MIMO Instantaneous Blind Identification and Separation based on Arbitrary Order Temporal Structure in the Data

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
Rector Magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op woensdag 21 november 2007 om 16.00 uur

door

Jakob van de Laar

geboren te ’s-Hertogenbosch

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. J.W.M. Bergmans
en
prof.dr.ir. M. Moonen

Copromotor:
dr.ir. P.C.W. Sommen

© Copyright 2007 Jakob van de Laar
All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission from the copyright owner.

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Laar, Jakob van de

MIMO Instantaneous Blind Identification and Separation based on Arbitrary Order Temporal Structure in the Data / by Jakob van de Laar. - Eindhoven : Technische Universiteit Eindhoven, 2007.
Proefschrift.
ISBN 978-90-386-1159-4
NUR 959
Trefwoorden: digitale signaalverwerking / tijdreeksen / algebraïsche vergelijkingen / homotopie / datacommunicatie / identificeerbaarheid.
Subject headings: array signal processing / MIMO systems / blind source separation / higher order statistics / homotopy.

MIMO Instantaneous Blind Identification and Separation based on Arbitrary Order Temporal Structure in the Data

"There is a theory which states that if ever anyone discovers exactly what the Universe is for and why it is here, it will instantly disappear and be replaced by something even more bizarre and inexplicable. There is another theory which states that this has already happened."

- Douglas Adams (1952-2001) -

Samenstelling promotiecommissie:

Prof. dr. ir. A.C.P.M. Backx (voorzitter)
Department of Electrical Engineering, Technische Universiteit Eindhoven (TU/e)

Prof. dr. ir. J.W.M. Bergmans (eerste promotor)
Signal Processing Systems Group, Department of Electrical Engineering, Technische Universiteit Eindhoven (TU/e)

Prof. dr. ir. M. Moonen (tweede promotor)
Digital Signal Processing, Department of Electrical Engineering, Katholieke Universiteit Leuven (KUL)

Dr. ir. P.C.W. Sommen (co-promotor)
Signal Processing Systems Group, Department of Electrical Engineering, Technische Universiteit Eindhoven (TU/e)

Prof. dr. ir. A.-J. van der Veen
Circuits and Systems Group, Faculty of Electrical Engineering, Mathematics & Computer Science, Technische Universiteit Delft (TUD)

Prof. dr. ir. H.P. Urbach
Optics Research Group, Department Imaging Science & Technology, Technische Universiteit Delft (TUD)

Prof. dr. A.G. Tijhuis
Electromagnetics, Department of Electrical Engineering, Technische Universiteit Eindhoven (TU/e)

Ing. C.P. Janse
Digital Signal Processing Group, Philips Research Laboratories, Eindhoven

Summary

MIMO Instantaneous Blind Identification and Separation based on Arbitrary Order Temporal Structure in the Data

This thesis is concerned with three closely related problems. The first one is called Multiple-Input Multiple-Output (MIMO) Instantaneous Blind Identification, which we denote by MIBI. In this problem a number of mutually statistically independent source signals are mixed by a MIMO instantaneous mixing system and only the mixed signals are observed, i.e. both the mixing system and the original sources are unknown or 'blind'. The goal of MIBI is to identify the MIMO system from the observed mixtures of the source signals only. The second problem is called Instantaneous Blind Signal Separation (IBSS) and deals with recovering mutually statistically independent source signals from their observed instantaneous mixtures only. The observation model and assumptions on the signals and mixing system are the same as those of MIBI. However, the main purpose of IBSS is the estimation of the source signals, whereas the main purpose of MIBI is the estimation of the mixing system. If the number of source signals is not larger than the number of sensors, the source signals can be recovered by applying the inverse of the estimated mixing system to the observed mixtures. Hence, from this point of view IBSS is merely a direct application of MIBI. The third problem is called Instantaneous Semi-Blind Source Localization (ISBSL) and concerns the localization of a set of narrowband sources from their observed instantaneous mixtures only. In this case, the instantaneous mixing system is parameterized by the source position parameters. Hence, narrowband ISBSL can be considered as a parameterized version of MIBI in which more a priori knowledge is available than in the general 'fully blind' MIBI case. For this reason we call this problem 'semi-blind'. Because MIBI is a kind of abstraction or generalization of both IBSS and ISBSL, the main focus in this work is on MIBI, while IBSS and ISBSL are considered as applications or examples.

Many methods and algorithms for performing MIBI, IBSS and ISBSL have been developed during the last decade. Until now, mainly three different approaches have been used. These are based on the following properties of the source signals, several of which are related to the guiding principle of statistical independence: non-Gaussianity, second order spatial uncorrelatedness in combination with temporal non-whiteness/correlatedness, and second order spatial uncorrelatedness in combination with second order non-stationarity. What is lacking from those approaches is the exploitation of higher order temporal structure in the data, such as higher order non-whiteness/correlatedness and non-stationarity. Some methods for exploiting Higher Order Temporal Structure (HOTS) exist, but usually these are quite specific. In addition, most blind methods described in the literature cannot deal with a MIBI scenario of great interest, viz. one with more sources than sensors. In this work we present a unifying framework for exploiting arbitrary order temporal structure in the signals by means of cumulant functions, which possess convenient mathematical properties such as multilinearity. The MIBI problem is formulated in such a way that any kind of temporal structure in the data, such as arbitrary order non-stationarity and non-whiteness, is exploited in a unified manner. Based

on physically plausible assumptions on the temporal structure of the source and noise signals, and applying subspace techniques to a subspace matrix containing cumulant values arranged in a specific manner, it is shown that the MIBI problem can be 'projected onto' two dual mathematical problems in the sense that solving these problems solves the MIBI problem. In the first problem, MIBI is projected onto the problem of solving a system of multivariate homogeneous polynomial or so-called polyconjugal (polynomial-like) equations. The number of variables and the degree (of homogeneity) of the functions in the system equal the number of sensors and the order of the considered statistics, respectively. In the second problem, MIBI is projected onto the problem of solving a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem that is dual to the first problem. Possible solution approaches for those problems are described. In particular, a so-called homotopy method is used for solving the system of equations. Because of the connection between the system of homogeneous equations on the one hand, and the MMGEVD problem on the other hand, this in fact solves both mathematical problems.

The theory developed in this thesis is unifying in several senses. Firstly, it is general with respect to the order of the exploited temporal structure in the sense that the mathematical problem formulation has the same structure for any considered order. Secondly, all types of statistical variability in the data, such as arbitrary order non-stationarity and non-whiteness, are exploited in a unified manner. Finally, for complex-valued signals, the conjugation pattern of the arguments of the involved cumulant functions can be chosen arbitrarily. In practice, this should be done in accordance with the characteristics of the involved signals. Our approach allows the identification of a mixing system with more sources than sensors, even with second order statistics. Depending on the number of sensors, the order and type of the exploited temporal structure, the chosen conjugation pattern, and the arrangement of the statistics in the subspace matrix, a certain maximum number of mixing matrix columns can be determined that usually exceeds the number of sensors for orders larger than one. We provide insight into the computation of the maximum number of identifiable columns as a function of the number of sensors, the employed statistical order, and the conjugation pattern. The rationale behind the work presented in this thesis is based on providing insight and highlighting the geometric and algebraic structure of the different problem formulations that are developed. Therefore, the practical problems associated with the use of estimated sensor cumulants are not discussed in great detail. Nevertheless, the theory is directly applicable to many practical scenarios. This is demonstrated by various examples, including the identification of an instantaneous mixing system for different types of signals, the separation of instantaneously mixed artificial random signals, speech signals and images, Direction Of Arrival (DOA) estimation, etcetera. Finally, the theory allows us to make trade-offs between various related quantities such as the arrangement of the statistics in the subspace matrix, maximum number of identifiable mixing matrix columns, number of sensors, exploited type(s) of temporal structure, exploited order(s) of temporal structure, employed conjugation pattern(s), number of samples required for reliable estimation of the involved statistics, and so on.

Contents

Summary **i**

List of Figures **ix**

List of Tables **xiii**

List of Algorithms **xv**

Glossary **xvii**
List of Acronyms . . . xvii
List of Abbreviations used in Lists . . . xviii
List of Textual abbreviations . . . xviii
List of Mathematical Symbols and Variables . . . xix
List of Mathematical Operator Names . . . xxii

1 Introduction **1**
1.1 Main thesis topics and applications . . . 3
1.1.1 Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI) . . . 3
1.1.2 Instantaneous Blind Signal Separation (IBSS) . . . 5
1.1.3 Instantaneous Semi-Blind Source Localization (ISBSL) . . . 10
1.2 Motivations, contributions and problem statement . . . 16
1.2.1 Main approaches to blind problems . . . 16
1.2.2 Unifying approach for exploiting temporal structure of any kind . . . 17
1.2.3 Unifying method for exploiting temporal structure of any order . . . 17
1.2.4 More sources than sensors: underdetermined blind problems . . . 18
1.2.5 Annihilating the influence of noise with 'simple' structure . . . 20
1.2.6 Subspace methods and arranging sensor statistics . . . 20
1.2.7 Conjugation pattern . . . 21
1.2.8 Thesis problem statement, objectives and contributions . . . 22
1.3 Publication overview . . . 23
1.4 Notation . . . 24
1.5 Thesis outline . . . 25
1.6 Conclusions . . . 27

2 Theory, literature review and problem context **29**
2.1 Overview of development of blind methods . . . 30
2.2 Model assumptions . . . 31
2.2.1 Assumptions on mixing system . . . 31
2.2.2 Assumptions on source signals . . . 32
2.2.3 Assumptions on noise signals . . . 33

2.2.4 Underdetermined blind problems . . . 34
2.3 Temporal versus no temporal structure . . . 37
2.3.1 Mutual statistical independence and non-Gaussianity: ICA . . . 37
2.3.2 Temporal structure . . . 38
2.4 Model indeterminacies . . . 40
2.5 Objective and contrast functions, optimization methods . . . 41
2.6 Whitening the observations . . . 43
2.6.1 Whitening transforms and Principal Component Analysis . . . 44
2.6.2 ICA complexity reduction by whitening . . . 45
2.6.3 Two-stage approach to blind identification and separation . . . 46
2.7 Exploiting spatial independence and non-Gaussianity . . . 47
2.7.1 Non-Gaussianity . . . 47
2.7.2 Information-theoretic measures of statistical independence and non-Gaussianity . . . 49
2.7.3 Signal processing oriented measures of statistical independence and non-Gaussianity . . . 55
2.7.4 Algebraic and iterative algorithms based on cross-cumulants . . . 62
2.8 Exploiting spatial uncorrelatedness and temporal structure . . . 71
2.8.1 Second order temporal structure . . . 71
2.8.2 Work of Molgedey and Schuster . . . 73
2.8.3 AMUSE: an approach based on whitened observations and EVD . . . 74
2.8.4 Extension of AMUSE to several time lags: SOBI . . . 75
2.8.5 The general case: joint approximate diagonalization of several time- and lag-dependent correlation matrices . . . 76
2.8.6 Joint approximate diagonalization methods . . . 77
2.9 Conclusions and discussion . . . 82

3 Subspace methods for DOA identification 83
3.1 Direction Of Arrival observation model . . . 84
3.1.1 Arbitrary sensor positions for 2D-scenario . . . 84
3.1.2 Example: Uniform Linear Array (ULA) . . . 87
3.1.3 Geometrical properties of DOA observation model . . . 88
3.2 Subspace Methods for DOA estimation based on Second Order Statistics . . . 89
3.2.1 Sensor correlation structure, signal and noise subspaces . . . 89
3.2.2 SPECTRAL-MUSIC . . . 97
3.2.3 ROOT-MUSIC . . . 102
3.2.4 The Minimum-Norm subspace method . . . 106
3.3 Subspace Methods for DOA estimation based on Higher Order Statistics . . . 112
3.3.1 Assumptions on source signals, noise signals, and mixing matrix . . . 112
3.3.2 Structure and matrix of fourth order sensor cumulants . . . 114
3.3.3 Eigenstructure of $\mathbf{C}^x$, signal and noise subspaces . . . 118
3.3.4 Relation between generalized array response vectors and subspaces; obtaining a system of equations satisfied by source DOA's . . . 119
3.3.5 Dealing with estimated statistics . . . 121
3.3.6 SPECTRAL-MUSIC with HOS . . . 123
3.4 Retrospective analysis . . . 127
3.5 Conclusions and discussion . . . 130

