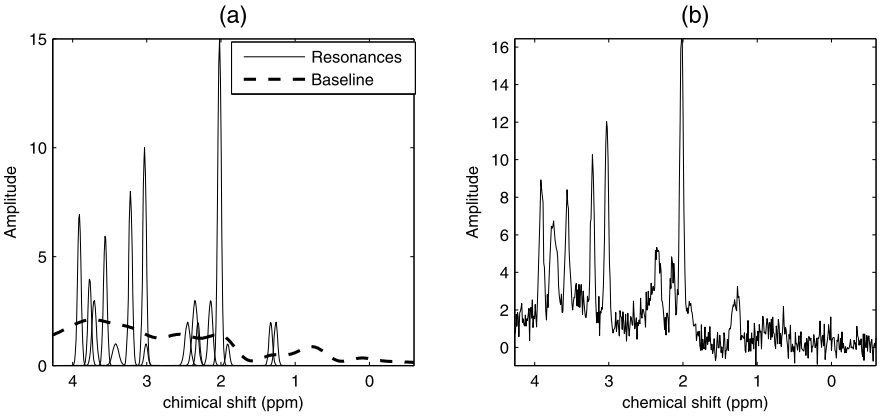


Fig. 14.3 Simulation of 1H human brain MRS spectra: (**a**) simulated resonances (*solid lines*) and simulated baseline (*dashed line*); (**b**) simulated observed spectrum with SNR = 30 dB and SBR = 10 dB

Here, for the consideration of computational efficiency, λ is set as $C\varepsilon^2$, where ε is the estimated noise power and C is a constant chosen by a thorough analysis of simulated data.

14.3.4 Experiments and Results

In this section, this method is firstly used to process some simulated 1H human brain MRS data and then its performance is compared with a commonly used nonlinear parameter estimation method. Finally, some processing results of real clinical MRS data are presented to illustrate the good performance of this method.

14.3.4.1 Simulated Experiments

(A) Simulated data

Each simulated 1H human brain MRS spectrum used here has 512 data points and consists of 16 resonances, a baseline, and a Gaussian noise as shown in Fig. 14.3. The resonances are simulated as Gaussian functions, whose parameters are summarized in Table 14.1. The baseline is obtained from a similar baseline of a true 1H human brain MRS spectrum. Spectra are with different signal-to-noise ratios (SNR = 25, 30, 35, and 40 dB) and signal-to-baseline ratios (SBR = 5, 10, 15, and 20 dB). Here, SNR is defined as the ratio of the highest amplitude of these simulated resonances to the noise standard deviation, and SBR is the power ratio of a mixed spectrum of interest to the simulated baseline. For a given baseline and noise condition, a set of 100 spectra is generated in order to give reliable estimation results

Table 14.1 Parameters of the simulated 1H MRS spectra

Metabolite	kth	f_k (ppm)	d_k (ppm)	a_k
Cr	1	3.91	0.03	7.00
Glu/Gln1	2	3.77	0.03	4.00
Glu/Gln2	3	3.71	0.04	3.00
mI	4	3.56	0.03	6.00
Tau	5	3.42	0.06	1.00
Cho	6	3.22	0.03	8.00
Cr/PCr	7	3.03	0.03	10.00
GABA1	8	3.01	0.03	1.00
GABA2	9	2.31	0.03	2.00
GABA3	10	1.91	0.03	1.00
Glu/Gln3	11	2.45	0.05	2.00
Glu/Gln4	12	2.35	0.05	3.00
Glu/Gln5	13	2.14	0.04	3.00
NAA	14	2.02	0.03	15.00
Lac1	15	1.33	0.03	2.00
Lac2	16	1.26	0.03	2.00

and to check the robustness of different methods. Figure 14.3(b) shows a simulated observed spectrum with SNR = 30 dB and SBR = 10 dB.

In simulation experiments, the central frequencies of these resonances (f_k ($k = 1, \ldots, 16$) listed in Table 14.1) and the range of their linewidths ($0.01 \leq d \leq 0.10$) are exploited as a priori knowledge to construct a dictionary. Consequently, the dictionary is composed of 320 basis functions and all 20 basis functions have the same central frequency as one of the simulated resonances and different linewidths taken from the range $0.01 \leq d \leq 0.10$ (ppm) with a sample step $\Delta d = 0.005$ (ppm).

The wavelet filter used to remove baselines performs a 5-level wavelet decomposition (Coiflet wavelet COIF5) of the observed MRS spectra. The detail coefficients of wavelet decompositions are retained to construct $\mathbf{x}_h$. The same wavelet decomposition and reconstruction of each basis vector in $\mathbf{D}$ is computed to construct a new dictionary $\mathbf{D}_h$. In simulation experiments, the regularization parameter λ of the proposed pursuit algorithm is set as $C\varepsilon^2$ and $C = 0.4$.

(B) Results on simulated data

We analyzed the quantitation performance of the proposed method under different noise and baseline conditions. The RRMSEs (Relative Root Mean Square Errors) of estimated peak amplitudes were calculated. RRMSE is defined as the ratio of RMSE (Root Mean Square Error) of estimated results to the real values. Figure 14.4 shows the results under a given baseline condition (SBR = 10 dB) and different noise conditions (SNR = 20, 25, 30, and 35 dB). It can be seen that when the level of noise increases, the estimation results deteriorate as with most other MRS quantitation methods and the noise has more important influence on the quantitation of

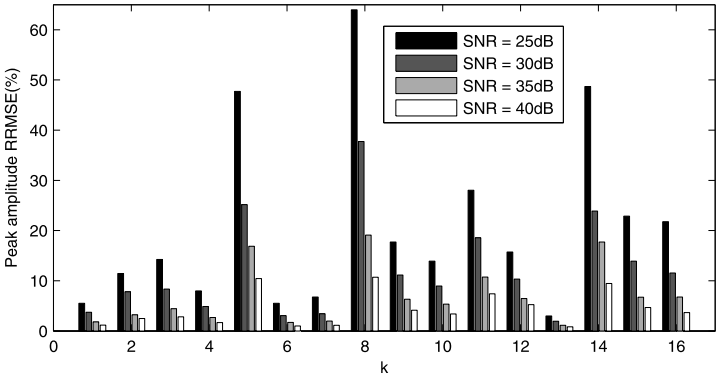


Fig. 14.4 Statistic quantitation results for the simulated spectra (SBR = 10 dB and SNR = 25, 30, 35, and 40 dB) with the proposed method

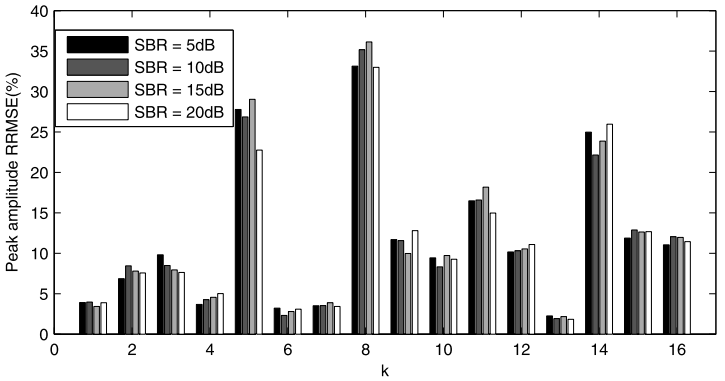


Fig. 14.5 Statistic quantitation results for the simulated spectra (SNR = 30 dB and SBR = 5, 10, 15, and 20 dB) with the proposed method

small peaks than on that of large peaks. Figure 14.5 shows the results under a given noise condition (SNR = 30 dB) and different baseline conditions (SBR = 5, 10, 15, and 20 dB). Because most of baseline components have been filtered by the wavelet filter and the few residual baseline components limit the influence of the amplitude change of baselines on the final estimation accuracy, the estimation results of the proposed method change little with the increase in baseline amplitude.

The proposed method was also compared with another frequency-domain MRS quantitation method proposed in [10] (called nonlinear method here). This method uses Levenberg–Marquardt algorithm to estimate the nonlinear model parameters of metabolite spectra and a wavelet filter to remove the baseline component in an iterative subtraction manner. The quantitation results of the two methods in 100 simulation experiments (SNR = 30 dB and SBR = 10 dB) are shown in Fig. 14.6. As observed in Fig. 14.6, the proposed method has better quantitation accuracy com-

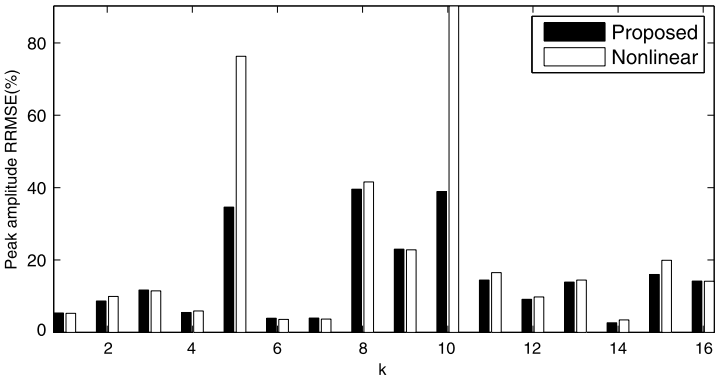


Fig. 14.6 Statistic quantitation results for the simulated spectra (SNR = 30 dB and SBR = 10 dB) with two methods: the proposed method (FOCUSS) and the nonlinear method (Nonlinear)

pared with the nonlinear method. For the metabolites with weak MRS peaks, such as the metabolites Tau ($k = 5$) and GABA ($k = 10$), the proposed method can provide much better quantitations than the nonlinear method in [10].

14.3.4.2 Quantitation of Human Brain 1H MRS Data

This method was applied to quantitate in vivo brain 1H MRS data of patients with brain lesions. MRS spectra were recorded for these patients at 1.5 T with PRESS sequence and with an echo-time of 35 ms in two opposite regions in their brains. The two regions chosen by the doctor correspond to one region with tumor (zone 1) and one presumed normal brain region (zone 2) in the contra lateral hemisphere, respectively. Figure 14.7 shows two observed spectra of a patient recorded at the same time from zone 1 and zone 2, respectively, as well as the corresponding spectral separation results. As shown in Fig. 14.7, the resonances of interest and the baseline in an observed spectrum overlap seriously with each other. It is difficult to directly achieve the accurate quantitation of the resonances of interest (computing their peak amplitudes or areas), which can be used to measure the chemical compositions of human tissues and further assist the diagnosis and treatment of diseases. Thus, the procedure of separating the overlapping components in observed MRS is important for the analysis of clinical MRS data.

14.3.4.3 Quantitation of Prostate 1H MRS Data

This method was also applied for the quantitation of in vivo prostate 1H MRS data. Three-dimensional MR spectroscopic imaging data (3D-MRSI) were acquired at 3 T with PRESS sequence and with an echo-time of 140 ms. Figure 14.8 shows two prostate 1H MRS spectra from a 3D-MRSI exam of a patient with prostate cancer

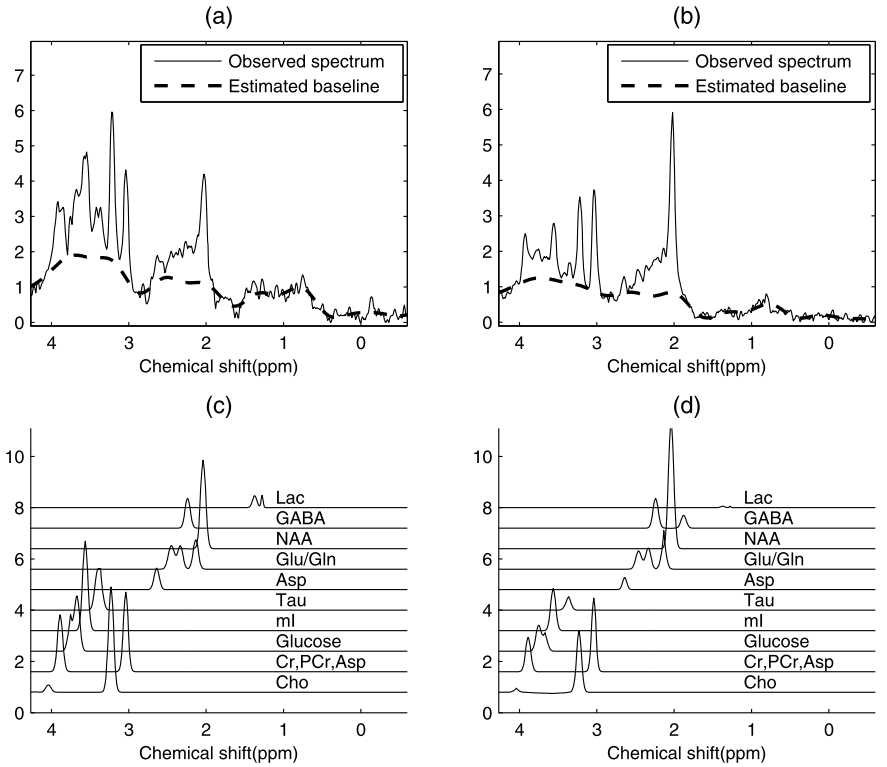


Fig. 14.7 Separation results of in vivo human brain 1H MRS spectra: (**a**) the observed spectrum from tumor tissue (zone 1) and the estimated baseline (*dashed line*); (**b**) the observed spectrum from normal tissue in the contralateral hemisphere (zone 2) and the estimated baseline (*dashed line*); (**c**) the estimated resonances with the spectrum in (**a**); (**d**) the estimated resonances with the spectrum in (**b**)

and the corresponding estimation results of the two spectra. The peak area ratios of (choline+creatine+polyamine) to citrate (Cho+Cr+PA/Cit) were calculated as final quantitation results. For the spectra in Fig. 14.8(a) and Fig. 14.8(b), the values of (Cho+Cr+PA/Cit) are respectively 1.18 and 0.39. As a published study [20] has shown that cancerous prostate tissues associate to relatively high (Cho+Cr+PA/Cit) value, we can conclude that the spectrum in Fig. 14.8(a) corresponds to cancerous tissue. The conclusion accords with the final biopsy diagnosis.

14.4 Summary

Single channel signal separation with a priori knowledge using sparse representation can relax in a certain degree the disjoint condition concerning source signals in the time–frequency domain. The basis of this kind of methods is decomposing the

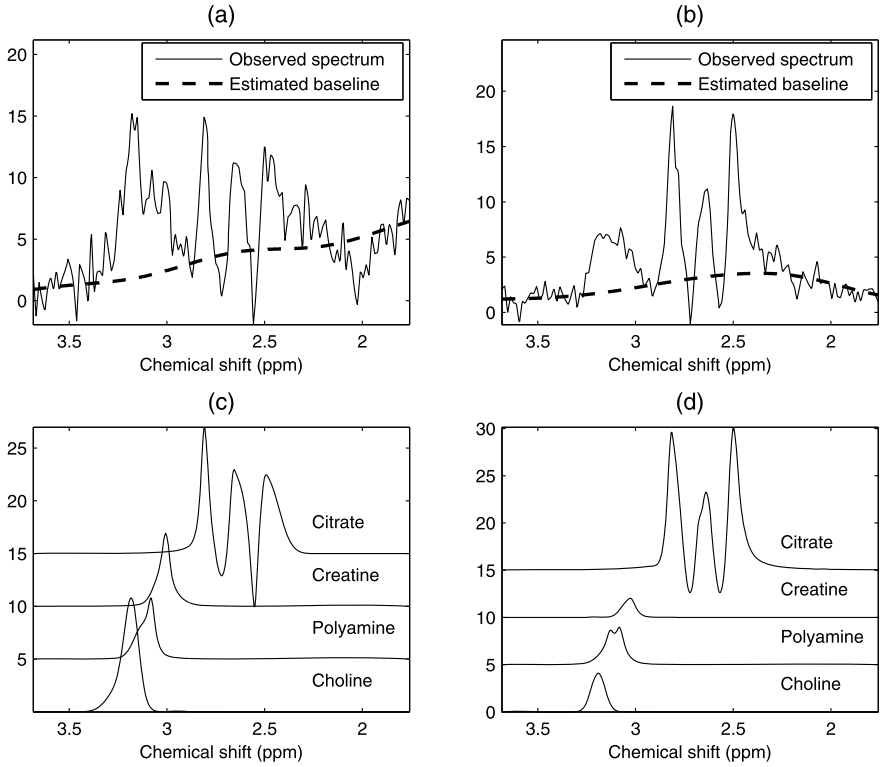


Fig. 14.8 Separation results of in vivo prostate 1H MRS spectra: (**a**) an observed spectrum from cancerous tissue (*solid line*) and the estimated baseline (*dashed line*); (**b**) an observed spectrum from normal tissue (*solid line*) and the estimated baseline (*dashed line*); (**c**) the estimated resonances with the spectrum in (**a**); (**d**) the estimated resonances with the spectrum in (**b**)

observed signal into different dictionaries, which can only sparsely represent one of the source signals. The a priori knowledge about the features of source signals can be used to construct these dictionaries. For example, in the application to analyze MRS data, the mathematical model of source signals (resonances of interest) and the range of model parameters are exploited for the dictionary construction. As for the procedure of sparse decomposition, many pursuit algorithms in the literature are available. However, these algorithms could have different performances on a specific application. Furthermore, researchers sometimes should develop existing algorithms according to the specific application for achieving a satisfying separation result. Additional constraints could lead to a better separation performance.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. **54**(11), 4311–4322 (2006)

