# ADAPTIVE SIGNAL PROCESSING ALGORITHMS FOR NONCIRCULAR COMPLEX DATA

*by*
SOROUSH JAVIDI

A thesis submitted in fulfilment of requirements for the degree of
Doctor of Philosophy of Imperial College London

Communications and Signal Processing Group
Department of Electrical and Electronic Engineering
Imperial College London
2010

# Abstract

The complex domain provides a natural processing framework for a large class of signals encountered in communications, radar, biomedical engineering and renewable energy. Statistical signal processing in $\mathbb{C}$ has traditionally been viewed as a straightforward extension of the corresponding algorithms in the real domain $\mathbb{R}$, however, recent developments in augmented complex statistics show that, in general, this leads to under-modelling. This direct treatment of complex-valued signals has led to advances in so called widely linear modelling and the introduction of a generalised framework for the differentiability of both analytic and non-analytic complex and quaternion functions. In this thesis, supervised and blind complex adaptive algorithms capable of processing the generality of complex and quaternion signals (both circular and noncircular) in both noise-free and noisy environments are developed; their usefulness in real-world applications is demonstrated through case studies.

The focus of this thesis is on the use of augmented statistics and widely linear modelling. The standard complex least mean square (CLMS) algorithm is extended to perform optimally for the generality of complex-valued signals, and is shown to outperform the CLMS algorithm. Next, extraction of latent complex-valued signals from large mixtures is addressed. This is achieved by developing several classes of complex blind source extraction algorithms based on fundamental signal properties such as smoothness, predictability and degree of Gaussianity, with the analysis of the existence and uniqueness of the solutions also provided. These algorithms are shown to facilitate real-time applications, such as those in brain computer interfacing (BCI). Due to their modified cost functions and the widely linear mixing model, this class of algorithms perform well in both noise-free and noisy environments. Next, based on a widely linear quaternion model, the FastICA algorithm is extended to the quaternion domain to provide separation of the generality of quaternion signals. The enhanced performances of the widely linear algorithms are illustrated in renewable energy and biomedical applications, in particular, for the prediction of wind profiles and extraction of artifacts from EEG recordings.

# Contents

# List of Figures

Figure Page

# List of Tables

# Acknowledgements

Firstly, I would like to sincerely thank my supervisor Dr. Danilo Mandic for his expert guidance through my PhD, I feel most fortunate to have worked with him. He introduced me to the wonderful world of higher dimensional signal processing, and his enthusiasm for the field has been a constant motivation for me. Despite his busy schedule, Dr. Mandic has always had time to monitor the progress in my research and provide valuable feedback. Through group research sessions, or social gatherings on a Friday evening, Dr. Mandic created a warm research environment for his students, which I greatly enjoyed and found valuable. I would also like to thank Dr. Mandic for his patience as I learnt the ropes early in my research, entrusting me with the design of the cover for his book, as well as introducing *mahi-mahi* and *cherkiz datasets* into my vocabulary.

I would like to show my appreciation to Prof. Kin Leung, Head of the Communications and Signal Processing research group at Imperial College, for providing me with the opportunity to design and develop the website for the University Defence Research Centre (UDRC). It has been both exciting and a privilege to be involved in some capacity with the Centre.

This work wouldn't have been possible without my friends and colleagues. Dr. Clive Cheong Took has always been there to help and to generously provide his time for my questions. David Looney and Cheolsoo Park have been both good friends and skilful colleagues in EEG data acquisition and analysis, and have been very helpful in discussions on the experimental parts of this research. I have also enjoyed the company of Beth Jelfs, Ling Li, Yili Xia, Naveed ur Rehman, Che Ahmad Bukhari and Cyrus Jahanchahi. I would like to extend my thanks to all my other colleagues from the Communications and Signal Processing research group at Imperial College, and in particular Ario Emaminejad for the discussions and debates on anything and everything outside of research. I am also grateful to Jing Liu, for being there throughout the highs and lows of my research.

Last but not least, my deepest gratitude to my parents for their constant love and support, not only during the past four years, but also throughout my education. They have always encouraged me to progress and excel in whatever I do, and have always been there for me. It has been a joy and source of great comfort to have the support of my brother Saeed during my PhD research, and I am ever thankful to him for his patience and understanding of my sometimes unsociable work routine.

Soroush Javidi
July 2010

# Statement of Originality

As far as I am aware, this work contains original contributions to the field of complex- and quaternion-valued adaptive signal processing, with any work and ideas pertaining to other people acknowledged and referenced accordingly. This is supported by publications, listed in the next section. The original contributions arising from this work are summarised as follows:

- A Widely linear Complex Least Mean Square (CLMS) algorithm, [C7].
- A class of prediction based noncircular complex blind source extraction algorithms, [J1].
- A class of kurtosis based noncircular complex blind source extraction algorithms, [J4].
- A fast converging algorithm for the extraction of smooth noncircular complex-valued sources, [C2].
- A Fast Independent Component Analysis (FastICA) algorithm for noncircular quaternion-valued signals, [J5].
- Establishing the Taylor Series Expansion (TSE) of real-valued functions of complex-values matrices in the $\mathbb{CR}$ calculus framework, for the analysis of algorithms with complex-valued matrix input, [C3].
- Analysis and comparison of the performance and computational complexity of real- and complex-valued block Least Mean Square (LMS) algorithms, [C2].
- Convergence analysis of the generalised complex FastICA algorithm (nc-FastICA).
- An online quaternion blind source extraction algorithm using the temporal structure of proper and improper quaternion signals, [C1].

# Publications

The following are contributions resulting from this work.

**Book article**

[B1] B. Jelfs, P. Vayanos, S. Javidi, S. L. Goh and D. P. Mandic. Collaborative Adaptive Filters for Online Knowledge Extraction and Information Fusion, in *Signal Processing Techniques for Knowledge Extraction and Information Fusion*, D. P. Mandic, M. Golz, A. Kuh, D. Obradovic and T. Tanaka, Eds., pp. 3–21, Springer, 2008.

**Journal articles**

[J1] S. Javidi, D. P. Mandic and A. Cichocki. Complex Blind Source Extraction from Noisy Mixtures using Second Order Statistics, *IEEE Transactions on Circuits and Systems I: Regular Papers*, 57(7):1404–1416, 2010.

[J2] B. Jelfs, S. Javidi, P. Vayanos and D. P. Mandic. Characterisation of Signal Modality: Exploiting Signal Nonlinearity in Machine Learning and Signal Processing, *Journal of Signal Processing Systems*, Springer, 61(1):105–115, 2010.

[J3] D. P. Mandic, S. Javidi, S. L. Goh, A. Kuh and K. Aihara. Complex-valued Prediction of Wind Profile Using Augmented Complex Statistics, *Renewable Energy*, 34:196–201, 2007.

[J4] S. Javidi, D. P. Mandic and A. Cichocki. Kurtosis Based Blind Source Extraction of Complex Noncircular Signals with Application in EEG Artifact Removal in Real-Time, submitted to *Neural Networks*.

[J5] S. Javidi, D. P. Mandic. A Fast Independent Component Analysis Algorithm for Improper Quaternion Signals, submitted to *IEEE Transactions on Neural Networks*, Revised August 2010.

**Conference proceedings**

[C1] S. Javidi, C. Cheong Took, C. Jahanchahi, N. Le Bihan and D. P. Mandic. Blind Extraction of Improper Quaternion Sources, submitted to *Proc. IEEE International Conference on Acoustic Speech and Signal Processing*, 2011.

[C2] S. Javidi and D. P. Mandic. A Fast Algorithm for Blind Extraction of Smooth Complex Sources with Application in EEG Conditioning, in *Proc. IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pp. 397–402, 2010.

[C3] S. Javidi, D. P. Mandic and A. Kuh. Optimisation of Real Functions of Complex Matrices for the Adaptive Estimation of Complex Sources, in *Proc. International Conference on Green Circuits and Systems*, pp. 30–35, 2010.

[C4] Y. Xia, S. Javidi and D. P. Mandic. A Regularised Normalised Augmented Complex Least Mean Square Algorithm, in *Proc. International Symposium on Wireless Communications Systems*, pp. 355–358, 2010.

[C5] S. Javidi, B. Jelfs and D. P. Mandic. Blind Extraction of Noncircular Complex Signals Using a Widely Linear Predictor, in *Proc. IEEE Workshop on Statistical Signal Processing*, pp. 501–504, 2009.

[C6] Y. Xia, C. Cheong Took, S. Javidi and D. P. Mandic. A Widely Linear Affine Projection Algorithm, *Proc. IEEE Workshop on Statistical Signal Processing*, pp. 373–376, 2009.

[C7] S. Javidi, M. Pedzisz, S. L. Goh and D. P. Mandic. The Augmented Complex Least Mean Square Algorithm, *Proc. of the 1st IARP Workshop on Cognitive Information Processing*, pp. 54–57, 2008.

[C8] D. P. Mandic, P. Vayanos, S. Javidi, B. Jelfs and K. Aihara. Online Tracking of the Degree of Nonlinearity Within Complex Signals, in *Proc. IEEE International Conference on Acoustic Speech and Signal Processing*, pp. 2061–2064, 2008.

[C9] D. P. Mandic, S. Javidi, G. Souretis and S. L. Goh. Why a Complex Valued Solution for a Real Domain Problem, in *Proc. IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pp. 384–389, 2007.

# List of Abbreviations

| | |
|---|---|
| **ACLMS** | Augmented Complex Least Mean Square |
| **AR** | Autoregressive |
| **CLMS** | Complex Least Mean Square |
| **BCI** | Brain Computer Interface |
| **BSE** | Blind Source Extraction |
| **BSS** | Blind Source Separation |
| **b-ACLMS** | Block Augmented Complex Least Mean Square |
| **b-CLMS** | Block Complex Least Mean Square |
| **b-DCRLMS** | Block Dual Channel Real Least Mean Square |
| **BPSK** | Binary Phase Shift Key |
| **CCA** | Canonical Correlation Analysis |
| **c-FastICA** | complex Fast Independent Component Analysis |
| **c-GGD** | Complex Generalised Gaussian Distribution |
| **DCRLMS** | Dual Channel Real Least Mean Square |
| **EEG** | Electroencephalography |
| **EMD** | Empirical Mode Decomposition |
| **EMG** | Electromyography |
| **EOG** | Electrooculography |
| **EVD** | Eigenvalue Decomposition |
| **FastICA** | Fast Independent Component Analysis |
| **FFT** | Fast Fourier Transform |
| **FIR** | Finite Impulse Response |
| **GGD** | Generalised Gaussian Distribution |
| **GNGD** | Generalised Normalised Gradient Descent |
| **H-H** | Hilbert-Huang |
| **ICA** | Independent Component Analysis |
| **JADE** | Joint Approximate Diagonalisation of Eigenmatrices |
| **K-cBSE** | Kurtosis based Blind Source Extraction |
| **LMS** | Least Mean Square |
| **MEMD** | Multivariate Empirical Mode Decomposition |
| **MSE** | Mean Square Error |
| **MSPE** | Mean Square Prediction Error |
| **nc-FastICA** | noncircular/generalised complex Fast Independent Component Analysis |
| **P-cBSE** | Prediction based Complex Blind Source Extraction |
| **PI** | Performance Index |

| | |
|---|---|
| **pdf** | Probability Density Function |
| **pPSD** | Pseudo Power Spectral Density |
| **PSD** | Power Spectral Density |
| **QAM** | Quadrature Amplitude Modulation |
| **q-FastICA** | Quaternion Fast Independent Component Analysis |
| **QLMS** | Quaternion Least Mean Square |
| **QPSK** | Quadrature Phase Shift Keying |
| **S-cBSE** | Smoothness based Complex Blind Source Extraction |
| **SNR** | Signal to Noise Ratio |
| **SOBI** | Second-Order Blind Identification |
| **SUT** | Strong Uncorrelating Transform |
| **T-F** | Time-Frequency |
| **TSE** | Taylor Series Expansion |
| **VSS** | Variable Step Size |
| **WL** | Widely Linear |

# Mathematical Notations

| | |
|---|---|
| $\otimes$ | Kronecker product |
| $\lvert\cdot\rvert$ | Modulus operator |
| $\lVert\cdot\rVert$ | Vector or matrix norm |
| $\lVert\cdot\rVert_2$ | The Euclidean norm |
| $\lVert\cdot\rVert_F$ | The Frobenius norm |
| $\lVert\cdot\rVert_{W^{p,q}}$ | The Sobolev norm |
| $(\cdot)^*$ | Complex conjugate operator |
| $(\cdot)^{-1}$ | Matrix inverse operator |
| $(\cdot)^{\#}$ | Matrix pseudo-inverse operator |
| $(\cdot)^T$ | Vector or matrix transpose operator |
| $(\cdot)^H$ | Conjugate Transpose (Hermitian) operator |
| $\triangleq$ | Defined as |
| $\nabla$ | Gradient operator |
| $\partial$ | Partial derivative operator |
| $\mathbf{0}$ | Vector or matrix with all zero elements |
| $\mathbf{A}$ | Mixing matrix |
| $\mathrm{cum}(\cdot)$ | Cumulant |
| $\mathbb{C}$ | Field of complex numbers |
| $\mathcal{C}_{\mathbf{zz}}$ | Covariance matrix of random vector $\mathbf{z}$ |
| $\mathcal{C}^a_{\mathbf{zz}}$ | Augmented covariance matrix of random vector $\mathbf{z}$ |
| $\mathcal{C}^R_{\mathbf{zz}}$ | Bivariate covariance matrix |
| $\det(\cdot)$ | Matrix determinant operator |
| $\mathrm{diag}(\cdot)$ | Diagonal matrix of elements |
| $E\{\cdot\}$ | Expectation operator |
| $E\{y\|x\}$ | Conditional expectation of $y$ given $x$ |
| $\mathcal{F}(\cdot)$ | Fourier transform operator |
| $\mathbf{g}$ | Filter coefficient vector |
| $\mathbf{h}$ | Filter coefficient vector |
| $\mathbb{H}$ | Field of quaternion numbers |
| $\mathbf{H}$ | Hessian matrix |
| $\mathbf{H}^a$ | Augmented Hessian matrix |
| $\imath$ | $\sqrt{-1}$ |
| $\mathbf{I}$ | Identity matrix |
| $\Im\{\cdot\}$ | Imaginary part of a complex number |
| $\jmath$ | $\sqrt{-1}$ |
| $\mathbf{J}_N$ | Real to Complex mapping matrix of size $2N \times 2N$ |

| | |
|---|---|
| $\mathbf{J}_{\mathbf{F}}$ | Jacobian matrix of vector function $\mathbf{F}$ |
| $\mathbf{J}^{c}_{\mathbf{F}}$ | Conjugate Jacobian matrix of vector function $\mathbf{F}$ |
| $\mathcal{J}(\cdot)$ | Cost function |
| $k$ | Discrete time index |
| $\kappa$ | $\sqrt{-1}$ |
| $K_c(\cdot)$ | Normalised kurtosis of a complex-valued random variable |
| $K_R(\cdot)$ | Normalised kurtosis of a real-valued random variable |
| $\mathrm{kurt}_c(\cdot)$ | Kurtosis of a complex-valued random variable |
| $\mathrm{kurt}_R(\cdot)$ | Kurtosis of a real-valued random variable |
| $\mathcal{L}(\cdot, \lambda)$ | Lagrangian function, with Lagrange parameter $\lambda$ |
| $\mathcal{O}(\cdot)$ | Order of computational complexity |
| $p_Z(\mathbf{z})$ | Probability density of a random vector $\mathbf{z}$ |
| $\mathcal{P}_{\mathbf{zz}}$ | Pseudo-covariance matrix of a random vector $\mathbf{z}$ |
| $q$ | Quaternion random variable |
| $\mathrm{q}$ | Quaternion number |
| $\mathbf{q}$ | Quaternion random vector |
| $\mathbf{q}_a$ | vector of real components of $\mathbf{q}$ |
| $\mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d$ | vector of imaginary components of $\mathbf{q}$ |
| $\mathbf{q}^a$ | Augmented quaternion vector |
| $\mathrm{q}^{\imath}, \mathrm{q}^{\jmath}, \mathrm{q}^{\kappa}$ | Involution about the $\imath$, $\jmath$ or $\kappa$ axis |
| $r$ | Degree of noncircularity |
| $\mathbb{R}$ | Field of real numbers |
| $\Re\{\cdot\}$ | Real part of a complex number |
| $s_i(k)$ | $i$th source signal at a discrete time $k$ |
| $\mathbf{s}(k)$ | Source vector at a discrete time $k$ |
| $\mathcal{S}_{\mathbf{z}}$ | Fourier transform of covariance matrix, Spectral matrix |
| $\mathcal{S}^a_{\mathbf{z}}$ | Augmented spectral matrix |
| $\breve{\mathcal{S}}_{\mathbf{z}}$ | Fourier transform of pseudo-covariance matrix, Pseudo-spectral matrix |
| $\mathrm{sgn}(\cdot)$ | Sign function |
| $\mathrm{sinv}(\cdot)$ | Self-inverse mapping operator |
| $\mathrm{Tr}(\cdot)$ | Matrix trace operator |
| $\mathbf{u}^{\star}$ | Vector of fixed-points |
| $\mathbf{v}(k)$ | Noise vector at discrete time $k$ |
| $\mathrm{vec}(\cdot)$ | The vectorise operator |
| $\mathbf{w}$ | Demixing vector |
| $\mathbf{W}$ | Demixing matrix |
| $\mathbf{x}(k)$ | Input vector at a discrete time $k$, observed mixture at a discrete time $k$ |
| $y_i(k)$ | $i$th output at a discrete time $k$, $i$th estimated source at a discrete time $k$ |
| $\mathbf{y}(k)$ | Vector of estimated sources at a discrete time $k$ |

| | |
|---|---|
| $z$ | Complex random variable |
| $\mathbf{z}$ | Complex random vector |
| $\mathbf{z}^a$ | Augmented complex random vector |
| $\mathbf{z}_r, \mathbf{z}_i$ | Vector of real/imaginary parts of $\mathbf{z}$ |
| $\mathbf{z}^R$ | Composite real vector $[\mathbf{z}_r^T, \mathbf{z}_i^T]^T$ |
| $\mathbf{Z}$ | Complex matrix |
| $\mathbf{Z}^a$ | Augmented complex matrix |
| $\mathbf{Z}^R$ | Composite complex matrix |
| $\delta$ | Discrete time delay |
| $\delta_0$ | Delta function |
| $\lambda$ | Mixing parameter of a hybrid filter |
| $\lambda^a$ | Eigenvalue of an augmented matrix |
| $\lambda^R$ | Eigenvalue of a composite matrix |
| $\rho(z)$ | Circularity quotient of random variable $z$ |
| $\rho_s(z)$ | Smoothness measure of $z$ |
| $\sigma_z^2$ | Variance of a random variable $z$ |
| $\tau_z^2$ | Pseudo-variance of a random variable $z$ |

# Chapter 1

# Introduction

## 1.1 Signal processing in $\mathbb{R}$

Adaptive signal processing has been at the centre of statistical signal processing research for the past five decades, and has found a wide range of applications, including channel equalisation in communications, beamforming, biomedical applications such as functional magnetic resonance imaging (fMRI) and electroencephalography (EEG) and radar [1]. While digital filters with fixed coefficients are only optimal for static scenarios, adaptive filters require no assumptions on the signal generating mechanism, and operate in nonstationary environments [2]. In addition, the increase in the processing power along with lower cost and lower power consumption requirements of digital processors, have allowed for the investigation of more ambitious and computationally complex problems.

Adaptive signal processing algorithms can be divided into two distinctive categories: supervised and unsupervised (blind). The presence of training signals in supervised algorithms results in more straightforward methods for adaptive filtering, whereby the operation is governed by the training signal. Blind algorithms process the output without the knowledge of the system, teaching inputs, or both. Such a scenario results in a more challenging problem where it is required to make certain prior assumptions on the input signal or system. The design of both supervised and blind signal processing algorithms relies on the choice of a suitable statistical signal model (architecture) as a prerequisite prior to the development of mathematical optimisation methods (algorithms).

Supervised adaptive algorithms based on the Wiener and Kalman filters have been extensively studied in the real domain $\mathbb{R}$. The Least Mean Square (LMS) algorithm, introduced in the 1960s by Widrow and Hoff, is the most well-known and used in practice supervised adaptive algorithm in $\mathbb{R}$, and much research has been dedicated

to the analysis of LMS and enhancement of its performance. This includes the class of variable step-size LMS algorithms proposed by Benveniste, Mathews, and Ang and Farhang-Boroujeny, which aim to adapt the LMS step-size in a 'linear' fashion to make it suitable for time varying and nonstationary conditions [3]. The Generalised Normalised Gradient Descent (GNGD) algorithm [4, 5, 6] adapts the learning rate in a 'nonlinear' manner; it is based on normalised LMS (NLMS) and avoids spurious solutions due to small signal magnitudes by adapting the regularisation parameter. While both strategies equip LMS with an adaptive step-size, the GNGD algorithm is more powerful, due to its nonlinear step-size update and also provides improved stability [4].

A hybrid filter, based on a combination of two adaptive sub-filters, was addressed in [7], whereby by virtue of the convex mixing parameter this collaborative structure provides enhanced performance. By selecting sub-filters whose natures complement one another, such a hybrid filter then outperforms the individual adaptive sub-filters. This results in, for example, fast convergence and always stable steady-state performance. Hybrid filters have also been utilised for collaborative adaptive filtering scenarios, and the online tracking of signal modality [8, 9], the so obtained information on the signal modality can then be used as prior knowledge to further processing units.

Blind signal processing algorithms have gained much attention in the past two decades, resulting in a wide range of algorithms with application in biomedical and communications fields [10, 11]. In their most fundamental form, the aim is to estimate unknown source signals from an array of observed signals, without knowledge of the mixing system or signal generation. Alternatively, under the umbrella of blind source separation (BSS), where possible such algorithms employ physically meaningful assumptions on the system and latent signals, in order to enhance performance.

A typical assumption on the mixing system is that the output signals are linear mixtures of the unknown input sources, with further assumption of the statistical independence of the latent sources, leading to Independent Component Analysis (ICA) [12]. While this assumption may not be realistic for real-world scenarios, for example, for correlated source signals due to reverberation, it is applicable to certain scenarios where some prior knowledge about the sources is available, such as in the estimation of biomedical signals originating from different organs, such as the mixture of electrical activity from brain functions (EEG) and electrocardiogram (ECG) signals from the heart. Likewise, it is common for the observed signals to be a mixture of two physical entities, such as mother and fetal ECG [13].

An insight into the unknown sources or mixing system allows for modelling more complex scenarios encountered in real-world problems. Blind source separation of signals with post-nonlinear mixing was addressed in [14] and further generalised in [15], for the separation of latent sources from a post-nonlinear mixture with an

ill-conditioned mixing matrix. Likewise, blind source separation in noisy environments has been studied in [16]. Two noise models were discussed in [12], where noise was considered either additive for each observed mixture (output) and termed sensor noise, or it was additive for the source signals prior to being mixed by the system, called source noise. The case of additive sensor noise was considered in [17], modelled as an additive white Gaussian noise, and removed through a bias removal method.

Another assumption in BSS is that of underdetermined mixtures; in the standard model the number of observed mixtures is considered equal to (or greater than) the source signals, while in a practical situation, the exact number of sources may be unknown or change in time. In the case of an underdetermined mixture, the number of sources is greater than the observed mixtures, which results in a mixing matrix which is not linearly invertible. To overcome this problem, various algebraic techniques and assumptions have been introduced. This includes the use of canonical decomposition [18], parallel factor analysis (PARAFAC) [19] and prior assumptions on the source characteristic function [20].

One of the concepts employed by real-valued BSS methodologies is to exploit the degree of Gaussianity of the source signals as a signal fingerprint. This is justified by the central limit theorem, where the observed mixture of signals has a more Gaussian distribution that the original source signals. Thus, based on the discussed assumptions, it is possible to estimate a set of sources that are independent, while being maximally non-Gausssian. This can be achieved using a higher order statistic, typically based on kurtosis as a measure of non-Gaussianity, and by maximising (or minimising) the kurtosis of the estimated sources.

As the kurtosis is sensitive to outliers, an information theoretic approach based on the utilisation of the negentropy function is a more general and robust approach to the use of kurtosis [12]. The negentropy function is a normalised variant of an entropy measure, such that it is zero for a Gaussian random variable and non-zero for random variables with non-Gaussian distributions. As knowledge of the negentropy function is generally not available, it is estimated using suitably chosen nonlinearities. This principle is utilised in the FastICA algorithm, which maximises the negentropy of the estimated sources using a fast converging fixed-point like Newton method [21, 22].

The simple offline Fourth Order Blind Identification (FOBI) algorithm [23] estimates sources by obtaining the inverse of the mixing matrix, called demixing matrix, using the eigenvalue decomposition (EVD) of a weighted covariance matrix. As the eigenvalues of the weighted covariance matrix are formed by the fourth order moments of the source signals, the performance of the algorithm is limited to only separating sources with distinct kurtosis values. The tensorial approach of the offline Joint Approximate Diagonalisation of Eigenmatrices (JADE) [24] method is a generalisation of FOBI, which utilises the EVD of the fourth order cumulant tensor. Due to the com-

plexity associated with the calculation of the EVD, the algorithm is only suitable for problems with small number of sources.

The class of algorithms for the estimation of sources using maximum likelihood (ML) rely on the estimation of the source probability density function (pdf). It is possible to utilise density estimates by using a pair of nonlinearities that encompass densities of both sub- and super-Gaussian random variables, however, the drawback is that the correct estimator has to be used in the algorithm. The ML based algorithm was introduced in [25] and the gradient adaptive ML based algorithm of Bell and Sejnowski based on the infomax principle in [26]. A modified algorithm was addressed in [27], a derivation based on the natural gradient (relative gradient) was discussed in [28, 29] and a fixed-point like (FastICA) variant of the algorithm is given in [12]. The natural gradient variant of the algorithm avoids matrix inversion calculations at each iteration of the gradient update, while the FastICA variant allows for faster convergence and the use of a fixed density estimator. Generalisation of the ML approach and maximisation of negentropy is based on the minimisation of the mutual information, or statistical dependence, of the estimated sources. Thus, the previously mentioned methods also operate on the basis of minimising the mutual information. For instance, the algorithm in [30] introduces a natural gradient based algorithm that minimises the Kullback-Liebler divergence, which is equivalent to minimising the mutual information.

The task of separating latent sources may be performed simultaneously in parallel, or, one-by-one in a deflationary manner. The option of choosing either method is dependent on the problem and the choice of algorithm. For example, while algorithms based on the maximisation of negentropy allow for both simultaneous and deflationary separation of sources, those based on the ML approach or direct linear algebraic manipulation only allow for simultaneous separation of sources [12]. This may not be desirable in problems with high dimensionality, or when only a few of the sources are required. Procedures pertaining to the estimation of a subset of sources are termed blind source extraction (BSE) algorithms. While source extraction using standard algorithms such as FastICA can be performed in a deflationary manner, it may be desirable to extract sources based on a certain fundamental signal property. This leads to lower computational complexity and the possibility to remove the need for pre- or post-processing.

Algorithms for blind source extraction of real-valued sources utilise both second- and higher-order statistical properties of signals to discriminate between the sources. Algorithms based on higher order statistics achieve this by minimising cost functions based typically on the skewness [31] and kurtosis (and generalised kurtosis) [10, 32, 33, 34]. Alternatively, the predictability of the sources (arising from their temporal structure) leads to another class of algorithms which minimise cost functions based

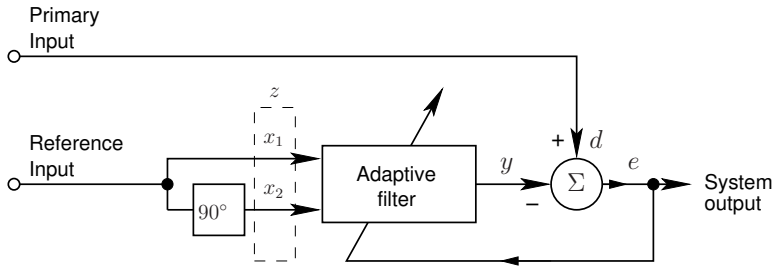



**Figure 1.1** Adaptive algorithm in interference cancelling mode, acting as an adaptive notch filter

on the mean square prediction error (MSPE) [10, 35, 36].

## 1.2 Signal processing in $\mathbb{C}$

Signals encountered in the complex domain $\mathbb{C}$ can be divided into two groups: those complex by design, and those made complex by convenience of representation. For instance, signals encountered in the communications field (e.g. QPSK) and signals obtained from an fMRI procedure are considered complex by design (complex by nature), while a complex wind signal is represented by convenience of representation by combining its speed and direction into a complex vector. Also, as a preliminary stage in beamforming problems, a phasor is created using a phase-quadrature demodulator, which is also complex [37]. Finally, consider the methodology presented in [38] for the removal of power line noise in ECG type applications. For enhanced performance, the input of an adaptive filter in the noise cancellation configuration (see Figure 1.1) is first phase shifted by $\pi/2$ radians and then coupled with its original version to effectively form a complex signal.

A complex signal can be represented by its real and imaginary, or phase and amplitude components. The adaptive processing of complex-valued signals can then be performed using three different approaches. Firstly, the real and imaginary components (or phase and amplitude) are considered as dual univariate signals and processed separately. Secondly, the two components can be considered as a real-valued bivariate signal and processed using a suitable real-valued two-dimensional algorithm. Alternatively, it would be natural to consider the signal directly in the complex domain $\mathbb{C}$ and process it by utilising algorithms designed directly for complex-valued signals.

An early example of such an approach is the extension of the LMS algorithm to the complex domain (CLMS) by Widrow et al. in 1975 [39]. More recently, the Least Mean Phase-Least Mean Square (LMP-LMS) algorithm [40] was introduced for the simultaneous processing of both the signal magnitude and phase. This is especially useful for scenarios occurring in communications where the phase and not the magnitude

of the signal is the information carrier. The LMP-LMS algorithm minimises the mean square error in both the signal magnitude and phase, however, it suffers from reduced performance for signals with small magnitudes.

In the field of unsupervised signal processing, Bingham and Hyvärinen extended the FastICA algorithm to the complex domain (c-FastICA) in [41]. Likewise, Anemüller et al. proposed the use of a complex ICA algorithm based on the ML approach [42]. The algorithm operates in the frequency domain and was designed for the processing of EEG signals. In their work, the authors consider the recorded EEG signal as a spatio-temporal mixture and propose the creation of complex EEG signals using the Fast Fourier Transform (FFT). However, the use of the FFT results in the processing of signals in piecewise stationary blocks, and online processing of signals may not be possible. In addition, the FFT acts as a smoothing filter and in effect flattens the true spectrum.

Traditionally, complex algorithms were considered as simple extensions of the established corresponding algorithms in the real domain. In particular, statistical modelling of complex-valued random vectors was taken as straightforward extensions from $\mathbb{R}$. For example, the covariance $E\{\mathbf{x}\mathbf{x}^T\}$ in $\mathbb{R}$ would be transformed to $E\{\mathbf{z}\mathbf{z}^H\}$ in $\mathbb{C}$, where only the change from the transposition operator $(\cdot)^T$ to the conjugate transpose (Hermitian) operator $(\cdot)^H$ was considered necessary. In this manner, the distribution of a complex-valued random vector is, either implicitly or explicitly, symmetric (or circular) within the complex domain. This assumption implies the independence of the real and imaginary signal components, which is not correct for the generality of complex-valued signals. Thus, the aforementioned complex-valued algorithms are only optimal for a subset of complex signals, those with a circularly symmetric distribution.

A generalised statistical framework in $\mathbb{C}$ for signal processing applications was introduced in the 1990s. Fundamental work by Picinbono, Neeser and Massey addressed the concept of complex circularity, second-order statistics of complex random variables and widely linear modelling of complex signals. A complex-valued random variable is considered circular if it has a rotation invariant distribution, and is otherwise known as noncircular [43]; this concept forming the building block for the consideration of complex statistics. The second-order statistics of complex-valued random vectors was addressed in [44] and [45], where it was shown that the covariance matrix does not sufficiently model the statistics and it is necessary to introduce the pseudo-covariance matrix to fully capture the relation between the real and imaginary components of random vectors. Thus both the covariance and pseudo-covariance matrices are required in order to model the complete second-order information available within the signal. In the case of second-order circular (also called proper) complex random variables, the pseudo-covariance matrix vanishes, which coincides with the assump-

tion of traditional algorithms in $\mathbb{C}$. However, for the case of second-order noncircular (or improper) random variables, the pseudo-covariance matrix is non-zero.

Based on this understanding, the widely linear model of complex-valued signals is introduced in [46] which incorporates information in both the covariance and pseudo-covariance matrices. It is shown that the standard linear model is only sufficient for modelling 'proper' signals, whereas an optimal model for 'improper' signals is provided by a widely linear model. Brief discussion on the extension of these methods to higher order statistics in given in [47]. The fundamental results of complex statistics were recently revisited by Schreier and Scharf in [48] and [49], in particular, the notion of augmented complex statistics is mentioned in [48], reflecting the construction of matrices ccomprising both the covariance and pseudo-covariance matrices. This is not performed ad hoc, and is a result of the isomorphism (duality) between the real and complex domains, which was first discussed by van den Bos [50, 51]. Treatment of statistics in $\mathbb{C}$ from the point of view of augmented random vectors also allows for insight into the duality between the second-order statistics in $\mathbb{C}$ and its counterpart in $\mathbb{R}^2$ for bivariate real-valued signals.

The duality between the real and complex domain was first exploited by van den Bos in [51] to provide a generalisation of the complex Gaussian distribution. Based on the traditional treatment of complex statistics, the complex Gaussian distribution was explicitly described for circular signals [52]. Thus, the generalised complex Gaussian distribution is suitable for modelling both circular and noncircular Gaussian probability distributions and the traditional complex Gaussian distribution is shown to be a special case.

The duality of the two domains is also exploited by the same author in [50], where he addresses the Taylor Series Expansion (TSE) of complex functions. The importance of his work is twofold. First, it provides a generalised TSE of complex functions and introduces a generalised Newton optimisation method for complex functions. Second, by considering the mapping between a complex value and its bivariate form, van den Bos subtly introduces the concept of duality between the two domains as well as a methodology for establishing the duality for analysis of complex functions. Another fundamental result in the treatment of functions in $\mathbb{C}$ is given by Brandwood in [53], where the gradient of functions of a complex variable are shown to be in the direction of the conjugate of the variable.

This result, along with the concept of duality, was thoroughly investigated and unified within the $\mathbb{CR}$ calculus framework by Kreutz-Delgado in [54], and provides a comprehensive reference to the treatment of functions of complex-valued variables. The so called $\mathbb{CR}$ calculus framework (also known as Wirtinger calculus [55]) allows for the treatment of functions of complex variable directly in the complex domain. This is particularly important for typical real-valued cost functions encountered in signal pro-

cessing problems. As such functions are non-analytic, the standard Cauchy-Riemann results are not applicable and it is customary to perform derivations individually on the real and imaginary components of the function. However, the $\mathbb{CR}$ calculus allows for the consideration of both analytic and non-analytic functions in a unified manner, and greatly simplifies the differentiation and analysis of complex functions.

The advantages offered by augmented complex statistics are just being exploited in supervised learning. In particular, the extensions of real-valued recurrent neural network structures [56] to those in the complex domain based on widely linear models have been recently designed. This has led to the introduction of the augmented complex real-time recurrent learning (ACRTRL) algorithm [57] and the augmented complex-valued extended Kalman filter (ACEKF) for complex recurrent neural networks [58]. These algorithms were shown to outperform their standard complex counterparts for the generality of complex-valued signals. The performance of complex recurrent neural networks were compared for the task of wind profile prediction using a dual univariate model and a complex model in [59, 60], where the complex representation of wind resulted in better performance when predicted using a trained CRTRL algorithm. In comparison, the ACRTRL algorithm achieved a better prediction performance, highlighting the associated benefits of considering augmented complex statistics [57]. Finally, the widely linear affine projection algorithm (WL-APA) and widely linear IIR filters have been demonstrated to be suitable for processing the generality of real-world data [61, 62]. In brief, the difference between these algorithms and standard complex supervised algorithms lies in the complete second-order statistical modelling of signals due to the use of augmented complex statistics; a comprehensive discussion is given in [63].

It is important to note that in earlier research in nonlinear complex signal processing problems, split-complex nonlinear functions were considered [64, 65, 66]. A split-complex nonlinearity allows for a bounded and non-analytic function operating separately on the real and imaginary components of the input. However, these functions are not true complex functions and do not provide adequate modelling of complex nonlinearities. A split-complex function assumes that the real and imaginary components of the complex-valued input signal are independent thus preserving the phase. In [67, 68], the use of fully-complex nonlinearities was discussed. A fully-complex nonlinearity is bounded almost everywhere, is analytic and allows for the transformation of both the phase and amplitude of the input.

Recent research in blind signal processing has also resulted in the introduction and extension of standard BSS methodologies to the complex domain. In comparison to earlier work in complex BSS, designed for circular complex-valued sources, recent algorithms have generalised assumptions relating to latent sources and have thus created enhanced algorithms for blind separation in $\mathbb{C}$. In addition, the use of the $\mathbb{CR}$

calculus has allowed for the analysis and derivation of algorithms directly in the complex domain.

The identifiability and separability of complex sources was addressed by Eriksson and Koivunen in [69]. A particularly interesting result from their work shows that unlike the real domain, the number of complex Gaussian sources which can be resolved is not limited, however, the Gaussian sources should have distributions with unique degrees of circularity. The authors also introduce the strong uncorrelating transform (SUT), which allows for the simultaneous diagonalisation of the covariance and pseudo-covariance matrix. Based on the Takagi factorisation [70], this provides a valuable tool for both the analysis and design of complex BSS algorithms. The SUT method was utilised by Douglas to introduce a FastICA implementation based on kurtosis and the diagonalisation of both the covariance and pseudo-covariance matrices [71]. A generalised uncorrelating transform (GUT), based on generalised estimators of the covariance and pseudo-covariance matrices, was used to perform ICA on latent complex-valued sources using direct matrix calculation [72]. A generalisation of the FOBI algorithm using generalised covariance matrix estimators was also proposed by Ollila et al. in [73].

In [74], Novey and Adalı extend blind separation using negentropy maximisation to the complex domain. In their work they use fully-complex nonlinearities in the estimation of the negentropy function. Gradient adaptive and Newton based algorithms using the definition of complex kurtosis were introduced in [75]. An ML approach for complex ICA using the natural gradient was outlined in [76] and its stability analysis presented in [77]. In this work, fully-complex nonlinear functions were used for the approximation of the source density function. In addition, the use of $\mathbb{CR}$ calculus was emphasised, and was shown to simplify the task of derivation of the gradient descent algorithm and its use in the analysis of the second order TSE of the update algorithm. While the standard complex FastICA [41] assumes circular sources, the algorithm was generalised for the processing of both complex circular and noncircular source signals in [78] and termed the noncircular (or generalised) complex FastICA (nc-FastICA). Performance of these algorithms in the estimation and separation of the generality of complex-valued sources demonstrated enhanced performance in comparison to standard BSS algorithms.

## 1.3 Motivation and aims

The focus of this work is on the extension of supervised and blind signal processing algorithms to higher dimensional spaces, and in particular to the complex domain $\mathbb{C}$ and the quaternion domain $\mathbb{H}$. As augmented complex statistics is maturing and

the use of $\mathbb{CR}$ calculus is becoming a standard tool for the analysis of functions in $\mathbb{C}$, practical learning algorithms are only just being introduced for both supervised and blind signal processing of noncircular signals.

This thesis introduces several contributions to supervised and blind adaptive signal processing of noncircular signals:

- The standard complex LMS algorithm, introduced over 35 years ago, was modelled using a simplified statistical model. It is thus important to provide an enhancement of the algorithm and generalise it so as to cater for both complex circular and noncircular signals. As a workhorse of adaptive signal processing, it is anticipated that a generalised variant of the algorithm will become a de facto standard adaptive complex algorithm based on the Wiener filter.

- The topic of complex blind source extraction based on fundamental signal properties is addressed. While it is possible to extract signals based on a deflationary method using recently introduced BSS algorithms, with prior knowledge of the desired signals, algorithms can be designed that selectively extract source signals. Such targeted algorithms are suitable in real-world applications where certain knowledge of the desired sources is available. For example in EEG conditioning, certain information on the properties of the pure EEG and aritfact signals is available. Therefore, real-time removal of artifacts based on their fundamental properties can aid in tasks such as brain computer interfacing (BCI). In this thesis, a class of algorithms capable of extracting desired complex-valued sources based on the signal predictability, smoothness and degree of Gaussianity are introduced, providing the capability of online treatment directly in the complex domain. These signal properties are statistically modelled using augmented complex statistics, and the algorithms are derived directly in $\mathbb{C}$ using the $\mathbb{CR}$ calculus.

- It is important to design algorithms that are applicable in real-world, and this thesis provides solutions that demonstrate the applicability of the introduced algorithms to a variety of problems such as wind prediction and EEG conditioning. The focus has been on the usefulness of signals made complex by convenience of representations, and on the design of algorithms capable of processing complex-valued signals in real-time and directly in the time domain.

- Another aim of this work is to expand the analytical framework of complex signal processing, and thus, provide the expansion of the $\mathbb{CR}$ calculus to functions of a complex matrix variable and the convergence analysis of the generalised complex FastICA algorithm.

- Finally, the statistical and analytical framework in this work is not limited to the complex domain, the extensions to the three-dimensional quaternion space $\mathbb{H}$ are explored. Signal processing in the quaternion domain is quickly becoming an active area of research, and thus it is timely to consider the application of current findings to the quaternion domain; this includes the proposed quaternion BSE and quaternion FastICA algorithms, both suitable for noncircular quaternion-valued signals.

## 1.4 Organisation of thesis

This thesis is organised as follows. In Chapter 2, augmented complex statistics is introduced, forming the statistical framework for the rest of the work in this thesis. Second-order statistics of complex sources is introduced using the duality of the complex and real domains, and the complex kurtosis and a measure of noncircularity are discussed. After a brief discussion of complex-valued noise, the widely linear model is introduced and a comparison with the standard linear model is provided. Chapter 3 introduces the augmented (widely linear) complex least mean square (ACLMS) algorithm, and its derivation using $\mathbb{CR}$ calculus is provided. The applicability of the algorithm in wind profile prediction and hybrid filtering is explored.

Chapter 4 introduces a prediction based complex BSE algorithm by exploiting the temporal structure of the latent sources. The algorithm is derived based on a modified cost function and is capable of extracting sources from mixtures in noisy environments. In Chapter 5, a class of complex BSE algorithms based on the degree of Gaussianity of sources, and capable of extraction of sources in noisy and noise free environments, is introduces. The chapter provides a study on real-time EEG artifact extraction using the proposed algorithm. Chapter 6 introduces a fast converging algorithm based on the generalised complex FastICA, capable of extracting smooth complex sources. The concept of smoothness in $\mathbb{C}$ is discussed and the application of the algorithm in EEG conditioning is given. Chapter 7 introduces a novel quaternion FastICA algorithm for the processing of the generality of quaternion sources. A preliminary study of quaternion algebra, statistics and calculus is provided and the application of the algorithm in the separation of EEG signals is demonstrated. Chapter 8 provides conclusions and directions for future work.

Several key concepts relevant to this work are provided in the appendices. Appendix A complements the discussion on augmented complex statistics and introduces the complex generalised Gaussian distribution, while Appendix B provides an overview of the $\mathbb{CR}$ calculus framework for functions of complex vector variables. Appendix C extends this discussion to functions of complex matrix variables and provides several

application examples. Appendix D gives an overview of the complex FastICA and the generalised complex FastICA algorithms and discusses the convergence of the generalised complex FastICA algorithm using three distinct approaches. Appendix E introduces a novel quaternion BSE algorithm, performing extraction of proper and improper quaternion sources by exploiting the temporal structure of quaternion signals.

# Chapter 2

# Background Theory: Augmented Complex Statistics and Widely Linear Modelling

## 2.1 Complex circularity and second-order statistics

### 2.1.1 Complex circularity

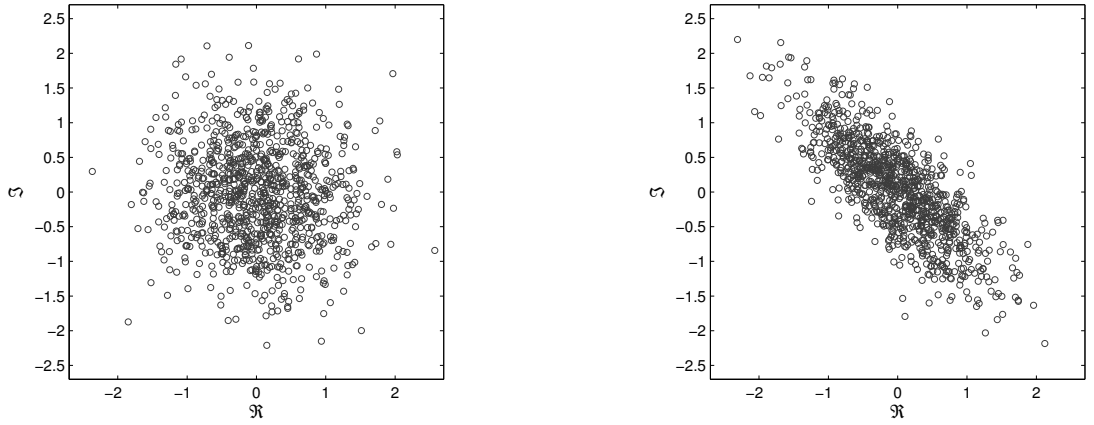
Consider the complex random vector

$$\mathbf{z} = \Re\{\mathbf{z}\} + \jmath\Im\{\mathbf{z}\} = \mathbf{z}_r + \jmath\mathbf{z}_i \in \mathbb{C}^N, \tag{2.1}$$

where $\Re\{\mathbf{z}\} = \mathbf{z}_r$ and $\Im\{\mathbf{z}\} = \mathbf{z}_i$ are respectively its real and imaginary components, $\jmath = \sqrt{-1}$ and $N$ is the number of elements of $\mathbf{z}$. In defining its statistical properties, the random vector $\mathbf{z}$ is called *symmetric* [73] if it has the same probability distribution as $-\mathbf{z}$. A more restricted version of this definition is *circular symmetry* [43], whereby $\mathbf{z}$ and $e^{\jmath\varphi}\mathbf{z}$ have the same probability distribution for all angles $\varphi \in \mathbb{R}$, or intuitively, the distribution of complex circular $\mathbf{z}$ is said to be rotation invariant. Conversely, a random vector which does not satisfy this condition is called complex *noncircular*. Historically, this definition has roots in past literature pertaining to the study of Gaussian distributions in the complex domain.

Considering the simple case of a scalar complex circular random variable $z = d_z e^{\jmath\theta_z}$, its probability density function (pdf) can be written in terms of its magnitude $d_z$ and phase $\theta_z$, taken as independent random variables with pdf's $p_D(d_z)$ and $p_\Theta(\theta_z)$ which is uniformly distributed in $[0, 2\pi]$. Thus [43],

$$p_Z(z) = p_{D,\Theta}(d_z, \theta_z) = \frac{1}{2\pi} p_D(d_z). \tag{2.2}$$




(a) Complex circular random variable (b) Complex noncircular random variable

**Figure 2.1** Scatter plots of circular and noncircular complex Gaussian random variables.

Figure 2.1 depicts the scatter plots of both circular and noncircular Gaussian distributions, where visual inspection confirms the nature of the circularity of the two distributions. The significance of complex circularity on the definition of second-order statistics in $\mathbb{C}$ is considered next.

### 2.1.2 The $\mathbb{R}^2$ interpretation of complex statistics

The pdf and statistical properties of $\mathbf{z}$ are given by the joint pdf of its components, such that $p_{\mathbf{Z}}(\mathbf{z}) = p_{\mathbf{Z}_r,\mathbf{Z}_i}(\mathbf{z}_r, \mathbf{z}_i)$ [44, 43]. This definition can be seen from the fact the standard probability distribution definition in $\mathbb{R}$ is not mathematically meaningful[1] in $\mathbb{C}$. Its expected value is then given by

$$E\{\mathbf{z}\} = E\{\mathbf{z}_r\} + \jmath E\{\mathbf{z}_i\} \tag{2.3}$$

and for a zero-mean random vector, the relationship between its real and imaginary components are given by the four covariance matrices

$$\begin{aligned} \mathcal{C}_{\mathbf{z_r z_r}} &= E\{\mathbf{z}_r \mathbf{z}_r^T\} & \mathcal{C}_{\mathbf{z_r z_i}} &= E\{\mathbf{z}_r \mathbf{z}_i^T\} \\ \mathcal{C}_{\mathbf{z_i z_r}} &= E\{\mathbf{z}_i \mathbf{z}_r^T\} & \mathcal{C}_{\mathbf{z_i z_i}} &= E\{\mathbf{z}_i \mathbf{z}_i^T\}, \end{aligned} \tag{2.4}$$

where $\mathcal{C}_{\mathbf{z_i z_r}} = \mathcal{C}_{\mathbf{z_r z_i}}^T$. A more compact representation is provided by considering the composite vector $\mathbf{z}^R = [\mathbf{z}_r^T \ \mathbf{z}_i^T]^T$, where the covariance matrices in (2.4) are represented

[1] For a real-valued random variable $x$, the cumulative distribution function $F_X$ is defined as $F_X = P(X \leq x)$. The $\mathbb{C}$ domain is not ordered and inequality relations such as '$<$' and '$>$' are thus not defined.

by [44, 45, 48]

$$\mathcal{C}_{\mathbf{zz}}^{R} = E\left\{\begin{bmatrix}\mathbf{z}_r \\ \mathbf{z}_i\end{bmatrix}\begin{bmatrix}\mathbf{z}_r^T & \mathbf{z}_i^T\end{bmatrix}\right\} = E\{\mathbf{z}^R\mathbf{z}^{RT}\}$$
$$= \begin{bmatrix}\mathcal{C}_{\mathbf{z_r z_r}} & \mathcal{C}_{\mathbf{z_r z_i}} \\ \mathcal{C}_{\mathbf{z_i z_r}} & \mathcal{C}_{\mathbf{z_i z_i}}\end{bmatrix} \in \mathbb{R}^{2N\times 2N}. \tag{2.5}$$

### 2.1.3 Augmented complex statistics

While defining the second-order statistics of a complex random vector $\mathbf{z}$ in terms of a pair of real-valued random vectors ($\mathbf{z}_r$ and $\mathbf{z}_i$) allows for its statistical analysis, it would be more appropriate to alternatively consider the statistical relationship directly in $\mathbb{C}$. To this end, complex random vectors can be modelled directly in the complex domain, by establishing the duality with its bivariate real alternative in $\mathbb{R}^2$.

The transformation[2]

$$\mathbf{J}_N = \begin{bmatrix}\mathbf{I} & \jmath\mathbf{I} \\ \mathbf{I} & -\jmath\mathbf{I}\end{bmatrix} \tag{2.6}$$

establishes this duality, where $\mathbf{J}_N$ is a square block matrix of size $2N \times 2N$ and $\mathbf{I}$ is the identity matrix of size $N \times N$. To keep the notation simple, wherever clear, the subscript $N$ is omitted from the definition. The duality between the two domains is then established as[3]

$$\mathbf{z}^a \triangleq \begin{bmatrix}\mathbf{z} \\ \mathbf{z}^*\end{bmatrix} = \mathbf{J}\mathbf{z}^R = \begin{bmatrix}\mathbf{I} & \jmath\mathbf{I} \\ \mathbf{I} & -\jmath\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{z}_r \\ \mathbf{z}_i\end{bmatrix} \tag{2.7}$$

where $\mathbf{z}^a$ is referred to as an *augmented* random vector[4]. Note that the pdf of the complex random vector can also be formally written as $p_{\mathbf{Z},\mathbf{Z}^*}(\mathbf{z},\mathbf{z}^*) = p_{\mathbf{Z}^a}(\mathbf{z}^a) = p_{\mathbf{Z}_r,\mathbf{Z}_i}(\mathbf{z}_r,\mathbf{z}_i)$.

An alternative view in support of the augmented representation of $\mathbf{z}$ simply notes that both $\mathbf{z}$ and its conjugate $\mathbf{z}^*$ are necessary to express the real and imaginary components, that is

$$\mathbf{z}_r = \frac{1}{2}(\mathbf{z}+\mathbf{z}^*) \qquad\qquad \mathbf{z}_i = \frac{1}{2\jmath}(\mathbf{z}-\mathbf{z}^*). \tag{2.8}$$

[2]Alternatively, by using the scaling factor $\frac{1}{\sqrt{2}}$ in the definition in (2.6), the matrix $\mathbf{J}_N$ can be defined as a unitary matrix [48].

[3]The inverse of this mapping can be easily calculated as $\mathbf{J}_N^{-1} = \frac{1}{2}\mathbf{J}_N^H$ providing the mapping from $\mathbb{C}^{2N}$ to $\mathbb{R}^{2N}$.

[4]The transformation $\mathbf{J}_N$ was used in earlier work by van den Bos [50, 51], and was formalised in [48] by Schreier and Scharf.

Thus, the augmented representation is required to fully model the second-order statistical information within the complex domain, in an equivalent manner to (2.4) or (2.5) given by the real bivariate random vector.

The augmented covariance matrix $\mathcal{C}^a_{\mathbf{zz}}$ is then given by [48]

$$\begin{aligned}\mathcal{C}^a_{\mathbf{zz}} &= E\left\{\begin{bmatrix}\mathbf{z}\\ \mathbf{z}^*\end{bmatrix}\begin{bmatrix}\mathbf{z}^H & \mathbf{z}^T\end{bmatrix}\right\}\\ &= \begin{bmatrix}\mathcal{C}_{\mathbf{zz}} & \mathcal{P}_{\mathbf{zz}}\\ \mathcal{P}^*_{\mathbf{zz}} & \mathcal{C}^*_{\mathbf{zz}}\end{bmatrix},\end{aligned} \tag{2.9}$$

where the covariance $\mathcal{C}_{\mathbf{zz}}$ and pseudo-covariance $\mathcal{P}_{\mathbf{zz}}$ matrices are defined as [44, 45]

$$\begin{aligned}\mathcal{C}_{\mathbf{zz}} &= E\{\mathbf{z}\mathbf{z}^H\} = \mathcal{C}_{\mathbf{z_r z_r}} + \mathcal{C}_{\mathbf{z_i z_i}} + \jmath(\mathcal{C}^T_{\mathbf{z_r z_i}} - \mathcal{C}_{\mathbf{z_r z_i}})\\ \mathcal{P}_{\mathbf{zz}} &= E\{\mathbf{z}\mathbf{z}^T\} = \mathcal{C}_{\mathbf{z_r z_r}} - \mathcal{C}_{\mathbf{z_i z_i}} + \jmath(\mathcal{C}^T_{\mathbf{z_r z_i}} + \mathcal{C}_{\mathbf{z_r z_i}}).\end{aligned} \tag{2.10}$$

Based on the established duality with $\mathbb{R}^2$, the augmented covariance matrix (2.9) provides an equivalent representation of the second-order statistical information available within the real and imaginary components, given by (2.5), directly within $\mathbb{C}$. The mapping and inverse mapping between the two covariance matrices are given by [51]

$$\begin{aligned}\mathcal{C}^a_{\mathbf{zz}} &= \mathbf{J}\mathcal{C}^R_{\mathbf{zz}}\mathbf{J}^H\\ \mathcal{C}^R_{\mathbf{zz}} &= \frac{1}{4}\mathbf{J}^H\mathcal{C}^a_{\mathbf{zz}}\mathbf{J}\end{aligned} \tag{2.11}$$

which can be calculated based on the transformation defined in (2.6). The consideration of the pseudo-covariance in addition to the covariance is referred to as *augmented* complex statistics.

### 2.1.4 The covariance and pseudo-covariance

Having established the augmented statistics in $\mathbb{C}$, the two matrices $\mathcal{C}$ and $\mathcal{P}$ are considered. In the literature, $\mathcal{P}$ is referred to as the relation matrix [45] or complementary covariance matrix [48] as well as the pseudo-covariance matrix [44]. The covariance matrix is complex, Hermitian and positive semi-definite, while the pseudo-covariance is complex and symmetric [45].

The standard covariance can be seen as the correlation of $\mathbf{z}$ and itself, while the pseudo-covariance measures the correlation between $\mathbf{z}$ and its conjugate $\mathbf{z}^*$ [48]. A complex random vector with a vanishing pseudo-covariance is termed *second order circular* or *proper* [43, 44], that is, $\mathcal{P}_{\mathbf{zz}} = \mathbf{0}$, or otherwise termed *improper*. The augmented covariance matrix $\mathcal{C}^a_{\mathbf{zz}}$ in Equation (2.9) for a proper complex random vector is then a block-diagonal matrix. In general, the term *circular* refers to a signal with rotation

invariant probability distribution, while properness (also, propriety or second order circularity) specifically refers to the second order statistical properties.

Likewise, using the bivariate representation of $\mathbf{z}$ and based on Equation (2.10), a complex random vector is proper if [44]

$$\mathcal{C}_{\mathbf{z_r z_r}} = \mathcal{C}_{\mathbf{z_i z_i}} \qquad \text{and} \qquad \mathcal{C}_{\mathbf{z_r z_i}} = -\mathcal{C}^T_{\mathbf{z_r z_i}}, \tag{2.12}$$

that is, the real and imaginary parts of each component $z_n$ of $\mathbf{z}$ possess equal power and are uncorrelated. The complex covariance and pseudo-covariance matrices in Equation (2.10) are then simplified as

$$\begin{aligned} \mathcal{C}_{\mathbf{zz}} &= \mathcal{C}_{\mathbf{z_r z_r}} - \jmath \mathcal{C}_{\mathbf{z_r z_i}} \\ \mathcal{P}_{\mathbf{zz}} &= \mathbf{0}. \end{aligned} \tag{2.13}$$

Note the following on the skew-symmetric structure of $\mathcal{C}_{\mathbf{z_r z_i}}$ owing to the properness of $\mathbf{z}$. Its main diagonal containing the covariances of the real and imaginary part of the $n$th component are uncorrelated and zero, $E\{z_{r,n} z_{i,n}\} = 0$, while the off-diagonal cross-covariance elements pertaining to the $n$th and $m$th components, $E\{z_{r,n} z_{i,m}\}$, are not necessarily zero. Therefore, while the covariance $\mathcal{C}$ is a standard complex covariance, the pseudo-covariance $\mathcal{P}$ accounts for the correlation between the real and imaginary components.

Rearranging the terms in Equation (2.10) and representing the covariance matrices in (2.4) in terms of the covariance $\mathcal{C}$ and pseudo-covariance $\mathcal{P}$, gives [44, 45]

$$\begin{aligned} \mathcal{C}_{\mathbf{z_r z_r}} &= \frac{1}{2}\Re\{\mathcal{C}_{\mathbf{zz}} + \mathcal{P}_{\mathbf{zz}}\} & \mathcal{C}_{\mathbf{z_r z_i}} &= -\frac{1}{2}\Im\{\mathcal{C}_{\mathbf{zz}} - \mathcal{P}_{\mathbf{zz}}\} \\ \mathcal{C}_{\mathbf{z_i z_r}} &= \frac{1}{2}\Im\{\mathcal{C}_{\mathbf{zz}} + \mathcal{P}_{\mathbf{zz}}\} & \mathcal{C}_{\mathbf{z_i z_i}} &= \frac{1}{2}\Re\{\mathcal{C}_{\mathbf{zz}} - \mathcal{P}_{\mathbf{zz}}\}. \end{aligned} \tag{2.14}$$

Irrespective of the properness of $\mathbf{z}$, the elements $z_n, n = \{1, \ldots, N\}$ of the random vector $\mathbf{z}$ are uncorrelated if all four real-valued covariance matrices are diagonal matrices. Alternatively, based on (2.14) the complex covariance and pseudo-covariance matrices $\mathcal{C}_{\mathbf{zz}}$ and $\mathcal{P}_{\mathbf{zz}}$ are diagonal matrices [44, 69].

An uncorrelated covariance matrix in $\mathbb{R}$ is achieved by using a whitening matrix. However, in $\mathbb{C}$, based on the above definition of uncorrelated random vectors, it is necessary to diagonalise both covariance and pseudo-covariance matrices. This is accomplished by using the procedure known as the strong uncorrelating transform (SUT) [69] and based on Takagi's factorisation [70], a special form of the singular value decomposition (SVD). In this manner, the covariance matrix $\mathcal{C}$ is diagonalised with diagonal elements with unit variance (whitened), while the pseudo-covariance matrix $\mathcal{P}$ is diagonalised with the diagonal elements being its singular values and termed the circularity coefficients [69] or canonical correlations [79].

Thus for a random vector with uncorrelated components, the diagonal elements of the covariance matrix form the standard complex variance and are denoted by

$$\sigma_{z_n}^2 = E\{z_n z_n^*\} = E\{|z_n|^2\} \tag{2.15}$$

and the diagonal elements of the pseudo-covariance matrix form the pseudo-variance, denoted by

$$\tau_{z_n}^2 = E\{z_n z_n\} = E\{z_n^2\}. \tag{2.16}$$

Note that while the variance $\sigma_{z_n}^2$ is real-valued, the pseudo-variance $\tau_{z_n}^2$ is normally complex-valued [72].

For completeness and based on the discussion so far on second-order circularity, a complex generalised Gaussian distribution (GGD) capable of modelling the pdf of both sub- and super-Gaussian circular and noncircular random vectors is provided in Appendix A. As a special case, the complex Gaussian distribution is studied and its properties discussed.

### 2.1.5 A measure of second-order circularity

The degree of noncircularity can be quantified by the circularity measure $r$, defined in [80] as the magnitude of the circularity quotient $\rho(z) = re^{\jmath\theta} \triangleq \tau_z^2/\sigma_z^2$, where

$$r = |\rho(z)| = \frac{|\tau_z^2|}{\sigma_z^2}, \quad r \in [0, 1] \tag{2.17}$$

measures the degree of noncircularity in the complex signal[5], with the circularity angle $\theta = \arg\big(\rho(z)\big)$ indicating orientation of the distribution. Note that for a purely circular signal, $r = 0$, with $\theta$ not providing additional information about the distribution.

This circularity measure can also be graphically interpreted using an ellipse (centred in the complex plane) of eccentricity $\epsilon$ and orientation $\alpha$, such that $r = \epsilon^2$ and $\theta = 2\alpha$ [80, Theorem1]. For $\epsilon = 0$, the shape becomes a circle, which also indicates a circular signal with $r = 0$, while for the extreme case of $\epsilon = 1$, corresponding to a highly noncircular signal with $r = 1$, the ellipse becomes elongated with a maximal major axis and minor axis of length zero. Note that the pseudo-variance of a general complex Gaussian distribution is then related to the elliptic shape by $\tau^2 = \epsilon^2 e^{\jmath 2\theta}$ [72].

[5]Other measures of noncircularity are also defined and may be used. A similar measure to (2.17) and given by $1 - r$ was defined in [81]. In [79], measures bounded between $[0, 1]$ and based on the canonical correlations are defined. The authors in [82] define the same measure as in Equation (2.17), albeit with different terminology. Finally, an unbounded measure in $[1, \infty]$ based on the ratio of the standard deviations of the real and imaginary components of the complex random variable was introduced in [78]. While the mentioned measures are quite similar, the simplicity of (2.17) and the embedded information within the circularity quotient $\rho(z)$, makes it a suitable noncircularity measure in this work.

### 2.1.6 Spectral interpretation of second-order circularity

A discrete complex random process $z(k)$ is termed *wide sense stationary* [47] if it has constant mean, and its covariance $\mathcal{C}_{zz}(k_1, k_2) = E\{z(k_1)z(k_2)^*\}$ is a function of the delay $\delta = k_1 - k_2$. In this definition, no assumption is made on the pseudo-covariance $\mathcal{P}_{zz}(k_1, k_2) = E\{z(k_1)z(k_2)\}$ of the random process. However, the more restricted definition *second-order stationarity* [47] imposes that both the covariance and pseudo-covariance are functions of the delay $\delta$. Thus, for a second-order stationary random process[6]

$$\begin{aligned}\mathcal{C}_{zz}(\delta) &= E\{z(k)z^*(k-\delta)\} \\ \mathcal{P}_{zz}(\delta) &= E\{z(k)z(k-\delta)\}.\end{aligned} \tag{2.18}$$

Then, the augmented covariance matrix of a complex random process $\mathbf{z}(k)$ is given by

$$\begin{aligned}\mathcal{C}^a_{\mathbf{zz}}(\delta) &= E\left\{\begin{bmatrix}\mathbf{z}(k)\\ \mathbf{z}^*(k)\end{bmatrix}\begin{bmatrix}\mathbf{z}^H(k-\delta) & \mathbf{z}^T(k-\delta)\end{bmatrix}\right\} \\ &= \begin{bmatrix}\mathcal{C}_{\mathbf{zz}}(\delta) & \mathcal{P}_{\mathbf{zz}}(\delta)\\ \mathcal{P}^*_{\mathbf{zz}}(\delta) & \mathcal{C}^*_{\mathbf{zz}}(\delta)\end{bmatrix}.\end{aligned} \tag{2.19}$$

The transformation of this matrix to the frequency domain gives the augmented spectral matrix [47, 48]

$$\mathcal{S}^a_{\mathbf{z}}(\omega) = \begin{bmatrix}\mathcal{S}_{\mathbf{z}}(\omega) & \breve{\mathcal{S}}_{\mathbf{z}}(\omega)\\ \breve{\mathcal{S}}^*_{\mathbf{z}}(-\omega) & \mathcal{S}_{\mathbf{z}}(-\omega)\end{bmatrix}, \tag{2.20}$$

with the Fourier transforms of the covariance and pseudo-covariance matrices defined respectively as $\mathcal{S}_{\mathbf{z}}(\omega)$ and $\breve{\mathcal{S}}_{\mathbf{z}}(\omega)$, that is

$$\begin{aligned}\mathcal{S}_{\mathbf{z}}(\omega) &= \mathcal{F}\big(\mathcal{C}_{\mathbf{zz}}(\delta)\big) = \mathcal{F}\big(E\{\mathbf{z}(k)\mathbf{z}^H(k-\delta)\}\big) \\ \breve{\mathcal{S}}_{\mathbf{z}}(\omega) &= \mathcal{F}\big(\mathcal{P}_{\mathbf{zz}}(\delta)\big) = \mathcal{F}\big(E\{\mathbf{z}(k)\mathbf{z}^T(k-\delta)\}\big)\end{aligned} \tag{2.21}$$

where $\mathcal{F}(\cdot)$ denotes the Fourier transform operator. For a proper complex random process, the augmented spectral matrix is block diagonal, with vanishing pseudo-spectral components, $\breve{\mathcal{S}}_{\mathbf{z}}(\omega) = \mathbf{0}$.

While the power spectrum provides information on the distribution of signal power over a frequency range, the magnitude of the pseudo-spectrum characterises the second-order circularity of the random variable in the frequency domain. The augmented spectral matrix in (2.20) is positive semi-definite which results in the condition [47]

$$|\breve{\mathcal{S}}_{\mathbf{z}}(\omega)|^2 \leq \mathcal{S}_{\mathbf{z}}(\omega) \cdot \mathcal{S}_{\mathbf{z}}(-\omega). \tag{2.22}$$

[6]Note that the terminology used by the authors in [48] defines wide sense stationarity as the restricted second-order stationarity given in [47] and in this work in Equation (2.18).

## 2.2 Kurtosis of complex random vectors

The definition of kurtosis in the complex domain based on fourth order cumulants is not a straightforward extension from $\mathbb{R}$. In fact, the placement of the random variable and its conjugate operator in the definition of the fourth order cumulant produces 16 variations[7] for its definition [83, 84]. The most common definition in literature [83] is considered in this work[8].

In the real domain, it is common to use the normalised kurtosis $K_R(\cdot)$ instead of the standard kurtosis $\mathrm{kurt}_R(\cdot)$, as it allows for the comparison of the degree of non-Gaussianity of random variables, irrespective of the range of amplitudes. Likewise, the normalised kurtosis of a complex random variable can be defined as

$$\begin{aligned} K_c(z) &= \frac{\mathrm{kurt}_c(z)}{(E\{|z|^2\})^2} \\ &= \frac{E\{|z|^4\}}{(E\{|z|^2\})^2} - \frac{|E\{z^2\}|^2}{(E\{|z|^2\})^2} - 2 \end{aligned} \tag{2.23}$$

with

$$\mathrm{kurt}_c(z) = E\{|z|^4\} - |E\{z^2\}|^2 - 2(E\{|z|^2\})^2. \tag{2.24}$$

The first term in (2.23) is the normalised fourth order moment, the second term is the square of the circularity coefficient $r$ (Equation (2.17)), whereas $\mathrm{kurt}_c(z)$ in (2.24) is the real-valued kurtosis of the complex random variable $z$. Similar to the kurtosis of a real-valued Gaussian random variable, the value of $K_c$ is zero for both circular and noncircular complex Gaussian random variables. Furthermore, for continuity, this measure makes kurtosis values of a sub-Gaussian complex random variable negative and that of a super-Gaussian complex random variable positive, irrespective of the degree of noncircularity.

The relation between the kurtosis of the real and imaginary components of a complex random variable, $\mathrm{kurt}_R(z_r)$ and $\mathrm{kurt}_R(z_i)$ and the kurtosis of the complex random variable $\mathrm{kurt}_c(z)$ is given by [85]

$$\mathrm{kurt}_R(z_r) = \mathrm{kurt}_R(z_i) = \left(\frac{3}{2+r^2}\right)\mathrm{kurt}_c(z), \tag{2.25}$$

that is, the complex kurtosis is a scaled version of the kurtosis of its real and imaginary components. Notice that for a proper random variable ($r = 0$), the scaling is 1.5, while for a highly improper random variable ($r = 1$), the complex kurtosis is equal to the kurtosis of its real and imaginary components.

[7] For example, consider the cumulants $\mathrm{cum}(z(k), z(k+\delta_1), z(k+\delta_2), z(k+\delta_3))$ and $\mathrm{cum}(z(k), z(k+\delta_1), z(k+\delta_2), z^*(k+\delta_3))$.

[8] That is, the cumulant $\mathrm{cum}(z(k), z^*(k+\delta_1), z(k+\delta_2), z^*(k+\delta_3))$ which results in a real-valued measure of complex kurtosis.

## 2.3 Complex-valued noise

It is important to notice that the treatment of a noise vector $\mathbf{v}(k)$ in $\mathbb{C}$ is different to that in the real domain [47]. While in $\mathbb{R}$ only the variance $\sigma_v^2$ of the noise signal is of concern, in $\mathbb{C}$ it is necessary to also consider the pseudo-variance $\tau_v^2$, in order to completely model the noise. White noise can be differentiated in the following cases.

i) *Circular white noise*, is considered white in terms of its diagonal covariance matrix, whereas the pseudo-covariance matrix vanishes, that is

$$\mathcal{C}_{\mathbf{vv}}(\delta) = \sigma_v^2\mathbf{I}, \quad \mathcal{P}_{\mathbf{vv}}(\delta) = \mathbf{0}, \quad \delta = 0 \tag{2.26}$$

In other words, the real and imaginary part of the complex noise $\mathbf{v}(k) = \mathbf{v}_r(k) + \jmath\mathbf{v}_i(k)$ are of equal power and uncorrelated, and as $E\{\mathbf{v}(k)\mathbf{v}^T(k)\} = E\{\mathbf{v}_r(k)\mathbf{v}_r^T(k)\} - E\{\mathbf{v}_i(k)\mathbf{v}_i^T(k)\} = \mathbf{0}$, the pseudo-covariance matrix of the second-order circular noise vanishes. In the frequency domain, the covariance spectrum $\mathcal{S}_{\mathbf{v}}(\omega)$ (also power spectrum, or PSD) of the circular white noise is flat, while the pseudo-covariance spectrum $\breve{\mathcal{S}}_{\mathbf{v}}(\omega)$ (or pPSD) is zero.

ii) *Noncircular doubly white noise*, is assumed white for both the covariance and pseudo-covariance matrices, where the distributions and power levels of the real and imaginary components may be different, such that

$$\mathcal{C}_{\mathbf{vv}}(\delta) = \sigma_v^2\mathbf{I}, \quad \mathcal{P}_{\mathbf{vv}}(\delta) = \tau_v^2\mathbf{I}, \quad \delta = 0,\ \sigma_v^2 \neq \tau_v^2. \tag{2.27}$$

In this case, the power spectrum is flat across all frequencies, while the pseudo-spectrum is non-zero. As the noise becomes more noncircular ($r \to 1$), the pseudo-spectrum approaches its upper-bound defined in (2.22) where for highly noncircular noise ($r \approx 1$), the magnitudes of the pPSD and PSD are similar.