4 MIBI based on Second Order Temporal Structure: 2×2 Mixing Case **131**
4.1 MIBI model structure . . . 132
4.2 MIBI model assumptions . . . 134
4.3 Formulating MIBI as system of homogeneous polynomial equations . . . 138
4.3.1 Structure of sensor correlation functions . . . 139
4.3.2 Notation for sets of correlation functions and index pairs . . . 140
4.3.3 Determining set of unique sensor correlation functions . . . 142
4.3.4 Derivation of system of bivariate homogeneous polynomial equations of degree two; functional notation . . . 143
4.3.5 Derivation of system of bivariate homogeneous polynomial equations of degree two; row vector notation . . . 147
4.3.6 Derivation of system of bivariate homogeneous polynomial equations of degree two; matrix-vector notation and subspace matrix . . . 150
4.3.7 Indexing schemes and ordering . . . 161
4.3.8 Using the SVD to find proper coefficients of the polynomials . . . 164
4.4 Algebraic and geometric structure . . . 172
4.4.1 Factorization of bivariate homogeneous polynomial of degree two into two linear terms . . . 173
4.4.2 Example 1: Full rank mixing matrix and stationary AR(1) signals . . . 175
4.4.3 Example 2: Rank-deficient mixing matrix and stationary AR(1) signals . . . 180
4.4.4 Example 3: Speech signals mixed by full rank mixing matrix . . . 186
4.4.5 Example 4: Images mixed by full rank mixing matrix . . . 187
4.5 Solving the system of polynomial equations . . . 189
4.5.1 Bivariate homogeneous polynomial cost function . . . 190
4.5.2 Bivariate polynomial cost function with unit-norm constraint . . . 191
4.5.3 Finding all different solutions of a bivariate homogeneous polynomial equation of arbitrary degree . . . 192
4.5.4 Univariate polynomial cost function . . . 193
4.5.5 Solving a bivariate polynomial equation by factorization . . . 198
4.5.6 Solving a system of bivariate polynomial equations: homotopy . . . 199
4.5.7 TIME-MUSIC Pseudo-Spectrum . . . 207
4.6 Writing MIBI as a GEVD problem . . . 210
4.6.1 Application to Example 1 in Section 4.4.2 . . . 212
4.6.2 Application to Example 2 in Section 4.4.3 . . . 212
4.7 Conclusions and discussion . . . 213

5 MIBI based on second order temporal structure: $D \times S$ mixing case **215**
5.1 MIBI model structure . . . 216
5.2 MIBI model assumptions . . . 218
5.2.1 Definitions of correlation functions and regions of support . . . 218
5.2.2 Example for scenario with real-valued system, stationary AR(1) source signals, and additive white noise . . . 223
5.3 Formulating MIBI as system of homogeneous polyconjugal equations . . . 227
5.3.1 Structure of sensor correlation functions . . . 227
5.3.2 Notation for (sets of) index pairs, correlation functions, etcetera . . . 229
5.3.3 Determining set of unique sensor correlation functions . . . 232

5.3.4 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; functional notation . . . 244
5.3.5 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; row vector notation . . . 248
5.3.6 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; matrix-vector notation and subspace matrix . . . 251
5.3.7 Writing the system of equations in matrix-vector notation . . . 260
5.3.8 Using the SVD to find proper coefficients of the polyconjugals . . . 260
5.4 Algebraic and geometric structure . . . 269
5.4.1 Real-valued scenarios with three sensors, stationary AR(1) source signals, and additive white noise . . . 270
5.4.2 Real-valued scenarios with three sensors, speech signals, additive white noise, and estimated subspace matrix . . . 280
5.5 Solving the system of equations by means of homotopy . . . 285
5.6 Writing MIBI as a Multi-Matrix GEVD problem . . . 290
5.7 Conclusions and discussion . . . 294

6 MIBI based on arbitrary order temporal structure: $D \times S$ mixing case 295
6.1 MIBI model assumptions for l-th order statistics . . . 296
6.2 Formulating MIBI as system of homogeneous polyconjugal equations . . . 300
6.2.1 Structure of sensor cumulant functions . . . 300
6.2.2 Notation for (sets of) index tuples, cumulant functions, etcetera . . . 301
6.2.3 Determining set of unique sensor cumulant functions . . . 304
6.2.4 Derivation of system of D-variate l-homogeneous polyconjugal equations; functional notation . . . 317
6.2.5 Derivation of system of D-variate l-homogeneous polyconjugal equations; row vector notation . . . 321
6.2.6 Derivation of system of D-variate l-homogeneous polyconjugal equations; matrix-vector notation and subspace matrix . . . 323
6.2.7 Writing the system of equations in matrix-vector notation . . . 332
6.2.8 Using the SVD to find proper coefficients of the polyconjugals . . . 333
6.3 Algebraic and geometric structure . . . 341
6.3.1 Real-valued scenarios with two sensors, stationary Gamma AR(1) source signals, additive white noise, and third order statistics . . . 342
6.3.2 Real-valued scenarios with two sensors, stationary Gamma AR(1) source signals, additive white noise, and fourth order statistics . . . 348
6.3.3 Generalizations and implications for identifiability . . . 352
6.4 Solving the system of equations . . . 355
6.5 Writing MIBI as a Multi-Matrix GEVD problem . . . 356
6.5.1 MGEVD structure . . . 357
6.5.2 Example with $D = 2$, $S = 3$, $l = 3$, and $\mathbf{c}_3 = (\circ, \circ, \circ)$ . . . 359
6.5.3 Generalized matrix pencil . . . 361
6.6 Maximum number of columns or minimum number of sensors for given conjugation tuple . . . 364
6.6.1 Number of linearly independent equations . . . 364
6.6.2 Derivation of condition for identifiability . . . 364
6.6.3 Determining the maximum number of columns $S^{\mathbf{c}_l}_{\max}(D)$ . . . 365

6.6.4 Determining the minimum number of sensors $D^{\mathbf{c}_l}_{\min}(S)$ 366
6.7 Conclusions and discussion 368

7 Semi-Blind Source Localization **369**
7.1 DOA estimation with TIME-SPECTRAL-MUSIC 370
7.1.1 Structure of system and equations 370
7.1.2 TIME-MUSIC null- and pseudo-spectra 372
7.1.3 Maximum number of source DOA's or minimum number of sensors for given array configuration and conjugation tuple 374
7.1.4 TIME-SPECTRAL-MUSIC examples 376
7.2 DOA estimation with ULA-TIME-SPECTRAL-MUSIC 387
7.2.1 Structure of system and equations 387
7.2.2 ULA TIME-MUSIC null- and pseudo-spectra 388
7.2.3 Maximum number of source DOA's or minimum number of sensors for given ULA and conjugation tuple 389
7.2.4 Examples of ULA-TIME-SPECTRAL-MUSIC 391
7.3 ULA-TIME-ROOT-MUSIC 395
7.4 Conclusions 397

8 Conclusions, trade-offs and future research **399**
8.1 Conclusions 399
8.2 Trade-offs 401
8.3 Future research 403

A Notation and definitions **405**
A.1 Vectors, matrices and vector spaces 405
A.2 Tuples and sets 406
A.3 Definitions and results from combinatorics 409
A.4 Kronecker delta functions 413
A.5 Conjugation patterns and notation by tuples 413
A.6 Signals, cumulant and moment functions, conjugation of arguments 414
A.7 Hadamard product 416
A.8 Kronecker product 416
A.9 Khatri-Rao product 417
A.10 Inner products 418

B Statistical prerequisites **419**
B.1 Characterizations of real-valued random variables 419
B.1.1 Cumulative distribution and probability density functions 419
B.1.2 Characteristic functions 420
B.1.3 Expectations, moments, central moments, and cumulants 420
B.2 Characterizations of real-valued random vectors 429
B.2.1 Joint cumulative distribution and probability density functions 429
B.2.2 Conditional probability density functions 431
B.2.3 Statistical independence 432
B.2.4 The Central Limit Theorem 433
B.2.5 Multivariate characteristic functions 433
B.2.6 Multivariate expectations, (central) moments and cumulants 434

B.3 Complex-valued statistical quantities 446

C Basic linear algebra facts **447**
C.1 Definitions 447
C.2 The four fundamental subspaces of a matrix 448
C.2.1 Definitions 448
C.2.2 Properties 450

D Singular and Eigen Value Decompositions **451**
D.1 SVD in standard matrix notation 451
D.2 SVD in functional notation 454
D.3 Generalized Eigenvalue Decomposition 457

E Homotopy methods for solving nonlinear systems **459**
E.1 Basic principles of homotopy methods 459
E.2 Example for univariate polynomials 466
E.3 Example 1 for bivariate polynomials 469

Scientific publications by the author **473**

Bibliography **475**

Index **489**

Acknowledgements **503**

Curriculum Vitae **505**

List of Figures

1.1 MIBI problem setup. . . . 3
1.2 IBSS problem setup. . . . 5
1.3 Linear instantaneous mixing with two sources and two sensors. . . . 6
1.4 Stationary Gaussian $AR(1)$ source signals (left) and their mixtures (right). . . . 7
1.5 Speech sources (left) and their mixtures (right). . . . 8
1.6 Image sources (left) and their mixtures (right). . . . 9
1.7 Source Localization problem setup. . . . 11
1.8 Determination of array response vector for ULA. . . . 13
1.9 Radio Telescope Array. . . . 15
1.10 Deployments of hydrophone arrays. . . . 15
1.11 Bijective mappings between functions, row vectors, function-valued vectors and matrices. . . . 26

2.1 Two-stage approach to ICA. . . . 47
2.2 Feedback system for Hérault-Jutten algorithm. . . . 59

3.1 Coplanar DOA estimation problem setup. . . . 84
3.2 Ideal and estimated MUSIC spectra $P_{\text{SM}}(\theta)$ for $D = 4$ and $S = 3$. . . . 100
3.3 Ideal and estimated MUSIC spectra $P_{\text{SM}}(\theta)$ for $D = 5$ and $S = 3$. . . . 100
3.4 Ideal ($\diamond$) and estimated ($\times$) roots of the ROOT-MUSIC polynomial $p_{\text{RM}}(z)$ for $D = 4$ and $S = 3$. . . . 109
3.5 Ideal ($\diamond$) and estimated ($\times$) roots of the ROOT-MUSIC polynomial $p_{\text{RM}}(z)$ for $D = 5$ and $S = 3$; left: all roots, right: area around unit circle enlarged. . . . 109
3.6 Ideal and estimated Min-Norm spectra $P_{\text{SMN}}(\theta)$ for $D = 4$ and $S = 3$. . . . 110
3.7 Ideal ($\diamond$) and estimated ($\times$) roots of the Root-Min-Norm polynomial $p_{\text{RMN}}(z)$ for $D = 4$ and $S = 3$. . . . 110
3.8 Ideal and estimated Min-Norm spectra $P_{\text{SMN}}(\theta)$ for $D = 5$ and $S = 3$. . . . 111
3.9 Ideal ($\diamond$) and estimated ($\times$) roots of the Root-Min-Norm polynomial $p_{\text{RMN}}(z)$ for $D = 5$ and $S = 3$. . . . 111
3.10 Ideal and estimated HOS-SPECTRAL-MUSIC spectra $P_{\text{HSM}}(\theta)$ for $D = 3$ and $S = 4$. . . . 126

4.1 2×2 MIBI problem setup. . . . 132
4.2 Regions of support in the domain of lags k. . . . 135
4.3 Source and noise auto-correlation functions on Ω_k^s. . . . 136
4.4 Source and noise auto-correlation functions on $\Omega_k^{s\setminus\nu}$. . . . 137
4.5 Graph of $f(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.1. . . . 177
4.6 Contour plot of $f(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.1. . . . 177