2. Bouchikhi, A., Boudraa, A.: Multicomponent AM–FM signals analysis based on EMD–B-splines ESA. Signal Process. (2012)
3. Chen, S., Billings, S., Luo, W.: Orthogonal least squares methods and their application to non-linear system identification. Int. J. Control **50**(5), 1873–1896 (1989)
4. Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. SIAM Rev., 129–159 (2001)
5. Cho, N., Kuo, C.: Sparse music representation with source-specific dictionaries and its application to signal separation. IEEE Trans. Audio Speech Lang. Process. **19**(2), 326–337 (2011)
6. Davies, M., James, C.: Source separation using single channel ICA. Signal Process. **87**(8), 1819–1832 (2007)
7. Davis, G., Mallat, S., Avellaneda, M.: Adaptive greedy approximations. Constr. Approx. **13**(1), 57–98 (1997)
8. Donoho, D., Huo, X.: Uncertainty principles and ideal atomic decomposition. IEEE Trans. Inf. Theory **47**(7), 2845–2862 (2001)
9. Gao, B., Woo, W., Dlay, S.: Single channel source separation using EMD-subband variable regularized sparse features. IEEE Trans. Audio Speech Lang. Process. **99**, 1 (2011)
10. Gillies, P., Marshall, I., Asplund, M., Winkler, P., Higinbotham, J.: Quantification of mrs data in the frequency domain using a wavelet filter, an approximated Voigt lineshape model and prior knowledge. NMR Biomed. **19**(5), 617–626 (2006)
11. Gorodnitsky, I., Rao, B.: Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm. IEEE Trans. Signal Process. **45**(3), 600–616 (1997)
12. Guo, Y., Ruan, S., Landre, J., Constans, J.: A sparse representation method for magnetic resonance spectroscopy quantification. IEEE Trans. Biomed. Eng. **57**(7), 1620–1627 (2010)
13. Hopgood, J., Rayner, P.: Single channel nonstationary stochastic signal separation using linear time-varying filters. IEEE Trans. Signal Process. **51**(7), 1739–1752 (2003)
14. Huggins, P., Zucker, S.: Greedy basis pursuit. IEEE Trans. Signal Process. **55**(7), 3760–3772 (2007)
15. Hyvarinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999)
16. Mallat, S., Zhang, Z.: Matching pursuits with time–frequency dictionaries. IEEE Trans. Signal Process. **41**(12), 3397–3415 (1993)
17. Radfar, M., Dansereau, R.: Single-channel speech separation using soft mask filtering. IEEE Trans. Audio Speech Lang. Process. **15**(8), 2299–2310 (2007)
18. Rao, B., Engan, K., Cotter, S., Palmer, J., Kreutz-Delgado, K.: Subset selection in noise based on diversity measure minimization. IEEE Trans. Signal Process. **51**(3), 760–770 (2003)
19. Saruwatari, H., Kawamura, T., Nishikawa, T., Lee, A., Shikano, K.: Blind source separation based on a fast-convergence algorithm combining ICA and beamforming. IEEE Trans. Audio Speech Lang. Process. **14**(2), 666–678 (2006)
20. Scheenen, T., Heijmink, S., Roell, S., Hulsbergen-Van de Kaa, C., Knipscheer, B., Witjes, J., Barentsz, J., Heerschap, A.: Three-dimensional proton MR spectroscopy of human prostate at 3 T without endorectal coil: Feasibility1. Radiology **245**(2), 507–516 (2007)
21. Starck, J., Elad, M., Donoho, D.: Image decomposition via the combination of sparse representations and a variational approach. IEEE Trans. Image Process. **14**(10), 1570–1582 (2005)
22. Tropp, J.: Greed is good: algorithmic results for sparse approximation. IEEE Trans. Inf. Theory **50**(10), 2231–2242 (2004)
23. Vanhamme, L., van den Boogaart, A., Van Huffel, S.: Improved method for accurate and efficient quantification of MRS data with use of prior knowledge. J. Magn. Reson. **129**(1), 35–43 (1997)

Chapter 15
Definition of a Discrete Color Monogenic Wavelet Transform

Raphael Soulard, Philippe Carré, and Christine Fernandez-Maloigne

Abstract In this chapter, we propose to review different approaches for the introduction of a color monogenic wavelet transform. Monogenic wavelets offer a geometric representation of grayscale images through an AM/FM model allowing invariance of coefficients to translations and rotations. The underlying concept of a local phase includes a fine contour analysis into a coherent unified framework. Wavelet based color image processing schemes have mostly been made by using a grayscale tool separately on color channels. In this chapter, we propose to discuss definitions that consider a color (vector) image right at the beginning of the mathematical definition. After a general description of the background of monogenic concept, we review a first approach built from the grayscale monogenic wavelets together with a color extension of the monogenic signal based on geometric algebra. Then, starting from a link with structure tensors, we discuss an alternative nontrivial extension of the monogenic framework to vector-valued signals. The crucial point is that our color monogenic wavelet transform is non-marginal and it inherits the coherent geometric analysis from the monogenic framework. Finally, we address the numerical aspect by introducing an innovative scheme that uses a discrete Radon transform based on discrete geometry.

15.1 Introduction

Since 2001, the *analytic signal* and its 2D generalizations have brought a great improvement to wavelets [8, 22, 27] by a natural embedding of an AM/FM analysis in the subband coding framework. This yields an efficient representation of the ge-

R. Soulard (✉) · P. Carré · C. Fernandez-Maloigne
XLIM Laboratory, UMR CNRS 7252 University of Poitiers, BP 30179, 86962 Futuroscope Chasseneuil Cedex, France
e-mail: raphael.soulard@univ-poitiers.fr

P. Carré
e-mail: philippe.carre@univ-poitiers.fr

C. Fernandez-Maloigne
e-mail: christine.fernandez@univ-poitiers.fr

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_15,

ometric structures in grayscale images thanks to a *local phase* carrying geometric information complementary to an *amplitude envelope* having good invariance properties. So, it codes the signal in a more coherent way than standard wavelets. The last and seemingly most appropriate proposition [27] of *analytic wavelets* for image analysis is based on the *monogenic signal* [12] defined with geometric algebra.

Despite this substantial research for the 2D case, signal tools are only applicable to grayscale images and hardly generalized to the vector-valued case. Yet, analyzing color data is essential for a lot of applications. Unfortunately, the processing of color images is most often based on a marginal scheme that is applying scalar tools separately on each color channel. Moreover, most of the work done with monogenic wavelet has been theoretical in nature and discussed in the context of continuous functions. The important bridge to discrete implementation and use in practical applications is tenuous at best. This chapter presents the new approaches that aim at representing color information with a discrete color Monogenic Wavelet transform based numerical algorithms.

The first section reviews a very recent color extension of Felsberg's work [10], and we describe its wavelet counterpart, proposed in a previous work [24] to carry out a non-marginal representation relying on vector extension of Cauchy–Riemann equations that are the fundamental basis of the monogenic framework. This definition needs to improve its underlying phase concept.

Next, we explain our *color monogenic wavelet transform* that extends the monogenic wavelets of [27] to color. This work goes further by proposing a fully interpretable color monogenic analysis based on a link between the Riesz transform and differential geometry. We may so expect to handle coherent information of multiresolution color geometric structure; which would make easier any wavelet based color image processing.

At the end of this chapter, we address the numerical aspect by proposing an innovative scheme that uses a discrete Radon transform based on discrete geometry. Radon domain signal processing and monogenic analysis is studied and performance is shown to be equivalent to the usual FFT-based algorithms. The advantage is that extensions to filterbanks and to higher dimensions are facilitated, thanks to the perfect invertibility and computational simplicity of the used Radon algorithm.

Notations

2-vector coordinates: $\boldsymbol{x} = (x, y)$, $\boldsymbol{\omega} = (\omega_1, \omega_2) \in \mathbb{R}^2$; $\boldsymbol{k} \in \mathbb{Z}^2$
Euclidean norm: $\|\boldsymbol{x}\| = \sqrt{x^2 + y^2}$
Complex imaginary number: $\boldsymbol{j} \in \mathbb{C}$
Argument of a complex number: arg
Convolution symbol: $*$
Fourier transform: $\mathscr{F}$

15.2 Analytical Signal and 2D Generalization

This section recalls existing definitions around the analytic signal and the monogenic signal. The multiscale aspect will be presented through an overview of existing *analytic wavelets* followed by a more detailed description of the monogenic wavelets by Unser et al. [27].

15.2.1 Analytic Signal (1D)

An *analytic signal* s_A is a multi-component signal associated to a real signal s to be analyzed. The definition is well known in the 1D case where $s_A(t) = s(t) + \boldsymbol{j}(h * s)(t)$ is the complex signal made of s and its Hilbert transform (with $h(t) = \frac{1}{\pi t}$).

The polar form of the 1D analytic signal provides an AM/FM representation of s with $|s_A|$ being the *amplitude envelope* and $\varphi = \arg(s_A)$ the *instantaneous phase*. This classical tool can be found in many signal processing books and is used in communications, for example. The growing interest in this tool within the image community is due to an alternative interpretation of amplitude, phase, and frequency in terms of a local geometric shape. We can interpret the phase in terms of a signal shape. Such a link between a 2D phase and local geometric structures of images would be very attractive in image processing. That is why there were several attempts to generalize it for 2D signals; and among them the *monogenic signal* [12] seems the most advanced since it is rotation invariant.

15.2.2 Monogenic Signal (2D)

We here review the key points of the fundamental construction of the monogenic signal, which will be necessary to understand the color extension. The 2D extension of the analytic signal has been defined in several ways [6, 14, 15]. We are interested in the *monogenic* signal [14] because this is rotation invariant and its generalization is according to both fundamental definition and signal interpretation. Given a 2D real (scalar) signal s, the associated monogenic signal s_M is 3-vector valued (instead of complex-valued in the 1D case) and must be taken in spherical coordinates:

$$s_M = \begin{bmatrix} s \\ \Re\{\mathscr{R}s\} \\ \Im\{\mathscr{R}s\} \end{bmatrix} = \begin{bmatrix} A\cos\varphi \\ A\sin\varphi\cos\theta \\ A\sin\varphi\sin\theta \end{bmatrix} \tag{15.1}$$

where $\mathscr{R}s$ is the complex-valued Riesz transform of s:

$$\{\mathscr{R}s\}(\boldsymbol{x}) = \text{p.v.} \int \frac{\tau_1 + \boldsymbol{j}\tau_2}{2\pi\|\boldsymbol{\tau}\|^3} s(\boldsymbol{x} - \boldsymbol{\tau})\, d\boldsymbol{\tau} \quad \overset{\mathscr{F}}{\longleftrightarrow} \quad \frac{\omega_2 - \boldsymbol{j}\omega_1}{\|\boldsymbol{\omega}\|}\hat{s}(\boldsymbol{\omega}). \tag{15.2}$$

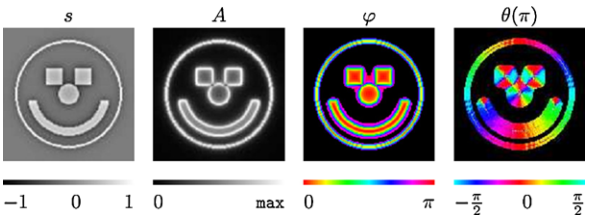


Fig. 15.1 Felsberg's monogenic signal associated to a narrow-band signal s. Orientation θ is shown modulo π for visual convenience. Phase values of small coefficients have no sense so they are replaced by black pixels

The monogenic signal is composed of the three following features:

$$\begin{array}{ll} \text{Amplitude:} & A = \sqrt{s^2 + |\mathscr{R}s|^2}, \\ \text{Orientation:} & \theta = \arg\{\mathscr{R}s\} \in [-\pi; \pi[, \\ \text{1D Phase:} & \varphi = \arg\{s + \boldsymbol{j}|\mathscr{R}s|\} \in [0; \pi]. \end{array} \tag{15.3}$$

A monogenic signal analysis is illustrated on Fig. 15.1. Felsberg shows a direct link between the angles θ and φ and the geometric local structure of s. The signal is expressed like an "A-strong" 1D structure with orientation θ. φ is analogous to the 1D local phase and indicates if the structure is a line or an edge. A direct drawback is that intrinsically 2D structures are not handled.

From a *signal processing* viewpoint, the AM/FM representation provided by an analytic signal is accordingly well suited for narrow-band signals. That is why it seems natural to embed it in a multiresolution transform that performs subband decomposition. We now present the monogenic analysis proposed in [27].

15.2.3 Monogenic Multiresolution

The first proposition of analytic wavelets is for the 1D scalar case with the *Dual-tree Complex Wavelet Transform* ($\mathbb{C}$WT) in 1999 [22]. It is a 1D discrete scheme consisting of two parallel filterbanks which filters are linked by Hilbert transforms. In fact, the Hilbert transforms are approximate because of discrete constraints. This method allows near shift-invariance of wavelet coefficients (shift-variance is a famous problem of classical wavelets).

In 2004, a Quaternion Wavelet Transform (QWT) [3, 8] based on the quaternionic analytic signal of [6] is proposed for grayscale images. The quaternionic signal is a 2D generalization of the *analytic signal* that is prior to and maybe less convincing than the monogenic signal.

Finally, in 2009 a Monogenic Wavelet Transform was proposed in [27]. This representation—specially defined for 2D signals—is a great theoretic improvement of the complex and quaternion wavelets.

It provides 3-vector valued monogenic subbands consisting of a rotation-covariant *magnitude* and this new 2D *phase*. The proposition of [27] consists of one

real-valued "primary" wavelet transform in parallel with an associated complex-valued wavelet transform. Both transforms are linked by the Riesz transform so they carry out a multiresolution monogenic analysis. We end up with 3-vector coefficients forming subbands that are monogenic.

15.2.3.1 Primary Transform

The primary transform is real-valued and relies on a dyadic pyramid decomposition tied to a wavelet frame. Only one 2D wavelet is needed and the dyadic downsampling is done only at the low frequency branch; leading to a redundancy of 4:3. The scaling function φ_γ is defined in the Fourier domain:

$$\varphi_\gamma \overset{\mathscr{F}}{\longleftrightarrow} \frac{(4(\sin^2\frac{\omega_1}{2}+\sin^2\frac{\omega_2}{2})-\frac{8}{3}\sin^2\frac{\omega_1}{2}\sin^2\frac{\omega_2}{2})^{\frac{\gamma}{2}}}{\|\boldsymbol{\omega}\|^\gamma} \tag{15.4}$$

and the mother wavelet ψ is defined as

$$\psi(\boldsymbol{x}) = (-\Delta)^{\frac{\gamma}{2}}\varphi_{2\gamma}(2\boldsymbol{x}) \tag{15.5}$$

where Δ is the Laplacian operator and φ_γ is a cardinal polyharmonic spline of order γ and spans the space of those splines with its integer shifts. It also generates—as a scaling function—a valid multiresolution analysis. This particular construction is made by an extension of a wavelet basis (non-redundant) related to a critically-sampled filterbank. In addition, a specific *subband regression* algorithm is used at the synthesis side. The construction is fully described in [28].

15.2.3.2 The Monogenic Transform

The second "Riesz part" transform is a complex-valued extension of the primary one with the Riesz transform:

$$\psi' = -\left(\frac{x}{2\pi\|\boldsymbol{x}\|^3} * \psi(\boldsymbol{x})\right) + j\left(\frac{y}{2\pi\|\boldsymbol{x}\|^3} * \psi(\boldsymbol{x})\right). \tag{15.6}$$

It can be shown that it generates a valid wavelet basis and that it can be extended to the pyramid described above. The joint consideration of both transforms form monogenic subbands from which the amplitude and phase can be extracted for an overall redundancy of 4:1.

In [27], a demonstration of AM/FM analysis is done with fine orientation estimation and gives very good results in terms of coherency and accuracy. Accordingly, this tool may be rather used for analysis tasks than processing. We propose now to describe a first generalization for color images.

15.3 First Extension of Color Monogenic Wavelet

Our first proposition of color Monogenic Wavelet combines a fundamental generalization of the monogenic signal to color with the monogenic wavelets described above. The challenge is to avoid the classical *marginal* definition that would be applying a *grayscale* monogenic transform on each of the three color channels of a color image.

15.3.1 The Color Monogenic Signal

Starting from Felsberg's approach that is originally expressed in the geometric algebra of $\mathbb{R}^3$, the extension proposed in [10] is written in the geometric algebra of $\mathbb{R}^5$. By simply increasing the dimension, we can embed each color channel along a different axis, and the original equation associated with the monogenic signal from Felsberg involving a 3D Laplace operator can be generalized in 5D. Then, the system can be simplified by splitting it into three systems with a 3D Laplace equation, reducing to an application of Felsberg's condition to each color channel. Instead of naively applying the Riesz transform to each color channel, this fundamental generalization carries out the following color monogenic signal: $s_A = (s_R, s_G, s_B, s_{r1}, s_{r2})$ where s_{r1} and s_{r2} are the Riesz transforms applied to $s_R + s_G + s_B$ [10]. Now, that the color extension of Felsberg's monogenic signal is defined, let us construct the color extension of the monogenic wavelets.

15.3.2 The Color Monogenic Wavelet Transform

We can now define a wavelet transform whose subbands are color monogenic signals. We can simply use the transforms presented above by applying the *primary* one on each color channel and the *Riesz part* on the sum of the three. The five related color wavelets forming one color monogenic wavelet ψ_A are:

$$\psi_R = \begin{pmatrix} \psi \\ 0 \\ 0 \end{pmatrix}, \qquad \psi_G = \begin{pmatrix} 0 \\ \psi \\ 0 \end{pmatrix}, \qquad \psi_B = \begin{pmatrix} 0 \\ 0 \\ \psi \end{pmatrix}, \tag{15.7}$$

$$\psi_{r1} = \begin{pmatrix} \frac{x}{2\pi \|\boldsymbol{x}\|^3} * \psi \\ \frac{x}{2\pi \|\boldsymbol{x}\|^3} * \psi \\ \frac{x}{2\pi \|\boldsymbol{x}\|^3} * \psi \end{pmatrix}, \qquad \psi_{r2} = \begin{pmatrix} \frac{y}{2\pi \|\boldsymbol{x}\|^3} * \psi \\ \frac{y}{2\pi \|\boldsymbol{x}\|^3} * \psi \\ \frac{y}{2\pi \|\boldsymbol{x}\|^3} * \psi \end{pmatrix}, \tag{15.8}$$

$$\psi_A = (\psi_R, \psi_G, \psi_B, \psi_{r1}, \psi_{r2}). \tag{15.9}$$

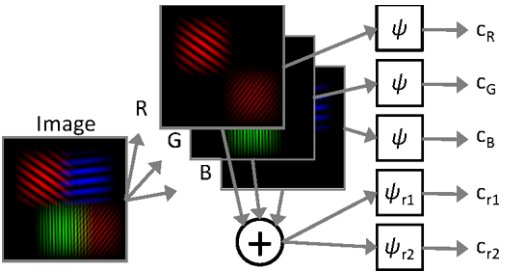


Fig. 15.2 Color MWT scheme. Each color channel is analyzed with the primary wavelet transform symbolized by a ψ bloc and the sum "$R + G + B$" is analyzed with the "Riesz part" wavelet transform (ψ_{r1} and ψ_{r2} blocs)

We then get 5-vector coefficients forming a color monogenic wavelet transform. The associated decomposition is described by the diagram in Fig. 15.2. This provides a multiresolution color monogenic analysis made of a 5-vector valued pyramid transform. The five decompositions of two images are shown in Fig. 15.3 from left to right. Each one consists of four juxtaposed image-like subbands resulting from a 3-level decomposition.