For a scalar complex white noise signal $v(k)$, the relations between the correlation and pseudo-correlation and the respective spectra are given by

$$\begin{aligned} \mathcal{C}(\delta) = E\{v(k)v^*(k-\delta)\} = \delta_0\sigma_v^2 \quad &\xrightarrow{\mathcal{F}} \quad \mathcal{S}(\omega) = |\sigma_v^2| \\ \mathcal{P}(\delta) = E\{v(k)v(k-\delta)\} = \delta_0\tau_v^2 \quad &\xrightarrow{\mathcal{F}} \quad \breve{\mathcal{S}}(\omega) = |\tau_v^2|, \end{aligned} \tag{2.28}$$

where $\delta_0$ is the Kronecker delta function. Then the bound can be expressed as

$$|\tau_v^2| \leq \sigma_v^2, \tag{2.29}$$

that is, the magnitude of the noise pseudo-variance cannot exceed the noise power.

Examples of circular white Gaussian and Laplacian noise with unit variance are illustrated in the left hand column of Fig. 2.2(a), whereas the right hand column demonstrates two examples of noncircular white noise, with the top-right plot showing a

noncircular Gaussian noise signal with circularity measure $r = 0.81$ with unit variance and pseudo-variance $\tau_v^2 = -0.38 + \jmath 0.71$, and the bottom-right plot illustrating the scatter plot of noncircular Laplacian noise with circularity measure $r = 0.81$ with unit variance and pseudo-variance of $0.45 - \jmath 0.66$. Also note that in Figure 2.2(a) the value of the kurtosis[9] is approximately zero for both the circular and noncircular Gaussian noise signals, whereas the kurtosis values for the circular and noncircular super-Gaussian noise signals follow the real-valued convention and are positive valued.

Figure 2.2(b) depicts the PSD and pPSD of circular ($r = 0$) white and noncircular doubly white Gaussian noise for the respective circularity measures $r = \{0.64, 1\}$. Observe that the pseudo-spectrum is zero for the circular noise, while it has a magnitude[10] of $|\tau_v^2| = 0.64$ for the noise with $r = 0.64$, and reaches it upper-bound of 1 in the third realisation where the noise is highly noncircular ($r = 1$). For the Gaussian noise, the spectrum $\mathcal{S}(\omega) = 1$ and the pseudo-spectrum $\breve{\mathcal{S}}(\omega) = |\tau_v^2| = |\epsilon^2 e^{\jmath 2\theta}| = |\epsilon^2| = r = 1$, across all frequencies, thus indicating that by increasing the eccentricity of the ellipse (degree of noncircularity), the magnitude of the pPSD approaches its maximum value of 1.

## 2.4 Widely linear modelling

Consider the minimum mean square error (MSE) estimator of a complex signal $y$ in terms of a complex valued observation vector $\mathbf{x}$, given by the conditional expectation $\hat{y} = E\{y|\mathbf{x}\}$. The MSE estimator of the real and imaginary components of the signal $y(k)$ are given by

$$\begin{aligned} \hat{y}_r &= E\{y_r|\mathbf{x}_r, \mathbf{x}_i\} \\ \hat{y}_i &= E\{y_i|\mathbf{x}_r, \mathbf{x}_i\} \end{aligned} \tag{2.30}$$

and $\hat{y}$ is then expressed as

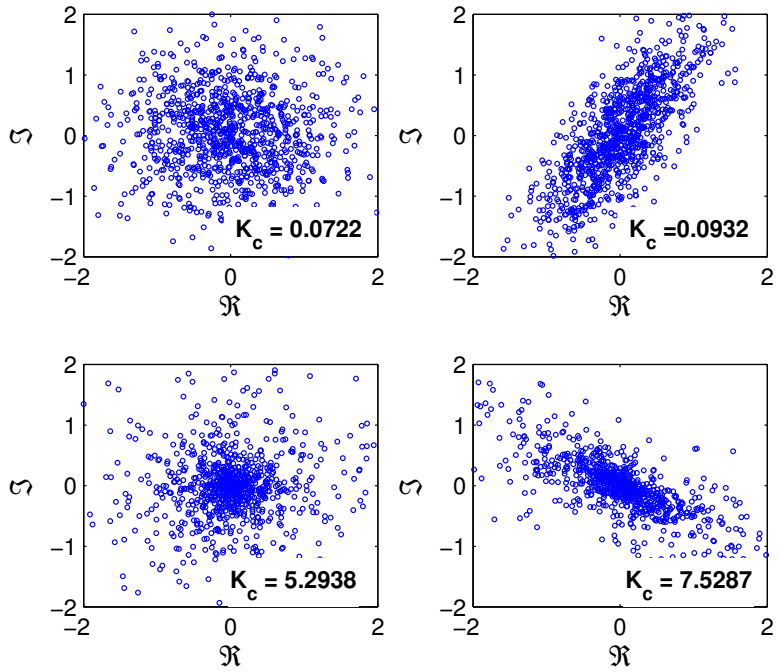
$$\begin{aligned} \hat{y} &= \hat{y}_r + \jmath \hat{y}_i \\ &= E\{y_r|\mathbf{x}_r, \mathbf{x}_i\} + \jmath E\{y_i|\mathbf{x}_r, \mathbf{x}_i\}. \end{aligned} \tag{2.31}$$

By using the relation (2.8), Equation (2.31) can be equivalently written as

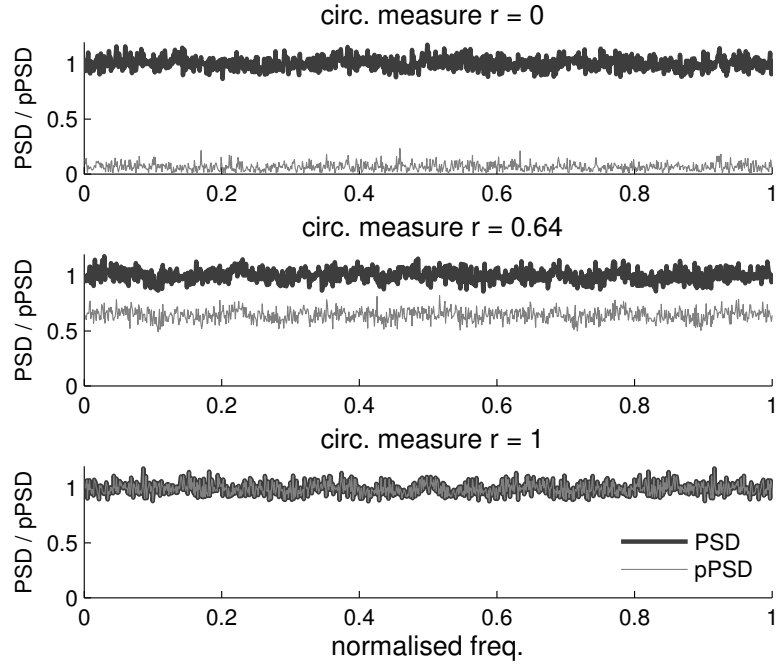
$$\hat{y} = E\{y_r|\mathbf{x}, \mathbf{x}^*\} + \jmath E\{y_i|\mathbf{x}, \mathbf{x}^*\}, \tag{2.32}$$

[9] The kurtosis values in Figure 2.2(a) are estimated based on 5000 samples and are not the true kurtosis value.

[10] Recall from Section 2.1.5 the relationship between the pseudo-variance $\tau_v^2$, elliptic eccentricity $\epsilon$ and circularity measure $r$ of a complex Gaussian random variable, given by $\tau_v^2 = \epsilon^2 e^{\jmath 2\theta} = r e^{\jmath 2\theta}$.




(a) Scatter plots of complex white noise realisations. *Top row:* circular Gaussian noise (left) and noncircular Gaussian noise ($r = 0.81$) (right). *Bottom row:* circular Laplacian noise (left) and noncircular Laplacian noise ($r = 0.81$) (right). The circularity measure $r$ is defined in (2.17). The kurtosis values $K_c$ are given for each case.




(b) Power spectra (thick gray line) and pseudo-power spectra (thin gray line) of complex Gaussian noises with varying degrees of noncircularity $r = \{0, 0.64, 1\}$

**Figure 2.2** Illustration of doubly white circular and noncircular complex-valued noises.

demonstrating that the estimator of $y$ is found in terms of the observation $\mathbf{x}$ and its conjugate $\mathbf{x}^*$. Thus, the solution is written as the *widely linear* (WL) model [46, 47]

$$\hat{y}_{WL} = \mathbf{h}^T\mathbf{x} + \mathbf{g}^T\mathbf{x}^* \tag{2.33}$$
$$= \mathbf{w}^{aT}\mathbf{x}^a \tag{2.34}$$

where $\mathbf{h}$ and $\mathbf{g}$ are coefficient vectors. The WL model can also be expressed using augmented vectors $\mathbf{w}^a = [\mathbf{h}^T\,\mathbf{g}^T]^T$ and $\mathbf{x}^a = [\mathbf{x}^T\,\mathbf{x}^H]^T$, which provides a more compact representation.

Note the contrast to the standard complex linear model[11],

$$\hat{y}_L = \mathbf{h}^H\mathbf{x} \tag{2.35}$$

which is sub-optimal in the minimum mean square error for noncircular complex-valued signals. This can be shown by considering the minimum MSE of the widely linear approach $E\{|e_{WL}|^2\} = E\{|y - \hat{y}_{WL}|^2\}$. Utilising the compact form of $\hat{y}$ in Equation (2.34), the Wiener-Hopf equations are solved by

$$\mathbf{w}^a = \mathcal{C}_{\mathbf{xx}}^{a^{-1}}\mathbf{p}_{\hat{y},\mathbf{x}^a} \tag{2.36}$$

where $\mathbf{p}_{\hat{y},\mathbf{x}^a} = E\{\hat{y}^*\mathbf{x}^a\} \triangleq [\mathbf{c}_1^T\,\mathbf{c}_2^T]^T$ is the cross-correlation between $\hat{y}$ and the augmented observation vector $\mathbf{x}^a$. The coefficient vectors $\mathbf{h}$ and $\mathbf{g}$ can be obtained[12] by using the Cholesky block factorisation of $\mathcal{C}_{\mathbf{xx}}^{a^{-1}}$, as given in [45], and simplifying (2.36) to obtain

$$\begin{aligned}\mathbf{h} &= \left(\mathcal{C} - \mathcal{P}\mathcal{C}^{*^{-1}}\mathcal{P}^*\right)^{-1}\left(\mathbf{c}_1 - \mathcal{P}\mathcal{C}^{*^{-1}}\mathbf{c}_2^*\right)\\ \mathbf{g} &= \left(\mathcal{C}^* - \mathcal{P}^*\mathcal{C}^{-1}\mathcal{P}\right)^{-1}\left(\mathbf{c}_2^* - \mathcal{P}^*\mathcal{C}^{-1}\mathbf{c}_2\right)\end{aligned} \tag{2.37}$$

where the subscripts have been omitted for clarity. The widely linear MSE is then given by [46]

$$E\{|e_{WL}|^2\} = E\{yy^*\} - \mathbf{h}^T\mathbf{c}_1 - \mathbf{g}^T\mathbf{c}_2^*. \tag{2.38}$$

However, by considering the linear model (2.35), the coefficient vector obtaining the minimum MSE is given by

$$\mathbf{h} = \mathcal{C}^{-1}\mathbf{c}_1 \tag{2.39}$$

[11] Both $y_L = \mathbf{h}^T\mathbf{x}$ and $y_L = \mathbf{h}^H\mathbf{x}$ are correct yielding the same output and the mutually conjugate coefficient vectors. The latter form is more common and the former was used in the original CLMS paper [39]. This also applies to the definition of the widely linear model in (2.33).

[12] Alternatively, the authors in [46] use the orthogonality principle to obtain this result.

and the linear MSE [13]

$$E\{|e_L|^2\} = E\{yy^*\} - \mathbf{c}_1^H \mathcal{C}^{-1} \mathbf{c}_1. \tag{2.40}$$

Comparison of the widely linear MSE (2.38) and the linear MSE (2.40) results in the magnitude difference $\Delta_{MSE}$ quantified as [46]

$$\Delta_{MSE} = \left(\mathbf{c}_2^* - \mathcal{P}^* \mathcal{C}^{-1} \mathbf{c}_1\right)^H \left(\mathcal{C}^* - \mathcal{P}^* \mathcal{C}^{-1} \mathcal{P}\right) \left(\mathbf{c}_2^* - \mathcal{P}^* \mathcal{C}^{-1} \mathbf{c}_1\right), \tag{2.41}$$

where the value $\Delta_{MSE} \geq 0$ and equals zero when $\mathbf{c}_2^* - \mathcal{P}^* \mathcal{C}^{-1} \mathbf{c}_1 = \mathbf{0}$. Thus the widely linear model (2.33) yields a smaller magnitude MSE compared to a linear model (2.35). The MSE difference $\Delta_{MSE} = 0$ only for a second-order circular signal $y$ and observation $\mathbf{x}$, such that $\mathcal{P}_{\mathbf{xx}} = \mathbf{0}$ and cross-correlation $\mathbf{c}_2 = \mathbf{0}$ [46].

Based on the above results, observe that the linear model is sub-optimal for the generality of complex-valued signals, and can be seen as a special case of the WL model suitable for only second-order circular signals. While the utilisation of a WL model may not appear intuitive at first, the preceding discussions on second-order circularity and augmented statistics along with the comparison of the MSE of the two models demonstrate its usefulness as a de facto standard for linear estimation in $\mathbb{C}$.

[13] The linear MSE can be in fact seen as a straightforward extension of the Wiener-Hopf solution from the real domain.

# Chapter 3

# The Widely Linear Complex Least Mean Square Algorithm

## 3.1 Introduction

The Least Mean Square (LMS) [1] algorithm is a workhorse of adaptive signal processing in $\mathbb{R}$. Direct processing of a complex-valued signal using the LMS algorithm results in a dual univariate approach, whereby the real and imaginary components of the input signal are processed separately. However, the cross-information contained in the real and imaginary components would not be modelled, leading to inadequate performance. Alternatively, bivariate algorithms operating in $\mathbb{R}$, such as the dual channel LMS [86], allow for the consideration of the available cross-information.

A natural extension of the real-valued LMS algorithm for the adaptive filtering directly in the field of complex numbers $\mathbb{C}$ was the Complex LMS (CLMS), introduced by Widrow et al. in 1975 [39]. This algorithm benefits from the robustness and stability of the LMS and enables simultaneous filtering of the real and imaginary components of complex-valued data and accounts for second-order cross-information between the channels. The algorithm was originally designed to cater for cases where a complex output was desired, such as the adaptive filtering of high frequency narrowband signals in the frequency domain [39]. However, the algorithm can also be utilised for processing signals made complex by convenience of representation, such as wind vectors, as discussed in [59].

The CLMS algorithm has been derived as a straightforward extension from the real domain, and under the assumption of circular signals and noises. In this chapter an improved CLMS algorithm is introduced, derived based on the concept of augmented complex statistics and widely linear modelling [47, 46], leading to an optimal algorithm for the generality of signals in $\mathbb{C}$. Based on this principle, the Widely Linear

LMS was introduced in the communications field for use in a direct-sequence code division multiple access (DS–CDMA) receiver [87, 88]. It was shown that the algorithm has a lower complexity, while having an equally good performance to standard linear algorithms.

Recently, the augmented Complex Extended Kalman filter (ACEKF) and augmented Complex Real-Time Recurrent Learning (ACRTRL) were introduced, benefiting from augmented complex statistics and widely linear modelling [89, 57]. Both ACEKF and ACRTRL were derived for general adaptive filtering architectures (recurrent neural networks (RNN)). Although a widely linear CLMS can be seen as a degenerate version of ACRTRL[1], given the number of applications based on CLMS, there is a need to derive a widely linear CLMS directly for a complex-valued FIR filter.

In this chapter, the derivation of the widely linear LMS algorithm, or augmented CLMS (ACLMS), is provided in an adaptive prediction context, and illustrates the improvement in the performance of this algorithm as compared to the standard CLMS algorithm in an adaptive prediction setting for general complex signals. The derivation of the algorithm is provided using the $\mathbb{CR}$ calculus framework where both the derivation directly in $\mathbb{C}$ and also based on the real and imaginary components in $\mathbb{R}$ are presented, highlighting the simplicity of the analysis framework. The application focus is on the forecasting of wind profile, an important problem in renewable energy. In the second part of this chapter, hybrid filtering based on a pair of linear (CLMS) and widely linear (ACLMS) algorithms is introduced, and its application in prediction and signal modality tracking is discussed.

## 3.2 The Augmented CLMS algorithm

The original CLMS algorithm was derived by considering the complex output

$$y_L = \mathbf{h}^H(k)\mathbf{x}(k), \tag{3.1}$$

which as discussed in Chapter 2 is a linear model optimal only for proper complex signals. A more general algorithm can be designed by considering the augmented statistics. Then, the output $y(k)$ of an FIR filter can be written as a widely linear process (see Section 2.4), given by[2]

$$y(k) = \mathbf{h}^T(k)\mathbf{x}(k) + \mathbf{g}^T(k)\mathbf{x}^*(k) \tag{3.2}$$

[1]Since a finite impulse response (FIR) filter can be derived from an RNN by removing the nonlinearity, feedback, and all but one neuron.

[2]Note that the lack of conjugation on the weight vectors $\mathbf{h}$ and $\mathbf{g}$ in Equation (3.2) does not affect the performance of the algorithm. Both forms are correct and result in the same output. The use of conjugation is more common and the use of only the transpose was noted in the original CLMS paper [39] using the linear model.

where $\mathbf{h}(k)$ and $\mathbf{g}(k)$ are complex-valued adaptive weight vectors, $\mathbf{x}(k)$ is the filter input vector, and the weights are updated by minimising the cost function

$$\mathcal{J}(\mathbf{h},\mathbf{g}) = E\{|e(k)|^2\} = E\{e(k)e^*(k)\} = E\{|d(k) - y(k)|^2\} \tag{3.3}$$

where $e(k)$ is the output error and $d(k)$ is the desired signal, and $k$ is the discrete time index.

Derivation of the optimisation algorithm can be performed twofold. The standard derivation method consists of the calculation of the gradients by considering the derivative of $\mathcal{J}$ with respect to the real and imaginary components of the weight vectors $\mathbf{h}$ and $\mathbf{g}$. Alternatively, the $\mathbb{CR}$ calculus (Wirtinger calculus) framework [55, 53, 54] facilitates a simpler derivation method by considering the cost function $\mathcal{J}$ as a function of the conjugate coordinates of the weight vectors, that is $(\mathbf{h},\mathbf{g},\mathbf{h}^*,\mathbf{g}^*)$, allowing for the calculation of the derivatives directly in $\mathbb{C}$. A brief description of $\mathbb{CR}$ calculus is provided in Appendix B.

Thus, in order to demonstrate the usefulness of $\mathbb{CR}$ calculus in comparison to the standard Cauchy-Riemann derivation method, two derivation methods are provided.

### 3.2.1 Derivation based on the real and imaginary components

Using the stochastic gradient based adaptation[3], for the update of the weight vectors gives

$$\mathbf{h}(k+1) = \mathbf{h}(k) - \mu_h \nabla\mathcal{J}\Big|_{h=h(k)} \tag{3.4}$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) - \mu_g \nabla\mathcal{J}\Big|_{g=g(k)} \tag{3.5}$$

and

$$\nabla\mathcal{J}\Big|_{h=h(k)} = \frac{1}{2}\left(\frac{\partial\mathcal{J}}{\partial h_{r,n}(k)} + \jmath\frac{\partial\mathcal{J}}{\partial h_{i,n}(k)}\right) \tag{3.6}$$

$$\nabla\mathcal{J}\Big|_{g=g(k)} = \frac{1}{2}\left(\frac{\partial\mathcal{J}}{\partial g_{r,n}(k)} + \jmath\frac{\partial\mathcal{J}}{\partial g_{i,n}(k)}\right) \tag{3.7}$$

In this setting, $\mu_h$ and $\mu_g$ are the step-sizes, $(\cdot)_r$ and $(\cdot)_i$ denote respectively the real and imaginary parts of a complex number and the subscript $n$ denotes the $n$th element of the weight vector. Since the input to the filter is complex, the error $e(k)$ is also complex

[3] In this and all following chapters, stochastic gradient assumptions are made in the derivation of algorithms.

and therefore the gradients from (3.6) and (3.7) should be evaluated as

$$\frac{\partial \mathcal{J}}{\partial h_{r,n}(k)} = \left( e(k)\frac{\partial e^*(k)}{\partial h_{r,n}(k)} + e^*(k)\frac{\partial e(k)}{\partial h_{r,n}(k)} \right) \tag{3.8}$$

$$\frac{\partial \mathcal{J}}{\partial h_{i,n}(k)} = \left( e(k)\frac{\partial e^*(k)}{\partial h_{i,n}(k)} + e^*(k)\frac{\partial e(k)}{\partial h_{i,n}(k)} \right) \tag{3.9}$$

$$\frac{\partial \mathcal{J}}{\partial g_{r,n}(k)} = \left( e(k)\frac{\partial e^*(k)}{\partial g_{r,n}(k)} + e^*(k)\frac{\partial e(k)}{\partial g_{r,n}(k)} \right) \tag{3.10}$$

$$\frac{\partial \mathcal{J}}{\partial g_{i,n}(k)} = \left( e(k)\frac{\partial e^*(k)}{\partial g_{i,n}(k)} + e^*(k)\frac{\partial e(k)}{\partial g_{i,n}(k)} \right) \tag{3.11}$$

Rewriting (3.3) in terms of its real and imaginary parts and substituting in (3.8)–(3.11) yields

$$\nabla \mathcal{J}\Big|_{h=h(k)} = -e(k)\mathbf{x}^*(k) \tag{3.12}$$

$$\nabla \mathcal{J}\Big|_{g=g(k)} = -e(k)\mathbf{x}(k) \tag{3.13}$$

The weight update equations (3.4) and (3.5) are now given as

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h e(k)\mathbf{x}^*(k) \tag{3.14}$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g e(k)\mathbf{x}(k) \tag{3.15}$$

In order to consolidate (3.14)–(3.15) into a compact vector form, the augmented weight vector $\mathbf{w}^a(k)$ is defined as

$$\mathbf{w}^a(k) = [\mathbf{h}^T(k)\ \mathbf{g}^T(k)]^T \tag{3.16}$$

to give the augmented weight update

$$\mathbf{w}^a(k+1) = \mathbf{w}^a(k) + \mu e^a(k)\mathbf{x}^{a*}(k) \tag{3.17}$$

where

$$e^a(k) = d(k) - \underbrace{\mathbf{x}^{aT}(k)\mathbf{w}^a(k)}_{y(k)}, \tag{3.18}$$

$$\mathbf{x}^a(k) = [\mathbf{x}^T(k)\ \mathbf{x}^H(k)]^T, \tag{3.19}$$

and $\mu = \mu_h = \mu_g$.

### 3.2.2 Derivation using the $\mathbb{CR}$ calculus

The stochastic gradient updates of the two weight vectors of the WL filter using a steepest descent adaptation are given by

$$\mathbf{h}(k+1) = \mathbf{h}(k) - \mu_h \nabla_{\mathbf{h}^*} \mathcal{J} \tag{3.20}$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) - \mu_g \nabla_{\mathbf{g}^*} \mathcal{J}. \tag{3.21}$$

Recall that the direction of steepest descent is given by the $\mathbb{R}^*$–derivative for both update equations[4]. By using $\mathbb{CR}$ calculus and the chain rule (given in Equations (B.13d)–(B.13e)), can be then simply calculated as

$$\nabla_{\mathbf{h}^*}\mathcal{J} = -e(k)\mathbf{x}^*(k)$$
$$\nabla_{\mathbf{g}^*}\mathcal{J} = -e(k)\mathbf{x}(k)$$

and substituted in (3.20) and (3.21) to form the complete update equations for the ACLMS algorithm

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h e(k)\mathbf{x}^*(k) \tag{3.22}$$
$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g e(k)\mathbf{x}(k). \tag{3.23}$$

By making use of an equivalent representation it is also possible to consider the complex vectors as 'augmented' vectors, given by the pair of the complex vector and its complex conjugate, to obtain

$$\mathbf{w}^a(k+1) = \mathbf{w}^a(k) + \mu e^a(k)\mathbf{x}^{a*}(k) \tag{3.24}$$

where $\mu = \mu_h = \mu_g$, $\mathbf{w}^a(k) = [\mathbf{h}^T(k)\ \mathbf{g}^T(k)]^T$ is the augmented coefficient vector, $\mathbf{x}^a(k) = [\mathbf{x}^T(k)\ \mathbf{x}^H(k)]^T$ is the augmented input vector and $e^a(k) = d(k) - \mathbf{x}^{aT}(k)\mathbf{w}^a(k)$ is a complex scalar value measuring the distance of the output of the predictor to the desired signal.

This concludes the derivation of the augmented CLMS (ACLMS) algorithm. Both methods result in the same formulation for the ACLMS algorithm. However, it can be seen that derivation of the algorithm directly in $\mathbb{C}$ using the $\mathbb{CR}$ calculus, results in a simpler and more intuitive way to derive complex valued algorithms. Also note that the derivation using the Cauchy-Riemann equations is equivalent to the calculation of the $\mathbb{R}^*$–derivative based on the real and imaginary components, as shown in the right hand side of relation (B.7).

For second-order circular signals, the ACLMS algorithm reduces to the standard CLMS algorithm

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h e(k)\mathbf{x}^*(k), \tag{3.25}$$

where $\mathbf{g} = \mathbf{0}$ in Equation (3.23). As discussed in Section 2.4, this results from the fact that the mean square error difference between the widely linear model and linear model is zero when modelling circular sources. Therefore, the standard CLMS algorithm can be considered a special case of the ACLMS algorithm, suitable for the processing of proper complex signals.

[4]See Appendix B

Recently, a study on the ACLMS and dual channel LMS established the duality between the complex algorithm with its bivariate counterpart [90]. It was shown that for the same input and output, the two algorithms have the same dynamics. Analysis of the covariance matrices of the input signal of both algorithms shows that the eigenvalues of the augmented covariance matrix are twice the eigenvalues of the bivariate covariance matrix. Thus, based on the relation of the eigenvalues with the modes of convergence, it was concluded that for the same step-size and given the same final misadjustment, the ACLMS algorithm converges twice as fast as the dual channel LMS. This analysis is generalised in Appendix C where the duality of the block ACLMS (b-ACLMS) and block dual channel real-valued LMS (b-DCRLMS) is addressed.

## 3.3 Performance of the ACLMS algorithm

The advantage of the ACLMS algorithm over the standard CLMS is in the utilisation of the full second order statistical information available within the signal, achieved through WL modelling. For circular signals, where the pseudo-covariance is zero, it is anticipated that both algorithms will perform well, while ACLMS is expected to outperform the CLMS when applied to noncircular (improper) data. To demonstrate this, benchmark complex autoregressive AR(4) process and Ikeda map signal [91] were used, followed by real-world complex-valued wind signals.

The performance was assessed based on the prediction gain $R_p$ given by [92]

$$R_p = 10 \log_{10} \left( \frac{\sigma_x^2}{\hat{\sigma}_e^2} \right) \tag{3.26}$$

where $\sigma_x^2$ denotes the variance of the input signal $\mathbf{x}(k)$, whereas $\hat{\sigma}_e^2$ denotes the estimated variance of the forward prediction error $\{e(k)\}$, where $e(k) = d(k) - y(k)$ defined in Section 3.2.

### 3.3.1 Prediction of complex-valued autoregressive signal

In the first experiment, a synthesised stable and circular complex-valued AR(4) process is used, given by

$$x(k) = 1.79x(k-1) - 1.85x(k-2) + 1.27x(k-3) - 0.41x(k-4) + n(k) \tag{3.27}$$

where $n(k) = n_r(k) + jn_i(k)$ is a complex circular white Gaussian noise of zero mean and unit variance, where the real and imaginary parts are independent real white Gaussian sequences with $\sigma_n^2 = \sigma_{n_r}^2 + \sigma_{n_i}^2 = 1$.

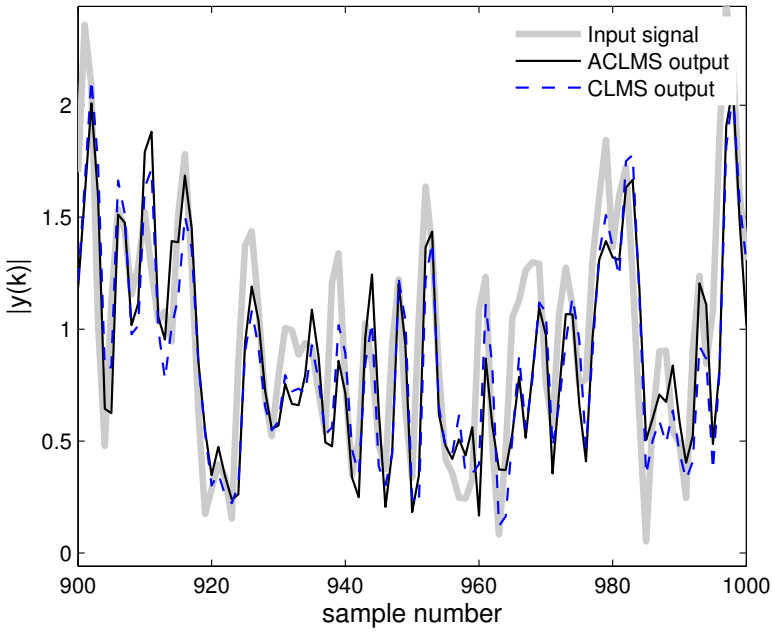



**Figure 3.1** The input and predicted signals obtained by using the CLMS (dash) and ACLMS (solid) algorithms.

The adaptive filter with $N = 10$ taps was trained using 1000 samples of the input $x(k)$, the step-size $\mu = 0.01$ was kept constant for both the algorithms. The obtained prediction gains were $R_{p,\text{CLMS}} = 3.22$ dB and $R_{p,\text{ACLMS}} = 3.99$ dB. Figure 3.1 demonstrates the convergence of the predicted signal to the original, which has been zoomed in for better clarity. The quantitative performances of both algorithms were adequate, with similar values of $R_p$. This was expected, since the AR(4) signal is circular and there is no information available in the pseudo-covariance matrix to facilitate the performance of the ACLMS. Table 3.1(a) summarises the performance results.

### 3.3.2 Prediction of complex-valued Ikeda map

In this simulation, prediction of an Ikeda map using the ACLMS and CLMS algorithms is investigated. The Ikeda map is expressed as

$$\begin{aligned} u(k) &= 1 + \alpha\Big(u(k-1)\cos\big(t(k-1)\big) - v(k)\sin\big(t(k-1)\big)\Big) \\ v(k) &= \alpha\Big(u(k-1)\sin\big(t(k-1)\big) + v(k-1)\cos\big(t(k-1)\big)\Big) \\ t(k-1) &= 0.4 - \frac{6}{1 + u^2(k-1) + v^2(k-1)}, \end{aligned} \tag{3.28}$$

where the parameter $\alpha$ affects the behaviour of the generated process, and has a typical value of $\alpha = 0.8$. Figure 3.2 demonstrates the Ikeda map used for this simulation, where the input $x(k) = u(k) + jv(k)$. Observe that the complex signal $x(k)$

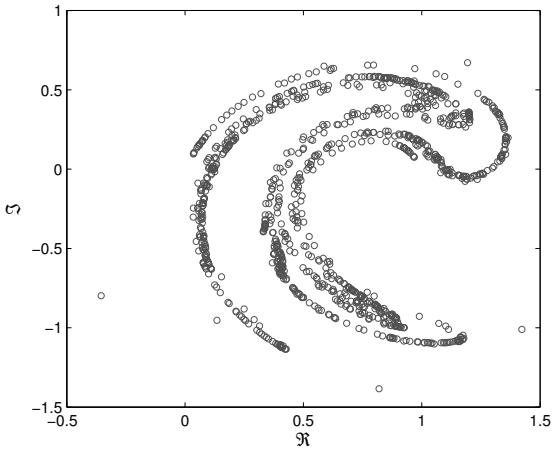



**Figure 3.2** Scatter plot of the Ikeda map given in Equation (3.28) with $\alpha = 0.8$.

**Table 3.1** Performance of the ACLMS and CLMS algorithms for prediction of benchmark and real-world signals

(a) Prediction gain (dB) for proper and improper benchmark signals

| | AR(4) | Ikeda map |
|---|---|---|
| CLMS | 3.22 | 2.13 |
| ACLMS | 3.99 | 3.51 |

(b) Prediction gain (dB) for the Low, Medium and High wind regions and according to window size

| | Wind region | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Low | | | Medium | | | High | | |
| $w_F$ | $r$ | ACLMS | CLMS | $r$ | ACLMS | CLMS | $r$ | ACLMS | CLMS |
| 1 | 0.52 | 2.53 | 1.85 | 0.28 | 5.32 | 4.43 | 0.65 | 6.68 | 6.35 |
| 2 | 0.53 | 2.77 | 2.02 | 0.29 | 5.76 | 4.74 | 0.67 | 8.62 | 7.80 |
| 10 | 0.57 | 3.76 | 2.72 | 0.32 | 7.37 | 5.96 | 0.75 | 13.51 | 11.80 |
| 20 | 0.59 | 4.51 | 3.03 | 0.35 | 8.32 | 6.76 | 0.80 | 15.07 | 13.00 |
| 60 | 0.64 | 5.21 | 2.88 | 0.43 | 9.63 | 7.69 | 0.86 | 16.53 | 14.30 |

is improper, with a noncircularity measure $r = 0.3418$, defined in (2.17). The linear and widely linear adaptive filters were trained with 1000 samples of $x(k)$ and with the step-size $\mu = 0.02$. The prediction gain obtained using the ACLMS algorithm was $R_{p,\text{ACLMS}} = 3.51$ dB, while the performance of the CLMS algorithm measured $R_{p,\text{CLMS}} = 2.13$ dB, demonstrating that the widely linear algorithm better modelled the complex improper signal. For comparison, these results are also presented in Table 3.1(a).

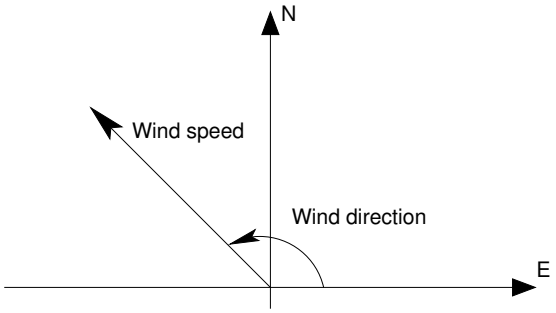



**Figure 3.3** Wind vector representation

### 3.3.3 Prediction of complex-valued wind using ACLMS

Wind field was measured using an ultrasonic anemometer[5] over a period of 24 hours sampled at 50Hz. A moving average filter was used to reduce the effects of high frequency noise; the signal was then resampled at 1Hz. The window size $w_F$ of the moving average filter varied according to

$$w_F = \{1, 2, 10, 20, 60\}, \tag{3.29}$$

where the window size is given in seconds.

The wind speed reading were taken in the north–south ($V_N$) and east–west ($V_E$) direction, which was used to create the complex wind signal $V = v\, e^{\jmath\varphi}$, as

$$v = \sqrt{V_E^2 + V_N^2}, \qquad \varphi = \arctan\left(\frac{V_N}{V_E}\right) \tag{3.30}$$

where $v$ is the wind speed, and $\varphi$ is the wind direction (see Figure 3.3).

Based on the modulus of the complex wind data dynamics, changes in the wind intensity were identified and labelled as regions *high*, *medium* and *low*, as shown on Figure 3.4. To investigate the advantage of WL modelling for such intermittent and improper complex data, 5000 samples were taken from each region to train CLMS and ACLMS adaptive predictors for one step ahead prediction, with simulations results shown in Figure 3.5 and summarised in Table 3.1(b).

As the wind signals become smoother and less noisy by increasing the window size, they also become more improper, as seen by the increase in the value of the noncircularity measure $r$. This is also reflected in Figure 3.5, where the performance of both algorithms improves with the increase in $w_F$, however, the ACLMS outperforms the standard CLMS in all wind regions due to its widely linear modelling of the wind signals.

[5]Recorded in an urban environment at the Institute of Industrial Science, University of Tokyo, Japan.

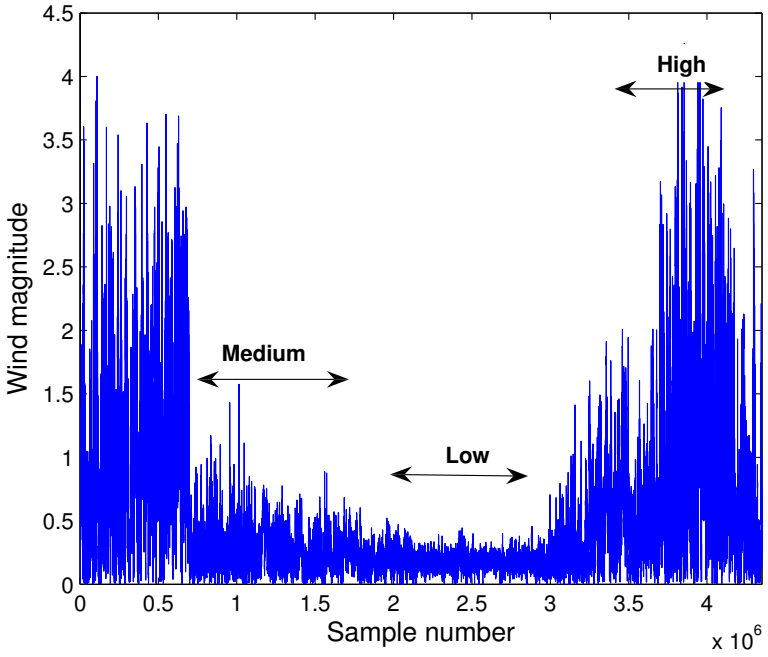



**Figure 3.4** Complex wind signal magnitude. Three wind speed regions have been identified as low, medium and high.

It is evident that the ACLMS algorithm has provided better predictions compared to the CLMS algorithm in all the three considered regions. The best prediction was obtained for the *high* region where the wind speed had strongest variations, giving a maximum prediction gain of 16.20 dB. Figure 3.6 shows the original and predicted signals from the *medium* region after 5000 iterations. It is seen that the ACLMS algorithm was able to track the dynamics of the input better and outperformed the CLMS algorithm.

Complex-valued wind is a noncircular signal, and thus the use of augmented statistics helped to extract the full second order statistical information available within the data. The results of the ACLMS prediction clearly indicate the benefits of using augmented statistics for noncircular complex-valued data, resulting in better prediction performance.

## 3.4 Hybrid filtering using linear and widely linear algorithms

A hybrid adaptive filter is designed as a combination of two (or more) independent adaptive filters, such that the combined (hybrid) filter has an improved performance over the two sub-filters [7]. The improvement in the output $y(k)$ of the hybrid filter in the prediction setting, shown in Figure 3.7, is achieved by considering the convex

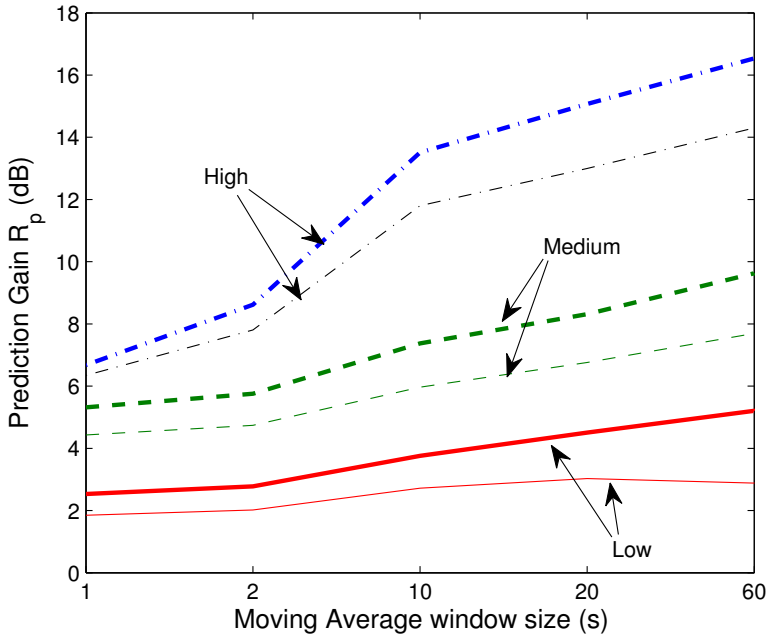



**Figure 3.5** Prediction gain of the ACLMS (thick lines) and CLMS (thin lines) algorithms in the *low* (solid), *medium* (dashed) and *high* (dot-dash) regions

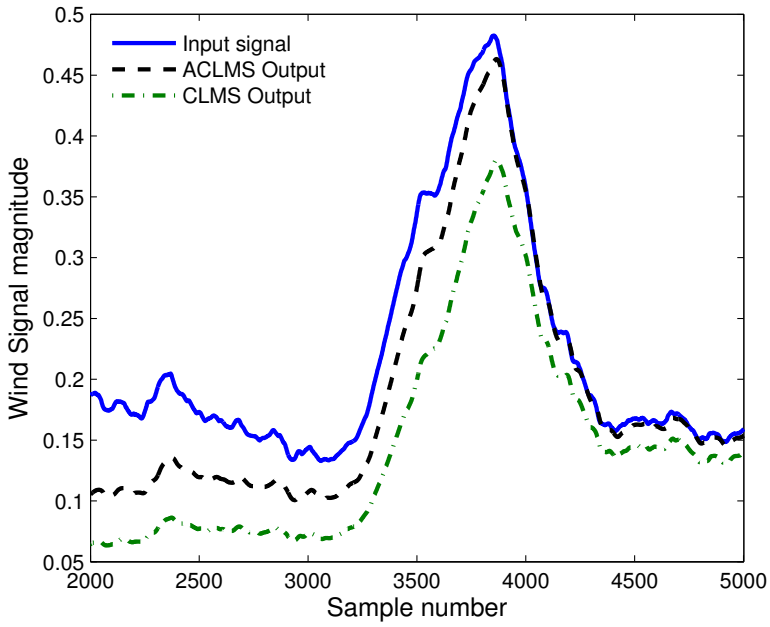



**Figure 3.6** Input and predicted signal of the *medium* region, comparing the performance of the ACLMS and CLMS after 5000 iterations (zoomed area).

combination of the filter outputs $y_1(k)$ and $y_2(k)$, given by

$$y(k) = \lambda(k)y_1(k) + \big(1 - \lambda(k)\big)y_2(k), \tag{3.31}$$

where $\lambda(k)$ is the mixing parameter. Intuitively, since a convex combination of two points $a$ and $b$ is defined as $\lambda a + (1-\lambda)b,\ \lambda \in [0,1]$ (shown in Figure 3.8), the value of $\lambda$ can be adapted to indicate which of the sub-filters is better suited to the nature of the input. This is contrast to a mixed-norm algorithm which uses a convex combination of suitable cost functions, rather than outputs [93].

For instance, consider the combination of adaptive sub-filters containing an algorithm with low steady-state error and one with fast initial convergence. The resultant hybrid filter inherits the initial fast convergence properties of the first sub-filter, and the stable steady-state performance of the second sub-filter. Such a combination using the LMS and Generalised Normalised Gradient Descent (GNGD) [4] algorithms was introduced in [94].

While hybrid filtering was originally conceived to enhance the performance of adaptive filters, it has recently found application in signal modality characterisation. This is achieved using a collaborative signal processing approach revealing changes in the nature of real-world data (degree of sparsity, or nonlinearity) and is very important in online applications [95]. By tracking the modality of a signal in real-time, it can be possible to, for example, provide prior knowledge to a blind algorithm. In such applications, the output $y(k)$ of the hybrid filter is not of interest, and the mixing parameter $\lambda$ is instead used to track the changes in the signal modality.

Characterisation of the nature of complex-valued signals has been addressed by considering the degree of nonlinearity and circularity of complex-valued signals using complex adaptive algorithms [96, 97, 98, 9]. The degree of nonlinearity is measured by utilising a hybrid filter with a pair of nonlinear and linear algorithms. Likewise, the signal circularity is indicated by using a pair of nonlinear adaptive algorithms with split- and fully-complex activation functions[6]. Thus, it is possible to track signals with high degree of correlation between the real and imaginary components (noncircular) and those with a smaller degree or lack of correlation (circular).

In this section, a hybrid filter consisting of a pair of linear and widely linear adaptive algorithms is considered. The optimisation algorithm for the mixing parameter $\lambda$ is derived, and benchmark simulations using autoregressive and Ikeda map are presented. It is shown that the hybrid filter has better performance than either sub-filter,

[6] A split-complex activation function $\Phi_S(z) \triangleq f(z_r) + \jmath f(z_i),\ f : \mathbb{R} \mapsto \mathbb{R}$, while a fully-complex activation function $\Phi_F(z) \triangleq g(z_r + \jmath z_i),\ g : \mathbb{C} \mapsto \mathbb{C}$ [65, 64, 99]. A split-complex activation function is not a true complex nonlinearity, and its use is only appropriate when the real and imaginary components are not correlated.

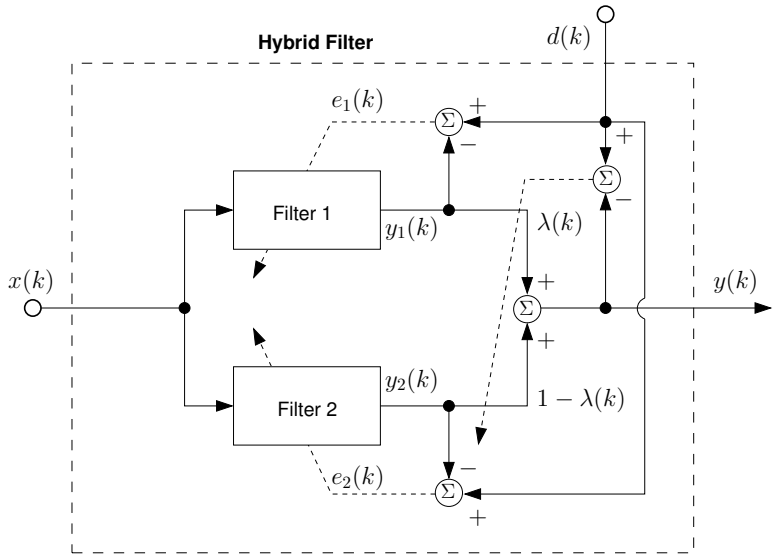



**Figure 3.7** Hybrid filter with input $x(k)$, consisting of two sub-filters.

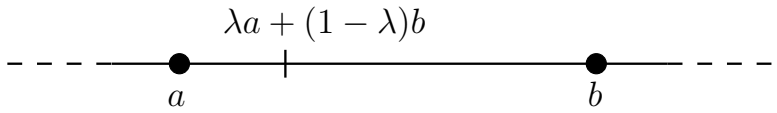



**Figure 3.8** Convex combination of two points $a$ and $b$.

while the mixing parameter can be interpreted as an indicator of the nature of the second-order circularity of the input signal.

### 3.4.1 Adaptation of the mixing parameter

The cost function for the hybrid filter is based on the output error power, given by

$$\mathcal{J}_H(\lambda) = E\{|e(k)|^2\} = E\{e(k)e^*(k)\} = E\{|d(k) - y(k)|^2\} \tag{3.32}$$

where $e(k)$ is the output error of the hybrid filter, $d(k)$ is the desired prediction signal and $y(k)$ is defined in (3.31). Recall that as the input $x(k)$ and desired signal $d(k)$ are complex-valued, the error $e(k)$ is also complex-valued, while the mixing parameter is real-valued.

The cost function (3.32) is minimised by updating the mixing parameter $\lambda$ via a gradient descent type algorithm such as the LMS. Thus, the update for $\lambda$ is written as

$$\lambda(k+1) = \lambda(k) - \mu_\lambda \nabla_\lambda \mathcal{J}_H(k), \tag{3.33}$$

where $\mu_\lambda$ is the step-size. Although $\lambda$ is real-valued, it is possible to utilise the $\mathbb{CR}$ calculus framework to derive the update as

$$\begin{aligned}\nabla_\lambda \mathcal{J}_H(k) &= \frac{\partial e(k)}{\partial \lambda(k)} e^*(k) + e(k) \frac{\partial e^*(k)}{\partial \lambda(k)} \\ &= -\big(y_1(k) - y_2(k)\big) e^*(k) - \big(y_1^*(k) - y_2^*(k)\big) e(k) \\ &= -2\Re\big\{\big(y_1(k) - y_2(k)\big)^* e(k)\big\} \end{aligned} \tag{3.34}$$

and (3.33) is then expressed as

$$\lambda(k+1) = \lambda(k) + \mu_\lambda 2\Re\big\{\big(y_1(k) - y_2(k)\big)^* e(k)\big\}. \tag{3.35}$$

### 3.4.2 Performance of the hybrid filter

The performance of the hybrid filter in the prediction of benchmark complex-valued signals is considered. For this task, the values of the sub-filter outputs $y_1(k)$ and $y_2(k)$ are respectively updated using the standard linear CLMS ('Filter 1') and widely linear ACLMS ('Filter 2') algorithms. The coefficients vectors of the two sub-filters are then updated using the algorithms given in (3.25) and (3.24). Given this configuration, both the CLMS and ACLMS algorithms are suitable for the processing of complex proper signals ($\lambda \to 0.5$), while only the ACLMS algorithm provides an optimal model for the prediction of improper signals ($\lambda \to 0$). The simulations below confirm this theoretical observation.

In the first simulation, 5000 samples of a complex proper AR(4) signal, given in (3.27), were processed using one step ahead prediction. Each sub-filter had $N = 10$ taps and the step-size of the CLMS and ACLMS algorithms were respectively set as $\mu_{\text{CLMS}} = \mu_{\text{ACLMS}} = 0.05$, while $\mu_\lambda = 2$ and the mixing parameter was initialised[7] as $\lambda(0) = 1$. The variation in $\lambda(k)$ is shown in Figure 3.9 and the performance, measured using the prediction gain $R_p$ defined in (3.26), is summarised in Table 3.2. It is seen that both algorithms had a similar performance with $R_{p,\text{CLMS}} = 4.61$ dB and $R_{p,\text{ACLMS}} = 4.27$ dB, while the combined performance, given by the hybrid filter, had a prediction gain of $R_{p,\text{Hybrid}} = 5.08$ dB.

With $\lambda(0) = 1$, the hybrid filter output was initially determined entirely by the CLMS algorithm. However, it is seen that the value first converged to $\lambda = 0.2$ after 200 samples which corresponds to the ACLMS algorithm, and then to an approximate value of $\lambda = 0.6$, that is, the output of either algorithm was acceptable and the output $y(k)$ was the average of the two. This also indicates that the AR(4) signal was proper.

In the second simulation, 5000 samples taken from an Ikeda map (see Equation (3.28)) are used to train the hybrid filter with $N = 10$ taps and step-sizes $\mu_{\text{CLMS}} = \mu_{\text{ACLMS}} =$

[7] It is also plausible to choose $\lambda(0) = 0.5$.

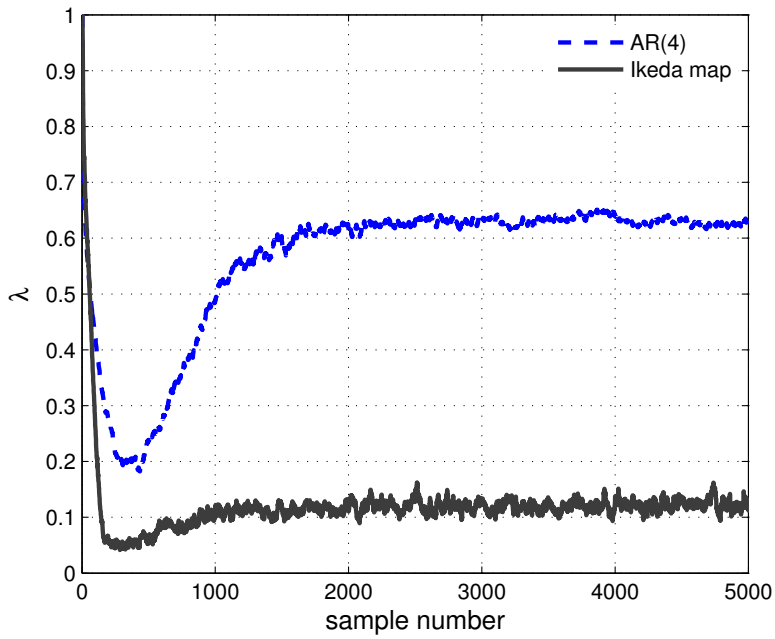



**Figure 3.9** Variation of the mixing parameter $\lambda(k)$ for AR(4) signal and Ikeda map.

**Table 3.2** Performance of the hybrid filter for prediction of AR(4) signal and Ikeda map, measured using the prediction gain (dB)

| | AR(4) | Ikeda map |
|---|---|---|
| CLMS | 4.6107 | 2.5287 |
| ACLMS | 4.2649 | 4.6047 |
| Hybrid | 5.0768 | 4.6220 |

$0.02$, $\mu_\lambda = 0.5$ and $\lambda(0) = 1$. The value of the predication gain of the algorithms, given in Table 3.2, shows that the hybrid filter had better performance that either sub-filter. The variation in $\lambda(k)$ also indicates the improper nature of the Ikeda map, where $\lambda(k)$ converged to an approximate value of 0.1 and the widely linear modelling of the ACLMS algorithm resulted in a smaller mean square error value.

## 3.5 Summary

By utilising recent advances in complex statistics, namely widely linear modelling, the standard complex LMS algorithm has been extended for the processing of the generality of complex-valued signals. In comparison to the CLMS algorithm which is based on a linear model, the introduced algorithm, the augmented complex LMS, is based on a widely linear model and is capable of capturing the complete second-order information available within the signal. It is seen that the CLMS algorithm is a special case of the ACLMS algorithm, suitable for processing proper signals. Derivation of the

algorithm has been provided based on the $\mathbb{CR}$ calculus framework, demonstrating its convenience for signal processing optimisation problems in the complex domain.

Simulations have illustrated the performance of the algorithm for the prediction of proper and improper complex signals, where the CLMS and ACLMS had similar performance for second-order circular data, while the ACLMS algorithm outperformed the standard CLMS algorithm on the prediction of improper data. Furthermore, the application of the algorithm in the prediction of real-world wind data has been demonstrated, where it outperformed the standard CLMS algorithm for the regimes with different wind magnitude intensities.

In the second part of this chapter, the application of linear and widely linear algorithms in the design of hybrid filters has been addressed. It has been shown that the convex combination of the linear CLMS and widely linear ACLMS algorithms result in an enhanced performance compared to either algorithm separately. The applicability of the hybrid filter for the online tracking of the signal circularity has also been discussed.

# Chapter 4

# Complex Blind Source Extraction from Noisy Mixtures using Second Order Statistics

## 4.1 Introduction

Blind source separation methods based on the temporal structure of the sources have been extensively investigated [100, 101]. Methods relying on higher-order statistics attempt to find a suitable demixing matrix such that in comparison to the observed mixtures, the estimated sources are as non-Gaussian as possible; this follows from the central limit theorem. In contrast, methods based on second-order statistics utilise the autocovariance to find a suitable demixing matrix, such that the current and delayed cross-covariances between the estimated sources are zero. The AMUSE algorithm [100] achieves this by considering a single time lag, while the SOBI algorithm [101] generalises this method by taking several time lags, and by minimising the off-diagonal entries of the covariance matrices.

In large mixtures where only a few sources are of interest, this concept can be used to devise blind source extraction algorithms. Blind extraction of sources based on the fundamental property of predictability was previously explored in the real domain in [102] and [10]. The predictability, described by second-order statistics, allows for the extraction of desired signals based on their temporal structure. This is achieved by assuming sources with temporal correlation, and thus modelling the extracted signals as an autoregressive (AR) model. Then, by minimising the squared prediction error at the output of an adaptive linear finite impulse response (FIR) predictor, sources pertaining to different degrees of predictability can be extracted. The uniqueness of the

temporal structure of the sources defines the success of the algorithm [10], in contrast to methods utilising the non-Gaussianity of the sources [12].

Blind adaptive and batch algorithms for prediction based on single and multiple time lags were described in [10]. In [35] a prediction-based BSE algorithm with an adaptive step-size was introduced, wherein comparison to a fixed step-size algorithm it resulted in better extraction performance for nonstationary signals and mixtures with a time-varying mixing matrix.

While sources with a unique temporal structure give different prediction errors, changes in the signal magnitude through mixing result in changes in their power levels and thus the values of the prediction errors can vary. The normalised MSPE was thus proposed as an alternative extraction criterion, in order to remove the ambiguity associated with the error power levels [103]. A modified version of this cost function was subsequently used to extract source signals from noisy mixtures based on their temporal features [104, 105]. This was achieved by removing the bias on the cost function due to additive noise.

The consideration of BSE algorithms based on the predictability of complex-valued sources is not a trivial extension of the results in the real domain. In considering the temporal structure of signals in $\mathbb{C}$, it is necessary to utilise a widely linear AR signal model so as to capture both the autocovariance and pseudo-autocovariance within the sources. Therefore a class of algorithms for the blind extraction of the generality of complex-valued sources from both noise-free and noisy mixtures is introduced. Algorithms based on prewhitened mixtures are also derived, and are shown to provide simpler solutions. By considering a general complex doubly white noise model, these algorithms are designed so as to successfully extract sources from noisy mixtures containing both circular and noncircular additive noise.

## 4.2 Complex BSE of noise-free and noisy mixtures

### 4.2.1 The normalised mean square prediction error

The observed mixture vector $\mathbf{x}(k) \in \mathbb{C}^N$ at time index $k$ is observed from the linear mixture of the complex sources $\mathbf{s}(k) \in \mathbb{C}^{N_s}$ as

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{v}(k) \tag{4.1}$$

where $\mathbf{A} \in \mathbb{C}^{N \times N_s}$ is the mixing matrix and $\mathbf{v}(k) \in \mathbb{C}^N$ denotes the additive noise. Here, it is assumed that the number of observations equals that of the sources; the next section shows how the overdetermined case can be used for the estimation of the second-order statistics of the noise $\mathbf{v}(k)$.

The sources $\mathbf{s}(k)$ are assumed to be stationary and spatially uncorrelated with unit variance and zero mean, with no assumptions regarding their second-order circularity. For a lag $\delta$, the covariance $\mathcal{C}_{\mathbf{ss}}$ and pseudo-covariance $\mathcal{P}_{\mathbf{ss}}$ can be formulated as

$$\begin{aligned}\mathcal{C}_{\mathbf{ss}}(\delta) &= E\{\mathbf{s}(k)\mathbf{s}^H(k-\delta)\} = \operatorname{diag}\big(\sigma_1^2(\delta),\ldots,\sigma_{N_s}^2(\delta)\big)\\ \mathcal{P}_{\mathbf{ss}}(\delta) &= E\{\mathbf{s}(k)\mathbf{s}^T(k-\delta)\} = \operatorname{diag}\big(\tau_1^2(\delta),\ldots,\tau_{N_s}^2(\delta)\big).\end{aligned} \tag{4.2}$$

Figure 4.1 shows the blind extraction architecture for complex signals, based on the minimisation of the MSPE. For the observation vector $\mathbf{x}(k)$, the extracted signal $y(k)$ is formed as

$$y(k) = \mathbf{w}^H\mathbf{x}(k). \tag{4.3}$$

The aim of the demixing process is to find a demixing vector $\mathbf{w}$ such that $\mathbf{u}^H = \mathbf{w}^H\mathbf{A} = [0,\ldots,u_n,\ldots,0]$ and thus extract only a single source with the smallest MSPE. The prediction error is given by

$$e(k) = y(k) - y_{WL}(k) \tag{4.4}$$

where $y_{WL}(k)$ denotes the output of the prediction filter and given by,

$$y_{WL}(k) = \mathbf{h}^T(k)\mathbf{y}(k) + \mathbf{g}^T(k)\mathbf{y}^*(k) \tag{4.5}$$

where $\mathbf{h}(k)$ and $\mathbf{g}(k)$ are the coefficient vectors of length $M$, and $\mathbf{y}(k)$ is a delayed version of the extracted signal given by $\mathbf{y}(k) = [y(k-1),\ldots,y(k-M)]^T$. The length $M$ of the filter affects the performance of the predictor, such that sources with rapid variations can be extracted using a short tap length, while smoother sources require a much larger tap length [35]. By updating the coefficient vectors adaptively, it is possible to introduce the largest relative difference in the MSPE as a criterion[1] for extraction [103].

The MSPE $E\{|e(k)|^2\}$ can then be calculated as

$$\begin{aligned}E\{|e(k)|^2\} &= E\{e(k)e^*(k)\}\\ &= \mathbf{w}^H\mathbf{A}\hat{\mathcal{C}}_{\mathbf{ss}}\mathbf{A}^H\mathbf{w} + \Re\{\mathbf{w}^H\mathbf{A}\hat{\mathcal{P}}_{\mathbf{ss}}\mathbf{A}^T\mathbf{w}^*\}\end{aligned} \tag{4.6}$$

where

$$\begin{aligned}\hat{\mathcal{C}}_{\mathbf{ss}} = \mathcal{C}_{\mathbf{ss}}(0) &- 2\Re\Big\{\sum_{m=1}^{M} h_m^*(k)\mathcal{C}_{\mathbf{ss}}(m)\Big\}\\ &+ \sum_{m,\ell=1}^{M}\big[h_m(k)h_\ell^*(k) + g_m^*(k)g_\ell(k)\big]\mathcal{C}_{\mathbf{ss}}(\ell-m)\end{aligned}$$

[1] It is also possible to assign fixed values to the coefficient vectors $\mathbf{h}$ and $\mathbf{g}$, however, this results in poorer performance.

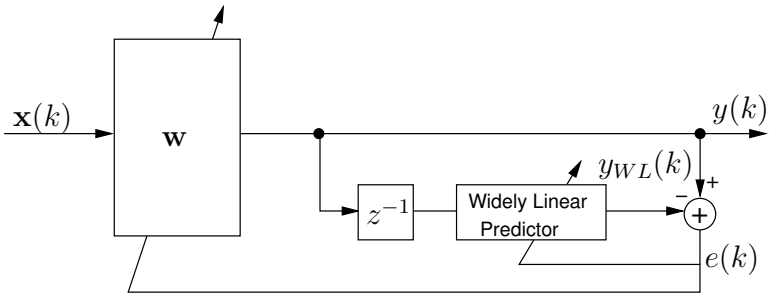



**Figure 4.1** The complex BSE algorithm using a widely linear predictor

and

$$\hat{\mathcal{P}}_{\mathbf{ss}} = -2\sum_{m=1}^{M} g_m^*(k)\mathcal{P}_{\mathbf{ss}}(m) + 2\sum_{m,\ell=1}^{M} h_m(k)g_\ell^*(k)\mathcal{P}_{\mathbf{ss}}(\ell - m).$$

The operator $\Re\{\cdot\}$ denotes the real part of a complex quantity. Observe that the prediction error is a function of both the covariance and pseudo-covariance of the sources, and, as the sources are assumed uncorrelated, $\hat{\mathcal{C}}_{\mathbf{ss}}$ and $\hat{\mathcal{P}}_{\mathbf{ss}}$ are diagonal matrices, with the value of the $n$th element corresponding to the error of the $n$th source, $s_n(k)$. Denoting this value by $e_n(k)$, the MSPE relating to $s_n(k)$ is given as

$$\begin{aligned} E\{|e_n(k)|^2\} = E\Big\{ & |s_n(k) - \mathbf{h}^T(k)\mathbf{s}_n(k)|^2 \\ & - 2\Re\{(s_n(k) - \mathbf{h}^T(k)\mathbf{s}_n(k))(\mathbf{g}^H(k)\mathbf{s}_n(k))\} \\ & + |\mathbf{g}^H(k)\mathbf{s}_n(k)|^2\Big\} \end{aligned} \tag{4.7}$$

where $\mathbf{s}_n(k) = [s_n(k-1), \ldots, s_n(k-M)]^T$. Due to the vanishing pseudo-covariance ($\mathcal{P}_{\mathbf{ss}} = \mathbf{0}$) of complex circular sources, the expressions for MSPE in (4.6) and that given in (4.7) for the $n$th source simplify, and are only functions of the covariance matrix. A complete derivation of the extraction based on MSPE as the extraction criterion is given in Appendix 4.A at the end of this chapter.

### 4.2.2 Noise-free complex BSE

#### 4.2.2.1 The cost function

The algorithms derived for complex BSE of noise-free mixtures are based on a cost function that minimises the normalised MSPE. As described in [103], the variation in the magnitude of source signals results in an ambiguity of the power levels and so algorithms based on the minimisation of the MSPE cannot effectively extract a source of interest. This can be seen by considering (4.6) and noticing that changes in the values of $\hat{\mathcal{C}}_{\mathbf{ss}}$ and $\hat{\mathcal{P}}_{\mathbf{ss}}$ can be effectively absorbed into the mixing matrix, thus enabling the

minimisation independent of the source power levels. This way, by using the MSPE, this ambiguity is removed as different signals exhibit different degrees of normalised predictability, despite the time-varying power levels.

Following [103], the normalised MSPE cost function is given by

$$\mathcal{J}_1(\mathbf{w}, \mathbf{h}, \mathbf{g}) = \frac{E\{|e(k)|^2\}}{E\{|y(k)|^2\}} \tag{4.8}$$

where $\mathcal{J}_1 \in \mathbb{R}$ and is a function of the demixing vector and the coefficient vectors. In the noise-free case, the alternating optimisation problem for the demixing vector can be expressed as

$$\mathbf{w}_{opt} = \arg \max_{||\mathbf{w}||_2=1} \mathcal{J}_1(\mathbf{w}, \mathbf{h}, \mathbf{g}) \tag{4.9}$$

where the norm of $\mathbf{w}$ is constrained to unity, and $\mathbf{u}_{opt}^H = \mathbf{w}_{opt}^H \mathbf{A}$ has only a single non-zero value with unit magnitude that corresponds to the source with the smallest normalised MSPE. This can be illustrated by observing the cost function (4.8), and its components (4.6),

$$\begin{aligned} E\{|y(k)|^2\} &= \mathbf{w}^H \mathcal{C}_{\mathbf{xx}}(0)\mathbf{w} \\ &= \mathbf{w}^H \mathbf{A}\mathcal{C}_{\mathbf{ss}}(0)\mathbf{A}^H\mathbf{w} + \mathbf{w}^H\mathcal{C}_{\mathbf{vv}}(0)\mathbf{w} \end{aligned} \tag{4.10}$$

and noting that the sources have unit variance and noise variance is zero. The cost function (4.8) then becomes

$$\mathcal{J}_1(\mathbf{w}, \mathbf{h}, \mathbf{g}) = \frac{\mathbf{u}^H \hat{\mathcal{C}}_{\mathbf{ss}}\mathbf{u} + \Re\{\mathbf{u}^H \hat{\mathcal{P}}_{\mathbf{ss}}\mathbf{u}^*\}}{\mathbf{u}^H\mathbf{u}} \tag{4.11}$$

Consider a new variable $\breve{\mathbf{u}} = \mathbf{u}/||\mathbf{u}||$, and the associated cost function

$$\mathcal{J}_1(\mathbf{w}, \mathbf{h}, \mathbf{g}) = \breve{\mathbf{u}}^H \hat{\mathcal{C}}_{\mathbf{ss}}\breve{\mathbf{u}} + \Re\{\breve{\mathbf{u}}^H \hat{\mathcal{P}}_{\mathbf{ss}}\breve{\mathbf{u}}^*\} \tag{4.12}$$

where $\breve{\mathbf{u}}^H\breve{\mathbf{u}} = 1$. With this constraint, the minimum of (4.12) is a vector $\breve{\mathbf{u}}_{opt}$ with a single non-zero element with arbitrary phase and unit magnitude, at a position corresponding to the smallest combination of the diagonal elements of $\hat{\mathcal{C}}_{\mathbf{ss}}$ and $\Re\{\hat{\mathcal{P}}_{\mathbf{ss}}\}$. In the case of circular sources, this argument simplifies, so that only the smallest diagonal element of $\hat{\mathcal{C}}_{\mathbf{ss}}$ is considered. This solution is similar for $\mathbf{u}_{opt}$, with only a single non-zero value. Likewise, the optimal value of the demixing vector can be recovered as $\mathbf{w}_{opt} = \mathbf{A}^{H\#}\mathbf{u}_{opt}$ where the symbol $(\cdot)^\#$ denotes the matrix pseudo-inverse. As described in [104], if a value $\mathbf{w}_{opt}$ exists such that $\mathbf{u}$ and hence $\breve{\mathbf{u}}$ respectively assume their optimal value $\mathbf{u}_{opt}$ and $\breve{\mathbf{u}}_{opt}$, then the cost function of (4.8) can be successfully minimised with respect to $\mathbf{w}$.

#### 4.2.2.2 Algorithms for the noise-free case

A gradient descent approach is used to update the values of the demixing vector $\mathbf{w}$ and the coefficient vectors $\mathbf{h}$ and $\mathbf{g}$. As mentioned earlier, the value of the demixing vector is constrained to unit norm, and is normalised after each update. The complex gradients are thus calculated as

$$\begin{aligned}
\nabla_{\mathbf{w}^*}\mathcal{J}_1 &= \Big[e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) - \frac{\sigma_e^2(k)}{\sigma_y^2(k)}y^*(k)\mathbf{x}(k)\Big]\frac{1}{\sigma_y^2(k)} \\
\nabla_{\mathbf{h}^*}\mathcal{J}_1 &= \frac{-1}{\sigma_y^2(k)}e(k)\mathbf{y}^*(k) \\
\nabla_{\mathbf{g}^*}\mathcal{J}_1 &= \frac{-1}{\sigma_y^2(k)}e(k)\mathbf{y}(k)
\end{aligned} \tag{4.13}$$

where

$$\begin{aligned}
\hat{\mathbf{x}}_h(k) &\triangleq \mathbf{x}(k) - \sum_{m=1}^{M} h_m(k)\mathbf{x}(k-m) \\
\hat{\mathbf{x}}_g(k) &\triangleq \sum_{m=1}^{M} g_m^*(k)\mathbf{x}(k-m)
\end{aligned}$$

and the MSPE $\sigma_e^2(k)$ and variance of the extracted signal $\sigma_y^2(k)$ are estimated by an online moving average relation [10]

$$\begin{aligned}
\sigma_e^2(k) &= \beta_e\sigma_e^2(k-1) + (1-\beta_e)|e(k)|^2 \\
\sigma_y^2(k) &= \beta_y\sigma_y^2(k-1) + (1-\beta_y)|y(k)|^2
\end{aligned} \tag{4.14}$$

with $\beta_e$ and $\beta_y$ the corresponding forgetting factors for the MSPE and signal power.

The update algorithm (P-cBSE) of the demixing vector $\mathbf{w}$ for the noise-free case and the filter coefficient updates are given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\Big[e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) - \frac{\sigma_e^2(k)}{\sigma_y^2(k)}y^*(k)\mathbf{x}(k)\Big]\frac{1}{\sigma_y^2(k)} \tag{4.15a}$$

$$\mathbf{w}(k+1) \leftarrow \frac{\mathbf{w}(k+1)}{||\mathbf{w}(k+1)||_2}$$

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h\frac{1}{\sigma_y^2(k)}e(k)\mathbf{y}^*(k), \tag{4.15b}$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g\frac{1}{\sigma_y^2(k)}e(k)\mathbf{y}(k). \tag{4.15c}$$

From the expressions for gradients $\nabla_{\mathbf{h}^*}\mathcal{J}_1$ and $\nabla_{\mathbf{g}^*}\mathcal{J}_1$ in (4.13), the update equations (4.15b) and (4.15c) can be combined to form a normalised ACLMS type adaptation [106]. Recall that for circular sources, the pseudo-covariance matrix vanishes; thus a standard complex linear predictor (say based on CLMS) can be used. However, this case is already incorporated within the WL predictor as e.g. the conjugate part of the ACLMS

weight vector vanishes for circular data ($\mathbf{g} = \mathbf{0}$), demonstrating the flexibility of the proposed approach.