4.7 Stationary Gaussian $AR(1)$ source signals (left) and their noisy mixtures (right) for example with full rank mixing matrix in Section 4.4.2.2. 179
4.8 Contour plot of $\hat{f}(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.2. 179
4.9 Contour plots of $f_1(z_1, z_2)$ (left) and $f_2(z_1, z_2)$ (right) for example with rank-deficient mixing matrix in Section 4.4.3.1. 182
4.10 Graph of $f_{\mathbf{A}}(z_1, z_2)$ for example in Section 4.4.3.1 with rank-deficient mixing matrix. 183
4.11 Contour plot of $f_{\mathbf{A}}(z_1, z_2)$ for example in Section 4.4.3.1 with rank-deficient mixing matrix. 183
4.12 Stationary Gaussian $AR(1)$ source signals (left) and their noisy mixtures (right) for example in Section 4.4.3.2 with rank-deficient mixing matrix. . . . 185
4.13 Contour plots of $\hat{f}_1(z_1, z_2)$ (left) and $\hat{f}_2(z_1, z_2)$ (right) for example in Section 4.4.3 with rank-deficient mixing matrix. 185
4.14 Contour plots of $f(z_1, z_2)$ (left) and $\hat{f}(z_1, z_2)$ (right) for example with speech signals in Section 4.4.4. 186
4.15 Contour plots of $f(z_1, z_2)$ (left) and $\hat{f}(z_1, z_2)$ (right) for example with images in Section 4.4.5. 188
4.16 Zero contour plot of function $f(z_1, z_2)$ occurring in full rank example of Section 4.4.2 together with unit-norm constraint. 189
4.17 Gaussian $AR(1)$ source signals (left) and noise-free estimated source signals (right) for 'full rank example' in Section 4.4.2.2. 195
4.18 Original speech source signals (left) and estimated speech signals (right) for example with speech signals in Section 4.4.4. 197
4.19 Original source images (left) and separated images (right) for example with image signals in Section 4.4.5. 198
4.20 Real and imaginary parts of $z_1(|\lambda|)$-components (solid lines) and $z_2(|\lambda|)$-components (dotted lines) of homotopy solution paths of $\mathbf{h}(\mathbf{z}, \lambda) = \mathbf{0}$ for Example 1 in Section 4.4.2 with ideal subspace matrix. 206
4.21 Real and imaginary parts of $z_1(|\lambda|)$-components (solid lines) and $z_2(|\lambda|)$-components (dotted lines) of homotopy solution paths of $\mathbf{h}(\mathbf{z}, \lambda) = \mathbf{0}$ for Example 1 in Section 4.4.2 with estimated subspace matrix. 206
4.22 Pseudo-spectrum $P(\theta)$ for Example 1 in Section 4.4.2. 209
4.23 Pseudo-spectrum $P(\theta)$ for Example 2 in Section 4.4.3. 209

5.1 MIBI problem setup. 216
5.2 Regions of support in the domain of time index pairs (n_1, n_2). 220
5.3 Correlation functions and regions of support for scenario with four AR(1) source signals and additive white noise. 224
5.4 Correlation functions on Noise-Free ROS for scenario with four AR(1) source signals and additive white noise. 226
5.5 Zero contour level cones of $f^2_{3,1}(z_1, z_2, z_3)$, $f^2_{3,2}(z_1, z_2, z_3)$ and $f^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with ideal subspace matrix. 273
5.6 Zero contour spherical ellipses of $f^2_{3,1}(z_1, z_2, z_3)$, $f^2_{3,2}(z_1, z_2, z_3)$ and $f^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with ideal subspace matrix. 273
5.7 Stationary Gaussian $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 3×3 example. 274

5.8 Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with estimated subspace matrix. . . . 275
5.9 Zero contour level cones of $f^2_{3,1}(z_1, z_2, z_3)$ and $f^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with ideal subspace matrix. 277
5.10 Zero contour spherical ellipses of $f^2_{3,1}(z_1, z_2, z_3)$ and $f^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with ideal subspace matrix. 277
5.11 Stationary Gaussian $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 3×4 example. 278
5.12 Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with estimated subspace matrix. 279
5.13 Two-dimensional auto-correlation function of first source signal used in speech examples. 281
5.14 Two-dimensional auto-correlation function of second source signal used in speech examples. 281
5.15 Speech sources (left) and their noisy mixtures (right) for 3×3 example. . . . 283
5.16 Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,3}(z_1, z_2, z_3)$ for 3×3 speech example. 283
5.17 Speech sources (left) and their noise-contaminated mixtures (right) for 3×4 example. 284
5.18 Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ for 3×4 speech example. 285
5.19 Spherical ellipses and estimated solutions of system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$, $\hat{f}^2_{3,3}(z_1, z_2, z_3)\}$ for 3×3 example with AR(1) signals. . . . 287
5.20 Spherical ellipses and estimated solutions of system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)\}$ for 3×4 example with AR(1) signals. 287
5.21 Spherical ellipses and estimated solutions of system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$, $\hat{f}^2_{3,3}(z_1, z_2, z_3)\}$ for 3×3 example with speech signals. . . . 288
5.22 Spherical ellipses and estimated solutions of system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)\}$ for 3×4 example with speech signals. 288
5.23 Speech sources (left) and noise-free estimated source signals (right) for 3×3 example. 289

6.1 Regions Of Support in the domain of time index tuples $\mathbf{n}_l$. 298
6.2 Gamma probability density function for $\alpha = 2$ and $\lambda = 1$. 344
6.3 Stationary Gamma $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 2×2 example. 345
6.4 Contour plots of $\hat{f}^3_{2,1}(z_1, z_2)$ (left) and $\hat{f}^3_{2,2}(z_1, z_2)$ (right) for $D = 2$, $S = 2$, and $l = 3$. 346
6.5 Stationary Gamma $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 2×3 example. 347
6.6 Contour plot of $\hat{f}^3_{2,1}(z_1, z_2)$ for $D = 2$, $S = 3$, and $l = 3$. 348
6.7 Contour plots of $\hat{f}^4_{2,1}(z_1, z_2)$ (top left), $\hat{f}^4_{2,2}(z_1, z_2)$ (top right) and $\hat{f}^4_{2,3}(z_1, z_2)$ (bottom) for $D = 2$, $S = 2$, and $l = 4$. 350
6.8 Contour plots of $\hat{f}^4_{2,1}(z_1, z_2)$ (left) and $\hat{f}^4_{2,2}(z_1, z_2)$ (right) for $D = 2$, $S = 3$, and $l = 4$. 351

6.9 Contour plot of $\hat{f}^4_{2,1}(z_1, z_2)$ for $D = 2$, $S = 4$, and $l = 4$. 352
6.10 Minimum number of sensors $D^{(\circ)_2}_{\min}(S)$. 367
6.11 Minimum number of sensors $D^{(\circ)_3}_{\min}(S)$. 367
6.12 Minimum number of sensors $D^{(\circ)_4}_{\min}(S)$. 367

7.1 Example 1: Source pole positions for $S = 6$. 377
7.2 Example 1: Normalized sensor positions for $D = 3$. 378
7.3 Example 1: Ideal TIME-SPECTRAL-MUSIC spectrum $P^{\circ *}_3(\theta)$ and its estimate $\hat{P}^{\circ *}_3(\theta)$ for $D = 3$, $S = 6$ and $\mathbf{c}_2 = (\circ, *)$. 378
7.4 Minimum number of sensors ${}_\theta D^{(\circ)_2}_{\min}(S) \equiv {}_\theta D^2_{\min}(S)$. 381
7.5 Minimum number of sensors ${}_\theta D^{(\circ)_3}_{\min}(S) \equiv {}_\theta D^3_{\min}(S)$. 381
7.6 Minimum number of sensors ${}_\theta D^{(\circ)_4}_{\min}(S) \equiv {}_\theta D^4_{\min}(S)$. 381
7.7 Example 2: Normalized array configuration with $D = 10$, 4 and 3 sensors for $l = 1$, 2 and 3 respectively. 382
7.8 Example 2: Ideal TIME-MUSIC spectra $P^{(\circ)_1}_{10}(\theta)$, $P^{(\circ)_2}_4(\theta)$ and $P^{(\circ)_3}_3(\theta)$. . . 383
7.9 Example 2: Ideal TIME-MUSIC spectra $P^{(\circ)_1}_{10}(\theta)$, $P^{(\circ)_2}_4(\theta)$ and $P^{(\circ)_3}_3(\theta)$. . . 383
7.10 Example 3: Pole-zero pairs of 'ALLPASS(1)' source signals for $S = 3$. . . . 385
7.11 Example 3: Probability density functions of w_{r} and w_{r}. 386
7.12 Example 3: Ideal TIME-SPECTRAL-MUSIC spectra $P^{\circ\circ *}_2(\theta)$ and its estimate $\hat{P}^{\circ\circ *}_2(\theta)$ for $D = 2$, $S = 3$ and $\mathbf{c}_3 = (*, \circ, \circ)$. 386
7.13 ULA: Minimum number of sensors ${}_\theta D^{\mathbf{c}_1}_{\min}(S) \triangleq {}_\theta D^1_{\min}(S)$. 391
7.14 ULA: Minimum number of sensors ${}_\theta D^{\mathbf{c}_2}_{\min}(S) \triangleq {}_\theta D^2_{\min}(S)$. 391
7.15 ULA: Minimum number of sensors ${}_\theta D^{\mathbf{c}_3}_{\min}(S) \triangleq {}_\theta D^3_{\min}(S)$. 391
7.16 ULA: Minimum number of sensors ${}_\theta D^{\mathbf{c}_4}_{\min}(S) \triangleq {}_\theta D^4_{\min}(S)$. 391
7.17 ULA Example 1: Ideal ULA-TIME-SPECTRAL-MUSIC spectrum $P^{\circ *}_4(\theta)$ and its estimate $\hat{P}^{\circ *}_4(\theta)$ $D = 4$, $S = 6$ and $\mathbf{c}_2 = (\circ, *)$. 392
7.18 ULA Example 2: Ideal ULA-TIME-MUSIC spectra $P^{(\circ)_1}_{13}(\theta)$, $P^{(\circ)_2}_7(\theta)$, $P^{(\circ)_3}_5(\theta)$ and $P^{(\circ)_4}_4(\theta)$ for $S = 11$. 394
7.19 ULA Example 2: Ideal ULA-TIME-MUSIC spectra $P^{(\circ)_1}_{13}(\theta)$, $P^{(\circ)_2}_7(\theta)$, $P^{(\circ)_3}_5(\theta)$ and $P^{(\circ)_4}_4(\theta)$ for $S = 12$. 394
7.20 Ideal ($\diamond$) and estimated ($\times$) roots of the ULA-TIME-ROOT-MUSIC polynomial $p_{\mathrm{UTRM}}(z)$ for Example 3 of Section 7.1.4.3. 398

A.1 Index set notation. 408

B.1 Platykurtic (uniform), mesokurtic (Gaussian) and leptokurtic (Super-Gaussian) distributions. 424

E.1 Possible continuation parameter paths in the complex plane. 464
E.2 Solution paths of homotopy system $h(z, \lambda) = 0$ with $h(z, \lambda)$ defined in (E.2.4) and $p(z)$ in (E.2.12). 468
E.3 Intersecting zero contour curves of bivariate homogeneous polynomial (solid blue line) and unit-norm constraint function (dashed green line). 470
E.4 Real and imaginary parts of $z_1(\lambda)$-components (solid lines) and $z_2(\lambda)$-components (dotted lines) of solution paths of $\mathbf{h}(\mathbf{z}, \lambda) = \mathbf{0}$ in (E.3.11). 471

List of Tables

5.1 Number $M^{2}_{\mathrm{u},D}$ of unique products and sensor correlation functions. 237
5.2 Number $M^{c_1c_2}_{\mathrm{u},D}$ of unique products and sensor correlation functions. 241

6.1 Number $M^{l}_{\mathrm{u},D}$ of unique products and sensor cumulant functions. 310
6.2 Number $M^{\mathbf{c}_l}_{\mathrm{u},D}$ of unique products and sensor cumulant functions. 314
6.3 Values of $S^{(\circ)_l}_{\mathrm{max}}(D)$ for $Q^{(\circ)_l}_{\mathrm{min},D} = D-1$, $1 \leq l \leq 4$, and $1 \leq D \leq 5$. 366