Note that low intensity corresponds to "no structure", i.e., where the image has no geometric information. It is coherent not to display the orientation (low intensity makes the hue invisible) for these coefficients since this data has no sense in those cases. These are three primary transforms c_R, c_G, and c_B where white (resp., black) pixels are large positive (resp., negative) values.

Whereas marginal separable transforms show three arbitrary orientations within each color channel—which is not easily interpretable—the color monogenic wavelet transform provides a more compact *energy* representation of the color image content regardless of the local orientation. The color information is well separated through c_R, c_G, and c_B. In each of the three decompositions, it is clear that every orientation

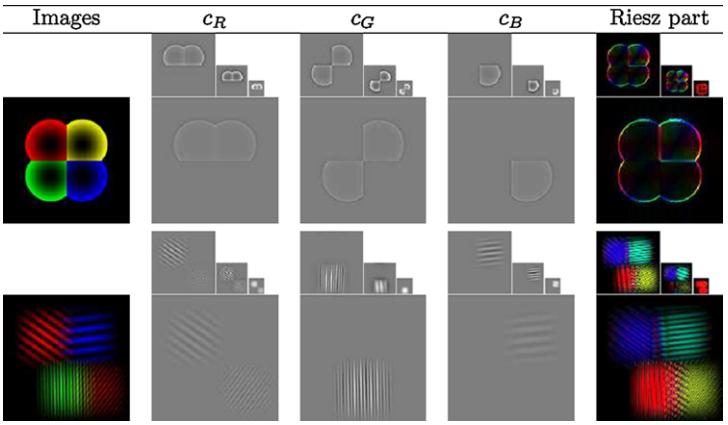


Fig. 15.3 Color MWT of images. The two components of the *Riesz part* are displayed in the same graphic with the magnitude of $c_{r1} + jc_{r2}$ encoded in the intensity and argument (local orientation) encoded in the hue

is equally represented all along the round contours. That is different from a separable transform that privileges particular directions.

Now look at the "2-in-1" last decomposition forming the Riesz part. It is displayed in one color map where the geometric energy $\sqrt{c_{r1}^2 + c_{r2}^2}$ is encoded into the intensity (with respect to the well known HSV color space) and the orientation $\arg(c_{r1} + jc_{r2})(\pi)$ is encoded in the hue (e.g., red is for $\{0, \pi\}$ and cyan is for $\pm\frac{\pi}{2}$). This way of displaying the Riesz part reveals the provided geometric analysis of the image well.

The Riesz part makes a precise analysis that is *local* both in space and scale. If there is a local color geometric structure in the image at a certain scale, the Riesz part exhibits a high intensity in the corresponding position and subband. This is completed with an orientation analysis (hue) of the underlying structure. For instance, a horizontal (resp., vertical) structure in the image will be coded by a cyan (resp., red) intense point in the corresponding subband. The orientation analysis is strikingly coherent and accurate. See, for example, that color structures with constant orientation (second image) exhibit a *constant* hue in the Riesz part over the whole structure. This transform is non-marginal because RGB components are considered as well as the intensity $(R + G + B)$, which involves two different color spaces. Image processing tasks such as denoising need the synthesis part of filterbanks. In the case of redundant representations, there are often several ways to perfectly reconstruct the transformed image.

This issue occurs in the scalar case since the pyramids we use have redundancy of 4:3. The associated reconstruction algorithm is well defined by the authors in [28] and consists of using spatial redundancy of each subband at the synthesis stage, by using the so-called *subband regression* algorithm.

In our case, we have to face another kind of redundancy, that of the monogenic model. Apart from any wavelet decomposition, monogenic representation (as well as analytic) is already basically redundant since additional signals are processed (Riesz part). In our case, the following distinct reconstructions are possible:

- We can reconstruct the whole color image (R, G, B) from the sole primary part (c_R, c_G, c_B).
- The Riesz part $c_{r1} + jc_{r2}$ can be used to reconstruct $R + G + B$, which is only a partial reconstruction.
- One can as well combine both reconstructions with a specific application driven method.

In every case, the reconstruction is perfect. What is unknown is the meaning of the wavelet domain processing with respect to a chosen reconstruction method.

The main way to improve this work is to focus on a physical interpretation when designing the key color monogenic concept. Although the generalization is not strictly marginal, it has a marginal style since it reduces to applying the Riesz transform on the intensity of the image. We propose now to work on a second numerical definition of color monogenic wavelets, where the physical interpretation is taken more into account, and the local geometry is studied deeper through a vector differential geometry point of view.

15.4 Second Approach for Color Monogenic Signal: A Tensor Approach

Given the great success of differential approaches in color vision [11, 26], we propose now to take the advantage of the well established *color structure tensor* for a new color extension of monogenic analysis. The theoretical link between Riesz transform and gradient has already been studied and used in the grayscale case in [13, 18, 27]. We extend it to color, thanks to the vector structure tensor concept.

15.4.1 Link Between Riesz and Gradient

Here is the recalled classical gradient-based local analysis of grayscale images. A detailed explanation will be found in [17]. The gradient of an image s is defined by:

$$\nabla s = \begin{bmatrix} \frac{\partial s}{\partial x} & \frac{\partial s}{\partial y} \end{bmatrix}^{\mathsf{T}} = [s_x \quad s_y]^{\mathsf{T}} \overset{\mathscr{F}}{\longleftrightarrow} [\boldsymbol{j}\omega_1 \hat{s} \quad \boldsymbol{j}\omega_2 \hat{s}]^{\mathsf{T}}. \tag{15.10}$$

It points toward the direction of the *local maximum variation* of s, and its amplitude is relative to the strength of this variation:

$$\mathscr{N} = \sqrt{s_x^2 + s_y^2} \quad \text{(gradient norm)}, \tag{15.11}$$

$$\theta_+ = \arg\{s_x + \boldsymbol{j} s_y\} \quad \text{(gradient direction)}. \tag{15.12}$$

The *edge strength* $\mathscr{N}$ and *orientation* θ_+ form the well-known basic features for edge detection. However, the gradient analysis is only efficient for edge-like structures (see 'intrinsically 1D' or 'simple neighborhoods'), which is tied to the fact that it is done *pointwise*.

Let us now consider the neighborhood to define a more relevant oriented local variation [17], with h being a window function defining the neighborhood and acting like a smoothing kernel. The measure is squared in order to merge opposite directions; this provides a *quadratic form*. Its maximization is known to be equivalent to finding eigenvalues/eigenvectors of the underlying symmetric positive-definite matrix:

$$T(s) = \begin{bmatrix} h * s_x^2 & h * s_x s_y \\ h * s_x s_y & h * s_y^2 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ T_{12} & T_{22} \end{bmatrix} \tag{15.13}$$

called the *structure tensor*. The eigenvalues can be derived analytically:

$$\lambda_\pm = \frac{1}{2}\Big(T_{11} + T_{22} \pm \sqrt{(T_{22} - T_{11})^2 + 4T_{12}^2}\Big). \tag{15.14}$$

The eigenvector tied to λ_+ is parallel to $[\cos(\theta_+) \ \sin(\theta_+)]$ with

$$\theta_+ = \frac{1}{2}\arg\{T_{11} - T_{22} + \boldsymbol{j}2T_{12}\}. \tag{15.15}$$

As already studied in [18, 27], the Riesz transform is analogous to the gradient. More precisely, Eqs. (15.2) and (15.10) give

$$\mathscr{R}s = \left(-(-\Delta)^{-\frac{1}{2}} s_x\right) + \boldsymbol{j}\left(-(-\Delta)^{-\frac{1}{2}} s_y\right). \tag{15.16}$$

As a result, $\mathscr{R}$ can be viewed either like the smoothed gradient of s or like the gradient of a smoothed version of s. In [27], a Riesz counterpart of the structure tensor is derived to improve the Riesz analysis. Based on this fact, the whole structure tensor formalism can be derived with $\mathscr{R}$ replacing ∇, so that we get the Riesz based tensor T_{rz} defined as follows:

$$T_{\mathrm{rz}}(s) = h * \left[\Re\{\mathscr{R}s\} \quad \Im\{\mathscr{R}s\}\right]^{\mathsf{T}} \left[\Re\{\mathscr{R}s\} \quad \Im\{\mathscr{R}s\}\right]. \tag{15.17}$$

The Riesz features are equivalent to the structure tensor features of a smoothed version of s:

$$|\mathscr{R}s| \equiv \mathscr{N}, \tag{15.18}$$

$$\arg\{\mathscr{R}\} \equiv \theta_+. \tag{15.19}$$

Finally, the building block of the monogenic analysis $\mathscr{R}$ performs the same efficient orientation analysis as a gradient. The advantage over the classical gradient is that it gives access to the local phase and frequency, thanks to the monogenic concept that also includes the subband component s^{bp} in a unified framework. It is now possible to extend it to color signals.

15.4.2 Color Riesz Analysis

The color structure tensor is the central tool of color differential approaches. The idea was first proposed by Di Zenzo in [11], and then further developed in [21]. Given a color image $s = (s^{\mathsf{R}}, s^{\mathsf{G}}, s^{\mathsf{B}})$, consider its marginal gradients along x and y:

$$\left[\nabla s^{\mathsf{R}}, \nabla s^{\mathsf{G}}, \nabla s^{\mathsf{B}}\right] = \left[s_x^{\mathsf{R}}, s_y^{\mathsf{R}}, s_x^{\mathsf{G}}, s_y^{\mathsf{G}}, s_x^{\mathsf{B}}, s_y^{\mathsf{B}}\right]. \tag{15.20}$$

The color structure tensor M is defined as follows:

$$M(s) = T\left(s^{\mathsf{R}}\right) + T\left(s^{\mathsf{G}}\right) + T\left(s^{\mathsf{B}}\right) = \begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix} \tag{15.21}$$

with

$$M_{11} = h * \left(\left(s_x^{\mathsf{R}}\right)^2 + \left(s_x^{\mathsf{G}}\right)^2 + \left(s_x^{\mathsf{B}}\right)^2\right), \tag{15.22}$$

$$M_{12} = h * \left(s_x^{\mathsf{R}} s_y^{\mathsf{R}} + s_x^{\mathsf{G}} s_y^{\mathsf{G}} + s_x^{\mathsf{B}} s_y^{\mathsf{B}}\right), \tag{15.23}$$

$$M_{22} = h * \left(\left(s_y^{\mathsf{R}}\right)^2 + \left(s_y^{\mathsf{G}}\right)^2 + \left(s_y^{\mathsf{B}}\right)^2\right). \tag{15.24}$$

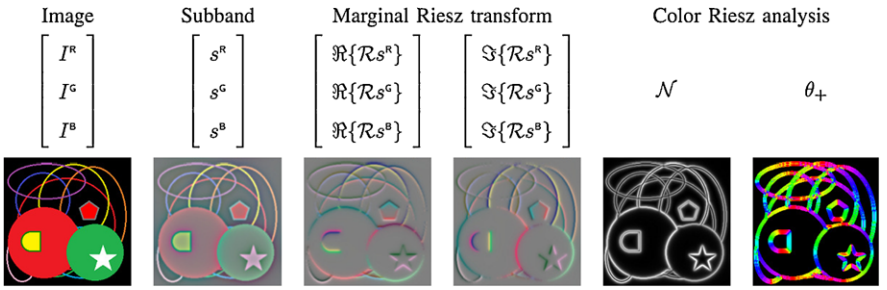


Fig. 15.4 Color Riesz analysis. *From left to right*: Color image, marginal bandpass filtering output, marginal Riesz transform (2 components), color Riesz features (norm and direction)

The norm $\mathcal{N}$ and direction θ_+ of the maximum local variation are again obtained from the eigenvalues and eigenvectors according to Eqs. (15.14) and (15.15).

We saw above that Riesz features are equivalent to the gradient norm and direction, so we straightforwardly obtain the following color Riesz features:

$$\mathcal{N} = \sqrt{\left|\mathscr{R}s^{\mathsf{R}}\right|^2 + \left|\mathscr{R}s^{\mathsf{G}}\right|^2 + \left|\mathscr{R}s^{\mathsf{B}}\right|^2}, \tag{15.25}$$

$$\theta_+ = \frac{1}{2}\arg\left\{\sum_{\mathsf{C}\in\{\mathsf{R},\mathsf{G},\mathsf{B}\}} \Re\left\{\mathscr{R}s^{\mathsf{C}}\right\}^2 - \Im\left\{\mathscr{R}s^{\mathsf{C}}\right\}^2 + \boldsymbol{j}\sum_{\mathsf{C}\in\{\mathsf{R},\mathsf{G},\mathsf{B}\}} 2\Re\left\{\mathscr{R}s^{\mathsf{C}}\right\}\Im\left\{\mathscr{R}s^{\mathsf{C}}\right\}\right\}. \tag{15.26}$$

The advantage of defining color Riesz features with differential geometry is the proper analysis of *color* discontinuities; as illustrated in Fig. 15.4.

Figure 15.4 shows a bandpass filtering of a color image followed by the color Riesz analysis. We observe that the color Riesz analysis gives coherent orientations of all color contours, including isoluminant ones like the border between red and green disks that would disappear in an intensity-based scheme like this of the first description.

As for the grayscale approach, we measure the spatial orientation corresponding to the geometry of image content that is given by θ_+ from the Riesz analysis. The last step is to build the 1D phase for color images. In the grayscale case, the analytic signal consists in combining a signal s with a phase-shifted version of itself ($\mathscr{H}s$ or $|\mathscr{R}s|$) to extract local amplitude A and phase φ. In 2D, orientation is locally taken into account in the definition of the phase-shift so that $|\mathscr{R}s|$ is analogous to $\mathscr{H}s$ in the direction of maximum variation. With Eq. (15.18) it turns out that this phase-shifted signal is, in fact, the Riesz gradient norm $\mathcal{N}$ which also holds in the color case with Eq. (15.25). However, s is a 3-vector while $\mathcal{N}$ is still scalar. Fortunately, the Euclidean norm $\|s\| = \sqrt{(s^{\mathsf{R}})^2 + (s^{\mathsf{G}})^2 + (s^{\mathsf{B}})^2}$ carries all the needed information to compute the meaningful amplitude and phase.

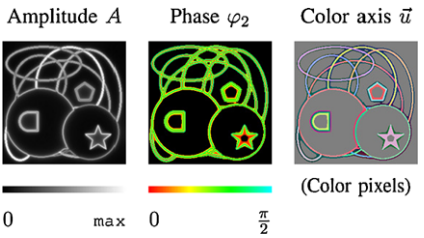


Fig. 15.5 Second version of color monogenic signal of image used in Fig. 15.4. Here again, the color axis and phase data are not displayed (black or gray) for coefficients with low amplitude

The color monogenic model is defined as follows:

$$s = \underbrace{\sqrt{\|s\|^2 + \mathcal{N}^2}}_{A} \cos\underbrace{\left(\arg\left\{\|s\| + \boldsymbol{j}\mathcal{N}\right\}\right)}_{\varphi_2} \underbrace{\mathbf{u}}_{\text{'axis'}} \tag{15.27}$$

where $\mathbf{u} = s/\|s\|$ indicates a direction in the 3D color space and φ_2 is the usual 1D phase. The gradient norm $\mathcal{N}$ is obtained with Eq. (15.25). Finally, the amplitude and phase can be retrieved with the sole Euclidean norm of s. The new color monogenic signal is built like a 4-vector whose spherical coordinates are the amplitude, phase, and color axis:

$$\begin{aligned} &s_M^{\text{color}} = [s^{\mathsf{R}}\ s^{\mathsf{G}}\ s^{\mathsf{B}}\ \mathcal{N}]^{\mathsf{T}}, \\ &\text{Amplitude: } A = \sqrt{\|s\|^2 + \mathcal{N}^2} \in [0; +\infty[, \\ &\text{1D Phase: } \varphi_2 = \arg\left\{\|s\| + \boldsymbol{j}\mathcal{N}\right\} \in [0; \tfrac{\pi}{2}[, \\ &\text{Color axis: } \mathbf{u} = s/\|s\|. \end{aligned} \tag{15.28}$$

Let us observe Fig. 15.5 illustrating this color monogenic signal. Like previously, the analysis is done on a subband of the color image obtained with the same filter. We can see that the amplitude is again coherent with geometrical structures (including isoluminant ones) and highlights oriented elements equally regardless of their orientation—due to Riesz transform isotropy. Its invariance to shift and rotation is due to the sharing of geometric information with φ_2 which forms a coherent coding. This also allows coding a line by a 'simple line' in amplitude instead of a 'double line' thanks to the encoding of the *kind of discontinuity* by φ_2. Orientation θ_+ is exactly that of Fig. 15.4; $\mathbf{u}$ carries some information of the local color direction.

Based on this extension of the monogenic signal, we can derive the corresponding color extension of the MWT presented previously.

15.4.3 Tensor Based Color Monogenic Wavelet Transform

The extension to the wavelet domain is direct since the above construction relies on a marginal Riesz transform (non-marginality occurs when combining marginal outputs into meaningful data). So we can again directly use polyharmonic spline wavelets of Unser et al. but with a different combination of Cartesian coefficients. This time components will be combined to carry out color AM/FM analysis.

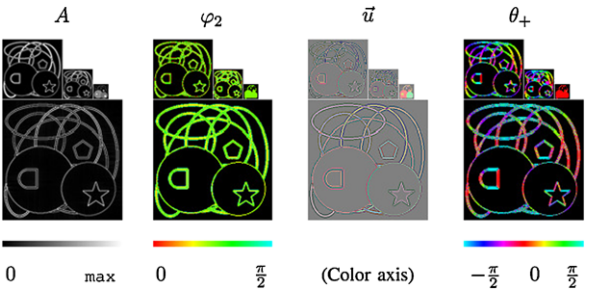


Fig. 15.6 The color monogenic wavelet transform ($\gamma = 3$)

We have first to compute the 6 subband decompositions. The amplitude, phase, and color axis can be retrieved with Eq. (15.28), and the orientation with Eq. (15.26). Figure 15.6 illustrates the multiscale color monogenic features obtained from our color monogenic wavelet transform.