One way to remove the effects of source power ambiguity is to prewhiten the observation vector $\mathbf{x}(k)$, so as to make power levels of the output (extracted) signals constant. This also helps to orthogonalise an ill-conditioned mixing matrix, however, performing prewhitening for an online algorithm is not convenient. Denoting the prewhitening matrix $\mathbf{V} = \mathbf{D}^{-1/2}\mathbf{E}$, where $\mathbf{D}$ a diagonal matrix containing the eigenvalues of $\mathcal{C}_{\mathbf{xx}}(0)$ and $\mathbf{E}$ an orthogonal matrix whose columns are the eigenvectors of $\mathcal{C}_{\mathbf{xx}}(0)$, the covariance matrix $\mathcal{C}_{\breve{\mathbf{x}}\breve{\mathbf{x}}}(0) = \mathbf{V}\mathcal{C}_{\mathbf{xx}}\mathbf{V}^H = \mathbf{I}$; the symbol $\breve{\mathbf{x}}(k)$ denotes a prewhitened observation vector. From (4.10) and the constraint on the norm of $\mathbf{w}$, $E\{|y(k)|^2\} = \mathbf{w}^H\mathbf{w} = 1$, the cost function in (4.8) can be simplified to

$$\begin{aligned}\nabla_{\mathbf{w}^*}\mathcal{J}_1 &= \big[e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k)\big]\\ \nabla_{\mathbf{h}^*}\mathcal{J}_1 &= -e(k)\mathbf{y}^*(k)\\ \nabla_{\mathbf{g}^*}\mathcal{J}_1 &= -e(k)\mathbf{y}(k).\end{aligned} \tag{4.16}$$

Thus, the resulting coefficient updates

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\big[e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k)\big] \tag{4.17a}$$

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h e(k)\mathbf{y}^*(k) \tag{4.17b}$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_h e(k)\mathbf{y}(k) \tag{4.17c}$$

are simpler than those in (4.15a)–(4.15c) and the coefficients of the WL predictor in (4.17b)–(4.17c) are updated using the ACLMS algorithm.

### 4.2.3 Noisy complex BSE

#### 4.2.3.1 The cost function

The algorithms described above do not account for the effect of the additive noise $\mathbf{v}(k)$ and thus underperform for the extraction of sources from noisy mixtures. By modifying the cost function, it is possible to derive a new class of algorithms for the extraction of complex sources from noisy mixtures. The modified cost function described in [104] which employs a normalised MSPE type cost function, can be used to remove the effect of noise from the MSPE and output variance.

Taking a closer look at the covariance and pseudo-covariance of the observation vector with additive noise,

$$\begin{aligned}\mathcal{C}_{\mathbf{xx}}(\delta) &= \mathbf{A}\mathcal{C}_{\mathbf{ss}}(\delta)\mathbf{A}^H + \mathcal{C}_{\mathbf{vv}}(\delta)\\ \mathcal{P}_{\mathbf{xx}}(\delta) &= \mathbf{A}\mathcal{P}_{\mathbf{ss}}(\delta)\mathbf{A}^T + \mathcal{P}_{\mathbf{vv}}(\delta)\end{aligned} \tag{4.18}$$

it is noted that the MSPE can be divided into two parts $e_s^2$ and $e_v^2$, where the first term is related to the MSPE relevant to the sources (4.6) and the second term pertains to that of the noise, and so $E\{|e(k)|^2\} = e_s^2 + e_v^2$. The expression for $e_v^2$ is derived in Appendix 4.A at the end of this chapter. The cost function for the noisy BSE thus becomes

$$\begin{aligned}\mathcal{J}_2(\mathbf{w},\mathbf{h},\mathbf{g}) &= \frac{E\{|e(k)|^2\} - e_v^2}{E\{|y(k)|^2\} - \mathbf{w}^H \mathcal{C}_{\mathbf{vv}}(0)\mathbf{w}} \\ &= \frac{E\{|e(k)|^2\} - c_1\sigma_v^2 - \Re\{c_2\tau_v^2 \mathbf{w}^H\mathbf{w}^*\}}{E\{|y(k)|^2\} - \sigma_v^2}\end{aligned} \tag{4.19}$$

where the signal variance is given in (4.10). The existence of a solution to the minimisation of the cost function can be addressed similarly to the noiseless case. By removing the effect of noise from $\mathcal{J}_2$, the resultant cost function is expanded exactly as in (4.11) and a similar argument can be used for the analysis.

#### 4.2.3.2 Algorithms for the noisy case

The cost function (4.19) is minimised using steepest descent and the coefficient vectors $\mathbf{w}, \mathbf{h}$ and $\mathbf{g}$ are updated via an online algorithm, similarly to the noise-free case. The corresponding gradients are calculated as

$$\begin{aligned}\nabla_{\mathbf{w}^*}\mathcal{J}_2 &= \frac{1}{\sigma_y^2(k) - \sigma_v^2}\Bigg[e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) - \Re\{c_2\tau_v^2\}\mathbf{w}^*(k) \\ &\quad - \frac{\sigma_e^2(k) - c_1\sigma_v^2 - \Re\{c_2\tau_v^2\mathbf{w}^H(k)\mathbf{w}^*(k)\}}{\sigma_y^2(k) - \sigma_v^2} y^*(k)\mathbf{x}(k)\Bigg] \\ \nabla_{\mathbf{h}^*}\mathcal{J}_2 &= \frac{-1}{\sigma_y^2(k) - \sigma_v^2}\big(e(k)\mathbf{y}^*(k) + \sigma_v^2\mathbf{h}(k)\big) \\ \nabla_{\mathbf{g}^*}\mathcal{J}_2 &= \frac{-1}{\sigma_y^2(k) - \sigma_v^2}\big(e(k)\mathbf{y}(k) + \sigma_v^2\mathbf{g}(k) + \Re\{\mathbf{w}^H\mathbf{w}^*\tau_v^2\}\mathbf{h}(k)\big)\end{aligned} \tag{4.20}$$

where

$$\begin{aligned}c_1 &= 1 + \mathbf{h}^H(k)\mathbf{h}(k) + \mathbf{g}^H(k)\mathbf{g}(k) \\ c_2 &= 2\mathbf{g}^H(k)\mathbf{h}(k)\end{aligned}$$

and the demixing vector $\mathbf{w}(k+1)$ is normalised after each update, so that

$$\mathbf{w}(k+1) \leftarrow \frac{\mathbf{w}(k+1)}{||\mathbf{w}(k+1)||_2}.$$

It is apparent from (4.20) that the estimation of the noise variance and pseudo-variance is necessary for the operation of this BSE method as discussed in the next section. Finally note that for a circular white additive noise, the pseudo-variance $\tau_v^2$ is zero and thus the terms related to the pseudo-covariance in (4.20) vanish.

The coefficient updates for BSE of noisy mixtures are given by

$$\begin{aligned}\mathbf{w}(k+1) = \mathbf{w}(k) &\\ - \frac{\mu}{\sigma_y^2(k) - \sigma_v^2} &\Bigg[ e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) - \Re\{c_2\tau_v^2\}\mathbf{w}^*(k) \\ - \frac{\sigma_e^2(k) - c_1\sigma_v^2 - \Re\{c_2\tau_v^2\mathbf{w}^H(k)\mathbf{w}^*(k)\}}{\sigma_y^2(k) - \sigma_v^2} &y^*(k)\mathbf{x}(k)\Bigg] \qquad (4.21a)\end{aligned}$$

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu_h \frac{1}{\sigma_y^2(k) - \sigma_v^2}\big(e(k)\mathbf{y}^*(k) + \sigma_v^2\mathbf{h}(k)\big) \qquad (4.21b)$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g \frac{1}{\sigma_y^2(k) - \sigma_v^2}\big(e(k)\mathbf{y}(k) + \sigma_v^2\mathbf{g}(k) + \Re\{\mathbf{w}^H\mathbf{w}^*\tau_v^2\}\mathbf{h}(k)\big) \qquad (4.21c)$$

The case of the prewhitened observation vector $\breve{\mathbf{x}}$ is next considered, where the variance of the extracted signal is constant and the resulting algorithms are somewhat simpler. The prewhitened covariance and pseudo-covariance are now given as

$$\begin{aligned}\mathcal{C}_{\breve{\mathbf{x}}\breve{\mathbf{x}}}(0) &= \breve{\mathbf{A}}\mathcal{C}_{\mathbf{ss}}(0)\breve{\mathbf{A}}^H + \breve{\mathcal{C}}_{\mathbf{vv}}(0) = \mathbf{I} \\ \mathcal{P}_{\breve{\mathbf{x}}\breve{\mathbf{x}}}(0) &= \breve{\mathbf{A}}\mathcal{P}_{\mathbf{ss}}(0)\breve{\mathbf{A}}^T + \breve{\mathcal{P}}_{\mathbf{vv}}(0)\end{aligned} \qquad (4.22)$$

with $\breve{\mathbf{A}} = \mathbf{VA}$, $\breve{\mathcal{C}}_{\mathbf{vv}}(0) = \mathbf{V}\mathcal{C}_{\mathbf{vv}}(0)\mathbf{V}^H$ and $\breve{\mathcal{P}}_{\mathbf{vv}}(0) = \mathbf{V}\mathcal{P}_{\mathbf{vv}}(0)\mathbf{V}^T$. It is possible to use a strong uncorrelating transform (SUT) [69] to whiten the covariance matrix and diagonalise the pseudo-covariance such that $\mathcal{P}_{\breve{\mathbf{x}}\breve{\mathbf{x}}}(0) = \mathbf{\Lambda}$ contains non-negative real values. In the case of circular signals, the SUT simplifies to a standard whitening operation.

This way, the term $e_v^2$ can be expanded as

$$e_v^2 = \mathbf{w}^H\hat{\breve{\mathcal{C}}}_{\mathbf{vv}}\mathbf{w} + \Re\{\mathbf{w}^H\hat{\breve{\mathcal{P}}}_{\mathbf{vv}}\mathbf{w}^*\} \qquad (4.23)$$

and the variance of the extracted signal

$$\begin{aligned}E\{|y(k)|^2\} &= \mathbf{w}^H\mathcal{C}_{\breve{\mathbf{x}}\breve{\mathbf{x}}}(0)\mathbf{w} \\ &= \mathbf{w}^H\breve{\mathbf{A}}\mathcal{C}_{\mathbf{ss}}(0)\breve{\mathbf{A}}^H\mathbf{w} + \mathbf{w}^H\breve{\mathcal{C}}_{\mathbf{vv}}(0)\mathbf{w} \\ &= \mathbf{w}^H\mathbf{w} = 1.\end{aligned} \qquad (4.24)$$

The cost function (4.19) can thus be rewritten as

$$\begin{aligned}\mathcal{J}_2 &= \frac{E\{|e(k)|^2\} - \mathbf{w}^H\hat{\breve{\mathcal{C}}}_{\mathbf{vv}}\mathbf{w} + \Re\{\mathbf{w}^H\hat{\breve{\mathcal{P}}}_{\mathbf{vv}}\mathbf{w}^*\}}{E\{|y(k)|^2\} - \mathbf{w}^H\breve{\mathcal{C}}_{\mathbf{vv}}(0)\mathbf{w}} \\ &= \frac{E\{|e(k)|^2\} - c_1\mathbf{w}^H\hat{\breve{\mathcal{C}}}_{\mathbf{vv}}(0)\mathbf{w} + \Re\{c_2\mathbf{w}^H\hat{\breve{\mathcal{P}}}_{\mathbf{vv}}(0)\mathbf{w}^*\}}{\mathbf{w}^H\mathbf{w} - \mathbf{w}^H\breve{\mathcal{C}}_{\mathbf{vv}}(0)\mathbf{w}} \\ &= E\{|e(k)|^2\} - c_1E\{|y(k)|^2\} - \Re\{c_2\tau_v^2\mathbf{w}^H\mathbf{V}\mathbf{V}^T\mathbf{w}^*\} + 1\end{aligned} \qquad (4.25)$$

where the demixing vector is normalised as

$$\mathbf{w}(k+1) \leftarrow \frac{\mathbf{w}(k+1)}{\sqrt{\mathbf{w}^H(k+1)[\mathbf{I} - \breve{\mathcal{C}}_{\mathbf{vv}}(0)]\mathbf{w}(k+1)}}. \tag{4.26}$$

This normalisation allows for the denominator in (4.25) to become unity. The gradients within the updates of the online algorithms for noisy BSE can be calculated as

$$\begin{aligned}
\nabla_{\mathbf{w}^*}\mathcal{J}_2 &= e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) - \Re\{c_2\tau_v^2\}\mathbf{V}^T\mathbf{V}\mathbf{w}^*(k) - c_1 y^*(k)\mathbf{x}(k) \\
\nabla_{\mathbf{h}^*}\mathcal{J}_2 &= -e(k)\mathbf{y}^*(k) \\
\nabla_{\mathbf{g}^*}\mathcal{J}_2 &= -\big(e(k)\mathbf{y}(k) + \Re\{\tau_v^2\mathbf{w}^H\mathbf{V}\mathbf{V}^T\mathbf{w}^*\}\mathbf{h}(k)\big)
\end{aligned} \tag{4.27}$$

to form the final online update for the BSE of prewhitened noisy mixtures, with the update algorithm for the demixing vector and the update equations for the filter coefficient vectors given by

$$\begin{aligned}
\mathbf{w}(k+1) &= \mathbf{w}(k) - \mu e^*(k)\hat{\mathbf{x}}_h(k) - e(k)\hat{\mathbf{x}}_g(k) \\
&\quad - \Re\{c_2\tau_v^2\}\mathbf{V}^T\mathbf{V}\mathbf{w}^*(k) - c_1 y^*(k)\mathbf{x}(k) && \text{(4.28a)} \\
\mathbf{h}(k+1) &= \mathbf{h}(k) + \mu_h e(k)\mathbf{y}^*(k) && \text{(4.28b)} \\
\mathbf{g}(k+1) &= \mathbf{g}(k) + \mu_g\big(e(k)\mathbf{y}(k) + \Re\{\tau_v^2\mathbf{w}^H\mathbf{V}\mathbf{V}^T\mathbf{w}^*\}\mathbf{h}(k)\big) && \text{(4.28c)}
\end{aligned}$$

### 4.2.4 Remark on the estimation of noise variance and pseudo-variance

The adaptive algorithms derived in the previous section require estimation of the noise variance and pseudo-variance for their operation. As mentioned in Chapter 2, the noise is considered to have a constant variance $\sigma_v^2$ and pseudo-variance $\tau_v^2$ so that $\mathcal{C}_{\mathbf{vv}} = \sigma_v^2\mathbf{I}$ and $\mathcal{P}_{\mathbf{vv}} = \tau_v^2\mathbf{I}$. Furthermore, two variants of complex noise were discussed: circular white noise and doubly white noise. One possible method for the estimation of the variance of circular white noise is by means of a subspace method [1] and can be intuitively extended for the estimation of the pseudo-variance of doubly white noise, as detailed below.

Consider the number of observations larger than that of the sources ($N > N_s$); it is then possible to estimate the noise variance and pseudo-variance, based on

$$\begin{aligned}
\mathcal{C}_{\mathbf{xx}} &= \mathbf{A}\mathcal{C}_{\mathbf{ss}}\mathbf{A}^H + \mathcal{C}_{\mathbf{vv}} = \mathbf{\Theta} + \sigma_v^2\mathbf{I} \\
\mathcal{P}_{\mathbf{xx}} &= \mathbf{A}\mathcal{P}_{\mathbf{ss}}\mathbf{A}^T + \mathcal{P}_{\mathbf{vv}} = \mathbf{\Xi} + \tau_v^2\mathbf{I}.
\end{aligned} \tag{4.29}$$

where $\mathbf{\Theta} = \mathbf{A}\mathcal{C}_{\mathbf{ss}}\mathbf{A}^H$ and $\mathbf{\Xi} = \mathbf{A}\mathcal{P}_{\mathbf{ss}}\mathbf{A}^T$. For both cases, by assuming that the matrix $\mathbf{A}$ is of full column rank, $\text{Rank}(\mathbf{A}) = N_s$, and that $\mathbf{s}$ is non-singular, then $\text{Rank}(\mathbf{\Theta}) = \text{Rank}(\mathbf{\Xi}) = N_s$ and so the $(N - N_s)$ smallest eigenvalues of $\mathbf{\Theta}$ and $\mathbf{\Xi}$ are zero. Hence, the $(N - N_s)$ smallest eigenvalues of $\mathcal{C}_{\mathbf{xx}}$ and $\mathcal{P}_{\mathbf{xx}}$ are respectively equal to $\sigma_v^2$ and $\tau_v^2$.

**Table 4.1** Source properties for noise-free extraction experiments

| Source | distribution | circ. measure ($r$) | norm. MSPE |
|---|---|---|---|
| $s_1(k)$ | Super-Gaussian | 0.03 | 1.45 |
| $s_2(k)$ | Sub-Gaussian | 1.00 | 1.69 |
| $s_3(k)$ | Super-Gaussian | 1.00 | **1.34** |

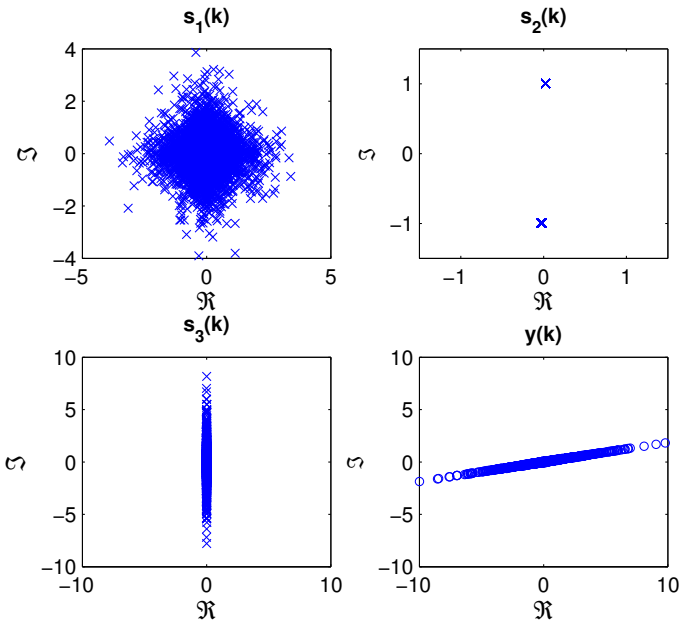



**Figure 4.2** Scatter plots of the complex sources $s_1(k), s_2(k)$ and $s_3(k)$ whose properties are described in Table 4.1. The scatter plot of the extracted signal $y(k)$, corresponding to the source $s_3(k)$, is given in the bottom right plot.

## 4.3 Simulations and discussion

### 4.3.1 Performance analysis for synthetic data

The performances of the proposed algorithms were analysed using sources with different degrees of noncircularity and for different probability distributions, and in various simulation settings comprising both noise-free and noisy mixtures.

Performances of the algorithms were measured using the Performance Index (PI) [10], which for $\mathbf{u} = \mathbf{A}^H\mathbf{w} = [u_1, \ldots, u_M]$ is given as

$$PI = 10\log_{10}\left(\frac{1}{M}\left(\sum_{i=1}^{M}\frac{|u_i|^2}{\max\{|u_1|^2,\ldots,|u_M|^2\}} - 1\right)\right). \tag{4.30}$$

and indicates the closeness of $\mathbf{u}$ to having only a single non-zero element. The values of the step-sizes $\mu, \mu_h$ and $\mu_g$ were set empirically, the mixing matrix $\mathbf{A}$ was generated

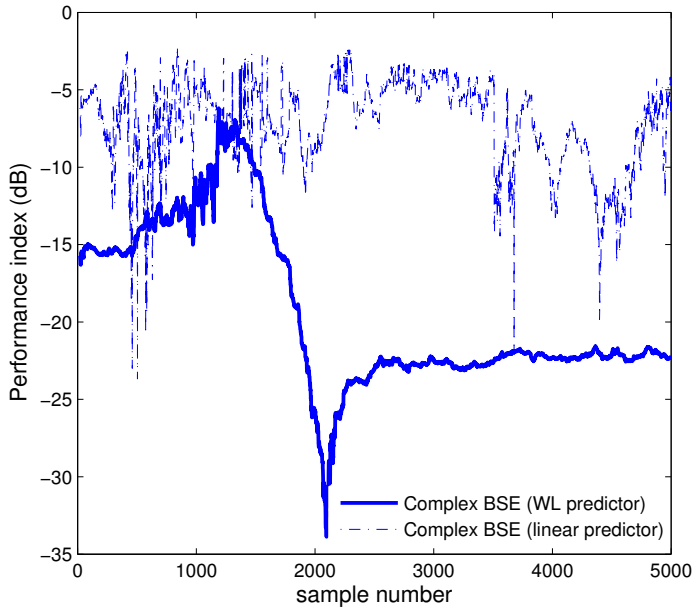



**Figure 4.3** Learning curves for extraction of complex sources from noise-free mixtures using algorithm (4.15a)–(4.15c), based on WL predictor (solid line) and linear predictor (broken line).

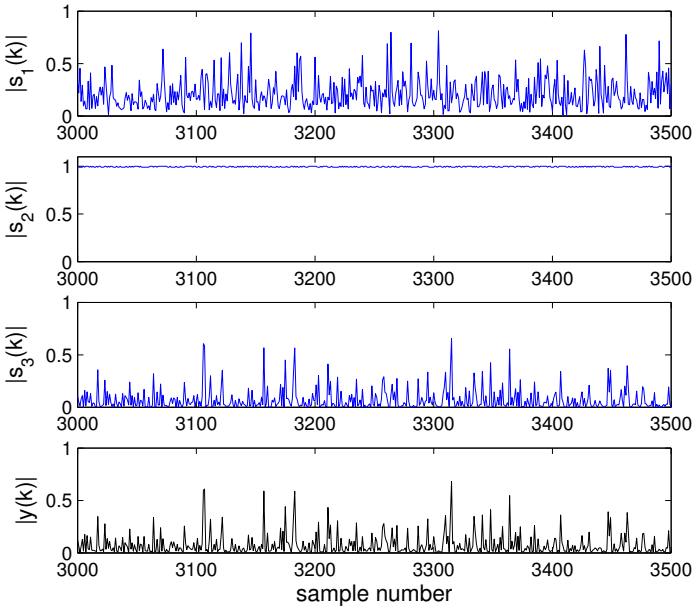



**Figure 4.4** Normalised absolute values of the sources $s_1(k), s_2(k)$ and $s_3(k)$, whose properties are described in Table 4.1. The extracted source $y(k)$, shown in the bottom plot, is obtained from a noise-free mixture using algorithm (4.15a)–(4.15c).

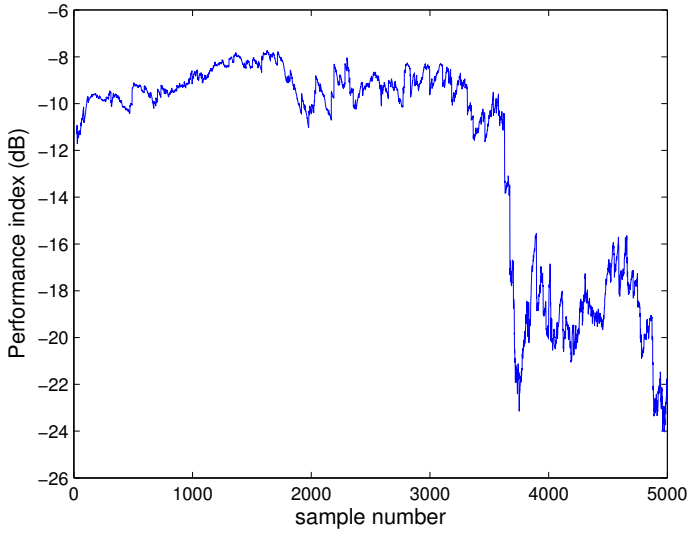



**Figure 4.5** Extraction of complex sources from a noise-free prewhitened mixture using algorithm (4.17a)–(4.17c), based on a WL predictor.

randomly and in all experiments the forgetting factors $\beta_e = \beta_y = 0.975$. The additive noise $\mathbf{v}(k)$ had a Gaussian distribution in two variants of proper white ($r = 0$) and doubly white improper ($r = 0.93$). Its variance and pseudo-variance were estimated using the subspace method (4.29).

In the first set of experiments, $N_s = 3$ sources with 5000 samples were generated (Figure 4.2) and subsequently mixed to form a noise-free mixture. The sources were mixed using a $3 \times 3$ mixing matrix and the resultant observation vector was input to the adaptive algorithm of (4.15a) with a step-size of $\mu = 5 \times 10^{-3}$ chosen empirically. The coefficients of the WL predictor were updated using (4.15b) and (4.15c) with filter length $M = 20$ and $\mu_h = \mu_g = 10^{-5}$. The resultant learning curve shown in Figure 4.3 was averaged over 100 trials with the initial demixng vector chosen randomly. The source properties are shown in Table 4.1, which also include the circularity measure and the value of the normalised MSPE corresponding to the source (4.7).

The algorithm was able to extract the source with the smallest normalised MSPE, with the PI reaching a value of -22 dB at steady-state after 2000 samples (Figure 4.3). The normalised absolute values of the sources $s_i(k), i = 1, 2, 3$ and $y(k)$ are shown in Figure 4.4, illustrating that the desired source $s_3(k)$, with the smallest MSPE, was extracted successfully. Figure 4.2 shows the scatter plots of the three sources and the extracted signal. The scatter plot of the extracted signal $y(k)$ is a scaled and rotated version of $s_3(k)$ due to the ambiguity problem of BSS.

Next, for the same setting, the resulting mixture was prewhitened and extraction was performed using the algorithm (4.17a)–(4.17c). The resulting learning curve shown in

Figure 4.5 exhibits slow convergence with an average steady-state value of -19 dB after 4000 samples. The step-size parameters were set to $\mu = 5 \times 10^{-3}$ and $\mu_h = \mu_g = 10^{-4}$.

For comparison, the performance of the algorithm (4.15a)–(4.15c) is demonstrated, which uses a standard linear predictor for the extraction of the complex sources. The extraction of the noncircular sources (whose properties are given in Table 4.1) is performed using the same mixing matrix as in the previous experiments. This is straightforward by assuming the conjugate part of the coefficient vector of the WL predictor in (4.15b)–(4.15c) $\mathbf{g} = \mathbf{0}$ and updating only the coefficient vector $\mathbf{h}$, as shown in Section 4.2. As shown in the analysis, the linear predictor is not suited for modelling the full second-order information and did not provide satisfactory extraction (as seen from Figure 4.3), and reaching an average PI of only -6.5 dB as opposed to -22 dB for the WL case using the ACLMS.

In the next set of experiments, the performances of the proposed algorithms for the noisy case were investigated. A new set of three complex source signals were generated with 5000 samples, their properties are described in Table 4.2, and the $4 \times 3$ mixing matrix was generated randomly. Circular white Gaussian noise with variance $\sigma_v^2 = 0.1$ was added to the mixture to create the observed noisy mixture. The algorithm given in (4.21a) was used to minimise the cost function and extract the source with the smallest normalised MSPE. The values of the widely linear predictor coefficient vectors were updated via (4.21b) and (4.21c), with filter length $M = 20$ and step-size values $\mu = 5 \times 10^{-3}$ and $\mu_h = \mu_g = 10^{-3}$. The learning curve in Figure 4.6 demonstrates the performance of the algorithm, reaching steady-state after 2000 samples and with an average PI of -30 dB, indicating a successful extraction of the source $s_3(k)$.

The effect of doubly white noncircular Gaussian noise with circularity measure $\xi = 5$ is investigated, while keeping the source and mixing matrix values unchanged. The noise variance was $\sigma_v^2 = 0.1$ and the estimated pseudo-variance of the noise was $\tau_v^2 = -0.0894 - \jmath 0.0002$ (using the subspace method (4.29)). The learning curve in Figure 4.7 indicating the algorithm (4.21a)–(4.21c) converging to a solution in around 1500 samples and with an average steady-state value of -21 dB, for the step-sizes $\mu = 5 \times 10^{-3}$ and $\mu_h = \mu_g = 10^{-5}$. For comparison, the learning curve using the algorithm (4.15a)–(4.15c) is also included illustrating the inability to extract the desired source from the noisy noncircular mixture. Finally, the input was prewhitened and sources extracted based on (4.28a) for the update of the demixing vector, and using (4.28b) and (4.28c) for the update of the coefficient vectors, to produce the learning curve in Figure 4.8. In this scenario, the step-size parameters were chosen as $\mu = 10^{-4}$ and $\mu_h = \mu_g = 10^{-6}$, leading to slow convergence.

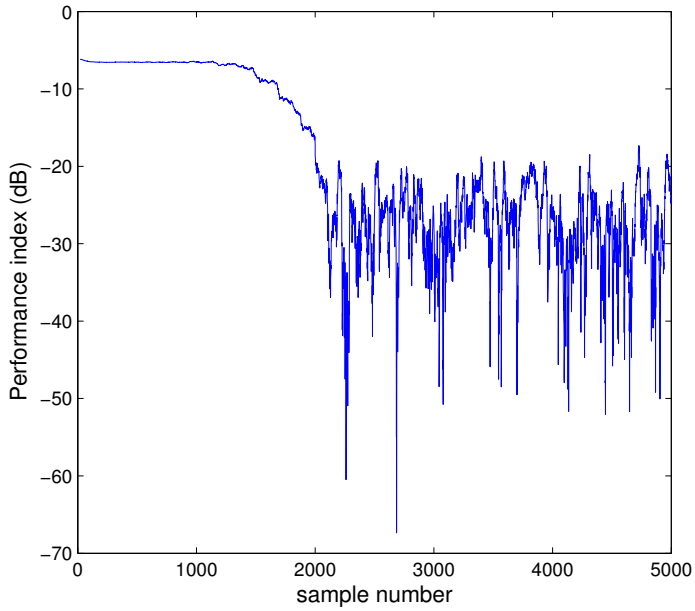



**Figure 4.6** Extraction of complex sources from a noisy mixture with additive circular white Gaussian noise, using algorithm (4.21a)–(4.21c) with a WL predictor.

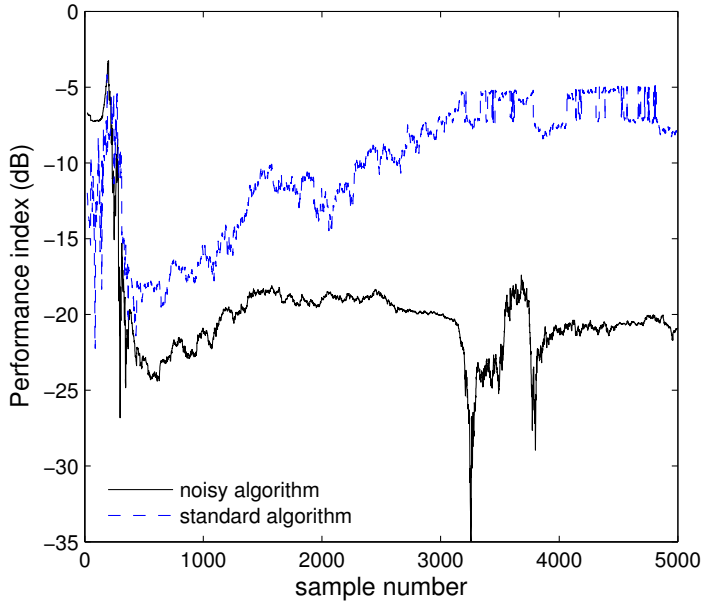



**Figure 4.7** Extraction of complex sources from a noisy mixture with additive doubly white noncircular Gaussian noise using algorithm (4.21a)–(4.21c) (solid line) and algorithm (4.15a)–(4.15c) (broken line), with a WL predictor.

**Table 4.2** Source properties for noisy extraction experiments

| Source | distribution | circ. measure ($r$) | norm. MSPE |
|---|---|---|---|
| $s_1(k)$ | Super-Gaussian | 0.02 | 2.81 |
| $s_2(k)$ | Sub-Gaussian | 1.00 | 2.83 |
| $s_3(k)$ | Super-Gaussian | 1.00 | **2.80** |

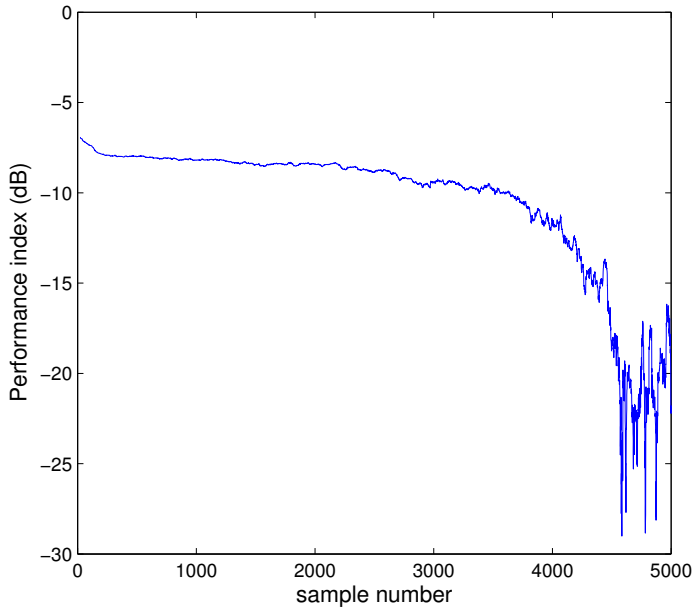



**Figure 4.8** Extraction of complex sources from a prewhitened noisy mixture with additive doubly white noncircular Gaussian noise, using algorithm (4.28a)–(4.28c) with a WL predictor.

### 4.3.2 EEG artifact extraction

Next, the usefulness of the proposed complex BSE scheme on the extraction of eye muscle activity (electrooculogram–EOG) from real-world EEG recordings is demonstrated. In real-time brain computer interfaces (BCI) it is desirable to identify and remove such artifacts from the contaminated EEG [107].

In the experiment, EEG signals used were from the electrodes Fp1, Fp2, C5, C6, O1, O2 with the ground electrode placed at Cz, as shown in Figure 4.9. In addition, EOG activity was also recorded from vEOG and hEOG channels, to provide a reference for the performance assessment of the extraction[2]. Data were sampled at 512 Hz and recorded for 30 seconds. Notice that the effects of the artifacts diminish with the distance from the eyes, being most pronounced for the frontal electrodes Fp1 and Fp2

[2]As there is no knowledge of the mixing matrix, comparison of power spectra of the original and extracted EOG is used to validate the performance of the proposed complex BSE algorithms.

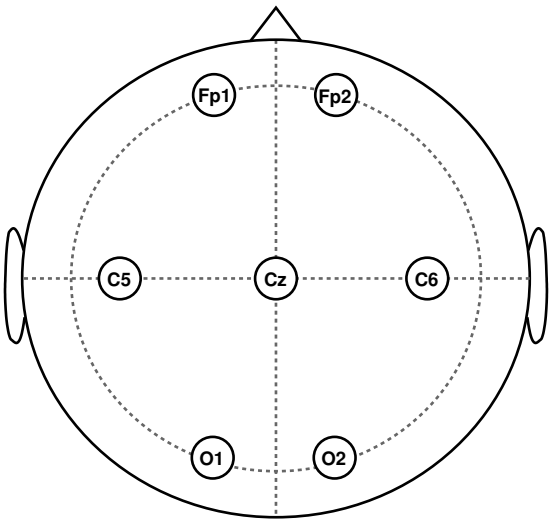



**Figure 4.9** EEG channels used in the experiment (according to the 10-20 system)

(Figure 4.10(a)).

Pairing spatially symmetric electrodes to form complex signals facilitates the use of cross-information, and simultaneous modelling of the amplitude-phase relationships. Thus, pairs of symmetric electrodes were combined to form three temporal complex EEG signals given by

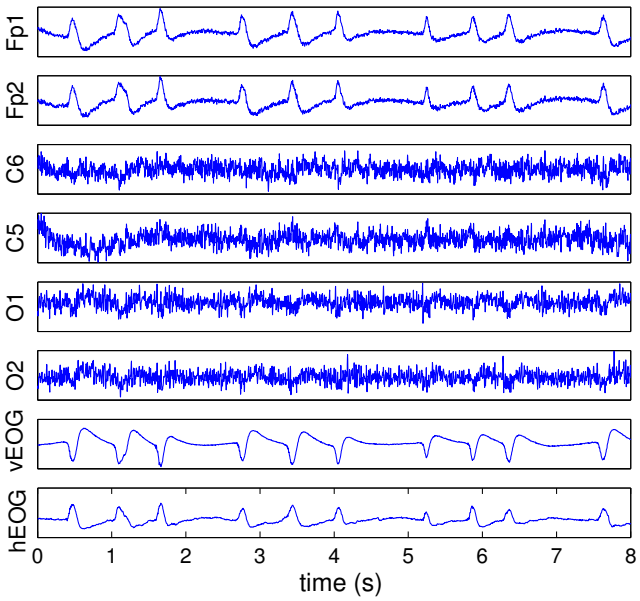
$$\begin{aligned} x_1(k) &= \text{Fp1}(k) + \jmath\, \text{Fp2}(k) \\ x_2(k) &= \text{C5}(k) + \jmath\, \text{C6}(k) \\ x_3(k) &= \text{O1}(k) + \jmath\, \text{O2}(k), \end{aligned} \tag{4.31}$$

and $\mathbf{x} = [x_1(k), x_2(k), x_3(k)]^T$.

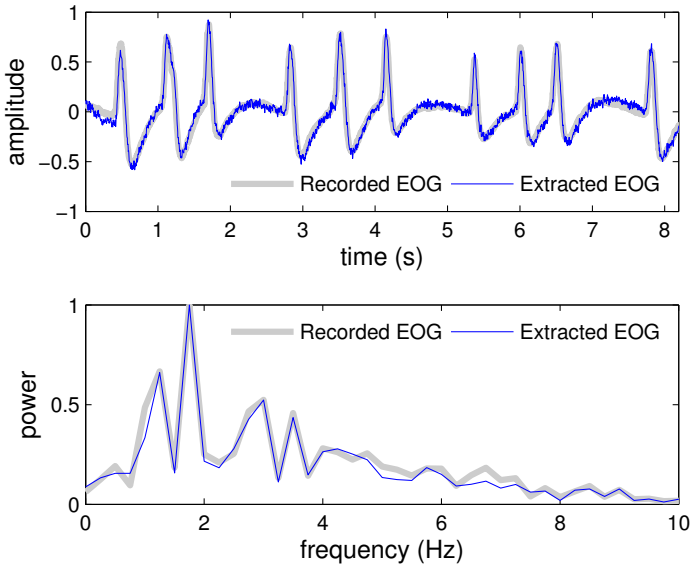
First, the algorithm in (4.15a)–(4.15c) was used to remove EOG, using the step-size $\mu = 5 \times 10^{-3}$, with filter length $M = 70$ and step-sizes $\mu_h = \mu_g = 10^{-4}$ for the standard and conjugate coefficients of ACLMS. The estimated EOG artifact was represented by the real component of the extracted signal, $\Re\{y(k)\}$, as illustrated in Figure 4.10(b), in both the time and frequency domain (the normalised power spectrum). The original vEOG signal is included for reference, confirming a successful extraction of the EOG artifact from EEG.

## 4.4 Summary

The blind source extraction of complex signals from both noise-free and noisy mixtures has been addressed. The normalised MSPE, measured at the output of a widely linear predictor, has been utilised as a criterion to extract sources based on their degree of predictability. The effectiveness of the widely linear model in this context has




(a) The first 8 seconds of the EEG and EOG recordings




(b) *Top:* first 8 seconds of the extracted EOG signal (thick grey line) and recorded vEOG signal (thin line), after normalising amplitudes, *Bottom:* normalised power spectra of the extracted EOG signal (thin line) and the original vEOG signal (thick grey line)

**Figure 4.10** Extraction of the EOG artifact due to eye movement from EEG data, using algorithm (4.15a)–(4.15c).

been demonstrated, verifying that the proposed approach is suitable for both second-order circular (proper) and noncircular (improper) signals, and for general doubly white additive complex noises (improper). For circular sources, the proposed BSE approach (P-cBSE) has been shown to perform as good as standard approaches, whereas for noncircular sources it has been shown to exhibit theoretical and practical advantages over the existing methods. The performance of the proposed algorithm has been illustrated by simulations in noise-free and noisy conditions. In addition, the application of the proposed method has been demonstrated in the extraction of artifacts from corrupted EEG signals directly in the time domain.

## 4.A Derivation of the Mean Square Prediction Error

The error at the output of the WL predictor, $e(k)$ can be written as

$$\begin{aligned} e(k) &= y(k) - y_{WL}(k) \\ &= y(k) - \mathbf{h}^T(k)\mathbf{y}(k) - \mathbf{g}^T(k)\mathbf{y}^*(k) \\ &= \mathbf{w}^H \underbrace{\Big(\mathbf{x}(k) - \sum_{m=1}^{M} h_m(k)\mathbf{x}(k-m)\Big)}_{\triangleq \bar{\mathbf{x}}_h(k)} - \mathbf{w}^T \underbrace{\sum_{m=1}^{M} g_m(k)\mathbf{x}^*(k-m)}_{\triangleq \bar{\mathbf{x}}_g(k)} \\ &= \mathbf{w}^H \bar{\mathbf{x}}_h(k) - \mathbf{w}^T \bar{\mathbf{x}}_g(k) \end{aligned} \tag{4.32}$$

and, the MSPE can be expanded as

$$\begin{aligned} &E\{|e(k)|^2\} = \\ &\quad E\Big\{ \big(\mathbf{w}^H \bar{\mathbf{x}}_h(k) - \mathbf{w}^T \bar{\mathbf{x}}_g(k)\big)\big(\bar{\mathbf{x}}_h^H(k)\mathbf{w} - \bar{\mathbf{x}}_g^H(k)\mathbf{w}^*\big) \Big\} \\ &\quad = \mathbf{w}^H E_1 \mathbf{w} - \mathbf{w}^H E_2 \mathbf{w}^* - \mathbf{w}^T E_3 \mathbf{w} + \mathbf{w}^T E_4 \mathbf{w}^* \end{aligned} \tag{4.33}$$

where

$$\begin{aligned} E_1 = E\{\bar{\mathbf{x}}_h(k)\bar{\mathbf{x}}_h^H(k)\},\ E_2 = E\{\bar{\mathbf{x}}_h(k)\bar{\mathbf{x}}_g^H(k)\} \\ E_3 = E\{\bar{\mathbf{x}}_g(k)\bar{\mathbf{x}}_h^H(k)\},\ E_4 = E\{\bar{\mathbf{x}}_g(k)\bar{\mathbf{x}}_g^H(k)\}. \end{aligned}$$

Recall that the observation $\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{v}(k)$, so the MSPE can be divided into terms relating to the source (denoted by $e_s^2$) and those relating to the noise (denoted by $e_v^2$), giving $E\{|e(k)|^2\} = e_s^2 + e_v^2$. Assuming a noise-free case, that is, $e_v^2 = 0$, the

values of $E_i, i = \{1, 2, 3, 4\}$ can be expressed as

$$E_1 = \mathcal{C}_{\mathbf{ss}}(0) - \sum_{m=1}^{M} h_m(k)\mathcal{C}_{\mathbf{ss}}(-m) - \sum_{m=1}^{M} h_m^*(k)\mathcal{C}_{\mathbf{ss}}(m) + \sum_{m,\ell=1}^{M} h_m(k)h_\ell^*(k)\mathcal{C}_{\mathbf{ss}}(\ell - m) \tag{4.34}$$

$$E_2 = \sum_{m=1}^{M} g_m^*(k)\mathcal{P}_{\mathbf{ss}}(m) - \sum_{m,\ell=1}^{M} h_m(k)g_\ell^*(k)\mathcal{P}_{\mathbf{ss}}(\ell - m) \tag{4.35}$$

$$E_3 = \sum_{m=1}^{M} g_m(k)\mathcal{P}_{\mathbf{ss}}^*(m) - \sum_{m,\ell=1}^{M} h_m^*(k)g_\ell(k)\mathcal{P}_{\mathbf{ss}}^*(\ell - m) \tag{4.36}$$

$$E_4 = \sum_{m,\ell=1}^{M} g_m^*(k)g_\ell(k)\mathcal{C}_{\mathbf{ss}}(\ell - m). \tag{4.37}$$

Since $E_3 = E_2^H$ and $z + z^* = 2\Re\{z\}$, equations (4.34)–(4.37) can be simplified and substituted in (4.33) to produce the final result $E\{|e(k)|^2\} = e_s^2$, as given in (4.6).

To derive the MSPE relating to the $n$th source, notice that the sources are assumed uncorrelated and so the covariance and pseudo-covariance matrices are diagonal. It is then straightforward to express the $n$th diagonal element of (4.34)–(4.37) to produce (4.7).

In the noisy case, the values of $E_i$ pertaining to $e_v^2$ (denoted by $E_{i,v}$) can be evaluated in a similar fashion to that in (4.34)–(4.37), noticing that $\mathcal{C}_{\mathbf{vv}}(\delta) = \mathcal{P}_{\mathbf{vv}}(\delta) = \mathbf{0}$ for $\delta \neq 0$. Thus,

$$E_{1,v} = \mathcal{C}_{\mathbf{vv}}(0) + \sum_{m=1}^{M} h_m(k)h_m^*(k)\mathcal{C}_{\mathbf{vv}}(0) \tag{4.38}$$

$$E_{2,v} = -\sum_{m=1}^{M} h_m(k)g_m^*(k)\mathcal{P}_{\mathbf{vv}}(0) \tag{4.39}$$

$$E_{3,v} = -\sum_{m=1}^{M} h_m^*(k)g_m(k)\mathcal{P}_{\mathbf{vv}}(0) \tag{4.40}$$

$$E_{4,v} = \sum_{m=1}^{M} g_m(k)g_m^*(k)\mathcal{C}_{\mathbf{vv}}(0) \tag{4.41}$$

which when substituted in (4.33) and simplified results in

$$e_v^2 = \mathbf{w}^H\hat{\mathcal{C}}_{\mathbf{vv}}\mathbf{w} + \Re\{\mathbf{w}^H\hat{\mathcal{P}}_{\mathbf{vv}}\mathbf{w}^*\} \tag{4.42}$$

and

$$\hat{\mathcal{C}}_{\mathbf{vv}} = [1 + \mathbf{h}^H(k)\mathbf{h}(k) + \mathbf{g}^H(k)\mathbf{g}(k)]\sigma_v^2\mathbf{I} \tag{4.43}$$

$$\hat{\mathcal{P}}_{\mathbf{vv}} = \begin{cases} 2\mathbf{g}^H(k)\mathbf{h}(k)\tau_v^2\mathbf{I}, & \mathbf{v}(k) \text{ for doubly white} \\ \mathbf{0}, & \mathbf{v}(k) \text{ for circular white} \end{cases} \tag{4.44}$$

where $\hat{\mathcal{C}}_{\mathbf{vv}}$ and $\hat{\mathcal{P}}_{\mathbf{vv}}$ are written in their vector form.

# Chapter 5

# Kurtosis Based Blind Source Extraction of Complex Noncircular Signals

## 5.1 Introduction

The maximisation of non-Gaussianity is an established optimisation paradigm in blind source separation, and in particular in Independent Component Analysis (ICA). This rests upon the central limit theorem, as an observed mixture of several independent random processes has a more Gaussian distribution than the individual distributions of the original [12]. This opens the possibility to recover sources based on their degree of non-Gaussianity. This has led to the introduction of information theoretic approaches based on the maximisation of negentropy [12, 108], defined as a non-negative measure of entropy normalised such that it is zero for a Gaussian random variable.

It is common to approximate the negentropy function of a given distribution using some suitable nonlinearities. In the real domain, a simple nonlinear approximation is the kurtosis[1], the fourth order moment of a random variable, which provides a simple yet effective means to model the degree of Gaussianity within a signal, measuring the deviation from a Gaussian distribution. The kurtosis of a Gaussian random variable is zero, while sub- and super-Gaussian signals have respectively negative and positive kurtosis values. The design of suitable cost functions based on the kurtosis measure can thus allow for the estimation of the latent sources from the observed mixture.

The online nature of gradient decent optimisation for kurtosis based algorithms allows for the sequential estimation of sources, which, can also be viewed as blind

[1]The nonlinear function $G(y) = y^4$ is an approximation of the negentropy function based on the kurtosis measure.

source extraction. Alternatively, optimisation of kurtosis based cost functions based on the Newton method leads to the class of fixed-point like algorithms [21, 12], such as the FastICA algorithm using kurtosis. These algorithms have the advantage of fast convergence, and allow for the sequential or simultaneous separation of sources. However, their offline batch mode of operation does not make them suitable for real-time applications.

The kurtosis measure is sensitive to outliers, and to this end the scale invariant normalised kurtosis measure, was introduced to reduce the effect of outliers, while providing a uniform measure for the comparison of various signals. The algorithm in [109] also known as the KuicNet algorithm, utilises a normalised kurtosis cost function, however, it is not stable in the separation of sub-Gaussian sources [10]. The kurtosis based blind source extraction algorithm proposed in [34], uses a cost function based on the normalised kurtosis, and is capable of extracting real-valued desired sources from a noisy mixture.

In the complex domain, kurtosis can be defined in various forms, however, the most common one is based on a real-valued measure which follows the definition in $\mathbb{R}$ and is zero for complex Gaussian random variables and negative or positive for sub- and super-Gaussian random variables; see Section 2.2. In the past few years, extension of kurtosis-based BSS algorithms to the complex domain has been considered. The original complex FastICA algorithm by Bingham and Hyvärinen [41], assumed circular sources and was designed for the estimation of the negentropy function using generalised nonlinearities. The assumption of the properness of sources allows for the simplification of the kurtosis definition $\mathbb{C}$ (see Equation (2.24)) and results in a simple nonlinearity, however this limits the optimal scope of the algorithm to the class of proper complex sources.

In [71], Douglas introduced a fixed-point kurtosis based algorithm with prewhitening using the strong uncorrelating transform (SUT) [69] to diagonalise both covariance and pseudo-covariance matrices. The authors in [75] investigated kurtosis-based algorithms for separation of complex-valued sources using both gradient and Newton method optimisation. The algorithms of [71] and [75] were designed for the generality of complex-valued sources and thus outperformed the complex FastICA algorithm of [41] with the kurtosis-based nonlinearity.

The above mentioned algorithms provide kurtosis based methodologies for the separation of sources in $\mathbb{C}$, however, they do not consider the blind extraction of complex-valued sources in the presence of additive noise. Furthermore, the performance of such BSE algorithms in real-time applications has not been assessed. To this end, in this chapter, a new class of complex BSE algorithms based on the degree of kurtosis, and in the presence of complex-valued additive noise is explored. This provides an extension of the methodology presented in [34] to the generality of complex signals,

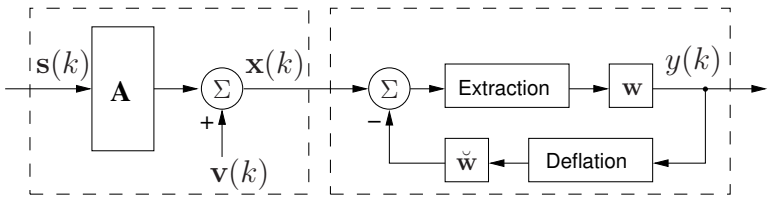



**Figure 5.1** The noisy mixture model, and BSE architecture.

both complex circular and noncircular. A modified cost function is also proposed so as to cater for blind extraction from noisy mixtures. The performance is first assessed through benchmark simulations using various synthetic sources. Extensive studies on the extraction of artifacts from electroencephalograph (EEG) signals demonstrate the usefulness of the algorithm, and are supported by performance studies using both qualitative and quantitative metrics.

## 5.2 BSE of Complex Noisy Mixtures

The diagram in Figure 5.1 shows the complex BSE architecture, where at time instant $k$, the observed signal $\mathbf{x}(k) \in \mathbb{C}^N$ is given by a linear mixture

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{v}(k) \tag{5.1}$$

where $\mathbf{s}(k) \in \mathbb{C}^{N_s}$ is the vector of latent sources, $\mathbf{A} \in \mathbb{C}^{N \times N_s}$ is the mixing matrix, and $\mathbf{v}(k) \in \mathbb{C}^N$ is the vector of additive doubly white Gaussian noise (noncircular). The sources are assumed to be independent and of zero mean and distinct kurtosis values, while no assumptions are made about the source circularity. When $\mathbf{v}(k) = \mathbf{0}$, that is, in a noise-free environment, the number of mixtures is assumed to be equal to that of the sources, however, in the case of noisy mixtures, an overdetermined mixture is necessary so as to estimate the second-order statistics of noise parameters.

The adaptive gradient descent algorithm at the extraction stage adapts the parameters of the demixing vector $\mathbf{w}$ such that the source signal with the largest (smallest) kurtosis,

$$\begin{aligned} y(k) &= \mathbf{w}^H \mathbf{x}(k) \\ &= \underbrace{\mathbf{w}^H \mathbf{A}}_{\triangleq \mathbf{u}^H} \mathbf{s}(k) + \mathbf{w}^H \mathbf{v}(k) \end{aligned} \tag{5.2}$$

is first extracted. The variance of $y(k)$ can be written in an expanded form as

$$\begin{aligned} E\{|y(k)|^2\} &= \mathbf{u}^H \mathcal{C}_{\mathbf{ss}}(0)\mathbf{u} + \mathbf{w}^H \mathcal{C}_{\mathbf{vv}}(0)\mathbf{w} \\ &= \mathbf{u}^H \mathbf{u} + \sigma_v^2 \mathbf{w}^H \mathbf{w} \end{aligned} \tag{5.3}$$

where the differences in the diagonal elements of $\mathcal{C}_{\mathbf{ss}}(0)$ are absorbed into the mixing matrix $\mathbf{A}$ to achieve an identity matrix, and the noise covariance matrix $\mathcal{C}_{\mathbf{vv}}(0) = \sigma_v^2\mathbf{I}$ (due to the whiteness assumption).

In the same spirit, the normalised kurtosis of the extracted signal $y(k)$ can be written as

$$\begin{aligned} K_c(y) &= \sum_{n=1}^{N_s} K_c(u_n^* s_n) + \sum_{n=1}^{N} \underbrace{K_c(w_n^* v_n)}_{=0} \\ &= \sum_{n=1}^{N_s} |u_n|^4 E\{|s_n|^4\} - 2|u_n|^4 (E\{|s_n|^2\})^2 - |u_n|^4 |E\{s_n^2\}|^2 \\ &= \sum_{n=1}^{N_s} |u_n|^4 K_c(s_n) \end{aligned} \tag{5.4}$$

thus having zero value for Gaussian noise. In a vectorised form, this is equivalent to

$$K_c(y) = \hat{\mathbf{u}}^H \mathbf{K}_c(\mathbf{s}) \hat{\mathbf{u}} \tag{5.5}$$

where

$$\begin{aligned} \hat{\mathbf{u}} &= [u_1^2, \ldots, u_{N_s}^2] \\ \mathbf{K}_c(\mathbf{s}) &= \operatorname{diag}\big(K_c(s_1), \ldots, K_c(s_{N_s})\big). \end{aligned} \tag{5.6}$$

The next stage within the proposed BSE scheme is the deflation process which aims to remove the extracted source $y(k)$ from the mixture $\mathbf{x}(k)$, such that

$$\mathbf{x}(k) \leftarrow \mathbf{x}(k) - \breve{\mathbf{w}} y(k) \tag{5.7}$$

where the deflation weight coefficient vector $\breve{\mathbf{w}}$ is updated using an adaptive gradient descent algorithm detailed later in this section. In principle, for $y(k)$ being an estimate of one of the original sources, say $s_n(k)$, the ideal deflation weight vector should be equal to the $n$th column of the mixing matrix $\mathbf{A}$, such that the effect of this particular source is removed from the mixture. Finally, a threshold can be set on the deflation process, so that extraction is continued until some or all the required sources have been successfully extracted [110].

### 5.2.1 Cost function

The cost function employed for the extraction of general complex sources from noisy mixtures is given by

$$\mathcal{J}(\mathbf{w}) = -\beta \frac{\operatorname{kurt}_c\big(y(k)\big)}{\big(E\{|y(k)|^2\} - \mathbf{w}^H \mathcal{C}_{\mathbf{vv}}(0)\mathbf{w}\big)^2}. \tag{5.8}$$

Note that $\mathcal{J} \in \mathbb{R}$, represents a modified version of the normalised kurtosis defined in (2.23) and is a generalisation of the methodology presented in [34]. The numerator of the cost function represents the kurtosis of the complex extracted signal, while the denominator is the square of the extracted signal power where the contributions due to noise is removed. Collectively, this forms the modified normalised kurtosis of the extracted signal minus the contributions from the noise. By using the modified normalised kurtosis instead of the standard complex kurtosis, extraction of signal with different dynamic ranges can be performed in a uniform scale, and avoid the use of a prewhitening stage. As illustrated in (5.3), the variance of $y(k)$ contains the noise variance $\sigma_v^2$, thus allowing us to remove the effect of noise from (5.8) such that only contributions from the latent sources are accounted for. Also note that while the noise variance $\sigma_v^2$ is present in the cost function (5.8), its pseudo-covariance $\tau_v^2$ is not present, suggesting that the complex domain BSE based on kurtosis is unaffected by the pseudo-spectral effects of the additive noise; this is further elaborated in Section 5.3.

In the cost function (5.8), the parameter $\beta$ dictates the order of extraction where for

i) $\beta = 1$, the order of extraction is from the high to low degree of non-Gaussianity (super-Gaussian sources are extracted first),

ii) $\beta = -1$, the order of extraction is from low to high degree of non-Gaussianity (sub-Gaussian sources are extracted first).

The optimisation of $\mathcal{J}$ with respect to $\mathbf{w}$ can thus be stated as

$$\mathbf{w}_{opt} = \arg \min_{\|\mathbf{w}\|_2^2=1} \mathcal{J}(\mathbf{w}) \tag{5.9}$$

where the norm of the demixing vector is constrained to unity to avoid very small coefficient values.
Rewriting and simplifying (5.8) in terms of (5.3) and (5.6) results in

$$\mathcal{J}(\mathbf{w}) = -\frac{\hat{\mathbf{u}}^H |\mathbf{K}_c(\mathbf{s})| \hat{\mathbf{u}}}{(\mathbf{u}^H \mathbf{u})^2} = -\acute{\mathbf{u}}^H |\mathbf{K}_c(\mathbf{s})| \acute{\mathbf{u}} \tag{5.10}$$

where

$$\acute{\mathbf{u}}^H \triangleq \frac{\hat{\mathbf{u}}^H}{\mathbf{u}^H \mathbf{u}} = \frac{\hat{\mathbf{u}}^H}{\|\mathbf{u}\|_2^2}. \tag{5.11}$$

Notice that $\|\acute{\mathbf{u}}\|_2^2 = \frac{\|\hat{\mathbf{u}}\|_2^2}{(\|\mathbf{u}\|_2^2)^2} \leq 1$ and is equal to unity only if one of the components in the vector $\mathbf{u}$ is non-zero. Given the constraint on $\|\acute{\mathbf{u}}\|$, the solution to the optimisation of (5.10) is a vector $\acute{\mathbf{u}}_{opt}$ of unit norm such that $\mathbf{u}_{opt}$ has a single non-zero component at a position corresponding to the diagonal element in $\mathbf{K}_c(\mathbf{s})$ having the largest magnitude. For this to be valid, a demixing vector assumes the form $\mathbf{w}_{opt} = \mathbf{A}^{H\#}\mathbf{u}_{opt}$, where the symbol $(\cdot)^{\#}$ denotes the matrix pseudo-inverse operator [34].

### 5.2.2 Adaptive algorithm for extraction

Optimisation of (5.8) is performed using an adaptive gradient descent algorithm which updates the values of $\mathbf{w}$ so as to maximise the modified normalised kurtosis and thus minimise the cost function $\mathcal{J}(\mathbf{w})$. Based on $\mathbb{CR}$ calculus and Brandwood's result[2] (see Appendix 2), the gradient is thus expressed as

$$\begin{aligned}\nabla_{\mathbf{w}^*}\mathcal{J} &= \frac{\beta\,\mathbf{x}(k)}{(m_2(y)-\sigma_v^2)^3}\Big(y^*(k)\big(m_4(y)-2m_2^2(y)-|p_2(y)|^2\big)\\ &\quad+\big(m_2(y)-\sigma_v^2\big)\big(-y(k)y^{*2}(k)+2m_2(y)y^*(k)+p_2^*(y)y(k)\big)\Big)\\ &= \phi\big(y(k)\big)\mathbf{x}(k)\end{aligned} \tag{5.12}$$

where the symbol $\phi\big(y(k)\big)$ is used for simplification and $m_\ell(y)$ and $p_\ell(y)$ are respectively the $\ell$-th moment and pseudo-moment at time instant $k$ (the time index dropped), estimated using the moving average estimators

$$\begin{aligned}m_\ell\big(y(k)\big) &= (1-\alpha)m_\ell\big(y(k-1)\big)+\alpha|y(k)|^\ell,\ \ell=\{2,4\}\\ p_\ell\big(y(k)\big) &= (1-\alpha)p_\ell\big(y(k-1)\big)+\alpha\big(y(k)\big)^\ell,\ \ell=2\end{aligned} \tag{5.13}$$

where $\alpha\in[0,1]$ is the forgetting factor.
The kurtosis based BSE update algorithm (K-cBSE) for the demixing vector is thus given by

$$\mathbf{w}(k+1)=\mathbf{w}(k)-\mu\phi\big(y(k)\big)\mathbf{x}(k), \tag{5.14}$$

or in an expanded form as

$$\begin{aligned}\mathbf{w}(k+1)=\mathbf{w}(k)-\frac{\mu\beta}{\big(m_2(y)-\sigma_v^2\big)^3}\Big(&y^*(k)\big(m_4(y)-2m_2^2(y)-|p_2(y)|^2\big)\\ &+\big(m_2(y)-\sigma_v^2\big)\big(-y(k)y^{*2}(k)+2m_2(y)y^*(k)+p_2^*(y)y(k)\big)\Big)\mathbf{x}(k),\end{aligned} \tag{5.15}$$

where $\mu$ is the small positive step-size.

To preserve the unit norm property, the demixing vector is normalised at each iteration, that is

$$\mathbf{w}(k+1)\leftarrow\frac{\mathbf{w}(k+1)}{\|\mathbf{w}(k+1)\|_2}. \tag{5.16}$$

Notice that in extracting circular sources, the moment $p_\ell$ vanishes, further simplifying the algorithm. Moreover, as mentioned earlier, the cost function and thus the gradient descent algorithm are not dependent on the pseudo-variance of the noise, $\tau_v^2$. The estimation of the noise variance can be performed using a subspace method, as described in [111], see Section 4.2.4.

[2] Recall that the conjugate gradient $\frac{\partial\mathcal{J}}{\partial\mathbf{w}^*}$ corresponds to the maximum change of the gradient.

### 5.2.3 Modifications to the update algorithm

In order to enhance the performance of the online gradient descent algorithm, adaptive step-size update algorithms are considered, and in particular, the complex-valued variable step-size (VSS) algorithm [3] and the complex-valued generalised normalised gradient descent (GNGD) type algorithm [4] are used.

By adapting the step-size of the algorithm at each iteration, it is possible to automatically adjust the speed of convergence of the algorithm without employing empirical values for the step-size. Thus, the algorithm will have a larger step-size when the K-cBSE algorithm is far from the solution of the optimisation problem (5.9), while the step-size becomes smaller when the the algorithm is closer to the solution. As a result, the algorithm has a faster convergence compared to one with a fixed step-size. However, the VSS algorithm is not suitable for use in a nonstationary and noisy environments, where the update in the step-size does not aid the algorithm.

The GNGD algorithm is distinguished from the VSS algorithm as it adjusts the regularisation parameter in a normalised algorithm. While in a standard normalised algorithm, a small input magnitude can lead to unstability in the algorithm, the GNGD algorithm adapts the regularisation parameter to ensure robust performance for signals of small magnitude.

At each iteration $k$, the VSS algorithm minimises the cost function $\mathcal{J}$ in (5.8) with respect to $\mu(k-1)$ to provide the update of the step-size, given as

$$\begin{aligned} \mu(k) &= \mu(k-1) - \eta \nabla_{\mu} \mathcal{J}\big|_{\mu=\mu(k-1)} \\ \nabla_{\mu} \mathcal{J} &= \nabla_{\mathbf{w}^*} \mathcal{J} \cdot \frac{\partial \mathbf{w}^*(k)}{\partial \mu(k-1)} \\ \boldsymbol{\psi}(k) &= \gamma \boldsymbol{\psi}(k-1) - \nabla_{\mathbf{w}^*} \mathcal{J}\big|_{\mathbf{w}^*=\mathbf{w}^*(k-1)} \end{aligned} \tag{5.17}$$

where $\boldsymbol{\psi}(k) \triangleq \frac{\partial \mathbf{w}^*(k)}{\partial \mu(k-1)} \approx \frac{\partial \mathbf{w}^*(k)}{\partial \mu(k)}$ and $\eta$ and $\gamma$ are step-sizes.

The GNGD-type algorithm is based on a normalised version of (5.15), given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{\mu}{|\phi\big(y(k)\big)|^2 \cdot \|\mathbf{x}(k)\|_2^2 + \epsilon(k)} \phi\big(y(k)\big)\mathbf{x}(k) \tag{5.18}$$

where $\epsilon(k)$ is an adaptive regularisation parameter and $\phi\big(y(k)\big)$ is defined in Equation (5.12). The gradient adaptive regularisation parameter is then given by

$$\epsilon(k+1) = \epsilon(k) - \rho\mu \frac{\Re\big\{\phi\big(y(k)\big)\mathbf{x}^T(k)\phi^*\big(y(k-1)\big)\mathbf{x}^*(k-1)\big\}}{\big(|\phi\big(y(k-1)\big) \cdot \|\mathbf{x}(k-1)\|_2^2 + \epsilon(k-1)\big)^2} \tag{5.19}$$

where $\rho$ is a step-size. The derivation of the algorithm is given in Appendix 5.A at the end of this chapter.

### 5.2.4 Adaptive algorithm for deflation

The deflation procedure insures that after each extraction stage, the estimated source is removed from all the mixture vectors, so that the next source with maximum (minimum) kurtosis can be extracted. This can be achieved based on the cost function [110]

$$\mathcal{J}_d(\breve{\mathbf{w}}) = \|\mathbf{x}_{n+1}(k)\|^2 = \mathbf{x}_{n+1}^H(k)\mathbf{x}_{n+1}(k) \tag{5.20}$$

which is minimised with respect to the deflation weight coefficient $\breve{\mathbf{w}}$. The notation $\mathbf{x}_n(k)$ denotes the mixture at the $n$th extraction stage, which is given by vectors

$$\mathbf{x}_{n+1}(k) = \mathbf{x}_n(k) - \breve{\mathbf{w}}(k)y_n(k). \tag{5.21}$$

Given an invertible mixing matrix $\mathbf{A}$, the vector $\breve{\mathbf{w}}$ is ideally equal to a column of $\mathbf{A}^{-1}$, which corresponds to the $n$th extracted source $y_n(k)$. The gradient can thus be calculated as

$$\nabla_{\breve{\mathbf{w}}^*}\mathcal{J}_d = \frac{\partial \mathcal{J}_d}{\partial \mathbf{x}_{n+1}^*} \cdot \frac{\partial \mathbf{x}_{n+1}^*}{\partial \breve{\mathbf{w}}^*} = -y_n^*(k)\mathbf{x}_{n+1}(k) \tag{5.22}$$

and the online algorithm for BSE then becomes

$$\breve{\mathbf{w}}(k+1) = \breve{\mathbf{w}}(k) + \mu_d y_n^*(k)\mathbf{x}_{n+1}(k), \tag{5.23}$$

with $\mu_d$ a step-size. The drawback of this method is that any errors in the deflation process will propagate and affect the extraction and deflation of subsequent stages. It is therefore important that the step-size parameter is set appropriately for each $n$th deflation stage to ensure successful removal of the extracted source $y_n(k)$.

In the design of complex adaptive algorithms, it is common to utilise a widely linear model to ensure that the algorithm is capable of processing the generality of complex signals [63]. In the case of the update for the deflation weight coefficient (5.23), however, a linear model is considered as the original BSS mixing model (4.1) is strictly linear and thus a widely linear deflation model is not required.

## 5.3 Simulations and Discussions

The extraction of synthetic sources from noise-free and noisy mixtures, with various degrees of complex noncircular noise levels are considered. The performance for the synthetic data were measured using the Performance Index (PI) [10] given by Equation (4.30).

For each synthetic experiment, the results were produced through averaging 100 independent trials. The mixing matrix $\mathbf{A}$ was generated randomly as a full rank complex

matrix and the demixing vector was initialised randomly. The values of the extraction and deflation step-size $\mu$ and $\mu_d$ were set empirically, and the forgetting factor $\alpha$ in (5.13) was set as 0.975. The complex additive Gaussian noise was both of circular white with circularity measure $r = 0$ and noncircular doubly white with $r = 0.93$, with $r$ defined in Equation (2.17). The real-world sources were the electroencephalogram data corrupted by power line noise and electrooculogram artifacts.

### 5.3.1 Benchmark Simulation 1: Synthetic sources

In the first set of simulations, a noise-free mixture of 3 complex sources with various degrees of circularity and $N = 5000$ samples were generated and mixed using a $3 \times 3$ mixing matrix. These signals are illustrated in Figure 5.2 and their properties listed in Table 5.1(a). Extraction was performed in order from highest to lowest kurtosis, hence the value of $\beta = 1$ in (5.8).

In the first experiment, the performance of the algorithm (5.15) using the adaptive step-size methods was compared in the extraction of the first source with the value of $\mu$ set to $0.01$ and the initial demixing vector set randomly and fixed for all consecutive extraction steps. It can be seen from the performance curves in Figure 5.3 that the best performance was achieved using the GNGD method with a PI of approximately -45 dB at the steady-state. The performance curve resulting from the normalised method indicates successful extraction with a PI of around -25 dB. The performance of the algorithm using the standard step-size and VSS were comparable, with a PI of around -20 dB. In the following simulations, the GNGD based K-cBSE algorithm is utilised.

In the next set of simulations, the extraction of all the three sources (Figure 5.2) was considered. The value of $\mu$ was set respectively to $0.01, 0.008$ and $10^{-5}$ for the consecutive extraction stages. As shown in Figure 5.4, the algorithm successfully extracted all the three sources, as indicated by a PI of less than -20 dB at the steady-state for the extraction iteration $i = \{1, 2, 3\}$, converging to steady-state after 2500 samples in the first extraction stage ($i = 1$) and around 1000 samples in the second and third extraction stage ($i = \{2, 3\}$). The decreasing PI value at each consecutive extraction stage can be attributed to the unavoidable errors accumulated in the deflation.

The scatter plot of the three estimated sources $y_1(k), y_2(k)$ and $y_3(k)$ are illustrated in Figure 5.2. The normalised kurtosis of the estimated sources were respectively calculated as $K_c(y_1) = 11.84, K_c(y_2) = 1.36$ and $K_c(y_3) = -2.00$ corresponding to those of the original sources, given in Table 5.1(a); the scale and rotation ambiguities of the source estimates are also visible.

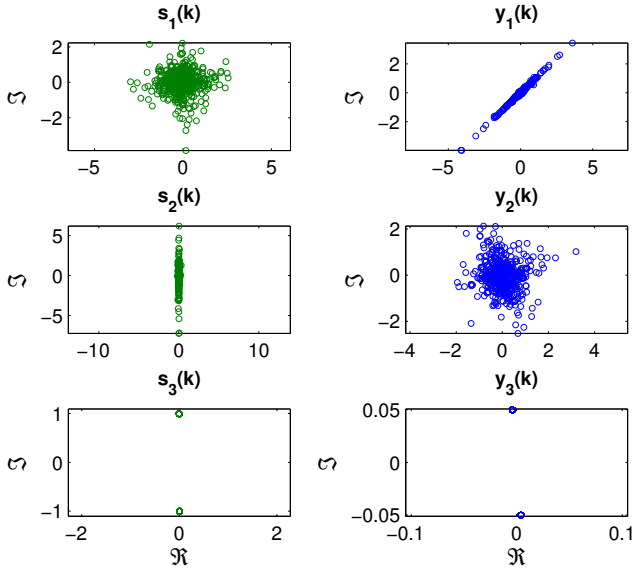



**Figure 5.2** Scatter plot of the complex-valued sources $s_1(k)$, $s_2(k)$ and $s_3(k)$, with the signal properties described in Table 5.1(a) (*left hand column*). Scatter plot of estimated sources $y_1(k)$, $y_2(k)$ and $y_3(k)$, extracted according to a decreasing order of kurtosis ($\beta = 1$) (*right hand column*).