7.1 ${}_{\theta}S^{(\circ)_l}_{\mathrm{max},D}$ values for $1 \leq l, D \leq 4$. 380
7.2 Uniform Linear Array: ${}_{\theta}S^{l}_{\mathrm{max},D} = l(D-1)$ for $1 \leq l \leq 4$, $1 \leq D \leq 14$. . . . 390
7.3 Uniform Linear Array: ${}_{\theta}D^{\mathbf{c}_l}_{\mathrm{min}}(S) = 1 + \lceil S/l \rceil$ for $1 \leq l \leq 4$, $1 \leq S \leq 14$. . . 390

A.1 Multiset coefficients $M_{p,G} = \left\langle {G \atop p} \right\rangle = \binom{p+G-1}{p} = \binom{p+G-1}{G-1}$. 411

B.1 Example of partitioning $\mathcal{I} = \{1,2,3\}$. 426
B.2 Example of partitioning $\mathcal{I} = \{1,2,3,4\}$. 428
B.3 Example of joint moment definition for $G = 3$ and $l = 2$. 436
B.4 Partitioning of $\mathcal{I} = \{1,2\}$. 441

List of Algorithms

2.1 IBSS based on minimizing mutual information. 54
2.2 Hérault-Jutten algorithm for IBSS based on zeroing nonlinear de-correlation. 60
2.3 Cichocki-Unbehauen algorithm for IBSS based on zeroing nonlinear de-correlation. 61
2.4 Gradient ascent algorithm for IBSS based on kurtosis. 62
2.5 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ based on EVD of cumulant matrix of whitened data. 67
2.6 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ based on JADE. 68
2.7 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ based on optimizing objective functions incorporating cumulant criteria. 69
2.8 Blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on the GEVD of two cumulant matrices. 70
2.9 Molgedey and Schuster algorithm for blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on GEVD of two correlation matrices. 74
2.10 AMUSE algorithm for blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ based on temporal correlation structure. 75
2.11 SOBI algorithm for blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ based on JAD of correlation matrices. 75
2.12 Blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on JAD of correlation matrices. 76
3.1 SPECTRAL-MUSIC. 98
3.2 ROOT-MUSIC. 104
3.3 Spectral-Min-Norm. 108
3.4 Root-Min-Norm. 108
3.5 HOS-SPECTRAL-MUSIC. 124
3.6 Key ingredients of a general subspace algorithm. 129
4.1 Pre-overview of the 2×2 MIBI method developed in this chapter. 132
4.2 Overview of results obtained so far. 161
4.3 High-level algorithm for subspace based real-valued 2×2 MIBI. 171
4.4 Homotopy continuation method employed in thesis. 202
4.5 Overview of our 2×2 MIBI method developed in this chapter. 214
5.1 Pre-overview of the $D \times S$ MIBI method developed in this chapter. 216
5.2 High-level algorithm for $D \times S$ MIBI exploiting Second Order Temporal Structure with arbitrary conjugation pairs (c_1, c_2). 268
6.1 Pre-overview of the $D \times S$ MIBI method based on l-th order statistics and conjugation tuple $\mathbf{c}_l$. 296
6.2 High-level algorithm for $D \times S$ MIBI exploiting l-th order temporal structure with arbitrary conjugation tuple $\mathbf{c}_l$. 341
7.1 Arbitrary order TIME-SPECTRAL-MUSIC algorithm for $D \times S$ DOA estimation scenario with arbitrary conjugation tuple $\mathbf{c}_l$. 374
7.2 ULA-TIME-ROOT-MUSIC. 397

E.1 Overview of homotopy continuation method for solving a nonlinear system of equations by an increment-and-fix method with constant step size. 463
E.2 Overview of homotopy continuation method with ‘generalized gamma trick’ and ’generalized continuation parameter’. 465
E.3 Homotopy continuation method for finding all roots of the univariate polynomial $p(z) = \sum_{p=0}^{P} a_p(z)^p$ by an increment-and-fix method with constant step size. 467

Glossary

List of Acronyms

AMUSE	Algorithm for Multiple Unknown Signals Extraction
ASP	Array Signal Processing
BI	Blind Identification
BSE	Blind Signal Extraction
BSP	Blind Signal Processing
BSS	Blind Signal Separation
CBSS	Convolutive Blind Signal Separation
cdf	cumulative distribution function
CDMA	Code Division Multiple Access
CGF	Cumulant Generating Function
CLT	Central Limit Theorem
cpdf	conditional probability density function
DOA	Direction Of Arrival
EFOBI	Extended Fourth Order Blind Identification
EVD	Eigenvalue Decomposition
FCF	First Characteristic Function
FJCF	First Joint Characteristic Function
FOBI	Fourth Order Blind Identification
FOSS	Fourth Order Signal Subspace
GEVD	Generalized Eigenvalue Decomposition
HOS	Higher Order Statistics
HOTS	Higher Order Temporal Structure
IBI	Instantaneous Blind Identification
IBSS	Instantaneous Blind Signal Separation
ICA	Independent Component Analysis
ISBSL	Instantaneous Semi-Blind Source Localization
JAD	Joint Approximate Diagonalization
JADE	Joint Approximate Diagonalization of Eigenmatrices
JAND	Joint Approximate Non-Orthogonal Diagonalization
JAOD	Joint Approximate Orthogonal Diagonalization
jcdf	joint cumulative distribution function
jpdf	joint probability density function
MCBI	Multiple-Input Multiple-Output Convolutive Blind Identification
MGF	Moment Generating Function
MIBI	Multiple-Input Multiple-Output Instantaneous Blind Identification
MIMO	Multiple-Input Multiple-Output
MIN-NORM	Minimum-Norm
mjpdf	marginal joint probability density function
MMGEVD	Multi-Matrix Generalized Eigenvalue Decomposition

mpdf	marginal probability density function
MUSIC	Multiple Signal Classification
PCA	Principal Component Analysis
pdf	probability density function
PHD	Pisarenko Harmonic Decomposition
ROOT-MUSIC	Root Multiple Signal Classification
ROS	Region Of Support
SCF	Second Characteristic Function
SJCF	Second Joint Characteristic Function
SKAI	Square Kilometer Array Interferometer
SOBI	Second Order Blind Identification
SOS	Second Order Statistics
SOTS	Second Order Temporal Structure
TRM	TIME-ROOT-MUSIC
TSM	TIME-SPECTRAL-MUSIC
UTRM	ULA-TIME-ROOT-MUSIC
UTSM	ULA-TIME-SPECTRAL-MUSIC
SVD	Singular Value Decomposition
ULA	Uniform Linear Array
VLA	Very Large Array

List of Abbreviations used in Lists

AS	ASsumptions on MIBI scenarios (pages 90, 114, 138, 222, 299)
ASM	ASsumptions on Mixing system (page 31)
ASN	ASsumptions on Noise Signals (page 33)
ASS	ASsumptions on Source Signals (page 32)
CFP	Contrast Function Property/Requirement (page 42)
DPS	Desired Properties of Pseudo-Spectrum (page 99)
JP	Desired Properties of objective/cost function $J(\mathbf{y})$ (page 41)
MAS	Main ASsumption of MIBI problem formulations in thesis (page 23)
MNC	Min-Norm algorithm Constraints (page 107)
MSA	Main Subspace Assumption (page 127)
PIF	Possible InFluences on pseudo-spectrum (page 101)
PR	Instantaneous Blind Identification PRinciples (page 16)
ST	Two-STage approach to instantaneous blind problems (page 46)
TPS	Thesis Problem Statement (page 22)

List of Textual Abbreviations

e.g.	exempli gratia: for example
i.e.	id est: that means, in other words
i.i.d.	independently and identically distributed
s.t.	subject to
viz.	videlicet: namely
w.r.t.	with respect to

List of Mathematical Symbols and Variables

a_i^j	Transfer coefficient from j-th source to i-th sensor, i.e. element in the i-th row and j-th column of the mixing matrix
$\mathbf{A}$	Mixing or array response matrix
$\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$	Array response matrix parameterized by DOA θ
$\mathbf{A}\big(\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\zeta}}\big)$	Array response matrix parameterized by azimuth θ and elevation ζ
$\mathbf{A}\big(\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\zeta}}, \tilde{\boldsymbol{\rho}}\big)$	Array response matrix parameterized by azimuth θ, elevation ζ and range ρ
$\tilde{\mathbf{a}}_i$	i-th row of $\mathbf{A}$
$\mathbf{a}^j$	j-th column or array response vector of $\mathbf{A}$
$\mathbf{a}(\theta)$	Array response vector parameterized by DOA θ
$\mathbf{a}\big(\theta, \zeta\big)$	Array response vector parameterized by azimuth θ and elevation ζ
$\mathbf{a}\big(\theta, \zeta, \rho\big)$	Array response vector parameterized by azimuth θ, elevation ζ and range ρ
c	Signal propagation velocity
$\mathbf{c}_p$	Conjugation tuple $\mathbf{c}_p \triangleq (c_1, \ldots, c_p)$ of length p, where each of the symbols $c_1, \ldots, c_p$ can either be '$*$' or '$\circ$', meaning 'conjugation' and 'no conjugation' respectively
$\bar{\mathbf{c}}_p$	Complement of conjugation tuple $\mathbf{c}_p \triangleq (\bar{c}_1, \ldots, \bar{c}_p)$, where $\bar{*} = \circ$ and $\bar{\circ} = *$
$\mathbb{C}$	Set of complex-valued numbers
$\mathbb{C}[\Omega]$	Space of complex-valued functions defined on Ω
$\mathbb{C}^N$	Vector space of complex row vectors of length N
$\mathbb{C}_M$	Vector space of complex column vectors of length M
$\mathbb{C}_M^N$	Vector space of complex matrices of size M by N
$\mathbb{C}_M[\Omega]$	Space of length-M complex function-valued column vectors defined on Ω
d	Sensor spacing of ULA
d_{n}	Sensor spacing of ULA that is normalized w.r.t. the carrier wavelength
$d(\mathbf{V})$	Rank of $\mathbf{V}$
D	Number of sensors
$\mathcal{D}_{\mathbf{V}}$	Ordered set of diagonal elements of $\mathbf{V}$
$\mathbf{e}^i$	i-th standard basis vector of $\mathbb{R}_M$
$\tilde{\mathbf{e}}_j$	j-th standard basis vector of $\mathbb{R}^N$
f_{c}	Frequency of a carrier
$\odot$	Hadamard product
$\mathbf{i}_p$	Length-p tuple $(i_1, \ldots, i_p)$ with indices $i_1, \ldots, i_p$
$i_{m,q}$	m-th index of a numbered length-p index tuple $\mathbf{i}_{p,q}$ ($1 \leq m \leq p$)
$\mathcal{K}_{\text{t},D}^{x,\mathbf{c}_l}$	Set of all sensor cumulant functions with conjugation tuple $\mathbf{c}_l$
$M_{\text{t},D}^{x,\mathbf{c}_l}$	Cardinality of $\mathcal{K}_{\text{t},D}^{x,\mathbf{c}_l}$
$\mathcal{K}_{\text{u},D}^{x,\mathbf{c}_l}$	Set of all unique sensor cumulant functions with conjugation tuple $\mathbf{c}_l$
$M_{\text{u},D}^{x,\mathbf{c}_l}$	Cardinality of $\mathcal{K}_{\text{u},D}^{x,\mathbf{c}_l}$
$\mathcal{M}_{\text{u},D}^{(\circ)_l}\langle\mathcal{Z}\rangle$	Set of all l-multisets of $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$
$\mathcal{M}_{\text{u},D}^{(\circ)_l}(\mathcal{Z})$	Set of unique l-multisets of $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$