15.4.4 Algorithm Discussion

In this subsection, let us give some practical remarks. The monogenic analysis is basically defined in a continuous framework. Constraints related to filterbank design—perfect reconstruction, small redundancy—conflict with desirable properties of isotropy and rotation invariance. The important bridge to discrete implementation and use in practical applications is tenuous at best. The choice that is made by Unser et al. [27] is to provide the 'minimally-redundant wavelet counterpart of Felsberg's monogenic signal'. The presented color monogenic wavelet transform is by extension in the same spirit. Since filters cannot be exactly isotropic, the analysis is expected to mildly favor some directions. In addition, the subbands are highly subsampled (yet not 'critically' since the number of coefficients is larger than the number of pixels), implying that the phase data is varying fast with respect to sampling.

In the last part of this chapter, we believe that such a signal processing tool must be studied from a discrete viewpoint. This last part presents a new approach that aims at representing color information with a discrete color Monogenic Wavelet transform based on discrete Monogenic transform.

We address this issue by introducing a scheme that uses a discrete Radon transform based on discrete geometry. The advantage is that extensions to filterbanks and to higher dimensions are facilitated, thanks to the perfect invertibility and computational simplicity of the used Radon algorithm.

The analysis now presented is based on two facts:

- There is a fundamental link between the monogenic framework and the Radon transform [18];
- 'True' discrete counterparts of Radon transforms have already been defined (for example, in [7]).

More precisely, the monogenic concept is basically made of a Radon transform joined with a 1D phase analysis, so we can say that the Radon transform is responsible for *isotropy*. This crucial point has always been a deep issue in the discrete world, while at the core of monogenic analysis.

In the past, we have proposed [7] a discrete Radon transform with exact reconstruction is designed with the help of discrete analytical geometry. This last part studies the use of this well established discrete representation to perform monogenic analysis.

We now present the algorithm implementing the discrete Radon transform.

15.5 The Radon Domain for Numerical Monogenic Transform

It is shown in [5] that the Riesz transform is equivalent to an independent Hilbert transform (1D) on each Radon projection, combined with a sine-like weighting depending on θ:

$$\{\mathscr{R}s\}_\theta(t) = \{\mathscr{H}s_\theta\}(t)e^{\mathbf{j}\theta} \tag{15.29}$$

where $\mathscr{H}s_\theta$ is the Hilbert transform of s_θ defined by

$$\{\mathscr{H}s_\theta\}(t) = \left(s_\theta(\cdot) * \frac{1}{(\pi\cdot)}\right)(t). \tag{15.30}$$

The Hilbert transform is well known and already integrated, for example, in some *analytic* wavelet transforms. Performing some monogenic analysis (based on the Riesz transform) then reduces to a more classical problem in the Radon domain.

The Radon domain represents 2D functions by a set of 1D projections at several orientations. It forms a fundamental link with 1D and 2D Fourier transforms and so handles isotropic filtering well.

15.5.1 The Radon Transform

Given a 2D function $s(x, y)$, its projection into the Radon domain along direction θ is defined by

$$s_\theta(t) = \int_{\mathbb{R}} s(\tau\sin\theta + t\cos\theta, -\tau\cos\theta + t\sin\theta)\,d\tau. \tag{15.31}$$

The Radon transform can be obtained by applying the 1D inverse Fourier transform to the 2D Fourier transform restricted to radial lines going through the origin (this is exactly what we are going to do in the discrete Fourier domain with the help of discrete analytical lines):

$$\widehat{s}(\omega\cos\theta, \omega\sin\theta) = \int_{\mathbb{R}} e^{-j\omega t} Rs(\theta, t)\,dt$$

where $\widehat{s}$ is the 2D Fourier transform of s. This is the projection-slice formula that is used in image reconstruction from projection methods.

The discretization of the Radon transform is difficult to achieve. The majority of methods proposed in the literature have been devised for computerized tomography or to approximate the continuous formula. None of them, however, were specifically designed to be invertible transforms for discrete images and can therefore not be used for the discrete Riesz transform.

The discrete Radon transform can be computed with one the two following strategies:

- Spatial strategy for digital Radon transform. The Radon transform is defined as summations of image pixels over a certain set of lines that are defined in a finite geometry.
- Fourier strategy for digital Radon transform. The projection-slice formula suggests that approximate Radon transforms for digital data can be based on discrete Fourier transforms (DFT). The Fourier-domain computation of an approximate digital Radon transform is defined as:

  1. Compute the 2D DFT of s.
  2. Extract Fourier coefficients along the lines L_θ going through the origin.
  3. Compute the 1D inverse DFT on each line L_θ (defined for each value of the angular parameter θ).

  This approach can be problematic since step 2 is not naturally defined on discrete data.

In this last part, we propose to define a fast and simple reversible digital Riesz transform. For this, we use the Fourier strategy for the associated digital Radon transform. Our lines L_θ are defined with the help of the discrete analytical geometry theory in the Fourier domain [1, 20]. This solution allows us to have different Riesz transforms according to the arithmetical thickness of the discrete lines. This approach presents a limited wrap-around effect. This representation is redundant, however, the degree of redundancy can be adapted by our thickness parameter. Our Radon backprojection is very simple and permits an exact reconstruction.

15.5.2 Discrete Radon

The work in [7] consists in defining a true discrete decomposition by using the discrete geometry. Based on the Fourier slice theorem, discrete lines are defined in the 2D Fourier domain before performing an inverse 1D Fourier transform to each extracted line. An *arithmetical* thickness parameter can be used to control both redundancy and straight line connectivity. In all cases, perfect reconstruction is guaranteed, and the algorithm is as simple as it is fast.

The idea behind our associated discrete Radon transform is to represent each direction by a discrete analytical straight line. For this we need a discrete straight line

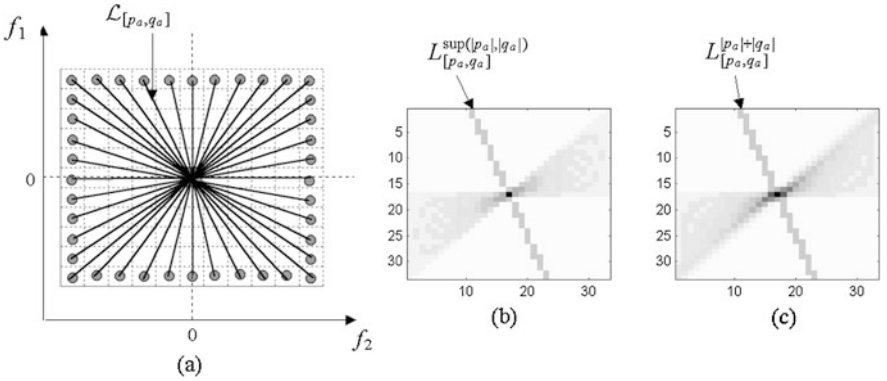


Fig. 15.7 (**a**) Cover of the Fourier domain with the Euclidean line $\mathscr{L}_{[p,q]}$. (**b**) Redundancy on the cover of the Fourier lattice by closed naive lines, (**c**) by supercover lines

that has a central symmetry and that forms a "good" approximation of the corresponding Euclidean straight line (i.e., direction). We chose discrete analytical 2D lines. It defines not a unique line but a family of lines with a thickness parameter, called arithmetical thickness. The arithmetical thickness provides a control over the transform redundancy factor and properties such as the connectivity of the straight line.

The discrete analytical lines we use for our transform are defined as follows [1]:

$$L^{\omega}_{[p,q]} = \left\{ (x_1, x_2) \in \mathbb{Z}^2 \;\middle|\; |qx_1 - px_2| \leq \frac{\omega}{2} \right\}$$

with $[p, q] \in \mathbb{Z}^2$ the direction of the Radon projection (we have $\theta = \arctan(\frac{q}{p})$) and ω, a function of (p, q), the arithmetical thickness. It is easy to see that these discrete analytical lines $L^{\omega}_{[p,q]}$ have a central symmetry regardless of the value of ω. The arithmetical thickness ω is an important parameter that controls, among other things, the connectivity of the discrete line $L^{\omega}_{[p,q]}$ [2]: naive lines with $\omega = \max(|p|, |q|)$ where $L^{\omega}_{[p,q]}$ is 8-connected and the supercover lines $\omega = |p| + |q|$, where $L^{\omega}_{[p,q]}$ is 4-connected, for example.

We use the Fourier domain for the computation of our discrete Radon transform: Fourier coefficients of $\hat{s}$ are extracted along the discrete analytical line $L^{\omega}_{[p,q]}$ (the extracted points of the line are ordered in a natural way) and we take the 1D inverse discrete Fourier transform on each value of the direction $[p, q]$ to obtain the Radon projection $R^{\omega}s([p, q], \cdot)$.

Figure 15.7 illustrates the cover of the Fourier lattice for two different types of discrete lines. The gray value of the pixel represents the redundancy in the projection (number of times a pixel belongs to a discrete line). One isolated line is drawn to illustrate the shape of the discrete lines depending on its arithmetical thickness.

At last, the set of discrete directions $[p, q]$ for a complete representation has to be determined. The set of line segments must cover all the square lattice in the Fourier

domain. For this, we define the directions $[p, q]$ according to pairs of symmetric points from the boundary of the 2D discrete Fourier spectra.

We now briefly discuss the strategy for inverting our discrete Radon Transform. Our analytical reconstruction procedure works as follows:

1. Compute the 1D FFT transform for each set $R^{\omega}s([p, q], \cdot)$ to obtain $P^{\omega}_{[p,q]}$.
2. Substitute the sampled value of $\hat{s}$ on the lattice where the points fall on lines $L^{\omega}_{[p,q]}$ with the sampled value of $\hat{s}$ on the square lattice:

$$\hat{s}^{[p,q]}\left(f_1^k, f_2^k\right) = P^{\omega}_{[p,q]}(k)$$

such that $|qf_1^k - pf_2^k| \leq \frac{\omega}{2}$ for $0 < k < K + 1$ with $K + 1$ the length of $L^{\omega}_{[p,q]}$ and for all the directions $[p, q]$.

Due to the redundancy, some Fourier coefficients belong to more than one discrete line. In this case, the Fourier value is defined by the mean average:

$$\hat{s}(f_1, f_2) = \frac{1}{R} \sum \hat{s}^{[p_r,q_r]}(f_1, f_2) \tag{15.32}$$

such that $|q_r f_1^k - p_r f_2^k| \leq \frac{\omega}{2}$. R is the number of times the pixel (f_1, f_2) belongs to a discrete line. It depends on the frequency (it is more important at low frequencies) and the type of discrete lines.

3. Apply the 2D IFFT transform.

The previous procedure allows us to obtain an exact reconstruction if the set of directions of lines provide a complete cover of the square lattice: analytical Radon transform followed by backprojection analytical Radon transform is a one-to-one transform (in this case all the coefficients $\hat{s}^{[p_r,q_r]}(f_1, f_2)$ in Eq. (15.32) are equal to the original value of $\hat{s}(f_1, f_2)$.

15.5.3 Discrete Radon Based Riesz Transform

We propose now to use this discrete Radon representation to perform monogenic analysis. Computing of a discrete Radon based monogenic analysis can now be done as follows:

- Apply 2D bandpass filtering to select some scale (an isotropic bandpass 2D filtering). As we have seen, bandpass filtering is natural in monogenic analysis that is usually presented either in a scale-space formalism or in a wavelet transform;
- Process the discrete analytical Radon transform of the filtering signal s;
- Process the Hilbert transform of every projection s_θ;
- Multiply every projection by $e^{j\theta}$ (by using the computation of θ explained above);

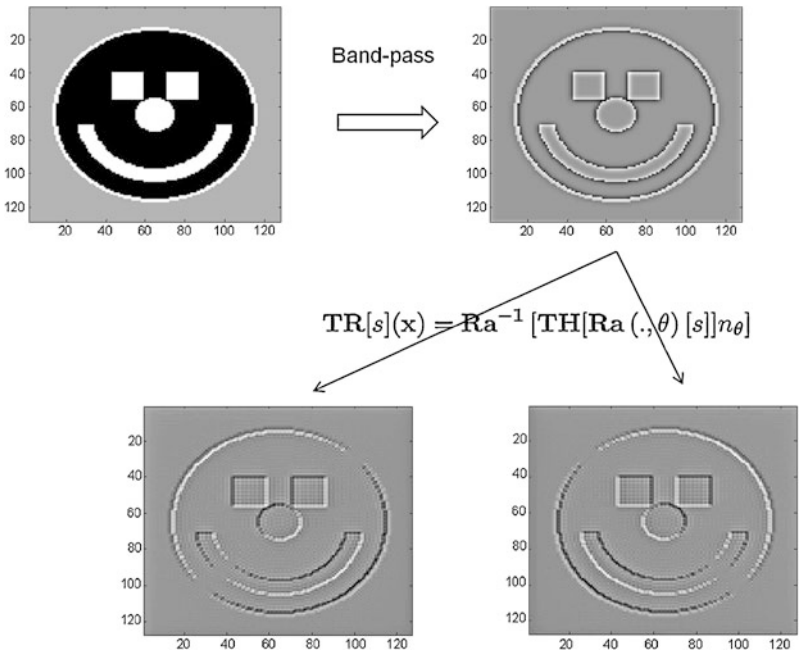


Fig. 15.8 Radon-based monogenic signal on a bandpass version of 'Smiley' image

- Process the inverse Radon transform on real and imaginary parts separately (we get s_{r_1} and s_{r_2});
- Convert s, s_{r_1}, and s_{r_2} to spherical coordinates as in Eq. (15.1).

An example of such a decomposition is given in Fig. 15.8. The result is very analogous to an FFT-based method. But the whole scheme is fast, thanks to the simplicity of the chosen Radon algorithm.

15.5.4 Discrete Radon Based Monogenic Wavelet Transform

In order to compute the Monogenic Wavelet transform, the isotropic bandpass 2D filtering is inserted into a classical discrete 2D Wavelet scheme to select some scale. The discrete wavelet transform (DWT) stems from the multiresolution analysis and filterbank theory [19] with two couples of filters: the filters $h, \widetilde{h}$ (analysis and reconstruction lowpass filters) and $g, \widetilde{g}$ (analysis and reconstruction highpass filters) that are quadrature mirror filters.

In order to get an exact restoration, two conditions are required on the conjugate filters [9]:

$$H(\omega)\widetilde{H}^*(\omega) + G(\omega)\widetilde{G}^*(\omega) = 1 \tag{15.33}$$

which implies a correct data restoration of one scale to the other, and

$$H(\omega + \pi/2)\widetilde{H}(\omega) + G(\omega + \pi/2)\widetilde{G}(\omega) = 0 \tag{15.34}$$

which represents the compensation of recovery effects introduced by the downsampling.

Because of decimation, the Mallat's decomposition is completely time variant. Moreover, it is difficult to define a 2D filter bank associated with one 2D isotropic bandpass component at each scale.

A simple way to obtain a time-invariant and isotropic system is to compute all the integer shifts of the signal. This algorithm was named algorithm "à trous"[16] and its link with the Mallat's algorithm is discussed in [23]. Because the decomposition is not decimated, filters are dilated between each projection. Therefore, in the signal case, each wavelets' scale has the same number of points as the original signal. For the scale L, these N points correspond to 2^L different decompositions obtained with the decimated transform using all the circulant shifts of the signal. These decompositions, each one composed of $N/2^L$ points, are intertwined.

The algorithm "à trous" presents many advantages:

- *A simpler filter selection*. Condition (15.34), which was required for a perfect reconstruction, is no longer necessary because coefficients are no longer downsampled.
- *Knowledge of all wavelets' coefficients*. Coefficients removed during the downsampling are not necessary for a perfect reconstruction, but they may contain information and are necessary to obtain a time invariant decomposition.

In this work, we propose a Monogenic Wavelet transform using the decomposition "à trous". For a perfect reconstruction, the algorithm "à trous" requires that the filters verify condition (15.33). A lot of works (for example, [4]) propose defining the highpass decomposition filter $G(\omega)$ as

$$G(\omega) = 1 - H(\omega) \tag{15.35}$$

where $H(\omega)$ is the lowpass filter, and the reconstruction filters are defined as

$$\widetilde{H}(\omega) = \widetilde{G}(\omega) = 1. \tag{15.36}$$

It is easy to verify that filters defined in Eqs. (15.35) and (15.36) satisfy Eq. (15.33) for all $H(\omega)$.

From Eq. (15.35), wavelets' coefficients are simply computed by the difference between two successive smoothed sequences, and the reconstruction is the sum of all wavelets' scales, plus the smoothed signal at the coarsest scale.

The generalization to the 2D case is done by the application of the lowpass filter along the two directions. Wavelets' coefficients are computed by the difference between two successive smoothed sequences and consequently are associated to isotropic bandpass.

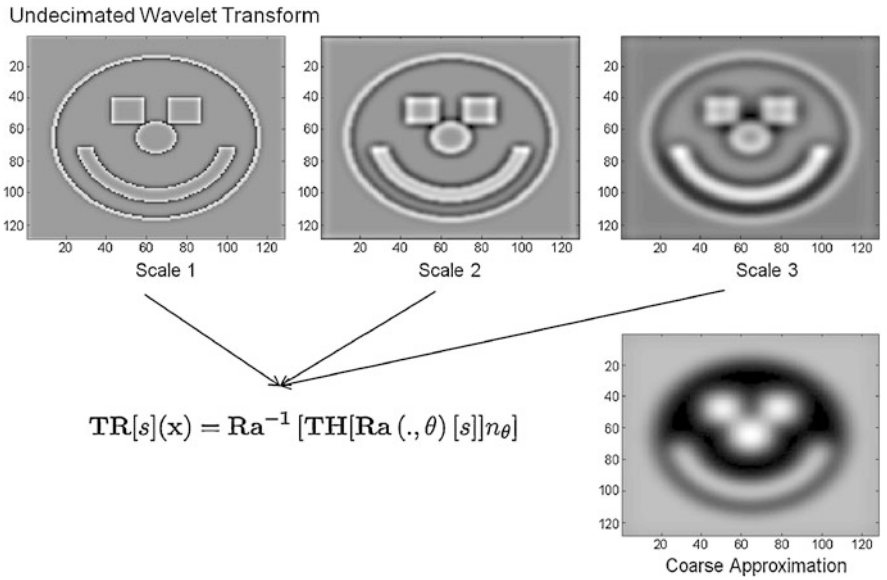


Fig. 15.9 Principle of the undecimated monogenic Wavelet transform

The scaling function is the B_3-Spline, and the associated filter H is

$$H(z) = \frac{1}{16}z^{-2} + \frac{1}{4}z^{-1} + \frac{3}{8} + \frac{1}{4}z + \frac{1}{16}z^2. \tag{15.37}$$

The limitation of this definition is that filters are not quadrature mirror filters. A consequence is the nonorthogonal decomposition, with a correlation between scales.