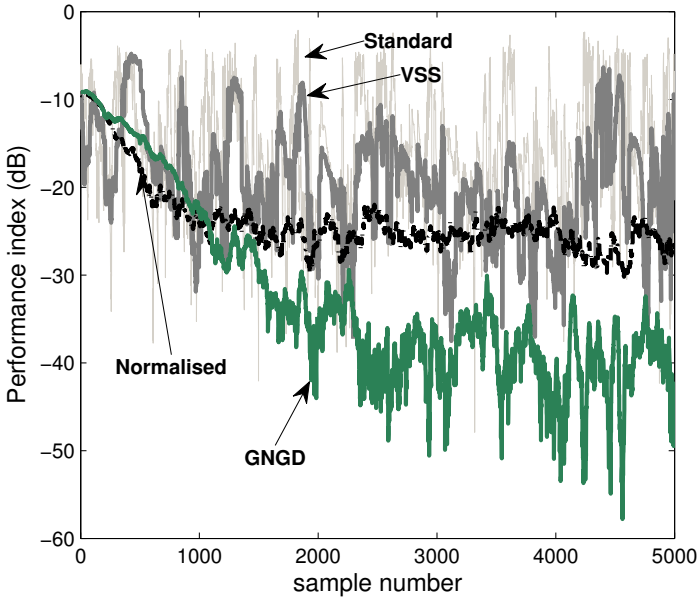



**Figure 5.3** Comparison of the effect of step-size adaptation on the performance of algorithm (5.15) for the extraction of a single source.

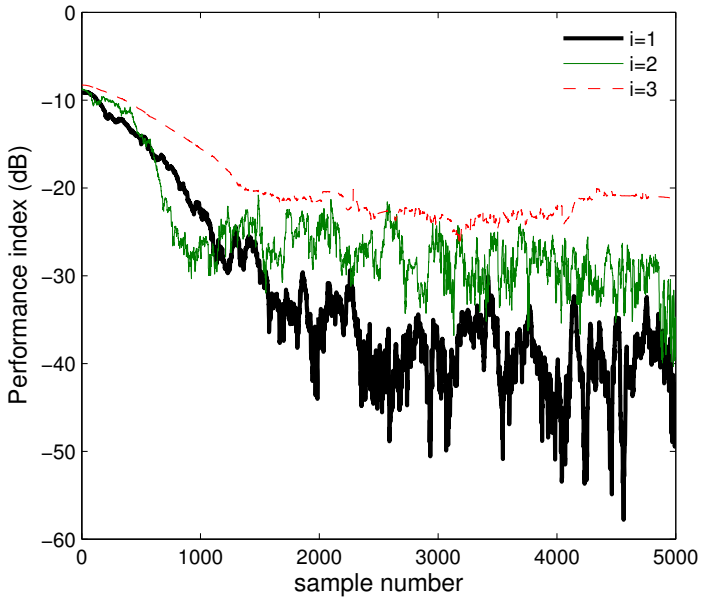



**Figure 5.4** Extraction of complex circular and noncircular sources from a noise-free mixture based on kurtosis.

### 5.3.2 Benchmark Simulation 2: Communication sources

The extraction of BPSK, QPSK and 16-QAM sources is demonstrated next, illustrated in Figure 5.5, from a noise-free mixture; the source properties are given in Table 5.1(b). The BSPK source is noncircular, while the QPSK and 16-QAM sources are second-order circular. The value of $\beta = -1$, such that source with the smallest kurtosis is extracted first (BSPK), followed on to the least sub-Gaussian (16-QAM). The number of samples generated was $N = 5000$ and the value of $\mu$ was chosen empirically and set respectively to 0.95, 2 and 0.1 for each iteration $i = \{1, 2, 3\}$ of the extraction stage.

The algorithm had a very fast convergence in extracting the source signals (see Figure 5.6) in the desired order. The scatter plots of the extracted sources are given in Figure 5.5 with the respective normalised kurtosis values calculated as $K_c(y_1) = -2.00, K_c(y_2) = 1.00$ and $K_c(y_3) = -0.67$ which are in close proximity to the true kurtosis values in Table 5.1(b).

### 5.3.3 Benchmark Simulation 3: Noisy mixture

In the next experiment, the extraction of complex-valued sources from a noisy mixture was considered. Three sources of 5000 samples were considered (see Table 5.1(c), Figure 5.7) and were mixed using a randomly generated $4 \times 3$ mixing matrix $\mathbf{A}$ to allow for the estimation of the noise variance and pseudo-variance. The additive

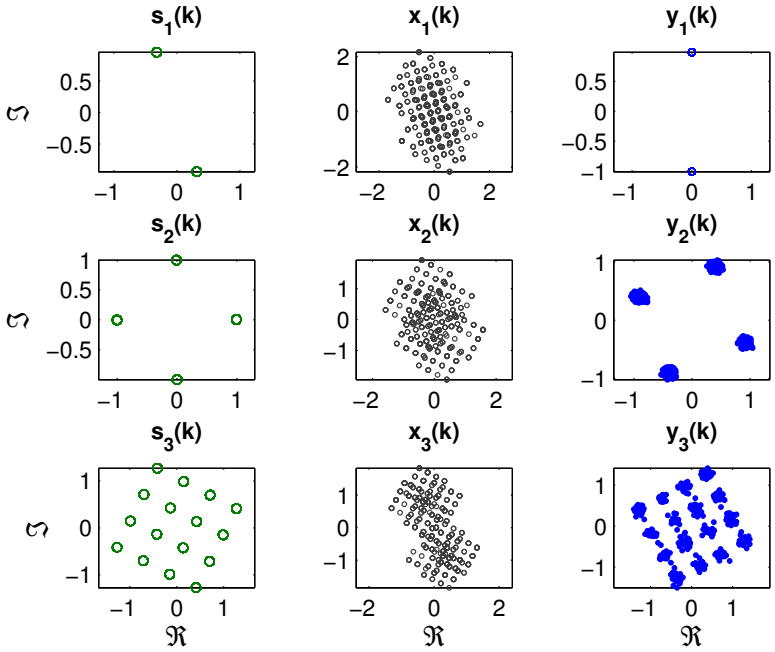



**Figure 5.5** Scatter plot of the BSPK, QPSK and 16-QAM sources $s_1(k)$, $s_2(k)$ and $s_3(k)$, with properties given in Table 5.1(b) (*left column*), observed mixtures $x_1(k)$, $x_2(k)$ and $x_3(k)$ (*middle column*), and the estimated sources $y_1(k)$, $y_2(k)$ and $y_3(k)$ (*right column*).

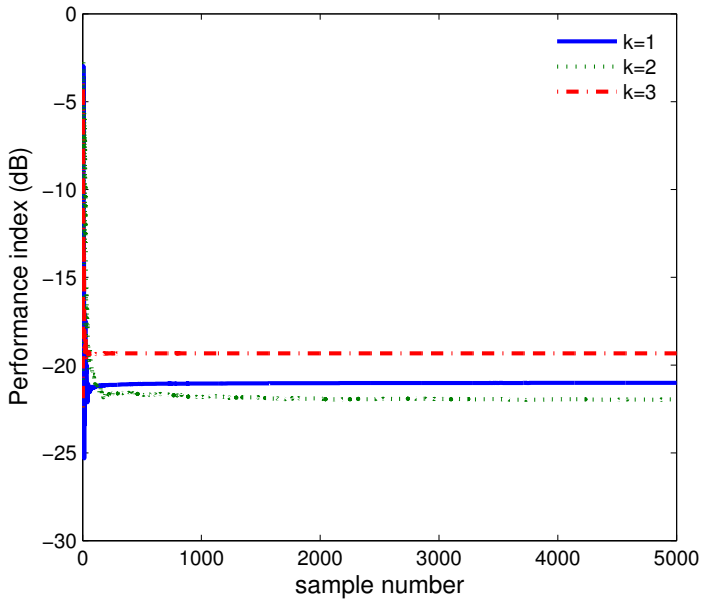



**Figure 5.6** Extraction of communication sources (properties given in Table 5.1(b)) in a noise-free environment.

noise was doubly white Gaussian noise with variance $\sigma_v^2 = 0.1$ and pseudo-variance $\tau_v^2 = 0.0924 + \jmath 0.0011$, estimated using the subspace method described in Section 5.2. The sources were extracted in an increasing order of kurtosis ($\beta = -1$) with the step-size $\mu = 0.5$.

The scatter plot of the first estimated source with the smallest kurtosis, $y_1(k)$ is illustrated in Figure 5.7 with a calculated normalised kurtosis of $K_c(y_1) = -1.80$, which is within a $10\%$ range of the true value, given in Table 5.1(c). The Performance Index, shown in Figure 5.8, demonstrates a fast convergence to a value of around -40 dB in approximately 1000 samples, and continuing a steady convergence to -50 dB by 5000 samples.

It was shown in Section 5.2 that the performance of the algorithm (5.15) was not affected by the degree of circularity of the additive noise, such that doubly white noise is treated in a similar manner to circular white noise, where the pseudo-covariance vanishes. This was explored experimentally by systematically analysing the effect of various noise levels on the BSE algorithm (5.15). The circularity measure $r$ was varied from a value of $r = 0$ (circular) to a value of $r = 1$ (highly noncircular), while the signal-to-noise ratio (SNR) was adjusted from a near-zero noise SNR of 50 dB to a high noise environment with SNR value of -10 dB. The initial values were generated randomly and PI was averaged over 100 trials. Figure 5.9 illustrates the performance curve for the different variations in the noise properties, and confirms that while the performance is dependent on the SNR value, it does not vary with changes in the degree of noise noncircularity. In addition, the algorithm had an acceptable performance in the extraction of sources (PI $<$ -20 dB) when the SNR was above 1 dB.

## 5.4 EEG artifact extraction

In order to obtain useful information from EEG data in real-time, it is often necessary to perform post-processing to remove artifacts such as line noise and biological artifacts including those pertaining to eye movement, captured in the form of electrooculogram (EOG) and facial muscle activity represented as electromyogram (EMG). Removal of the effect of such signals from the contaminated EEG has been subject of study in previous years, with several methodologies introduced that attempt to accomplish this utilising both online and offline algorithms [112, 113, 114, 115, 116, 117, 118]. While offline algorithms are suitable for processing the recorded EEG data in clinical applications, it is necessary to utilise online algorithms for real-time applications such as those encountered in brain computer interface (BCI) scenarios.

In [118] the authors propose an online algorithm whereby the recorded EEG signals are transformed to the wavelet domain and the EOG contaminants are removed using

**Table 5.1** Source properties for Benchmark simulations

(a) Source properties for noise-free extraction Benchmark Simulation 1

| Source | Distribution | Kurtosis | circ. measure ($r$) |
|---|---|---|---|
| $s_1(k)$ | Super-Gaussian | 1.36 | 0.04 |
| $s_2(k)$ | Super-Gaussian | 11.89 | 1.00 |
| $s_3(k)$ | Sub-Gaussian | -2.00 | 1.00 |

(b) Properties of the BPSK, QPSK and 16-QAM sources used in Benchmark Simulation 2

| Source | Type | Distribution | Kurtosis | circ. measure ($r$) |
|---|---|---|---|---|
| $s_1(k)$ | BSPK | Sub-Gaussian | -2.00 | 1.00 |
| $s_2(k)$ | QPSK | Sub-Gaussian | -1.00 | 0.00 |
| $s_3(k)$ | 16-QAM | Sub-Gaussian | -0.68 | 0.00 |

(c) Source properties for noisy extraction in Benchmark Simulation 3

| Source | Distribution | Kurtosis | circ. measure ($r$) |
|---|---|---|---|
| $s_1(k)$ | Sub-Gaussian | -1.9985 | 1.0000 |
| $s_2(k)$ | Super-Gaussian | 19.1167 | 0.9988 |
| $s_3(k)$ | Super-Gaussian | 1.5426 | 0.0147 |

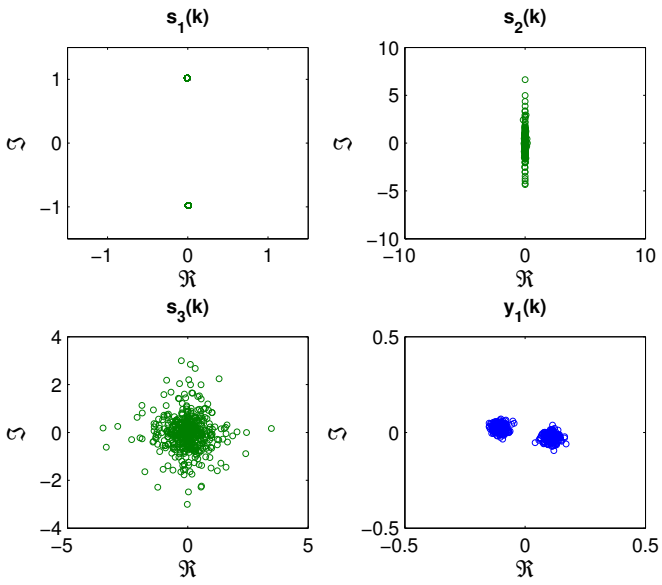



**Figure 5.7** Scatter plots of the original sources $s_1(k), s_2(k)$ and $s_3(k)$. The scatter diagram of the first estimated source $y_1(k)$ is shown in the bottom-right plot.

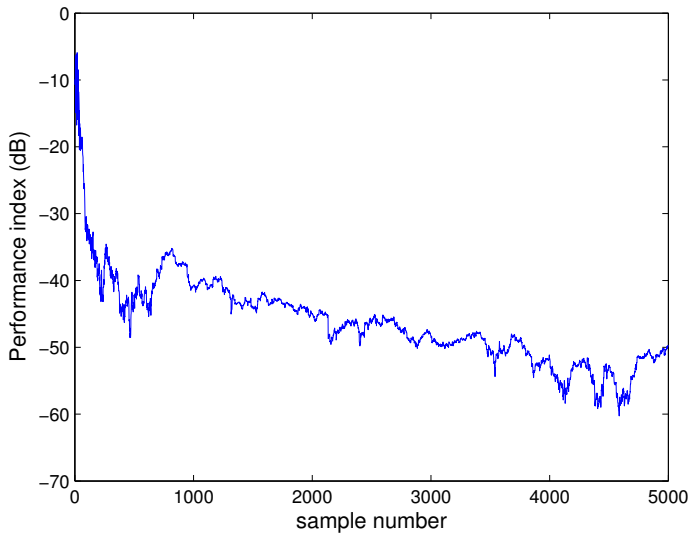



**Figure 5.8** Extraction of a complex-valued source from a noisy mixture, with the source properties given in Table 5.1(c).

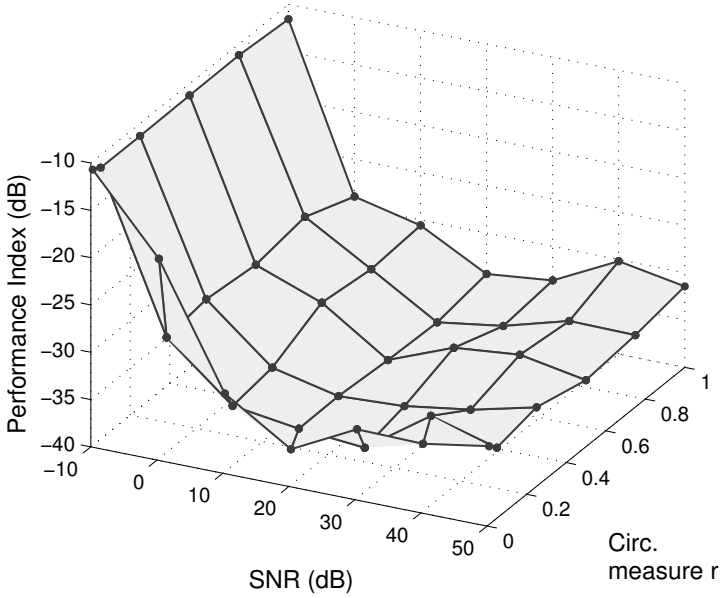



**Figure 5.9** Comparison of the performance of algorithm (5.15) with respect to changes in the SNR and the degree of noise circularity.

an adaptive recursive least squares (RLS) algorithm, before transforming the signal back to the time domain. Simulations demonstrate good performance from the algorithm, however, it would be advantageous to perform all the necessary processing in the time domain, as this way the signals are retained in their original form and less computation is required. Another wavelet domain approach to biological signal extraction was employed in [119] in order to extract the fetal electrocardiogram from a noisy mixture.

In its basic form, ICA can be applied to the contaminated EEG recording and the artifacts removed through visual inspection. As detailed in [112], an ICA algorithm separates the recorded EEG mixture into its original sources as independent components (ICs), with artifact sources identified and removed. In semi-automatic [116] and automatic [114] artifact removal methodologies, several classifications (markers) based on the statistical characteristics of the ICs are considered that allow for the detection of artifacts in the contaminated EEG. These are then compared against thresholds that determine the rejection of particular components.

In these methods, both the kurtosis and entropy of independent components have been utilised to identify and remove the artifacts. While the EEG mixtures typically have near-zero kurtosis values, artifacts such as EOG exhibit peaky distributions with highly positive kurtosis values [114], while periodic power line noise has a highly negative kurtosis value. This has been used as the main discrimination in defining classifications based on the the fourth order moment.

### 5.4.1 Data acquisition and method

The aim is to remove artifacts as independent sources extracted from the recorded EEG mixture directly in the time domain. To this end, the contaminated EEG signals were paired as the real and imaginary components of a complex signal and processed using the architecture described in Section 5.2.

In this manner, the full cross-statistical information between the corresponding electrodes and the resultant recorded EEG is maintained, while allowing for the simultaneous processing of both channels. Further iterations of the extraction process can then be used to obtain the individual pure EEG signals, or even, pipelined to a further post-processing stage, which would then extract the EEG signals based on a desired fundamental property, such as predictability.

The electrodes were placed according to the 10-20 system (Figure 5.10), and sampled at 256 Hz for 30 seconds. The EEG activity was recorded from electrodes placed at positions Fp1, Fp2, C3, C4, O1, O2 with the ground placed at Cz, while the EOG activity was recorded from the vEOG and hEOG channels with electrodes placed above and on the side of the left eye socket.

Three studies were performed with the aim to remove the artifacts simultaneously. While the rejection of the power line noise artifact is feasible by passing the recorded EEG signals through a notch filter, this solution also leads to the removal of useful information around the 50 Hz range pertaining to the EEG signals, in particular those within the gamma band (25 Hz-100 Hz).

It would therefore be desirable to automatically extract the line noise artifact along with the biological artifact from the corrupted EEG signals. In the first study the removal of EOG artifacts ('EYEBLINK' set) is considered, the second study focused on eye muscle artifacts from rolling the eyes ('EYEROLL' set), whereas the third study addressed the removal of muscle activity from raising the eyebrow ('EYEBROW' set).

In all the studies, the temporal signals from each channel pair were combined to form three complex EEG channels, given by

$$\begin{aligned} x_1(k) &= \text{Fp1}(k) + \jmath\, \text{Fp2}(k) \\ x_2(k) &= \text{C3}(k) + \jmath\, \text{C4}(k) \\ x_3(k) &= \text{O1}(k) + \jmath\, \text{O2}(k). \end{aligned} \tag{5.24}$$

This construction of the complex EEG signals allows for the simultaneous processing of the amplitude and phase information using the K-cBSE algorithm (5.15). Note that the EOG channels were not part of the mixtures considered. They are only used to assess the performance of the proposed BSE algorithm in the extraction of the EOG artifacts.

### 5.4.2 Performance measures

As no knowledge of the mixing process is available, the Performance Index (4.30) is not applicable for this case and thus several alternative quantitative and qualitative measures were used for the evaluation of the algorithm performance. These are briefly discussed below.

1. Quantitative metrics

   a) *Kurtosis*: The kurtosis values $K_c$ of the complex extracted signals indicate the success of the algorithm in extracting super-Gaussian or sub-Gaussian artifact in a specified order. In addition, the magnitude of the kurtosis $K_R$ of the real and imaginary components of the extracted sources are used to automatically select desired components. In this manner, components with negative kurtosis are labelled as power line noise, those with large positive kurtosis values are chosen as biological artifacts, while components belonging to EEG sources have a near-Gaussian distribution and have kurtosis values close to zero.

b) *Power spectra Correlation*: In a similar manner to [115], the correlation coefficient[3] between the magnitudes of the power spectra of the complex-valued recorded artifact (e.g. EOG) and extracted sources, and likewise, the correlation coefficient between the pseudo-power spectra of the complex-valued recorded artifact and the extracted sources is calculated.

This measure indicates the degree of similarity between the extracted and originally recorded artifact, and can be used to automatically select the extracted source pertaining to the biological artifact, while also quantifying the degree of performance of the extraction algorithm.

2. Qualitative metrics

a) *Hilbert-Huang Time-Frequency Analysis*: By employing time-frequency (T-F) analysis using the Hilbert-Huang (H-H) transform [120, 121], the extraction performance can be qualitatively assessed through comparison of the frequency components of the mixture and extracted source during the recording session. Also, the T-F analysis of the extracted artifacts will demonstrate the corresponding frequency components and their changes over time, making it possible to assess the quality of the extraction procedure over the recording time.

In comparison to Fourier transform based T-F analysis, such as the Short-Time Fourier Transform, the H-H transform results in much more detailed spectrogram for a given resolution. The intrinsic mode functions (IMFs) required by the H-H transform were obtained using a multivariate empirical mode decomposition (MEMD) algorithm [122], where the real and imaginary component of the complex-valued signals were taken as a single multivariate signal and processed simultaneously. It was observed that this resulted in a spectrogram with better resolution than those obtained through the separate processing of the individual components using the standard EMD algorithm.

b) *Power Spectral Distribution*: The power and pseudo-power spectra of the complex-valued extracted artifacts were compared to those belonging to the complex-valued recorded artifact. In addition, the pseudo-spectrum demonstrates the quality of the proposed method in extracting noncircular sources by observing the magnitude of both spectra[4] and noting relation (2.22). Recall that the power spectrum $\mathcal{S}_{y_n}$ and pseudo-power spectrum $\breve{\mathcal{S}}_{y_n}$ of extracted signal $y_n(k)$ are re-

[3] Recall that the correlation coefficient $\varrho_{xy}$ between two random variables $x$ and $y$ is given by $\varrho_{xy} = \sigma_{x,y}/\sigma_x\sigma_y$, where $\sigma_x$ and $\sigma_x$ are the standard deviations, and $\sigma_{x,y}$ is the cross-covariance of $x$ and $y$.

[4] It is also possible to consider the cross-spectrum of the recorded and extracted sources [123].

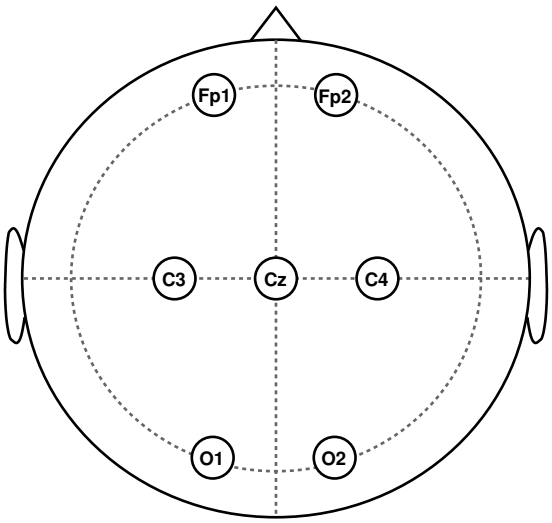



**Figure 5.10** Placement of the EEG electrodes on the scalp according to the recording 10-20 system.

spectively given by

$$\begin{aligned} \mathcal{S}_{y_n} &= \mathcal{F}\big(\mathcal{C}_{y_n y_n}(\delta)\big) = \mathcal{F}\big(E\{y_n(k)y_n^*(k-\delta)\}\big) \\ \breve{\mathcal{S}}_{y_n} &= \mathcal{F}\big(\mathcal{P}_{y_n y_n}(\delta)\big) = \mathcal{F}\big(E\{y_n(k)y_n(k-\delta)\}\big). \end{aligned} \tag{5.25}$$

Also see Equation (2.18) and discussion in Section 2.1.6.

### 5.4.3 Case Study 1 – EOG extraction

The 'EYEBLINK' dataset contained the EEG recordings contaminated with eye blink artifact as well as line noise. The recorded EEG and EOG signals are plotted in Figure 5.11(a), where the effect of the EOG activity is pronounced in the frontal lobe (Fp1 and Fp2 channels), with the effect diminishing with an increase in the distance of the electrodes to the eyes. The effect of the line noise is also visible on the occipital O1 and O2 channels.

The H-H T-F spectrogram (Fig 5.11(b)) describes the frequency changes of the ensemble average of the 6 EEG channels over the recording period. In correspondence with the time plot, the EOG artifacts are visible (with a duration of around 1 seconds); constant frequency components are seen around the 50 Hz range due to the line noise. Note that due to the low sampling rate of the recording device, the 50 Hz frequency component is not well defined in the T-F analysis and results in scattering of frequency components between 40 Hz-60 Hz.

The complex EEG signals formed using (5.24) were processed using the K-cBSE algorithm with the value of $\mu = \{5, 0.09\}$ and $\beta = \{-1, 1\}$ for the consecutive iterations

and $\alpha = 0.975$. The choice of value for $\beta$ ensures that the line noise is initially extracted, followed by the EOG components in the second iteration. The normalised kurtosis values of the original real-valued EEG signals and the extracted EEG signals are given in Tables 5.2 and 5.3.

The order of the extracted complex signals were as expected, with the first extracted source $y_1(k)$ (line noise) being sub-Gaussian and $y_2(k)$ (EOG) super-Gaussian. The imaginary component of $y_1(k)$ had the smallest kurtosis, and was automatically chosen as the extracted line noise source, while the near zero kurtosis of the real component $\Re\{y_1(k)\}$ indicates an EEG source. Also, both components of the second extracted source, having a high kurtosis value, were considered as the extracted EOG sources. Figure 5.11(c) shows the T-F plots of the imaginary components of the first extracted signal $y_1(k)$ where the presence of the power line artifact is seen, while in Figure 5.11(d) the T-F plot of the real and imaginary components of $y_2(k)$ is shown where the frequency components of the EOG artifacts are seen.

The power spectrum and pseudo-power spectrum of the complex EOG signal is next considered, constructed in a similar manner to that in (5.24); the extracted sources $y_1(k)$ and $y_2(k)$ are depicted in Figure 5.11(e). Notice that the distribution of power $\mathcal{S}_{\text{EOG}}$ and pseudo-power $\breve{\mathcal{S}}_{\text{EOG}}$ is concentrated respectively in the frequency range (0-5) Hz and 50 Hz. The spectrum $\mathcal{S}_{y_1}$ and pseudo-spectrum $\breve{\mathcal{S}}_{y_2}$ of the first extracted source can be seen to contain around 0 dB of power for a frequency of 50 Hz, while having an average power of -40 dB in the (0-5) Hz frequency range.

These results can also be seen by comparing the frequency components of the recorded EEG mixture and extracted artifactual sources around the 50 Hz range, shown in Figure 5.11(f). While the presence of the power line artifact is evident in all recorded channels, after the extraction procedure the 50 Hz frequency component is only present in $\Im\{y_1(k)\}$. Likewise, the spectra of $y_2(k)$ illustrate the diminished effect of the line noise source with a power of -20 dB, while retaining the frequency components of the EOG in the low frequency range. To quantify the observed results, the correlation coefficient between the recorded EOG's PSD and pPSD and those of the extracted sources were calculated [115] and presented in Table 5.3. For the extracted source $y_1(k)$ these values were respectively 0.23 and 0.28, whereas for the source $y_2(k)$ they were 0.97 and 0.98. The correspondence of the results between the power and pseudo-power spectra demonstrate the effectiveness of the proposed methodology in extracting artifacts in the complex domain.

**Table 5.2** Normalised kurtosis values of the recorded EEG/EOG signals in real- and complex-valued form

| | Set | | |
|---|---|---|---|
| Electrode | 'EYEBLINK' | 'EYEROLL' | 'EYEBROW' |
| Fp1 | 7.75 | 3.36 | 7.42 |
| Fp2 | 6.48 | 2.26 | 7.50 |
| C3 | -0.29 | -0.09 | -0.50 |
| C4 | 1.15 | 1.25 | 1.53 |
| O1 | -0.26 | 0.83 | -0.60 |
| O2 | -0.96 | -0.68 | -0.95 |
| vEOG | 7.75 | 4.84 | 10.87 |
| hEOG | -0.15 | 2.39 | -0.33 |
| $x_1(k)$ | 7.03 | 2.64 | 6.12 |
| $x_2(k)$ | 0.10 | 0.45 | -0.01 |
| $x_3(k)$ | -0.92 | -0.46 | -0.93 |

**Table 5.3** Normalised kurtosis values of the extracted artifacts, and the correlation coefficient of the power and pseudo-power spectra respectively with the spectra of the recorded EOG

| | | | | Spectra corr. | |
|---|---|---|---|---|---|
| Set | Signal | $K_c$ | $K_R(\Re, \Im)$ | PSD | pPSD |
| 'EYEBLINK' | $y_1(k)$ | -1.22 | -0.09, -1.24 | 0.23 | 0.18 |
| | $y_2(k)$ | 7.39 | 7.51, 5.16 | 0.97 | 0.98 |
| 'EYEROLL' | $y_1(k)$ | -1.17 | -1.20, -0.03 | 0.08 | 0.18 |
| | $y_2(k)$ | 3.06 | 3.52, 2.73 | 0.82 | 0.82 |
| 'EYEBROW' | $y_1(k)$ | -1.01 | -0.73, -1.13 | 0.13 | 0.11 |
| | $y_2(k)$ | 4.51 | 5.43, 6.38 | 0.76 | 0.79 |

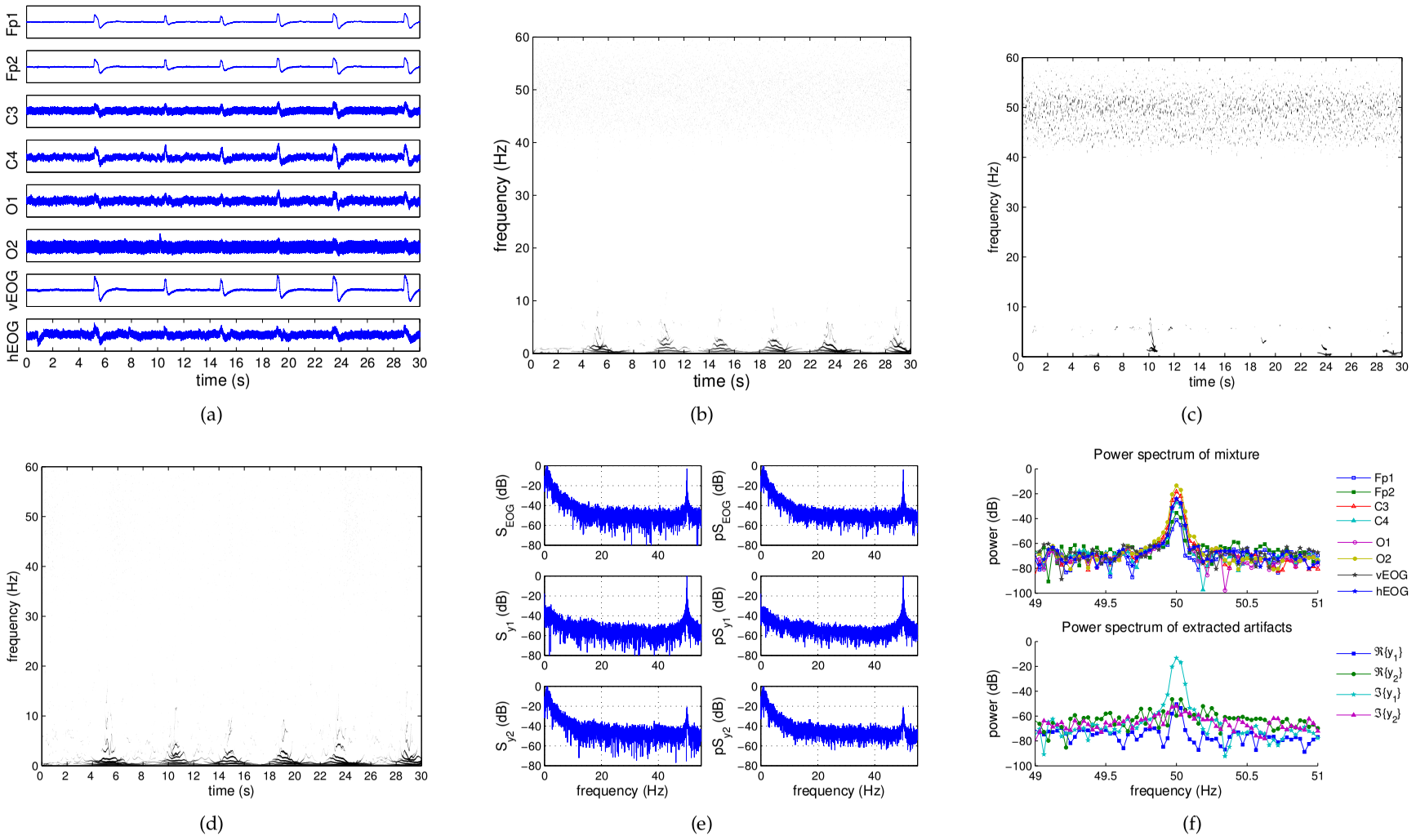



**Figure 5.11** Recorded and extracted artifacts from the 'EYEBLINK' set. (a) Recorded EEG signals from the 'EYEBLINK' set. (b) The Hilbert-Huang time-frequency plot of the recorded EEG signals. (c) The Hilbert-Huang time-frequency plot of the extracted line noise $\Im\{y_1(k)\}$. (d) The Hilbert-Huang time-frequency plot of the extracted EOG $\Re\{y_2(k)\}, \Im\{y_2(k)\}$. (e) The power spectra (S) and pseudo-spectra (pS) of the recorded EOG, and the extracted signals $y_1(k)$ and $y_2(k)$. (e) The power spectra (S) and pseudo-spectra (pS) of the recorded EOG, and the extracted signals $y_1(k)$ and $y_2(k)$. (f) Frequency components of the recorded EEG signals and the extracted artifacts around the 50Hz frequency range. After extraction, the power line noise is contained in $\Im\{y_1\}$.

### 5.4.4 Case Study 2 – Eye muscle artifact extraction

The 'EYEROLL' dataset had contained artifacts from round movement of the eye during the recording session with EOG activity from eye blinks, shown in Figure 5.12(a) and kurtosis values given in Table 5.2.

The resultant electrical activity from the artifacts were recorded using the vEOG and hEOG channels, with EOG activity seen on the vEOG channel at time instants 5s, 13s, 17s, 23s, 25s and 29s, and eye muscle activity present more clearly on the hEOG channel with a duration of around 2s. The eye muscle artifact was present in all six EEG channels, while the EOG artifact is strong on the Frontal lobe electrodes and the effect of the power line noise is seen more strongly on the central and occipital lobe electrodes. The H-H T-F analysis of Figure 5.12(b) illustrates the presence of frequency components up to 10 Hz, as well as scattered frequencies belonging to the 50 Hz power line noise.

In the extraction procedure, the step-size of the K-cBSE algorithm was $\mu = \{5, 0.2\}$ and $\beta = \{-1, 1\}$, while $\alpha = 0.975$. The T-F analysis of the extraction are illustrated in Figure 5.12(c)–(d), and the kurtosis values of the complex-valued extracted signals and their real and imaginary components given in Table 5.3.

The real component of the first extracted source, $\Re\{y_1(k)\}$, having the smallest kurtosis of $K_c(\Re\{y_1) = -1.20$ contained the power line noise artifact. The eye muscle activity and EOG artifacts were collectively extracted using the real and imaginary components of the second extracted source $y_2(k)$. The five instances of the eye muscle activity and the EOG can be detected in Figure 5.12(d), while the lack of power line noise frequency components in the 50 Hz range is visible.

These results were also confirmed based on the power spectra of the recorded artifacts and the extracted sources, given in Figure 5.12(e). While the PSD and pPSD of the complex-valued $y_1(k)$ contained the 50 Hz components, these were suppressed to -40 dB in the spectra of $y_2(k)$. The frequency components of the mixture channels and extracted artifacts in the 50 Hz range also showed that the line noise artifact was successfully removed (see Figure 5.12(f)). Conversely, the spectral components pertaining to the eye muscle and EOG artifacts are present in the PSD and pPSD of $y_2(k)$ corresponding to the (0-10) Hz range of the PSD and pPSD of the complex-valued EOG. The correlation coefficient between the PSD spectra of the complex-valued recorded EOG channel and extracted source $y_2(k)$ is 0.82, while the correlation between the pPSD spectra was 0.82; these values were respectively 0.08 and 0.18 for $y_1(k)$.

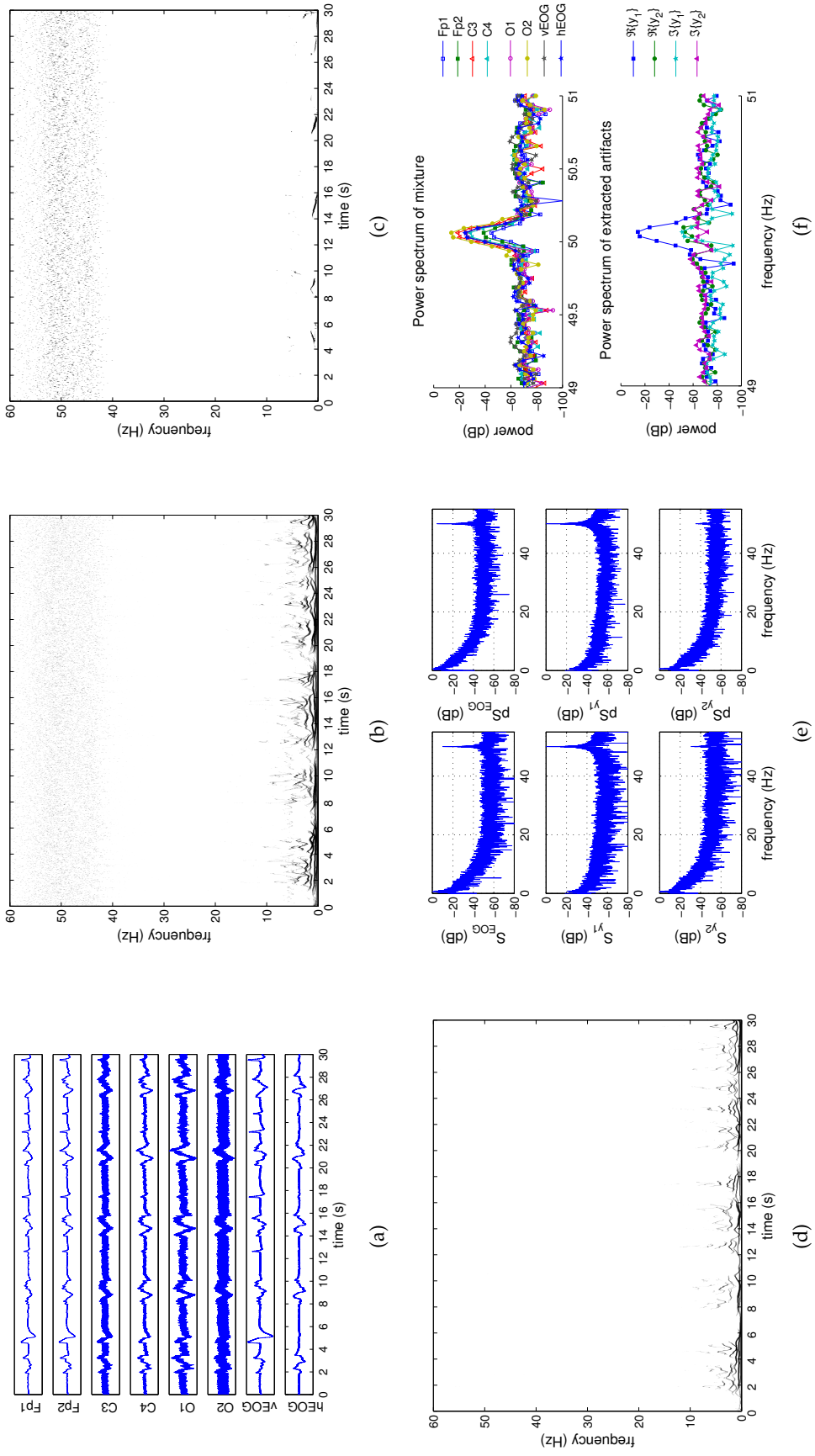



**Figure 5.12** Recorded and extracted artifacts from the 'EYEROLL' set. (a) Recorded EEG signals from the 'EYEROLL' set. (b) The Hilbert-Huang time-frequency plot of the recorded EEG signals. (c) The Hilbert-Huang time-frequency plot of the extracted line noise $\Re\{y_1(k)\}$. (d) The Hilbert-Huang time-frequency plot of the extracted EOG $\Re\{y_2(k)\}$, $\Im\{y_2(k)\}$. (e) The power spectra (S) and pseudo-spectra(pS) of the recorded EOG, and the extracted signals $y_1(k)$ and $y_2(k)$. (f) Frequency components of the recorded EEG signals and the extracted artifacts around the 50Hz frequency range. After extraction, the power line noise is contained in $\Re\{y_1\}$.

### 5.4.5 Case Study 3 – EMG extraction

In the 'EYEBROW' set, the EEG mixture was heavily contaminated with EMG artifacts from raising the eyebrows, and are shown in Figure 5.13(a) with kurtosis values given in Table 5.2.

The EMG signals were recorded using the vEOG and hEOG electrodes, with the effect more prominent on the vEOG recording. All EEG channels were affected by the artifact, though this is not clearly visible in the occipital lobe channels due to the strong presence of power line noise. In the T-F domain (Figure 5.13(b)) the EMG frequency range had a large span containing both low and high frequency components, present in the duration of the raising of the eyebrows and lasting for around 2s. In addition, the 50 Hz frequency component cloud reflecting the power line noise can also be seen.

The extraction of the artifacts was performed using the K-cBSE algorithm (5.15) with step-size $\mu = \{2, 0.2\}$, $\beta = \{-1, 1\}$ and $\alpha = 0.975$. As shown in Figure 5.13(c) and Figure 5.13(d), the algorithm successfully extracted the power line noise as the imaginary component of the first extracted signal $y_1(k)$ and the EMG signal as the real and imaginary components of the second extracted signal $y_2(k)$. From the T-F plot of $y_2(k)$ in Figure 5.13(d), the complete EMG frequency component range was successfully extracted, with power line noise frequency components not present.

Considering the power spectra $\mathcal{S}_{\text{EMG}}$ and pseudo-power spectra $\breve{\mathcal{S}}_{\text{EMG}}$ in Figure 5.13(e), the spectral distribution of the power and pseudo-power spectral density were strong in the (0-10) Hz range with an amplitude of around -10 dB and in the (20-40) Hz range, though having a much lower value. In addition, a single spike at 50 Hz of amplitude -10 dB indicates the presence of power line noise. After the extraction, the power line noise was contained in the spectra of the $y_1(k)$ while the (0-10) Hz and (20-40) Hz frequency components were present in the PSD and pPSD of $y_2(k)$.

For the 'EYEBROW' set, the spectra correlation coefficients between $\mathcal{S}_{\text{EMG}}$ and $\breve{\mathcal{S}}_{\text{EMG}}$ and those of $y_1(k)$ and $y_2(k)$ were respectively $\{0.13, 0.11\}$ and $\{0.76, 0.80\}$. Also, the 50 Hz frequency range for the contaminated mixture and the extracted artifacts are shown in Figure 5.13(f). It can be seen that after the extraction procedure, the 50 Hz component is contained in $\Im\{y_1(k)\}$, while in comparison to the EOG and eye muscle extracted components from the 'EYEBLINK' and 'EYEROLL' studies (see Figure 5.11(f) and Figure 5.12(f)), components $\Re\{y_2(k)\}$ and $\Im\{y_2(k)\}$ had a higher power level in this range, reflecting the wider frequency range of the EMG artifact.

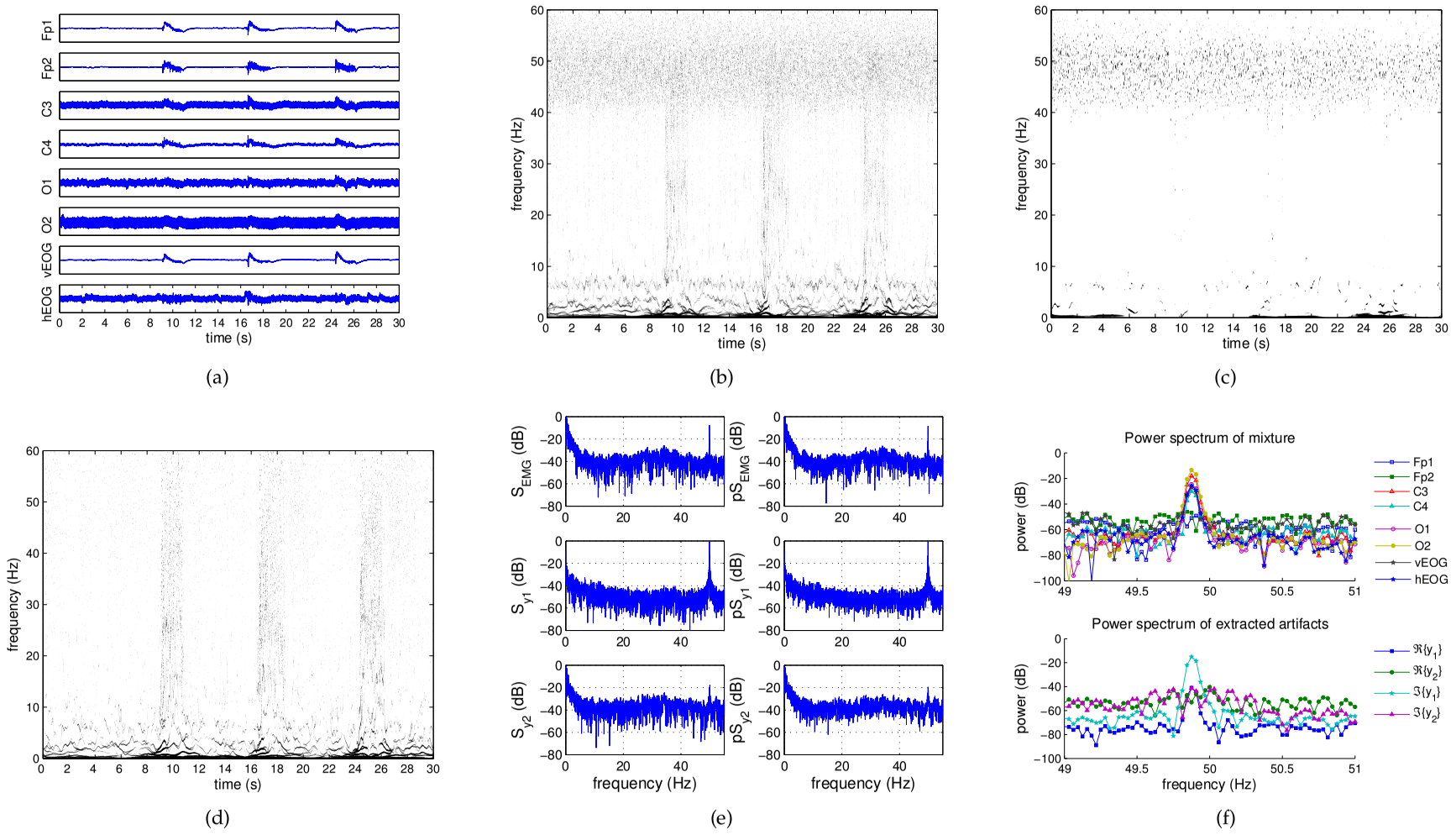



**Figure 5.13** Recorded and extracted artifacts from the 'EYEBROW' set. (a) Recorded EEG signals from the 'EYEBROW' set. (b) The Hilbert-Huang time-frequency plot of the recorded EEG signals. (c) The Hilbert-Huang time-frequency plot of the extracted line noise $\Im\{y_1(k)\}$. (d) The Hilbert-Huang time-frequency plot of the extracted EMG $\Re\{y_2(k)\}, \Im\{y_2(k)\}$. (e) The power spectra (S) and pseudo-spectra (pS) of the recorded EMG, and the extracted signals $y_1(k)$ and $y_2(k)$. (f) Frequency components of the recorded EEG signals and the extracted artifacts around the 50Hz frequency range. After extraction, the power line noise is contained in $\Im\{y_1\}$.

## 5.5 Summary

Blind source extraction of the generality of complex-valued signals based on the degree of non-Gaussianity and from noisy mixtures has been addressed. A cost function based on the normalised kurtosis has been utilised to perform blind extraction, and the corresponding online algorithm (K-cBSE) has been derived. The existence and uniqueness of the solutions have been discussed and variable step-size variants of the algorithm have been addressed.

It has been shown that the algorithm is robust to the degree of noncircularity of the additive noise and the success of the algorithm over increasing noise levels has been demonstrated. Simulations in noise-free and noisy environments illustrate the successful performance of the algorithm in the extraction of both circular and non-circular signals, while the extraction of EOG and EMG artifacts from recorded EEG signals in real-time demonstrate a practical application for the proposed methodology.

## 5.A Appendix: Update of $\epsilon(k)$ for the GNGD-type complex BSE

The gradient descent update for the regularisation parameter $\epsilon(k)$ is written as

$$\epsilon(k+1) = \epsilon(k) - \rho \nabla_\epsilon \mathcal{J}\big|_{\epsilon=\epsilon(k-1)}$$

and the gradient derived as follows. Defining the adaptive step-size in (5.18) as

$$\upsilon(k) \triangleq \frac{\mu}{|\phi\big(y(k)\big)|^2 \cdot \|\mathbf{x}(k)\|_2^2 + \epsilon(k)}$$

the gradient $\nabla_\epsilon \mathcal{J}$ is given by

$$\nabla_\epsilon \mathcal{J} = \big(\nabla_{\mathbf{w}^*} \mathcal{J}\big)^T \cdot \frac{\partial \mathbf{w}^*(k)}{\partial \upsilon(k-1)} \cdot \frac{\partial \upsilon(k-1)}{\partial \epsilon(k-1)} \tag{5.26}$$

where

$$\begin{aligned}\frac{\partial \mathbf{w}^*(k)}{\partial \upsilon(k-1)} = \frac{\partial \mathbf{w}^*(k)}{\partial \upsilon(k-1)} - \phi^*\big(y(k-1)\big)\mathbf{x}^*(k-1) - \\ \frac{\partial \phi^*\big(y(k-1)\big)}{\partial \upsilon(k-1)} \upsilon(k-1)\mathbf{x}^*(k-1) \\ \approx -\phi^*\big(y(k-1)\big)\mathbf{x}^*(k-1)\end{aligned}$$

and only the driving term of the recursion is considered, and

$$\frac{\partial \upsilon(k-1)}{\partial \epsilon(k-1)} = \frac{-\mu}{\big[|\phi\big(y(k-1)\big)|^2 \cdot \|\mathbf{x}(k-1)\|_2^2 + \epsilon(k-1)\big]^2}.$$

While the derivative in (5.26) is calculated according to the $\mathbb{CR}$ calculus, $\epsilon(k)$ is real-valued and so only the real component of the $\mathbb{R}^*$–derivative in (5.26) is required. This leads to the update equation given in (5.19).

# Chapter 6

# A Fast Algorithm for Blind Extraction of Smooth Complex Sources

## 6.1 Introduction

Smoothness is a fundamental signal property, and can be modelled based on the behaviour of gradients of data vectors. Employing smoothness can also aid BSS and BSE as, for instance, in electroencephalography (EEG), artifacts coming from eye muscles are smoother than the background EEG. An algorithm for BSE of real-valued smooth signals in the time-domain was introduced in [124], and an implementation in the frequency domain was recently proposed in [125]. Processing in the time domain has its merits in retaining the signals in their original form and avoiding extra computations. In addition, performing the Fourier Transform using a block-based approach results in the inadvertent smoothing of the data.

A blind extraction algorithm for complex-valued signals in time domain is proposed. In a manner similar to [124], a fast converging algorithm is introduced by using a fixed-point type update based on the existing complex FastICA algorithm [41, 78]. Such an extraction algorithm can thus be seen as a constrained version of the complex FastICA algorithm, and as shown in the derivation, it simplifies into the unconstrained complex FastICA when the smoothness constraint is removed. Original contributions in this chapter is the use of the Sobolev norm to define smoothness in the complex domain, where lexicographic ordering is not permitted, as well as the use of $\mathbb{CR}$ calculus for the optimisation solution to the smoothness constraint generalised complex FastICA.

The performance is verified on the removal of artifacts from real-world EEG recordings. It is shown that several types of eye movement artifacts can be successfully removed using the proposed algorithm, thus making it attractive for brain computer interface (BCI) applications. This has a number of applications, as by removing the artifact related sources, further processing on the remaining pure EEG signals is made possible in real-time.

## 6.2 Smoothness-based Blind Source Extraction

### 6.2.1 The Concept of Smoothness in $\mathbb{C}$

The mathematical concept of a smooth function is based on differentiability. Consider the Sobolev space $W^{p,q} \subset \mathbb{R}^N$ defined as the space where the $p$-th power of a function $f \in W^{p,q}$ together with its first $q$-th derivatives are integrable [126]. The norm is then defined as

$$\|f\|_{W^{p,q}} = \left( \sum_{i=0}^{q} \|D^{(i)} f\|_p^p \right)^{1/p} \tag{6.1}$$

where $D^{(i)} f$ denotes the $i$th derivative of $f$. Due to the duality between $\mathbb{C}$ and $\mathbb{R}^2$ [54], the above definition can also be adopted for complex-valued functions. The Sobolev norm for the space $W^{2,1}$ is utilised, where only the second power of the function and its first derivative are considered. Taking an arbitrary upper bound of the ratio between the Sobolev and Euclidean norms of the function $f$ yields

$$\frac{\|f\|_{W^{2,1}}^2}{\|f\|_2^2} = \frac{\|D^{(1)} f\|_2^2}{\|f\|_2^2} \leq \rho_s \tag{6.2}$$

where $\rho_s$ is the upper bound of the ratio, also referred to as the smoothness factor. For a discrete signal $z(k)$, a simplified form is given by

$$E\{|\Delta z(k)|^2\} - \rho_s E\{|z(k)|^2\} \leq 0 \tag{6.3}$$

where $\Delta z(k) = z(k) - z(k-1)$; a geometric interpretation is given in Figure 6.1. In a similar fashion to the real-valued case, Equation (6.3) models a complex-valued signal with a slow varying temporal profile as a smooth signal. Intuitively, a complex-valued signal $z(k)$ is smooth if the variance of the difference between consecutive samples is less than a pre-defined fraction of the variance of the signal itself. This can also be interpreted as measuring the variation in the gradient of the signal[1].

[1] In $\mathbb{C}$ relationships such as '$>$' and '$<$' do not apply and it is necessary to resort to the duality between $\mathbb{R}^2$ and $\mathbb{C}$, and to use so called lexicographic ordering.

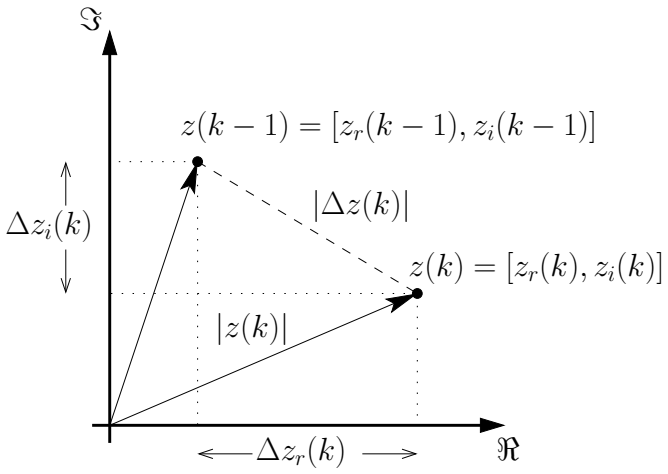



**Figure 6.1** Geometric interpretation of the smoothness definition given in (6.3)

Notice that the smoothness definition based on the Sobolev norm of $W^{2,1}$ is based on the covariances $\mathcal{C}_{zz}(0)$ and $\mathcal{C}_{zz}(1)$, that is, the covariances of lag zero and one. This can be observed by expanding the terms in (6.3) such that

$$\begin{aligned} E\big\{\big(z(k)-z(k-1)\big)\big(z(k)-z(k-1)\big)^*\big\} - \rho_s E\{z(k)z^*(k)\} \leq 0 \\ E\{z(k)z^*(k)\} + E\{z(k-1)z^*(k-1)\} \\ - 2E\{z(k)z^*(k-1)\} - \rho_s E\{z(k)z^*(k)\} \leq 0, \end{aligned} \tag{6.4}$$

and based on the definition in Equation (2.18),

$$(2-\rho_s)\mathcal{C}_{zz}(0) - 2\mathcal{C}_{zz}(1) \leq 0. \tag{6.5}$$

Alternatively, consider the definition (6.2) for $z(k) = z_r(k) + \jmath z_i(k)$, expressed in its dual form $z^R(k) = [z_r(k),\ z_i(k)]^T \in \mathbb{R}^2$. Then,

$$\begin{aligned} &E\big\{\ \langle \Delta z^R(k), \Delta z^R(k)\rangle\ \big\} - \rho_s E\big\{\ \langle z^R(k), z^R(k)\rangle\ \big\} \leq 0 \\ &E\{\Delta z_r^2(k)\} + E\{\Delta z_i^2(k)\} - \rho_s\big(E\{z_r^2(k) + z_i^2(k)\}\big) \leq 0 \end{aligned} \tag{6.6}$$

where the symbol $\langle\cdot,\cdot\rangle$ denotes the inner product.

### 6.2.2 The BSE Problem

Consider an observation $\mathbf{x}(k) \in \mathbb{C}^N$ formed from the linear weighted combination of latent sources $\mathbf{s}(k) \in \mathbb{C}^{N_s}$, given by

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) \tag{6.7}$$

where $\mathbf{A} \in \mathbb{C}^{N\times N_s}$ is the mixing matrix, and $N_s$ the number of sources. The sources are assumed independent and the observation mixture is whitened prior to processing.

The aim is to find a demixing vector $\mathbf{w}$ that will recover one of the sources, given by

$$y(k) = \mathbf{w}^H \mathbf{x}(k). \tag{6.8}$$

Following the standard BSS methodology [12, 41], this can be achieved by maximising the non-Gaussianity of $y(k)$ reflected in the cost function

$$\mathcal{J}_N(\mathbf{w}, \mathbf{w}^*) = E\big\{G(|\mathbf{w}^H \mathbf{x}|^2)\big\} \tag{6.9}$$

where $G$ is a nonlinearity used to approximate the associated negentropy, and for generality, $\mathcal{J}_N$ is expressed using both the coordinates $\mathbf{w}$ and $\mathbf{w}^*$.

To ensure that components with certain smoothness characteristics are extracted, further constraints are imposed on $\mathcal{J}_N$. Based on the definition in (6.3), for the removed source $y(k)$ given in (4.3), the smoothness measure becomes

$$\mathcal{J}_S(\mathbf{w}, \mathbf{w}^*) = \beta\big(E\{|\mathbf{w}^H \Delta\mathbf{x}(k)|^2\} - \rho_s E\{|\mathbf{w}^H \mathbf{x}(k)|^2\}\big) \tag{6.10}$$

The constant $\beta = \{-1, 1\}$ gives us a degree of freedom in dealing with smooth sources, for instance $\beta = -1$ the extraction of the most non-smooth source will be achieved.

Thus, the optimisation problem of BSE of latent sources based on the smoothness constraint (S-cBSE) can be stated as

$$\begin{aligned} \mathbf{w}_{opt} = \arg \max_{\|\mathbf{w}\|_2^2 = 1} & \mathcal{J}_N(\mathbf{w}, \mathbf{w}^*) \\ \text{subject to } & \mathcal{J}_S(\mathbf{w}, \mathbf{w}^*) \leq 0, \end{aligned} \tag{6.11}$$

where after every step, the demixing $\mathbf{w}$ is normalised to avoid spurious solutions.

Removing the smoothness constraint in the optimisation problem (6.11) results in the formulation of the cost function for negentropy based ICA in the complex domain [41, 78]. In [41], the authors derive the standard complex FastICA (c-FastICA) which assumes second-order circular sources. The generalised complex FastICA (nc-FastICA) algorithm [78] is instead derived for the generality of complex sources. An overview of the c-FastICA and nc-FastICA algorithms is given in Appendix D, where the formulation of the two algorithms and discussions on their convergence behaviour are provided.

The difference in these assumptions is evident in the derivation of the two algorithms using the augmented Newton method (see Equation (B.31) and (B.32)). The derivation of the S-cBSE algorithm will also be based on the generalised complex FastICA algorithm, and is thus capable of processing both proper and improper sources. For comparison, the derivation of the S-cBSE algorithm with the circularity assumption (that is, based on the c-FastICA algorithm) is also provided in the Appendix at the end of this chapter.

To solve the optimisation problem in (6.11), the method of Lagrangian multipliers is employed. The extrema of the Lagrangian can be found using the Newton method, resulting in faster convergence to the solution; this method has been shown to be stable for a related unconstrained problem in the complex domain [78]; a detailed proof of the derivation is given in Appendix 6.A at the end of this chapter. The Newton based optimisation of the Lagrangian is performed as

$$\begin{aligned} \Delta\mathbf{w} &= \left(\mathbf{H}_{\mathbf{w}\mathbf{w}^*} - \mathbf{H}_{\mathbf{w}^*\mathbf{w}^*}\mathbf{H}_{\mathbf{w}^*\mathbf{w}}^{-1}\mathbf{H}_{\mathbf{w}\mathbf{w}}\right)^{-1} \cdot \left(\mathbf{H}_{\mathbf{w}^*\mathbf{w}^*}\mathbf{H}_{\mathbf{w}^*\mathbf{w}}^{-1}\frac{\partial\mathcal{L}}{\partial\mathbf{w}} - \frac{\partial\mathcal{L}}{\partial\mathbf{w}^*}\right) \\ \Delta\lambda &= \nabla_\lambda\mathcal{L} \\ \mathbf{w}(k+1) &\leftarrow \mathbf{w}(k+1)/\|\mathbf{w}(k+1)\|_2 \end{aligned} \tag{6.12}$$

where $\mathcal{L}(\mathbf{w}, \mathbf{w}^*, \lambda)$ is the Lagrangian function, $\lambda$ is the Lagrangian multiplier and the $\mathbf{H}$ matrices are the Hessians of $\mathcal{L}$.

To extract successive smooth (non-smooth) sources, a deflationary orthogonalisation process using the Gram-Schmidt method is performed after each iteration of the extraction algorithm in (6.12). While this allows for unambiguous extractions, errors in the extraction and thus deflation process can accumulate, resulting in decreased performance over consecutive extractions[2]. The deflation procedure for the $i$th demixing vector can be compactly written as

$$\mathbf{w}_i(k+1) \leftarrow \mathbf{w}_i(k+1) - \check{\mathbf{W}}\check{\mathbf{W}}^H\mathbf{w}_i(k+1) \tag{6.13}$$

where $\check{\mathbf{W}} = [\mathbf{w}_1(k+1),\ \ldots,\ \mathbf{w}_i(k+1)]$.

## 6.3 Performance Benchmarking

To illustrate the performance of the proposed algorithm, sub-Gaussian and super-Gaussian complex-valued sources with different degrees of noncircularity were used. The smoothness degree of the sources

$$\rho_s(z) = \frac{E\{|\Delta z(k)|^2\}}{E\{|z(k)|^2\}} \tag{6.14}$$

was measured using (6.3), while the degree of circularity was assessed using the measure $r$ given in Equation (2.17) as the ratio of the absolute value of the pseudo-variance $\tau_z^2 = E\{z^2\}$ to the variance $\sigma_z^2 = E\{|z|^2\}$ of the source, as described in [80]. Note that the value $r = 0$ denotes a second-order circular source, while $r = 1$ indicates a highly noncircular source.

[2]In practical applications, this usually does not pose a pose a problem, as only 1-2 smooth sources (artifacts) are of interest.

The performance of the algorithm was measured using the Performance Index (PI) expressed in Equation (4.30), where a value of less than -20 dB indicates good performance. Four complex-valued sources of 5000 samples were mixed using a randomly generated $4 \times 4$ mixing matrix to form the observed mixtures. The magnitude of the sources are shown in Figure 6.2 and the signal properties given in Table 6.1, where all sources are highly improper. The mixture was whitened and the latent sources were extracted using the S-cBSE algorithm (6.12). In the first experiment, the value of $\beta = 1$, $\rho_s = 0.9$, $\lambda = 1$ and $\mu_\lambda = 0.01$. As the signals were synthetically generated, the value of $\rho_s$ was chosen based on measurements of the signal smoothness. The nonlinearity $G(z) = \log\cosh(z)$ ensured that the negentropy of both sub-Gaussian and super-Gaussian sources were sufficiently approximated for maximisation.

The performance of the S-cBSE algorithm based on the standard complex FastICA, given in Equation (6.22), is first considered. Figure 6.3 shows the performance of the algorithm, where the simplified algorithm did not have adequate performance, and was not suitable for the extraction of improper sources. This is in agreement with the results in [78], where the non-constrained c-FastICA algorithm did not provide suitable separation performance.

Figure 6.2 shows the sources which were successfully extracted based on the smoothness criterion. For comparison, the measured smoothness factors for the extracted sources (denoted by $\widehat{\rho}_s$) are given in Table 6.1. Notice that as $\{\rho_s(s_3), \rho_s(s_1)\} \leq 0.9$ it is expected that only sources $s_3(k)$ and $s_1(k)$ were to be extracted, however, the algorithm also successfully extracted the subsequent sources $s_2(k)$ and $s_1(k)$. This can be attributed to the strong non-Gaussianity condition in (6.9), which was sufficient for successful extraction. The performance index at each iteration (Figure 6.4) shows that the algorithm achieved convergence with a PI of around -30 dB for the source estimates $y_1(k)$, $y_2(k)$ and $y_4(k)$ in under 10 iterations, while source estimate $y_3(k)$ achieved a PI of under -35 dB in 19 iterations. Alternatively, expressed in terms of the signal-to-interference ratio (SIR), the values for the consecutive extractions were respectively 29.81 dB, 23.23 dB, 21.76 dB and 25.68 dB.

In the next experiment, the objective was to extract the non-smooth sources, for which $\beta = -1$ and $\rho_s = 2$. The values of the other parameters were set empirically to $\lambda = 20$ and $\mu_\lambda = 1$ and the nonlinear function $G$ was kept as before. The sources were extracted in the order of increasing smoothness, with the performance indices over the extraction process plotted in Figure 6.4. The PI value for the source estimate $y_1(k)$ was around -30 dB while $y_2(k)$ achieved a limit cycle with a varying PI of around -22 dB to -30 dB. Source estimate $y_3(k)$ initially converged but diverged after 3 iterations and $y_4(k)$ only achieved a PI of around -20 dB. While source $s_4(k)$ was the only non-smooth signal according to the value set for $\rho_s$, source $s_2(k)$ was also successfully extracted due to the close proximity to the smoothness criterion. However, note that sources

**Table 6.1** Source properties for extraction simulations, $\widehat{\rho_s}$ is the estimated smoothness measure.

| | | | $\beta = 1$ | $\beta = -1$ |
|---|---|---|---|---|
| Source | $r$ | $\rho_s$ | $\widehat{\rho_s}$ | $\widehat{\rho_s}$ |
| $s_1(k)$ | 0.9997 | 0.1154 | 0.1200 | 0.0193 |
| $s_2(k)$ | 0.9865 | 1.4771 | 1.4745 | 1.4782 |
| $s_3(k)$ | 0.9998 | 0.0148 | 0.0150 | 0.1136 |
| $s_4(k)$ | 0.9995 | 2.0214 | 2.0219 | 2.0204 |

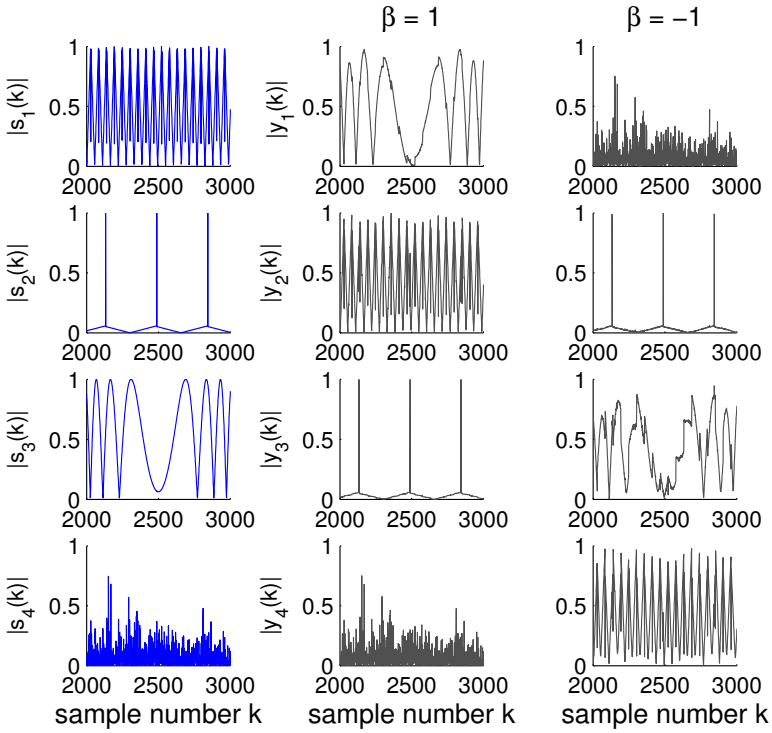



**Figure 6.2** Performance of the algorithm (6.12) in the extraction of smooth ($\beta = 1$) and non-smooth ($\beta = -1$) sources

$s_1(k)$ and $s_3(k)$ were not successfully extracted due to the disparity between the values of $\rho_s(s_1)$ and $\rho_s(s_3)$ to $\rho_s = 2$ as set for this experiment. The SIR for the consecutive extractions were respectively 23.87 dB, 27.45 dB, 3.93 dB and 3.87 dB.

## 6.4 Artifact Extraction from EEG

The S-cBSE algorithm was next utilised to extract power line noise, biological eye blink (electrooculogram, EOG), and eye muscle activity (electromyogram, EMG) artifacts, common in EEG recordings. The aim was to condition the contaminated recordings so that further processing, such as those in real-time BCI, can be performed. The test

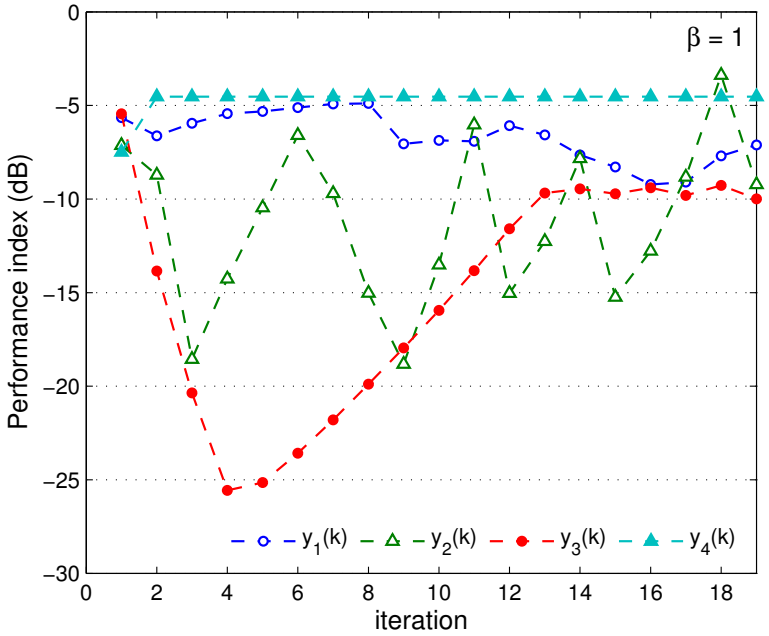



**Figure 6.3** Performance of the S-cBSE algorithm based on the standard complex FastICA (6.22) for the extraction of smooth ($\beta = 1$) sources

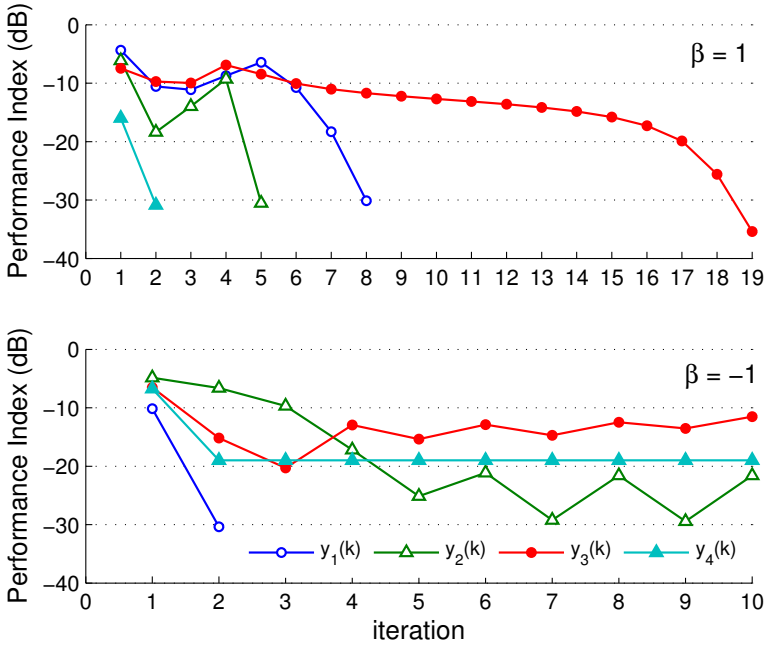



**Figure 6.4** Performance of the algorithm (6.12) in the extraction of smooth ($\beta = 1$) sources and non-smooth ($\beta = -1$) sources.