$\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}\langle\mathcal{Z}\rangle$	Set of all l-multisets of $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$ with $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z} = \{z_1, \ldots, z_D\}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^* = \{(z_1)^*, \ldots, (z_D)^*\}$
$\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$	Set of unique l-multisets of $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$ with $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z} = \{z_1, \ldots, z_D\}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^* = \{(z_1)^*, \ldots, (z_D)^*\}$
N	Cardinality of a Region Of Support containing time, lag, or time-lag tuples
$\#\{\cdot\}$	Operator that yields the number of elements in the argument set
$\diamond$	Khatri-Rao product
$\otimes$	Kronecker product
$\mathcal{L}(\cdot)$	Linear span of a set of vectors in a vector space
$\mathcal{L}_{\mathrm{r}}(\cdot)$	Linear span of the conjugated functions in the argument of a function-valued column vector
λ_{c}	Wavelength of a carrier
ω_{c}	Angular frequency of a carrier
Ω_k^s	ROS in the domain of lag indices on which source correlation functions exist
Ω_k^ν	ROS in the domain of lag indices on which noise correlation functions exist
$\Omega_k^{s\setminus\nu}$	Noise-Fee ROS in the domain of lag indices
$\Omega_{n;k}^{s,c_1c_2}$	ROS in the domain of time-lag index pairs $(n;k)$ on which source correlation functions with conjugation pair (c_1, c_2) exist
$\Omega_{n;k}^{\nu,c_1c_2}$	ROS in the domain of time-lag index pairs $(n;k)$ on which noise correlation functions with conjugation pair (c_1, c_2) exist
$\Omega_{n;k}^{s\setminus\nu,c_1c_2}$	Noise-Fee ROS in the domain of time-lag index pairs $(n;k)$
$\Omega_{\mathbf{n}_l}^{s,\mathbf{c}_l}$	ROS in the domain of time index tuples $\mathbf{n}_l$ on which source cumulant functions with conjugation tuple $\mathbf{c}_l$ exist
$\Omega_{\mathbf{n}_l}^{\nu,\mathbf{c}_l}$	ROS in the domain of time index tuples $\mathbf{n}_l$ on which noise cumulant functions with conjugation pair $\mathbf{c}_l$ exist
$\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}$	Noise-Fee ROS in the domain of time index tuples $\mathbf{n}_l$
$\tilde{\mathbf{p}}_i$	Position of the i-th sensor of a sensor array
$\tilde{\mathbf{p}}_{i,\mathrm{n}}$	Position of the i-th sensor of a sensor array that is normalized w.r.t. the carrier wavelength
$p_{\,\mathrm{RM}}(z)$	ROOT-MUSIC polynomial
$p_{\,\mathrm{RMN}}(z)$	ROOT-Min-Norm polynomial
$P_{\mathrm{HSM}}(\theta)$	HOS-SPECTRAL-MUSIC pseudo-spectrum
$P_{\mathrm{SM}}(\theta)$	SPECTRAL-MUSIC pseudo-spectrum
$P_{\mathrm{SMN}}(\theta)$	Spectral-Min-Norm pseudo-spectrum
r_i^j	Ratio $r_i^j \triangleq z_j/z_i$ of z_j to z_i
$\mathbb{R}$	Set of real-valued numbers
$\mathbb{R}[\Omega]$	Space of real-valued functions defined on Ω
$\mathbb{R}^N$	Vector space of real-valued row vectors of length N
$\mathbb{R}_M$	Vector space of real-valued column vectors of length M
$\mathbb{R}_M^N$	Vector space of real-valued matrices of size M by N
$\mathbb{R}_M[\Omega]$	Space of length-M real function-valued column vectors defined on Ω
$\mathcal{R}(\cdot)$	Range of a matrix or linear operator/mapping

ρ	Range of a source in a polar, cylindrical or spherical coordinate system
$\tilde{\boldsymbol{\rho}}$	Row vector of the ranges of all sources incident on an array
S	Number of sources
θ	Direction Of Arrival (DOA) or azimuthal angle
$\tilde{\boldsymbol{\theta}}$	Row vector of the DOA's of all sources incident on an array
Θ	Parameter space of interest for DOA θ
$\mathbf{u}(\theta)$	Unit vector pointing in DOA θ
$\mathbf{v}$	Notation for a (general) column vector
v_i	i-th element of the column vector $\mathbf{v}$
$\mathbf{v}_M$	Notation for a column vector $\mathbf{v}$ with explicitly denoted length M
$\tilde{\mathbf{v}}$	Notation for a (general) row vector
v^j	j-th element of the row vector $\tilde{\mathbf{v}}$
$\tilde{\mathbf{v}}^N$	Notation for a row vecto $\tilde{\mathbf{v}}$ with explicitly denoted length N
$\mathbf{V}$	Notation for a (general) matrix
v_i^j	Element in the i-th row and j-th column of the matrix $\mathbf{V}$
$\mathbf{V}_M^N$	Notation for $\mathbf{V}$ with explicitly denoted size $M \times N$
$\mathbf{V}^T$	Transpose of $\mathbf{V}$
$\mathbf{V}^H$	Hermitian of $\mathbf{V}$
$\mathbf{V}^{-1}$	Inverse of the square matrix $\mathbf{V}$
$\mathbf{V}^\dagger$	Pseudo-inverse (Moore-Penrose inverse) of $\mathbf{V}$
$\|\mathbf{V}\|_{\mathrm{F}}$	Frobenius norm of $\mathbf{V}$
$\tilde{\mathbf{v}}_i$	i-th row of $\mathbf{V}$
$\mathbf{v}^j$	j-th column of $\mathbf{V}$
$\mathcal{V}$	Notation for a (general) set
$v := w$	w is assigned to v
$v \doteqdot w$	w is isomorphic to v
$\mathbf{V}_{\odot}^{p}$	p-th order Hadamard product of $\mathbf{V}$
$\mathbf{V}_{\odot}^{\mathbf{c}_p}$	p-th order Hadamard product of $\mathbf{V}$ with conjugation tuple $\mathbf{c}_p$
$\mathbf{V}_{\diamond}^{p}$	p-th order Khatri-Rao product of $\mathbf{V}$
$\mathbf{V}_{\diamond}^{\mathbf{c}_p}$	p-th order Khatri-Rao product of $\mathbf{V}$ with conjugation tuple $\mathbf{c}_p$
$\mathbf{V}_{\otimes}^{p}$	p-th order Kronecker product of $\mathbf{V}$
$\mathbf{V}_{\otimes}^{\mathbf{c}_p}$	p-th order Kronecker product of $\mathbf{V}$ with conjugation tuple $\mathbf{c}_p$
$\mathbf{V} \fallingdotseq \mathcal{N}_{\mathrm{l}}(\mathbf{W})$	Rows of $\mathbf{V}$ span the left null space of $\mathbf{W}$
$\mathbf{0}_M^N$	Zero matrix of size $M \times N$
$\mathbf{0}_M$	Zero column vector of length M
$\tilde{\mathbf{0}}^N$	Zero row vector of length N
ζ	Elevation angle
$\tilde{\boldsymbol{\zeta}}$	Row vector of the elevation angles of all sources incident on an array

List of Mathematical Operator Names

$\mathrm{cum}(\cdot, \cdots, \cdot)$	Cumulant function
$\mathrm{diag}(\mathbf{V})$	Ordered set constructed from the diagonal elements of $\mathbf{V}$
$\mathrm{diag}(\mathcal{D})$	Diagonal matrix constructed from the ordered set $\mathcal{D}$
$\mathrm{ddiag}(\mathbf{V})$	Diagonal matrix constructed from the diagonal elements of $\mathbf{V}$
$\mathrm{mom}(\cdot, \cdots, \cdot)$	Moment function
$\mathrm{off}(\mathbf{V})$	Sum of squared absolute values of the off-diagonal elements of $\mathbf{V}$
$\mathrm{offd}(\mathbf{V})$	Matrix $\mathbf{V}$ with zeroed diagonal elements
$\mathrm{rank}(\mathbf{V})$	Rank of $\mathbf{V}$
$\mathrm{tr}(\mathbf{V})$	Trace of $\mathbf{V}$

CHAPTER 1

Introduction

In this chapter, we introduce, contextualize and motivate the main topics considered in this thesis. These topics concern the so-called *Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI)*, the *Instantaneous Blind Signal Separation (IBSS)*, and the *Instantaneous Semi-Blind Source Localization (ISBSL)* problems; see Figures 1.1, 1.2 and 1.7 respectively. MIBI, IBSS and ISBSL all concern systems where multiple source signals propagate simultaneously to multiple sensors via a certain physical transfer mechanism or mixing system. Such systems are commonly called *Multiple-Input Multiple-Output (MIMO)* systems. The kinds of topics mentioned above typically are investigated in a research area called *Array Signal Processing (ASP)* [62, 84, 147, 179, 184]. In general, ASP can be defined as the part of signal processing that deals with the extraction of information from signals collected by an array of sensors, where each sensor observes a different mixture of the source signals. In this work, we will only consider passive sensor arrays, i.e. arrays that only receive signals as opposed to active arrays that also transmit signals. The type of sensor that is employed in a specific application depends on the physical characteristics of the source(s) and propagation medium [84, 119, 179]. For example, microphone arrays are used for measuring acoustic signals in air, e.g. speech, whereas hydrophone arrays are used for measuring acoustic signals in water, e.g. submarine engine/noise signals. Similarly, phased-array radar systems or radio antenna arrays measure electromagnetic signals that can be used for communication purposes or localization of sources or targets. Likewise, X-ray, radio and optical telescope arrays measure electromagnetic radiation from our universe for astronomical investigation [3, 66, 67, 136, 142, 160]. The significance of the ASP research area is evident from the wide variety of practical applications.

The information of interest can be either the transfer system from the sources to the sensors, the (waveforms of the) source signals, or the number and location of the sources. For example, in radio and speech communication applications the main interest usually is in the particular waveforms of the signals, whereas in submarine and defense applications the main interest usually is in the location of the sources or emitters. Also, for several applications, both the signals themselves, as well as the locations of the sources are of interest, e.g. pointing a video camera to a speaking person, determining the types and locations of submarines, etc. For MIBI the main interest is in the transfer system from the sources to the sensors, for IBSS in the source signals, and for ISBSL in the positions of the sources. Depending on the amount of information that is known a priori, a certain problem can be characterized as *blind*, *semi-blind*, or *non-blind*, if little, medium, or a lot of information is available respectively. The less information is available, i.e. the more blind a problem is, the more strict the assumptions on the 'remaining information' need to be, and vice versa. For example, we call MIBI and IBSS 'blind' because both the source signals and the transfer system are unknown; see also [47, 75, 80] and the references therein. This considerable lack of information has to be compensated for by strong assumptions on the statistics[‡] of the signals and the form of the

[‡] See Appendix B for a review of the statistical concepts underlying the work in this thesis.

transfer system. Specifically, the results in this thesis are mainly based on the common *Blind Signal Processing (BSP)* assumptions that the source signals are mutually statistically independent and that the transfer or mixing system is instantaneous and linear. Many researchers have contributed to the development of, and current interest in, BSP. Some of them are: Jutten and Hérault [55,91], Cardoso [26,27,31,33–35], Comon [36,52], Bell and Sejnowski [11,12], Hyvärinen [79, 80, 83], and Cichocki and Amari [6, 7, 47].

The fact that many blind or semi-blind problems with multiple sources can be tackled by processing multiple sensor signals is due to the different kinds of *diversity* assumed to be present in the system. On the one hand, the mixing system has to possess a certain kind of *spatial diversity*, i.e. each of the sensors should observe a mixture of the source signals that is sufficiently different from the others. In practice, this is usually achieved by properly positioning the sensors. On the other hand, the source signals are usually required to possess a certain kind of *statistical diversity* regarding either the probability distribution(s) of, or the temporal relation(s) between, the signals samples, e.g. blind identification methods may require that the source signals have different kurtoses. In this work, we are particularly interested in exploiting another type of statistical diversity, namely any kind of temporal structure in the source signals of any chosen order, such as arbitrary order non-whiteness/correlatedness and non-stationarity. For instance, when Second Order Statistics (SOS) are employed [14, 41, 43, 45, 46, 48], the required *temporal diversity* means that the source signals have auto-correlation functions that are sufficiently different from each other. *Our main focus in this thesis will be on the development of a unified method for exploiting the temporal structure in the data of any kind and any order for solving the MIBI, IBSS, and ISBSL problems.* We do not consider the conventional (fixed and/or adaptive) beamforming approach to array signal processing that is mainly concerned with the design of a beampattern, i.e. the (power) response to a monochromatic plane wave with a certain frequency and arriving from a certain direction. This topic, which in some sense is complementary to our work, is treated in several excellent works, e.g. see [57, 84, 90, 97, 180].