And finally, to compute the second "Riesz part", we apply on each bandpass filtering signal the process previously presented (Discrete Radon transform, Hilbert transform, normalization, and inverse Radon transform). An example of the principle of the decomposition is given in Fig. 15.9.

This last part shows that a discrete monogenic analysis can be performed in the Radon domain by using existing Radon transform algorithms. The improvement over the classical FFT-based method may become significant in more developed processes like wavelet transforms. The existing discrete Radon transform is a good tool having exact reconstruction and computational simplicity, thanks to discrete geometry.

15.6 Conclusion

In this chapter, we introduce and analyze some numerical color extensions of the monogenic wavelet transform. This new transform is a geometric non-marginal

color wavelet transform. These extensions are non-marginal since they take care of considering a vector signal at the very beginning of the fundamental construction and lead to a definition basically different from the marginal approach.

The use of non-separable wavelets jointly with the monogenic framework allows for a good orientation analysis well separated from the color information. This color transform can be a great color image analysis tool, thanks to this good separation of information through various data.

The first section outlines the first color extension that we proposed in [25] based on the recent color monogenic signal of [10]. This follows Felsberg's approach [14] by extending the Cauchy–Riemann equations within geometric algebra (a.k.a. 'Clifford algebra'). Finally, we keep advantages of the grayscale case in terms of accuracy of directional analysis and the fact that we have only one subband per scale. The color and geometric information of the image are well separated from each other and the invariance properties are kept. This transform is non-marginal because RGB components are considered jointly with the intensity ($R + G + B$). So, the original goal of defining a 'true' color monogenic wavelet transform is fulfilled. However, this color generalization is not developed enough to define some intuitive *phase* (the sole orientation analysis does not form a complete phase concept).

In addition, some color contours are not analyzed by the 'Riesz–Laplace' part because it is only based on the intensity data. It turns out that a deep study of the *color phase* concept is necessary to complete this generalization. To that end, the next section presents a second approach to the color extension of monogenic analysis.

The second construction is based on a theoretical link between the Riesz transform, the building block of the monogenic framework, and the gradient, the basis of the structure tensor. Thanks to the vector differential geometry, we build a color phase concept tied to a non-marginal color extension of the grayscale monogenic signal. The efficient tensor-based geometric analysis is joined to the physically interpretable amplitude/phase modeling, carrying out a unified representation of color images through *amplitude*, *phase*, *orientation,* and *color axis* data.

And finally, in order to address the issue of a discrete use of the continuous monogenic framework, we propose a scheme that uses a discrete Radon transform based on discrete geometry. The experimental equivalence with an FFT-based computation of the monogenic analysis is observed, but the prospects are more promising since the Radon domain is well handled for discrete data, as well as it extends well to higher dimensions. Exact reconstruction of the used Radon transform is also a fundamental property. The extension to a monogenic filterbank could be facilitated by this method.

This elegant distribution of the information allows a sparse representation, where most coefficients have low amplitude and thus insignificant phase and color axis. In some works, we illustrate this sparsity through compression experiments revealing the visual information carried by the different coefficients. Geometric multiscale analysis of color images is also investigated through invariant keypoint detection. While the famous SIFT algorithm is widely used for its invariance properties, we introduce the basis of a new method where color is naturally handled and physical interpretation of the data is possible, thanks to our signal processing approach. The

future work associated with this new numerical method will include defining higher level local descriptors to fully characterize the keypoints, as well as integrating human visual system tools for *interest point* detection.

References

1. Andres, E.: Modélisation analytique discrète d'objets géométriques. Habilitation, Université de Poitiers (2000)
2. Andres, E., Acharya, R., Sibata, C.: Discrete analytical hyperplanes. Graph. Models Image Process. **59**(5), 302–309 (1997)
3. Bayro-Corrochano, E., De la Torre Gomora, M.A.: Image processing using the quaternion wavelet transform. In: Proc. CIARP', Puebla, Mexico, pp. 612–620 (2004)
4. Bijaoui, A., Starck, J., Murtagh, F.: Restauration des images multi-échelles par l'algorithme à trous. Trait. Signal **11**, 232–243 (1994)
5. Brackx, F., Knock, B.D., Schepper, H.D.: On generalized Hilbert transforms and their interaction with the Radon transform in Clifford analysis. Math. Methods Appl. Sci. **30**, 1071–1092 (2007)
6. Bülow, T.: Hypercomplex spectral signal representation for the processing and analysis of images. Thesis (1999)
7. Carré, P., Andres, E.: Discrete analytical ridgelet transform. Signal Process. **84**, 2165–2173 (2004)
8. Chan, W.L., Choi, H.H., Baraniuk, R.G.: Coherent multiscale image processing using dual-tree quaternion wavelets. IEEE Trans. Image Process. **17**(7), 1069–1082 (2008)
9. Cohen, A.: Ondelettes et Traitement Numérique du Signal. Masson, Paris (1992)
10. Demarcq, G., Mascarilla, L., Courtellemont, P.: The color monogenic signal: a new framework for color image processing. In: Proc. IEEE Int'l Conf. on Image Processing (2009)
11. Di Zenzo, S.: A note on the gradient of a multi-image. Comput. Vis. Graph. Image Process. **33**(1), 116–125 (1986). doi:10.1016/0734-189X(86)90223-9
12. Felsberg, M.: Low-level image processing with the structure multivector. Thesis (2002)
13. Felsberg, M., Köthe, U.: Get: the connection between monogenic scale-space and Gaussian derivatives. In: Kimmel, R., Sochen, N., Weickert, J. (eds.) Proc. Scale-Space. LNCS, vol. 3459, pp. 192–203. Springer, Berlin (2005)
14. Felsberg, M., Sommer, G.: The monogenic signal. IEEE Trans. Signal Process. **49**(12), 3136–3144 (2001)
15. Hahn, S.L.: Multidimensional complex signals with single-orthant spectra. Proc. IEEE **80**(8), 1287–1300 (1992)
16. Holschneider, M., Kronland-Martinet, R., Morlet, J., Tchamitchian, P.: A real-time algorithm for signal analysis with the help of the wavelet transform. In: Combes, J., Grossmann, A., Tchamitchian, P. (eds.) Wavelet, Time–Frequency Methods and Phase Space, pp. 289–297. Springer, Berlin (1989)
17. Jähne, B.: Digital Image Processing, 6th edn. Springer, Berlin (2005)
18. Köthe, U., Felsberg, M.: Riesz-transforms versus derivatives: on the relationship between the boundary tensor and the energy tensor. In: Kimmel, J.W.R., Sochen, N. (eds.) Proc. Scale-Space. LNCS, vol. 3459, pp. 179–191. Springer, Berlin (2005)
19. Mallat, S.: A theory for multiresolution signal decomposition: the wavelet transform. IEEE Trans. Pattern Anal. Mach. Intell. **11**(7), 674–693 (1989)
20. Reveillès, J.P.: Géométrie discrète, calcul en nombres entiers et algorithmique. Habilitation, Université Louis Pasteur de Strasbourg (1991)
21. Sapiro, G., Ringach, D.L.: Anisotropic diffusion of multivalued images with applications to color filtering. IEEE Trans. Image Process. **5**(11), 1582–1586 (1996)

22. Selesnick, I.W., Baraniuk, R.G., Kingsbury, N.G.: The dual-tree complex wavelet transform—a coherent framework for multiscale signal and image processing. IEEE Signal Process. Mag. **22**(6), 123–151 (2005)
23. Shensa, M.: Wedding the à trous and Mallat algorithms. IEEE Trans. Signal Process. **40**(10), 2464–2482 (1992)
24. Soulard, R., Carré, P.: Color extension of monogenic wavelets with geometric algebra: application to color image denoising. In: ICCA9 Proceedings, Weimar, Allemagne, p. 179 (2011). 10 pages
25. Soulard, R., Carré, P.: Color monogenic wavelets for image analysis. In: Proc. IEEE Int'l Conf. on Image Processing, Brussels, Belgium, pp. 277–280 (2011)
26. Tschumperlé, D., Deriche, R.: Vector-valued image regularization with PDEs: a common framework for different applications. IEEE Trans. Pattern Anal. Mach. Intell. **27**(4), 506–517 (2005)
27. Unser, M., Sage, D., Van de Ville, D.: Multiresolution monogenic signal analysis using the Riesz–Laplace wavelet transform. IEEE Trans. Image Process. **18**(11), 2402–2418 (2009)
28. Unser, M., Van de Ville, D.: The pairing of a wavelet basis with a mildly redundant analysis via subband regression. IEEE Trans. Image Process. **17**(11), 1–13 (2008)

Chapter 16
On Image Matching and Feature Tracking for Embedded Systems: A State-of-the-Art

Edwige E. Pissaloux, Steve Maybank, and Ramiro Velázquez

Abstract This chapter presents a state-of-the-art on image and feature matching in 2D and 3D. Only methods suitable for embedded or wearable real-time system implementation are considered. The implementation may be supported by a dedicated VLSI system. Heuristic guided predictive approaches to image matching are classified as area-based or feature-based. Correlation-based matching, Fourier matching, and mutual information approaches are area-based. Graph, series, and their combinations, including pyramidal or multiresolution algorithms, are feature-based. First, relaxation, maximal clique, tree search, region growing, and dynamic programming methods are briefly described. Next, the correlation-based methods, with a fixed size or adaptive sized window, pyramidal methods, the iterative closest point (ICP) algorithm, and probability (saliency)-based approaches are sketched. Some hardware architectures which support these methods offer new computational models for image matching and image processing. Methods for feature tracking are split into two classes: correlation-based methods and Bayesian methods. Kanade–Lucas–Tomassini (KLT), three-steps/new-three-steps, four-steps, diamond efficient search, and some of their new extensions with inertial data represent the first class, while Kalman and other filters, and their recent improvements represent the second class. The importance of matching is attested by the wide number of applications which include robot navigation, navigation assistance for impaired people, navigation in virtual systems, the processing of medical, satellite and urban imagery,

E.E. Pissaloux (✉)
ISIR (Institut des Systèmes Intelligents et de Robotique), UPMC (Université Paris 6), Paris, France
e-mail: Edwige.Pissaloux@upmc.fr

S. Maybank
Department of Computer Science and Information Systems, Birkbeck College, University of London, London, UK
e-mail: sjmaybank@dcs.bbk.ac.uk

R. Velázquez
Mecatrónica y Control de Sistemas, Universidad Panamericana, Aguascalientes, Mexico
e-mail: rvelazquez@ags.up.mx

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5_16,

human computer interaction, stereo vision, 3D reconstruction, multimodal fusion. processing, remote sensing, etc.

16.1 Introduction

A digital image is a representation of a scene by a regular grid of numbers. The places in the grid are pixels and the numbers are the pixel values. Each number codes the intensity of the signal received from a small part of the scene. An image sequence records the changes in the scene over time, for example, changes in illumination or changes due to relative motion between the sensing device and the objects in the scene.

Image processing includes both local and global operations on images. The most important local operation is the construction of features which summarize the information in small regions of the image. Important global operations include segmentation and the matching of features between consecutive images in a sequence of images. The applications of feature matching include stereo vision, structure from motion, and the detection and tracking of moving objects.

There are many different types of feature, however, hardware limits and recent results in psycho-cognitive vision suggest that point features are good candidates for high quality image processing which can be carried out on dedicated (parallel) hardware. Each point feature is a single location in the image such that the values of the surrounding pixels have some easily identified property. Point features can be efficiently and reliably matched or tracked.

The applications of image and feature matching and tracking are very numerous. Even so, more than 50 years of research have not resulted in unique algorithms for matching and tracking. Indeed, vision algorithms usually depend heavily on specific heuristics which are tailored to the application. However, despite the many papers and communications on tracking and matching published in recent years [103], few of them take into account the constraints on implementation found in embedded or wearable systems. This chapter overviews some methods for image matching and feature tracking which are implemented in academic or research prototype systems or which could be implemented in hardware suitable for an embedded or wearable system. The algorithms are not presented in detail. The basic concepts of each algorithm are described and the effects of their system integration on the quality of the results are discussed.

The rest of the chapter is organized as follows: Sect. 16.2 addresses the matching concept, extraction of the interest/saliency points as basic primitive for targeted hardware implementation, the most popular algorithms for image matching, and hardware supports for image/vision matching operations based graphs. Section 16.3 discusses the feature tracking problem, correlation, and Bayesian approaches. Section 16.4 provides some final comments on the considered approaches of matching and tracking, and some potential research points for new hardware developments.

16.2 Matching: Concepts, Algorithms, and Architectures

Matching and tracking are essential functions for many vision-based processes. However, they still present difficult problems, especially for implementation in embedded or wearable systems.

16.2.1 Matching: Basic Concepts

In general, matching deals with the identification of certain attributes or characteristics associated with a given relationship. As far as image matching is considered, this definition can be reformulated as follows: matching two images of the same (2D or 3D) scene involves the identification of the geometric (2D or 3D) transformation that superposes one image onto another. The parameters of the transformation may be estimated from matches between selected features, sometimes referred to as control points.

The mathematical model of the geometric transformation depends on the image acquisition sensors, the required accuracy, and any bounds on algorithmic complexity. The image acquisition system may produce noisy data subject to geometric distortions. If the error model is known, then it is useful to pre-process images with an error reverse model. In the case of noise, a Gaussian model is often appropriate even if the true but unknown distribution is known not to be Gaussian. The most common geometric image distortions, i.e., deviations from rectilinear projection, are radial and tangential distortions. The former include quadratic, barrel, and pincushion distortions. These distortions can be corrected with the Brown distortion model [14]. Camera calibration can include distortion error correction [2, 97, 101].

16.2.2 Characteristics for Matching

Matching can be performed in the space domain or in the frequency domain. In applications which require fast matching using parallel hardware, matching in the space domain is preferred.

Image matching methods are classified as area based or feature based, as follows:

1. Area-based methods: matching is carried out using a direct comparison of pixel values. A dense set of image matches is often sought.
2. Feature based methods vary according to the selected features. Possible features include:
   - Locally salient features, for example, local extrema, in which an image region differs in some systematic way from its neighboring regions [61, 82, 93].
   - Predefined features in which a feature is defined by a specific property of a set of pixels, for example, a standard deviation in a particular range or a particular texture [42, 61].
   - Application specific features [33, 64], for example, faces and gestures.

Features defined using local extrema are potential candidates for implementation in embedded hardware. The following features are considered: edge points, interest points (local curvature points, inflection points, application specific points, corners), and salient points/regions.

Almost all edge detectors use edge models first defined in the 1970s [25, 41]. Edges are mainly detected using gradient based operators such as Sobel, Kirsch, Prewitt, Nagao, Canny, and Deriche [30]. Laplacian based edge detectors discard the gradient orientation; therefore, they are less useful for subsequent image processing such as edge following.

The definition of interest points can be based on properties of the human visual system [63].

The proposed methods differ in the way in which the interest measure is calculated and the way in which the interest points are detected. The interest measure can be based on the gradient distribution in the neighborhood of a pixel, quantified by image local auto-correlation [63], a nonuniform variation of the local gradient orientation [48], curvature discontinuities [5, 56], curve inflections, which are frequently exploited in medical imagery [29], and corners.

Interest point detectors are classed as raw intensity based [31, 38, 48, 63, 83, 85], contour based [5, 46], and parametric [91, 104].

Detectors are often designed to be invariant to translation, rotation and changes in illumination. It is possible to define detectors which are scale invariant and invariant to affine transformations.

The performance of an interest point detector is measured by its repeatability and stability under changes in image geometry or illumination [82] and by the distribution of the detected image points [26, 99].

The Moravec corner detector [63] is invariant to translation and rotation, but the location of the interest point is imprecise and unstable. Harris and Stephens [38] and Schmid et al. [82] have stabilized Moravec's corner detector and improved the precision with which the interest point is located. Tuytelaars and Mikolajczyk [93] have made the Harris–Stephen's detector invariant to affine transforms.

The Smith and Brady [85] SUSAN operator (SUSAN = smallest univalue segment assimilating nucleus) is based on the ranking of the pixel values in the neighborhood of a potential corner. It is computationally simple and could be implemented in hardware. However, Tissainayagam and Suter [90] have shown that SUSAN is less stable than the Harris–Stephen's detector and the localization of the interest points is less accurate. Figure 16.1 shows the detection of interest points with Harris–Stephens detector in two images which differ by a 90° rotation and illumination. Both images have the same depth in the field of view. It is easy to check visually (red circles) that all the corners in the chess board have been detected but their localization is poor.

Sojka [86] has proposed a corner detector based on Bayesian estimation of the variance of the gradient directions in a neighborhood. The corner detector is efficient and fast. Chen and Liu [18] have proposed a Monte Carlo method based on wavelets and the first derivative of a Gaussian that allows robust detection of corners and line intersections. Trujillo and Olague [91] have proposed a genetic programming approach to synthesize stable interest point detectors.

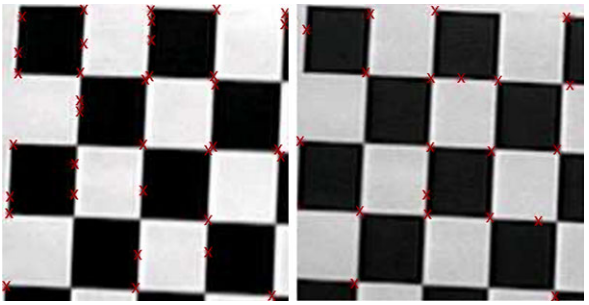


Fig. 16.1 Interest points detected with the Harris–Stephens detector applied to the image of a plane surface parallel to the image plane. Both images are taken at the same depth

Some authors have introduced local region descriptors to quantify the interest of a point. Lowe [54] has defined the scale invariant feature transform (SIFT) which combines a detector of scale invariant regions and a descriptor based on the gradient distribution in detected regions. SIFT is robust to local geometric distortions. Ke and Sukthankar [44] have introduced a PCA descriptor; Trujillo et al. [92] defined a region descriptor based on image local singularity.