EEG signal was recorded at the Imperial College Smart Environment Lab (SEL), with the electrodes were placed according to the 10-20 system at positions Fp1, Fp2, C3, C4, O1, O2 and the ground electrode was placed at Cz. The electrical activity from the EOG and EMG artifacts were recorded using the vEOG and hEOG channels, with electrodes placed around the eye. The recording lasted 30s and the data were sampled at a rate of 256 Hz. In the first study, the participants were asked to blink at random intervals while looking straight. In the second study, the instructions were to move the eyes in a vertical motion at random intervals.

The recorded EEG channels were combined into temporal complex-valued mixtures such that the real and imaginary components comprised symmetric EEG channels. In this manner, the cross-information due to the phase and magnitude relationship between pairs of symmetric electrodes was utilised by the extraction algorithm [127]. The complex EEG mixtures $\mathbf{x}(k)$ were generated as (see Figure 5.10 for electrode positions)

$$\begin{aligned} x_1(k) &= \text{Fp1}(k) + \jmath\, \text{Fp2}(k) \\ x_2(k) &= \text{C3}(k) + \jmath\, \text{C4}(k) \\ x_3(k) &= \text{O1}(k) + \jmath\, \text{O2}(k). \end{aligned} \tag{6.15}$$

In the EOG study, the algorithm (6.12) was used to extract two independent sources, and was initialised respectively with $\beta_1 = 1, \beta_2 = -1$, and $\rho_{s,1} = 0.01, \rho_{s,2} = 0.9$ for the first and second extraction steps, while the value of $\lambda = 80$ and $\mu_\lambda = 1$ for both steps. These values were deduced from prior information about both artifacts; the periodic power line noise was non-smooth[3], while the intermittent EOG activity was smooth in comparison with the pure EEG data.

The real and imaginary components of the complex-valued extracted signal $y(k)$ represent the actual real-valued latent sources. After the completion of each extraction stage, the smoothness of the real and imaginary components were measured, and the component matching the criterion was removed. The smoothness values for the extracted signals $y_1(k)$ and $y_2(k)$, and their respective real and imaginary components are given in Table 6.2. A qualitative assessment of the extraction was performed by comparing the power spectrum of the reference biological artifact and the power spectrum of the extracted artifacts, such as the EOG shown in the left column of Figure 6.5. The power spectrum of the raw EOG illustrates the presence of frequencies from 0 Hz-5 Hz and the power line activity at 50 Hz. The power spectrum of $\Im\{y_1(k)\}$ shows that the algorithm successfully extracted the EOG source, while attenuating the 50 Hz frequency. The 50 Hz source was contained within the real component of the second extracted source $\Re\{y_2(k)\}$, as seen from the corresponding power spectrum.

[3]This can be attributed to the low sampling rate, a limitation of the recording hardware.

**Table 6.2** Smoothness properties for extracted EEG artifacts. The rejected components are shown in bold font.

| Dataset | Source | $\rho_s$ | $\rho_s(\Re, \Im)$ |
|---|---|---|---|
| 'EOG' | $y_1(k)$ | 0.0274 | 0.2706, **0.0085** |
| | $y_2(k)$ | 1.2910 | **1.3179**, 0.8494 |
| 'EMG' | $y_1(k)$ | 0.7333 | **0.7748**, 0.2323 |
| | $y_2(k)$ | 0.1438 | **0.0142**, 0.1242 |

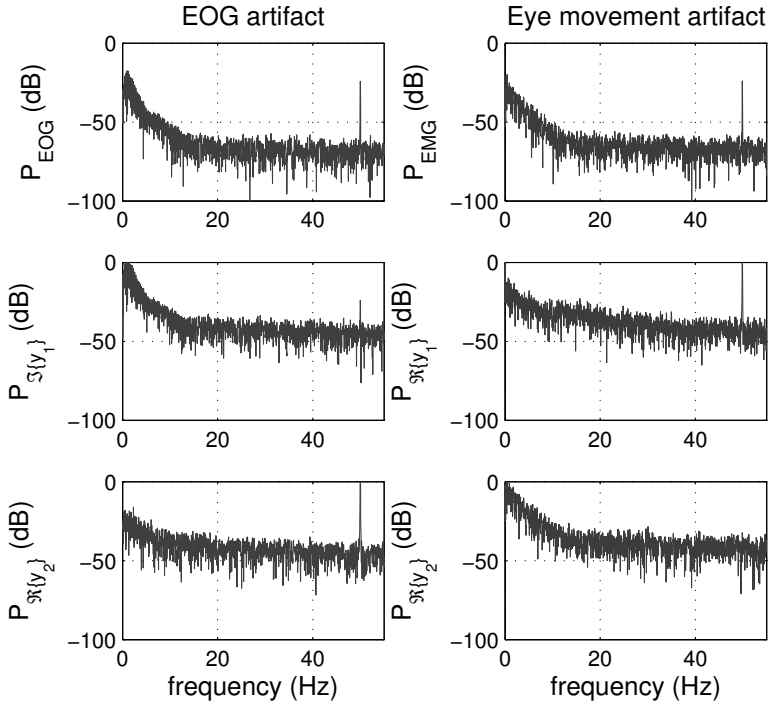



**Figure 6.5** *Left*: Power spectrum of the recorded EOG and the extracted artifacts, *Right*: Power spectrum of the EMG due to eye movement and the extracted artifacts.

For the EMG study, the S-cBSE algorithm was initialised such that $\beta_1 = -1, \beta_2 = 1$ and $\rho_{s,1} = 0.9, \rho_{s,2} = 0.05$. The parameters $\lambda_1 = 1, \lambda_2 = 10$ and $\mu_\lambda = 1$ for both extractions steps. The smoothness factor of the extracted sources and their respective components are given in Table 6.2, and the power spectrum associated with the recorded eye muscle activity and the extracted components is given in Figure 6.5. Observe that the real component of $y_1(k)$ contained the power line activity, while the real component of $y_2(k)$ represented the EMG activity.

## 6.5 Summary

An algorithm for complex blind source extraction (S-cBSE) based on a smoothness criterion has been introduced. The concept of smoothness has been defined for gen-

eral complex-valued signals and was employed to define a constrained cost function, based on the maximisation of non-Gaussianity. The fast convergence of the algorithm is inherited from FastICA, confirmed on benchmark data. Further, an application in the extraction of power line noise and biological artifacts from contaminated EEG recordings has been addressed.

## 6.A Appendix: Derivation of the S-cBSE Algorithm

First, note that due to the whiteness of $\mathbf{x}(k)$, the cost $\mathcal{J}_S$ in (6.10) can be expanded as

$$\begin{aligned}\mathcal{J}_S &= \mathbf{w}^H E\{\Delta\mathbf{x}\Delta\mathbf{x}^H\}\mathbf{w} - \rho_s \mathbf{w}^H E\{\mathbf{x}\mathbf{x}^H\}\mathbf{w} \\ &= \mathbf{w}^H \underbrace{\left[\mathcal{C}_{\mathbf{\Delta x \Delta x}} - \rho_s \mathbf{I}\right]}_{\triangleq \mathbf{B}} \mathbf{w}\end{aligned} \tag{6.16}$$

where $\mathbf{B} = \mathbf{B}^H$ and $\mathbf{I}$ is the identity matrix.

To solve the constrained optimisation problem (6.11), consider the Lagrangian function $\mathcal{L}(\mathbf{w}, \mathbf{w}^*, \lambda) : \mathbb{C}^N \times \mathbb{C}^N \times \mathbb{R} \mapsto \mathbb{R}$ given by

$$\mathcal{L}(\mathbf{w}, \mathbf{w}^*, \lambda) = \mathcal{J}_N(\mathbf{w}, \mathbf{w}^*) + \lambda \mathcal{J}_S(\mathbf{w}, \mathbf{w}^*) \tag{6.17}$$

where $\lambda \in \mathbb{R}$ is the Lagrange multiplier. For the inequality constraint $\mathcal{J}_S$, the Karush-Kuhn-Tucker conditions are to be considered and satisfied. However, the method in [124] is used to transform the smoothness inequality constraint into the equality constraint $\acute{\mathcal{J}}_S = \max(\mathcal{J}_S, 0) = 0$, resulting in a simpler solution. The Newton method is then used to find the extrema of the Lagrangian, defined in augmented complex form as [50] (see Section B.2.2 in Appendix B)

$$\Delta\mathbf{w}^a = -\mathbf{H}_{\mathbf{ww}}^{a^{-1}} \left( \frac{\partial \mathcal{L}}{\partial \mathbf{w}^{a*}} \right) \tag{6.18}$$

where $\mathbf{w}^a = [\mathbf{w}^T,\ \mathbf{w}^H]^T$ denotes an augmented complex column vector and $\mathbf{H}_{\mathbf{ww}}^a$ is the augmented Hessian matrix, given by

$$\mathbf{H}_{\mathbf{ww}}^a = \begin{bmatrix} \mathbf{H}_{\mathbf{ww}^*} & \mathbf{H}_{\mathbf{w}^*\mathbf{w}^*} \\ \mathbf{H}_{\mathbf{ww}} & \mathbf{H}_{\mathbf{w}^*\mathbf{w}} \end{bmatrix} \tag{6.19}$$

Expanding the augmented Newton update and solving for $\Delta\mathbf{w}$ results in the Newton step given in (6.12) (see also [54]), where the individual gradient components, calculated using $\mathbb{CR}$ calculus, are given by

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{w}^*} &= E\{g(|y|^2) y^* \mathbf{x}\} + \lambda \epsilon \beta \mathbf{B}\mathbf{w} \\ \frac{\partial \mathcal{L}}{\partial \mathbf{w}} &= \left( \frac{\partial \mathcal{L}}{\partial \mathbf{w}^*} \right)^*,\end{aligned} \tag{6.20}$$

and the Hessian components are given by

$$\begin{aligned}
\mathbf{H}_{\mathbf{w}^*\mathbf{w}^*} &= \frac{\partial}{\partial \mathbf{w}^*}\left(\frac{\partial \mathcal{L}}{\partial \mathbf{w}^*}\right)^T = E\{g'(|y|^2)y^{*2}\mathbf{x}\mathbf{x}^T\} \approx E\{g'(|y|^2)y^{*2}\}E\{\mathbf{x}\mathbf{x}^T\} \\
\mathbf{H}_{\mathbf{w}^*\mathbf{w}} &= \frac{\partial}{\partial \mathbf{w}^*}\left(\frac{\partial \mathcal{L}}{\partial \mathbf{w}}\right)^T = E\{g'(|y|^2)|y|^2 + g(|y|^2)\}\mathbf{I} + \lambda\epsilon\beta\mathbf{B} \\
\mathbf{H}_{\mathbf{w}\mathbf{w}} &= \left(\mathbf{H}_{\mathbf{w}^*\mathbf{w}^*}\right)^* \\
\mathbf{H}_{\mathbf{w}\mathbf{w}^*} &= \left(\mathbf{H}_{\mathbf{w}^*\mathbf{w}}\right)^*,
\end{aligned} \tag{6.21}$$

with $\epsilon = \big(\operatorname{sgn}(\mathcal{J}_S) + 1\big)/2$, and $g$ and $g'$ denote the first and second derivative of the nonlinearity $G$. As in [12], for whitened data the approximation $E\{f(x)xx\} \approx E\{f(x)\}E\{xx\}$ can be used. The value of $\lambda$ is updated using a gradient ascent method at each iteration, as given in (6.12). A value of $\lambda = 0$, results in the unconstrained problem, for which the solution given in [78], as a generalised complex FastICA algorithm (nc-FastICA).

For the calculation of the S-cBSE algorithm based on the standard complex FastICA, the block off-diagonal elements of $\mathbf{H}^a_{\mathbf{w}\mathbf{w}}$ in (6.19) are assumed to be zero, and form a quasi-Newton Hessian matrix[4]. Notice that the assumption of a quasi-Newton Hessian matrix can equivalently be viewed as the condition of having proper sources where $E\{\mathbf{x}\mathbf{x}^T\}$ vanishes. Thus, the corresponding values $\mathbf{H}_{\mathbf{w}^*\mathbf{w}^*}$ and $\mathbf{H}_{\mathbf{w}\mathbf{w}}$ in (6.21) are zero, and the S-cBSE algorithm is simplified as

$$\Delta\mathbf{w} = -\Big(E\{g'(|y|^2)|y|^2 + g(|y|^2)\}\mathbf{I} + \lambda\beta\mathbf{B}^T\epsilon\Big)^{-1} \cdot \Big(E\{g(|y|^2)y\mathbf{x}^*\} + \lambda\beta\mathbf{B}\epsilon\mathbf{w}\Big). \tag{6.22}$$

[4]Overview of the c-FastICA and nc-FastICA algorithms is given in Appendix D, with the nc-FastICA algorithm expressed in Equation (D.6) and c-FastICA algorithm is given in Equation (D.7).

# Chapter 7

# A Fast Independent Component Analysis Algorithm for Improper Quaternion Signals

## 7.1 Introduction

In the previous chapters, supervised and unsupervised adaptive signal processing algorithms in the complex domain based on augmented complex statistics and the $\mathbb{CR}$ calculus framework have been discussed. It has been shown that the augmented statistical modelling allows for consideration of general signals in $\mathbb{C}$. For example, in Chapter 3 comparison of the standard CLMS and augmented CLMS algorithms demonstrates better prediction of improper complex wind vectors. Likewise, in Chapter 6, the smoothness based complex blind source extraction (S-cBSE) algorithm using the generalised complex FastICA results in better extraction for the generality of complex sources, when compared to the standard circular complex FastICA. Derivations of such algorithms were based on real-valued cost functions, and the $\mathbb{CR}$ calculus framework has been shown to provide the flexibility and simplicity to enable their calculation.

In the same light, it is thus natural to consider the extension of such concepts to the higher dimensional quaternion domain $\mathbb{H}$. Indeed, there has been recent interest in adaptive signal processing algorithms in the quaternion domain, a natural domain for the processing of three- and four-dimensional signals. While modelling in the complex domain allows for the exhaustive and simultaneous processing of two-dimensional signals, quaternionic modelling allows for higher dimensional representations.

Research on quaternion-valued signal processing is currently in its inception phase with focus on understanding and addressing problems from a statistical and algorithmic point of view. The literature on quaternion-valued signal processing includes the algebraic [128, 129] as well as statistical approaches [130, 131]. More recent developments include the analysis of quaternion-valued random variables via augmented quaternion statistics [132], and the so called $\mathbb{HR}$ calculus, a unified framework for the analysis of non-analytic quaternion functions [133, 134].

These advances have been exploited through widely linear modelling of quaternion signals, allowing us to incorporate the full second-order information and have led to the class of widely linear quaternion least mean square (WL-QLMS) algorithms [135]. In nonlinear signal models, both split- and fully-quaternionic nonlinear models have been successfully implemented [136]. In the study of unsupervised adaptive algorithms, a quaternion ICA algorithm based on likelihood maximisation and the concept of Infomax was proposed by Le Bihan and Buchholz in [137]. In their study, it was concluded that a fully-quaternion nonlinearity results in a better separation performance.

In this chapter, the scope of the FastICA algorithm is extended by proposing an algorithm suitable for the separation of $\mathbb{Q}$-proper and $\mathbb{Q}$-improper quaternion-valued signals from an observed linear mixture. This is achieved by means of augmented quaternion statistics, widely linear modelling and $\mathbb{HR}$ calculus, and based on the augmented Newton method, whereby at the cost of additional complexity the complete statistical properties of the signals is captured and ensure successful separation of latent sources. The performance of the algorithm using synthetic $\mathbb{Q}$-proper and $\mathbb{Q}$-improper polytope signals in both deflationary and simultaneous separation scenarios is studied, and is followed by a real-world case study of electroencephalogram (EEG) artifact extraction.

## 7.2 Preliminaries on Quaternion Signals

In this section, a brief overview of algebra and statistics in $\mathbb{H}$ is provided. Quaternion algebra is a non-commutative algebra, while real and complex algebra are commutative. Also, statistics in $\mathbb{H}$ can be seen as a generalisation of the augmented complex statistics discussed in Chapter 2.

### 7.2.1 Quaternion algebra

Consider the quaternion variable

$$\mathrm{q} = \mathrm{q}_a + \imath \mathrm{q}_b + \jmath \mathrm{q}_c + \kappa \mathrm{q}_d \in \mathbb{H} \tag{7.1}$$

where $\mathsf{q}_a, \mathsf{q}_b, \mathsf{q}_c$ and $\mathsf{q}_d$ are real-valued scalars, and $\imath, \jmath$ and $\kappa$ are orthogonal unit vectors such that

$$\begin{aligned} &\imath = \jmath = \kappa = \sqrt{-1} \\ &\imath\jmath = \kappa \qquad \jmath\kappa = \imath \qquad \kappa\imath = \jmath \\ &\imath\jmath\kappa = \imath^2 = \jmath^2 = \kappa^2 = -1. \end{aligned} \tag{7.2}$$

The number $\mathsf{q}$ can also be written in terms of its real (scalar) part $\Re\{\mathsf{q}\} = \mathsf{q}_a$ and its vector part $\Im\{\mathsf{q}\} = \imath\Im_\imath\{\mathsf{q}\} + \jmath\Im_\jmath\{\mathsf{q}\} + \kappa\Im_\kappa\{\mathsf{q}\}$, such that

$$\begin{aligned} \mathsf{q} &= \Re\{\mathsf{q}\} + \Im\{\mathsf{q}\} \\ &= \Re\{\mathsf{q}\} + \imath\Im_\imath\{\mathsf{q}\} + \jmath\Im_\jmath\{\mathsf{q}\} + \kappa\Im_\kappa\{\mathsf{q}\} \end{aligned} \tag{7.3}$$

Alternatively, by adopting the Cayley-Dickson notation, $\mathsf{q}$ can be constructed from a pair of complex quantities $\mathsf{z}_1 = \mathsf{q}_a + \imath\mathsf{q}_b$ and $\mathsf{z}_2 = \mathsf{q}_c + \imath\mathsf{q}_d$, such that $\mathsf{q} = \mathsf{z}_1 + \mathsf{z}_2\jmath$, however in this work direct quaternionic notation will be used.

The identities in Equation (7.2) illustrate the non-commutative property of products in quaternion algebra, whereby $\mathsf{q}_1\mathsf{q}_2 \neq \mathsf{q}_2\mathsf{q}_1$. This can alternatively be seen directly from the multiplication of $\mathsf{q}_1$ and $\mathsf{q}_2$, which after simplification is given by

$$\begin{aligned} \mathsf{q}_1\mathsf{q}_2 &= (\mathsf{q}_{1a} + \imath\mathsf{q}_{1b} + \jmath\mathsf{q}_{1c} + \kappa\mathsf{q}_{1d})(\mathsf{q}_{2a} + \imath\mathsf{q}_{2b} + \jmath\mathsf{q}_{2c} + \kappa\mathsf{q}_{2d}) \\ &= (\mathsf{q}_{1a}\mathsf{q}_{2a} - \mathsf{q}_{1b}\mathsf{q}_{2b} - \mathsf{q}_{1c}\mathsf{q}_{2c} - \mathsf{q}_{1d}\mathsf{q}_{2d}) \\ &\quad + \mathsf{q}_{1a}\Im\{\mathsf{q}_2\} + \mathsf{q}_{2a}\Im\{\mathsf{q}_1\} + \Im\{\mathsf{q}_1\} \times \Im\{\mathsf{q}_2\} \end{aligned} \tag{7.4}$$

where the symbol '$\times$' denotes the vector product. It is then seen that the non-commutativity of the vector product results in the non-commutativity of the quaternion product.

In the quaternion domain, three self-inverse mappings[1] or *involutions* [138] can be considered about the $\imath, \jmath$ and $\kappa$ axes,

$$\begin{aligned} \mathsf{q}^\imath &= -\imath\mathsf{q}\imath = \mathsf{q}_a + \imath\mathsf{q}_b - \jmath\mathsf{q}_c - \kappa\mathsf{q}_d \\ \mathsf{q}^\jmath &= -\jmath\mathsf{q}\jmath = \mathsf{q}_a - \imath\mathsf{q}_b + \jmath\mathsf{q}_c - \kappa\mathsf{q}_d \\ \mathsf{q}^\kappa &= -\kappa\mathsf{q}\kappa = \mathsf{q}_a - \imath\mathsf{q}_b - \jmath\mathsf{q}_c + \kappa\mathsf{q}_d \end{aligned} \tag{7.5}$$

which form the bases for augmented quaternion statistics [132]. Intuitively, an involution represents a rotation along each respective axis, while the conjugate operator $(\cdot)^*$ forms an involution along all three directions, where

$$\mathsf{q}^* = \mathsf{q}_a - \imath\mathsf{q}_b - \jmath\mathsf{q}_c - \kappa\mathsf{q}_d. \tag{7.6}$$

[1] A self-inverse mapping operator $\mathrm{sinv}(\cdot)$ is such that $\mathrm{sinv}\big(\mathrm{sinv}(\mathsf{q})\big) = \mathsf{q}$.

The involutions have the property that $(\mathsf{q}_1\mathsf{q}_2)^\alpha = \mathsf{q}_1^\alpha\mathsf{q}_2^\alpha,\ \alpha = \{\imath, \jmath, \kappa\}$, while $(\mathsf{q}_1\mathsf{q}_2)^* = \mathsf{q}_2^*\mathsf{q}_1^*$. Finally, the norm (modulus) of a quaternion variable $\mathsf{q}$ is defined by

$$\|\mathsf{q}\|_2 = \sqrt{\mathsf{q}\mathsf{q}^*} = \sqrt{\mathsf{q}^*\mathsf{q}} = \sqrt{\mathsf{q}_a^2 + \mathsf{q}_b^2 + \mathsf{q}_c^2 + \mathsf{q}_d^2} \tag{7.7}$$

whereby for a vector $\mathbf{q}$ in a quaternion Hilbert space [130], the 2-norm is defined as $\|\mathbf{q}\|_2 = \sqrt{\mathbf{q}^H\mathbf{q}}$.

### 7.2.2 Augmented quaternion statistics

For a random vector $\mathbf{q} = \mathbf{q}_a + \imath\mathbf{q}_b + \jmath\mathbf{q}_c + \kappa\mathbf{q}_d \in \mathbb{H}^N$, the probability density function (pdf) is defined in terms of the joint pdf of its scalar and vector components, such that $p_Q(\mathbf{q}) \triangleq p_{Q_a,Q_b,Q_c,Q_d}(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d)$. Its mean is then calculated in terms of each respective component as

$$E\{\mathbf{q}\} = E\{\mathbf{q}_a\} + \imath E\{\mathbf{q}_b\} + \jmath E\{\mathbf{q}_c\} + \kappa E\{\mathbf{q}_d\} \tag{7.8}$$

and the quadrivariate covariance matrix of real-valued component vectors

$$\mathcal{C}^R_{\mathbf{qq}} = E\{\mathbf{q}^R\mathbf{q}^{RT}\} \in \mathbb{R}^{4N\times 4N} \tag{7.9}$$

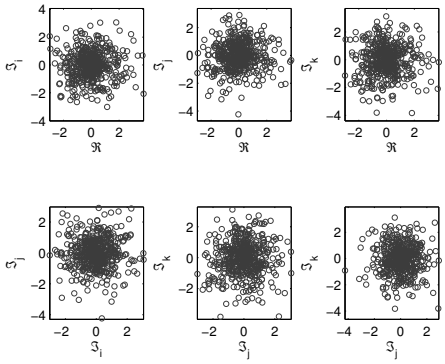
describes the second-order relationship between the respective components of $\mathbf{q}$, where $\mathbf{q}^R = [\mathbf{q}_a^T, \mathbf{q}_b^T, \mathbf{q}_c^T, \mathbf{q}_d^T]^T$. Representing the components of $\mathcal{C}^R_{\mathbf{qq}}$ by their equivalent quaternion counterparts allows for the complete second-order statistical information to be captured directly in $\mathbb{H}$ [132]. This is achieved by considering the relation between the components of the quaternion variable $\mathbf{q}$ and its involutions (7.5), given by

$$\begin{aligned}
\mathbf{q}_a &= \frac{1}{4}(\mathbf{q} + \mathbf{q}^\imath + \mathbf{q}^\jmath + \mathbf{q}^\kappa), \quad \mathbf{q}_b = \frac{1}{4}(\mathbf{q} + \mathbf{q}^\imath - \mathbf{q}^\jmath - \mathbf{q}^\kappa) \\
\mathbf{q}_c &= \frac{1}{4}(\mathbf{q} - \mathbf{q}^\imath + \mathbf{q}^\jmath - \mathbf{q}^\kappa), \quad \mathbf{q}_d = \frac{1}{4}(\mathbf{q} - \mathbf{q}^\imath - \mathbf{q}^\jmath + \mathbf{q}^\kappa).
\end{aligned} \tag{7.10}$$

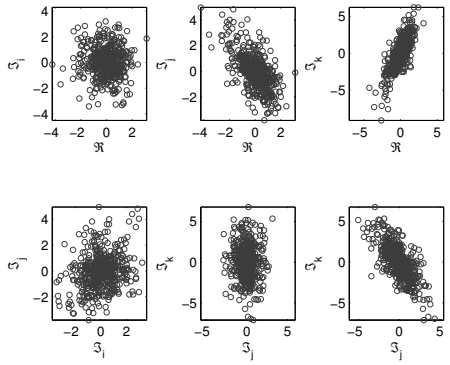
In analogy to the complex domain[2] where both $\mathbf{z}$ and $\mathbf{z}^*$ are used to define the augmented statistics [45, 48], it can be shown that the bases $\mathbf{q}, \mathbf{q}^\imath, \mathbf{q}^\jmath$ and $\mathbf{q}^\kappa$ provide a suitable means to define the augmented quaternion statistics [132]. This way, the augmented random vector $\mathbf{q}^a = [\mathbf{q}^T, \mathbf{q}^{\imath T}, \mathbf{q}^{\jmath T}, \mathbf{q}^{\kappa T}]^T$ is used to define the augmented covariance matrix

$$\begin{aligned}
\mathcal{C}^a_{\mathbf{qq}} &= E\{\mathbf{q}^a\mathbf{q}^{aH}\} \\
&= \begin{bmatrix}
\mathcal{C}_{\mathbf{qq}} & \mathcal{C}_{\mathbf{q}^\imath} & \mathcal{C}_{\mathbf{q}^\jmath} & \mathcal{C}_{\mathbf{q}^\kappa} \\
\mathcal{C}^H_{\mathbf{q}^\imath} & \mathcal{C}_{\mathbf{q}^\imath\mathbf{q}^\imath} & \mathcal{C}_{\mathbf{q}^\imath\mathbf{q}^\jmath} & \mathcal{C}_{\mathbf{q}^\imath\mathbf{q}^\kappa} \\
\mathcal{C}^H_{\mathbf{q}^\jmath} & \mathcal{C}_{\mathbf{q}^\jmath\mathbf{q}^\imath} & \mathcal{C}_{\mathbf{q}^\jmath\mathbf{q}^\jmath} & \mathcal{C}_{\mathbf{q}^\jmath\mathbf{q}^\kappa} \\
\mathcal{C}^H_{\mathbf{q}^\kappa} & \mathcal{C}_{\mathbf{q}^\kappa\mathbf{q}^\imath} & \mathcal{C}_{\mathbf{q}^\kappa\mathbf{q}^\jmath} & \mathcal{C}_{\mathbf{q}^\kappa\mathbf{q}^\kappa}
\end{bmatrix} \in \mathbb{H}^{4N\times 4N}
\end{aligned} \tag{7.11}$$

[2] Recall from Section 2.1.3 that in the complex domain, the real and imaginary components can be represented in terms of the conjugate coordinates $\mathbf{z}$ and $\mathbf{z}^*$ respectively as $\frac{1}{2}(\mathbf{z} + \mathbf{z}^*)$ and $\frac{1}{2\jmath}(\mathbf{z} - \mathbf{z}^*)$.



(a) Scatter plot of a $\mathbb{Q}$-proper quaternion random variable



(b) Scatter plot of a $\mathbb{Q}$-improper quaternion random variable

**Figure 7.1** Scatter plots of $\mathbb{Q}$-proper and $\mathbb{Q}$-improper quaternion Gaussian random variables.

which describes the complete second-order information available within a quaternion random vector. In (7.11), $\mathcal{C}_{\mathbf{q}^\imath}, \mathcal{C}_{\mathbf{q}^\jmath}, \mathcal{C}_{\mathbf{q}^\kappa}$ are respectively termed the $\imath$-, $\jmath$- and $\kappa$-covariance matrices $E\{\mathbf{q}\mathbf{q}^{\alpha H}\},\ \alpha = \{\imath, \jmath, \kappa\}$, while $\mathcal{C}_{\mathbf{qq}} = E\{\mathbf{q}\mathbf{q}^H\}$ is the standard covariance matrix. The $\imath$-, $\jmath$- and $\kappa$-covariance matrices are referred to as the complementary or pseudo-covariance matrices [48].

The concept of properness (rotation invariant pdf) can be extended from the complex to the quaternion domain and has been discussed in [130] and [131]. Following the involution-based augmented bases, a random vector is considered $\mathbb{Q}$-proper (see Figure 7.1(a)) if it is not correlated with its involutions, or, $\mathcal{C}_{\mathbf{q}^\imath} = \mathcal{C}_{\mathbf{q}^\jmath} = \mathcal{C}_{\mathbf{q}^\kappa} = \mathbf{0}$, and all cross-covariance matrices vanish, and is otherwise termed $\mathbb{Q}$-improper [132]. In the example scatter plot in Figure 7.1(b), the quaternion random variable is not rotation invariant, with correlated scalar and vector components. Therefore, for a $\mathbb{Q}$-proper random vector, the augmented covariance matrix (7.11) has a block-diagonal structure. More restricted definitions of properness can also be found, whereby one or more pseudo-covariances are non-zero ($\mathbb{C}$-proper) [131]. This can be intuitively understood as rotation invariance along one or more of the quaternion axes; $\mathbb{Q}$-properness thus reflects rotation invariance along all the three imaginary axes.

### 7.2.3 Widely linear modelling in $\mathbb{H}$

Recall that the solution to the mean square error (MSE) estimator of a real-valued signal $y \in \mathbb{R}$ in terms of an observation $x$, expressed as $\hat{y} = E\{y|x\}$, is given by $\hat{y} = \mathbf{h}^T\mathbf{x}$, where $\mathbf{h}$ is a coefficient vector and $\mathbf{x}$ the regressor. As a generalisation, the MSE estimator for a quaternion-valued signal $y \in \mathbb{H}$ can then be written in terms of

the MSE estimators of its respective components, given by

$$\begin{aligned}\hat{y}_a &= E\{y_a|q_a, q_b, q_c, q_d\}\\ \hat{y}_b &= E\{y_b|q_a, q_b, q_c, q_d\}\\ \hat{y}_c &= E\{y_c|q_a, q_b, q_c, q_d\}\\ \hat{y}_d &= E\{y_d|q_a, q_b, q_c, q_d\},\end{aligned} \tag{7.12}$$

such that

$$\begin{aligned}\hat{y} &= \hat{y}_a + \imath\hat{y}_b + \jmath\hat{y}_c + \kappa\hat{y}_d\\ &= E\{y_a|q_a, q_b, q_c, q_d\} + \imath E\{y_b|q_a, q_b, q_c, q_d\}\\ &+ \jmath E\{y_c|q_a, q_b, q_c, q_d\} + \kappa E\{y_d|q_a, q_b, q_c, q_d\}.\end{aligned} \tag{7.13}$$

Observe that by using the relations (7.10), the MSE estimator of $y$ can be equivalently written as

$$\begin{aligned}\hat{y} &= E\{y|q, q^\imath, q^\jmath, q^\kappa\} + \imath E\{y^\imath|q, q^\imath, q^\jmath, q^\kappa\}\\ &+ \jmath E\{y^\jmath|q, q^\imath, q^\jmath, q^\kappa\} + \kappa E\{y^\kappa|q, q^\imath, q^\jmath, q^\kappa\},\end{aligned} \tag{7.14}$$

and results in the widely linear estimator [132, 135]

$$\begin{aligned}y &= \mathbf{h}^H\mathbf{q} + \mathbf{g}^H\mathbf{q}^\imath + \mathbf{u}^H\mathbf{q}^\jmath + \mathbf{v}^H\mathbf{q}^\kappa\\ &= \mathbf{w}^{aH}\mathbf{q}^a\end{aligned} \tag{7.15}$$

where the augmented weight vector $\mathbf{w}^a = [\mathbf{h}^T, \mathbf{g}^T, \mathbf{u}^T, \mathbf{v}^T]^T$. Thus (7.15) is the optimal estimator for the generality of quaternion-valued signals, both proper and improper.

### 7.2.4 An overview of $\mathbb{HR}$ calculus

In signal processing problems, it is common to define a real-valued cost function, typically the error power. In a similar fashion to the $\mathbb{CR}$ calculus framework where a function is defined based on the conjugate coordinates $\mathbf{z}$ and $\mathbf{z}^*$ [55, 54] (also see discussion in Appendix B), in the context of $\mathbb{HR}$ calculus [133], $f(\mathbf{q}) : \mathbb{H}^N \mapsto \mathbb{R}$ can be considered as a function of the orthogonal quaternion basis vectors $\mathbf{q}, \mathbf{q}^\imath, \mathbf{q}^\jmath$ and $\mathbf{q}^\kappa$, such that

$$f(\mathbf{q}, \mathbf{q}^\imath, \mathbf{q}^\jmath, \mathbf{q}^\kappa) : \mathbb{H}^N \times \mathbb{H}^N \times \mathbb{H}^N \times \mathbb{H}^N \mapsto \mathbb{R}. \tag{7.16}$$

Likewise, the duality between a quaternion function $f$ and its real-valued equivalent $g$ can be expressed as

$$\begin{aligned} f(\mathbf{q}) &= f(\mathbf{q}, \mathbf{q}^{\imath}, \mathbf{q}^{\jmath}, \mathbf{q}^{\kappa}) \\ &= f_a(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d) + \imath f_b(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d) \\ &\quad + \jmath f_c(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d) + \kappa f_d(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d) \\ &= g(\mathbf{q}_a, \mathbf{q}_b, \mathbf{q}_c, \mathbf{q}_d) \end{aligned} \tag{7.17}$$

Then, by considering the components of the quaternion variable $\mathbf{q}$ and the orthogonal bases given in (7.10), a relation can be established between the derivatives taken with respect to the components of the quaternion variable and those taken directly with respect to the quaternion basis variables, forming a fundamental result of $\mathbb{HR}$ calculus. These relations, know as $\mathbb{HR}$ derivatives, are given by [133, 134]

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{q}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} - \imath\frac{\partial f}{\partial \mathbf{q}_b} - \jmath\frac{\partial f}{\partial \mathbf{q}_c} - \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\imath}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} - \imath\frac{\partial f}{\partial \mathbf{q}_b} + \jmath\frac{\partial f}{\partial \mathbf{q}_c} + \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\jmath}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} + \imath\frac{\partial f}{\partial \mathbf{q}_b} - \jmath\frac{\partial f}{\partial \mathbf{q}_c} + \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\kappa}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} + \imath\frac{\partial f}{\partial \mathbf{q}_b} + \jmath\frac{\partial f}{\partial \mathbf{q}_c} - \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right). \end{aligned} \tag{7.18}$$

The so called $\mathbb{HR}^*$ derivatives can then readily be written from (7.18) by using the property $\left(\frac{\partial f}{\partial \mathbf{q}}\right)^* = \frac{\partial f}{\partial \mathbf{q}^*}$, where $f$ is a real-valued function. Thus,

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{q}^*} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} + \imath\frac{\partial f}{\partial \mathbf{q}_b} + \jmath\frac{\partial f}{\partial \mathbf{q}_c} + \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\imath *}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} + \imath\frac{\partial f}{\partial \mathbf{q}_b} - \jmath\frac{\partial f}{\partial \mathbf{q}_c} - \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\jmath *}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} - \imath\frac{\partial f}{\partial \mathbf{q}_b} + \jmath\frac{\partial f}{\partial \mathbf{q}_c} - \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right) \\ \frac{\partial f}{\partial \mathbf{q}^{\kappa *}} &= \frac{1}{4}\left(\frac{\partial f}{\partial \mathbf{q}_a} - \imath\frac{\partial f}{\partial \mathbf{q}_b} - \jmath\frac{\partial f}{\partial \mathbf{q}_c} + \kappa\frac{\partial f}{\partial \mathbf{q}_d}\right). \end{aligned} \tag{7.19}$$

Similar to the conjugate derivatives property, an involution property is also applicable to real-valued functions, and is given by

$$\left(\frac{\partial f}{\partial \mathbf{q}}\right)^{\alpha} = \frac{\partial f}{\partial \mathbf{q}^{\alpha}}, \quad \alpha = \{\imath, \jmath, \kappa\}. \tag{7.20}$$

It has been shown that in the quaternion domain, the direction of steepest descent (maximum rate of change of $f(\mathbf{q})$) is given by the derivative with respect to $\mathbf{q}^*$, or $\frac{\partial f}{\partial \mathbf{q}^*}$. This can be seen as an extension of Brandwood's result for functions of complex variables [53], and it is thus natural to consider this gradient in the optimisation of

cost functions. Finally, note that while real-valued functions have been considered in the above discussion, the $\mathbb{HR}$ calculus framework can be equally utilised for the analysis of general quaternion-valued functions. Appendices 7.A and 7.B at the end of this chapter provide further information on the chain rule and augmented Newton method in $\mathbb{HR}$ calculus.

## 7.3 The Quaternion FastICA Algorithm

Consider the standard ICA model

$$\check{\mathbf{x}} = \mathbf{A}\mathbf{s} \tag{7.21}$$

whereby the observed mixtures $\check{\mathbf{x}} \in \mathbb{H}^N$ are a weighted sum of $N_s$ latent sources $\mathbf{s} \in \mathbb{H}^{N_s}$ in a noise-free environment, and the rows of $\mathbf{A} \in \mathbb{H}^{N \times N_s}$ form the respective mixing parameters. While no knowledge of the mixing process is available, the sources are assumed statistically independent; for convenience they have zero mean and unit variance and no assumption is made regarding the $\imath-, \jmath-$ and $\kappa-$variances. The mixing matrix $\mathbf{A}$ is assumed square ($N = N_s$), well-conditioned and invertible.

For a quaternion random vector $\mathbf{q} \in \mathbb{H}^N$, its whitening matrix $\mathbf{V}$ is given by

$$\mathbf{V} = \mathbf{\Lambda}^{-1/2}\mathbf{E}^H, \tag{7.22}$$

where $\mathbf{\Lambda}$ is the diagonal matrix of right eigenvalues[3] and $\mathbf{E}$ is the matrix of corresponding eigenvectors of the covariance matrix of $\mathbf{q}$.

To prove this, write the covariance matrix in terms of the quaternion right eigenvalue decomposition $\mathcal{C}_{\mathbf{qq}} = E\{\mathbf{q}\mathbf{q}^H\} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^H$ [139]. The covariance matrix of the whitened random vector $\mathbf{p} = \mathbf{V}\mathbf{q}$ is then expressed as

$$\begin{aligned} E\{\mathbf{p}\mathbf{p}^H\} &= \mathbf{V}E\{\mathbf{q}\mathbf{q}^H\}\mathbf{V}^H \\ &= \mathbf{\Lambda}^{-1/2}\mathbf{E}^H\big(\mathbf{E}\mathbf{\Lambda}\mathbf{E}^H\big)\mathbf{E}\mathbf{\Lambda}^{-1/2} = \mathbf{I} \end{aligned} \tag{7.23}$$

where $\mathbf{I}$ is the identity matrix. This result will be used for the whitening of the observed mixture $\check{\mathbf{x}}$ in (7.21).

As a preprocessing step to aid the ICA algorithm, the quaternion mixture $\check{\mathbf{x}}$ is whitened such that

$$E\{\mathbf{x}\mathbf{x}^H\} = \mathbf{M}E\{\mathbf{s}\mathbf{s}^H\}\mathbf{M}^H = \mathbf{I} \tag{7.24}$$

[3] Due to the non-commutativity of the quaternion algebra, left and right scalar multiplications are different and lead to left and right eigenvalues [139].

where $\mathbf{x} = \mathbf{V}\check{\mathbf{x}} = \mathbf{VAs}$ and $\mathbf{M} \triangleq \mathbf{VA}$ is the new unitary mixing matrix containing the whitening matrix $\mathbf{V}$, given in (7.22). The aim is to obtain a demixing matrix $\mathbf{W}$ such that $\mathbf{W}^H\mathbf{x}$ is an estimate of the original sources, albeit with a scaling, phase and permutation ambiguity. Then for the $n$th source estimate

$$y_n = \mathbf{w}_n^H\mathbf{x} = \mathbf{w}_n^H\mathbf{Ms} = \mathbf{u}^H\mathbf{s} = e^{\boldsymbol{\xi}\varphi}s_m \tag{7.25}$$

where $\mathbf{w}_n$ is the $n$th column of the demixing matrix $\mathbf{W}$, $\mathbf{u}$ is a vector with a single non-zero value given by $e^{\boldsymbol{\xi}\varphi}$ at the $n$th entry signifying an arbitrary direction within $\mathbb{H}$, $\varphi$ is an arbitrary and unknown angle and $\boldsymbol{\xi} = \frac{(\imath q_b + \jmath q_c + \kappa q_d)}{\sqrt{q_b^2+q_c^2+q_d^2}}$ is the unit pure quaternion vector[4]. Finally, note that by constraining the demixing vector $\mathbf{w}_n$ to unit norm, the estimated source $y_n$ is of unit variance, that is

$$E\{y_n y_n^*\} = \mathbf{w}_n^H E\{\mathbf{x}\mathbf{x}^H\}\mathbf{w}_n = \mathbf{w}_n^H\mathbf{w}_n = 1 \tag{7.26}$$

while the matrix $\mathbf{W}$ becomes unitary.

### 7.3.1 A Newton-update based ICA algorithm

The quaternion FastICA (q-FastICA) algorithm is based on the maximisation of the negentropy of the separated sources, following from previous implementations of the FastICA algorithm in the real and complex domains [12, 41, 78]. This is achieved by utilising an appropriate nonlinear function $G(y)$, so as to make a suitable approximation of the negentropy function.

In [137], three distinct quaternion nonlinearities were identified whereby the nonlinear operation is split on each component of $y$ (split-quaternion function), on the components of the Cayley-Dickson form of $y$ (split-complex function), or applied directly on $y$ (full-quaternion function). It was also shown that the full-quaternion nonlinearity resulted in the best separation performance. Under the stringent analyticity conditions of the Cauchy-Riemann-Feuter [140] equations, the only analytic function in $\mathbb{H}$ is a constant. As an alternative, local analyticity conditions may be considered in the calculation of the derivatives [141]. However, this depends on assumptions that may not be valid for general nonlinear functions. Thus, to avoid problems associated with the derivation of fully-quaternion nonlinearities, a real-valued smooth and even nonlinearity $G : \mathbb{R} \mapsto \mathbb{R}$ is utilised, while implementing an augmented Newton method so as to employ the full information available within general $\mathbb{Q}$-improper mixtures.

The q-FastICA cost function is then defined as

$$\mathcal{J}(\mathbf{w}, \mathbf{w}^\imath, \mathbf{w}^\jmath, \mathbf{w}^\kappa) = E\big\{G(|\mathbf{w}^H\mathbf{x}|^2)\big\} \tag{7.27}$$

[4] A pure '*imaginary*' quaternion is referred to as the imaginary or vector part of a quaternion variable.

where the cost function $\mathcal{J}$ is written in terms of the four basis vectors for emphasis on the equivalent notation. The optimisation problem based on (7.27) can then be stated as

$$\mathbf{w}_{opt} = \arg \max_{\|\mathbf{w}\|_2^2=1} \mathcal{J}(\mathbf{w}, \mathbf{w}^\imath, \mathbf{w}^\jmath, \mathbf{w}^\kappa) \tag{7.28}$$

where the demixing vector is normalised to avoid very small values of $\mathbf{w}$, while keeping the variance of the extracted sources equal to unity.

The solution of this constrained optimisation problem is found through the method of Lagrangian multipliers and by utilising the Newton method to perform a fast iterative search to the optimal value $\mathbf{w}_{opt}$. In summary, the quaternion FastICA algorithm for the estimation of one source is expressed in its augmented form as

$$\begin{aligned} \mathbf{w}^a(k+1) &= \mathbf{w}^a(k) - (\mathbf{H}^a_{\mathbf{ww}})^{-1} \nabla_{\mathbf{w}^{a*}} \mathcal{L} \\ \lambda(k+1) &= \lambda(k) + \mu \nabla_{\mathbf{w}^{a*}} \mathcal{L} \\ \mathbf{w}(k+1) &\leftarrow \frac{\mathbf{w}(k+1)}{\|\mathbf{w}(k+1)\|_2} \end{aligned} \tag{7.29}$$

where the augmented demixing vector $\mathbf{w}^a = [\mathbf{w}, \mathbf{w}^\imath, \mathbf{w}^\jmath, \mathbf{w}^\kappa]^T$, $\mathcal{L}$ is the Lagrangian function and $\lambda$ is the Lagrange parameter updated via a gradient ascent method with step-size $\mu$. The vector $\nabla_{\mathbf{w}^{a*}} \mathcal{L}$ and matrix $\mathbf{H}^a_{\mathbf{ww}}$ are respectively the augmented gradient vector and Hessian matrix of the Lagrangian function. The full derivation is provided in Appendix 7.C at the end of this chapter.

The estimation of multiple sources can be performed one by one through a deflationary procedure, where for the $n$th estimated source is given by the following Gram-Schmidt orthogonalisation procedure

$$\begin{aligned} \mathbf{w}_n(k+1) &\leftarrow \mathbf{w}_n(k+1) - \check{\mathbf{W}} \check{\mathbf{W}}^H \mathbf{w}_n(k+1) \\ \check{\mathbf{W}} &= \big[\mathbf{w}_1(k+1), \ldots, \mathbf{w}_n(k+1)\big] \end{aligned} \tag{7.30}$$

or simultaneously via a symmetric orthogonalisation method

$$\mathbf{W}(k+1) \leftarrow \big(\mathbf{W}(k+1)\mathbf{W}^H(k+1)\big)^{-1/2} \mathbf{W}(k+1), \tag{7.31}$$

where the orthogonalisation procedures in the quaternion domain follow from the already established results.

## 7.4 Simulations and Discussion

### 7.4.1 Benchmark simulations

The performance of the algorithm is first assessed through simulations using synthetic four dimensional signal codes located on the edges of geometric polytopes [142] with

a varying degree of $\mathbb{Q}$-improperness. To assess the degree of $\mathbb{Q}$-improperness of the generated sources, a measure based on the ratio of the complementary variances to the standard variance is defined, expressed as

$$r_q = \frac{\left|E\{qq^{\imath *}\}\right| + \left|E\{qq^{\jmath *}\}\right| + \left|E\{qq^{\kappa *}\}\right|}{3E\{qq^*\}}, \quad r_q \in [0,1]. \tag{7.32}$$

This way, a measure of $r_q = 0$ indicates a $\mathbb{Q}$-proper source, while for a highly $\mathbb{Q}$-improper source $r_q = 1$.

The performance of the quaternion FastICA algorithm using the deflationary orthogonalisation was assessed using the Performance Index (PI) [10], which for $\mathbf{u}^H = \mathbf{w}^H\mathbf{V}\mathbf{A} = [u_1, \ldots, u_N]^H$ is given as

$$PI = 10\log_{10}\left(\frac{1}{N}\Big(\sum_{i=1}^{N}\frac{|u_i|^2}{\max\{|u_1|^2,\ldots,|u_N|^2\}} - 1\Big)\right) \tag{7.33}$$

and indicates the proximity of $\mathbf{u}$ to a vector with a single non-zero element. For the deflationary approach, a PI of less than -20dB indicates good separation performance. For the q-FastICA algorithm with symmetric orthogonalisation, the full PI measure was used, given by

$$\begin{aligned} PI = 10\log_{10}\Bigg(&\frac{1}{N}\sum_{i=1}^{N}\Big(\sum_{j=1}^{N}\frac{|u_{ij}|}{\max\{|u_{i1}|,\ldots,|u_{iN}|\}} - 1\Big) \\ &+ \frac{1}{N}\sum_{j=1}^{N}\Big(\sum_{i=1}^{N}\frac{|u_{ij}|}{\max\{|u_{1j}|,\ldots,|u_{Nj}|\}} - 1\Big)\Bigg). \end{aligned} \tag{7.34}$$

where $\mathbf{U}^H = \mathbf{W}^H\mathbf{V}\mathbf{A}$ and $u_{ij} = (\mathbf{U})_{ij}$ and a PI less than -10dB signifies good separation performance.

In the simulations, 5000 samples of four polytope sources were mixed using a randomly generated quaternion-valued $4 \times 4$ mixing matrix. The observed mixtures were then whitened and processed using the q-FastICA algorithm (7.29), using the deflationary and symmetric orthogonalisation.

#### 7.4.1.1 Deflationary orthogonalisation

The scatter plots of the four quaternion sources are shown in Figure 7.2(a) and their properties are given in Table 7.1(a). Source $s_1(k)$ was a cubic polytope, $s_2(k)$ and $s_3(k)$ were generated from cyclic groups with two and three points, and $s_4$ was a simplex with five vertices. The nonlinearity $G(y) = \log\cosh(y)$, the demixing vector $\mathbf{w}$ was initialised randomly and the step-size of the gradient ascent update $\mu = 1$ and $\lambda = 5$. The scatter plot of the normalised estimated sources are given in Figure 7.2(b) and the

performance of the q-FastICA algorithm in the separation of each source and at each iteration stage is shown in Figure 7.2(c).

It can be seen that the algorithm was successful in estimating all the sources, converging to a solution with a performance below the PI threshold of -30 dB in as few as four iterations. As expected from a deflationary orthogonalisation procedure, the performance of the algorithm deteriorated after each stage due to the accumulation of errors, with the final PI value for the first estimated source $y_1(k)$ of -39.93dB, while for $y_4(k)$ this value reduced to -26.28dB. Note that due to the symmetry of the signal codes, rotations of the extracted sources relative to the original source are not visible, and can only be observed in the scatter plot of $y_3(k)$.

#### 7.4.1.2 Symmetric orthogonalisation

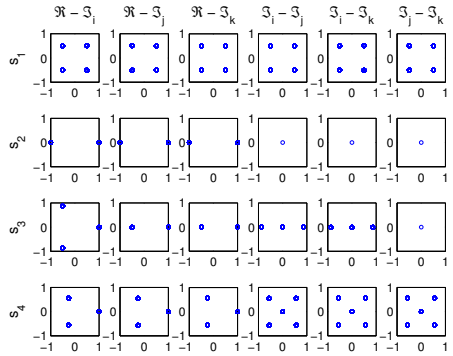
In this simulation, the sources were estimated simultaneously using the algorithm (7.29) and the orthogonalisation procedure (7.31). Table 7.1(b) describes the source properties; visual scatter plot representations are given in Figure 7.3(a). Sources $s_1(k)$ to $s_4(k)$ were respectively generated from cubic, 5 point dicyclic, 2 point cyclic and 3 point cyclic groups, source $s_3(k)$ had a high degree of $\mathbb{Q}$-improperness, the value of $r_q = 0.3351$ for $s_4(k)$, and the other two sources were $\mathbb{Q}$-proper.

For performance comparison, the nonlinearity $G$ was chosen as in [41], with $G_1(y) = \log\cosh(y), G_2(y) = \sqrt{0.1 + y}$ and $G_3(y) = \log(0.1 + y)$. The demixing matrix $\mathbf{W}$ was initialised randomly, and the step-sizes $\mu_1 = 1, \mu_2 = 0.1, \mu_3 = 0.5$ and $\lambda = 5$ for the gradient ascent update algorithm. As shown in Figure 7.3(c), the algorithm successfully separated all the four sources by achieving a PI below the -10 dB threshold with the respective PI values of -17.87 dB, -15.81 dB and -19.49 dB. Figure 7.3(b) depicts the scatter plots of the normalised estimated sources with nonlinearity $G_1$, note that sources were estimated in a random order.

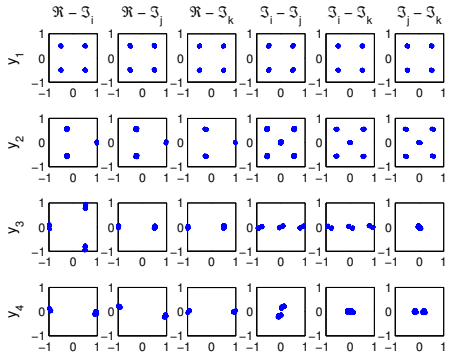
### 7.4.2 EEG artifact extraction

In a practical EEG recording session, each EEG recording channel consists of a superposition of a pure EEG signal corresponding to the collective neural activity within the brain, and electrical activity pertaining to distinctive artifacts such as movement of the head, line noise and eye blinks. In modelling the EEG signal, the artifacts, both external and biological, are considered statistically independent from the pure EEG recording [143, 116, 118]. The usefulness of the real-valued FastICA algorithm in the extraction of eyeblink artifacts was studied in [112].

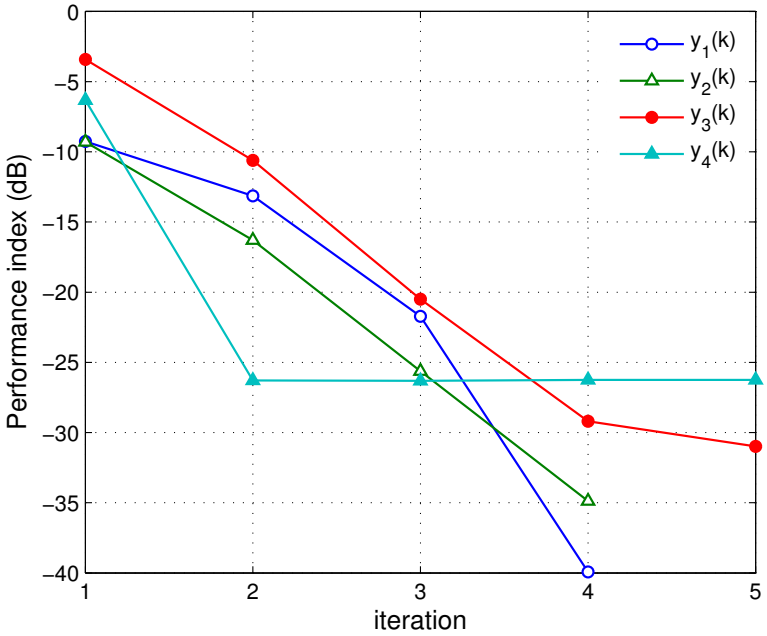
In the experimental setup, data was sampled at 4.8kHz for 30s from 12 electrodes placed symmetrically on the scalp according to the 10-20 system, as shown in Fig-



(a) The scatter plot of the quaternion sources, properties given in Table 7.1(a).

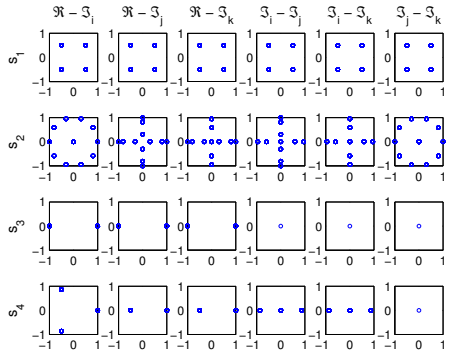


(b) The scatter plot of the estimated sources.

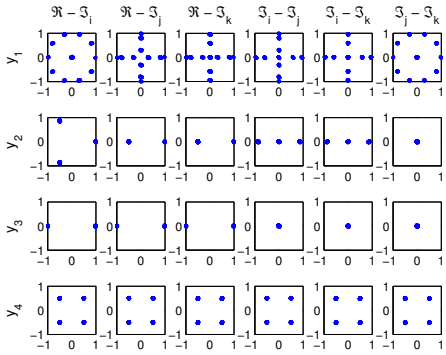



(c) The PI at each iteration of the ICA procedure.

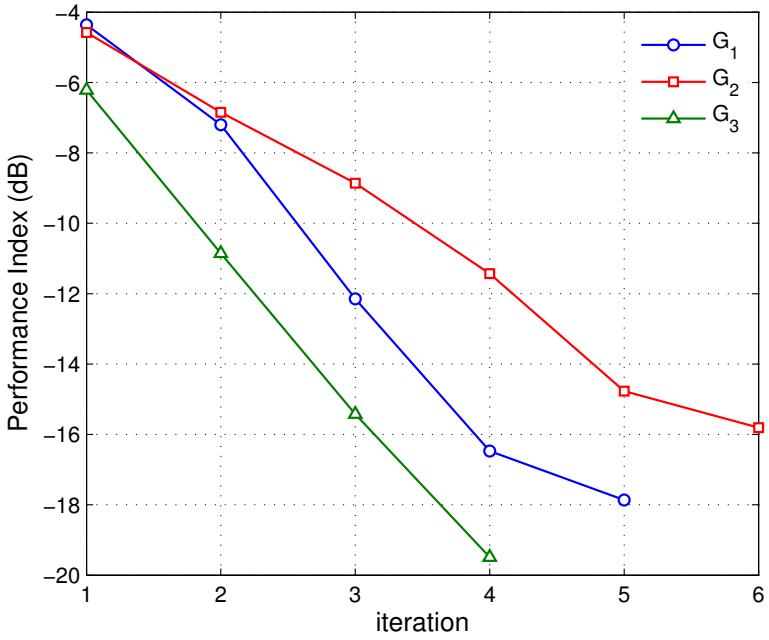
**Figure 7.2** The performance of the quaternion FastICA algorithm for the separation of four sources using a deflationary orthogonalisation procedure.



(a) The scatter plot of the quaternion sources, properties given in Table 7.1(b).



(b) The scatter plot of the estimated sources.




(c) The PI at each iteration of the ICA procedure.

**Figure 7.3** The performance of the quaternion FastICA algorithm for the separation of four sources using a symmetric orthogonalisation procedure.

**Table 7.1** Source properties for benchmark simulations using the quaternion FastICA algorithm (7.29)

(a) Source properties for benchmark simulation with deflationary approach

| Source | polytope | $\mathbb{Q}$-improperness measure ($r_q$) |
|---|---|---|
| $s_1(k)$ | Cubic | 0.01 |
| $s_2(k)$ | Cyclic (2 point) | 1.00 |
| $s_3(k)$ | Cyclic (3 point) | 0.34 |
| $s_4(k)$ | 5-Simplex | 0.00 |

(b) Source properties for benchmark simulation with symmetric orthogonalisation approach

| Source | polytope | $\mathbb{Q}$-improperness measure ($r_q$) |
|---|---|---|
| $s_1(k)$ | Cubic | 0.01 |
| $s_2(k)$ | Dicyclic (5 point) | 0.01 |
| $s_3(k)$ | Cyclic (2 point) | 1.00 |
| $s_4(k)$ | Cyclic (3 point) | 0.34 |

ure 7.4, with the reference and ground electrodes placed respectively on the right earlobe and forehead. The electrodes used were the AF7, AF8, AF3, AF4, ML, MR, C3, C4, PO7, PO8, PO3 and PO4, where the ML and MR electrodes were placed respectively on the left and right mastoid. In addition, the voltage difference between the two pairs of electrodes placed above and to the side of the eye sockets measured the electrooculogram (EOG), that is, the electrical activity due to eye blinks and eye movement.

The 4-tuple quaternion-valued EEG signals were formed from four symmetric electrodes from the frontal (AF7, AF8, AF3, AF4), central (ML, MR, C3, C4) and occipital (PO7, PO8, PO3, PO4) regions of the head. The $\mathbb{Q}$-improper quaternion signals were constructed as

$$\begin{aligned} x_1(k) &= \mathrm{AF8}(k) + \imath\, \mathrm{AF4}(k) + \jmath\, \mathrm{AF3}(k) + \kappa\, \mathrm{AF7}(k) \\ x_2(k) &= \mathrm{MR}(k) + \imath\, \mathrm{C4}(k) + \jmath\, \mathrm{C3}(k) + \kappa\, \mathrm{ML}(k) \\ x_3(k) &= \mathrm{PO8}(k) + \imath\, \mathrm{PO4}(k) + \jmath\, \mathrm{PO3}(k) + \kappa\, \mathrm{PO7}(k) \end{aligned} \tag{7.35}$$

and the observed EEG mixture at time instant $k$ were then represented by the vector $\mathbf{x} = [x_1(k),\, x_2(k),\, x_3(k)]^T$. The degree of $\mathbb{Q}$-impropropriety of the signals were respectively 0.89, 0.68 and 0.89, according to (7.32). In this scheme, the quaternion FastICA algorithm (7.29) was first utilised to estimate the source signals, with the step-size $\mu = 1$ and initial Lagrange parameter $\lambda = 5$, while the nonlinearity was chosen as $G(y) = \log\cosh(y)$, to provide good overall performance. Next, the estimated source pertaining to the EOG artifact was selected through examination of the kurtosis values

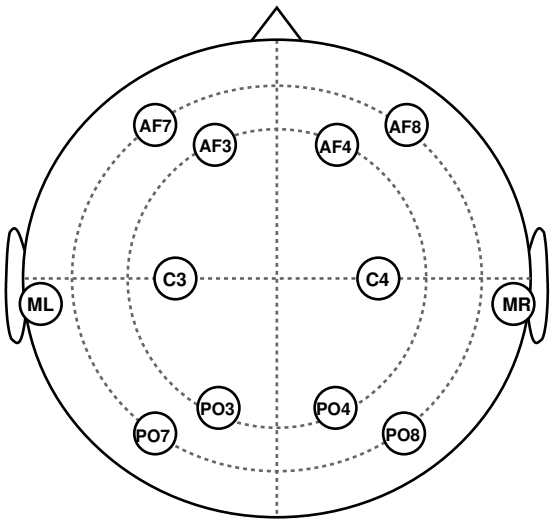
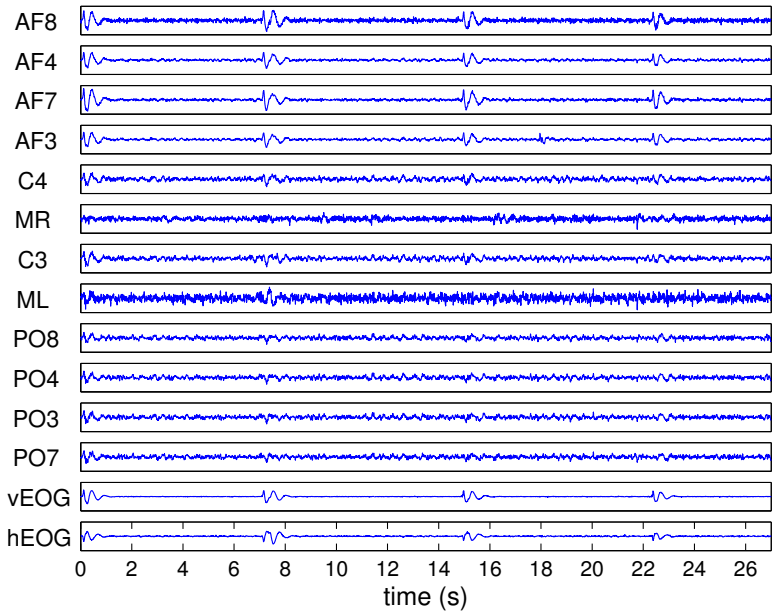



**Figure 7.4** Placement of the EEG recording electrodes.

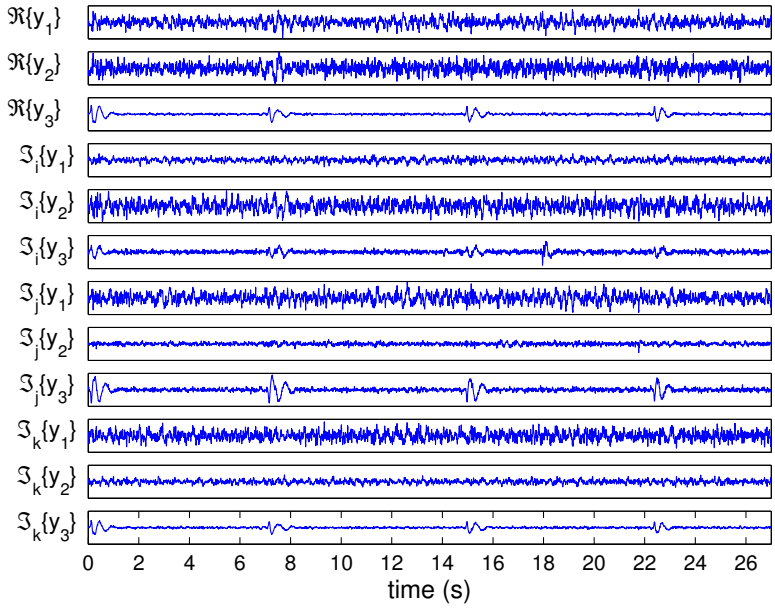
of the components of the separated sources. Pure EEG signals typically have near-zero kurtosis values, while the EOG artifacts have super-Gaussian distributions and thus large kurtosis values [114], this being attributed to the sparse nature of eye blinks.

A time plot of the original recorded channels and the components of the quaternion-valued separated sources are depicted respectively in Figure 7.5(a) and Figure 7.5(b). The occurrence of the eye blinks can be seen at the beginning of the recording, then at around 7s, 15s and 22s, where the effect of the EOG artifact was more prominent on the frontal lobe channel, and less severe in the central and occipital channels. By visual inspection, the separated EOG artifact can be seen to span the components of the third extracted source $y_3(k)$, that is $\Re\{y_3(k)\}, \Im_i\{y_3(k)\}, \Im_j\{y_3(k)\}, \Im_\kappa\{y_3(k)\}$; this is confirmed through comparison of the kurtosis values of each components (Figure 7.5(c)). While most estimated sources had a near-zero measure of kurtosis, the real and imaginary components of $y_3(k)$ have, in comparison, very large kurtosis values.

To study the effectiveness of the algorithm in removing the artifact, the components of $y_3(k)$ were reconstructed to form the EOG signal and then compared to the original combined EOG recording. Figure 7.5(d) depicts both the signals along with the residual error of the estimation process, having a mean square error of $1.21 \times 10^{-4}$. Also, by excluding the components of $y_3(k)$ the clean EEG mixture was reconstructed and a 3s window between 6s–9s for each reconstructed channel is shown in Figure 7.5(e), where the effect of the EOG present at 7s was diminished in the channels.

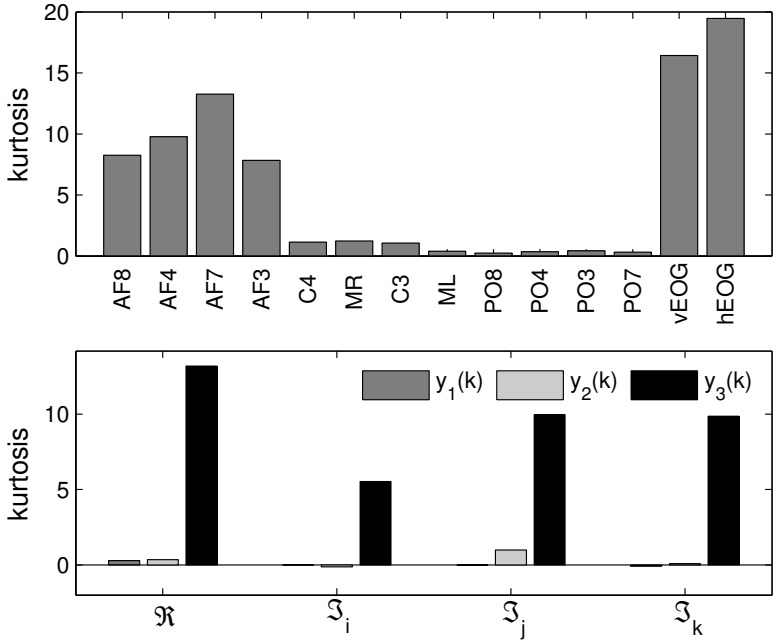



(a) The recorded EEG and EOG channels.




(b) The components of the estimated sources.

**Figure 7.5** Removal of EOG artifact from an EEG recording using the quaternion FastICA algorithm




(c) Kurtosis values of the recorded EEG channels, *bottom:* Kurtosis values of each component of the estimated quaternion-valued sources.

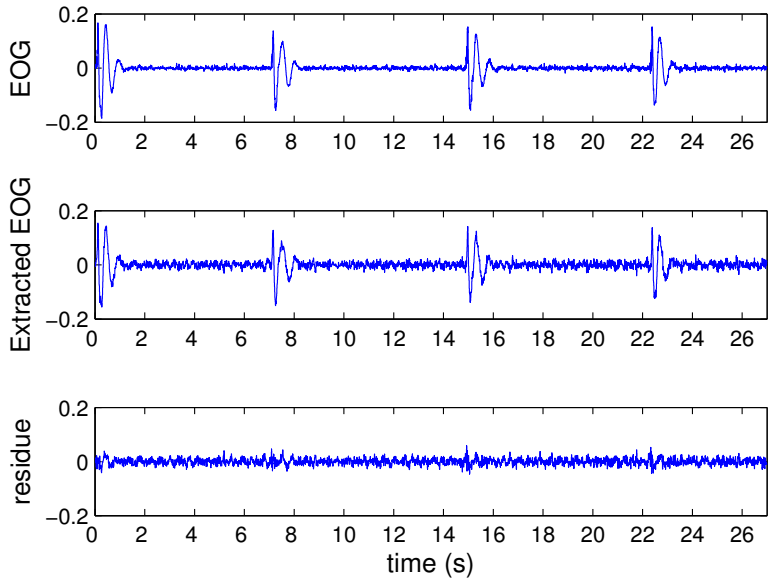
**Figure 7.5** Continued

## 7.5 Summary

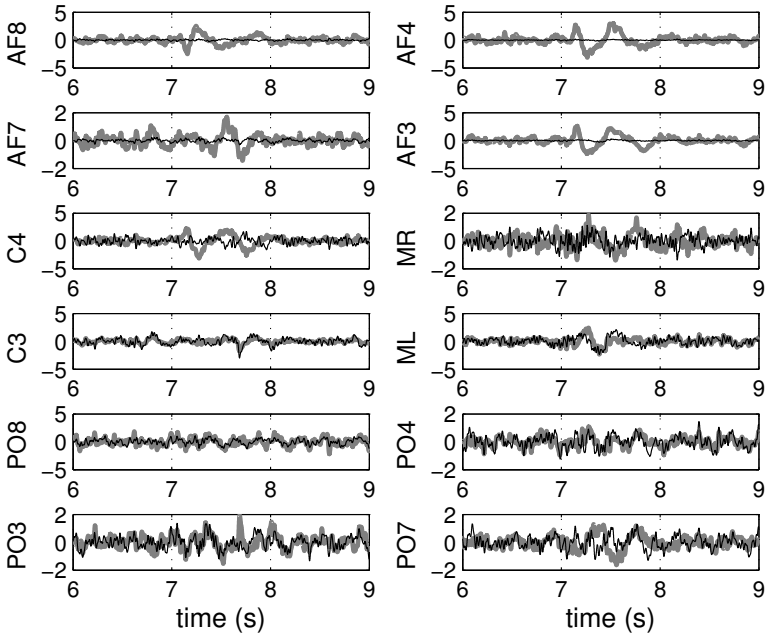
An ICA algorithm suitable for the blind separation of both $\mathbb{Q}$-proper and $\mathbb{Q}$-improper sources has been introduced. The well-known negentropy-based cost function has been utilised to estimate independent quaternion-valued sources, while an augmented Newton method implementation has allowed for the extension of the FastICA methodology to the quaternion domain. The performance of the quaternion FastICA (q-FastICA) algorithm in deflationary and simultaneous separation using benchmark quaternion polytope signals has been discussed, and the algorithm has been shown to be effective in the removal of ocular artifacts from EEG signals.