It is widely recognized that many possible applications exist for blind identification, blind separation and source localization problems. Since these three topics are strongly related, their application areas overlap. In fact, we will show in the coming sections that both IBSS and ISBSL are applications or examples of the more general MIBI problem. Therefore, only examples of IBSS and ISBSL will be given in Sections 1.1.2 and 1.1.3 respectively. Several examples of IBSS can be found in the field of biomedical engineering, where the purpose of various applications is to reveal independent sources in different kinds of biological signals like EEG's and ECG's. Other examples can be found in the separation of speech signals from competing speakers in the so-called 'cocktail party' problem, images, data communication signals, etcetera. Examples of ISBSL, which essentially is a semi-blind MIBI problem with a known parameterization of the mixing system, can be found in many sensor array systems. Typical examples include high-resolution *Direction Of Arrival (DOA)* and range estimation in sonar and phased-array radar systems.

The outline of this chapter is as follows. In Section 1.1, the main problems considered in this thesis, i.e. MIBI, IBSS and ISBSL, are described and some of their application areas are mentioned. Next, in Section 1.2 the motivations, objectives, and main contributions of the thesis are formulated. In Section 1.3 we give an overview of the contents of our publications. Then, in Section 1.4 we discuss some notational issues. Subsequently, in Section 1.5 an outline of the thesis is given. Finally, conclusions are drawn in Section 1.6.

1.1 Main thesis topics and applications

The next sections describe each of the three main thesis topics and their possible applications in turn. The structure and main assumptions of the mathematical models associated with these problems are presented. Without loss of generality, everywhere in the thesis we will assume that all signals have zero mean. In addition, we assume that all involved signals are sampled at discrete time instants. We will emphasize the relationships and differences between the various topics. In order to clearly distinguish the objectives associated with the different problems, and for future reference, a problem statement is formulated for each of them. A general thesis problem statement will be formulated in Section 1.2.8.

1.1.1 Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI)

One of the main problems considered in this thesis is the so-called *Multiple-Input Multiple-Output (MIMO) Instantaneous Blind Identification (IBI)* problem. We will concisely denote this problem by the acronym *MIBI*. The MIBI problem setup is shown in Fig. 1.1. A set $\{s_1[n], \ldots, s_S[n]\}$ of S mutually statistically independent source signals is mixed by a *Multiple-Input Multiple-Output (MIMO)* linear instantaneous (i.e. memory-less) mixing system $\mathbf{A}$, and only a set $\{x_1[n], \ldots, x_D[n]\}$ of D mixed sensor/observation signals corrupted by a set of D additive noise signals $\{\nu_1[n], \ldots, \nu_D[n]\}$ is available. Hence, both the mixing system and the original sources are unknown. For this reason, the MIBI problem is called 'blind' [47,80]. The main conditions and purpose of MIBI can now be formulated as follows.

MIBI problem statement:

The purpose of MIBI is to identify the MIMO instantaneous mixing system from the observed mixtures of the source signals only. The main assumptions are that the source signals are mutually statistically independent and that they possess sufficient statistical diversity.

As we will see in the next chapter, the type of statistical diversity that is assumed is algorithm-specific. For example, one algorithm may exploit the non-Gaussianity of the source signals, whereas another may exploit their temporal auto-correlation structure (see Chapter 2). Depending on the considered application, the signals and mixing system can be real- or complex-valued. In this thesis, we will always denote the number of sources by S and the number of sensors by D. Mathematically, the MIBI observation model can be stated as follows:

$$\mathbf{x}[n] = \sum_{j=1}^{S} \mathbf{a}^j \, s_j[n] + \boldsymbol{\nu}[n] = \mathbf{A}\,\mathbf{s}[n] + \boldsymbol{\nu}[n] \qquad \forall\, n \in \mathbb{Z}\,, \tag{1.1.1}$$

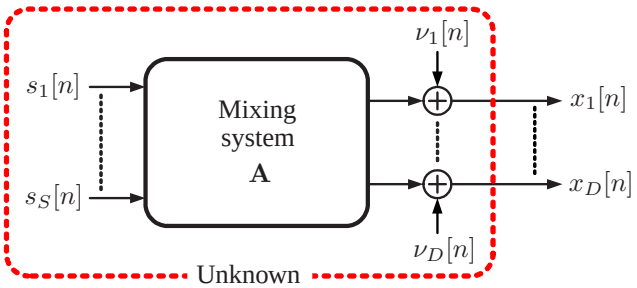

Figure 1.1: *MIBI problem setup.*

where:

$$\mathbf{x}[n] \triangleq \begin{bmatrix} x_1[n] \\ \vdots \\ x_D[n] \end{bmatrix}, \quad \mathbf{s}[n] \triangleq \begin{bmatrix} s_1[n] \\ \vdots \\ s_S[n] \end{bmatrix}, \quad \boldsymbol{\nu}[n] \triangleq \begin{bmatrix} \nu_1[n] \\ \vdots \\ \nu_D[n] \end{bmatrix}, \quad \text{and} \quad \mathbf{a}^j \triangleq \begin{bmatrix} a_1^j \\ \vdots \\ a_D^j \end{bmatrix}$$

are column vectors of sensor signals, source signals, additive noise signals and mixing elements, respectively. Subscript indices are used to index the components of a column vector, whereas superscript indices are used to index the components of a row vector[‡]. Furthermore, the symbol n denotes discrete time. The vectors $\mathbf{x}[n]$, $\boldsymbol{\nu}[n]$, and $\mathbf{a}^1, \ldots, \mathbf{a}^S$ are elements of $\mathbb{R}_D\big(\mathbb{C}_D\big)$, i.e. the vector space of real (complex) column vectors of length D[‡]. The length-S source signal vector $\mathbf{s}[n]$ is an element of $\mathbb{R}_S\big(\mathbb{C}_S\big)$. The coefficient a_i^j denotes the instantaneous transfer from the j-th source to the i-th sensor. As model (1.1.1) shows, the whole unknown transfer system is modelled by a mixing matrix $\mathbf{A}$ that is an element of $\mathbb{R}_D^S\big(\mathbb{C}_D^S\big)$, i.e. the vector space of matrices containing real- or complex-valued elements with D rows and S columns[‡]. For convenience, $\mathbf{A}$ is often written in terms of its columns as follows $\mathbf{A} = \big[\mathbf{a}^1 \cdots \mathbf{a}^S\big]$; see also (2.2.2) on page 31 for other expressions. As is shown in Section 2.4, two ambiguities are inherent to the MIBI model. Namely, the norms and the order of the columns of the mixing matrix cannot be resolved. In general, these two *indeterminacies* do not cause serious problems because for many applications the most relevant information is contained in the 'directions' of the columns rather than in their magnitudes or order.

In order to be able to identify the mixing system and/or recover the source signals, the apparently large lack of information needs to be compensated for by means of a strong hypothesis on the source signals. This hypothesis is the so-called *statistical independence*[§] *assumption* on the source signals [34, 80]. Although it is strong, it is physically plausible in many practical situations because the source signals are often generated independently of each other by different sources. Usually, also some other assumptions that characterize the statistical properties of the individual source signals are required. Typical examples can be found in Section 2.2.2. In addition to assumptions on the source signals, assumptions on the mixing system and noise signals are required. Each sensor should receive a combination of the source signals that is sufficiently different from the others. In other words, there should be sufficient spatial diversity in the mixing system. In practice, this is often achieved by positioning the sensors properly. Assumptions on the diversity of the mixing system often are formulated in terms of the size and rank of the mixing matrix. Typical examples can be found in Section 2.2.1. Finally, in order to be able to properly cope with additive observation or sensor noise, certain assumptions about the statistical structure of the noise signals have to be made. See Section 2.2.3 for characteristic examples.

As we have stated earlier and will explain in the next two sections, *MIBI is a kind of abstraction or generalization of both IBSS and ISBSL*. This implies that applications of IBSS and ISBSL are applications of MIBI as well. Therefore, the IBSS and ISBSL examples presented in the next sections can be considered as examples of MIBI. Finally, we note that many practical problems can be described more adequately by more complex mixing models such as convolutive or nonlinear models. However, apart from being applicable to many practical problems that do (approximately) satisfy the model described above, MIBI can also be seen and used as a stepping stone for more complicated identification problems.

[‡] See Appendix A for our notational conventions.

[§] See Section B.2.3 for the definition of statistical independence.

1.1.2 Instantaneous Blind Signal Separation (IBSS)

A problem closely related to MIBI is *Instantaneous Blind Signal Separation (IBSS)*, which deals with the separation/recovery of mutually statistically independent source signals from their observed instantaneous mixtures only. The problem setup is depicted in Fig. 1.2 for S sources and D sensors. The observation model and assumptions on the signals and mixing system are the same as those of MIBI. However, while the main purpose of MIBI is the estimation of the mixing system (only), the main purpose of IBSS is the estimation of the (waveforms of the) source signals [47,75,80]. Summarizing, the main conditions and purpose of IBSS can be formulated as follows.

> ***IBSS problem statement:***
> The purpose of IBSS is to recover the source signals from their observed mixtures only. The main assumptions are that the source signals are mutually statistically independent and that they possess sufficient statistical diversity.

As with MIBI, the type of statistical diversity that is assumed is algorithm-specific. Mathematically, the IBSS problem can be formulated as the estimation of a separation/de-mixing matrix $\mathbf{W}$ in such a way that the output vector $\mathbf{y}[n]$ defined by:

$$\mathbf{y}[n] \triangleq \mathbf{W}\mathbf{x}[n] \tag{1.1.2}$$

contains estimates of the waveforms of the source signals $s_1[n], \ldots, s_S[n]$. The IBSS problem is typically tackled by designing a de-mixing system $\mathbf{W}$ in such a way that the (statistical) properties of the source signals are restored at the output. For instance, the output signals $y_1[n], \ldots, y_S[n]$ should be statistically independent. Since the data observation model is the same as that of the MIBI model (1.1.1), two IBSS indeterminacies are present that correspond to the MIBI indeterminacies, viz. the scaling and order of the original source signals cannot be recovered. Taking these into account, the goal of IBSS is to recover the source signals in some arbitrary order and with some arbitrary nonsingular scaling. Hence, the de-mixing matrix $\mathbf{W}$ should satisfy $\mathbf{WA} \approx \mathbf{PD}$, where $\mathbf{P}$ is some permutation matrix and $\mathbf{D}$ is some nonsingular diagonal matrix. See Section 2.4 for more information about this issue.

As stated above, contrary to MIBI the main interest in IBSS is in the source signals instead of the mixing system. In fact, once MIBI has been performed, the source signals can be recovered (approximately) by applying the (pseudo-)inverse of the estimated mixing system (provided it is nonsingular) to the observed mixtures. Hence, from this point of view IBSS is merely a direct application of MIBI. Therefore, the main focus in this work will be on MIBI, while IBSS will be considered as an application or example.