More recently, the concept of the saliency of a region has been redefined. It is still linked to the human perceptual distinctiveness of a region but this distinctiveness is not confined to regions with high grey levels or high grey level gradients [42, 61]. The definition of saliency is based on the probability of matching two regions from different images [61]. The definition of saliency can be made quantitative using the Kullback–Leiber divergence between two conditional probability density functions (pdfs), one defined on pairs of regions that do not necessarily match and one defined on pairs of regions which are known to match.

16.2.3 Popular Matching Methods

Several classes of image matching methods exist [15, 100, 103]. They can be classified according to the image characteristics used for matching, for example, raw pixel values, spatial distribution of features, and symbolic features. The symbolic features are outside the scope of this chapter.

An image matching method usually has two steps: feature matching and estimation of the mapping function. These two steps can be carried out sequentially or in parallel. Methods for estimating mapping functions are not considered here.

16.2.3.1 Raw Data/Area Based Methods

Matching methods based on raw pixel values can be subdivided into three classes [103]: correlation (or template) matching, Fourier based, and the mutual information based methods.

The correlation methods are based on the photometric properties of images. A sub-image, referred to as a window or a correlation kernel, is chosen in the

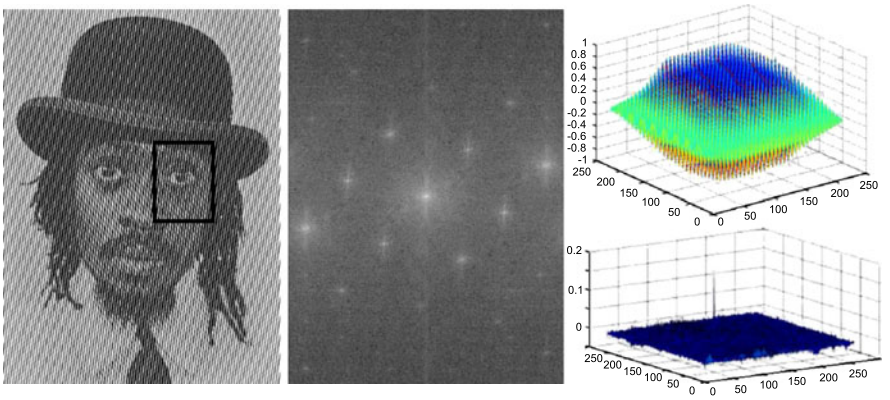


Fig. 16.2 The relative simplicity of a Fourier transformed image simplifies the phase correlation matching. *From left to right*: original image with periodic noise and a subimage (*square*), Fourier representation of the original image, (*top*) ZNCC correlation, (*bottom*) phase correlation

first image. The correlations between this sub-image and a set of sub-images in the second image are obtained. The sub-image in the first image is matched with the sub-image in the second image for which the highest value of the correlation is obtained. Different measures could be used [30], for example, sum of absolute differences (SAD, ZSAD), sum of squared differences (SSD, ZSSD), cross-correlation (NCC, ZNCC), etc. The results are good if the two images are acquired under similar illuminations and the geometric transformation between the images is small. If constraints on the matching are known, for example, the epipolar geometry in stereo images, or constraints associated with known geometric structures in the two images, then it is possible to obtain the correct matches using correlation even if there are significant variations in illumination or image geometry [28, 95], sometimes with subpixel precision [1, 47, 53, 78, 94]. Hu and Ahuja [40] compute correlations in several directions, under the linearity hypothesis between two adjacent directions, in order to make correlation based matching invariant to rotation.

Fourier representation based methods (Fig. 16.2) for image matching are useful when the images are acquired in difficult conditions involving high levels of noise or time varying illumination or when fast matching is required.

Matching is carried out by searching for the maximum of the inverse of the cross-power spectrum of the two images I1 and I2 (Eq. (16.1)):

$$\frac{F(I1)F(I2)^*}{|F(I1)F(I2)^*|} = e^{2\pi i(ux_0+vy_0)}. \tag{16.1}$$

The proposed approaches are based on the Fourier shift theorem applied to phase correlation. Images can be simply translated with respect to each other or rotated [27], or scaled [10, 45, 51, 77]. Fourier matching allows the use of kernels larger than those used for correlation based matching in the space domain, while still retaining the ability to find the correct matches quickly, especially if the Fourier based matching is carried out on specialized digital signal processors.

Mutual information methods, initially applied in medical imagery [72], have recently been extended to the matching of arbitrary objects [21, 87]. The mutual information $MI(X;Y)$ measures the amount of information that one random variable (image X) contains about another random variable (image Y). It is defined by Eq. (16.2):

$$MI(X;Y) = H(Y) - H(Y/X) = H(X) + H(Y) - H(X;Y) \qquad (16.2)$$

where $H(X)$ is the entropy of an image X, $H(Y)$ is the entropy of Y, and $H(Y|X)$ is the entropy of Y conditional on X [72]. The best image match corresponds to the maximum value of the mutual information. The maximization can be carried out using any of several optimization techniques, sometimes combined. The techniques include gradient descent, Marquardt–Levenberg, hierarchical simulating annealing, and multiresolution optimization [18, 89]. Cole-Rhodes et al. found that the time complexity of image matching based on mutual information was one third of the time complexity of image matching based on correlations [21].

16.2.3.2 Feature Based Approaches

Feature based methods often make use of mathematical structures such as graphs, combinations of graphs and series, and pyramids. They usually produce a sparse set of correspondences.

Features are usually represented by graph vertices while the edges of the graph represent relations between features such as adjacency, order, or relative position. Pyramids can be regarded as combinations of series and graphs; in fact, they are used for multi-resolution image processing.

In this context, feature matching approaches are based on graph or pyramid data structure processing such as traversal, matching, and indexing. Matching methods based on graphs [16, 22, 35] aim to find a correspondence between the nodes and edges of two graphs. The correspondences might define a graph isomorphism or an isomorphism between sub-graphs. The most frequently used graph algorithms in computer vision are relaxation, maximal cliques, tree search, region growing, and multi-resolution correlation with fixed or adaptive windows. In addition, dynamic programming can be considered as a 1D graph matching.

Relaxation methods can be considered as a special case of the consistent labeling problem. Targeted features of both images are labeled and combinatorial methods are used to search for feature correspondences that are compatible with the labeling [76, 79]. Geometric and physical constraints on the features are used to reduce the complexity of the search. The relaxation is usually implemented as follows: (i) a set of pairs of vertices is selected and a confidence measure is assigned to the correspondences defined by these pairs; (ii) the matching is extended to pairs of vertices outside the original set of pairs. The correspondences are iteratively modified in order to find the best set of pairs of matching vertices. Ranade and Rosenfeld [76] express the cost of a match using the geometric displacement between sets of features. Their method is invariant to translation and local distortions. The relaxation works

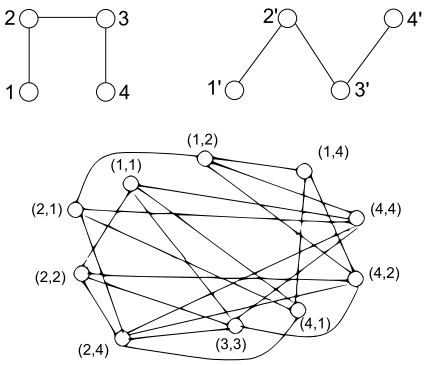


Fig. 16.3 Two graphs and their association graph

well for global image features, such as the boundaries between strongly contrasting regions [52]. However, the combinatorial complexity of these methods is so high that, for real time implementations, it is necessary to introduce contextual heuristics such as uniqueness and continuity [57], reinforced disparity gradient [74], or local consistency measure [84]. Probabilistic relaxation approaches have also been proposed, for example, in [20].

The maximal clique method is based on the concept of an association graph. A clique in a given graph is a completely connected subgraph. Let G_1 and G_2 be two graphs of features obtained from image 1 and image 2, respectively. A vertex of the association graph of G_1 and G_2 is a pair of vertices, one from G_1 and one from G_2, such that the two vertices have the same labels.

An edge in the association graph links two of its vertices if the relation between two vertices linked in G_1 is the same as the two vertices linked in G_2 (Fig. 16.3).

The search for the best match of the features from the two images reduces to a search for a maximal clique in the association graph [3]. The main disadvantage of this approach is that the task of finding a maximal clique is NP-hard. A practical solution consists of reducing the number of features, and thus the size of the graphs. The basic algorithm has been extended to trees, both rooted and unrooted, in [67, 68].

The tree-search algorithm is frequently based on the graph traversal operation. The depth-first search (or branch-and-bound) algorithm [24], for example, matches features of the first image with features of the second image one by one for as long as possible. On the first mismatch of a new feature, the algorithm backtracks to the previous match. In order to improve the performance of the basic algorithm, the authors of [37] suggest various heuristics such as forward checking to try and avoid vertices where matching will fail, and the recording of previous failed searches.

Dynamic programming (DP) [6, 8] can be considered as a search for a minimum cost path in a graph with weighted edges. The N features of interest from the first and second images are listed on the X and Y axes, respectively, as shown in the left most picture in Fig. 16.4. The task is to find a matching between the two sets of N features that minimizes a cost function. Each matching defines a path, for example, as shown by the dark line in the left hand picture in Fig. 16.4. The path with the

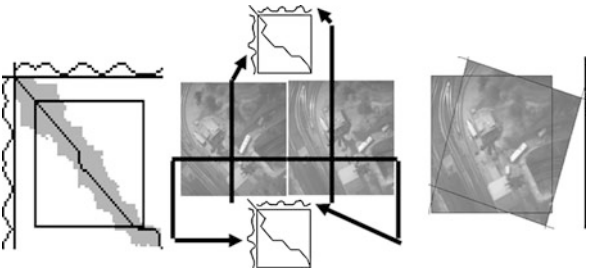


Fig. 16.4 Principle of the orthogonal dynamic programming for image (dense) matching and results

minimal cost is calculated by backtracking. Different constraints, for example, a requirement that the path be monotonic [30], can be added in order to speed up the search. DP has been recently applied for speech processing [75].

The further extension of DP to (two-dimension) orthogonal DP is based on a possible decomposition of nonlinear problems, with all variables not necessary inter-independent, into a sequence of sub-problems with some of the variables separated so that they do not appear together in the same sub-problem. The orthogonal DP (Fig. 16.4 central picture) builds a field of the local orthogonal displacements at every pixel. The field is smoothed with an appropriate algorithm to tune the parameters of the global 2D transformation which links the matched images.

The effectiveness of this approach is shown by the matching achieved in the rightmost picture in Fig. 16.4 [69, 70]. The associated cost function "compresses" the luminosity range in the matched images. The speed of the DP algorithm can be increased using dedicated hardware.

Correlation-based methods can produce a dense set of correspondences between two images. Two pixels, one from each image, are matched if the neighborhoods of the pixels are similar. The similarity of two neighborhoods is measured using the correlation score. The size and shape of the neighborhoods influence the quality and the time complexity of the matching. Typically, square regions of size 3×3, 5×5, or 5×7 are used. The matching between images I_1 and I_2 is found as follows: A pixel is selected in I_1. A square region of an appropriate size $m \times m$ and centered on this pixel is chosen. This region is referred to as the correlation kernel. A region of interest (ROI) in I_2 is defined, and for each given pixel in the ROI, the correlation between the $m \times m$ square of pixels centered on the given pixel and the correlation kernel is found. The pixel in I_1 is matched with the pixel in the ROI for which the highest correlation is obtained. This matching process is continued for all the pixels of interest in I_1.

The correlation approach is also used with adaptive correlation kernels in which the size and shape of the kernel are functions of the local variation of the intensity and of the disparity between previously matched pixels [7, 12, 32, 34, 43, 78, 81]. The algorithm for adaptive correlation matching proceeds in two steps: (i) the initial disparity D_0 is estimated using correlation with a non-adaptive kernel (usually 3×3), then (ii) for each pixel, the window size and disparity are updated, $D_1 = D_1 + d$. Step 2 is usually implemented iteratively, until the D_i converges or until the maximal (predefined) number of iterations has been reached as follows:

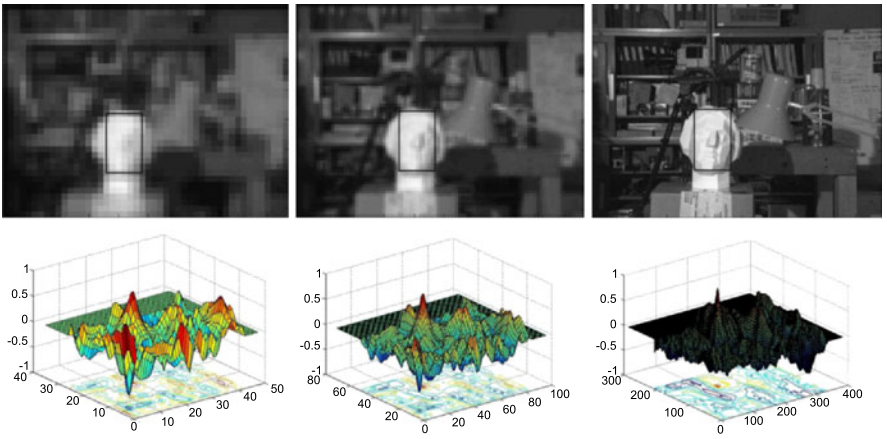


Fig. 16.5 Pyramidal matching (*top*) and corresponding correlation (*bottom*). *From left to right*: lowest level, intermediate level, and highest level

(a) the uncertainty of the disparity is calculated over the window centered on the current pixel; (b) the window is extended in four orthogonal directions $(+/-x, +/-y)$ and the uncertainty of the disparities of the four new windows is calculated, (c) the window is extended in the direction of the smallest uncertainty.

Region growing proceeds as follows: Image seeds, such as interest points, are selected based on their correlation scores. Local geometric constraints are used to guide the matching process. The final matches are checked by estimating the global mapping function. A match is accurate if it agrees with the match predicted by the mapping function. Pyramidal matching methods [17, 34] combine graphs and series' operations. They use images at different resolutions in order to reduce the time complexity of matching. Figure 16.5 shows the result of the correlation within a 3-level pyramid of the same image; at each level, a single correlation kernel is used. The matched features at the lower resolution levels guide the search for matching features at higher resolutions. The main disadvantage of pyramidal multiresolution approach is that matches may be lost entirely at the low resolution levels and may not be found during the progress to the high resolution levels. An edge improvement technique for pyramids has been proposed in [66].

Correlation-based methods have several drawbacks including sensitivity to noise, scene clutter, variations in texture, occlusions, perspective distortions, illumination, and view angle changes [58]. However, they are efficient in many applications.

Several probabilistic approaches to image matching have been proposed recently. Maybank [61] obtains two feature vectors $v(1)$, $v(2)$ from regions in the first and second images, respectively. Let B be the background hypothesis that $v(1)$ and $v(2)$ are obtained from independent image regions and let H be the hypothesis that $v(1)$ and $v(2)$ are obtained from matching image regions. The probability density functions $p(v(1), v(2)|B)$ and $p(v(1), v(2)|H)$ are learnt using one or more training images. The saliency of a given image region with feature vector $v(1)$ is, by definition, equal to the Kullback–Leibler divergence of $p(v(2)|v(1), B)$ from $p(v(2)|v(1), H)$.

It follows that the given image region has a high saliency if $v(1)$ contains a significant amount of information about the feature vector of the matching image region. A match to the given image region can be selected using a log likelihood ratio. In detail, let $v(2,1),\ldots,v(2,N)$ be a set of vectors obtained from N candidate matches to the given image region. The best match $v(2,j)$ is defined such that:

$$j = \operatorname{argmax} i \mapsto \ln\bigl(p\bigl(v(2,i)|v(1),H\bigr)/p\bigl(v(2,i)|v(1),B\bigr)\bigr). \tag{16.3}$$

Experiments with stereo images show that the accuracy of matching based on the above log likelihood ratio is similar to the accuracy of matching based on the sum of absolute differences of pixel values.

The iterative closest point (ICP) algorithm [9, 19] is one of the most popular recent algorithms for 3D point matching. The algorithm makes an iterative search for the best 3D transform which links two sets of 3D features, for example, points, curves, or surfaces. Each iteration has two steps, namely feature matching and 3D transform estimation. The list of the matched points found in the current iteration is used as input for the next iteration. The algorithm finishes when the distance error between the matched points is less than a predefined threshold. The choice of an initial 3D transform is difficult because it must be accurate enough to ensure that the algorithm does not yield a locally optimal solution which is not globally optimal. Several improvements have been proposed. Weik [96] restricts ICP to points at which there is a large luminosity gradient. Masuda et al. [60] select at each iteration a random subset of points with few outliers. Rusinkiewicz and Levoy [80] have investigated the relationship between the convergence of the ICP and spatial distribution of the mesh points. They showed that the random subset should be replaced by the normal-space sampling in order to obtain high precision real time algorithm (recall: in normal-space sampling points are chosen such that the distribution of normals among the selected points is as large as possible).

16.2.4 Hardware Systems for Image Matching

Several "universal" hardware systems exist for efficient image processing and analysis. Three systems which efficiently support graph-based processing are presented. These are the μDP circuit for DP matching, the Sphinx pyramid for multiresolution processing, and the MAO for image processing and vision graph operations.

The μDP circuit (Fig. 16.6, left) simulates in hardware the local path parallel development (Fig. 16.6, right) when comparing two vectors. μDP is a 2D mesh of $N \times N$ elementary processors (PE_{ij}) each evaluating the local cost of matching two selected features. In order to be able to develop paths in three possible directions, two orthogonal and one diagonal, for the global matching of a line, each PE is 3-connected to its east, south, and south-east neighbors.

Suppose that it is required to match two vectors U, V. The calculations of all distances $d_{ij} = |U_i - V_j|$ between any two components can be performed in parallel, leading to a reduction of the sequential complexity of the matching algorithm

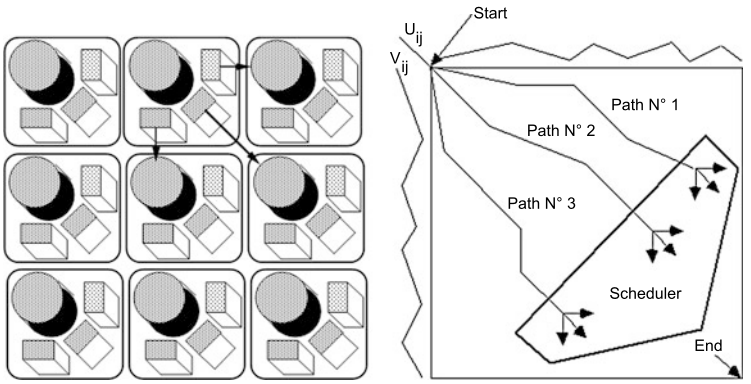


Fig. 16.6 The internal architecture of the μDP circuit and its operation during matching (*left*: path parallel development)

to $O(N)$. Each PE has necessary calculation and memory resources for calculating values of the form d_{ij}. Moreover, each PE remembers the PE which activated it. This information is useful for backtracking in order to find the optimal match. RECEPTION and ACTIVATION are two signals implemented in each PE for a data asynchronous exchange protocol. The whole matching algorithm is executed in $2N$ steps in the worst case.