## 7.A Some relevant results from $\mathbb{HR}$ calculus

Several results, used in the derivation of the q-FastICA algorithm (7.29) are discussed here.




(d) The original and reconstructed EOG signals, along with the residual estimation error.




(e) The original recorded EEG (thick gray line) and clean EEG mixture after artifact removal (thin black line), shown between 6s–9s.

**Figure 7.5** Continued

### 7.A.1 Chain rule in $\mathbb{HR}$ calculus

For a quaternion composite function $F \circ G = F(G(q)) : \mathbb{H} \mapsto \mathbb{H}$, the chain rule is expressed as

$$\frac{\partial F}{\partial \xi} = \frac{\partial F}{\partial G}\frac{\partial G}{\partial \xi} + \frac{\partial F}{\partial G^{\imath}}\frac{\partial G^{\imath}}{\partial \xi} + \frac{\partial F}{\partial G^{\jmath}}\frac{\partial G^{\jmath}}{\partial \xi} + \frac{\partial F}{\partial G^{\kappa}}\frac{\partial G^{\kappa}}{\partial \xi} \tag{7.36}$$

and $\xi = \{q, q^{\imath}, q^{\jmath}, q^{\kappa}\}$. To show this, the total differential of $F(\bar{q})$ can be written as [133, 134]

$$dF = \frac{\partial F}{\partial \bar{q}}d\bar{q} + \frac{\partial F}{\partial \bar{q}^{\imath}}d\bar{q}^{\imath} + \frac{\partial F}{\partial \bar{q}^{\jmath}}d\bar{q}^{\jmath} + \frac{\partial F}{\partial \bar{q}^{\kappa}}d\bar{q}^{\kappa} \tag{7.37}$$

where the dummy variable $\bar{q} \triangleq G(q)$. Likewise, the total differential for $G(q)$ is given by

$$dG = \frac{\partial G}{\partial q}dq + \frac{\partial G}{\partial q^{\imath}}dq^{\imath} + \frac{\partial G}{\partial q^{\jmath}}dq^{\jmath} + \frac{\partial G}{\partial q^{\kappa}} \tag{7.38}$$

By substituting (7.38) into (7.37), and after rearranging the expressions, the total differential of $F$ with respect to $q$ is obtained as

$$\begin{aligned} dF = &\left(\frac{\partial F}{\partial G}\frac{\partial G}{\partial q} + \frac{\partial F}{\partial G^{\imath}}\frac{\partial G^{\imath}}{\partial q} + \frac{\partial F}{\partial G^{\jmath}}\frac{\partial G^{\jmath}}{\partial q} + \frac{\partial F}{\partial G^{\kappa}}\frac{\partial G^{\kappa}}{\partial q}\right)dq \\ &+ \left(\frac{\partial F}{\partial G}\frac{\partial G}{\partial q^{\imath}} + \frac{\partial F}{\partial G^{\imath}}\frac{\partial G^{\imath}}{\partial q^{\imath}} + \frac{\partial F}{\partial G^{\jmath}}\frac{\partial G^{\jmath}}{\partial q^{\imath}} + \frac{\partial F}{\partial G^{\kappa}}\frac{\partial G^{\kappa}}{\partial q^{\imath}}\right)dq^{\imath} \\ &+ \left(\frac{\partial F}{\partial G}\frac{\partial G}{\partial q^{\jmath}} + \frac{\partial F}{\partial G^{\imath}}\frac{\partial G^{\imath}}{\partial q^{\jmath}} + \frac{\partial F}{\partial G^{\jmath}}\frac{\partial G^{\jmath}}{\partial q^{\jmath}} + \frac{\partial F}{\partial G^{\kappa}}\frac{\partial G^{\kappa}}{\partial q^{\jmath}}\right)dq^{\jmath} \\ &+ \left(\frac{\partial F}{\partial G}\frac{\partial G}{\partial q^{\kappa}} + \frac{\partial F}{\partial G^{\imath}}\frac{\partial G^{\imath}}{\partial q^{\kappa}} + \frac{\partial F}{\partial G^{\jmath}}\frac{\partial G^{\jmath}}{\partial q^{\kappa}} + \frac{\partial F}{\partial G^{\kappa}}\frac{\partial G^{\kappa}}{\partial q^{\kappa}}\right)dq^{\kappa} \end{aligned} \tag{7.39}$$

where the derivatives $\frac{\partial F}{\partial \xi}$ are given by the terms within the brackets, and form the chain rule. The chain rule for the $\mathbb{HR}^*$ derivatives can be obtained similarly, and the result of (7.36) can be extended to vector-valued functions to form a generalised chain rule for the derivatives.

## 7.B The Augmented quaternion Newton method

The duality between $\mathbb{R}^4$ and $\mathbb{H}$ allows for the consideration of the relations between the derivatives in the two domains. This methodology was previously considered in [50] and resulted in the derivation of the augmented complex Newton method. The extension of this work to the quaternion domain based on the involution bases was detailed in [134, 133]. A short summary is presented below.

For a function $f(\mathbf{q}) : \mathbb{H}^N \mapsto \mathbb{R}$, its augmented gradient $\nabla_{\mathbf{q}^{a*}} f = \frac{\partial f}{\partial \mathbf{q}^{a*}}$ and Hessian $\mathbf{H}^a_{\mathbf{qq}} = \frac{\partial}{\partial \mathbf{q}^{a*}} \left(\frac{\partial f}{\partial \mathbf{q}^{a*}}\right)^T$, where the augmented vector $\mathbf{q}^a = [\mathbf{q}^T, \mathbf{q}^{\imath T}, \mathbf{q}^{\jmath T}, \mathbf{q}^{\kappa T}]^T$. The augmented Newton update can then be written as

$$\Delta \mathbf{q}^a = -\left(\mathbf{H}^a_{\mathbf{qq}}\right)^{-1} \cdot \nabla_{\mathbf{q}^{a*}} f, \tag{7.40}$$

where $\Delta \mathbf{q}^a = \mathbf{q}^a(k+1) - \mathbf{q}^a(k)$ is the change in $\mathbf{q}^a$ in each consecutive update.

Finally, observe that the elements of the augmented Hessian matrix

$$\mathbf{H}^a_{\mathbf{qq}} = \begin{bmatrix} \mathbf{H}_{\mathbf{q}^*\mathbf{q}^*} & \mathbf{H}_{\mathbf{q}^\imath{}^*\mathbf{q}^*} & \mathbf{H}_{\mathbf{q}^\jmath{}^*\mathbf{q}^*} & \mathbf{H}_{\mathbf{q}^\kappa{}^*\mathbf{q}^*} \\ \mathbf{H}_{\mathbf{q}^*\mathbf{q}^\imath{}^*} & \mathbf{H}_{\mathbf{q}^\imath{}^*\mathbf{q}^\imath{}^*} & \mathbf{H}_{\mathbf{q}^\jmath{}^*\mathbf{q}^\imath{}^*} & \mathbf{H}_{\mathbf{q}^\kappa{}^*\mathbf{q}^\imath{}^*} \\ \mathbf{H}_{\mathbf{q}^*\mathbf{q}^\jmath{}^*} & \mathbf{H}_{\mathbf{q}^\imath{}^*\mathbf{q}^\jmath{}^*} & \mathbf{H}_{\mathbf{q}^\jmath{}^*\mathbf{q}^\jmath{}^*} & \mathbf{H}_{\mathbf{q}^\kappa{}^*\mathbf{q}^\jmath{}^*} \\ \mathbf{H}_{\mathbf{q}^*\mathbf{q}^\kappa{}^*} & \mathbf{H}_{\mathbf{q}^\imath{}^*\mathbf{q}^\kappa{}^*} & \mathbf{H}_{\mathbf{q}^\jmath{}^*\mathbf{q}^\kappa{}^*} & \mathbf{H}_{\mathbf{q}^\kappa{}^*\mathbf{q}^\kappa{}^*} \end{bmatrix} \tag{7.41}$$

can be written in terms of its first row by utilising the involution property (7.20) and noting that $\left((\cdot)^\alpha\right)^\beta = (\cdot)^\gamma, \alpha \neq \beta \neq \gamma = \{\imath, \jmath, \kappa\}$.

## 7.C Derivation of the augmented q-FastICA update algorithm

### 7.C.1 First and second derivatives of the cost function $\mathcal{J}(\mathbf{w})$

The first and second derivatives of the q-FastICA cost function given in Equation 7.27 are now derived. For simplicity, the equation is reproduced here, and is given by

$$\mathcal{J}(\mathbf{w}, \mathbf{w}^\imath, \mathbf{w}^\jmath, \mathbf{w}^\kappa) = E\{G(|\mathbf{w}^H\mathbf{x}|^2)\} = E\{G(|y|^2)\}. \tag{7.42}$$

and $y = \mathbf{w}^H\mathbf{x}$.

First, by using the product rule, the derivatives of the involutions of $|y|^2 = yy^* = |\mathbf{w}^H\mathbf{x}|^2$ with respect to the conjugate demixing vector $\mathbf{w}^*$ are calculated as

$$\begin{aligned} \frac{\partial yy^*}{\partial \mathbf{w}^*} &= \frac{\partial y}{\partial \mathbf{w}^*} y^* + y \frac{\partial y^*}{\partial \mathbf{w}^*} = \mathbf{x}y^* - \frac{1}{2} y\mathbf{x}^* \\ \frac{\partial (yy^*)^\imath}{\partial \mathbf{w}^*} &= \frac{\partial y^\imath}{\partial \mathbf{w}^*} y^{\imath *} + y^\imath \frac{\partial y^{\imath *}}{\partial \mathbf{w}^*} = \frac{1}{2} y^\imath \mathbf{x}^{\imath *} \\ \frac{\partial (yy^*)^\jmath}{\partial \mathbf{w}^*} &= \frac{\partial y^\jmath}{\partial \mathbf{w}^*} y^{\jmath *} + y^\jmath \frac{\partial y^{\jmath *}}{\partial \mathbf{w}^*} = \frac{1}{2} y^\jmath \mathbf{x}^{\jmath *} \\ \frac{\partial (yy^*)^\kappa}{\partial \mathbf{w}^*} &= \frac{\partial y^\kappa}{\partial \mathbf{w}^*} y^{\kappa *} + y^\kappa \frac{\partial y^{\kappa *}}{\partial \mathbf{w}^*} = \frac{1}{2} y^\kappa \mathbf{x}^{\kappa *}. \end{aligned} \tag{7.43}$$

Then by using the chain rule (7.36) and after simplification the gradients of the cost function are obtained as

$$\begin{aligned} \nabla_{\mathbf{w}^*} \mathcal{J} &= E\{2g(|y|^2)\mathbf{x}y^*\} \\ \nabla_{\mathbf{w}^{\imath *}} \mathcal{J} &= E\{2g(|y|^2)\mathbf{x}y^{\imath *}\} \\ \nabla_{\mathbf{w}^{\jmath *}} \mathcal{J} &= E\{2g(|y|^2)\mathbf{x}y^{\jmath *}\} \\ \nabla_{\mathbf{w}^{\kappa *}} \mathcal{J} &= E\{2g(|y|^2)\mathbf{x}y^{\kappa *}\} \end{aligned} \tag{7.44}$$

where $g$ is the first derivative of $G$; this result can also be interpreted based on the involution property (7.20).

After some simplifications and considering the whiteness of $\mathbf{x}$, the second derivatives of $\mathcal{J}$ can then be calculated as

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{w}^*}\left(\frac{\partial \mathcal{J}}{\partial \mathbf{w}^*}\right)^T &= E\{4g'(|y|^2)\mathbf{x}y^*\mathbf{x}^T y^* - g(|y|^2)\mathbf{I}\} \\
\frac{\partial}{\partial \mathbf{w}^{\imath *}}\left(\frac{\partial \mathcal{J}}{\partial \mathbf{w}^*}\right)^T &= E\{2g'(|y|^2)(\mathbf{x}y^*)^{\imath}(\mathbf{x}^T y^*) + g(|y|^2)\mathbf{I}\} \\
\frac{\partial}{\partial \mathbf{w}^{\jmath *}}\left(\frac{\partial \mathcal{J}}{\partial \mathbf{w}^*}\right)^T &= E\{2g'(|y|^2)(\mathbf{x}y^*)^{\jmath}(\mathbf{x}^T y^*) + g(|y|^2)\mathbf{I}\} \\
\frac{\partial}{\partial \mathbf{w}^{\kappa *}}\left(\frac{\partial \mathcal{J}}{\partial \mathbf{w}^*}\right)^T &= E\{2g'(|y|^2)(\mathbf{x}y^*)^{\kappa}(\mathbf{x}^T y^*) + g(|y|^2)\mathbf{I}\},
\end{aligned} \tag{7.45}$$

where $g'$ is the second derivative of $G$ and the calculations of the remaining derivatives follow from property (7.20). Finally, notice that the non-commutativity of the quaternion product prohibits further simplification of the derivatives in (7.45).

### 7.C.2 The augmented Newton update

The Lagrangian function $\mathcal{L}$ for the optimisation problem in (7.28) is given by

$$\mathcal{L}(\mathbf{w}, \lambda) = \mathcal{J}(\mathbf{w}) + \underbrace{\lambda(\mathbf{w}^H\mathbf{w} - 1)}_{\triangleq c} \tag{7.46}$$

where $\lambda \in \mathbb{R}$ is the Lagrange parameter. The Newton method (7.40) is utilised to find the extrema of (7.46), where

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{w}^{a*}} &= \frac{\partial \mathcal{J}}{\partial \mathbf{w}^{a*}} + \frac{\partial c}{\partial \mathbf{w}^{a*}} \\
\frac{\partial}{\partial \mathbf{w}^{a*}}\left(\frac{\partial \mathcal{L}}{\partial \mathbf{w}^{a*}}\right)^T &= \mathbf{H}^a_{\mathbf{ww}} + \frac{\partial}{\partial \mathbf{w}^{a*}}\left(\frac{\partial c}{\partial \mathbf{w}^{a*}}\right)^T
\end{aligned} \tag{7.47}$$

and the augmented gradient and Hessian of $\mathcal{J}$ are obtained using (7.44) and (7.45). The gradients of $c$ are then given by

$$\begin{aligned}
\frac{\partial c}{\partial \mathbf{w}^*} &= \lambda(\mathbf{w} - \frac{1}{2}\mathbf{w}^*) \\
\frac{\partial c}{\partial \mathbf{w}^{\imath *}} &= \frac{\lambda}{2}\mathbf{w}^* \\
\frac{\partial c}{\partial \mathbf{w}^{\jmath *}} &= \frac{\lambda}{2}\mathbf{w}^* \\
\frac{\partial c}{\partial \mathbf{w}^{\kappa *}} &= \frac{\lambda}{2}\mathbf{w}^*
\end{aligned} \tag{7.48}$$

and the Hessian can be calculated from

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{w}^*}\left(\frac{\partial c}{\partial \mathbf{w}^*}\right)^T &= -\lambda \mathbf{I} \\
\frac{\partial}{\partial \mathbf{w}^{\imath *}}\left(\frac{\partial c}{\partial \mathbf{w}^*}\right)^T &= -\frac{\lambda}{2}\mathbf{I} \\
\frac{\partial}{\partial \mathbf{w}^{\jmath *}}\left(\frac{\partial c}{\partial \mathbf{w}^*}\right)^T &= -\frac{\lambda}{2}\mathbf{I} \\
\frac{\partial}{\partial \mathbf{w}^{\kappa *}}\left(\frac{\partial c}{\partial \mathbf{w}^*}\right)^T &= -\frac{\lambda}{2}\mathbf{I}.
\end{aligned} \tag{7.49}$$

By substituting these results in (7.40), the Newton update for the Lagrangian is obtained. Finally, the Lagrange parameter $\lambda$ is updated using a gradient ascent method, whereby at each iteration the demixing vector $\mathbf{w}$ is first updated via the augmented Newton method, followed by the update of $\lambda$ using the current value of $\mathbf{w}$ and normalisation of the demixing vector [144], as in (7.29).

# Chapter 8

# Conclusions and Future Work

## 8.1 Conclusions

In this thesis, a class of algorithms suitable for the processing of the generality of complex-valued signals has been introduced, analysed and tested in practical applications. This has been achieved based on a novel statistical model of complex-valued signals, so called augmented complex statistics. Derivation and analysis of the derived algorithms have been performed using the $\mathbb{CR}$ calculus, which allows for the consideration of non-analytic functions, such as the real-valued error power commonly found in signal processing problems, that is, without the restrictions due to the standard Cauchy-Riemann equations.

This work has addressed both supervised and blind complex algorithms and their usefulness has been shown through the analysis and simulations on benchmark complex-valued signals, as well as on real-world signals including complex wind vectors and EEG signals made complex by convenience of representation.

One of the main aims of this thesis was the development of blind source extraction algorithms for the estimation of complex-valued sources based on fundamental signal properties. While recent research in complex domain blind source separation has resulted in the extension and generalisation of topics and methodologies from the real domain, the exploitation of fundamental signal properties as a means of signal extraction has not been widely explored. Therefore, algorithms based on the predictability, degree of Gaussianity and smoothness of complex-valued signals have been a focus of this work. The application of these algorithms in noise-free and noisy environments has been assessed using both qualitative and quantitative measures, and supported by theoretical analysis.

As a generalisation, the introduced complex domain blind source separation models have been extended to the higher dimensional quaternion domain. This has been

achieved based on the recently introduced widely linear quaternion model [132, 145], effectively demonstrating the generalisation of the complex-valued concepts discussed in this work.

A summary of the contributions in this thesis is given below.

1. The augmented (widely linear) complex least mean square (ACLMS) algorithm has been derived based on a widely linear model. Unlike the standard complex least mean square (CLMS) algorithm which was based on a strictly linear model, the full second-order statistical model of the signal is captured by the ACLMS algorithm.

   It has been shown that the CLMS algorithm is a special case of the ACLMS algorithm and provides optimal performance for only proper complex signals, while the ACLMS algorithm is capable of processing both complex proper and improper signals. The simplicity of the $\mathbb{CR}$ calculus framework in the derivation of the algorithm directly in the complex domain has also been highlighted.

2. A local widely linear prediction based complex blind source extraction (P-cBSE) algorithm using the temporal structure of complex-valued signals has been introduced. By using a modified cost function, the algorithm extracts sources based on the normalised mean square prediction error and is capable of extracting desired sources from mixtures with additive complex-valued noise. Both direct solutions and those requiring prewhitening have been provided, and the existence and uniqueness of the solutions for both cases have also been considered. The normalised mean square prediction error is measured at the output of a widely linear predictor, thus catering for the generality of complex-valued sources, both circular and noncircular. Simulations have demonstrated the enhanced extraction performance of the proposed P-cBSE algorithm, compared to existing complex extraction algorithms based on a standard linear model.

3. A blind source extraction algorithm based on the kurtosis of complex-valued signals (K-cBSE algorithm) has been derived. The algorithm is a modified cost function that is capable of extracting sources with different dynamic ranges. By removing the bias associated with additive complex-valued noise from the cost function, the algorithm is shown to be capable of operating in both noisy and noise-free environments. The existence and uniqueness of the solution have also been addressed; iÃ§n addition, it has been shown that the algorithm is unaffected by the degree of circularity of the additive noise. To enhance the performance, variable step-size variants of the algorithm have been derived, and have been shown to outperform the fixed step-size variants. The application of the K-cBSE algorithm in real-time removal of artifacts from complex-valued

EEG mixtures has been demonstrated and verified using both qualitative and quantitative metrics.

4. The smoothness based complex blind source extraction (S-cBSE) algorithm has been introduced. The concept of smoothness in the complex domain has been discussed, and a constrained cost function has been defined based on the maximisation of non-Gaussianity and the definition of complex smoothness. By utilising the augmented Newton method, the algorithm has been derived based on a constrained generalised complex FastICA (nc-FastICA) algorithm, thus resulting in fast convergence, and ability to extract both complex proper and improper latent sources. For comparison, the algorithm has also been derived based on the standard complex FastICA algorithm. Simulations have shown that the S-cBSE algorithm based on the generalised complex FastICA algorithm is capable of extracting the desired smooth (or, non-smooth) sources successfully, while the S-cBSE algorithm using the c-FastICA algorithm results in poor performance. The S-cBSE algorithm has been successfully utilised for the extraction of power line noise, eye blinks and eye movements from EEG recordings, demonstrating its application in real-world problems.

5. A quaternion FastICA (q-FastICA) algorithm has been derived for the separation of the generality of quaternion-valued sources. Based on recent advancements in augmented quaternion statistics and so called $\mathbb{HR}$ calculus, the statistical and analytical concepts discussed for complex domain signal processing have been extended to the quaternion domain. The q-FastICA algorithm is based on the maximisation of non-Gaussianity by utilising suitable nonlinearities for the approximations of the negentropy function. The derivation of the algorithm uses the recently introduced $\mathbb{HR}$ calculus and employs the augmented Newton method for quaternion functions. The assessment of the performance of the algorithm using both quaternion proper and improper four dimensional polytopes has demonstrated successful source separation, and an application in separation of pure EEG and arifactual sources support the analysis.

## 8.2 Future work

The foundation of this work is based on the augmented complex statistics and the $\mathbb{CR}$ calculus framework. The areas for the extension of the work presented in this thesis, include

1. *Complex blind source separation using Canonical Correlation Analysis* — Blind source separation based on the canonical correlation analysis (CCA) approach has been

previously explored in the real domain, and analytical studies of its performance have been provided, e.g. in [146, 147]. In the real domain, online blind source separation using CCA is shown to be closely related to blind source separation using a linear predictor. In this work, blind source extraction based on the temporal structure of sources and using a widely linear predictor has been proposed, the P-cBSE algorithm. It is therefore possible to explore the CCA approach in complex blind source separation and provide a link with the P-cBSE algorithm.

In the real domain, blind source separation using the CCA approach relies on maximising the correlation of two linear combinations of variables with a joint distribution. In the complex domain, it is necessary to consider both the correlation and pseudo-correlation of complex-valued linear combinations. In addition, by using the weighted sum of such linear combinations, the widely linear predictor is expected to result in optimal second-order performance. Further work will include analysis of the existence and convergence of the algorithm, as well as the derivation of cost functions suitable for blind separation from noisy mixtures.

2. *Prediction based quaternion blind source extraction* — Blind source separation in the quaternion domain is currently in its early stages [148], the extension of the P-cBSE algorithm to the quaternion domain would allow for the extraction of both quaternion proper and improper sources from both noise-free and noisy mixtures. Analysis of the mean square prediction error of quaternion signals can provide insight into the operation of the algorithm, and a quaternion widely linear predictor can be ultimately utilised for the implementation of an online extraction algorithm.

   A widely linear quaternion predictor based on the LMS algorithm has been recently introduced in [135] and has shown enhanced performance for improper signals over the standard quaternion predictor, making it suitable for quaternion blind source extraction based on the temporal structure of the signals. Study of quaternion-valued noise will also allow for the design of more robust cost functions, such that the resulting algorithms will be capable of extracting sources from noisy mixtures.

3. *Post-nonlinear complex blind source separation* — In this work, a linear mixture model has been considered for complex blind source separation. This assumption can be generalised to consider post-nonlinear mixtures, using complex nonlinear functions. The effect of split- and fully-complex models can be compared, where it is expected that a fully-complex nonlinear function result in the best model. A simple extraction method can be based on a nonlinear widely linear predictor, where the nonlinearity may be estimated in a prior stage.

Finally, in the real domain, while it is possible to separate latent sources based on a post-nonlinear model, separation of sources based on a nonlinear model is considered to result in non-unique solutions [14]. This study can be extended to the case of complex sources passed through a fully-complex nonlinearity, where it may be possible to exploit information on the degree of noncircularity of sources to aid in blind separation from complex nonlinear mixtures.

# Appendix A

# The Complex Generalised Gaussian Distribution

The generalised Gaussian distribution (GGD) consists of a family of distributions whose deviation from the standard Gaussian ('normal') distribution are determined via a shape parameter. Variation in the parameters result in a range of distributions with negative kurtosis (sub-Gaussian distribution), zero kurtosis (Gaussian distribution) and positive kurtosis (super-Gaussian distribution). The extension of this family of distributions to the complex domain is provided here. As a special case, the complex Gaussian distribution is introduced and discussed.

Consider a complex random variable $\mathbf{z} = \mathbf{z}_r + \jmath\mathbf{z}_i \in \mathbb{C}^N$, where the distribution of its real and imaginary components can be considered as a real-valued multivariate GGD given by [149, 150]

$$f_{\mathbf{Z}_r,\mathbf{Z}_i}(\mathbf{z}_r,\mathbf{z}_i) = f_{\mathbf{Z}^R}(\mathbf{z}^R) = \alpha \exp\Big( -\big(\gamma(\mathbf{z}^R-\boldsymbol{\mu})^T\mathcal{C}_{\mathbf{zz}}^{R^{-1}}(\mathbf{z}^R-\boldsymbol{\mu})\big)^c\Big) \tag{A.1}$$

$$\alpha = \frac{c\gamma}{\pi^N\Gamma(\frac{1}{c})\big(\det(\mathcal{C}_{\mathbf{zz}}^R)\big)^{\frac{1}{2}}}$$

$$\gamma = \frac{\Gamma(\frac{2}{c})}{2\Gamma(\frac{1}{c})}$$

where $c$ is the shape parameter, $\Gamma(\cdot)$ is the Gamma function, $\boldsymbol{\mu}$ is the statistical mean vector and $\det(\cdot)$ denotes the matrix determinant operator. The covariance matrix $\mathcal{C}_{\mathbf{zz}}^R$ is defined in (2.5) and defines the second-order statistical properties of the distribution, and

$$\mathbf{z}^R = \begin{bmatrix}\mathbf{z}_r\\ \mathbf{z}_i\end{bmatrix} = \frac{1}{2}\mathbf{J}^H\mathbf{z}^a \in \mathbb{R}^{2N}. \tag{A.2}$$

By utilising the duality [51] established between $\mathbb{C}^2$ and $\mathbb{R}^2$ in Section 2.1.3, the multivariate GGD can be expressed as

$$f_{\mathbf{Z}_r,\mathbf{Z}_i}(\mathbf{z}_r,\mathbf{z}_i) = \alpha \exp\Big( - \big(\gamma(\frac{1}{2}\mathbf{J}^H\mathbf{z}^a)^H(\frac{1}{4}\mathbf{J}^H\mathcal{C}^a_{\mathbf{zz}}\mathbf{J})^{-1}(\frac{1}{2}\mathbf{J}^H\mathbf{z}^a)\big)^c\Big) \tag{A.3}$$

where the relations in Equations (A.2) and (2.11) is used. Noting that $\frac{1}{2}\mathbf{J}\mathbf{J}^H = \mathbf{I}$ and the expressions (2.8) on the relation between the real and imaginary components with the complex random vector and its conjugate, the distribution is then written as

$$f_{\mathbf{Z},\mathbf{Z}^*}(\mathbf{z},\mathbf{z}^*) = \alpha \exp\Big( - \big(\gamma \mathbf{z}^{aH}\mathcal{C}^{a^{-1}}_{\mathbf{zz}}\mathbf{z}^a\big)^c\Big) \tag{A.4}$$
$$\alpha = \frac{c\gamma}{(\frac{\pi}{2})^N\Gamma(\frac{1}{c})\big(\det(\mathcal{C}^a_{\mathbf{zz}})\big)^{\frac{1}{2}}}.$$

This completes the derivation of the complex generalised Gaussian distribution (c-GGD). Thus, while the distribution in (A.1) provides a valid model for the distribution of a complex random vector, the derived pdf (A.4) results in a more natural model, applicable directly in $\mathbb{C}$. The statistical properties of the c-GGD are dictated by the shape parameter $c$ and the augmented covariance matrix $\mathcal{C}^a_{\mathbf{zz}}$. For the range of values $0 < c < 1$, the distribution is super-Gaussian, for $c = 1$ it is Gaussian and for $c > 1$ it is sub-Gaussian. Likewise, the second-order circularity of the random vector is chosen by designing[1] a suitable augmented covariance matrix $\mathcal{C}^a_{\mathbf{zz}}$.

## A.1 The Complex Gaussian Distribution

A special case of the complex Gaussian distribution is obtained from the c-GGD pdf (A.4) with shape parameter $c = 1$. Its pdf is then given by [51]

$$f_{\mathbf{Z},\mathbf{Z}^*}(\mathbf{z},\mathbf{z}^*) = \frac{1}{\pi^N\big(\det(\mathcal{C}^a_{\mathbf{zz}})\big)^{\frac{1}{2}}}\exp\Big(-\frac{1}{2}\mathbf{z}^{aH}\mathcal{C}^{a^{-1}}_{\mathbf{zz}}\mathbf{z}^a\Big). \tag{A.5}$$

It is noteworthy that this result was derived by van den Bos in [51] by considering the multivariate Gaussian pdf and introducing the transformation matrix $\mathbf{J}$ to map between the real and complex domains.

For further insight, consider the simple case of a scalar random variable $z = z_r + \jmath z_i$, where $N = 1$. After simplification, the pdf (A.5) can be expressed as

$$f_{Z,Z^*}(z,z^*) = \frac{1}{\pi\sigma_{z_r}\sigma_{z_i}\sqrt{1-\varrho^2}}\exp\left(-\frac{(z+z^*)^2}{4\sigma^2_{z_r}} - \jmath\varrho\frac{(z^2-z^{*2})}{2\sigma_{z_r}\sigma_{z_i}} + \frac{(z-z^*)^2}{4\sigma^2_{z_i}}\right) \tag{A.6}$$

where $\sigma_{z_r}$ and $\sigma_{z_i}$ are the standard deviations of the real and imaginary components and $\varrho = \frac{\sigma_{z_r,z_i}}{\sigma_{z_r}\sigma_{z_i}}$ is the correlation coefficient. Scatter plots of two Gaussian random variables with different second-order statistics are illustrated in 2.1.

[1] In [151], the authors detail the generation of samples with a desired c-GGD.

Given a proper random variable, the real and imaginary components are uncorrelated and with equal variance, that is, the correlation coefficient $\varrho = 0$ and $\sigma_{z_r}^2 = \sigma_{z_i}^2 = \sigma_z^2$. Thus, the pdf of a second-order circular (proper) complex Gaussian random variable becomes

$$f_Z(z) = \frac{1}{\pi\sigma_z^2} \exp\left( - \frac{|z|^2}{\sigma_z^2} \right), \tag{A.7}$$

which is only a function of the magnitude of the random variable, and does not depend on its phase. This is the classic definition of the complex Gaussian pdf [52], which as shown here is actually a restricted case of the complex Gaussian pdf, and does not account for the generality of complex random variables.

Finally, the entropy of a complex Gaussian random vector $\mathbf{z}$ is given by [44]

$$H(\mathbf{z}) \leq \log\left( (\pi e)^N \det(\mathcal{C}_{\mathbf{zz}}) \right). \tag{A.8}$$

This result can be similarly obtained by considering the entropy of the multivariate real-valued Gaussian random vector [152] and establishing the complex equivalent (A.8) through the utilisation of the duality between the two domains. An interesting result, presented by Neeser and Massey in [44], show that the entropy $H(\mathbf{z})$ is maximised for second-order circular (proper) Gaussian random vectors. This can be seen by noting that the determinant of a general augmented covariance matrix is smaller than the determinant of the block diagonal augmented covariance matrix of a proper random vector [48].

# Appendix B

# Brief overview of $\mathbb{CR}$ calculus

A class of functions, which are of special interest in signals processing optimisation problems are real valued functions of complex variables, typically encountered as cost functions based on the error power. However, these functions are non-analytic (non-differentiable) within the stringent conditions set by the Cauchy-Riemann equations, and thus a flexible and generalised calculus framework is needed for their study.

To this end, the so called $\mathbb{CR}$ calculus framework [55, 54] achieves this aim, and is briefly introduced here. The framework was originally introduced by Wirtinger [55] in 1927 and is known as *Wirtinger calculus* within the German speaking engineering community. More recently, the technical notes by Kreutz-Delgado [54] provided a comprehensive overview of the topic, and referred to the framework as $\mathbb{CR}$ calculus due to the dual real and complex perspective of complex functions within this framework.

It is common to consider the function $f(z) : \mathbb{C}^N \mapsto \mathbb{R}$ directly as a function of the complex vector variable $\mathbf{z}$, or as a composite function of its real and imaginary components $\mathbf{z}_r$ and $\mathbf{z}_i$, such that

$$f(\mathbf{z}) = g(\mathbf{z}_r, \mathbf{z}_i) = u(\mathbf{z}_r, \mathbf{z}_i) + \jmath v(\mathbf{z}_r, \mathbf{z}_i). \tag{B.1}$$

Then, the Cauchy-Riemann conditions specify that

$$\frac{\partial u}{\partial \mathbf{z}_r} = \frac{\partial v}{\partial \mathbf{z}_i} \qquad\qquad \frac{\partial v}{\partial \mathbf{z}_r} = -\frac{\partial u}{\partial \mathbf{z}_i}, \tag{B.2}$$

which induces strict conditions on the differentiability of $f(z)$. For example, an analytic function such as $f_1(\mathbf{z}) = \mathbf{z}^2$ satisfies this condition and is complex differentiable where $f_1'(\mathbf{z}) = \frac{\partial u}{\partial \mathbf{z}_r} + \jmath \frac{\partial v}{\partial \mathbf{z}_i} = 2\mathbf{z}$, while $f_2(\mathbf{z}) = \mathbf{z}\mathbf{z}^* = |\mathbf{z}|^2$ does not satisfy the Cauchy-Riemann equations and in this light is not complex differentiable. The common method in circumventing this problem is by considering $f(\mathbf{z})$ in terms of its

composite real-valued function and performing partial derivatives with respect to the real and imaginary components.

Establishing the duality between the real- and complex-valued derivatives in the $\mathbb{CR}$ calculus framework, results in the calculation of the Taylor Series Expansion (TSE) in $\mathbb{R}$ and $\mathbb{C}$. This is especially important for the formulation of first-order optimisation methods such as gradient descent, and second-order optimisation based on the Newton method.

## B.1 $\mathbb{CR}$ calculus

The function $f(\mathbf{z})$ may alternatively be considered a function of $\mathbf{z}$ and $\mathbf{z}^*$, that is $f(\mathbf{z}, \mathbf{z}^*)$. Note that although $\mathbf{z}$ and $\mathbf{z}^*$ are not truly independent, the introduced methodology can be considered as a formalism whereby $f$ is analytic in $\mathbf{z}$ and $\mathbf{z}^*$ is considered fixed, and vice versa $f$ is considered analytic in $\mathbf{z}^*$ while $\mathbf{z}$ is a fixed parameter [54]. In this context, the variables $\mathbf{z}$ and $\mathbf{z}^*$ are termed *conjugate coordinates*, and the representation in (B.1) may be rewritten as

$$f(\mathbf{z}) = f(\mathbf{z}, \mathbf{z}^*) = g(\mathbf{z}_r, \mathbf{z}_i) = u(\mathbf{z}_r, \mathbf{z}_i) + \jmath v(\mathbf{z}_r, \mathbf{z}_i). \tag{B.3}$$

The relation between the derivatives of $f$ with respect to the conjugate coordinates $\mathbf{z}$ and $\mathbf{z}^*$, that is $\frac{\partial f}{\partial \mathbf{z}}$ and $\frac{\partial f}{\partial \mathbf{z}^*}$, and the partial derivatives with respect to the real and imaginary components $\mathbf{z}_r$ and $\mathbf{z}_i$, given by $\frac{\partial f}{\partial \mathbf{z}_r}$ and $\frac{\partial f}{\partial \mathbf{z}_i}$, was proven in [53]. A different approach [63] based on the total differential of $f$ is highlighted below.

The total differential of the function $f(\mathbf{z}) = g(\mathbf{z}_r, \mathbf{z}_i)$ is given by

$$dg(\mathbf{z}_r, \mathbf{z}_i) = \frac{\partial g}{\partial \mathbf{z}_r} d\mathbf{z}_r + \frac{\partial g}{\partial \mathbf{z}_i} d\mathbf{z}_i. \tag{B.4}$$

Thus, after algebraic manipulation using the relation (B.3) and noting that based on the established duality in (2.8), $d\mathbf{z}_r = \frac{1}{2}(d\mathbf{z} + d\mathbf{z}^*)$ and $d\mathbf{z}_i = \frac{1}{2\jmath}(d\mathbf{z} - d\mathbf{z}^*)$, the total differential is given by

$$dg(\mathbf{z}_r, \mathbf{z}_i) = \frac{1}{2}\Big(\frac{\partial g}{\partial \mathbf{z}_r} - \jmath \frac{\partial g}{\partial \mathbf{z}_i}\Big) d\mathbf{z} + \frac{1}{2}\Big(\frac{\partial g}{\partial \mathbf{z}_r} + \jmath \frac{\partial g}{\partial \mathbf{z}_i}\Big) d\mathbf{z}^*, \tag{B.5}$$

or equivalently

$$df(\mathbf{z}) = df(\mathbf{z}, \mathbf{z}^*) = \frac{\partial f}{\partial \mathbf{z}} d\mathbf{z} + \frac{\partial f}{\partial \mathbf{z}^*} d\mathbf{z}^*. \tag{B.6}$$

This leads to one of the important results in $\mathbb{CR}$ calculus, given by

$$\begin{aligned} \mathbb{R}\text{–derivative}: \quad & \frac{\partial f}{\partial \mathbf{z}}\bigg|_{\mathbf{z}^*=const} = \frac{1}{2}\left(\frac{\partial f}{\partial \mathbf{z}_r} - \jmath \frac{\partial f}{\partial \mathbf{z}_i}\right) \\ \mathbb{R}^*\text{–derivative}: \quad & \frac{\partial f}{\partial \mathbf{z}^*}\bigg|_{\mathbf{z}\ =const} = \frac{1}{2}\left(\frac{\partial f}{\partial \mathbf{z}_r} + \jmath \frac{\partial f}{\partial \mathbf{z}_i}\right), \end{aligned} \tag{B.7}$$

where the function $f$ is considered $\mathbb{R}$-analytic, that is, it is differentiable with respect to $\mathbf{z}_r$ and $\mathbf{z}_i$.

Thus, using the paradigm of conjugate coordinates and the relation (B.7), it is possible to consider the derivatives of both analytic and non-analytic complex functions. For an analytic function satisfying the conditions in (B.2), the $\mathbb{R}$-derivatives are simplified such that

$$\begin{aligned}\frac{\partial f}{\partial \mathbf{z}} &= \frac{\partial u}{\partial \mathbf{z}_r} + \jmath\frac{\partial v}{\partial \mathbf{z}_i} = f'(\mathbf{z})\\ \frac{\partial f}{\partial \mathbf{z}^*} &= \mathbf{0}.\end{aligned} \tag{B.8}$$

It is thus concluded that the Cauchy-Riemann conditions in (B.2) can be succinctly written within the $\mathbb{CR}$ calculus framework as

$$\frac{\partial f}{\partial \mathbf{z}^*} = \mathbf{0}. \tag{B.9}$$

The elegance of this framework lies in the fact that when applied to analytic functions, the derivative $\frac{\partial}{\partial \mathbf{z}^*}$ vanishes and so equals the standard complex derivative defined based on the Cauchy-Riemann equations ($\mathbb{R}$–derivative), whereas when applied to non-analytic functions such as real-valued cost functions, it is equal to the standard pseudo-gradient ($\mathbb{R}^*$–derivative). Also note that, while the emphasis here is on real-valued functions, the $\mathbb{CR}$ calculus framework is general to complex-valued functions. Referring to the examples given earlier, consider the non-analytic function $f_2(\mathbf{z}) = \|\mathbf{z}\|_2^2 = \mathbf{z}\mathbf{z}^*$. Then, $\frac{\partial f_2}{\partial \mathbf{z}} = \mathbf{z}^*$ and $\frac{\partial f_2}{\partial \mathbf{z}^*} = \mathbf{z}$. In contrast, for the analytic function $f_1(\mathbf{z}) = \mathbf{z}^2$, $\frac{\partial f_1}{\partial \mathbf{z}} = 2\mathbf{z}$ and $\frac{\partial f_1}{\partial \mathbf{z}^*} = \mathbf{0}$.

Another important result in $\mathbb{CR}$ calculus, referred to as *Brandwood's result* [53] states that the direction of steepest descent is given by the derivative with respect to $\mathbf{z}^*$, the $\mathbb{R}^*$-derivative. This can be shown by using the first order Taylor Series Expansion (TSE) of $f$ [50]; the magnitude of a small change in the function $f$ is given by

$$|\delta f| = 2\left|\Re\left\{\left(\frac{\partial f}{\partial \mathbf{z}^*}\right)^H \delta\mathbf{z}\right\}\right| \tag{B.10}$$

and the Cauchy-Schwarz Inequality shows that

$$|\delta f| \leq 2\left\|\frac{\partial f}{\partial \mathbf{z}^*}\right\| \cdot \|\delta\mathbf{z}\| \tag{B.11}$$

and so $|\delta f|$ is maximised when

$$\arccos \frac{\langle \frac{\partial F}{\partial \mathbf{z}^*}, \delta\mathbf{z}\rangle}{\|\frac{\partial F}{\partial \mathbf{z}^*}\| \, \|\delta\mathbf{z}\|} = 0, \tag{B.12}$$

where $\langle\cdot,\cdot\rangle$ is the inner product operator. In other words, the maximum change of the gradient is in the direction of the conjugate of the weight vector [53, 54].

### B.1.1 Properties of $\mathbb{R}$-derivatives

Several important properties of derivative obtained from $\mathbb{CR}$ calculus are stated here [54]. Consider the function $f(\mathbf{z}) \in \mathbb{R}$, where

$$\frac{\partial f^*}{\partial \mathbf{z}^*} = \left(\frac{\partial f}{\partial \mathbf{z}}\right)^* \tag{B.13a}$$

$$\frac{\partial f^*}{\partial \mathbf{z}} = \left(\frac{\partial f}{\partial \mathbf{z}^*}\right)^* \tag{B.13b}$$

$$df = \frac{\partial f}{\partial \mathbf{z}} d\mathbf{z} + \frac{\partial f}{\partial \mathbf{z}^*} d\mathbf{z}^* \qquad \text{Differential rule}^1 \tag{B.13c}$$

$$\frac{\partial (f \circ g)}{\partial \mathbf{z}} = \frac{\partial f(g)}{\partial \mathbf{z}} = \frac{\partial f}{\partial g}\frac{\partial g}{\partial \mathbf{z}} + \frac{\partial f}{\partial g^*}\frac{\partial g^*}{\partial \mathbf{z}} \qquad \text{Chain rule} \tag{B.13d}$$

$$\frac{\partial (f \circ g)}{\partial \mathbf{z}^*} = \frac{\partial f(g)}{\partial \mathbf{z}^*} = \frac{\partial f}{\partial g}\frac{\partial g}{\partial \mathbf{z}^*} + \frac{\partial f}{\partial g^*}\frac{\partial g^*}{\partial \mathbf{z}^*} \qquad \text{Chain rule} \tag{B.13e}$$

Note in particular that property (B.13b) only applies to real-valued functions as the conjugation operator has no effect on the real-valued function $f$, while the other properties can be generalised to any complex function.

## B.2 Taylor Series Expansion of Real-valued functions of Complex Variables

The TSE of $f(\mathbf{z}) : \mathbb{C}^N \mapsto \mathbb{R}$ up to a 2nd order approximation is considered. This is achieved by considering the function $f$ in three equivalent forms

$$f(\mathbf{z}) \longleftrightarrow f(\mathbf{z}, \mathbf{z}^*) \triangleq f(\mathbf{z}^a) \longleftrightarrow f(\mathbf{z}_r, \mathbf{z}_i) \triangleq f(\mathbf{z}^R) \tag{B.14}$$

and establishing the duality between the derivatives of functions in $\mathbb{C}^{2N}$ and $\mathbb{R}^{2N}$; this approach was utilised in [50] by van den Bos. In (B.14), the augmented vectors $\mathbf{z}^a = [\mathbf{z}^T \; \mathbf{z}^H]^T \in \mathbb{C}^{2N}$ and $\mathbf{z}^R = [\mathbf{z}_r^T \; \mathbf{z}_i^T]^T \in \mathbb{R}^{2N}$. In [54], Kreutz-Delgado provided an alternative and more rigorous approach by first establishing the isomorphism (duality) between vectors in $\mathbb{C}^{2N}$ and $\mathbb{R}^{2N}$, and identifying the *Jacobian of the transformation* for the calculation of derivatives. The first and second derivatives for the terms of the TSE are then readily calculated. The terms of the TSE in $\mathbb{C}^N$ are then easily found through expansion of the augmented complex TSE terms.

The transformation between the two augmented spaces is provided by $\mathbf{J}$, given in (2.6) and

$$\mathbf{z}^a = \mathbf{J}\mathbf{z}^R \tag{B.15}$$

$$\mathbf{z}^R = \mathbf{J}^{-1}\mathbf{z}^a = \frac{1}{2}\mathbf{J}^H\mathbf{z}^a, \tag{B.16}$$

[1] See the derivation in (B.4)– (B.6)

where the inverse mapping $\mathbf{J}^{-1} = \frac{1}{2}\mathbf{J}^H$. Then, as this coordinate transformation is linear and one to one, the $\mathbb{C}^{2N}$ and $\mathbb{R}^{2N}$ spaces may be considered isomorphic [54]. The Jacobian of the transformation from $\mathbb{R}^{2N}$ to $\mathbb{C}^{2N}$ is given by[2]

$$J_{\mathbb{C}} = \frac{\partial}{\partial \mathbf{z}^R}\mathbf{z}^a = \frac{\partial}{\partial \mathbf{z}^R}\mathbf{J}\mathbf{z}^R = \mathbf{J} \tag{B.17}$$

and the Jacobian of the transformation from $\mathbb{C}^{2N}$ to $\mathbb{R}^{2N}$ by

$$J_{\mathbb{R}} = \frac{\partial}{\partial \mathbf{z}^a}\mathbf{z}^R = \frac{\partial}{\partial \mathbf{z}^a}\mathbf{J}^{-1}\mathbf{z}^R = \mathbf{J}^{-1} = \frac{1}{2}\mathbf{J}^H. \tag{B.18}$$

Therefore, the Jacobian of the transformation $\mathbf{J}_{\mathbb{C}}$ (or inverse of the Jacobian of the transformation $\mathbf{J}_{\mathbb{R}}$) equals the coordinate transformation $\mathbf{J}$ (or inverse coordinate transformation $\mathbf{J}^{-1}$), and thus transformations between partial derivatives in the two spaces can be established as

$$\frac{\partial}{\partial \mathbf{z}^a} = \frac{1}{2}\frac{\partial}{\partial \mathbf{z}^R}\mathbf{J}^H \tag{B.19}$$

$$\frac{\partial}{\partial \mathbf{z}^R} = \frac{\partial}{\partial \mathbf{z}^a}\mathbf{J}. \tag{B.20}$$

The TSE expansion in $\mathbb{R}^{2N}$ up to the second term is known to be given by

$$f(\mathbf{z}^R + \Delta\mathbf{z}^R) = f(\mathbf{z}^R) + \frac{\partial f}{\partial \mathbf{z}^R}\Delta\mathbf{z}^R + \frac{1}{2}\Delta\mathbf{z}^{RT}\mathbf{H}^R_{\mathbf{zz}}\Delta\mathbf{z}^R, \tag{B.21}$$

where $\mathbf{H}^R_{\mathbf{zz}} = \frac{\partial}{\partial \mathbf{z}^R}\left(\frac{\partial f}{\partial \mathbf{z}^R}\right)^T$ is the real-valued augmented Hessian matrix.

The first order term in the augmented complex space is calculated as

$$\begin{aligned}\frac{\partial f}{\partial \mathbf{z}^R}\Delta\mathbf{z}^R &= \frac{\partial f}{\partial \mathbf{z}^a}\mathbf{J}\cdot\mathbf{J}^{-1}\mathbf{z}^a \\ &= \frac{\partial f}{\partial \mathbf{z}^a}\Delta\mathbf{z}^a,\end{aligned} \tag{B.22}$$

where the relations (B.16) and (B.20) are used. Now consider the augmented complex Hessian matrix

$$\mathbf{H}^a_{\mathbf{zz}} = \frac{\partial}{\partial \mathbf{z}^a}\left(\frac{\partial f}{\partial \mathbf{z}^a}\right)^H = \begin{bmatrix}\mathbf{H}_{\mathbf{zz}} & \mathbf{H}_{\mathbf{z}^*\mathbf{z}} \\ \mathbf{H}_{\mathbf{zz}^*} & \mathbf{H}_{\mathbf{z}^*\mathbf{z}^*}\end{bmatrix}, \tag{B.23}$$

where its equivalence with $\mathbf{H}^R_{\mathbf{zz}}$ is established as

$$\mathbf{H}^R_{\mathbf{zz}} = \mathbf{J}^H\mathbf{H}^a_{\mathbf{zz}}\mathbf{J}. \tag{B.24}$$

Thus, the second-order term of the augmented complex TSE is calculated as

$$\frac{1}{2}\Delta\mathbf{z}^{RT}\mathbf{H}^R_{\mathbf{zz}}\Delta\mathbf{z}^R = \frac{1}{2}\Delta\mathbf{z}^{aH}\mathbf{H}^a_{\mathbf{zz}}\Delta\mathbf{z}^a. \tag{B.25}$$

[2]Following the convention in [54], derivatives are defined as row vectors in this appendix.

Thus, using relations (B.14), (B.22) and (B.25) the TSE expansion in $\mathbb{C}^{2N}$ (augmented TSE) up to the second term can be expressed as

$$f(\mathbf{z}^a + \Delta\mathbf{z}^a) = f(\mathbf{z}^a) + \frac{\partial f}{\partial \mathbf{z}^a}\Delta\mathbf{z}^a + \frac{1}{2}\Delta\mathbf{z}^{aH}\mathbf{H}^a_{\mathbf{zz}}\Delta\mathbf{z}^a. \tag{B.26}$$

Expansion of the terms in (B.26) results in the TSE expressed directly in $\mathbb{C}^N$, which is given by

$$f(\mathbf{z} + \Delta\mathbf{z}) = f(\mathbf{z}) + 2\Re\left\{\frac{\partial f}{\partial \mathbf{z}}\Delta\mathbf{z}\right\} + \Re\left\{\Delta\mathbf{z}^H\mathbf{H}_{\mathbf{zz}}\Delta\mathbf{z} + \Delta\mathbf{z}^H\mathbf{H}_{\mathbf{z}^*\mathbf{z}}\Delta\mathbf{z}^*\right\}. \tag{B.27}$$

It is seen that the complex TSE is not a trivial extension from the TSE in $\mathbb{R}$, and while its direct derivation from the multivariate form (B.21) is not trivial and requires cumbersome algebraic manipulation, the augmented TSE provides a straightforward means for its calculation. Note that the augmented TSE (B.26) also serves as a compact representation of the TSE in the complex domain.

Appendix C generalises the discussion of this section and addresses the TSE of real-valued functions of complex matrix variable.

### B.2.1 Eigenvalues of the Augmented Real and Complex Hessian Matrices

Further insight into the structure of the augmented Hessian matrices $\mathbf{H}^R_{\mathbf{zz}}$ and $\mathbf{H}^a_{\mathbf{zz}}$ may be obtained through analysis of Equation (B.24) [50, 54]. Consider the linear system

$$(\mathbf{H}^a_{\mathbf{zz}} - \lambda^a\mathbf{I})\mathbf{u} = \mathbf{0} \tag{B.28}$$

with the set of solutions spanning the eigenspace. Using relation (B.24) and noting that $\frac{1}{2}\mathbf{J}\mathbf{J}^H = \mathbf{I}$, the left-hand side of (B.28) can be rewritten so that

$$\begin{aligned}\mathbf{H}^a_{\mathbf{zz}} - \lambda^a\mathbf{I} &= \frac{1}{4}\mathbf{J}\mathbf{H}^R_{\mathbf{zz}}\mathbf{J}^H - \frac{1}{2}\lambda^a\mathbf{J}\mathbf{J}^H \\ &= \frac{1}{4}\mathbf{J}(\mathbf{H}^R_{zz} - \underbrace{2\lambda^a}_{\triangleq\lambda^R}\mathbf{I})\mathbf{J}^H.\end{aligned} \tag{B.29}$$

This illustrates that the eigenvalues $\{\lambda^R\}$ of the real-valued Hessian matrix $\mathbf{H}^R_{\mathbf{zz}}$ are twice the eigenvalues $\{\lambda^a\}$ of the complex-valued Hessian matrix $\mathbf{H}^a_{\mathbf{zz}}$ [50].

### B.2.2 The Augmented Newton Method

Utilising the Taylor Series Expansions (B.21) and (B.26), the formulation for the Newton method in the augmented real and complex domains are expressed as

$$\Delta \mathbf{z}^R = -\mathbf{H}_{\mathbf{zz}}^{R^{-1}} \left( \frac{\partial f}{\partial \mathbf{z}^R} \right)^T \tag{B.30}$$

$$\Delta \mathbf{z}^a = -\mathbf{H}_{\mathbf{zz}}^{a^{-1}} \left( \frac{\partial f}{\partial \mathbf{z}^a} \right)^H, \tag{B.31}$$

and the formulation in $\mathbb{C}^N$ is obtained through expansion of (B.31), detailed below.

The Equation (B.31) expressed in its expanded form, is given by

$$\begin{bmatrix} \mathbf{H}_{\mathbf{zz}} & \mathbf{H}_{\mathbf{z}^*\mathbf{z}} \\ \mathbf{H}_{\mathbf{z}\mathbf{z}^*} & \mathbf{H}_{\mathbf{z}^*\mathbf{z}^*} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{z}^* \end{bmatrix} = - \begin{bmatrix} \left( \frac{\partial f}{\partial \mathbf{z}} \right)^H \\ \left( \frac{\partial f}{\partial \mathbf{z}^*} \right)^H \end{bmatrix}. \tag{B.32}$$

Solving for $\Delta \mathbf{z}^*$ and $\Delta \mathbf{z}$, and after substitution, we obtain the Newton method in $\mathbb{C}$, given by

$$\Delta \mathbf{z} = \left( \mathbf{H}_{\mathbf{zz}} - \mathbf{H}_{\mathbf{z}^*\mathbf{z}} \mathbf{H}_{\mathbf{z}^*\mathbf{z}^*}^{-1} \mathbf{H}_{\mathbf{z}\mathbf{z}^*} \right)^{-1} \left( \mathbf{H}_{\mathbf{z}^*\mathbf{z}} \mathbf{H}_{\mathbf{z}^*\mathbf{z}^*}^{-1} \left( \frac{\partial f}{\partial \mathbf{z}^*} \right)^H - \left( \frac{\partial f}{\partial \mathbf{z}} \right)^H \right). \tag{B.33}$$

It is seen that the derivation of the complex Newton method is not trivial if calculated directly from (B.27), while the augmented Newton methods provides a simple methods for its calculation. Also note that the expression for the complex Newton method is more involved in comparison to its real-valued counterpart. By simplifying the second order terms of the TSE and assuming a quasi-Newton method whereby the block off-diagonals of $\mathbf{H}_{\mathbf{zz}}^a$ are zero, the complex Newton method in (B.33) is simplified as

$$\Delta \mathbf{z} = -\mathbf{H}_{\mathbf{zz}}^{-1} \left( \frac{\partial f}{\partial \mathbf{z}} \right)^H. \tag{B.34}$$

This, however, results in a sub-optimal optimisation methodology for the generality of signal processing problems in the complex domain, and its use is limited to the case of analytic functions where the condition (B.9) is satisfied.

# Appendix C

# Real-valued Functions of Complex Matrices

As algorithms based on so called augmented complex statistics are emerging, leading to more accurate but mathematically involved solutions, revisiting some aspects of complex calculus is a prerequisite to providing a set of analytic tools to support these developments. In this direction, for real-valued functions of complex vector variables, the work by van den Bos [50] has provided a platform for modelling and optimisation via so called augmented vector spaces, with a thorough overview given in [54], where the duality between these spaces is explored (also see Appendix B). The application of these results have been recently utilised in various statistical signal processing fields, such as adaptive filtering [63].

Complex optimisation problems often involve real-valued functions[1] of complex matrices; these are a standard in communications and signal processing problems, such as in optimisation problems in Multiple-input and Multiple-output (MIMO) systems and in blind source separation. In this appendix, by complementing the work in [153], [154], we extend the concept of duality between vectors $\mathbb{R}^N$ and $\mathbb{C}^N$ in [54] to the case of complex matrix spaces, and formalise the equivalence of real-valued functions of complex matrix variables in the standard and augmented spaces up to their second-order Taylor Series Expansion.

It is shown that this is sufficient for the derivation and analysis of standard gradient-based learning algorithms. This also helps with the analysis of general signal processing algorithms in augmented matrix spaces and allows for simpler closed form solutions. Applications in Newton optimisation and blind source separation demonstrate the potential of the introduced complex matrix calculus results. This is followed

[1]For instance the cost function in complex adaptive filtering is $\mathcal{J} = e(k)e^*(k)$ and is a real function of complex error $e(k)$.

by a comparison of adaptive algorithms in the real and complex matrix spaces and demonstrate the trade-offs associated with the algorithms.

## C.1 Representations of complex matrices

The complex matrix $\mathbf{Z} = \mathbf{Z}_r + \jmath\mathbf{Z}_i \in \mathbb{C}^{M\times N}$, with $\mathbf{Z}_r$ and $\mathbf{Z}_i$ denoting respectively the real and imaginary components, can be equivalently described as a matrix $\mathbf{Z}^R$ in the real-valued space $\mathbb{R}^{2M\times 2N}$, given by

$$\mathbf{Z}^R = \begin{bmatrix} \mathbf{Z}_r & -\mathbf{Z}_i \\ \mathbf{Z}_i & \mathbf{Z}_r \end{bmatrix} \in \mathbb{R}^{2M\times 2N} \triangleq \mathcal{R} \tag{C.1}$$

or as a matrix $\mathbf{Z}^a$ in the complex conjugate-coordinate space[2] $\mathbb{C}^{2M\times 2N}$, given by

$$\mathbf{Z}^a = \begin{bmatrix} \mathbf{Z} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}^* \end{bmatrix} \in \mathbb{C}^{2M\times 2N} \triangleq \mathcal{C} \tag{C.2}$$

where $\mathbf{Z}^a$ is referred to as the augmented form of the complex matrix $\mathbf{Z}$ and $\mathbf{0}$ is a zero-valued matrix of size $M \times N$ [54]. This equivalent notation is possible due to the duality (isomorphism) between the spaces $\mathcal{R}$ and $\mathcal{C}$ and is formalised by the transformation between $\mathbf{Z}^R$ and $\mathbf{Z}^a$, described by the matrix[3]

$$\mathbf{J}_K = \begin{bmatrix} \mathbf{I} & \jmath\mathbf{I} \\ \mathbf{I} & -\jmath\mathbf{I} \end{bmatrix}. \tag{C.3}$$

Matrix $\mathbf{J}_K$, introduced in [50] and [54], is a square block matrix of size $2K \times 2K$ and $\mathbf{I}$ is the identity matrix of size $K \times K$. The inverse of this mapping is given by

$$\mathbf{J}_K^{-1} = \frac{1}{2}\mathbf{J}_K^H \tag{C.4}$$

and thus matrices $\mathbf{Z}^R$ and $\mathbf{Z}^a$ are related by

$$\mathbf{Z}^a = \frac{1}{2}\mathbf{J}_M\mathbf{Z}^R\mathbf{J}_N^H, \qquad \mathbf{Z}^R = \frac{1}{2}\mathbf{J}_M^H\mathbf{Z}^a\mathbf{J}_N. \tag{C.5}$$

Alternatively, the mapping in (C.5) can be written using the $\mathrm{vec}(\cdot)$ operator[4]. In this manner[5],

$$\mathrm{vec}(\mathbf{Z}^a) = \frac{1}{2}(\mathbf{J}_N^* \otimes \mathbf{J}_M)\,\mathrm{vec}(\mathbf{Z}^R) = \breve{\mathbf{J}}\,\mathrm{vec}(\mathbf{Z}^R) \tag{C.6}$$

$$\mathrm{vec}(\mathbf{Z}^R) = \frac{1}{2}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H)\,\mathrm{vec}(\mathbf{Z}^a) = \breve{\mathbf{J}}^{-1}\,\mathrm{vec}(\mathbf{Z}^a) \tag{C.7}$$

[2] For simplicity, we use the notations $\mathcal{R} \triangleq \mathbb{R}^{2M\times 2N}$ and $\mathcal{C} \triangleq \mathbb{C}^{2M\times 2N}$ in the following sections.

[3] Alternatively, by using the scaling factor $1/\sqrt{2}$ in the definition in (C.3), the matrix $\mathbf{J}$ becomes a unitary matrix [48].

[4] The vec operator stacks the columns of a matrix into a single column in a chronological order [153].

[5] The vec operator and Kronecker product $\otimes$ are related by $\mathrm{vec}(\mathbf{RQS}) = (\mathbf{S}^T \otimes \mathbf{R})\,\mathrm{vec}(\mathbf{Q})$.

and allows for a simplified and convenient method of describing the coordinate transformation, denoted by $\breve{\mathbf{J}} \in \mathbb{R}^{4MN \times 4MN}$. Note that by using the vectorised variant using the vec operator, we can treat the matrices as a single column vector, however, the transformation between the augmented spaces is then dictated by the new transformation matrix $\breve{\mathbf{J}}$, and not the vector coordinate transformation $\mathbf{J}_K$ given in Equation (C.3).

Therefore, the Jacobian of the transformation [54] from $\mathcal{R}$ to $\mathcal{C}$ is given by

$$J_{\mathcal{C}} = \frac{\partial}{\partial \mathbf{Z}^R}\mathbf{Z}^a = \frac{\partial \operatorname{vec}(\mathbf{Z}^a)}{\partial \operatorname{vec}^T(\mathbf{Z}^R)} = \frac{1}{2}(\mathbf{J}_N^* \otimes \mathbf{J}_M) = \breve{\mathbf{J}} \tag{C.8}$$

and the Jacobian of the transformation from $\mathcal{C}$ to $\mathcal{R}$ by

$$J_{\mathcal{R}} = \frac{\partial}{\partial \mathbf{Z}^a}\mathbf{Z}^R = \frac{\partial \operatorname{vec}(\mathbf{Z}^R)}{\partial \operatorname{vec}^T(\mathbf{Z}^a)} = \frac{1}{2}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H) = \breve{\mathbf{J}}^{-1}. \tag{C.9}$$

This illustrates that the Jacobian of the transformation $J_{\mathcal{C}}$ in (C.8) is equal to the coordinate transformation $\breve{\mathbf{J}}$, and the Jacobian of the transformation $J_{\mathcal{R}}$ in (C.9) is equal to the inverse transformation $\breve{\mathbf{J}}^{-1}$ [54]. As a result, the partial derivative transformations[6] between the two spaces in the vectorised format are given by

$$\frac{\partial \operatorname{vec}(\cdot)}{\partial \operatorname{vec}^T(\mathbf{Z}^a)} = \frac{1}{2}\frac{\partial \operatorname{vec}(\cdot)}{\partial \operatorname{vec}^T(\mathbf{Z}^R)}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H) \tag{C.10}$$

$$\frac{\partial \operatorname{vec}(\cdot)}{\partial \operatorname{vec}^T(\mathbf{Z}^R)} = \frac{1}{2}\frac{\partial \operatorname{vec}(\cdot)}{\partial \operatorname{vec}^T(\mathbf{Z}^a)}(\mathbf{J}_N^* \otimes \mathbf{J}_M) \tag{C.11}$$

and are row vectors of size $1 \times 4MN$. Note that the partial derivative is defined as a row operator [54] with the transpose notation $\frac{\partial \operatorname{vec}(\cdot)}{\partial \operatorname{vec}^T(\cdot)}$ used to emphasise this fact.
For a real-valued scalar function of vector complex variables $f(\mathbf{Z}, \mathbf{Z}^*) : \mathbb{C}^{M\times N} \times \mathbb{C}^{M\times N} \mapsto \mathbb{R}$, the partial derivative transforms can be simplified to an equivalent form [154]

$$\frac{\partial f}{\partial \mathbf{Z}^a} = \frac{1}{2}\mathbf{J}_N \frac{\partial f}{\partial \mathbf{Z}^R}\mathbf{J}_M^H \tag{C.12}$$

$$\frac{\partial f}{\partial \mathbf{Z}^R} = \frac{1}{2}\mathbf{J}_N^H \frac{\partial f}{\partial \mathbf{Z}^a}\mathbf{J}_M \tag{C.13}$$

where $\frac{\partial f}{\partial \mathbf{Z}^a}$ and $\frac{\partial f}{\partial \mathbf{Z}^R}$ are matrices of size $2N \times 2M$. The proof for this alternative form is given in the next section, and follows directly from the first order expansion of $f(\mathbf{Z}, \mathbf{Z}^*)$. Also, note that $\frac{\partial(\cdot)}{\partial \mathbf{Z}^a}$ and $\frac{\partial(\cdot)}{\partial \mathbf{Z}^R}$ are shorthand notations and are calculated as

$$\begin{aligned}
\frac{\partial(\cdot)}{\partial \mathbf{Z}^a} &= \begin{bmatrix} \frac{\partial(\cdot)}{\partial \mathbf{Z}} & \mathbf{0} \\ \mathbf{0} & \frac{\partial(\cdot)}{\partial \mathbf{Z}^*} \end{bmatrix}^T, \\
\frac{\partial(\cdot)}{\partial \mathbf{Z}^R} &= \begin{bmatrix} \frac{\partial(\cdot)}{\partial \mathbf{Z}_r} & -\frac{\partial(\cdot)}{\partial \mathbf{Z}_i} \\ \frac{\partial(\cdot)}{\partial \mathbf{Z}_i} & \frac{\partial(\cdot)}{\partial \mathbf{Z}_r} \end{bmatrix}^T.
\end{aligned} \tag{C.14}$$

[6] Also termed the cogradient transformations in [54].

The real-valued scalar function $f$ can be equivalently described in terms of coordinates in either $\mathbb{C}^{M\times N}, \mathcal{R}$ and $\mathcal{C}$. Following on [54], the TSE of the function $f(\mathbf{Z}^R)$ up to the second term is

$$f(\mathbf{Z}^R + \Delta\mathbf{Z}^R) = f(\mathbf{Z}^R) + \mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^R}\Delta\mathbf{Z}^R\Big) + \frac{1}{2}\,\mathrm{vec}^T(\Delta\mathbf{Z}^R)\mathbf{H}^R_{\mathbf{ZZ}}\,\mathrm{vec}(\Delta\mathbf{Z}^R) \tag{C.15}$$

where symbol $\mathrm{Tr}(\cdot)$ denotes the matrix trace operator, $\Delta\mathbf{Z}^R$ and $\Delta\mathbf{Z}^a$ are of the form given in (C.1) and (C.2), and $\mathbf{H}^R_{\mathbf{ZZ}}$ is a real valued Hessian matrix given by

$$\mathbf{H}^R_{\mathbf{ZZ}} = \frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^R)}\,\mathrm{vec}\left(\left[\frac{\partial f}{\partial \mathbf{Z}^R}\right]^T\right) \in \mathbb{R}^{4MN\times 4MN}. \tag{C.16}$$

### C.1.1 Duality of First-Order Taylor Series Expansions

Upon rewriting the first-order expansion term in (C.15) in the vectorised format, and using (C.7) and (C.10), gives

$$\begin{aligned}
\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^R}\Delta\mathbf{Z}^R\Big) &= \frac{\partial f}{\partial\,\mathrm{vec}^T(\mathbf{Z}^R)}\,\mathrm{vec}(\Delta\mathbf{Z}^R)\\
&= \frac{1}{2}\frac{\partial f}{\partial\,\mathrm{vec}^T(\mathbf{Z}^R)}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H)\,\mathrm{vec}(\Delta\mathbf{Z}^a)\\
&= \frac{\partial f}{\partial\,\mathrm{vec}^T(\mathbf{Z}^a)}\,\mathrm{vec}(\Delta\mathbf{Z}^a)\\
&= \mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Delta\mathbf{Z}^a\Big)
\end{aligned} \tag{C.17}$$

which is the first-order TSE of $f(\mathbf{Z}^a)$ in $\mathcal{C}$. Furthermore, using the relations (C.5), we have

$$\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^R}\Delta\mathbf{Z}^R\Big) = \mathrm{Tr}\Big(\frac{1}{2}\frac{\partial f}{\partial \mathbf{Z}^R}\mathbf{J}_M^H(\Delta\mathbf{Z}^a)\mathbf{J}_N\Big) \tag{C.18}$$

$$\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Delta\mathbf{Z}^a\Big) = \mathrm{Tr}\Big(\frac{1}{2}\frac{\partial f}{\partial \mathbf{Z}^a}\mathbf{J}_M(\Delta\mathbf{Z}^R)\mathbf{J}_N^H\Big) \tag{C.19}$$

and due to the duality between $\mathcal{R}$ and $\mathcal{C}$, and the equivalence in the first-order terms in the corresponding TSEs we have[7]

$$\begin{aligned}
\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^R}\Delta\mathbf{Z}^R\Big) &= \mathrm{Tr}\Big(\frac{1}{2}\frac{\partial f}{\partial \mathbf{Z}^a}\mathbf{J}_M(\Delta\mathbf{Z}^R)\mathbf{J}_N^H\Big)\\
&= \mathrm{Tr}\Big(\frac{1}{2}\mathbf{J}_N^H\frac{\partial f}{\partial \mathbf{Z}^a}\mathbf{J}_M(\Delta\mathbf{Z}^R)\Big)
\end{aligned} \tag{C.20}$$

and

$$\begin{aligned}
\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Delta\mathbf{Z}^a\Big) &= \mathrm{Tr}\Big(\frac{1}{2}\frac{\partial f}{\partial \mathbf{Z}^R}\mathbf{J}_M^H(\Delta\mathbf{Z}^a)\mathbf{J}_N\Big)\\
&= \mathrm{Tr}\Big(\frac{1}{2}\mathbf{J}_N\frac{\partial f}{\partial \mathbf{Z}^R}\mathbf{J}_M^H(\Delta\mathbf{Z}^a)\Big).
\end{aligned} \tag{C.21}$$

[7] We also make use of the identity $\mathrm{Tr}(\mathbf{RQ}) = \mathrm{Tr}(\mathbf{QR})$.

The equivalence of the terms on both sides of relations (C.20) and (C.21) results in the simplified partial derivative transforms given in (C.12) and (C.13).
Now, to produce the first-order expansion of $f(\mathbf{Z})$ in $\mathbb{C}^{M\times N}$, the first-order terms of $f(\mathbf{Z}^a)$ can be expanded to yield

$$\begin{aligned}\mathrm{Tr}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Delta\mathbf{Z}^a\Big) &= \mathrm{Tr}\bigg(\Big(\frac{\partial f}{\partial \mathbf{Z}}\Big)^T\Delta\mathbf{Z} + \Big(\frac{\partial f}{\partial \mathbf{Z}^*}\Big)^T\Delta\mathbf{Z}^*\bigg)\\ &= 2\Re\bigg\{\mathrm{Tr}\Big(\Big(\frac{\partial f}{\partial \mathbf{Z}}\Big)^T\Delta\mathbf{Z}\Big)\bigg\}\end{aligned} \tag{C.22}$$

where $\frac{\partial f}{\partial \mathbf{Z}^*} = (\frac{\partial f}{\partial \mathbf{Z}})^*$, as $f \in \mathbb{R}$. Also note that the gradient in the direction of steepest descent is given by $\frac{\partial f}{\partial \mathbf{Z}^*}$ [153, 154].