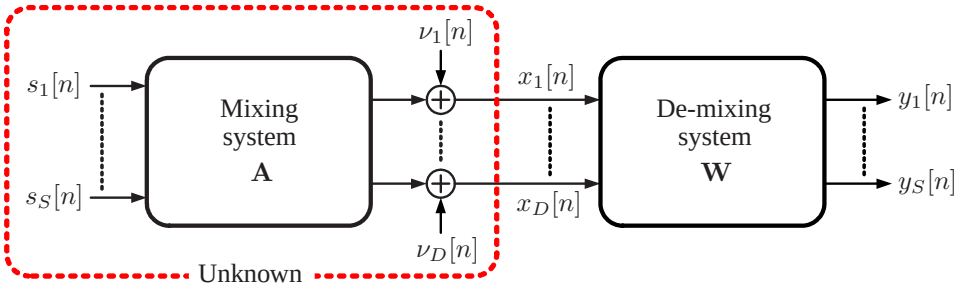

Figure 1.2: *IBSS problem setup.*

Examples

IBSS can be used in several real-world applications that are relatively simple, but it also often serves as a stepping stone to more complicated problems, such as Convolutive Blind Signal Separation (CBSS) [47, 75]. In this work we will not focus on the question whether or not a specific real-world application can be represented accurately by the model in (1.1.1) and its related assumptions, but rather on the actual estimation or separation process. Therefore, in this section some examples with artificially mixed signals are presented. During the development of the theory in the coming chapters, we will apply our method to similar examples and explain how the source signals can be recovered from the mixtures by exploiting a specific type of statistical diversity in the signals. In order to focus on main principles, for the moment we do not consider additive sensor noise but defer this issue till later chapters. Likewise, for the sake of simplicity of exposition the examples in this section involve only two source and two sensor signals. More complicated scenarios, including ones with more sources than sensors, will be considered later on. Hence, in the following examples we consider the instantaneous mixing model depicted in Fig. 1.3 (see Section 4.1 for details).

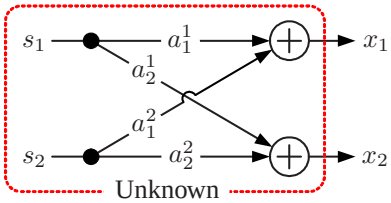

Figure 1.3: *Linear instantaneous mixing with two sources and two sensors.*

Stationary Gaussian AR(1) source signals In this example, two computer generated stationary Gaussian autoregressive (AR) source signals of order one are mixed. The source signals are generated independently of each other according to the following AR(1) model:

$$s_j[n] = \rho_j \, s_j[n-1] + w_j[n], \qquad j = 1, 2\,, \tag{1.1.3}$$

where ρ_1 and ρ_2 are AR(1) regression coefficients, and $\{w_1[n]\}_{n\in\mathbb{N}}$ and $\{w_2[n]\}_{n\in\mathbb{N}}$ are sequences of independently and identically distributed (i.i.d.) Gaussian random variables. Here, we choose $\rho_1 = 0.95$ and $\rho_2 = 0.40$. Furthermore, the variances of $w_1[n]$ and $w_2[n]$ have been chosen in such a way that $s_1[n]$ and $s_2[n]$ both have unit variance. The two source signals are mixed according to the model in (1.1.1), where the mixing matrix is given by:

$$\mathbf{A} = \begin{bmatrix} a_1^1 & a_1^2 \\ a_2^1 & a_2^2 \end{bmatrix} = \frac{1}{2}\sqrt{2} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}. \tag{1.1.4}$$

Note that the subscript indices of the elements correspond to row indices, which are associated with the sensors, whereas the superscript indices correspond to column indices, which are associated with the sources. For convenience and without loss of generality we have normalized the columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of $\mathbf{A}$ to unit norm. Fig. 1.4 shows 500 samples of $s_1[n]$ and $s_2[n]$ at the left side, and the two mixtures $x_1[n]$ and $x_2[n]$ obtained by substituting (1.1.4) into (1.1.1) at the right side. From (1.1.3) it is clear that both source signals are time series that possess an 'AR(1) temporal structure'. As can clearly be seen in Fig. 1.4, the variations in $s_1[n]$ are much slower than those in $s_2[n]$. This is due to the fact that the pole ρ_1 of AR(1) filter $H_1(z) \triangleq z/(z-\rho_1)$ associated with $s_1[n]$ is located much closer to the unit circle than

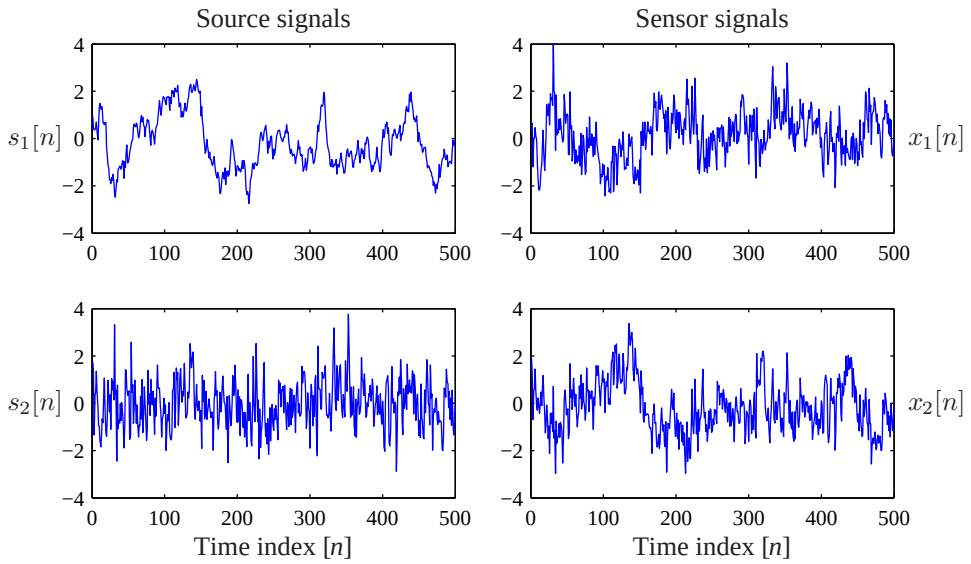

Figure 1.4: *Stationary Gaussian $AR(1)$ source signals (left) and their mixtures (right).*

the pole ρ_2 of AR(1) filter $H_2(z) \triangleq z/(z-\rho_2)$ associated with $s_2[n]$. Formulated differently, the correlation coefficient between adjacent samples of time series $s_1[n]$ is larger than that for $s_2[n]$. Furthermore, as expected, the right part of Fig. 1.4 shows that the mixed signals have very similar characteristics, e.g. they roughly have the same 'speed of variation'. The purpose of applying IBSS to this example is the estimation of $s_1[n]$ and $s_2[n]$ from $x_1[n]$ and $x_2[n]$ only, i.e. without knowing the mixing system $\mathbf{A}$. Blind identification of $\mathbf{A}$, and thus recovery of $s_1[n]$ and $s_2[n]$, is possible because two different mixtures are observed and the statistical characteristics of the source signals are sufficiently different. More specifically, the fact that the source signals have different AR(1) coefficients implies that they have linearly independent auto-correlation functions. This property will prove to be essential for the identification/separation method that we will develop in later chapters. It is reminiscent of the condition of sufficient statistical diversity posed in the IBSS problem statement on page 5. The presence of statistical dependence between different time samples of a signal is commonly referred to as *correlatedness*, *coloredness*, or *non-whiteness*. Since correlatedness is associated with the temporal relations between different time samples, it is an example of the presence of temporal structure in the data. In the next example, we will encounter another kind of temporal structure.

Speech signals Now, two real-world speech signals $s_1[n]$ and $s_2[n]$, which have been sampled at 8 kHz and normalized to unit variance, are mixed artificially by the same mixing matrix as in the previous example. Fig. 1.5 shows 8000 samples of the source signals $s_1[n]$ and $s_2[n]$ at the left side, and the two mixtures $x_1[n]$ and $x_2[n]$ at the right side. When several people are talking simultaneously, at the 'signal level' their words and sentences generally are independent of each other because the signal of one of the speakers does not provide any predictive information about the signal of another speaker. Hence, loosely speaking we can

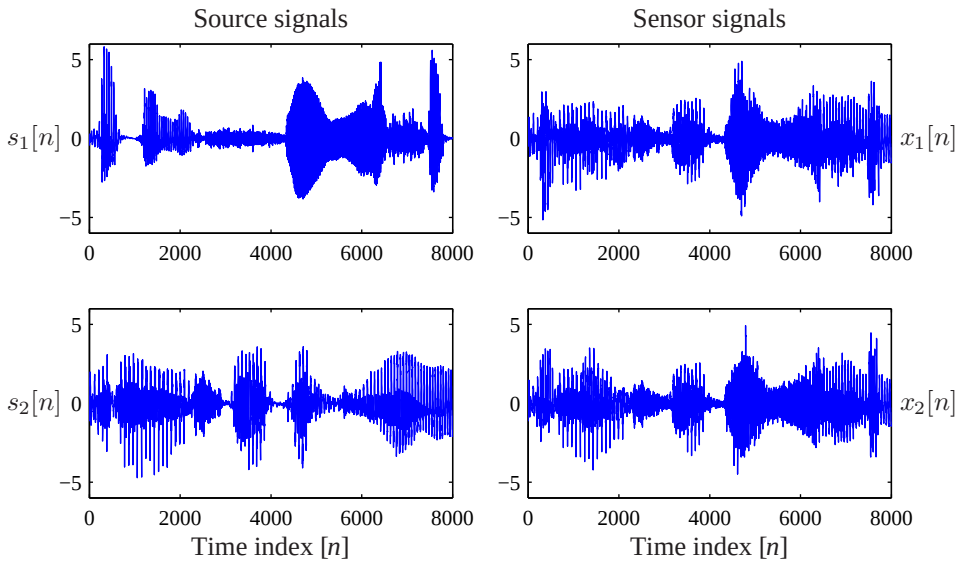

Figure 1.5: *Speech sources (left) and their mixtures (right).*

consider the sound signals produced by different speakers as being statistically independent. As in the previous example, the mixed signals are very similar and again blind recovery of the original speech signals is possible because two different mixtures are observed and the source signals have certain statistical characteristics that are sufficiently diverse. Now we can distinguish two types of temporal structure in the data. In addition to being non-white (for most parts), speech signals are also non-stationary at the word or sentence level. This means that the signal statistics change in time. For example, Fig. 1.5 clearly shows that the (short-term) variances change in time. Different stationary parts of speech represent different sounds because they possess different temporal characteristics. For example, the short-term time-averaged auto-correlation function of a speech signal corresponding to the vowel 'a' is different from that of 'u'. Likewise, speech is non-stationary because different sounds (and their corresponding statistics) are produced in the course of time. In fact, those changes in the statistics make a speech signal interesting as an information carrier.

The cocktail party problem mentioned at the beginning of this chapter is an example of the separation of speech signals. The examples given above with AR(1) and speech signals demonstrate typical signal scenarios with one-dimensional signals. Several other examples of IBSS with 1D signals can be found for example in telecommunication scenarios and also in the field of biomedical engineering, where the purpose of various applications is to reveal independent sources in different kinds of biological signals like EEG's and ECG's. In the next section, we present an example with non-white and non-stationary two-dimensional signals, viz. images. The exploitation of second order non-whiteness and non-stationarity for IBSS has been investigated by several researchers, e.g. see [14, 41, 43, 45, 46, 48]. In later chapters, we will develop a MIBI/IBSS method that can exploit both types of temporal structure in a completely similar and transparent way, both for second and higher orders.