The code below gives the main steps of the matching algorithm. The TIMER variable records the temporal length (equivalent to the matching cost) of the best global path found so far.

```
/* local cost calculation in forward direction */
   PE(0,0)=ACTIVE; /μPD calculation starts by external activation of PE(0,0) */
   FIN=FALSE;
   Score_temp := 0 ;
        WHILE (NOT FIN) DO IN PARALLEL on ALL Active PEs
                IF RECEPTION(Score_temp) from neighbor THAN
                     TIMER := score_temp;
                     MEMORIZE the direction s of activation;
                     UP DATE TIMER_temp in 3 directions;
                     WAIT until TIMER=dij.C(diagonal) THAN
                     ACTIVATE & FORWARD TIMER_temp to diagonal neighbor;
                     WAIT until TIMER=dij.C(orthogonal) THAN
                     ACTIVATE & FORWARD TIMER_temp to orthogonal neighbors;
                     FIN=IF(no_more PE) THAN TRUE;
                     DEACTIVATE yourself;
                END IF;
        END WHILE
   SCORE= TIMER (N,N);
   /* Backtracking of the optimal path; POSITION gives the PEij through which optimal path passes */
   i=2*N;
   PATH(i)= (POSITION=(N,N));
   WHILE (NOT PATH(i)=(0,0)) DO
        POSITION=POSITION-s(POSITION);
        i=i-1;
        PATH(i)=POSITION;
   END WHILE /* PATH = optimal path */
```

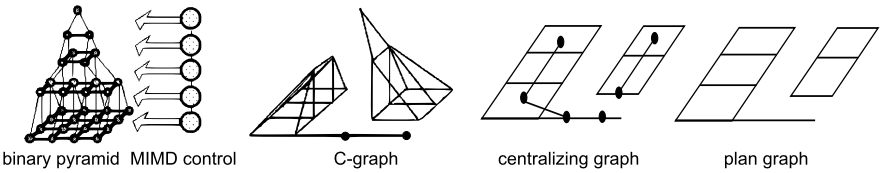


Fig. 16.7 The Binary MIMD pyramid and examples of its basic data structures: C-graph (communication graph), centralizing graph, and plan graph

Multiresolution processing is naturally supported by hardware pyramids. Figure 16.7 shows a binary pyramid [11, 65] which consists of a parallel computer organized as a series of stacked layers of processing elements. The layers decrease in size and are interconnected by a mesh-based (matrix) four-neighbor interconnection network within a layer, and a pure binary tree network between layers. In multiresolution processing, each layer can process one image of a given resolution independently of the other layers. The basic data structures include partial or subpyramids and communication graphs which are referred to as C-graphs. C-graphs allow the application in parallel of the same operation to all graph nodes. Data can be exchanged in parallel between two C-graphs which are embedded in the same layer of the pyramid. A centralizing graph is formed at the pyramid base by pyramidal projection of all nodes involved in a given process (a centralizing node has no ancestor).

Communication operations allow mesh (matrix) plane communications and binary tree traversal operations (send-down, send-up) with possible local computations using any associative and commutative operator such as AND, OR, $+$, or $*$.

The MAO (Maille Associative d'Orsay) is a reconfigurable matrix computer for operations on graphs representing a parallel variable [62]. Figure 16.8 shows the matrix (2D) organization and communication capability of each basic processing element (PE). Each PE can be connected to eight of its neighbors; the effective connections are selected using a local network configuration register. This is a simple support for dynamic graph connectivity, useful for feature matching and for selective application of an operation (called association) to graph connected component. An association modifies the value of a parallel variable v by combining, using an associative and commutative operator, the values of all antecedents u of v. (The variable u is an antecedent of v if there is a path from u to v).

16.3 Feature Tracking

In feature tracking, the task is to follow a feature from one image to the next in a time sequence of images. The tracking is easier if the time interval between successive images is small. Feature tracking is important in automatic surveillance, motion capture, gesture recognition, vehicle guidance, targeting, and human computer interaction through unconventional interfaces such as brain/body computer interfaces or gaze based interfaces.

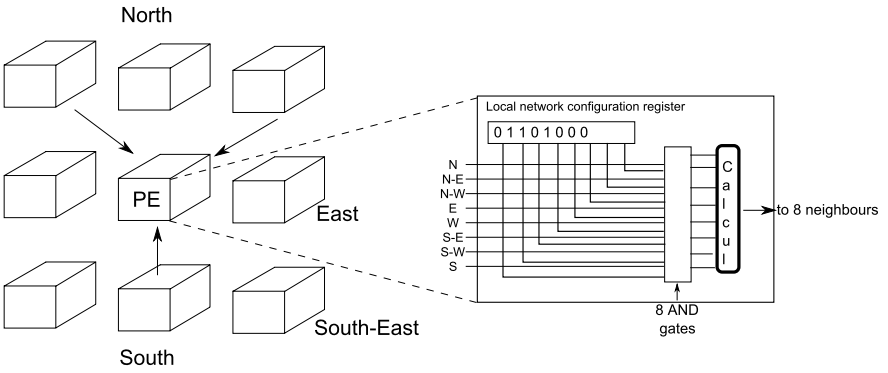


Fig. 16.8 MAO architecture for dynamic operation of graphs

This section contains a short overview of the most popular methods for feature tracking, especially those which can be implemented in embedded hardware. These methods are essentially correlation-based but they may use in addition qualitative motion heuristics and Bayesian reasoning.

16.3.1 Correlation-Based Feature Tracking

In general, it is very difficult to estimate displacements from one image to another. The difficulties are caused by changes in illumination, the complexity of the displacements, and the effects of un-modeled noise. One solution is to track salient points rather than an entire image region, on the grounds that salient points can be matched even if the noise level is high.

One of the most popular methods for tracking is proposed by [55]. The method involves the matching of relatively small squares of pixels from one image to the next. The displacements are assumed to be small translations, usually of the order of a pixel. This assumption is reasonable if the movement is slow compared to the camera sampling speed. The windows most suited to matching are those in which the eigenvalues of the matrix in Eq. 16.4) are above a threshold [83] and have a weak dynamics:

$$G = \int_F g g^T w \tag{16.4}$$

where F is a given window, g is the luminosity gradient in F as a function of position, and w is a weighting function. The displacement of each window is determined by Eq. (16.5):

$$Gd = \int_F h g w \tag{16.5}$$

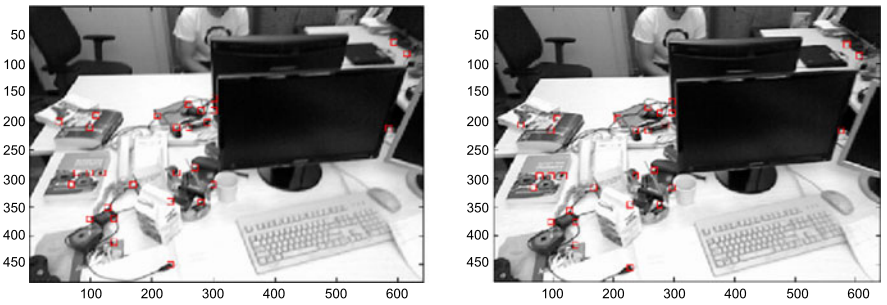


Fig. 16.9 Match between regions with a large displacement, obtained using KLT tracking with a window size of 10×10 pixels^2. The score is the sum of the squared differences of the pixel grey levels

where h is the luminosity gradient between two successive images in the considered pixel, and d is the required displacement vector.

Shi and Tomasi [83] proposed the KLT tracker which iteratively computes the translation vector of a region centered on an interest point. A typical region size is 25×25 pixels^2. The region is compared to all candidate regions in the next image. The comparison is carried out using a scalar matching score. The region with the highest score is chosen as the matching region. The score can be based on correlation or on the sum of the squared difference of the pixels' luminosity. In Fig. 16.9, the coordinates of the four windows of the book are (62; 312), (72; 292), (92; 292), and (112; 292) in the first image, and (64:37; 306:19), (74:44; 286; 26), (94:35; 286:16), and (114:66; 286:21) in the second image, respectively, with the standard deviation of the pixel grey level of 13.2, 14.3, 14.0, and 20.1. The displacement of each window is (2:37; −5:81), (2:44; −5:74), (2:35; −5:84), and (2:66; −5:79). Despite the important errors (in the scale of 255), the windows/KLT found displacements are coherent as the windows are close one to another and are still localized on the same physical object.

KLT tracking may fail if the object appearance changes too much or if the object is distorted from one image to the next. The main inconvenience of approaches based on a systematic search of a set of candidate matching regions is their computational cost which is proportional to the number and size of the regions. Optimization is possible if the nature of the expected displacement is known or if a search strategy is adopted. Further optimization is possible if the search is supported by dedicated embedded hardware.

Different heuristics can be used to reduce the number of correlation score calculations. The heuristics include the assumption of a maximum authorized displacement D. Koga et al. [49] describe a three-step search (TSS, 3SS) algorithm which replaces the exhaustive 1-pixel at a time search through the candidate matches by a spatially uniform convolution step with convolution step size of $S = D/2, D/4, D/8, D/16, \ldots$ pixels iteratively until $S = 1$ pixel. The length D is a parameter of the algorithm. The algorithm starts with a feature predicted position (for example, the central black point in Fig. 16.10) and calculates 8 correlations at

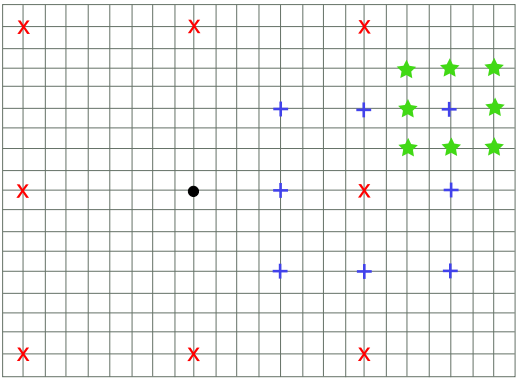


Fig. 16.10 Three-step search (3SS) algorithm for tracking: 3 iterations with $D = 8$, $D = 4$, and $D = 2$ pixels for reaching the correlation maximum

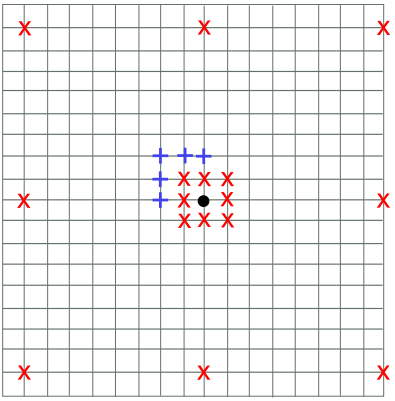


Fig. 16.11 NTSS algorithm convolution principle

distances of $\pm D$ pixels. The algorithm selects the region with the largest correlation score and calculates 8 new correlations at distances of $\pm D/2$ pixels from the center of the selected region. The algorithm finds a correct match under the hypothesis that surface defined by the values of the correlations is unimodal. The main drawback of the TSS algorithm is that small motions can be missed because of the lower bound on the step size and search window size.

A new three-step search (NTSS, N3SS, Fig. 16.11) algorithm [50], implemented in the MPEG1/2/4 and H.261 standards, is more reliable. At initialization, the correlation is calculated at 16 points: at a 3×3 ($S = 1$ from the predicted match) and at a $D \times D$ ($S = D$ from the predicted match) window. If the best correlation is found in the initial prediction, the algorithm ends; otherwise, the point with the maximal correlation score is selected as algorithm's new starting point and the whole correlation process restarts with D replaced by $D/2$ (Note that some of the correlation scores have already been calculated in the previous step).

Po and Ma [73] propose an improvement to NTSS (N3SS) named the four step search algorithm (FSS, 4SS). Two square patterns of different sizes $D = 2$ or $D = 1$ (Fig. 16.12, left and central), but with a common center, are used to select the pixels

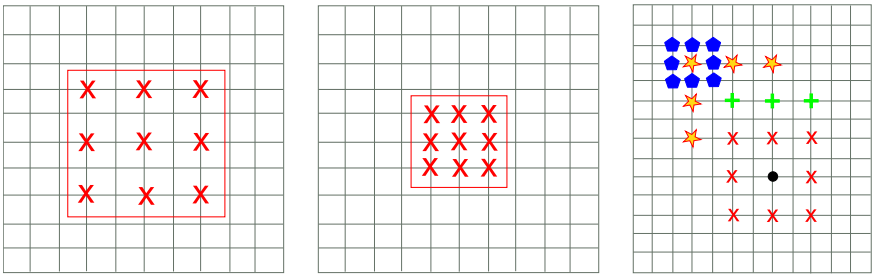


Fig. 16.12 NTSS for image matching with sparse convolution step

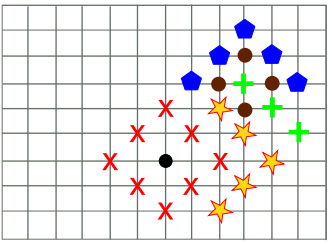


Fig. 16.13 Convolution principle for diamond search algorithm: the search is centered respectively on black, east-red, north-east yellow and north green pixels

where the correlation is calculated. At initialization, the correlation step is 2 pixels ($S = 2$), and the correlation is calculated at nine points in a 5×5 window. If the maximum correlation is obtained at the central pixel, then S is set equal to 1 and the appropriate correlations are calculated. If the maximum correlation is attained at a pixel other than the central pixel, then this other pixel is made a new central pixel and the calculation is repeated with $S = 2$.

The constant number of steps of this algorithm is a very attractive characteristic; however, it often fails to find the correct match. Moreover, all search algorithms based on squares of pixels assume that the displacement vectors are distributed uniformly around the search center. The algorithm proposed by [102], namely diamond search (DS), replaces the squares used by 4SS with diamonds (Fig. 16.13) and proceeds iteratively until the best match is obtained. The expected number of correlations is reduced significantly. The new cross-diamond search (NCDS) [39] uses an asymmetric (horizontal or vertical) diamond search pattern in order to overcome the restrictions on the displacement imposed by search algorithms based on squares and find the best match.

Information from additional sensors, such as an Inertial Measurement Unit (IMU) can be used to guide the selection of matching pairs of pixels. Indeed, as the physical displacements of 3D scene interest points are limited in size from one image to the next, the integration of data provided by IMU to a recursive position estimator allows the prediction of the maximum displacement of scene points relative to the camera, and the estimation of their projected positions in the image. In this way, the size of the search window for potential matches can be reduced and the matching process can be speeded up. This principle is proposed in [23], and evaluated on real

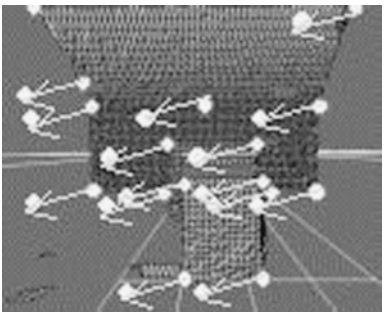


Fig. 16.14 Tracking of interest points in translated images via visuo-inertial method

images where interest points have been tracked (Fig. 16.14) for integration in an assistive device for the visually impaired [71].

Position estimation errors in the presence of occlusions are the main problem for all the above algorithms. Statistical methods can help to solve these tracking problems.

16.3.2 Bayesian Approaches

The position of an object in an image at times $t = 1, 2, \ldots$ is defined by a series of states X^t, $t = 1, 2, \ldots$. The state evolution is modeled by dynamic Eq. (16.6):

$$X^t = f^t\left(X^{t-1}\right) + W^t \tag{16.6}$$

where W^t is white noise. The relationship between a measurement Z^t and the state X^t is given by Eq. (16.7):

$$Z^t = h^t\left(X^t, N^t\right) \tag{16.7}$$

where N^t is white noise independent of W^t.

The state X^t is estimated using all the measurements Z^s, $s = 0, 1, \ldots, t$, obtained up to an including time t. The information in the measurements is summarized by the probability density function (pdf) for the state conditional on the measurements, $p(X^t|Z^1, \ldots, Z^t)$.

A theoretical optimal solution can be obtained using a recursive Bayesian filter which consists of two steps: prediction and correction (update). The prediction step uses the dynamic Eq. (16.6) to infer the pdf $p(X^t|Z^1, \ldots, Z^{t-1})$. The update step uses this pdf and the likelihood function $p(Z^t|X^t)$ to estimate the pdf $p(X^t|Z^1, \ldots, Z^t)$. If the scene contains only a single moving object, if the functions f^t and h^t are linear, and if the initial state X^1 and the noise both have Gaussian distributions, then the pdf $p(X^t|Z^1, \ldots, Z^t)$ is given by the Kalman filter [13].

In the general case, the pdf for the state is not assumed to be Gaussian. The pdf can be approximated using particle filters [4, 88]. There are four steps in particle filtering: particle sampling, prediction, pdf updating, and particle re-sampling, in order to eliminate particles in regions with very low probabilities. The particle filter

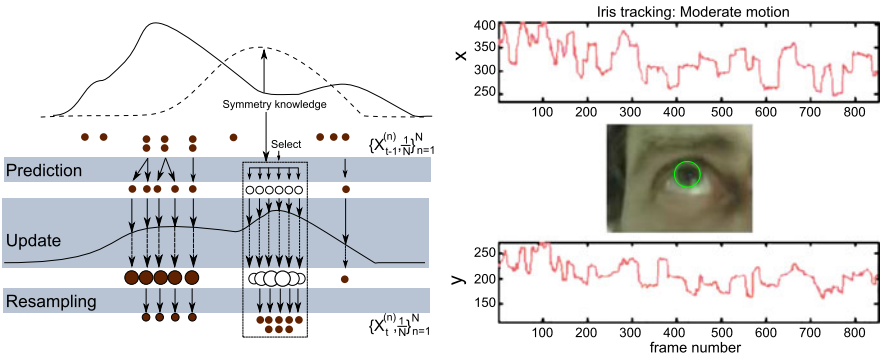


Fig. 16.15 Visible spectrum radial symmetry guided particle filter. *Left image* summarizes the principle of the pdf update. *Right image* shows the tracking results for x and y pupil center position; in *red*: raw true data, in *blue*: the estimated data

can be initialized by either using the first measurement or by training the filter using sample sequences.