### C.1.2 Eigenvalue analysis of Hessian matrices

The relationships between second-order terms in the TSE of a scalar $f$ in the spaces $\mathbb{C}^{M\times N}, \mathcal{R}$ and $\mathcal{C}$ shall now be established. In addition, by analysing the relationship between the Hessian matrices in $\mathcal{R}$ and $\mathcal{C}$, a relation between the eigenvalues of the corresponding Hessian matrices is provided.

Observe the relationship between the real Hessian matrix $\mathbf{H}^R_{\mathbf{ZZ}}$ in (C.16) and the complex Hessian matrix $\mathbf{H}^a_{\mathbf{ZZ}}$, given by[8]

$$\mathbf{H}^a_{\mathbf{ZZ}} = \frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^a)}\,\mathrm{vec}\bigg(\Big[\frac{\partial f}{\partial \mathbf{Z}^a}\Big]^H\bigg) \in \mathbb{C}^{4MN\times 4MN}. \tag{C.23}$$

From (C.16), we have[9]

$$\begin{aligned}\mathbf{H}^R_{\mathbf{ZZ}} &= \frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^R)}\,\mathrm{vec}\bigg(\Big[\frac{\partial f}{\partial \mathbf{Z}^R}\Big]^H\bigg)\\ &= \frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^R)}\bigg\{\mathrm{vec}\bigg(\frac{1}{2}\Big(\mathbf{J}_N^H\frac{\partial f}{\partial \mathbf{Z}^a}\mathbf{J}_M\Big)^H\bigg)\bigg\}\\ &= \frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^a)}\bigg\{\frac{1}{2}(\mathbf{J}_N^T\otimes\mathbf{J}_M^H)\,\mathrm{vec}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Big)^H\bigg\}\frac{1}{2}(\mathbf{J}_N^*\otimes\mathbf{J}_M)\\ &= \frac{1}{4}(\mathbf{J}_N^T\otimes\mathbf{J}_M^H)\frac{\partial}{\partial\,\mathrm{vec}^T(\mathbf{Z}^a)}\,\mathrm{vec}\Big(\frac{\partial f}{\partial \mathbf{Z}^a}\Big)^H(\mathbf{J}_N^*\otimes\mathbf{J}_M)\\ &= \frac{1}{4}(\mathbf{J}_N^T\otimes\mathbf{J}_M^H)\mathbf{H}^a_{\mathbf{ZZ}}(\mathbf{J}_N^*\otimes\mathbf{J}_M)\end{aligned} \tag{C.24}$$

[8]The notation $\mathrm{vec}([\cdot]^T)$ is used interchangeably with $\mathrm{vec}(\cdot)^T$. Note the difference from $\mathrm{vec}^T(\cdot)$.

[9]Notice that since $\mathbf{H}^R_{\mathbf{ZZ}}$ in (C.16) is real-valued, for convenience the complex conjugate operator is applied to both sides of (C.16) and hence replace $(\cdot)^T$ by $(\cdot)^H$.

which is the relationship between real and complex Hessian matrices, written in terms of $\mathbf{H}^a_{\mathbf{ZZ}}$. This relationship can also be expressed in terms of the real Hessian matrix $\mathbf{H}^R_{\mathbf{ZZ}}$ by noticing that the two Kronecker product terms are the inverse of one another[10]. Thus

$$\mathbf{H}^a_{\mathbf{ZZ}} = \frac{1}{4}(\mathbf{J}_N^* \otimes \mathbf{J}_M)\mathbf{H}^R_{\mathbf{ZZ}}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H). \tag{C.25}$$

The analysis of the eigenvalues of the two Hessian matrices will assist in understanding their duality. Following the approach in [50] and [54], consider the linear system

$$(\mathbf{H}^a_{\mathbf{ZZ}} - \lambda^a\mathbf{I})\mathbf{u} = \mathbf{0} \quad \Rightarrow \quad (\mathbf{H}^a_{\mathbf{ZZ}} - \lambda^a\mathbf{I}) = \mathbf{0} \tag{C.26}$$

where the set of solutions spans the eigenspace. Using the relation (C.25) we have

$$\begin{aligned}\mathbf{H}^a_{\mathbf{ZZ}} - \lambda^a\mathbf{I} &= \frac{1}{4}(\mathbf{J}_N^* \otimes \mathbf{J}_M)\mathbf{H}^R_{\mathbf{ZZ}}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H) - \lambda^a\frac{1}{4}(\mathbf{J}_N^* \otimes \mathbf{J}_M)(\mathbf{J}_N^T \otimes \mathbf{J}_M^H) \\ &= \frac{1}{4}(\mathbf{J}_N^* \otimes \mathbf{J}_M)\big(\underbrace{\mathbf{H}^R_{\mathbf{ZZ}} - \lambda^a\mathbf{I}}_{\Rightarrow\lambda^a=\lambda^R}\big)(\mathbf{J}_N^T \otimes \mathbf{J}_M^H)\end{aligned} \tag{C.27}$$

where $\{\lambda^a\}$ are the eigenvalues of the complex Hessian matrix. This demonstrates that for every eigenvalue $\lambda^a$ of the complex-valued Hessian matrix $\mathbf{H}^a_{\mathbf{ZZ}}$, there is a corresponding eigenvalue $\lambda^R$ of the real-valued Hessian matrix $\mathbf{H}^R_{\mathbf{ZZ}}$, and that these eigenvalues are equal

$$\lambda^R = \lambda^a. \tag{C.28}$$

### C.1.3 Duality of Second-Order Taylor Series Expansions

This section effectively extends the analysis for the vector case presented in [54]. The second-order expansion term in $\mathcal{C}$ is obtained from (C.15) using the relationship (C.24) such that

$$\begin{aligned}\frac{1}{2}\operatorname{vec}^T(\Delta\mathbf{Z}^R)\mathbf{H}^R_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^R) &= \frac{1}{2}\operatorname{vec}^H(\Delta\mathbf{Z}^R)\mathbf{H}^R_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^R) \\ &= \frac{1}{2}\Big(\frac{1}{2}\operatorname{vec}^H(\Delta\mathbf{Z}^a)(\mathbf{J}_N^* \otimes \mathbf{J}_M)\Big)\mathbf{H}^R_{\mathbf{ZZ}}\Big(\frac{1}{2}(\mathbf{J}_N^T \otimes \mathbf{J}_M^H)\operatorname{vec}(\Delta\mathbf{Z}^a)\Big) \\ &= \frac{1}{2}\operatorname{vec}^H(\Delta\mathbf{Z}^a)\mathbf{H}^a_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^a).\end{aligned} \tag{C.29}$$

[10] This can be observed from (C.8) and (C.9). Alternatively, the identity $(\mathbf{R}\otimes\mathbf{Q})^{-1} = \mathbf{R}^{-1}\otimes\mathbf{Q}^{-1}$ and (C.4) can be used to obtain the same result, i.e. $\frac{1}{4}(\mathbf{J}_N^* \otimes \mathbf{J}_M)(\mathbf{J}_N^T \otimes \mathbf{J}_M^H) = \mathbf{I}$.

The components of the second-order expansions in $\mathcal{C}$ can now be written in terms of matrix $\mathbf{Z}$ to derive the second-order expansion in the standard $\mathbb{C}^{M\times N}$ space, that is

$$\begin{aligned}\frac{1}{2}\operatorname{vec}^H(\Delta\mathbf{Z}^a)\mathbf{H}^a_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^a) = \\ \frac{1}{2}\Big(&\operatorname{vec}^H(\Delta\mathbf{Z})\frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z})^*}{\partial\operatorname{vec}^T(\mathbf{Z})}\operatorname{vec}(\Delta\mathbf{Z}) + \operatorname{vec}^T(\Delta\mathbf{Z})\frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z}^*)^*}{\partial\operatorname{vec}^T(\mathbf{Z})}\operatorname{vec}(\Delta\mathbf{Z}) \\ &+ \operatorname{vec}^H(\Delta\mathbf{Z})\frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z})^*}{\partial\operatorname{vec}^T(\mathbf{Z}^*)}\operatorname{vec}^*(\Delta\mathbf{Z}) + \operatorname{vec}^T(\Delta\mathbf{Z})\frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z}^*)^*}{\partial\operatorname{vec}^T(\mathbf{Z}^*)}\operatorname{vec}^*(\Delta\mathbf{Z})\Big) \\ &= \Re\Big\{\operatorname{vec}^H(\Delta\mathbf{Z})\mathbf{H}_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}) + \operatorname{vec}^H(\Delta\mathbf{Z})\mathbf{H}_{\mathbf{Z}^*\mathbf{Z}}\operatorname{vec}^*(\Delta\mathbf{Z})\Big\},\end{aligned} \tag{C.30}$$

where $\mathbf{H}_{\mathbf{ZZ}} \triangleq \frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z})^*}{\partial\operatorname{vec}^T(\mathbf{Z})}$ and $\mathbf{H}_{\mathbf{Z}^*\mathbf{Z}} \triangleq \frac{\partial\operatorname{vec}(\partial f/\partial\mathbf{Z})^*}{\partial\operatorname{vec}^T(\mathbf{Z}^*)}$.
To summarise, the expansion of $f$ in $\mathcal{R}$ is illustrated in (C.15), whereas the expansion in $\mathcal{C}$ is shown through the isomorphism between the two spaces given in (C.17) and (C.29), to yield

$$f(\mathbf{Z}^a + \Delta\mathbf{Z}^a) = f(\mathbf{Z}^a) + \operatorname{Tr}\Big(\frac{\partial f}{\partial\mathbf{Z}^a}\Delta\mathbf{Z}^a\Big) + \frac{1}{2}\operatorname{vec}^H(\Delta\mathbf{Z}^a)\mathbf{H}^a_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^a) \tag{C.31}$$

Similarly, the TSE of a scalar function of complex matrix variables $f$ in $\mathbb{C}^{M\times N}$ is given by (C.22) for the first term, and in (C.30) for the second term, that is

$$\begin{aligned}f(\mathbf{Z}+\Delta\mathbf{Z}) = f(\mathbf{Z}) &+ 2\Re\left\{\operatorname{Tr}\left(\left(\frac{\partial f}{\partial\mathbf{Z}}\right)^T\Delta\mathbf{Z}\right)\right\} \\ &+ \Re\Big\{\operatorname{vec}^H(\Delta\mathbf{Z})\mathbf{H}_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}) + \operatorname{vec}^H(\Delta\mathbf{Z})\mathbf{H}_{\mathbf{Z}^*\mathbf{Z}}\operatorname{vec}^*(\Delta\mathbf{Z})\Big\}\end{aligned} \tag{C.32}$$

## C.2 Application examples

To illustrate the potential of the derived results, two case studies are considered: Newton optimisation and Blind Source Separation.

### C.2.1 Optimisation in the Augmented Matrix Spaces

A classic optimisation application, illustrated in [50], is the minimisation of the real-valued function $f : \mathbb{C}^N \times \mathbb{C}^N \mapsto \mathbb{R}$ using the Newton method. The extension of this approach to functions of complex matrices $f : \mathbb{C}^{M\times N} \times \mathbb{C}^{M\times N} \mapsto \mathbb{R}$ is considered, to calculate the minima $\partial f/\partial\mathbf{Z}^R = \mathbf{0}$ and $\partial f/\partial\mathbf{Z}^a = \mathbf{0}$. By taking the derivative of the second order expansion term of $f(\mathbf{Z}^R)$ in (C.15), and $f(\mathbf{Z}^a)$, in (C.31), and equating to zero, we have

$$\mathbf{H}^R_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^R) = -\Big(\frac{\partial f}{\partial\operatorname{vec}^T(\mathbf{Z}^R)}\Big)^T \tag{C.33}$$

$$\mathbf{H}^a_{\mathbf{ZZ}}\operatorname{vec}(\Delta\mathbf{Z}^a) = -\Big(\frac{\partial f}{\partial\operatorname{vec}^T(\mathbf{Z}^a)}\Big)^H. \tag{C.34}$$

The benefit of this formulation is that it allows complex optimisation problems to be cast in augmented matrix spaces, which when combined with $\mathbb{CR}$ calculus, provide a simpler and easier to understand way of calculating the optimal solution.

### C.2.2 Derivative calculation in blind source separation

In the derivation of the complex blind source separation algorithm based on maximum likelihood, it is necessary to calculate the derivative $\frac{\partial \log|\det(\mathbf{Z}^R)|}{\partial \mathbf{Z}^*}$. The method provided in [155] requires the introduction of a new symmetric matrix and further algebraic manipulation. A more straightforward calculation, based on the introduced framework, gives

$$\begin{aligned}\log|\det(\mathbf{Z}^R)| &= \log|\det(\frac{1}{2}\mathbf{J}^H\mathbf{Z}^a\mathbf{J})| \\ &= \log|\det(\frac{1}{2}\mathbf{J}^H)\det(\mathbf{Z}^a)\det(\mathbf{J})| \\ &= \log|\det(\mathbf{Z}^a)| \\ &= \log|\det(\mathbf{Z})\cdot\det(\mathbf{Z}^*)| \\ &= \log|\det(\mathbf{Z})| + \log|\det(\mathbf{Z}^*)| \end{aligned} \tag{C.35}$$

and therefore

$$\frac{\partial \log|\det(\mathbf{Z}^R)|}{\partial \mathbf{Z}^*} = \left[\frac{\partial \log|\det(\mathbf{Z})|}{\partial \mathbf{Z}} + \frac{\partial \log|\det(\mathbf{Z}^*)|}{\partial \mathbf{Z}}\right]^* = \mathbf{Z}^{-H} \tag{C.36}$$

where some fundamental results from linear algebra [70] and matrix derivatives [154] have been used.

## C.3 Adaptive estimation of complex matrix sources

Several cost functions encountered in signal processing research are defined based on matrix inputs [153]. Here norm-based cost functions $\mathcal{J}(\mathbf{Z},\mathbf{Z}^*) : \mathbb{C}^{N\times N}\times\mathbb{C}^{N\times N}\mapsto\mathbb{R}$ given by

$$\mathcal{J}(\mathbf{A},\mathbf{A}^*) = \|\mathbf{A}\|_F^2 = \mathrm{Tr}(\mathbf{A}^H\mathbf{A}) \tag{C.37}$$

are addressed, where $\|\cdot\|_F$ denotes the Frobenius norm. Consider the linear predictor of $\mathbf{U}$ given by

$$\hat{\mathbf{U}} = \mathbf{W}^T\mathbf{Z}, \tag{C.38}$$

with estimation error $\mathbf{E} = \mathbf{U} - \hat{\mathbf{U}}$, input matrix $\mathbf{Z}$ and weight matrix $\mathbf{W}\in\mathbb{C}^{N\times N}$, and the norm-based cost function $\mathcal{J}(\mathbf{W},\mathbf{W}^*) = \|\mathbf{E}\|_F^2 = \mathrm{Tr}(\mathbf{E}^H\mathbf{E})$. The optimal value

of $\mathbf{W}$ can be obtained adaptively using a gradient descent method that minimises the cost function. Thus using $\mathbb{CR}$ calculus[11].

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla_{\mathbf{W}_k} \mathcal{J} = \mathbf{W}_k + \mu \mathbf{E}_k \mathbf{Z}_k^* \tag{C.39}$$

which will be referred to as the block complex least mean square (b-CLMS) algorithm, where $\mu$ is the step-size. Alternatively, by assuming a widely linear model (see Equation (2.33)) of $\mathbf{U}$ based on the input $\mathbf{Z}$ and its conjugate $\mathbf{Z}^*$, the output of the widely linear predictor is

$$\hat{\mathbf{U}}_{WL} = \mathbf{W}^T \mathbf{Z} + \mathbf{V}^T \mathbf{Z}^* \tag{C.40}$$

and $\mathbf{W}$ and $\mathbf{V}$ are the complex $N \times N$ weight matrices. The cost function can be minimised for both matrices to achieve the gradient descent algorithms[12]

$$\begin{aligned} \mathbf{W}_{k+1} &= \mathbf{W}_k + \eta \mathbf{E}_k \mathbf{Z}_k^* \\ \mathbf{V}_{k+1} &= \mathbf{V}_k + \eta \mathbf{E}_k \mathbf{Z}_k \end{aligned} \tag{C.41}$$

and $\eta$ is the step-size. We will refer to (C.41) as the block augmented complex least mean square (b-ACLMS) algorithm.

Now consider the matrix analog of the dual channel real least mean square (DCRLMS) algorithm described in [86], with real-valued input/output relation

$$\begin{bmatrix} \hat{\mathbf{Y}}_1 \\ \hat{\mathbf{Y}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} \\ \mathbf{H}_{21} & \mathbf{H}_{22} \end{bmatrix}^T \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} \tag{C.42}$$

where $\mathbf{X}_i$ are the real-valued input matrices and $\hat{\mathbf{Y}}_i$ are the estimated output. The matrix of weight matrices $\mathbf{H}_{pq} \in \mathbb{R}^{N \times N}$ is updated adaptively as

$$\mathbf{H}_{pq,k+1} = \mathbf{H}_{pq,k} + \rho \mathbf{E}_q \mathbf{X}_{p,k}, \quad p, q = \{1, 2\} \tag{C.43}$$

and $\mathbf{E}_{q,k} = \mathbf{Y}_{q,k} - \hat{\mathbf{Y}}_{q,k}$ is the estimation error and $\rho$ is the step-size. We will refer to the update algorithms (C.43) as block DCRLMS (b-DCRLMS).

In order to perform analysis between the update algorithms in $\mathbb{C}^{N \times N}$ and $\mathbb{R}^{N \times N}$, we will write the linear input relation (C.38) in terms of its real and imaginary components $\mathbf{U}^r$ and $\mathbf{U}^i$, to obtain

$$\begin{aligned} \hat{\mathbf{U}}^r &= \mathbf{W}^{r^T} \mathbf{Z}^r - \mathbf{W}^{i^T} \mathbf{Z}^i \\ \hat{\mathbf{U}}^i &= \mathbf{W}^{i^T} \mathbf{Z}^r + \mathbf{W}^{r^T} \mathbf{Z}^i \end{aligned} \tag{C.44}$$

[11] For clarity and simplicity in the discussion of this section, we will use an alternative notation. Then, $\mathbf{Z}_k$ denotes the value of complex-valued variable $\mathbf{Z}$ at sample $k$, while $\mathbf{Z}_k^r$ and $\mathbf{Z}_k^i$ respectively refer to the real and imaginary component of the complex-valued variable $\mathbf{Z}$ at sample $k$.

[12] See Section 3.2.2 for the derivation of the vector ACLMS algorithm.

and for the widely linear relation (C.40), we have

$$\hat{\mathbf{U}}^r_{WL} = (\mathbf{W}^r + \mathbf{V}^r)^T\mathbf{Z}^r + (\mathbf{V}^i - \mathbf{W}^i)^T\mathbf{Z}^i \tag{C.45}$$

$$\hat{\mathbf{U}}^i_{WL} = (\mathbf{W}^i + \mathbf{V}^i)^T\mathbf{Z}^r + (\mathbf{W}^r - \mathbf{V}^i)^T\mathbf{Z}^i. \tag{C.46}$$

Similarly, the update algorithms can be written in terms of the updates for the real and imaginary components of the weight matrices. For the b-CLMS algorithm (C.39), we thus have

$$\mathbf{W}^r_{k+1} = \mathbf{W}^r_k + \mu(\mathbf{E}^r_k\mathbf{Z}^r_k + \mathbf{E}^i_k\mathbf{Z}^i_k) \tag{C.47}$$

$$\mathbf{W}^i_{k+1} = \mathbf{W}^i_k + \mu(\mathbf{E}^i_k\mathbf{Z}^r_k - \mathbf{E}^r_k\mathbf{Z}^i_k), \tag{C.48}$$

while for the b-ACLMS algorithm (C.41)

$$\mathbf{W}^r_{k+1} = \mathbf{W}^r_k + \eta(\mathbf{E}^r_k\mathbf{Z}^r_k + \mathbf{E}^i_k\mathbf{Z}^i_k) \tag{C.49}$$

$$\mathbf{W}^i_{k+1} = \mathbf{W}^i_k + \eta(\mathbf{E}^i_k\mathbf{Z}^r_k - \mathbf{E}^r_k\mathbf{Z}^i_k) \tag{C.50}$$

$$\mathbf{V}^r_{k+1} = \mathbf{V}^r_k + \eta(\mathbf{E}^r_k\mathbf{Z}^r_k - \mathbf{E}^i_k\mathbf{Z}^i_k) \tag{C.51}$$

$$\mathbf{V}^i_{k+1} = \mathbf{V}^i_k + \eta(\mathbf{E}^i_k\mathbf{Z}^r_k + \mathbf{E}^r_k\mathbf{Z}^i_k). \tag{C.52}$$

### C.3.1 Adaptive Strictly Linear Algorithms

To compare the input/output relation and the dynamics of the b-CLMS and b-DCRLMS algorithms, for the same inputs from (C.44) and (C.42) we have

$$\mathbf{X}_1 = \mathbf{Z}^r, \quad \mathbf{X}_2 = \mathbf{Z}^i \tag{C.53}$$

and the corresponding errors are defined so that

$$\mathbf{E}_1 = \mathbf{E}^r, \quad \mathbf{E}_2 = \mathbf{E}^i. \tag{C.54}$$

Thus, for the same outputs $\hat{\mathbf{Y}}_1 = \hat{\mathbf{U}}^r$ and $\hat{\mathbf{Y}}_2 = \hat{\mathbf{U}}^i$, we have

$$\begin{aligned} \mathbf{H}_{11} &= \mathbf{W}^r & \mathbf{H}_{12} &= \mathbf{W}^i \\ \mathbf{H}_{21} &= -\mathbf{W}^i & \mathbf{H}_{22} &= \mathbf{W}^r \end{aligned} \tag{C.55}$$

It is clear that the b-CLMS input/output relation is a constrained version of the b-DCRLMS, where fixed values are assigned to the $\mathbf{H}_{ij}$ matrices.

The dynamic behaviour of the two update algorithms can be readily compared from (C.43) and (C.47), illustrating that the two algorithms are not equivalent, due to the different dynamics of the updates in $\mathbb{C}^{N\times N}$ and $\mathbb{R}^{N\times N}$. Also notice that while the updates $\Delta\mathbf{W}^r_k$ and $\Delta\mathbf{W}^i_k$ of the b-CLMS algorithm depend on both the real and imaginary error components, the b-DCRLMS update $\Delta\mathbf{H}_{ij}$ is calculated based on only the error

from one channel. However, by assuming the constraints (C.55) on the weights $\mathbf{H}_{ij}$, we can deduce that

$$\begin{aligned}\Delta\mathbf{H}_{11,k} = \Delta\mathbf{H}_{22,k} &= \frac{1}{2}(\mathbf{E}_{1,k}\mathbf{X}_{1,k} + \mathbf{E}_{2,k}\mathbf{X}_{2,k}) = \frac{1}{2}\Delta\mathbf{W}_k^r \\ \Delta\mathbf{H}_{12,k} = -\Delta\mathbf{H}_{21,k} &= \frac{1}{2}(\mathbf{E}_{2,k}\mathbf{X}_{1,k} - \mathbf{E}_{1,k}\mathbf{X}_{2,k}) = \frac{1}{2}\Delta\mathbf{W}_k^i \end{aligned} \tag{C.56}$$

and so for as equal step-size $\rho = \mu$, the b-DCRLMS algorithm converges to the optimal solution two times slower as the b-CLMS algorithm.

### C.3.2 Adaptive Widely Linear Algorithms

The input/output relation of the widely linear model (C.40) to the dual channel real-valued model in (C.42) is now compared. Assuming the same input relations (C.53) and by matching the output errors (C.54), the component expansions in (C.45)–(C.46) provide the relation between the corresponding outputs, such that

$$\begin{aligned}\mathbf{H}_{11} &= (\mathbf{W}^r + \mathbf{V}^r) & \mathbf{H}_{12} &= (\mathbf{W}^i + \mathbf{V}^i) \\ \mathbf{H}_{21} &= (\mathbf{V}^i - \mathbf{W}^i) & \mathbf{H}_{22} &= (\mathbf{W}^r - \mathbf{V}^r) \end{aligned} \tag{C.57}$$

result in the equivalent outputs $\hat{\mathbf{Y}}_1 = \mathbf{U}_{WL}^r$ and $\hat{\mathbf{Y}}_2 = \mathbf{U}_{WL}^i$.
The relationship between the dynamics of the b-ACLMS and b-DCRLMS algorithms through simple algebraic manipulations of (C.49)–(C.52) is established, where for the same step-size $\rho = \eta$ the following equivalence is given

$$\begin{aligned}\Delta\mathbf{H}_{11,k} &= \mathbf{E}_{1,k}\mathbf{X}_{1,k} = \frac{1}{2}(\Delta\mathbf{W}_k^r + \Delta\mathbf{V}_k^r) \\ \Delta\mathbf{H}_{12,k} &= \mathbf{E}_{2,k}\mathbf{X}_{1,k} = \frac{1}{2}(\Delta\mathbf{W}_k^i + \Delta\mathbf{V}_k^i) \\ \Delta\mathbf{H}_{21,k} &= \mathbf{E}_{1,k}\mathbf{X}_{2,k} = \frac{1}{2}(\Delta\mathbf{V}_k^i - \Delta\mathbf{W}_k^i) \\ \Delta\mathbf{H}_{22,k} &= \mathbf{E}_{2,k}\mathbf{X}_{2,k} = \frac{1}{2}(\Delta\mathbf{W}_k^r - \Delta\mathbf{V}_k^r). \end{aligned} \tag{C.58}$$

Therefore, the b-DCRLMS is the real-valued equivalent of the b-ACLMS algorithm, while having a convergence rate twice as slow as that of its complex counterpart. However, due to its design based on the optimisation of a widely linear model, the b-ACLMS is better suited for modelling of complex data as it is optimal for both second order circular and noncircular signals. Finally, note that these results are in line with the existing results on adaptive algorithms in $\mathbb{R}^N$ and $\mathbb{C}$ [63].

### C.3.3 Computational Complexity of Adaptive Algorithms

To compare the computational complexity of the b-CLMS, b-ACLMS and b-DCRLMS algorithms, the measurement used was the 'flop', defined as the number of floating

**Table C.1** Computational complexity of the real- and complex-valued adaptive algorithms. The variable $N$ denotes the size of a square matrix.

| Algorithm | Flops |
|---|---|
| b-CLMS | $2(3N^2 + 4N^3)$ |
| b-ACLMS | $4(3N^2 + 4N^3)$ |
| b-DCRLMS | $4(2N^2 + 2N^3)$ |

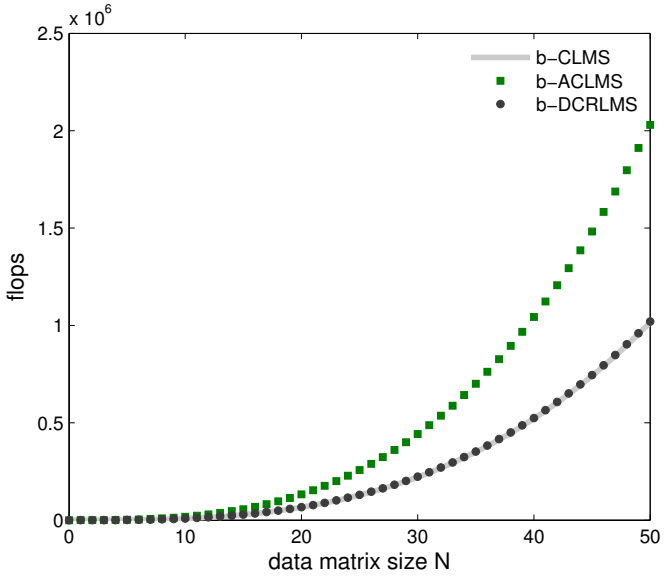



**Figure C.1** Computational complexity of the b-CLMS, b-ACLMS and b-DCRLMS algorithms

point operations [156]. Table C.3.3 states the number of flops for each adaptive algorithm, where $N$ is the length of a square matrix while Figure C.1 illustrates the increase in the computational complexity for an increase in the size of the data matrix for the respective algorithms.

It can be seen that while the computational complexity of the b-CLMS and b-DCRLMS algorithms are similar, the b-ACLMS algorithm has a higher computational cost for the same matrix size[13]. Likewise, for data matrices of size $N \geq 10$, the cost of computation becomes an important factor, while for $N < 10$, the number of flops are approximately the same across all algorithms and we focus on the performance of the algorithm. Given the equivalence of the b-ACLMS and b-DCRLMS algorithms, the implementation of the b-ACLMS is obviously less computationally effective than that of the b-DCRLMS, while providing a natural processing environment for complex data.

[13]The b-DCRLMS algorithm has an additional overhead of $\mathcal{O}(N^2)$ with $2N^2$ flops compared to the b-CLMS algorithm, while the extra computational complexity of the b-ACLMS compared to the b-DCRLMS is $\mathcal{O}(N^3)$, that is, $4N^2 + 8N^3$.

# Appendix D

# Convergence Analysis of the Generalised Complex FastICA Algorithm

## D.1 Introduction

The FastICA [21] algorithm is one of the most efficient methods for the blind separation of independent sources due to its use of fixed-point like updates which enable fast convergence [157]. The algorithm was subsequently extended to the complex domain by Bingham and Hyvärinen [41], termed the c-FastICA, with the explicit assumption of circularly symmetric distributions of the sources. Another fixed-point update for the complex ICA, proposed by Douglas [71], is the fixed-point FastICA algorithm based on the kurtosis cost function and utilising the strong uncorrelating transform (SUT) [69]; no circularity assumptions are needed as both covariance and pseudo-covariance matrices are diagonalised using the SUT instead of the conventional whitening of only the covariance matrix.

The more recent variant of the complex FastICA algorithm [78], the nc-FastICA algorithm, is a generalisation of the c-FastICA algorithm [41], which considers the possible noncircularity of complex sources and has been derived using the $\mathbb{CR}$ calculus. The nc-FastICA algorithm was shown to be stable for circular as well as for non-circular sources owing to an always positive-definite Hessian of the cost function. This is in contrast with the c-FastICA algorithm, whose fixed-point like updates are only stable for circular sources and are not stable for noncircular ones. The local stability analysis of the cost function in nc-FastICA indicates that for circular sources the solution is a stable point independent of whether maximising or minimising the cost function. For noncircular sources however, there is a region of instability whose size depends on

the deviation from Gaussianity and degree of the noncircularity of the signal, as well as the nonlinearity used in the cost function. For example, for a kurtosis based cost function, sub-Gaussian signals used in communications such as the circular QAM and noncircular BPSK lie close to this region of instability, with the stability compromised as the signals become more noncircular [158, 78].

The convergence of the real domain FastICA was investigated in [21] and [22] using a single unit case, where the orthogonalisation was not taken into account. In [159] Douglas also addresses the convergence of the real FastICA algorithm using one source update, and for a cubic cost function. Erdogan generalises the study of fixed-points in ICA algorithms in $\mathbb{R}$ and provides a proof for the monotonic convergence of fixed-point ICA algorithms with symmetrical orthogonalisation [160].

While the previous methods consider a single unit update, convergence analysis of FastICA algorithms can be performed by considering the orthogonalisation applied at each iteration of the update algorithm; two often used methods are the deflationary and simultaneous (parallel) orthogonalisation techniques. The deflationary orthogonalisation using the Gram-Schmidt method processes the signals sequentially, and so the convergence analysis becomes an extension of single unit convergence analyses. However, source estimation errors in an update stage accumulate and cause subsequent source estimates to be noisy [71]. The symmetric orthogonalisation allows for simultaneous estimation of all the sources and does not suffer from the estimation error propagation issue of the deflationary method. A complete analysis for the real FastICA based on the symmetrical orthogonalisation was performed recently by Oja and Yuan, whereby both single unit convergence and the orthogonalisation approach were considered [161].

It should be noted that each method has its merits; for example, while the parallel orthogonalisation method is unaffected by the accumulation of deflation errors, it is only suitable for the estimation of sources from small-scale mixtures, and will result in additional overhead for large-scale mixtures when only a subset of latent sources is of interest. For such applications, the deflationary orthogonalisation technique may be better suited; for example, in EEG conditioning, shown in Chapter 6, it is necessary to only estimate and extract one or two artifacts from a large-scale EEG dataset (as many as 64 channels).

For rigour, convergence of both the nc-FastICA and c-FastICA algorithms is considered under one umbrella, and will address the convergence utilising three different approaches. First, an overview of the generalised complex FastICA algorithm and its special case, the c-FastICA algorithm, is given.

- Then, in the first approach, analysis is performed by following the methodology of [161], where the convergence of the nc-FastICA algorithm with symmetric

orthogonalisation is considered. The convergence is analysed using a linear algebraic method. While this results in a simple analysis framework, it assumes initial local convergence.

- In the second approach, a second-order approximation using the complex domain Taylor Series Expansion, discussed in Appendix B, is used for the convergence analysis. Similar to the previous method, local convergence is assumed.
- Finally, an interpretation of the update algorithm as a fixed-point iteration is given, where its convergence behaviour in the phase-space is also observed. Here, the convergence is based on the assumptions of fixed-point theory, and as such, provides for a generalised analysis framework.

## D.2 An Overview of ICA in the Complex Domain

The ICA problem in the complex domain assumes latent sources $\mathbf{s} \in \mathbb{C}^{N_s}$, which are linearly combined through a complex mixing matrix $\mathbf{A}$ and are available through the observed vector $\check{\mathbf{x}}$, that is

$$\check{\mathbf{x}} = \mathbf{A}\mathbf{s} \tag{D.1}$$

The mixing matrix $\mathbf{A} \in \mathbb{C}^{N \times N_s}$ is assumed invertible and the aim is to find a demixing matrix $\mathbf{W}$ such that the sources can be estimated from the observed data. For convenience, a square mixing matrix is assumed, such that $N_s = N$. The sources $\mathbf{s} = [s_1, \ldots, s_{N_s}]^T$ are assumed to be non-Gaussian and mutually independent, with unit variances and zero means. In other words, the covariance matrix $E\{\mathbf{s}\mathbf{s}^H\} = \mathbf{I}$, however, no assumptions are made about the circularity of the sources. In the standard c-FastICA [41], however, the sources were explicitly taken as circular, with a vanishing pseudo-covariance, that is, $E\{\mathbf{s}\mathbf{s}^T\} = \mathbf{0}$.

It is common to initially orthogonalise the data through a whitening transform $\mathbf{V}$, such that

$$\mathbf{x} = \mathbf{V}\check{\mathbf{x}} = \mathbf{V}\mathbf{A}\mathbf{s} = \mathbf{M}\mathbf{s} \tag{D.2}$$

The vector of estimated sources $\mathbf{y} = \mathbf{W}^H\mathbf{x}$, and a single source estimate $y_i$ is given by

$$y_i = \mathbf{w}_i^H\mathbf{x} = \mathbf{w}_i^H\mathbf{M}\mathbf{s}, \quad i = 1, \ldots, N_s \tag{D.3}$$

where $\mathbf{w}_i$ is the $i$th column of $\mathbf{W}$. At the optimal solution, $\mathbf{u}_i^H = \mathbf{w}_i^H\mathbf{M}$ has a single non-zero complex component with unit magnitude and an unknown phase. That is

$$\mathbf{u}_i = [0, \ldots, e^{\jmath\varphi}, 0, \ldots, 0]^T \tag{D.4}$$

and $u_{ij}, j \in [1, N]$ is the $j$th element of column vector $\mathbf{u}_i$. This is due to the limitation of ICA, where a source is estimated up to a scaling factor and random order (permutation).

### D.2.1 The nc-FastICA and c-FastICA Algorithms

To find the optimal values for the demixing vector, a cost function

$$\mathcal{J}(\mathbf{w}, \mathbf{w}^*) = E\{G(|\mathbf{w}^H\mathbf{x}|^2)\} \tag{D.5}$$

is represented by its conjugate (augmented) coordinates $\mathbf{w}$ and $\mathbf{w}^*$ and is minimised under the constraint $\|\mathbf{w}\|_2^2 = 1$, where $G : \mathbb{R} \mapsto \mathbb{R}$ is an even nonlinear function. The cost function $\mathcal{J} : \mathbb{C}^N \mapsto \mathbb{R}$ is optimised for both $\mathbf{w}$ and its complex conjugate $\mathbf{w}^*$, that is, based on the $\mathbb{CR}$ calculus, where the real valued cost function is regarded as $\mathbb{R}$-analytic. This approach, which allows for the consideration of noncircular signals, was used in [78] to derive the weight update of the nc-FastICA algorithm, given by

$$\begin{aligned}\breve{\mathbf{w}}_i = &-E\{g(|y_i|^2)y_i^*\mathbf{x}\} + E\{g'(|y_i|^2)|y_i|^2 + g(|y_i|^2)\}\mathbf{w}_i \\ &+ E\{\mathbf{x}\mathbf{x}^T\}E\{g'(|y_i|^2)y_i^{*^2}\}\mathbf{w}_i\end{aligned} \tag{D.6}$$

for a single unit $\mathbf{w}_i$, and $y_i = \mathbf{w}_i^H\mathbf{x}$. The symbol $\breve{\mathbf{w}}_i$ denotes the $i$th single unit update before being normalised to unit norm. The function $g$ is the derivative of $G$ and $g'$ is the derivative of $g$. Notice that the last term in (D.6) contains the pseudo-covariance matrix, $E\{\mathbf{x}\mathbf{x}^T\}$, which caters for the noncircularity of complex signals. In the case of circular signals, this term becomes zero, giving the original c-FastICA update:

$$\breve{\mathbf{w}}_i = -E\{g(|y_i|^2)y_i^*\mathbf{x}\} + E\{g'(|y_i|^2)|y_i|^2 + g(|y_i|^2)\}\mathbf{w}_i. \tag{D.7}$$

Orthonormalisation of the updates can be performed by a deflationary or symmetrical orthogonalisation. Using the deflationary method, the independent components are estimated sequentially, whereas the symmetrical orthogonalisation allows for a parallel estimation of the independent components, that is

$$\mathbf{W} = (\breve{\mathbf{W}}\breve{\mathbf{W}}^H)^{-\frac{1}{2}}\breve{\mathbf{W}} = \breve{\mathbf{W}}(\breve{\mathbf{W}}^H\breve{\mathbf{W}})^{-\frac{1}{2}}. \tag{D.8}$$

Stability analyses of these algorithms showed that the fixed-point updates are always stable for circular sources, whereas for noncircular sources regions of instability [78] need to be identified.

### D.2.2 The Analysis Framework

Extending the approach from [161] to the complex domain, the convergence analysis framework shall now be introduced.

From (D.2), notice that $\mathbf{M} = \mathbf{V}\mathbf{A}$ is a unitary matrix. As $\mathbf{x}$ is whitened, gives

$$E\{\mathbf{x}\mathbf{x}^H\} = \mathbf{M}E\{\mathbf{s}\mathbf{s}^H\}\mathbf{M}^H = \mathbf{I} \quad \Rightarrow \mathbf{M}\mathbf{M}^H = \mathbf{I} \tag{D.9}$$

The source vector $\mathbf{s}$ can then be rewritten as

$$\mathbf{s} = \mathbf{M}^{-1}\mathbf{x} = \mathbf{M}^H\mathbf{x} \tag{D.10}$$

Define a linear transform

$$\mathbf{U}^H = \mathbf{W}^H\mathbf{M} \tag{D.11}$$

which for a single $i$th row of $\mathbf{U}^H$, denoted as $\mathbf{u}_i^H$, is given as[1]

$$\mathbf{u}_i^H = \mathbf{w}_i^H\mathbf{M} \tag{D.12}$$

Using the above transform, the symmetric orthogonalisation can be redefined by multiplying both sides of (D.8) by $\mathbf{M}^H$ from the left, that is

$$\mathbf{M}^H\mathbf{W} = \mathbf{M}^H\breve{\mathbf{W}}(\breve{\mathbf{W}}^H\mathbf{M}\mathbf{M}^H\breve{\mathbf{W}})^{-\frac{1}{2}} \tag{D.13}$$

$$\mathbf{U} = \breve{\mathbf{U}}(\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}}. \tag{D.14}$$

The single unit update for the nc-fastICA algorithm (D.6) can also be written in terms of the transformed vectors $\mathbf{u_i}$ and $\mathbf{s}$ by multiplying both sides by $\mathbf{M}^H$ from the left to yield

$$\begin{aligned}\breve{\mathbf{u}}_i = &-E\{g(|\mathbf{u}_i^H\mathbf{s}|^2)(\mathbf{u}_i^H\mathbf{s})^*\mathbf{s}\} \\ &+ E\{g'(|\mathbf{u}_i^H\mathbf{s}|^2)|\mathbf{u}_i^H\mathbf{s}|^2 + g(|\mathbf{u}_i^H\mathbf{s}|^2)\}\mathbf{u}_i \\ &+ E\{\mathbf{s}\mathbf{s}^T g'(|\mathbf{u}_i^H\mathbf{s}|^2)(\mathbf{u}_i^H\mathbf{s})^{*^2}\}\mathbf{u}_i^*\end{aligned} \tag{D.15}$$

where the independence assumption [41]

$$E\{xxf(x)\} \approx E\{xx\}E\{f(x)\} \tag{D.16}$$

was used in the third term of (D.15).

## D.3 Convergence analysis of the Parallel nc-FastICA based on an extension of the real domain approach in [161]

This analysis closely follows the convergence analysis in [161], and takes into account specific properties of the complex domain.

**Lemma 1.** *At convergence, the matrix* $\mathbf{U}$*, a diagonal matrix with components* $e^{\jmath\varphi}$*, with* $\varphi$ *an unknown phase, is the fixed point of* (D.14).

[1] Vector $\mathbf{u}_i$ is the $i$th column of $\mathbf{U}$.

*Proof.* As only the $i$th component of $\mathbf{u}_i^H$ is non-zero, gives

$$\mathbf{u}_i^H\mathbf{s} = e^{-\jmath\varphi}s_i \;\rightarrow\; |\mathbf{u}_i^H\mathbf{s}| = |e^{-\jmath\varphi}s_i| = |s_i|,\; g(|\mathbf{u}_i^H\mathbf{s}|^2) = g(|s_i|^2) \tag{D.17}$$

This way (D.15) is simplified into

$$\begin{aligned}\breve{\mathbf{u}}_i = &-E\{g(|s_i|^2)e^{\jmath\varphi}s_i^*\mathbf{s}\}\\ &+ E\{g'(|s_i|^2)|s_i|^2 + g(|s_i|^2)\}\mathbf{u}_i\\ &+ E\{\mathbf{s}\mathbf{s}^T g'(|s_i|^2)e^{\jmath 2\varphi}s_i^{*^2}\}\mathbf{u}_i^*\end{aligned} \tag{D.18}$$

Observe that:

i) Following on (D.7), for the c-FastICA algorithm readily yields

$$\breve{\mathbf{u}}_i = -E\{g(|s_i|^2)e^{\jmath\varphi}s_i^*\mathbf{s}\} + E\{g'(|s_i|^2)|s_i|^2 + g(|s_i|^2)\}\mathbf{u}_i \tag{D.19}$$

The $i$th component in the first term of (D.19) (resp. (D.18)) is $-E\{g(|s_i|^2)|s_i|^2e^{\jmath\varphi}\}$ and all other components are zero because the function $g$ depends on $s_i$ and so $E\{s_is_jg(|s_i|^2)\} = 0, j \neq i$.

By simplifying (D.19) further, gives

$$\breve{\mathbf{u}}_i = q_i\mathbf{u}_i, q_i \neq 0, q_i \in \mathbb{R} \tag{D.20}$$

where

$$q_i = -E\{g(|s_i|^2)|s_i|^2\} + E\{g'(|s_i|^2)|s_i|^2 + g(|s_i|^2)\} \tag{D.21}$$

To comprise updates for all sources, equation (D.20) can be expanded as

$$\breve{\mathbf{U}} = \mathbf{D}\mathbf{U} \tag{D.22}$$

where $\mathbf{D} = \mathrm{diag}(q_1, \ldots, q_N)$ is a diagonal matrix.

ii) For the nc-FastICA algorithm, the last term in (D.18) can be simplified into

$$\underbrace{E\{\mathbf{s}\mathbf{s}^T g'(|s_i|^2)s_i^{*^2}\}}_{\tilde{\mathbf{C}}}\mathbf{u}_i \tag{D.23}$$

A further insight shows that the $\tilde{c}_{jk} = (\tilde{\mathbf{C}})_{jk}$, that is the component of row $j$ and column $k$ (or $jk$th component) of $\tilde{\mathbf{C}}$ can be written as

$$\tilde{c}_{jk} = E\{s_js_kg'(|s_i|^2)s_i^{*^2}\} \tag{D.24}$$

For $k = i$, we have

$$\begin{aligned}\tilde{c}_{ji} &= E\{s_js_ig'(|s_i|^2)s_i^{*^2}\}\\ &= E\{s_js_i^*\underbrace{g'(|s_i|^2)|s_i|^2}_{r_i}\}\\ &= \begin{cases} 0 & ,j \neq i\\ r_i & ,j = i\end{cases}\\ &= \tilde{c}_{ij}\end{aligned} \tag{D.25}$$

As the sources are assumed independent, the approximation (D.16) is used, and since the pseudo-covariance matrix is complex symmetric, the elements $\tilde{c}_{jk} = \tilde{c}_{kj}$ [45]. The matrix $\tilde{\mathbf{C}}$ can then be written as

$$\tilde{\mathbf{C}} = \begin{bmatrix} \tilde{c}_{11} & \cdots & \cdots & \cdots & 0 & \cdots & \cdots & \tilde{c}_{1N} \\ \vdots & \ddots & & & \vdots & \ddots & & \vdots \\ \vdots & & & & 0 & & & \vdots \\ 0 & \cdots & \cdots & 0 & r_i & 0 & \cdots & 0 \\ \vdots & \ddots & & & 0 & \ddots & & \vdots \\ \vdots & & & & \vdots & & & \vdots \\ \tilde{c}_{N1} & \cdots & \cdots & \cdots & 0 & \cdots & \cdots & \tilde{c}_{NN} \end{bmatrix} \tag{D.26}$$

and the expression (D.23) becomes

$$\tilde{\mathbf{C}}\mathbf{u}_i = r_i\mathbf{u}_i \tag{D.27}$$

Substitute (D.27) in (D.18) to obtain

$$\begin{aligned} \breve{\mathbf{u}}_i &= q_i\mathbf{u}_i + r_i\mathbf{u_i} \\ &= d_i\mathbf{u}_i \quad , d_i = q_i + r_i \neq 0 \quad , d_i \in \mathbb{R} \end{aligned} \tag{D.28}$$

where $q_i$ is defined as in (D.21). By considering all $\breve{\mathbf{u}}_i$ in (D.28), this yields

$$\breve{\mathbf{U}} = \mathbf{D}\mathbf{U} \tag{D.29}$$

and $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_N)$.

The matrix $\breve{\mathbf{U}}$ in (D.29) has an identical structure to that obtained in the c-FastICA update, given in (D.22).

For convenience, examine

$$\begin{aligned} \breve{\mathbf{U}}^H\breve{\mathbf{U}} &= (\mathbf{D}\mathbf{U})^H(\mathbf{D}\mathbf{U}) \\ &= \mathbf{U}^H\mathbf{D}^H\mathbf{L}\mathbf{U} \\ &= |\mathbf{D}|^2 = \mathbf{D}^2 \\ \Rightarrow (\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}} &= \mathbf{D}^{-1} \end{aligned} \tag{D.30}$$

and so

$$\breve{\mathbf{U}}(\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}} = \mathbf{D}\mathbf{U}\mathbf{D}^{-1} = \mathbf{U} \tag{D.31}$$

that is, the mapping has reached its fixed point. However, note that this demonstrates an asymptotic convergence due to oscillations in each single unit update $\mathbf{u}_i$ once the

fixed point has been reached. This issue was addressed in the real domain in [162] and is attributed to sign flipping, whereas in $\mathbb{C}$ these oscillations are due to the phase uncertainty, as illustrated in Section D.6.

In the analysis here, the relation $\mathbf{DU} = \mathbf{UD}$ is used, as they are both diagonal matrices. The diagonal elements of $\mathbf{D}$ are assumed to be non-zero, making the matrix invertible. This, in return, proves that the diagonal matrix $\mathbf{U}$ contains the fixed points of (D.14), that is, both FastICA and nc-FastICA converge to a unique solution. □

This proof can now be extended to take into account the permutation ambiguity in the order of the fixed points in $\mathbf{U}$.

**Remark 1.** *Permutations of* $\mathbf{U}$ *are also fixed points of* (D.14).

*Proof.* Extending the result in [161] for real–valued FastICA, the permutation matrix $\mathbf{P}$ is a real valued orthogonal matrix, that is, $\mathbf{PP}^T = \mathbf{I}$. Thus, we need to show that $\mathbf{PU}$ and $\mathbf{UP}$ are also fixed points.

Adapting the proof given in Lemma 2 in [161] to the complex domain, it is straightforward to illustrate that $\mathbf{P}\breve{\mathbf{U}}$ and $\breve{\mathbf{U}}\mathbf{P}$ converge respectively to $\mathbf{PU}$ and $\mathbf{UP}$ using the symmetrical orthogonalisation given in (D.14). More specifically,

$$\begin{aligned}(\mathbf{P}\breve{\mathbf{U}})\left((\mathbf{P}\breve{\mathbf{U}})^H(\mathbf{P}\breve{\mathbf{U}})\right)^{-\frac{1}{2}} &= \mathbf{P}\breve{\mathbf{U}}(\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}} \\ &= \mathbf{PU} \end{aligned} \tag{D.32}$$

and

$$\begin{aligned}(\breve{\mathbf{U}}\mathbf{P})\left((\breve{\mathbf{U}}\mathbf{P})^H(\breve{\mathbf{U}}\mathbf{P})\right)^{-\frac{1}{2}} &= \breve{\mathbf{U}}(\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}}\mathbf{P} \\ &= \mathbf{UP} \end{aligned} \tag{D.33}$$

By using the expression

$$\left((\breve{\mathbf{U}}\mathbf{P})^H(\breve{\mathbf{U}}\mathbf{P})\right)^{-\frac{1}{2}} = \mathbf{P}^T(\breve{\mathbf{U}}^H\breve{\mathbf{U}})^{-\frac{1}{2}}\mathbf{P} \tag{D.34}$$

which is adapted from the real domain for the proof of (D.33).

Therefore, permutations $\breve{\mathbf{U}}\mathbf{P}$ and $\mathbf{P}\breve{\mathbf{U}}$ both converge to permutations of the fixed points $\mathbf{U}$, that is, $\mathbf{UP}$ and $\mathbf{PU}$. □

## D.4 Convergence of the nc-FastICA algorithm using a Taylor Series Expansion approach

The convergence of the nc-FastICA is now investigated using the Taylor Series Expansion (TSE) approximation of the update algorithm (D.6) in a manner similar to

that in [22]. The TSE of real-valued functions of complex variables was addressed in Section B.2 of Appendix B.

For simplicity, the algorithm is rearranged into the form given in (D.15), where the vector $\mathbf{u}_i$ is assumed to be close to the solution with $|u_{i1}| \approx 1$ and $|u_{ij}| \approx 0$, $\forall j \neq 1$. The TSE of a real-valued function of complex variables $f(\mathbf{z}) : \mathbb{C}^N \mapsto \mathbb{R}$ up to a second order around a value $\mathbf{z}_0$ is given by [54] (see Appendix B)

$$f(\mathbf{z}_0 + \Delta\mathbf{z}) \approx f(\mathbf{z}_0) + 2\Re\Big\{\frac{\partial f}{\partial \mathbf{z}}\Delta\mathbf{z}\Big\} + \Re\Big\{\Delta\mathbf{z}^H\mathbf{H}_{\mathbf{zz}}\Delta\mathbf{z} + \Delta\mathbf{z}^H\mathbf{H}_{\mathbf{z}^*\mathbf{z}}\Delta\mathbf{z}^*\Big\} \tag{D.35}$$

where $\Delta\mathbf{z} = \mathbf{z} - \mathbf{z}_0$ and $\mathbf{H}_{\mathbf{zz}} = \frac{\partial}{\partial \mathbf{z}}(\frac{\partial f}{\partial \mathbf{z}})^H$ and $\mathbf{H}_{\mathbf{z}^*\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}^*}(\frac{\partial f}{\partial \mathbf{z}})^H$ are the Hermitian matrices. While it is equally valid to define the TSE of $f$ in terms of the augmented coordinates $\mathbf{z}$ and $\mathbf{z}^*$, due to the equivalence of notations, the definition simplifies to that in (D.35) (see [54, p.39]).

The TSE of the nonlinearities $\{g, g'\} \in \mathbb{R}$ in the neighbourhood of $\mathbf{u}_i$ is then written as

$$\begin{aligned} g(|\mathbf{u}^H\mathbf{s}|^2) \approx\ & g(|s_i|^2) + 2g'(|s_i|^2)\Re\{\Delta\xi\} + g''_{\mathbf{uu}}(|s_i|^2)|\Delta\xi|^2 \\ & + g''_{\mathbf{u}^*\mathbf{u}}(|s_i|^2)\Re\{\Delta\xi^*\} \end{aligned} \tag{D.36}$$

and

$$\begin{aligned} g'(|\mathbf{u}^H\mathbf{s}|^2) \approx\ & g'(|s_i|^2) + 2g''_{\mathbf{u}^*\mathbf{u}}(|s_i|^2)\Re\{\Delta\xi\} + g'''_{\mathbf{uu}}(|s_i|^2)|\Delta\xi|^2 \\ & + g'''_{\mathbf{u}^*\mathbf{u}}(|s_i|^2)\Re\{\Delta\xi^*\} \end{aligned} \tag{D.37}$$

where $\Delta\xi \triangleq \Delta(\mathbf{u}^H\mathbf{s}) = \mathbf{u}^H\mathbf{s} - \mathbf{u}_i^H\mathbf{s} = (\Delta u_1)^* s_1 + \cdots + (\Delta u_j)^* s_j + \cdots + (\Delta u_N)^* s_N$, $(\Delta u_1)^* s_1 \approx 0$ and $g'''$ is the derivative of $g''$.

After the substitution of (D.36)–(D.37) in (D.15) and simplification, the elements of the vector $\breve{\mathbf{u}}_i$ can be expressed as

$$\breve{u}_{i1} = -E\{g(|s_i|^2)|s_i|^2\}u_{i1} + E\{2g'(|s_i|^2)|s_i|^2 + g(|s_i|^2)\}u_{i1} \tag{D.38}$$

and

$$\begin{aligned} \breve{u}_{ij} \\ {\scriptstyle j\neq 1} \end{aligned} = & -E\{g''_{\mathbf{uu}}(|s_j|^2)|s_j|^4 + g''_{\mathbf{uu}}(|s_j|^2)|s_j|^2\}|u_{ij}|^2 u_{ij} \\ & + E\{g'''_{\mathbf{uu}}(|s_j|^2)|s_j|^4\}|u_{ij}|^4 u_{ij} + E\{g'''_{\mathbf{uu}}(|s_j|^2)|s_j|^6\}|u_{ij}|^2 u_{ij}^{*3}. \end{aligned} \tag{D.39}$$

As $\breve{\mathbf{u}}_i$ is normalised after each update, observe from (D.38) and (D.39) that $|\breve{u}_{i1}| = 1$ and $|\breve{u}_{ij}| = 0$, with the algorithm exhibiting local convergence for the $i$th single unit update.

## D.5 Fixed Point Interpretation of Convergence

In Section D.3, the convergence of the generalised complex FastICA algorithm with symmetric orthogonalisation was presented, where at convergence the matrix $\mathbf{U}$ was shown to be the fixed point of (D.14). Deeper insight into the mechanism of the algorithm is provided by considering a fixed point interpretation of the convergence. This will be achieved by focusing on the cost function $\mathcal{J}$ in (D.5) and by analysing the convergence behaviour of the algorithm following the methodology in [163, 164].

Regalia and Kofidis [163] provided analysis for the convergence of the real domain FastICA algorithm using a gradient update method where the conditions for monotonic convergence of the algorithm using convex and non-convex cost functions were given (upper and lower bounds of the gradient update step-size). A general framework for the convergence of complex FastICA algorithms with symmetric orthogonalisation was recently proposed by Erdogan in [164], where it was shown that the algorithm is monotonically convergent for convex cost functions, and conditions for the convergence of non-convex functions were provided. The convergence behaviour for a convex cost function of a single unit update shall be considered.

**Theorem 1.** *For a non-decreasing nonlinearity $G(\mathbf{z})$ in Equation (D.5), the nc-FastICA algorithm converges monotonically to a maximum of the cost function $\mathcal{J}(\mathbf{u}, \mathbf{u}^*)$.*

*Proof.* First it is illustrated that the cost function $\mathcal{J}(\mathbf{u}, \mathbf{u}^*)$ is a convex function on $\mathbb{C}^{N\times N}$. Recall that a function $f : \mathbb{C}^N \mapsto \mathbb{C}^N$ is defined as a convex function if for two vectors $\mathbf{z}_1$ and $\mathbf{z}_2$

$$\big|f(\alpha\mathbf{z}_1 + (1-\alpha)\mathbf{z}_2)\big| \leq \alpha\big|f(\mathbf{z}_1)\big| + (1-\alpha)\big|f(\mathbf{z}_2)\big| \tag{D.40}$$

where $\alpha \in [0, 1]$.

The cost function (D.5) is given in terms of the modified demixing vector $\mathbf{u}$ as

$$\mathcal{J}(\mathbf{u}, \mathbf{u}^*) = E\{G(|\mathbf{u}^H\mathbf{s}|^2)\} \tag{D.41}$$

Notice that $\mathcal{J}(\mathbf{u}, \mathbf{u}^*)$ can be expressed as $G\big(H(\mathbf{u}^H\mathbf{s})\big)$, where $H(\cdot) = |\cdot|^2$, $H : \mathbb{C}^N \mapsto \mathbb{R}$; the Cauchy-Schwarz inequality (triangle inequality), then shows that $H$ is convex[2]. Then, the composite function $G \circ H$ is convex function if $G$ is non-decreasing [165], that is

$$\big|G\big(H(\alpha\mathbf{u}_1 + (1-\alpha)\mathbf{u}_2)\big)\big| \leq \alpha\big|G\big(H(\mathbf{u}_1)\big)\big| + (1-\alpha)\big|G\big(H(\mathbf{u}_2)\big)\big|. \tag{D.42}$$

Recall that the probability density function (pdf) $p_{\mathbf{Z}}(\mathbf{z})$ of a complex random variable $\mathbf{z} = \mathbf{z}_r + \jmath\mathbf{z}_i$ is defined in terms of the joint pdf of the real and imaginary components

[2] In the complex domain, the triangle inequality can be stated as $\|\mathbf{a} + \mathbf{b}\| \leq \|\mathbf{a}\| + \|\mathbf{b}\|, \quad \forall \mathbf{a}, \mathbf{b} \in \mathbb{C}$.

$p_{\mathbf{Z}}(\mathbf{z}) = p_{\mathbf{Z}_r,\mathbf{Z}_i}(\mathbf{z}_r, \mathbf{z}_i)$ and $0 \leq p_{\mathbf{Z}}(\mathbf{z}) \leq 1$. Following on from [163], the statistical mean for the function $G : \mathbb{C}^N \mapsto \mathbb{R}$ is then defined as

$$E\{G(\mathbf{z})\} = \iint\limits_{\mathbf{z}_r\mathbf{z}_i} G(\mathbf{z}) p_{\mathbf{Z}}(\mathbf{z}) d\mathbf{z}_r d\mathbf{z}_i. \tag{D.43}$$

Thus for two vectors $\mathbf{u}_1$ and $\mathbf{u}_2$, and using Equations (D.40),(D.41) and (D.43)

$$\begin{aligned}
\big|\mathcal{J}\left(\alpha\mathbf{u}_1 + (1-\alpha)\mathbf{u}_2\right)\big| &= \iint\limits_{\mathbf{s}_r\mathbf{s}_i} \big|G\big(H(\alpha\mathbf{u}_1 + (1-\alpha)\mathbf{u}_2)\big)\big| p_S(\mathbf{s}) d\mathbf{s}_r d\mathbf{s}_i \\
&\leq \iint\limits_{\mathbf{s}_r\mathbf{s}_i} \alpha\big|G(H(\mathbf{u}_1))\big| + (1-\alpha)\big|G(H(\mathbf{u}_2))\big| p_S(\mathbf{s}) d\mathbf{s}_r d\mathbf{s}_i \\
&= \alpha\big|\mathcal{J}(\mathbf{u}_1)\big| + (1-\alpha)\big|\mathcal{J}(\mathbf{u}_2)\big|
\end{aligned} \tag{D.44}$$

A comparison with (D.40), shows that $\mathcal{J}$ is convex.

For a convex $\mathcal{J}$, the gradient inequality up to a first order is expressed as [165]

$$\mathcal{J}(\mathbf{u}_{k+1}) \geq \mathcal{J}(\mathbf{u}_k) + 2\Re\Big\{\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}(\mathbf{u}_{k+1} - \mathbf{u}_k)\Big\} \tag{D.45}$$

where the first order term $2\Re\big\{\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}(\mathbf{u}_{k+1} - \mathbf{u}_k)\big\}$ can be readily obtained from the complex-valued Taylor Series Expansion given in (D.35), and the subscript $k$ denotes the iteration index.

The upper bound for the term $\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}\mathbf{u}_k = \big\langle \big(\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}\big)^H, \mathbf{u}_k \big\rangle = \big\langle \nabla_{\mathbf{u}_k^*}\mathcal{J}, \mathbf{u}_k \big\rangle$ is given by[3]

$$\Big\langle \Big(\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}\Big)^H, \mathbf{u}_k \Big\rangle \leq \Big\| \frac{\partial \mathcal{J}}{\partial \mathbf{u}_k^*} \Big\| \cdot \underbrace{\|\mathbf{u_k}\|}_{=1} \tag{D.46}$$

and as $\mathbf{u}_{k+1} = \nabla_{\mathbf{u}_k^*}\mathcal{J}/\|\nabla_{\mathbf{u}_k^*}\mathcal{J}\|$, the second term of the right hand side of inequality (D.45) can be expressed as

$$2\Re\Big\{ \underbrace{\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}\mathbf{u}_{k+1}}_{=\|\nabla_{\mathbf{u}_k^*}\mathcal{J}\|} - \underbrace{\frac{\partial \mathcal{J}}{\partial \mathbf{u}_k}\mathbf{u}_k}_{<\|\nabla_{\mathbf{u}_k^*}\mathcal{J}\|} \Big\} > 0, \quad \mathbf{u}_k \neq \mathbf{u}_{k+1} \tag{D.47}$$

and therefore $\mathcal{J}(\mathbf{u}_{k+1}) > \mathcal{J}(\mathbf{u}_k)$. Given that $\mathbf{u}$ is bounded to a unit norm, the cost function $\mathcal{J}$ is maximised as one of the fixed points is approached after each iteration and as $k \to \infty$. □

More generally, by considering a symmetric orthogonalisation, it can stated that $\mathcal{J}(\mathbf{U}_{k+1}) > \mathcal{J}(\mathbf{U}_k)$, which was presented in [164, Theorem 4].

[3]The inner product $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^H\mathbf{b}$.

### D.5.1 Contraction Mapping Theorem for Vector-valued Functions

The Contraction Mapping Theorem (CMT) was originally introduced for scalar functions $F : \mathbb{R}^N \mapsto \mathbb{R}$, and can help the convergence analysis by casting algorithms into a fixed point iteration (FPI) framework [157]. For example, it has been used to analyse the convergence and stability of nonlinear adaptive filters in both the real and complex domain, as well as to obtain the lower and upper error bounds of stability for contractive and expansive activation functions [166, 63]. The nc-FastICA (or c-FastICA) weight update algorithm is however a vector-valued function such that $\mathbf{F}(\mathbf{u}, \mathbf{u}^*) : \mathbb{C}^{2N\times 2N} \mapsto \mathbb{C}^{2N\times 2N}$, where $\mathbf{F}(\cdot)$ denotes the update algorithm (D.6) (or (D.7)) and is defined here in terms of the conjugate coordinates $\mathbf{u}$ and $\mathbf{u}^*$. By considering the duality between $\mathbb{C}^{2N}$ and $\mathbb{R}^{2N}$ [54](also see Appendix B), the CMT in this is stated below [167] (Theorem 5.3.2)

**Theorem 2** (CMT [167] (Theorem 5.3.2))**.** *For a closed subset $\mathcal{A} \in \mathbb{R}^{2N}$, the function $\mathbf{F}$ is considered a contraction iff*

1. $\mathbf{F} : \mathcal{A} \mapsto \mathcal{A}$*, i.e. the function* $\mathbf{F}$ *maps the set onto itself,*
2. $\exists\gamma$ *such that* $\|\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{y})\| \leq \gamma\|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{A},\ 0 \leq \gamma < 1.$

The parameter $\gamma$ is referred to as the Lipschitz constant where for values in $[0, 1)$, the function $\mathbf{F}$ is a contractive mapping on $\mathcal{A}$ and $\gamma$ defines the rate of convergence.

### D.5.2 Convergence Analysis of FPI based on the Jacobian Matrix

The eigenvalues of the Jacobian of the nonlinear function $\mathbf{F}$ in the neighbourhood of the fixed point $\mathbf{u}^\star$ are used to indicate the convergence behaviour. Eigenvalues situated within the unit circle result in convergence and show that $\mathbf{F}$ is a contraction, and eigenvalues outside the unit circle show that $\mathbf{F}$ is an expansion [63]. Using the complex Taylor expansion [50] it can be stated that

**Lemma 2.** *For a convergent twice differentiable function $\mathbf{F} : \mathbb{C}^N \mapsto \mathbb{C}^N$, the eigenvalues of the Jacobian and conjugate Jacobian matrix [54] evaluated at the fixed point $\mathbf{u}^\star$ must lie within the unit circle $\mathcal{U} = \{z \,|\, z \in \mathbb{C}, |z| < 1\}$. (See also [168])*

*Proof.* This condition was described in the paper by Ferrante et al. [168] for real functions; they are extended by considering the first order complex Taylor series expansion of $\mathbf{F}$ around the fixed point $\mathbf{u}^\star$ [50]. For the augmented vector $\mathbf{u}^a = [\mathbf{u},\ \mathbf{u}^*]^T$, the TSE of $\mathbf{F}$ is given by

$$\mathbf{F}(\mathbf{u}^a + \Delta\mathbf{u}^a) = \mathbf{F}(\mathbf{u}^a) + \frac{\partial \mathbf{F}(\mathbf{u}^a)}{\partial \mathbf{u}^a}\Delta\mathbf{u}^a + \dots. \tag{D.48}$$

Noting that $\mathbf{u}_k^a = \mathbf{u}^{\star a} - \mathbf{e}_k^a$ and $\mathbf{e}_k^a$ as the convergence error, the $(k+1)$th iteration can be expanded around the the fixed point as

$$\begin{aligned}\mathbf{u}_{k+1}^a &= \mathbf{F}(\mathbf{u}_k^a) = \mathbf{F}(\mathbf{u}^{\star a} - \mathbf{e}_k^a) \\ &= \mathbf{F}(\mathbf{u}^{\star a}) + \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^a}(\mathbf{u}_k^a - \mathbf{u}^{\star a}) + \dots \\ &= \mathbf{F}(\mathbf{u}^{\star a}) + \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^a}(-\mathbf{e}_k^a) + \dots, \quad \|\mathbf{e}_k\| \ll 1 \\ &\approx \mathbf{u}^{\star a} - \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^a}\mathbf{e}_k^a \end{aligned} \tag{D.49}$$

Substituting $\mathbf{u}_{k+1}^a = \mathbf{u}^{\star a} - \mathbf{e}_{k+1}^a$ results in

$$\begin{aligned}\mathbf{e}_{k+1}^a &= \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^a}\mathbf{e}_k^a \\ &= \left( \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^a}\Big|_{\mathbf{u}_k^a = \mathbf{u}^{\star a}} \right)^k \mathbf{e}_0^a \end{aligned} \tag{D.50}$$

Therefore as $k \to \infty$, the eigenvalues of the Jacobian $\mathbf{J}_\mathbf{F} = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k}$ and the conjugate Jacobian $\mathbf{J}_\mathbf{F}^c = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_k^*}$ matrices evaluated at the fixed point must be contained within the unit circle, for the error to diminish and FPI to converge. □

**Remark 2.** *The update algorithm $\mathbf{F}(\mathbf{u})$ is a contraction mapping on the unit hypersphere $\mathcal{S}_h \in \mathbb{C}^N$ and converges to a unique solution $\mathbf{u}^\star$ from any $\mathbf{u}_1 \in \mathcal{S}_h \in \mathbb{C}^N$.*

*Proof.* The two $N \times N$ Jacobian matrices of $\mathbf{F}$ and their respective eigenvalues are derived in Section D.A at the end of this Appendix. As both the Jacobian matrices contain only a single non-zero value at the $i$th diagonal element, the spectra of both matrices consist of a single non-zero eigenvalue with algebraic multiplicity of one and zero-valued eigenvalues with multiplicity of $(N-1)$, as shown in in Equation (D.61).

Following on from Lemma 2 it is apparent that the placement of the non-zero eigenvalues $\lambda$ and $\lambda^c$ given in (D.62) and (D.63) with respect to the unit circle $\mathcal{U}$ determine the convergence of the FPI for $\mathbf{F}(\mathbf{u}_i)$. A close inspection shows that the values of the latent sources along with the nonlinearity used in the FPI determine the convergence to the fixed points. Therefore, given $\{|\lambda|^2, |\lambda^c|^2\} < 1$, the update algorithm $\mathbf{F}$ is a contraction on the unit hypersphere $\mathcal{S}_h \in \mathbb{C}^N$ with $\gamma < 1$.

Then, $\mathbf{u} = \mathbf{F}(\mathbf{u})$ has a unique solution called the fixed point $\mathbf{u}^\star \in \mathcal{S}_h$ and the iteration

$$\mathbf{u}_{k+1} = \mathbf{F}(\mathbf{u}_k) \tag{D.51}$$

converges to $\mathbf{u}^\star$ for any starting value $\mathbf{u}_1 \in \mathcal{S}_h$. Considering the distance of the values at the $(k+1)$ update, $\mathbf{u}_{k+1}$, to the the fixed point $\mathbf{u}^\star$

$$\begin{aligned}\|\mathbf{u}_{k+1} - \mathbf{u}^\star\| &= \|\mathbf{F}(\mathbf{u}_k) - \mathbf{F}(\mathbf{u}^\star)\| \\ &\leq \gamma\|\mathbf{u}_k - \mathbf{u}^\star\| \quad \text{(2nd axiom of CMT)} \\ &\leq \gamma^k\|\mathbf{u}_1 - \mathbf{u}^\star\|\end{aligned}$$

and since $\lim_{k\to\infty} \gamma^k = 0$, then

$$\lim_{k\to\infty} \mathbf{u}_{k+1} = \mathbf{u}^\star. \tag{D.52}$$

In other words, after a sufficient number of updates, the distance to the unique solution reduces to zero. □

## D.6 Fixed Point Iteration in the Phase-Space

As discussed in Section D.3, the nc-FastICA algorithm can exhibit oscillations during convergence, occurring as the algorithm converges to several values with the same norm; this can be illustrated by using the phase-space approach. While convergence in the norm is usually used to assess the performance of algorithms, it is also useful to observe the convergence behaviour in phase-space. For example, in the study of the global asymptotic stability in linear systems, the effect of several conditions on the stability can be observed in the phase-space (geometric convergence), while this is not evident through the examination of convergence in the norm [169, 63].

In order to facilitate the study of convergence in phase-space, focus is given on an ICA problem with two latent sources and a $2 \times 2$ complex mixing matrix $\mathbf{A}$. As shown in Lemma 1 and Remark 1, at convergence there is a single non-zero value with unit magnitude in each of the columns of the modified demixing matrix

$$\mathbf{U} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2\end{bmatrix} = \begin{bmatrix}u_{11} & u_{21} \\ u_{12} & u_{22}\end{bmatrix} \in \mathbb{C}^{2\times 2}. \tag{D.53}$$

By observing one of the elements of $\mathbf{U}$, for example $u_{11}$, it is possible to construct a phase-space view of the fixed-point iteration and compare with convergence behaviour in the norm. In order to study the convergence in the norm, two measures are used: the convergence of the cost function $\mathcal{J}(\mathbf{u})$ to its maxima, and a measure quantifying the distance of $\mathbf{U}$ from the nearest permutation matrix [41]. It is expected that this value decreases as the algorithm converges to a solution. For the study of geometric convergence, a scatter plot of the value of $u_{11}$ at each iteration $k$, and the fixed point convergence error $|u_{11}^\star - u_{11}|$ are utilised.

From the simulations, it was observed that while the phase-space convergence behaviour of both the nc-FastICA and c-FastICA algorithms do not show strong dependence on the circularity of the signals, they are heavily dependent on the initial value

of the demixing matrix, degree of Gaussianity of the signal and the nonlinearity $G$. Also, in the analysis of mixtures with additive noise, while the algorithm performance deteriorated, the phase-space behaviour was similar to that in the noiseless case.

The convergence behaviour of the nc-FastICA algorithm is shown in Figure D.1. The mixtures of two complex sub-Gaussian sources were separated using the nonlinearity

$$G(y) = \frac{1}{a}\log\cosh(ay),\ a = 0.1 \tag{D.54}$$

after $k = 100$ iterations of the algorithm (D.6). The phase-space diagram in Figure D.1(a) and the fixed point convergence error curve in Figure D.1(b) (*top*) are shown for the $u_{11}$ element of the modified demixing matrix $\mathbf{U}$, while the distance of $\mathbf{U}$ to a permutation matrix and value of $\mathcal{J}(\mathbf{u_1})$ are shown respectively in Figure D.1(b) (*middle*) and Figure D.1(b) (*bottom*).

Figure D.1(a) shows that after $k = 4$ iterations, the algorithm achieved a limit cycle whereby the value of $y_{11}$ converged to values

$$\{1 \pm \jmath\epsilon, -1 \pm \jmath\epsilon\} = \{e^{\pm\jmath\varphi}\},\ \epsilon \ll 1$$

of unit norm, while oscillating between these fixed points.

This was also reflected in the convergence error curve in Figure D.1(b) (*top*), where the oscillation between the two fixed points is quantified as a distance with maximum attainable value of 2 due to the unit norm constraint of the algorithm. Observation of the convergence in the norm shows that the error diminishes to zero in Figure D.1(b) (*middle*), while in Figure D.1(b) (*bottom*) the cost function attains it maxima. Therefore, in correspondence with the results from the phase-space analysis, measures of convergence in the norm depict an initial convergence after around 4 iterations, however, they do not reflect the oscillatory convergence observed in the phase-space.

Next, the convergence behaviour for the separation of two super-Gaussian sources using the nc-FastICA algorithm for $k = 100$ iterations and using the nonlinearity $G$ in (D.54) was analysed and is shown in Figure D.2. In this scenario, the $u_{12}$ element of $\mathbf{U}$ was monitored, where it had a stable convergence in the phase-space after around 17 iterations, as seen in Figure D.2(a) and the convergence error curve of Figure D.2(b) (*top*). This observation is also in agreement with the diminishing distance of $\mathbf{U}$ to a permutation matrix and the convergence curve of the cost function to a local maxima (Figure D.2(b) (*middle*) and Figure D.2(b) (*bottom*)). In comparison with the previous experiment, it can be seen that while both scenarios demonstrate convergence in the norm, they have different behaviour in the phase-space; a limit cycle in the first experiment, and exponential convergence in the second experiment.

These simple experiments demonstrate the usefulness of the phase-space representation of the convergence behaviour together with the convergence analysis in the norm.

While the norm-based convergence analysis shows the proximity of the obtained solution to the true value, the geometric interpretation of the convergence behaviour can distinguish between the monotonic or oscillatory convergence.

## D.A Derivation of the eigenvalues of the Jacobian and conjugate Jacobian matrices of the FPI

The Jacobian $\mathbf{J_F}$ and conjugate Jacobian $\mathbf{J}^c_{\mathbf{F}}$ matrices for the FPI $\mathbf{F}(\mathbf{u}_{i,k})$ are given in (D.15). Denote $F_n$ as the $n$th element of the vector $\mathbf{F} = [F_1, \ldots, F_n, \ldots, F_N]^T$, then

$$\begin{aligned} F_n = &-E\{g(y_i y_i^*) y_i^* s_n\} \\ &+ E\{g'(y_i y_i^*)(y_i y_i^*) + g(y_i y_i^*)\} u_{in} \\ &+ E\{g'(y_i y_i^*) {y_i^*}^2\} \sum_{j=1}^{N} E\{s_n s_j\} u_{ij}^* \end{aligned} \tag{D.55}$$

where the iteration subscript $k$ is omitted for simplicity, the $u_{i\ell}$ is the $\ell$th element of $\mathbf{u}_i$ and $y_i = \mathbf{u}_i^H \mathbf{s}$. Using the chain rule for complex vectors within the $\mathbb{CR}$ calculus[4] , $\frac{\partial y_i}{\partial u_\ell} = 0$, $\frac{\partial y_i^*}{\partial u_\ell} = s_\ell^*$ and $\frac{\partial y_i}{\partial u_\ell^*} = s_\ell$, $\frac{\partial y_i^*}{\partial u_\ell^*} = 0$. Following the convention in [54], the rows of $\mathbf{J}$ are the derivatives of $F_n$ with respect to $\mathbf{u}_i$, so that

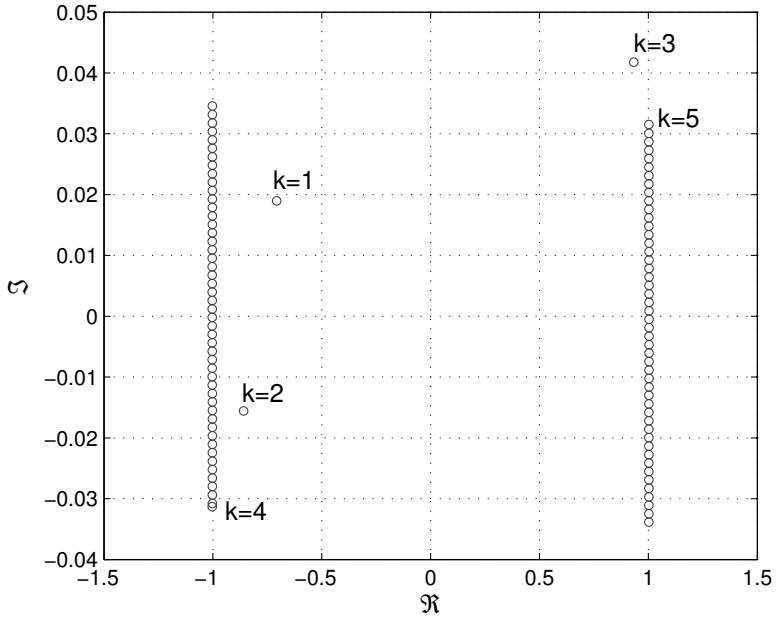
$$\mathbf{J_F} = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_i} = \begin{bmatrix} \frac{\partial F_1}{\partial u_{i1}} & \cdots & \frac{\partial F_1}{\partial u_{iN}} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_N}{\partial u_{i1}} & \cdots & \frac{\partial F_N}{\partial u_{iN}} \end{bmatrix} \in \mathbb{C}^{N \times N} \tag{D.56}$$

and follows similarly for $\mathbf{J}^c_{\mathbf{F}} = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_i^*}$.

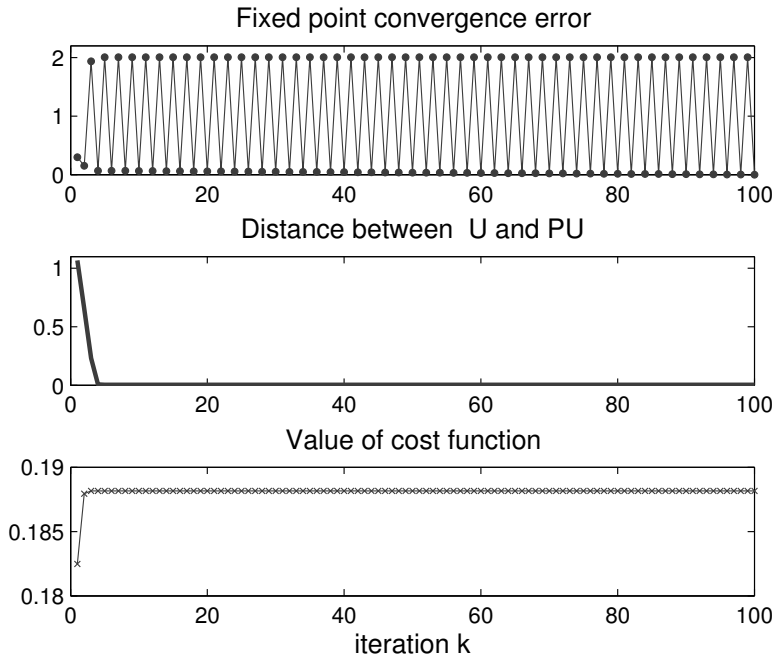
As the $\mathbb{CR}$ calculus applies to general complex functions, the two Jacobian matrices can be derived straightforwardly by noting that $\frac{\partial \mathbf{F}}{\partial \mathbf{u}_i} = \frac{\partial y_i^*}{\partial \mathbf{u}_i} \frac{\partial \mathbf{F}}{\partial y_i^*}$ and $\frac{\partial \mathbf{F}}{\partial \mathbf{u}_i^*} = \frac{\partial y_i}{\partial \mathbf{u}_i^*} \frac{\partial \mathbf{F}}{\partial y_i}$, to yield

$$\begin{aligned} \mathbf{J_F} = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_i} = &-E\big\{[g'(|y_i|^2)|y_i|^2 + g(|y_i|^2)]\mathbf{s}\mathbf{s}^T\big\} \\ &+ E\big\{[g''(|y_i|^2)|y_i|^2 y_i + 2g'(|y_i|^2) y_i]\mathbf{s}^*\big\}\mathbf{u}_i^T \\ &+ E\{g'(|y_i|^2)|y_i|^2 + g(|y_i|^2)\}\mathbf{I} \\ &+ E\big\{(\mathbf{s}^*\mathbf{u}^H)[g''(|y_i|^2)|y_i|^2 y_i^* + 2g'(|y_i|^2) y_i^*]\big\} E\{\mathbf{s}\mathbf{s}^T\} \end{aligned} \tag{D.57}$$

[4]For a complex vector-valued composite function $\mathbf{f} \circ \mathbf{g}$, the chain rule (B.13) states that $\frac{\partial \mathbf{f}(\mathbf{g})}{\partial \mathbf{z}} = \frac{\partial \mathbf{f}}{\partial \mathbf{g}} \frac{\partial \mathbf{g}}{\partial \mathbf{z}} + \frac{\partial \mathbf{f}}{\partial \mathbf{g}^*} \frac{\partial \mathbf{g}^*}{\partial \mathbf{z}}$ and $\frac{\partial \mathbf{f}(\mathbf{g})}{\partial \mathbf{z}^*} = \frac{\partial \mathbf{f}}{\partial \mathbf{g}} \frac{\partial \mathbf{g}}{\partial \mathbf{z}^*} + \frac{\partial \mathbf{f}}{\partial \mathbf{g}^*} \frac{\partial \mathbf{g}^*}{\partial \mathbf{z}^*}$.

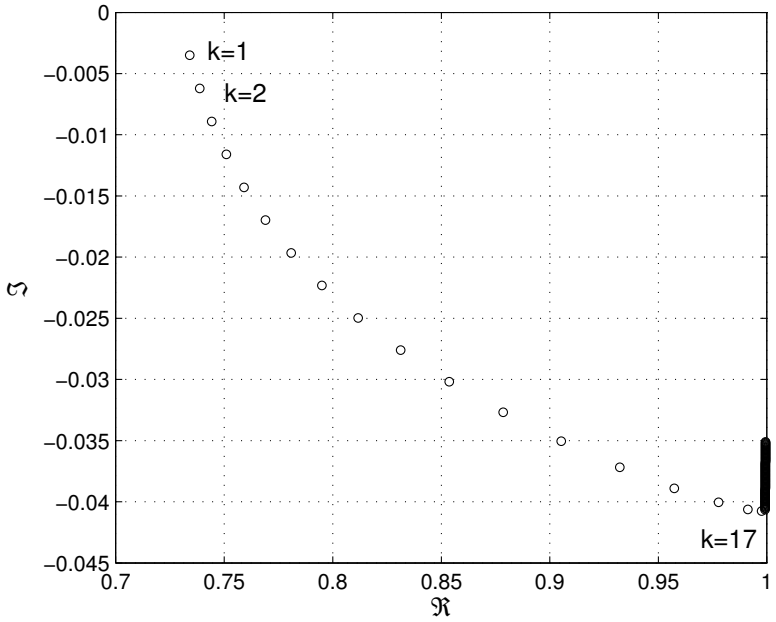



(a) Convergence of the $u_{11}$ element of $\mathbf{U}$ exhibiting a limit cycle

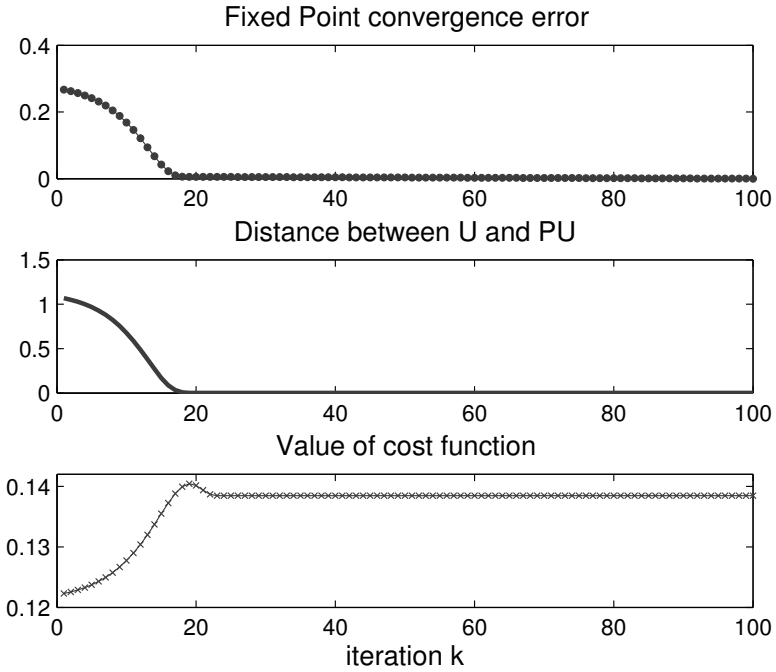



(b) *Top row:* The fixed point convergence error curve. *Middle row:* distance of $\mathbf{U}$ to the permutation matrix $\mathbf{PU}$. *Bottom row:* Convergence of the cost function $\mathcal{J}$ to a maximum.

**Figure D.1** Oscillatory convergence of the element $u_{11}$ of the modified demixing matrix $\mathbf{U}$, achieving a limit cycle when using the nc-FastICA algorithm in separating two sub-Gaussian sources based on the nonlinearity in (D.54).




(a) Stable convergence of the $u_{12}$ element of $\mathbf{U}$




(b) *Top row:* The fixed point convergence error curve. *Middle row:* distance of $\mathbf{U}$ to the permutation matrix $\mathbf{PU}$. *Bottom row:* Convergence of the cost function $\mathcal{J}$ to a maximum.

**Figure D.2** Stable convergence of the element $u_{12}$ of the modified demixing matrix $\mathbf{U}$, when using the nc-FastICA algorithm in separating two super-Gaussian sources based on the nonlinearity in (D.54).

and

$$\begin{aligned}\mathbf{J}_{\mathbf{F}}^{c} = \frac{\partial \mathbf{F}}{\partial \mathbf{u}_i^*} &= -E\{g'(|y_i|^2){y_i^*}^2 \mathbf{s}\mathbf{s}^T\} \\ &+ E\big\{[g''(|y_i|^2)|y_i|^2 y_i^* + 2g'(|y_i|^2)y_i^*]\mathbf{s}\big\}\mathbf{u}_i^T \\ &+ E\big\{(\mathbf{s}\mathbf{u}_i^H)g''(|y_i|^2){y_i^*}^3 + g'(|y_i|^2){y_i^*}^2\big\}E\{\mathbf{s}\mathbf{s}^T\}. \end{aligned} \tag{D.58}$$

Alternatively, the values of the elements of $\mathbf{J}_{\mathbf{F}}$ and $\mathbf{J}_{\mathbf{F}}^{c}$ can be found by considering the derivative of $F_n$ in (D.55) with respect to each element $u_{i\ell}$ as

$$\begin{aligned}\frac{\partial F_n}{\partial u_{i\ell}} &= -E\{g'(y_i y_i^*)y_i s_\ell^* y_i^* s_n + g(y_i y_i^*)s_\ell^* s_n\} \\ &+ E\{g''(y_i y_i^*)y_i s_\ell^* y_i y_i^* + g'(y_i y_i^*)y_i s_\ell^* + g'(y_i y_i^*)y_i s_\ell^*\}u_{in} \\ &+ E\{g'(y_i y_i^*)(y_i y_i^*) + g(y_i y_i^*)\}\frac{\partial u_{in}}{\partial u_{i\ell}} \\ &+ E\{g''(y_i y_i^*)y_i s_\ell^* {y_i^*}^2 + 2g'(y_i y_i^*)y_i^* s_\ell^*\}\sum_{j=1}^{N} E\{s_n s_j\}u_{ij}^* \end{aligned} \tag{D.59}$$

and

$$\begin{aligned}\frac{\partial F_n}{\partial u_{i\ell}^*} &= -E\{g'(y_i y_i^*)y_i^* s_\ell y_i^* s_n\} \\ &+ E\{g''(y_i y_i^*)y_i^* s_\ell y_i y_i^* + g'(y_i y_i^*)s_\ell y_i^* + g'(y_i y_i^*)y_i^* s_\ell\}u_{in} \\ &+ E\{g''(y_i y_i^*)y_i^* s_\ell {y_i^*}^2\}\sum_{j=1}^{N} E\{s_n s_j\}u_{ij}^* \\ &+ E\{g'(y_i y_i^*){y_i^*}^2\}\sum_{j=1}^{N} E\{s_n s_j\}\frac{\partial u_{ij}^*}{\partial u_{i\ell}^*}, \end{aligned} \tag{D.60}$$

where separate cases for the diagonal, $\ell = n$, and non-diagonal, $\ell \neq n$, elements of the two Jacobian matrices can be considered.

After substituting the value of the fixed point $\mathbf{u}_i^\star = [0, \ldots, e^{\jmath\varphi}, 0, \ldots, 0]^T$ and some simplifications, the non-diagonal values of $\mathbf{J}_{\mathbf{F}}$ and $\mathbf{J}_{\mathbf{F}}^{c}$ are evaluated as zero. Also, all the diagonal elements apart from the $i$th diagonal element are evaluated as zero. Therefore, the spectrum $\sigma$ of $\mathbf{J}_{\mathbf{F}}$ and $\mathbf{J}_{\mathbf{F}}^{c}$ consist of $(N-1)$ zero values and a single non-zero value denoted by $\lambda$ and $\lambda^c$, belonging respectively to the spectrum $\sigma(\cdot)$ of the Jacobian and conjugate Jacobian matrix. Thus

$$\begin{aligned}\sigma(\mathbf{J}_{\mathbf{F}}) &= \{\underbrace{0, \ldots, 0}_{(N-1)\text{ times}}, \lambda\} \\ \sigma(\mathbf{J}_{\mathbf{F}}^{c}) &= \{\underbrace{0, \ldots, 0}_{(N-1)\text{ times}}, \lambda^c\} \end{aligned} \tag{D.61}$$

and the value of the non-zero eigenvalues are given as

$$\begin{aligned}\lambda = \frac{\partial F_i}{\partial u_{ii}} &= -E\{g'(|s_i|^2)|s_i|^4 + g(|s_i|^2)|s_i|^2\} \\ &+ E\{g''(|s_i|^2)|s_i|^4 + 3g'(|s_i|^2)|s_i|^2 + g(|s_i|^2)\} \\ &+ E\{g''(|s_i|^2)|s_i|^2 s_i^{*^2} + 2g'(|s_i|^2)s_i^{*^2}\}E\{s_i s_i\} \end{aligned} \tag{D.62}$$

and

$$\begin{aligned}\lambda^c = \frac{\partial F_i}{\partial u_{ii}^*} &= -E\{g''(|s_i|^2)|s_i|^4 + 2g'(|s_i|^2)|s_i|^2\}e^{\jmath 2\varphi} \\ &+ E\{g''(|s_i|^2)|s_i|^2 s_i^{*^2}\}E\{s_i s_i\}. \end{aligned} \tag{D.63}$$

# Appendix E

# Blind Extraction of Improper Quaternion Sources

## E.1 Introduction

The extension of the widely linear model and augmented statistics to the four dimensional quaternion domain $\mathbb{H}$ has recently received plenty of attention due to its accuracy in modelling the coupling between signal components, and 3D rotation. In [130], the concept of proper quaternion random variables (also known as $\mathbb{Q}$-proper) was discussed as invariance of the probability distribution to rotations by angle $\frac{\pi}{2}$, and was generalised to any arbitrary angle in [170]. A unifying framework has recently been proposed in [132] which defines a set of four bases from which to construct augmented quaternion statistics, with a similar approach given in [145]. These bases can be seen as the quaternion analogue to the complex bases $\{\mathbf{z}, \mathbf{z}^*\}$ in augmented complex statistics, and allow for the exploitation of the complete second-order information present in quaternion signals. The quaternion widely linear model uses those bases to allow for the optimal minimum mean square error modelling of both $\mathbb{Q}$-proper and $\mathbb{Q}$-improper quaternion signals [132, 145, 135].

Existing blind source separation methodologies for the quaternion domain include a semi-blind block-based algorithm in [171] based on the calculation of rotation angle of whitened quaternion data, and the maximum likelihood approach in [137] where the choice of nonlinearities for the score function was discussed. On the other hand, blind source extraction (BSE) algorithms, designed so that only a few sources of interest from large-scale mixtures are recovered, are still in their infancy in $\mathbb{H}$ but have huge potential due to their ability to extract vector sources. Their introduction would offer both a reduced computational cost and will relax the need for further post-processing for the selection of the desired sources. This is especially important in real-world

applications, such as EEG conditioning for brain computer interfacing (BCI), where we may only be interested in removing artifacts from an observed mixture comprising of over 64 recording channels.

To this end, a class of BSE algorithms based on the local temporal structure of quaternion source signals is introduced. A quaternion widely linear predictor is used to extract both $\mathbb{Q}$-proper and $\mathbb{Q}$-improper sources, based on the smallest normalised prediction error, making such BSE independent of source powers. This is a generalisation of the complex widely linear prediction based BSE algorithm in Chapter 4, and is supported by simulations on both $\mathbb{Q}$-proper and $\mathbb{Q}$-improper signals.

## E.2 Quaternion Widely Linear Model

Consider the quaternion signal $y(k) = y_a(k) + \imath y_b(k) + \jmath y_c(k) + \kappa y_d(k)$, where $y_a(k), y_b(k), y_c(k)$ and $y_d(k)$ are real-valued scalars, and $\imath, \jmath$ and $\kappa$ are orthogonal unit vectors, where $\imath^2 = \jmath^2 = \kappa^2 = -1$. It has been shown that its optimal linear mean square estimate in terms of the observation $\mathbf{x}(k) \in \mathbb{H}^N$ is given by the widely linear model [132]. To show this, we can express the MSE estimator for a quaternion-valued signal $y \in \mathbb{H}$ in terms of the MSE estimators of its respective components, that is

$$\hat{y}_\alpha = E\{y_\alpha | x_a, x_b, x_c, x_d\}, \qquad \alpha = \{a, b, c\} \tag{E.1}$$

such that $\hat{y} = \hat{y}_a + \imath\hat{y}_b + \jmath\hat{y}_c + \kappa\hat{y}_d$. By employing the perpendicular involutions (self-inverse mappings) [138]

$$y^\beta = -\beta y \beta, \quad \beta = \{\imath, \jmath, \kappa\},$$

the MSE estimator in (E.1) can be written as[1]

$$\hat{y} = E\{y|x, x^\imath, x^\jmath, x^\kappa\} + \imath E\{y^\imath|x, x^\imath, x^\jmath, x^\kappa\} + \jmath E\{y^\jmath|x, x^\imath, x^\jmath, x^\kappa\} + \kappa E\{y^\kappa|x, q^\imath, x^\jmath, x^\kappa\}.$$

This results in the so called widely linear estimator

$$y(k) = \mathbf{h}^H(k)\mathbf{x}(k) + \mathbf{g}^H(k)\mathbf{x}^\imath(k) + \mathbf{u}^H(k)\mathbf{x}^\jmath(k) + \mathbf{v}^H(k)\mathbf{x}^\kappa(k) \tag{E.2}$$

where $\mathbf{h}, \mathbf{g}, \mathbf{u}$ and $\mathbf{v}$ are coefficient vectors and the symbol $(\cdot)^H$ denotes the Hermitian transpose operator. Thus, the complete second-order information in the observation $\mathbf{x}(k)$ is contained in the augmented covariance matrix

$$\mathcal{C}^a_{\mathbf{x}} = E\{\mathbf{x}^a \mathbf{x}^{aH}\} = \begin{bmatrix} \mathcal{C}_{\mathbf{xx}} & \mathcal{C}_{\mathbf{x}^\imath} & \mathcal{C}_{\mathbf{x}^\jmath} & \mathcal{C}_{\mathbf{x}^\kappa} \\ \mathcal{C}^H_{\mathbf{x}^\imath} & \mathcal{C}_{\mathbf{x}^\imath\mathbf{x}^\imath} & \mathcal{C}_{\mathbf{x}^\imath\mathbf{x}^\jmath} & \mathcal{C}_{\mathbf{x}^\imath\mathbf{x}^\kappa} \\ \mathcal{C}^H_{\mathbf{x}^\jmath} & \mathcal{C}_{\mathbf{x}^\jmath\mathbf{x}^\imath} & \mathcal{C}_{\mathbf{x}^\jmath\mathbf{x}^\jmath} & \mathcal{C}_{\mathbf{x}^\jmath\mathbf{x}^\kappa} \\ \mathcal{C}^H_{\mathbf{x}^\kappa} & \mathcal{C}_{\mathbf{x}^\kappa\mathbf{x}^\imath} & \mathcal{C}_{\mathbf{x}^\kappa\mathbf{x}^\jmath} & \mathcal{C}_{\mathbf{x}^\kappa\mathbf{x}^\kappa} \end{bmatrix} \in \mathbb{H}^{4N \times 4N} \tag{E.3}$$

[1] Since $y_a = \frac{1}{4}(y + y^\imath + y^\jmath + y^\kappa)$ , $y_b = \frac{1}{4}(y + y^\imath - y^\jmath - y^\kappa)$, $y_c = \frac{1}{4}(y - y^\imath + y^\jmath - y^\kappa)$ and $y_d = \frac{1}{4}(y - y^\imath - y^\jmath + y^\kappa)$ [132].

where $\mathbf{x}^a = [\mathbf{x}^T, \mathbf{x}^{\imath T}, \mathbf{x}^{\jmath T}, \mathbf{x}^{\kappa T}]^T$ is the augmented input vector. The matrices $\mathcal{C}_{\mathbf{x}^\imath}, \mathcal{C}_{\mathbf{x}^\jmath}, \mathcal{C}_{\mathbf{x}^\kappa}$ are called respectively the $\imath$-, $\jmath$- and $\kappa$-covariance matrices (or the pseudo-covariance matrices $\mathcal{C}_{\mathbf{x}^\beta} = E\{\mathbf{x}\mathbf{x}^{\beta H}\}$), while $\mathcal{C}_{\mathbf{xx}} = E\{\mathbf{x}\mathbf{x}^H\}$ is the standard covariance matrix. It is important to note that a $\mathbb{Q}$-proper random vector, $\mathbf{x}(k)$ is not correlated with its involutions; in this case the pseudo-covariance matrices vanish, and the augmented covariance matrix (E.3) becomes real-valued diagonal. A detailed account of the quaternion augmented statistics and WL model is provided in [132, 145, 135].

## E.3 Temporal BSE of Quaternion Signals

Consider the observation vector $\mathbf{x} \in \mathbb{H}^N$, a linear mixture of the latent sources $\mathbf{s} = [s_1, \ldots, s_N]^T \in \mathbb{H}^{N_s}$, given by

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) \tag{E.4}$$

where $\mathbf{A} \in \mathbb{H}^{N \times N_s}$ is the matrix of mixing coefficients. The sources are considered independent, with no assumptions made regarding their $\mathbb{Q}$-properness. The mixing matrix is assumed full rank and invertible, and is for simplicity considered to be square. Ideally, the recovered source $y(k) = \mathbf{w}^H\mathbf{x}(k)$, where $\mathbf{w}$ is a demixing vector such that $\mathbf{b}^H = \mathbf{w}^H\mathbf{A}$, has a single non-zero element $b_n$, corresponding to the $n$th source. If $\mathbf{x}(k)$ is whitened, then $b_n$ is of unit magnitude and an arbitrary rotation.

The proposed algorithm calculates the demixing vector $\mathbf{w}(k)$ by discriminating between the sources based on their degree of widely linear predictability, measured by the normalised mean square prediction error (MSPE); the extraction architecture is shown in Figure 4.1. The error $e(k)$ at the output of the widely linear predictor is given by

$$e(k) = y(k) - y_{WL}(k) \tag{E.5}$$

where $y_{WL}(k)$ is the widely linear predictor output, given in (E.2). The MSPE $E\{|e(k)|^2\}$ is normalised so that the relative temporal structure, and hence predictability, of the sources is unaffected by differences in the magnitude of the observed mixtures (scaling ambiguity), and the cost function is given by

$$\mathcal{J}(\mathbf{w}, \mathbf{h}, \mathbf{g}, \mathbf{u}, \mathbf{v}) = \frac{E\{|e(k)|^2\}}{E\{|y(k)|^2\}}. \tag{E.6}$$

Minimising this cost function with respect to the predictor coefficients results in differences between the prediction errors for various sources, and serves as a basis for

the proposed BSE. After some simplification, the MSPE can be expressed as

$$
\begin{aligned}
E\{|e(k)|^2\} &= \xi_0 - 2\sum_{m=1}^{M} \Re\Big\{\xi_m h_m(k) + \xi_{\imath|m} g_m(k) + \xi_{\jmath|m} u_m(k) + \xi_{\kappa|m} v_m(k)\Big\} \\
&+ 2\sum_{m,\ell=1}^{M} \Re\Big\{h_m^*(k)\xi_{\imath|\ell-m} g_\ell(k) + h_m^*(k)\xi_{\jmath|\ell-m} u_\ell(k) + h_m^*(k)\xi_{\kappa|\ell-m} v_\ell(k) \\
&\qquad + g_m^*(k)\xi_{\kappa|\ell-m}^{\imath} u_\ell(k) + g_m^*(k)\xi_{\jmath|\ell-m}^{\imath} v_\ell(k) + u_m^*(k)\xi_{\imath|\ell-m}^{\jmath} v_\ell(k)\Big\} \\
&+ \sum_{m,\ell=1}^{M} \Re\Big\{h_m^*(k)\xi_{\ell-m} h_\ell(k) + g_m^*(k)\xi_{\ell-m}^{\imath} g_\ell(k) + u_m^*(k)\xi_{\ell-m}^{\jmath} u_\ell(k) + v_m^*(k)\xi_{\ell-m}^{\kappa} v_\ell(k)\Big\}
\end{aligned}
\tag{E.7}
$$

where $\xi_{\alpha|\ell-m} \triangleq \mathbf{w}^H \mathbf{A}\mathcal{C}_{\mathbf{s}^\alpha}(\ell-m)\mathbf{A}^{\alpha H}\mathbf{w}^\alpha$ and $\xi_{\ell-m} \triangleq \mathbf{w}^H \mathbf{A}\mathcal{C}_{\mathbf{ss}}(\ell-m)\mathbf{A}^H\mathbf{w}$ and $\Re\{\cdot\}$ denotes the real or scalar part of a quaternion variable. The real-valued MSPE is related to the cross-correlation and cross-pseudo-correlation of the source components; as the sources are assumed orthogonal, these matrices are diagonal. For $\mathbb{Q}$-proper sources, the pseudo-covariances and thus the terms $\xi_{\alpha|\ell-m}$ vanish, simplifying the expression for the MSPE in (E.7).

A gradient based weight update based on the widely linear predictor is derived using the conjugate gradient within $\mathbb{HR}$ calculus [134], yielding

$$
\nabla_{\mathbf{w}^*}\mathcal{J} = \frac{1}{\sigma_y^2(k)}\Big(\breve{\mathbf{x}}_1(k)e^*(k) - \frac{1}{2}e(k)\breve{\mathbf{x}}_2(k) - \frac{\sigma_e^2(k)}{\sigma_y^2(k)}\big(\mathbf{x}(k)y^*(k) - \frac{1}{2}y(k)\mathbf{x}^*(k)\big)\Big) \tag{E.8}
$$

with

$$
\begin{aligned}
\breve{\mathbf{x}}_1(k) &= \mathbf{x}(k) - \sum_{m=1}^{M} h_m^*(k)\mathbf{x}(k-m) \\
\breve{\mathbf{x}}_2(k) &= \mathbf{x}^*(k) - \sum_{m=1}^{M} \big(\mathbf{x}^*(k-m)h_m(k) - \mathbf{x}^{\imath*}(k-m)g_m(k) \\
&\qquad - \mathbf{x}^{\jmath*}(k-m)u_m(k) - \mathbf{x}^{\kappa*}(k-m)v_m(k)\big).
\end{aligned}
\tag{E.9}
$$

The demixing vector $\mathbf{w}$ is then normalised to avoid spurious solutions. The moving average estimates $\sigma_y^2$ and $\sigma_e^2$ of the variance of $y(k)$ and $e(k)$ are given by

$$
\begin{aligned}
\sigma_e^2(k) &= \gamma_e\sigma_e^2(k-1) + (1-\gamma_e)|e(k)|^2 \\
\sigma_y^2(k) &= \gamma_y\sigma_y^2(k-1) + (1-\gamma_y)|y(k)|^2
\end{aligned}
\tag{E.10}
$$

where $\gamma_e$ and $\gamma_y$ are the respective forgetting factors[2].

[2] If $\mathbf{x}(k)$ is whitened, the source estimate power $\sigma_y^2(k) = 1$.

Finally, the gradient for the update of the widely linear predictor coefficients in Figure 4.1 is given by

$$\nabla_{\breve{\mathbf{w}}^{a*}} = \frac{1}{\sigma_y^2(k)}\Big( - \mathbf{y}^a(k)e^*(k) + \frac{1}{2}e(k)\mathbf{y}^{a*}(k)\Big) \tag{E.11}$$

where the vectors $\breve{\mathbf{w}}^a = [\mathbf{h}^T, \mathbf{g}^T, \mathbf{u}^T, \mathbf{v}^T]^T$, $\mathbf{y}(k) = [y(k-1), \ldots, y(k-L)]^T$, $\mathbf{y}^a(k) = [\mathbf{y}^T(k), \mathbf{y}^{\imath T}(k), \mathbf{y}^{\jmath T}(k), \mathbf{y}^{\kappa T}(k)]^T$ and $L$ is the predictor filter length. The algorithm in (E.11) is therefore a normalised variant of the WL-QLMS algorithm [135]. Note that in the derivation of the updates, non-commutativity of the quaternion multiplication should be taken into account. As desired, in the extraction of $\mathbb{Q}$-proper sources, the elements of $\breve{\mathbf{w}}^a$ become $\mathbf{h} \neq \mathbf{0}, \mathbf{g} = \mathbf{u} = \mathbf{v} = \mathbf{0}$.

## E.4 Simulations

To illustrate the performance of the proposed BSE algorithm two experimental settings were considered: synthetic benchmark data and real-world EEG data. In the first experiment, two $\mathbb{Q}$-improper benchmark sources of length $N_s = 1000$ were mixed using a random quaternion-valued square mixing matrix. Following [137], source $s_1$ was chosen as a pure phase-modulated 2 point cyclic polytope with improperness measure[3] $r_{s_1} = 1$, and source $s_2$ was an AR(4) signal generated using noncircular quaternion Gaussian noise, where $r_{s_2} = 0.44$. The sources were recovered using the proposed extraction algorithms in (E.8) and (E.11); the step-size was empirically chosen as $\mu_{\mathbf{w}} = 0.9$, predictor length $L = 10$, step-sizes for the WL predictor coefficient updates $\mu_{\breve{\mathbf{w}}^a} = 0.01$, and forgetting factors in (E.10) as $\gamma_e = \gamma_y = 0.975$. For these parameters, the MSPE of $s_1$ and $s_2$ were respectively 5.79 and 1.11. The performances were assessed using the Performance Index (PI) given in Equation (4.30). As desired, based on (E.11) the source $s_2$ with the smallest MSPE was first extracted, taking around 100 samples to converge to the PI of -43.24 dB, as shown in Figure E.1. When the same sources were extracted using the standard linear predictor the algorithm diverged, since due to the $\mathbb{Q}$-improperness of the sources the linear model was inadequate.

In the next experiment, the line noise and electroencephalogram (EOG) artifacts were extracted from an EEG mixture, recorded from 12 electrodes positioned according to the 10-20 system at AF8, AF4, AF7, AF3, C3, C4, PO7, PO3, PO4, PO8 and the left and right mastoids. In addition, 4 electrodes were placed around both eye sockets to directly record the reference EOG signals[4]. The frontal, central and occipital electrodes were combined into three 4-tuple quaternion-valued EEG signals. The widely

[3] The $\mathbb{Q}$-improperness index $r_s = \frac{\left|E\{ss^{\imath *}\}\right| + \left|E\{ss^{\jmath *}\}\right| + \left|E\{ss^{\kappa *}\}\right|}{3E\{ss^*\}}$ where $r_s \in [0, 1]$ and the value $r_s = 0$ indicates a $\mathbb{Q}$-proper source, while for a highly $\mathbb{Q}$-improper source $r_s = 1$.

[4] The EOG measurements were not part of the BSE process, they only served as a reference for performance assessment.

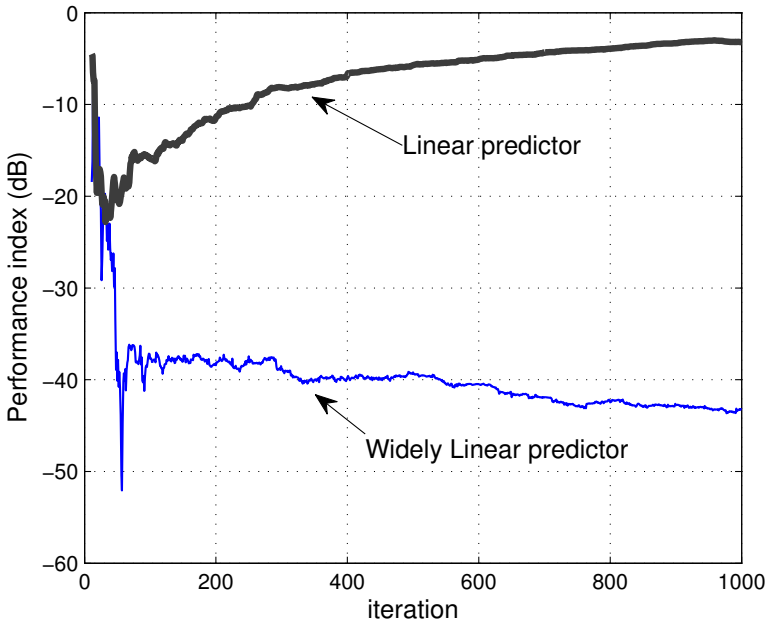



**Figure E.1** Learning curves for the quaternion BSE

linear predictor had $L = 10$ coefficients, step-sizes $\mu_{\mathbf{w}} = 0.9$ and $\mu_{\breve{\mathbf{w}}^a} = 9 \times 10^{-3}$, forgetting factors $\gamma_e = \gamma_y = 0.975$. Deflation was utilised to remove consecutive artifacts from the mixture; the real and imaginary components of the first and second extracted quaternion-valued signal contained respectively the line noise and EOG artifacts. The power spectra of the EOG artifact, extracted line noise and extracted EOG signal are shown in Figure E.2, with the boxed segments highlighting the extracted undesired components. The first extracted signal contained the 50Hz line noise, whereas the second extracted signal contains the EOG artifacts corresponding to the 1-8Hz activity. Figure E.3 shows the corresponding results for the strictly linear QLMS predictor; the bottom panel shows a 30 dB worse performance for the suppression of the power line noise.

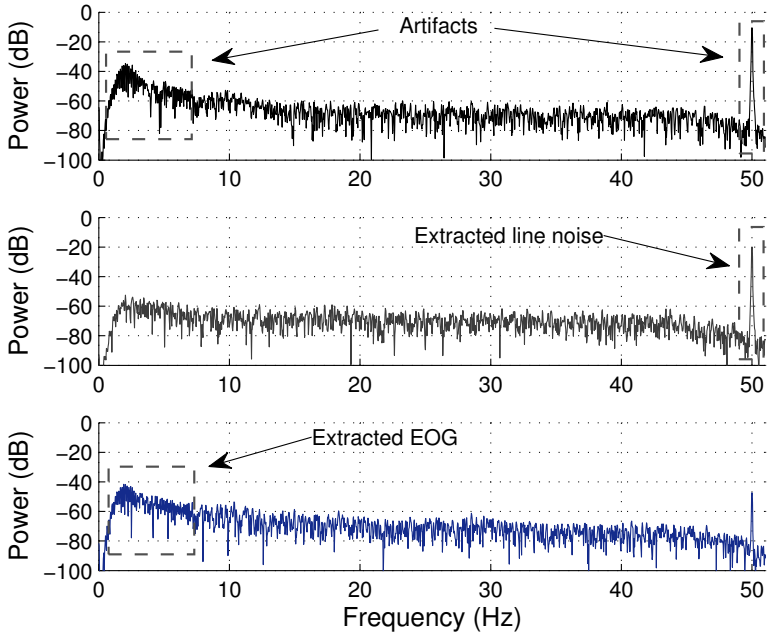



**Figure E.2** Power spectra of the reference EOG artifact (*top*), extracted line noise (*middle*) and extracted EOG (*bottom*) using the widely linear predictor.

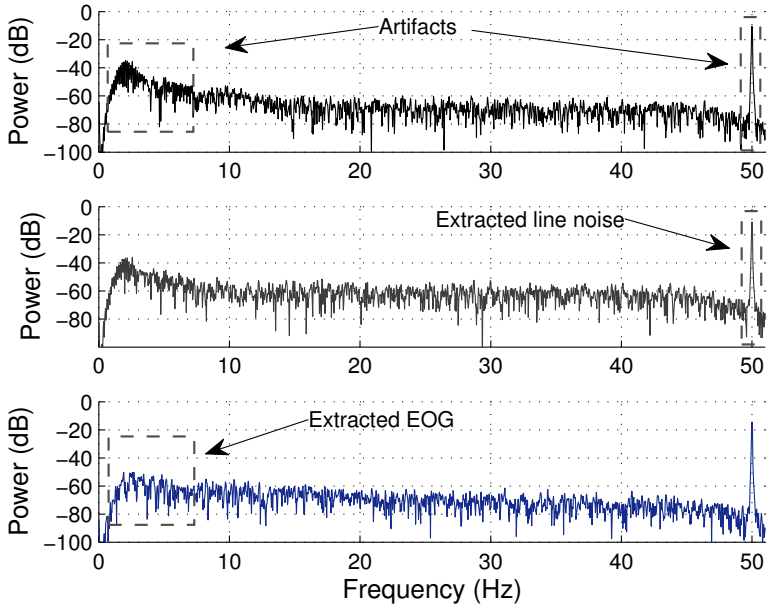



**Figure E.3** Power spectra of the reference EOG artifact (*top*), extracted line noise (*middle*) and extracted EOG (*bottom*) using the strictly linear predictor.

# References


[1] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 1996.

[2] P. S. R. Diniz. *Adaptive filtering: Algorithms and practical implementation*. Springer, 2008.

[3] W.-P. Ang and B. Farhang-Boroujeny. A new class of gradient adaptive step-size LMS algorithms. *IEEE Transactions on Signal Processing*, 49(4):805–810, 2001.

[4] D. P. Mandic. A generalized normalized gradient descent algorithm. *IEEE Signal Processing Letters*, 11(2):115–118, 2004.

[5] S. C. Douglas. Generalized gradient adaptive step sizes for stochastic gradient adaptive filters. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1396–1399, 1995.

[6] D. P. Mandic, A. I. Hanna, and M. Razaz. A normalized gradient descent algorithm for nonlinear adaptive filters using a gradient adaptive step size. *IEEE Signal Processing Letters*, 8(11):295–297, 2001.

[7] J. Arenas-Garcia, A. R. Figueiras-Vidal, and A. H. Sayed. Mean-square performance of a convex combination of two adaptive filters. *IEEE Transactions on Signal Processing*, 54(3):1078–1090, 2006.

[8] B. Jelfs, P. Vayanos, M. Chen, S. L. Goh, C. Boukis, T. Gautama, T. M. Rutkowski, T. Kuh, and D. P. Mandic. An online method for detecting nonlinearity within a signal. *Knowledge-Based Intelligent Information and Engineering Systems*, 4253/2006:1216–1223, 2006.

[9] B. Jelfs, S. Javidi, P. Vayanos, and D. P. Mandic. Characterisation of signal modality: Exploiting signal nonlinearity in machine learning and signal processing. *Journal of Signal Processing Systems*, 61(1):105–115, October 2010.

[10] A. Cichocki and S. Amari. *Adaptive Blind Signal and Image Processing, Learning Algorithms and Applications*. Wiley, 2002.

[11] D. P. Mandic, D. Obradovic, A. Kuh, T. Adalı, U. Trutschell, M. Golz, P. De Wilde, J. Barria, A. Constantinides, and J. Chambers. Data fusion for modern engineering applications: An overview. In *ICANN 2005*, volume 3697, pages 715–721. Springer, 2005.

[12] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley, 2001.

[13] J.-F. Cardoso. Multidimensional independent component analysis. In *ICASSP 1998*, volume 4, pages 1941–1944, 1998.

[14] A. Taleb and C. Jutten. Source separation in post-nonlinear mixtures. *IEEE Transactions on Signal Processing*, 47(10):2807–2820, 1999.

[15] W. Y. Leong and D. P. Mandic. Post-nonlinear blind extraction in the presence of ill-conditioned mixing. *IEEE Transactions on Circuits and Systems I*, 55:2631–2638, October 2008.

[16] J. Särelä and H. Valpola. Denoising source separation. *The Journal of Machine Learning Research*, 6:233–272, 2005.

[17] A. Hyvärinen. Fast independent component analysis with noisy data using Gaussian moments. In *International Symposium on Circuits and Systems*, pages 57–61, 1999.

[18] P. Comon. Blind identification and source separation in $2 \times 3$ under-determined mixtures. *IEEE Transactions on Signal Processing*, 52(1):11–22, 2004.

[19] L. De Lathauwer and J. Castaing. Blind identification of underdetermined mixtures by simultaneous matrix diagonalization. *IEEE Transactions on Signal Processing*, 56(3):1096–1105, 2008.

[20] P. Comon and M. Rajih. Blind identification of under-determined mixtures based on the characteristic function. *Signal Processing*, 86(9):2271–2281, September 2006.

[21] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[22] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, May 1999.

[23] J.-F. Cardoso. Source separation using higher order moments. In *ICASSP 1989*, volume 4, pages 2109–2112, 1989.

[24] J.-F. Cardoso and A. Souloumiac. Blind beamforming for non-Gaussian signals. *Radar and Signal Processing, IEE Proceedings F*, 140(6):362–370, 1993.

[25] D.-T. Pham, P. Garat, and C. Jutten. Separation of a mixture of independent sources through a maximum likelihood approach. In *EUSIPCO 92*, volume 2, pages 771–774, August 1992.

[26] A. J. Bell and T. J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[27] D.-T. Pham and P. Garat. Blind separation of mixture of independent sources through a quasi-maximum likelihood approach. *IEEE Transactions on Signal Processing*, 45(7):1712–1725, 1997.

[28] J.-F. Cardoso and B. H. Laheld. Equivariant adaptive source separation. *IEEE Transactions on Signal Processing*, 44(12):3017–3030, 1996.

[29] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, February 1998.

[30] S. Amari, A. Cichocki, and H. H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems*, pages 757–763. MIT Press, 1996.

[31] Q. Shi, R. Wu, and S. Wang. A novel approach to blind source extraction based on skewness. In *ICSP 2006*, volume 4, pages 3187–3190, November 2006.

[32] P. Georgiev and A. Cichocki. Robust blind source separation utilizing second and fourth order statistics. In *Artificial Neural Networks - ICANN 2002*, volume 2415, pages 1162–1167. Springer, 2002.

[33] A. Cichocki, R. Thawonmas, and S. Amari. Sequential blind signal extraction in order specified by stochastic properties. *Electronics Letters*, 33:64–65, 1997.

[34] W. Liu and D. P. Mandic. A normalised kurtosis-based algorithm for blind source extraction from noisy measurements. *Signal Processing*, 86(7):1580–1585, 2006.

[35] D. P. Mandic and A. Cichocki. An online algorithm for blind extraction of sources with different dynamical structures. In *4th Internation Symposium of Independent Component Analysis and Blind Signal Separation (ICA 2003)*, pages 645–650, 2003.

[36] B.-Y. Wang and W. X. Zheng. Blind extraction of chaotic signal from an instantaneous linear mixture. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 53(2):143–147, February 2006.

[37] B. Farhang-Boroujeny. *Adaptive Filters: Theory and Applications*. Wiley, 1998.

[38] B. Widrow and S. D. Stearns. *Adaptive Signal Processing*. Prentice-Hall, 1985.

[39] B. Widrow, J. M. McCool, and M. Ball. The complex LMS algorithm. *Proceedings of the IEEE*, 63(4):719–720, 1975.

[40] A. Tarighat and A. H. Sayed. Least mean-phase adaptive filters with application to communications systems. *IEEE Signal Processing Letters*, 11(2):220–223, 2004.

[41] E. Bingham and A. Hyvärinen. A fast fixed point algorithm for independent component analysis of complex valued signals. *Journal of Neural Systems*, 10:1–8, 2000.

[42] J. Anemüller, T. J. Sejnowski, and S. Makeig. Complex independent component analysis of frequency-domain electroencephalographic data. *Neural Networks*, 16(9):1311–1323, November 2003.

[43] B. Picinbono. On circularity. *IEEE Transactions on Signal Processing*, 42(12):3473–3482, 1994.

[44] F. D. Neeser and J. L. Massey. Proper complex random processes with applications to information theory. *IEEE Transactions on Information Theory*, 39(4):1293–1302, 1993.

[45] B. Picinbono. Second-order complex random vectors and normal distributions. *IEEE Transactions on Signal Processing*, 44(10):2637–2640, 1996.

[46] B. Picinbono and P. Chevalier. Widely linear estimation with complex data. *IEEE Transactions on Signal Processing*, 43(8):2030–2033, 1995.

[47] B. Picinbono and P. Bondon. Second-order statistics of complex signals. *IEEE Transactions on Signal Processing*, 45(2):411–420, 1997.

[48] P. J. Schreier and L. L. Scharf. Second-order analysis of improper complex random vectors and processes. *IEEE Transactions on Signal Processing*, 51(3):714–725, 2003.

[49] P. J. Schreier and L. L. Scharf. *Statistical Signal Processing of Complex-Valued Data*. Cambridge University Press, 2010.

[50] A. van den Bos. Complex gradient and Hessian. *IEE Proceedings of Vision, Image and Signal Processing*, 141(6):380–383, 1994.

[51] A. van den Bos. The multivariate complex normal distribution-a generalization. *IEEE Transactions on Information Theory*, 41(2):537–539, 1995.

[52] R. A. Wooding. The multivariate distribution of complex normal variables. *Biometrika*, 43(1-2):212–215, 1956.

[53] D. H. Brandwood. A complex gradient operator and its application in adaptive array theory. *IEE Proceedings F: Communications, Radar and Signal Processing*, 130(1):11–16, February 1983.

[54] K. Kreutz-Delgado. The complex gradient operator and the $\mathbb{CR}$-calculus. *Dept. of Electrical and Computer Engineering, UC San Diego, Course Lecture Supplement No. ECE275A*, pages 1–74, 2006.

[55] W. Wirtinger. Zur formalen theorie der funktionen von mehr komplexen veränderlichen. *Mathematische Annalen*, 97(1):357–375, December 1927.

[56] D. P. Mandic and J. A. Chambers. *Recurrent Neural Networks for Prediction*. John Wiley, 2001.

[57] S. L. Goh and D. P. Mandic. An augmented CRTRL for complex-valued recurrent neural networks. *Neural Networks*, 20(10):1061–1066, December 2007.

[58] S. L. Goh and D. P. Mandic. An augmented extended Kalman filter algorithm for complex-valued recurrent neural networks. *Neural Computation*, 19(4):1039–1055, 2007.

[59] S. L. Goh, M. Chen, D. H. Popovic, K. Aihara, D. Obradovic, and D. P. Mandic. Complex-valued forecasting of wind profile. *Renewable Energy*, 31(11):1733–50, 2006.

[60] S. L. Goh and D. P. Mandic. A complex-valued RTRL algorithm for recurrent neural networks. *Neural Computation*, 16(12):2699–2713, 2004.

[61] Y. Xia, C. Cheong Took, S. Javidi, and D. P. Mandic. A widely linear affine projection algorithm. In *IEEE Workshop on Statistical Signal Processing*, pages 373–376, 2009.

[62] C. Cheong Took and D. P. Mandic. Adaptive IIR filtering of noncircular complex signals. *IEEE Transactions of Signal Processing*, 57(10):4111–4118, October 2009.

[63] D. P. Mandic and S. L. Goh. *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models*. Wiley, 2009.

[64] N. Benvenuto and F. Piazza. On the complex backpropagation algorithm. *IEEE Transactions on Signal Processing*, 40(4):967–969, 1992.

[65] H. Leung and S. Haykin. The complex backpropagation algorithm. *IEEE Transactions on Signal Processing*, 39(9):2101–2104, 1991.

[66] G.M. Georgiou and C. Koutsougeras. Complex domain backpropagation. *IEEE Transactions on Circuits and Systems II*, 39(5):330–334, 1992.

[67] T. Kim and T. Adalı. Universal approximation of fully complex feed-forward neural networks. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 973–976, 2002.

[68] T. Kim and T. Adalı. Approximation by fully complex MLP using elementary transcendental activation functions. In *IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing XI*, pages 203–212, 2001.

[69] J. Eriksson and V. Koivunen. Complex random vectors and ICA models: Identifiability, uniqueness, and separability. *IEEE Transactions on Information Theory*, 52(3):1017–1029, 2006.

[70] R. A. Horn and C. A. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[71] S. C. Douglas. Fixed-point FastICA algorithms for the blind separation of complex-valued signal mixtures. In *Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, pages 1320–1325, 2005.

[72] E. Ollila and V. Koivunen. Complex ICA using generalized uncorrelating transform. *Signal Processing*, 89(4):365 – 377, 2009.

[73] E. Ollila, H. Oja, and V. Koivunen. Complex-valued ICA based on a pair of generalized covariance matrices. *Computational Statistics & Data Analysis*, 52(7):3789–3805, March 2008.

[74] M. Novey and T. Adalı. Complex ICA by negentropy maximization. *IEEE Transactions on Neural Networks*, 19(4):596–609, 2008.

[75] H. Li and T. Adalı. A class of complex ICA algorithms based on the kurtosis cost function. *IEEE Transactions on Neural Networks*, 19(3):408–420, 2008.

[76] T. Adalı, H. Li, M. Novey, and J.-F. Cardoso. Complex ICA using nonlinear functions. *IEEE Transactions on Signal Processing*, 56(9):4536–4544, 2008.

[77] H. Li and T. Adalı. Stability analysis of complex maximum likelihood ICA using Wirtinger calculus. In *ICASSP 2008*, pages 1801–1804, 2008.

[78] M. Novey and T. Adalı. On extending the complex FastICA algorithm to noncircular sources. *IEEE Transactions on Signal Processing*, 56(5):2148–2154, 2008.

[79] P. J. Schreier. Bounds on the degree of impropriety of complex random vectors. *IEEE Signal Processing Letters*, 15:190–193, 2008.

[80] E. Ollila. On the circularity of a complex random variable. *IEEE Signal Processing Letters*, 15:841–844, 2008.

[81] P. J. Schreier, L. L. Scharf, and A. Hanssen. A generalized likelihood ratio test for impropriety of complex signals. *IEEE Signal Processing Letters*, 13(7):433–436, July 2006.

[82] J. P. Delmas and H. Abeida. Asymptotic distribution of circularity coefficients estimate of complex random variables. *Signal Processing*, 89(12):2670–2675, December 2009.

[83] C. L. Nikias and A. P. Petropulu. *Higher-order spectra analysis: a nonlinear signal processing framework*. Prentice Hall, 1993.

[84] P. J. Schreier and L. L. Scharf. Higher-order spectral analysis of complex signals. *Signal Processing*, 86(11):3321–3333, November 2006.

[85] E. Ollila and V. Koivunen. Adjusting the generalized likelihood ratio test of circularity robust to non-normality. In *IEEE 10th Workshop on Signal Processing Advances in Wireless Communications*, pages 558–562, 2009.

[86] Y. Huang and J. Benesty. *Audio signal processing for next-generation multimedia communication systems*. Springer, 2004.

[87] R. Schober, W. H. Gerstacker, and L. H.-J. Lampe. A widely linear LMS algorithm for MAI suppression for DS-CDMA. *IEEE International Conference on Communications*, 4:2520–2525, 2003.

[88] R. Schober, W. H. Gerstacker, and L. H.-J. Lampe. Data-aided and blind stochastic gradient algorithms for widely linear MMSE MAI suppression for DS-CDMA. *IEEE Transactions on Signal Processing*, 52(3):746–756, 2004.

[89] S. L. Goh and D. P. Mandic. An augmented extended Kalman filter algorithm for complex-valued recurrent neural networks. In *Proceeding of IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 5, pages 561–564, 2006.

[90] D. P. Mandic, S. Still, and S. C. Douglas. Duality between widely linear and dual channel adaptive filtering. In *ICASSP 2009*, pages 1729–1732, 2009.

[91] S. M. Hammel, C. Jones, and J. V. Moloney. Global dynamical behavior of the optical field in a ring cavity. *Optical Society of America, Journal B: Optical Physics*, 2:552–564, 1985.

[92] S. Haykin and Liang Li. Nonlinear adaptive prediction of nonstationary signals. *IEEE Transactions on Signal Processing*, 43(2):526–535, 1995.

[93] J. A. Chambers, O. Tanrikulu, and A. G. Constantinides. Least mean mixed-norm adaptive filtering. *Electronics Letters*, 30(19):1574–1575, 1994.

[94] D. P. Mandic, P. Vayanos, C. Boukis, B. Jelfs, S. L. Goh, T. Gautama, and T. M. Rutkowski. Collaborative adaptive learning using hybrid filters. In *ICASSP 2007*, volume 3, pages 921–924, 2007.

[95] D. P. Mandic, M. Golz, A. Kuh, D. Obradovic, and T. Tanaka, editors. *Signal Processing Techniques for Knowledge Extraction and Information Fusion*. Springer, 2008.

[96] D. P. Mandic, P. Vayanos, S. Javidi, B. Jelfs, and K. Aihara. Online tracking of the degree of nonlinearity within complex signals. In *ICASSP 2008*, pages 2061–2064, April 2008.

[97] P. Vayanos, S. L. Goh, and D. P. Mandic. Online detection of the nature of complex-valued signals. In *Proceedings of the 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pages 173–178, 2006.

[98] B. Jelfs, S. Javidi, S. L. Goh, and D. P. Mandic. Collaborative adaptive filters for online knowledge extraction and information fusion. chapter 1, pages 3–21. Springer, March 2008.

[99] T. Kim and T. Adalı. Fully complex backpropagation for constant envelope signal processing. In *Proceedings of the 2000 IEEE Signal Processing Society Workshop Neural Networks for Signal Processing*, volume 1, pages 231–240, 2000.

[100] L. Tong, R. W. Liu, V. C. Soon, and Y. F. Huang. Indeterminacy and identifiability of blind identification. *IEEE Transactions on Circuits and Systems*, 38(5):499–509, 1991.

[101] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–444, 1997.

[102] A. Cichocki and R. Thawonmas. On-line algorithm for blind signal extraction of arbitrarily distributed, but temporally correlated sources using second order statistics. *Neural Processing Letters*, 12(1):91–98, August 2000.

[103] W. Liu, D. P. Mandic, and A. Cichocki. Blind source extraction of instantaneous noisy mixtures using a linear predictor. In *Proc. IEEE International Symposium on Circuits and Systems*, pages 4199–4202, 2006.

[104] W. Liu, D. P. Mandic, and A. Cichocki. Blind second-order source extraction of instantaneous noisy mixtures. *IEEE Transactions of Circuits and Systems II*, 53(9):931–935, 2006.

[105] W. Y. Leong, W. Liu, and D. P. Mandic. Blind source extraction: Standard approaches and extensions to noisy and post-nonlinear mixing. *Neurocomputing*, 71:2344 – 2355, 2008.

[106] S. Javidi, M. Pedzisz, S. L. Goh, and D. P. Mandic. The augmented complex least mean square algorithm with application to adaptive prediction problems. In *Proc. 1st IARP Workshop on Cognitive Information Processing*, pages 54–57, 2008.

[107] P. Georgiev, A. Cichocki, and H. Bakardjian. *Optimization Techniques for Independent Component Analysis with Applications to EEG Data*, chapter 3, pages 53–68. Quantitative Neuroscience: Models, Algorithms, Diagnostics, and Therapeutic Applications. Kluwer Academic Publishers, 2004.

[108] N. Delfosse and P. Loubaton. Adaptive blind separation of independent sources: A deflation approach. *Signal Processing*, 45(1):59–83, July 1995.

[109] S. Y. Kung and C. Mejuto. Extraction of independent components from hybrid mixture: Kuicnet learning algorithm and applications. In *ICASSP 1998*, volume 2, pages 1209–1212, 1998.

[110] R. Thawonmas, A. Cichocki, and S. Amari. A cascade neural network for blind signal extraction without spurious equilibria. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 81(9):1833–1846, 1998.

[111] M. H. Hayes. *Statistical Digital Signal Processing and Modeling*. Wiley, 1996.

[112] R. N. Vigário. Extraction of ocular artefacts from EEG using independent component analysis. *Electroencephalography and Clinical Neurophysiology*, 103(3):395–404, 1997.

[113] T. P. Jung, S. Makeig, C. Humphries, T. W. Lee, M. J. Mckeown, V. Iragui, and T. J. Sejnowski. Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, 37(02):163–178, 2000.

[114] A. Delorme, S. Makeig, and T. Sejnowski. Automatic artifact rejection for EEG data using high-order statistics and independent component analysis. In *International Workshop on ICA*, pages 457–462, 2001.

[115] G. Barbati, C. Porcaro, F. Zappasodi, P. M. Rossini, and F. Tecchio. Optimization of an independent component analysis approach for artifact identification and removal in magnetoencephalographic signals. *Clinical Neurophysiology*, 115(5):1220–1232, 2004.

[116] A. Greco, N. Mammone, F. C. Morabito, and M. Versaci. Semi-automatic artifact rejection procedure based on kurtosis, Renyi's entropy and independent component scalp maps. In *International Enformatika Conference*, pages 22–26, 2005.

[117] A. Delorme, T. Sejnowski, and S. Makeig. Enhanced detection of artifacts in EEG data using higher-order statistics and independent component analysis. *NeuroImage*, 34(4):1443–1449, 2007.

[118] P. S. Kumar, R. Arumuganathan, K. Sivakumar, and C. Vimal. An adaptive method to remove ocular artifacts from EEG signals using wavelet transform. *Journal of Applied Sciences Research*, 5:711–745, 2009.

[119] M.G. Jafari and J.A. Chambers. Fetal electrocardiogram extraction by sequential source separation in the wavelet domain. *IEEE Transactions on Biomedical Engineering*, 52(3):390–400, March 2005.

[120] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu. The Empirical Mode Decomposition and the Hilbert Spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A*, 454(1971):903–995, March 1998.

[121] N. E. Huang and S. S. Shen. *Hilbert-Huang transform and its applications*. World Scientific, 2005.

[122] N. Rehman and D. P. Mandic. Multivariate empirical mode decomposition. *Proceedings of the Royal Society A*, 466:1291–1302, 2010.

[123] J. A. Palmer, S. Makeig, and K. Kreutz-Delgado. A complex cross-spectral distribution model using normal variance mean mixtures. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3569–3572, 2009.

[124] N. Mitianoudis, T. Stathaki, and A. G. Constantinides. Smooth signal extraction from instantaneous mixtures. *IEEE Signal Processing Letters*, 14(4):271–274, 2007.

[125] L. T. Duarte, B. Rivet, and C. Jutten. Blind extraction of smooth signals based on a second-order frequency identification algorithm. *IEEE Signal Processing Letters*, 17(1):79–82, 2010.

[126] R. A. Adams and J. J. F. Fournier. *Sobolev spaces*. Academic Press, 1975.

[127] D. P. Mandic, S. Javidi, G. Souretis, and S. L. Goh. Why a complex valued solution for a real domain problem. In *IEEE Workshop on Machine Learning for Signal Processing*, pages 384–389, August 2007.

[128] J. P. Ward. *Quaternions and Cayley numbers*. Kluwer Academic Publishers, 1997.

[129] S. Sangwine and N. Le Bihan. Quaternion polar representation with a complex modulus and complex argument inspired by the Cayley-Dickson form. *Advances in Applied Clifford Algebras*, 20(1):111–120, March 2010.

[130] N. N. Vakhania. Random vectors with values in quaternion Hilbert spaces. *Theory of Probability and its Applications*, 43(1):99–115, January 1999.

[131] N. Le Bihan and P. O. Amblard. Detection and estimation of Gaussian proper quaternion valued random processes. In *7th IMA Conference on Mathematics in Signal Processing, Cirencester, UK*, 2006.

[132] C. Cheong Took and D. P. Mandic. Augmented second order statistics of quaternion random signals. *Signal Processing*, 91(2):214–224, February 2011.

[133] C. Jahanchahi, C. Cheong Took, and D. P. Mandic. On $\mathbb{HR}$ calculus, quaternion valued stochastic gradient, and adaptive three dimensional wind forecasting. In *International Joint Conference on Neural Networks*, pages 3154–3158, 2010.

[134] D. P. Mandic, C. Jahanchahi, and C. Cheong Took. A quaternion gradient operator and its applications. *IEEE Signal Processing Letters*, 2010 (accepted).

[135] C. Cheong Took and D. P. Mandic. A quaternion widely linear adaptive filter. *IEEE Transactions on Signal Processing*, 58(8):4427–4431, August 2010.

[136] B. Che-Ujang, C. Cheong Took, and D. P. Mandic. Split quaternion nonlinear adaptive filtering. *Neural Networks*, 23(3):426–434, April 2010.

[137] N. Le Bihan and S. Buchholz. Quaternionic independent component analysis using hypercomplex nonlinearities. In *7th IMA Conference on Mathematics in Signal Processing*, 2006.

[138] T. A. Ell and S. J. Sangwine. Quaternion involutions and anti-involutions. *Computers & Mathematics with Applications*, 53(1):137–143, January 2007.

[139] F. Zhang. Quaternions and matrices of quaternions. *Linear Algebra and its Applications*, 251:21–57, January 1997.

[140] A. Sudbery. Quaternionic analysis. *Mathematical Proceedings of the Cambridge Philosophical Society*, 85(2):199–225, 1979.

[141] S. De Leo and P. P. Rotelli. Quaternionic analyticity. *Applied Mathematics Letters*, 16(7):1077–1081, October 2003.

[142] L. H. Zetterberg and H. Brändström. Codes for combined phase and amplitude modulated signals in a four-dimensional space. *IEEE Transactions on Communications*, 25(9):943–950, 1977.

[143] Md. K. I. Molla, T. Tanaka, T. M. Rutkowski, and A. Cichocki. Separation of EOG artifacts from EEG signals using bivariate EMD. In *ICASSP 2010*, pages 562–565, 2010.

[144] S. Zhang and A. G. Constantinides. Lagrange programming neural networks. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 39(7):441–452, 1992.

[145] J. Vía, D. Ramírez, and I. Santamaría. Properness and widely linear processing of quaternion random vectors. *IEEE Transactions on Information Theory*, 56(7):3502–3515, 2010.

[146] W. Liu, D. P. Mandic, and A. Cichocki. Analysis and online realization of the CCA approach for blind source separation. *IEEE Transactions on Neural Networks*, 18(5):1505–1510, 2007.

[147] W. Liu, D. P. Mandic, and A. Cichocki. Blind source separation based on generalised canonical correlation analysis and its adaptive realization. In *Congress on Image and Signal Processing*, volume 5, pages 417–421, 2008.

[148] S. Javidi, C. Cheong Took, C. Jahanchahi, N. Le Bihan, and D. P. Mandic. Blind extraction of improper quaternion sources. In *International Conference on Acoustics, Speech, and Signal Processing*, 2011 (in submission).

[149] T. W. Lee and M. S. Lewicki. The generalized gaussian mixture model using ICA. In *International Workshop on Independent Component Analysis*, pages 239–244, 2000.

[150] M. Z. Coban and R. M. Mersereau. Adaptive subband video coding using bivariate generalized gaussian distribution model. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 1990–1993, 1996.

[151] M. Novey, T. Adalı, and A. Roy. A complex generalized Gaussian distribution — characterization, generation, and estimation. *IEEE Transactions on Signal Processing*, 58(3):1427–1433, 2010.

[152] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley, 1991.

[153] A. Hjørungnes and D. Gesbert. Complex-valued matrix differentiation: Techniques and key results. *IEEE Transactions on Signal Processing*, 55(6):2740–2746, 2007.

[154] A. Hjørungnes, D. Gesbert, and D. P. Palomar. Unified theory of complex-valued matrix differentiation. In *ICASSP 2007*, volume 3, pages 345–348, 2007.

[155] T. Adalı and H. Li. A practical formulation for computation of complex gradients and its application to maximum likelihood. In *ICASSP 2007*, pages 633–636, 2007.

[156] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins University Press, 1996.

[157] D. P. Mandic and I. Yamada. Tutorial lecture : Machine learning and signal processing applications of fixed point theory. In *IEEE ICASSP 2007: Tutorial Textbook*, pages 1–135, 2007.

[158] M. Novey and T. Adalı. On quantifying the effects of noncircularity on the complex FastICA algorithm. In *ICASSP 2008*, pages 1809–1812, 2008.

[159] S. C. Douglas. On the convergence behavior of the FastICA algorithm. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 409–414, 2003.

[160] A. T. Erdogan. On the convergence of ICA algorithms with symmetric orthogonalization. In *ICASSP 2008*, pages 1925–1928, April 2008.

[161] E. Oja and Z. Yuan. The FastICA algorithm revisited: Convergence analysis. *IEEE Transactions on Neural Networks*, 17(6):1370–1381, November 2006.

[162] H. Shen, M. Kleinsteuber, and K. Huper. Local convergence analysis of FastICA and related algorithms. *IEEE Transactions on Neural Networks*, 19(6):1022–1032, 2008.

[163] P. A. Regalia and E. Kofidis. Monotonic convergence of fixed-point algorithms for ICA. *Neural Networks, IEEE Transactions on*, 14(4):943–949, July 2003.

[164] A. T. Erdogan. On the convergence of ICA algorithms with symmetric orthogonalization. *IEEE Transactions on Signal Processing*, 57(6):2209–2221, June 2009.

[165] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[166] D. P. Mandic. Data-reusing recurrent neural adaptive filters. *Neural Computation*, 14(11):2693–2707, 2002.

[167] J. E. Dennis and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial Mathematics (SIAM) Press, 1996.

[168] A. Ferrante, A. Lepschy, and U. Viaro. Convergence analysis of a fixed-point algorithm. *Italian Journal of Pure and Applied Mathematics*, 9:179–186, 2001.

[169] D. P. Mandic and J. A. Chambers. On stability of relaxive systems described by polynomials with time-variant coefficients. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 47:1534–1537, 2000.

[170] P. O. Amblard and N. Le Bihan. On properness of quaternion valued random variables. In *IMA Conference on Mathematics in Signal Processing*, 2004.

[171] V. Zarzoso and A. K. Nandi. Closed-form semi-blind separation of three sources from three real-valued instantaneous linear mixtures via quaternions. In *Sixth International Symposium on Signal Processing and its Applications*, volume 1, pages 1–4, 2001.