Several methods based on particle filtering for feature tracking and for combining object models have been proposed. An application of particle filtering to eye tracking has been recently proposed. Hansen and Pece [36] combine a particle filter with a generalized Laplacian for coding grey-level differences, and a Gaussian distribution for deformation modeling; Wu et al. [98] use a particle filter and a 3D model for the eye; Martinez [59] combines a particle filter with the radial symmetry in visible spectrum images of the eye in order to update the associated pdf (Fig. 16.15).

A priori knowledge of the targeted object can be provided during an initialization stage using interest point detection. However, all these methods are not robust against occlusions or changes in the appearance of the object and have a high computational complexity. There is no suitable hardware for real time processing.

16.4 Final Comments and Potential Future Developments

This chapter has described algorithms for image and image sequence matching. Many of these algorithms are suitable for embedded hardware implementation. The new hardware oriented computational structures can be used not only for image matching, but for image processing and image analysis in general.

Image matching, despite its long history and significant progress, is still an active research area. 2D matching is better understood than 3D matching, however, 2D matching still lacks pertinent approaches to deal with occlusion and to include additional information about the matching.

There are very few efficient methods for 3D–3D matching and for 2D–3D matching. For future wearable and autonomous systems, it seems necessary to build the matching model using additional sensors in order to obtain reliable feature tracking

and image matching. Neuroscientific knowledge will probably offer new insights for image matching.

Single object tracking can use powerful techniques such as SVM or other regression methods. However, these methods have a high computational complexity.

References

1. Ackermann, F.: Digital image correlation: performance and potential application in photogrammetry. Photogramm. Rec. **64**(11), 429–439 (1984)
2. Alvarez, L., Gomez, L., Sendra, J.R.: Algebraic lens distortion model estimation. Image Process. On Line (2010)
3. Ambler, A.P., Barrow, H.G., Brown, C.M., Burstall, R.M., Popplestone, R.J.: A versatile computer controlled assembly system. In: IJCAI, pp. 298–307 (1973)
4. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filter for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
5. Asada, H., Brady, M.: The curvature primal sketch. IEEE Trans. Pattern Anal. Mach. Intell. **8**(1), 2–14 (1986)
6. Baker, H.H., Binford, T.O.: Depth from edge- and intensity-based stereo. In: Proc. of JCAI, pp. 631–636 (1981)
7. Becker, F., Wieneke, B., Yuan, J., Schnorr, Ch.: Variational approach to adaptive correlation for motion estimation in particle image velocimetry. In: Rigoll, G. (ed.) Pattern Recognition. LNCS, vol. 5096, pp. 335–344 (2008)
8. Bellman, R.: The theory of dynamic programming. Bull. Am. Math. Soc. **60**, 503–516 (1954)
9. Besl, P., McKay, N.: A method for registration of 3D shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992)
10. Bigot, J., Gamboa, F., Vimond, M.: Estimation of translation, rotation, and scaling between noisy images using the Fourier–Mellin transform. SIAM J. Imaging Sci. **2**(2), 614–645 (2009)
11. Bouaziz, S., Pissaloux, E., Merigot, A., Devos, F.: Some hardware and software considerations for the multi-SIMD control strategy of massively parallel machines. In: Proc. of IEEE CompEuro 91 Conf., pp. 180–184 (1991)
12. Boykov: Fast approximate energy minimization via graph cuts. In: ICCV, pp. 377–384 (1999)
13. Broida, T.J., Chellapa, R.: Estimation of object motion parameters from a sequence of noisy images. IEEE Trans. Pattern Anal. Mach. Intell. **8**(1), 90–99 (1986)
14. Brown, D.C.: Decentering distortion of lenses. Photogramm. Eng. **7**, 444–462 (1966)
15. Brown, L.G.: A survey of image registration techniques. ACM Comput. Surv. **24**, 326–376 (1992)
16. Bunke, H.: Recent developments in graph matching. In: IEEE ICPR (2000)
17. Burt, P., Adelson, E.: The Laplacian pyramid as a compact image code. In: IEEE COM-31, pp. 532–540 (1983)
18. Chen, R., Liu, J.S.: Mixture Kalman filters. J. R. Stat. Soc. B **62**(3), 493–508 (2000)
19. Chen, Y., Medioni, G.: Object modeling by registration of multiple range images. In: Proc. IEEE Conf. on Robotics and Automation (1991)
20. Christmas, W., Killter, J., Petrou, M.: Structural matching in computer vision using probabilistic relaxation. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 749–764 (1995)
21. Cole-Rhodes, A., Johnson, K., Le Moigne, J.: Multiresolution registration of remote-sensing images using stochastic gradient. In: Aerosense (2002)
22. Conte, D., Foggia, P., Sansone, C., Vento, M.: Thirty years of graph matching in pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **18**(3), 265–298 (2004)

23. Corke, P.: An inertial and visual sensing system for a small autonomous helicopter. J. Robot. Syst. **21**(2), 43–51 (2004)
24. Cormen, T.H., Leiserson, C.H., Rivest, R.L., Cliford, C.: Introduction to Algorithms, pp. 540–549. MIT Press and McGraw-Hill, Cambridge (2001)
25. Davis, L.: A survey of edge detection technique. Comput. Graph. Image Process. **4**(3), 248–270 (1975)
26. Davison, A.J., Molton, N.D.: MonoSlam: real-time monocamera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 1052–1067 (2007)
27. De Castro, E.D., Morandi, C.: Registration of translated and rotated images using finite Fourier transform. IEEE Trans. Pattern Anal. Mach. Intell. **95**, 700–703 (1987)
28. Deriche, R., Blaszka, T.: Recovering and characterizing image features using an efficient model based approach. In: Proc. of ICCVPR, pp. 530–535 (1993)
29. Djabelkhir, F., Khamadja, M., Odet, C.: Level set constrained segmentation using local curvature. In: Proc. of ISPA, pp. 152–155 (2007)
30. Faugeras, O.: Tree-Dimensional Computer Vision. MIT Press, Cambridge (1993)
31. Forstner, W.: A feature based correspondence algorithm for image matching. Int. Arch. Photogramm. Remote Sens. **26**, 150–166 (1986)
32. Fusiello, A., Roberto, V., Trucco, E.: Efficient stereo with multiple windowing. In: CVPR (1997)
33. Gao, D., Vasconcelos, N.: Discriminant saliency for visual recognition from cluttered scenes. In: Proc. of Neural Information Processing Systems (NIPS), pp. 481–488 (2004)
34. Giachetti, A.: Matching techniques to compute image motion. Int. J. Image Vis. Comput. **18**(3), 245–258 (2000)
35. Grimson, E.: Object Recognition by Computer: The Role of Geometric Constraints. MIT Press, Cambridge (1990)
36. Hansen, D., Pece, A.: Iris tracking with feature free contours. In: Proc. of AMFG, pp. 208–214 (2003)
37. Haralick, R.M., Elliott, G.L.: Increasing tree search efficiency for constraint satisfaction problems. Artif. Intell. **14**, 263–313 (1980)
38. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proc. of the 4th Alvey Vision Conference, pp. 147–151 (1988)
39. Hongjun, J., Li, Z.: A new cross diamond search algorithm for block motion estimation. In: Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, vol. 3, pp. 357–360 (2004)
40. Hu, X., Ahuja, N.: Feature extraction and matching as signal detection. Int. J. Pattern Recognit. Artif. Intell. **8**(6), 1343–1379 (1994)
41. Hueckel, M.: An operator which locates edges in digitized pictures. J. Assoc. Comput. Mach. **18**(1), 113–125 (1971)
42. Itti, L., Koch, C., Niebur, E.: A model of saliency-based attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. **20**(11), 1254–1259 (1998)
43. Kanade, T., Okutomi, M.A.: Stereo matching algorithm with an adaptive window: theory and experiment. IEEE Trans. Pattern Anal. Mach. Intell. **16**(9), 920–932 (1994)
44. Ke, Y., Sukthankar, R.: PC-SIFT: a more distinctive representation for local image descriptors. In: Proc. IEEE CVPR, vol. 2, pp. 506–513 (2004)
45. Keller, Y., Averbuch, A., Miller, O.: Robust phase correlation. In: Proc. of ICPR, pp. 740–743 (2004)
46. Khan, J.F.: Empirical mode decomposition based interest point detector. In: IEEE ICASSP, pp. 1317–1320 (2008)
47. Kishore, M.S., Veerabhadra Rao, K.: Robust correlation tracker. Sâdhana **26**(3), 227–236 (2001)
48. Kitchen, L., Rosenfeld, A.: Gray-level corner detection. Pattern Recognit. Lett. **1**(2), 54–69 (1982)
49. Koga, T., Linuma, K., Hirano, A., Iijima, Y., Ishiguro, T.: Motion-compensated interframe coding for video conferencing. In: Proc. of NTC, pp. C9.6.1–9.6.5 (1981)

50. Li, R., Zeng, B., Liou, M.: A new three-step search algorithm for block motion estimation. IEEE Trans. Circuits Syst. Video Technol. **4**(4), 438–442 (1994)
51. Liu, H., Guo, B., Feng, Z.: Pseudo-log-polar Fourier transform for image registration. Signal Process. Lett. **13**(1), 17–20 (2006)
52. Long, P., Giraudon, G.: Stereo matching based on contextual line-region primitives. In: Proc. of Int. Conference on Pattern Recognition, pp. 974–977 (1986)
53. Lotti, J.L., Giraudon, G.: Adaptive window algorithm for aerial image stereo. In: Proc. ICPR, pp. 701–703 (1994)
54. Lowe, D.: Distinctive image features from scale invariant key points. Int. J. Comput. Vis. **60**, 91–110 (2004)
55. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proc. of the 7th Int. Joint Conference on Artificial Intelligence, pp. 674–679 (1981)
56. Manjunath, B.S., Shekar, C., Chellapa, R.: A new approach to image feature detection with applications. Pattern Recognit. **29**, 627–640 (1996)
57. Marr, D., Poggio, T.: Cooperative computation of stereo disparity. Science **194**, 283–287 (1976)
58. Martin, L., Hede, P., Leroux, Ch., Pissaloux, E.: Orientation filtering: a fast uncalibrated image matching method. In: Proc. of IEEE Int. Conf. on Signal Processing, pp. 1500–1504 (2006)
59. Martinez, F., Carbone, A., Pissaloux, F.: Radial symmetry guided particle filter for robust iris tracking. In: Proc. of IAPR CAIP (2011)
60. Masuda, T., Sakaue, K., Yokoya, N.: Registration and integration of multiple range images for 3D model construction. In: Proc. of CVPR (1996)
61. Maybank, S.: A Probabilistic definition of salient regions for image matching. Neurocomputing (2012, to appear)
62. Merigot, A.: Associative nets: a graph based parallel computing model. IEEE Trans. Comput. **46**(5), 558–571 (1997)
63. Moravec, H.: Toward automatic visual obstacle avoidance. In: IJCAI, pp. 584–592 (1977)
64. Mudge, T.N., Turney, J.L., Volta, R.A.: Automatic generation of salient features for the recognition of partially occluded parts. Robotica **5**, 117–127 (1987)
65. Ni, Y., Merigot, A., Devos, F.: SPHINX-a VLSI processing element chip for pyramid computer. In: ASIC Seminar and Exhibit, 5-3/1-4 (1989)
66. Paris, S., Hasinoff, S.W., Kautz, J.: Local Laplacian filters: edge-aware image processing with a Laplacian pyramid. ACM Trans. Graph. **30**(4) (2011)
67. Pelillo, M.: Matching free trees, maximal cliques and monotone game dynamics. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1535–1541 (2002)
68. Pelillo, M., Siddiqi, K., Zucker, S.W.: Matching hierarchical structures using association graphs. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1105–1120 (1999)
69. Pissaloux, E., Le Coat, F., Bonnin, P., Bezencenet, G., Durbin, F., Tissot, A.: A very fast dynamic programming based parallel algorithm for aerial image matching. In: SPIE 's 11th Annual Int. Symp. on Aerospace/Defence Sensing, Simulation and Control, vol. 3069, pp. 354–360 (1997)
70. Pissaloux, E., Le Coat, F., Tissot, A., Durbin, F.: An adaptive parallel system dedicated to projective image matching. In: Proc. of IEEE ICIP (2000)
71. Pissaloux, E., Chen, Y., Velazquez, R.: Image matching optimization via vision and inertial data fusion: application to navigation of the visually impaired. Int. J. Image Graph. **10**(4), 545–555 (2010)
72. Pluim, J.P.W., Maintz, J.B.A., Viergever, M.A.: Mutual information based registration of medical images: a survey. IEEE Trans. Med. Imaging **22**(8), 986–1004 (2003)
73. Po, L., Ma, W.: A novel four-step search algorithm for block motion estimation. IEEE Trans. Circuits Syst. Video Technol. **6**(3), 313–317 (1996)
74. Pollard, S.B., Mayhew, J.E.W., Frisby, J.P.: PMF: a stereo correspondence algorithm using a disparity gradient constraint. Perception **14**, 449–470 (1985)

75. Quenot, G.M.: The orthogonal algorithm for optical flow detection using dynamic programming. In: IEEE ICASSP, vol. 3, pp. 249–252 (1992)
76. Ranade, S., Rosenfeld, A.: Point pattern matching by relaxation. Pattern Recognit. **12**(4), 269–275 (1980)
77. Reddy, B.S., Chatterji, B.N.: An FFT-based technique for fast image registration. IEEE Trans. Image Process. **5**(8), 1266–1271 (1996)
78. Remagnino, P., Brand, P., Mohr, R.: Correlation techniques in adaptive template matching with uncalibrated cameras. In: SPIE Int. Symp. on Photonic Sensors and Control for Commercial Applications, vol. III-2356, pp. 252–253 (1994)
79. Rosenfeld, A., Hummel, R.A., Zucker, S.W.: Scene labelling by relaxation operation. IEEE Trans. Syst. Man Cybern. **6**(6), 420–433 (1976)
80. Rusinkiewicz, S., Levoy, M.: Efficient variant of the ICP algorithm. In: Proc. of 3DIM, pp. 145–152 (2001)
81. Scharstein, D., Szeliski, R.: Stereo matching with nonlinear diffusion. Int. J. Comput. Vis. **28**(2), 155–174 (1998)
82. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. Int. J. Comput. Vis. **37**(2), 151–172 (2000)
83. Shi, J., Tomasi, C.: Good feature to track. In: Proc. IEEE CVPR, pp. 593–600 (1994)
84. Sidibe, D., Montesinos, Ph., Janaqi, S.: Fast and robust image matching using contextual information and relaxation. In: Proc. of Int. Conference on Computer Vision Theory and Applications, pp. 68–75 (2007)
85. Smith, S.M., Brady, J.M.: SUSAN: a new approach to low level image processing. Int. J. Comput. Vis. **23**(1), 45–78 (1997)
86. Sojka, E.: A new approach to detecting the corners in digital images. In: IEEE ICIP, vol. 3, pp. 445–448 (2003)
87. Suveg, I., Vosselman, G.: Mutual information based evaluation of 3D building models. In: ICPR, vol. III, pp. 557–560 (2002)
88. Tanizaki, H.: Non-Gaussian state-space modeling of nonstationary time series. J. Am. Stat. Assoc. **82**, 1032–1063 (1987)
89. Thevenaz, P., Unser, M.: Optimization of mutual information for multiresolution image registration. IEEE Trans. Image Process. **9**(12) (2000)
90. Tissainayagam, P., Suter, D.: Assessing the performance of corner detectors for point feature tracking applications. Image Vis. Comput., 663–679 (2004)
91. Trujillo, L., Olague, G.: Automated design of image operators that detect interest points. Evol. Comput. **16**(4), 483–507 (2008)
92. Trujillo, L., Olague, G., de Vega, F.F., Lutton, E.: Evolutionary feature selection for probabilistic object recognition, novel object detection and object saliency estimation using gmms. In: Proc. 18th BMVC, vol. 2, pp. 630–639 (2007)
93. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: a survey. Found. Trends Comput. Graph. Vis. **3**(3), 177–280 (2008)
94. Tzimiropoulos, G., Argyriou, V., Stathaki, T.: Subpixel registration with gradient correlation. IEEE Trans. Image Process. **20**(6), 1761–1767 (2011)
95. Vincent, E.: On feature point matching, in the calibrated and uncalibrated contexts, between widely and narrowly separated images. PhD thesis, Ottawa Carleton Institute for Computer Science (2004)
96. Weik, S.: Registration of 3-D partial surface models using luminance and depth information. In: Proc. of 3DIM (1997)
97. Weng, J., Cohen, P., Herniou, M.: Camera calibration with distortion models and accuracy evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **14**(10), 965–980 (1992)
98. Wu, H., Kitagawa, Y., Wada, T., Kato, T., Chen, Q.: Tracking iris contour with a 3D eye-model for gaze estimation. In: Proc. of ACCV, pp. 688–697 (2007)
99. Yang, G., Stewart, C.V., Sofka, M., Tsai, L.C.: Registration of challenging image pairs: initialization, estimation and decision. IEEE Trans. Pattern Anal. Mach. Intell. **29**(11), 1973–1989 (2007)

100. Zhang, Z.: Le probleme de la mise en correspondance: l'etat de l'art. INRIA, RR N' 2146 (1993)
101. Zhang, Z.: A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1330–1334 (2000)
102. Zhu, S., Ma, K.: A new diamond search algorithm for fast block-matching motion estimation. IEEE Trans. Image Process. **9**, 287–290 (2000)
103. Zitova, B., Flusser, J.: Image registration methods: a survey. Image Vis. Comput. **21**, 977–1000 (2003)
104. Zitova, B., Kautsky, J., Peters, G., Flusser, J.: Robust detection of significant points in multi-frame images. Pattern Recognit. Lett. **20**, 199–206 (1999)

Index

A. Chatterjee et al. (eds.), *Advances in Heuristic Signal Processing and Applications*,
DOI 10.1007/978-3-642-37880-5,