Images In this final example, two 8-bit grayscale images of 512 by 512 pixels are mixed according to model (1.1.1) with mixing matrix $\mathbf{A} = \frac{1}{5}\left[\begin{smallmatrix} 4 & 3 \\ 3 & 4 \end{smallmatrix}\right]$, which has unit-norm columns. Fig. 1.6 shows the two source images at the left side and the two mixtures at the right side. Instead of dealing with one-dimensional signals, as we did in the previous two examples, now we are dealing with two-dimensional signals because images have two spatial dimensions (note that the term 'spatial' now has a meaning that differs from the one used in the previous sections). For clarity we replace the time index n in (1.1.1) by two 'spatial indices' u and v that represent the pixel locations in the horizontal and vertical directions respectively. For example, the (u, v)-th pixel of the j-th source image is denoted by $s_j[u, v]$. Hence, we obtain the following equivalent of (1.1.1) for spatial signals (or 2D signals of any kind):

$$\mathbf{x}[u, v] = \sum_{j=1}^{S} \mathbf{a}^j \, s_j[u, v] + \boldsymbol{\nu}[u, v] = \mathbf{A}\,\mathbf{s}[u, v] + \boldsymbol{\nu}[u, v] \qquad \forall\ (u, v) \in \mathcal{P}, \tag{1.1.5}$$

where:

$$\mathbf{x}[u, v] \triangleq \begin{bmatrix} x_1[u, v] \\ x_2[u, v] \end{bmatrix}, \quad \mathbf{s}[u, v] \triangleq \begin{bmatrix} s_1[u, v] \\ s_2[u, v] \end{bmatrix}, \quad \boldsymbol{\nu}[u, v] \triangleq \begin{bmatrix} \nu_1[u, v] \\ \nu_2[u, v] \end{bmatrix},$$

and $\mathcal{P}$ is the set of all valid pixel pairs within the limits of the images. In this example, the noise terms are assumed to be zero. Similarly to the speech example, since the source images

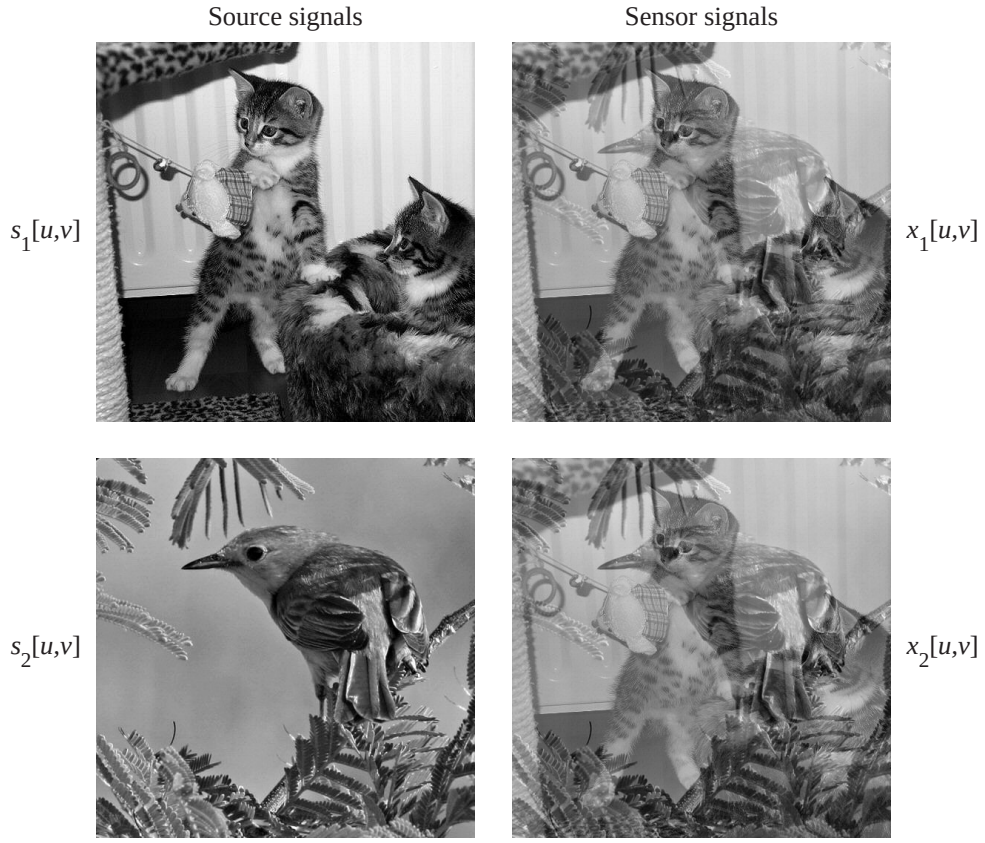

Figure 1.6: *Image sources (left) and their mixtures (right).*

have been captured from different unrelated source scenes, we can consider them as being statistically independent. This means that the intensity of a specific pixel in one source image does not provide any predictive information about the pixel intensity of any pixel in the other source image. Fig. 1.6 shows that both 'sensor images' contain mixtures of the kittens and the bird. Again, similarly to the previous examples blind recovery of the original images is possible because two different mixtures are observed and the source signals have certain statistical characteristics that are sufficiently diverse. Although the statistics are spatial in nature now (e.g. we need to compute spatial instead of temporal correlation functions, see Section 4.4.5), we can still distinguish the same two types of structure in the image data as we did for the speech data, viz. *spatial non-whiteness* and *spatial non-stationarity*. Fig. 4.19 on page 198 shows the separated images obtained by an algorithm that is developed in Chapter 4, and which exploits the spatial non-whiteness.

There are several real-world applications of blind image separation where the linear instantaneous model represents the physical situation reasonably well. Depending on the lighting, reflections, shadows, etcetera, the image of an object or scene as captured by a camera can vary significantly [23, 68]. It is often advantageous or needed to separate the intrinsic components from the extrinsic ones. The phenomenon of a virtual image, which is (semi-)reflected by a transparent medium that is situated somewhere along the optical axis between the imaged scene and the observation point, and superimposed on the imaged scene, is typical for many optical setups. It is often important to separate the real image of the scene from the virtual component. For example, the separation of reflections may be useful to facilitate tasks like object recognition, vision-based navigation (e.g. autonomous vehicles with on-board cameras where reflections from the windshield cause problems for navigation), for surveillance where activities behind a reflective window have to be guarded, for professional photographers (e.g. for the correct functioning of auto-focussing devices), etcetera. For IBSS purposes different observations of the same source images can be obtained for instance by using different polarizers. For more examples and details, see [23, 37, 68].

We will show in later chapters that the kind of signals we are dealing with, e.g. 1D time signals, 2D images, 3D volumetric data, or spectral data, does not make any difference for the MIBI/IBSS method presented in this thesis. Therefore, for convenience we will mostly consider time dependent signals and thus talk about 'temporal' structure. However, it should be kept in mind that 'temporal' can be replaced with any other appropriate adverb such as 'spatial' or 'spectral' that characterizes the nature of a signal.

1.1.3 Instantaneous Semi-Blind Source Localization (ISBSL)

Another problem that is closely related to MIBI is *Instantaneous Semi-Blind Source Localization (ISBSL) for narrowband signals*. The ISBSL problem deals with locating a set of narrowband sources using only the signals observed by an array of sensors [62, 84, 147, 179, 184]. As an example, we first consider the problem setup for a scenario with S sources and D sensors located in a two-dimensional plane, see Fig. 1.7. The sensors are indicated by black dots and the sources by white dots. The source signals propagate spatially through a medium such as air or water, and the resulting wavefront is sampled spatially and temporally by the sensor array. We assume that the sources in Fig. 1.7 are omnidirectional and located in the *far field*, i.e. the distances from the sources to the array are very large compared to the array aperture[‡],

[‡]The array aperture is the length/area/volume of the sensor constellation over which the energy is 'sensed'.

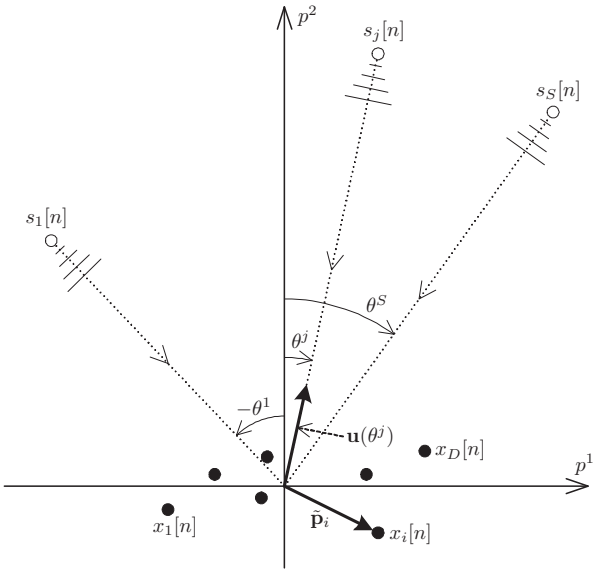

Figure 1.7: *Source Localization problem setup.*

which implies that the signal wavefronts are planar when impinging on the array. In this case, there is only one parameter characterizing the position of a source w.r.t. the chosen coordinate system, namely the so-called *Direction Of Arrival (DOA)*. The DOA is denoted by θ and defined as the angle between the line from the source to the origin and one of the axes of the coordinate system, e.g. in Fig. 1.7 the DOA of the j-th source w.r.t. the positive vertical axis is denoted by θ^j. For this simple example the purpose of ISBSL is to recover the DOA's of all sources. Often, it is convenient to model the position of a source in terms of polar or spherical coordinates. For a scenario in which the sensors and far field sources can be located 'anywhere' in three-dimensional space, two angles are necessary for uniquely characterizing a source position or direction of arrival. These angles are commonly called *azimuth* and *elevation* for the angles measured in the horizontal and vertical planes respectively. Evidently, for a scenario like this the purpose of ISBSL is to recover the azimuth and elevation of all sources. For the most general scenario, the purpose of ISBSL is to recover all parameters characterizing the source positions from the sensor data only. In fact, the quintessence of many array signal processing problems is the extraction of position related parameters such as DOA and range. These parameters may be of interest themselves or can for example be used to estimate (certain properties of) the source signals in the same spirit as IBSS. We will reason in the next paragraph and in Section 3.1 that by making assumptions on the source and noise signals that are similar or equal to those of MIBI (see Sections 1.1.1, 2.2.2, and 2.2.3), the ISBSL problem can be seen as a specific instance of a MIBI problem. In particular, we will assume in this thesis that the source signals are statistically independent and that the mixing matrix has a known *parameterization*, i.e. its functional dependence on the unknown parameters (e.g. DOA's) is known. Before explaining how the ISBSL problem can be cast into a MIBI problem, we formulate the main conditions and problem statement of ISBSL:

> ***Instantaneous Semi-Blind Source Localization problem statement:***
> The purpose of ISBSL is to identify the position parameters of all sources from the sensor array signals only. The main assumptions are that the source signals are narrowband, mutually statistically independent, that they possess sufficient statistical diversity, and that the propagation medium characteristics and sensor positions are known.

It is assumed that each source generates a wave field consisting of plane or spherical waves that propagate through space [62, 84, 179, 184]. We consider source signals that are generated by modulating a baseband information signal (which is the signal of interest) onto a carrier. At the receiver side, the signals are demodulated to baseband. When the bandwidth of the information signal is small compared to the carrier frequency, the resulting signal is called *narrowband* [90]. The complex representation of the modulating information signal is commonly called the *complex envelope* of the carrier or received signal. In the sequel, by 'source signal' we refer to this complex envelope. The narrowband assumption implies that the propagation time across the array for a narrowband signal from any direction is small compared to the time variations of the amplitude and phase of the complex envelope. This in turn implies that propagation time differences of a source signal between any pair of sensors can be modelled by a phase shift, which is a result that is crucial for obtaining an instantaneous mixing model. Such a model is adopted in the majority of the literature on source localization with sensor arrays that sense electromagnetic fields, such as the ones used in telecommunication applications. *Source localization is possible due to certain assumptions that are made on the propagation medium, the source signals, and the sensor characteristics and positions.* We adopt the common assumptions that the propagation medium is linear, homogeneous (i.e. not dispersive) and isotropic (i.e. propagation is uniform in all directions) [179]. In addition, we assume that the sources and sensors are omnidirectional. Under those assumptions, the response of the array to a single source signal is a function of the source position, the sensor positions, and the propagation medium parameters (e.g. propagation velocity). Because the propagation medium is linear the response to several incident source signals is given by the superposition of the individual source responses. In addition, because of the narrowband property the complex envelope of a received signal essentially remains constant during the time it takes for a wavefront to travel across the array aperture. Hence, the complex mixing process is instantaneous and thus narrowband source localization is also modelled by (1.1.1).

To get a firm grip on the relation between ISBSL and MIBI, we consider the *array response vector*, which in the context of ISBSL is defined as the noise-free response of the array to a unit-amplitude complex exponential plane wave coming from a certain source position. It is a function-valued vector of the parameters employed for characterizing a source position. For example, for the scenario in Fig.1.7 the array response vector is denoted by $\mathbf{z}(\theta)$ and is a function of the DOA θ, i.e. it represents the noise-free complex *array response* to a unit-amplitude complex exponential plane wave arriving from DOA θ. Its functional form for general sensor positions is given by equation (3.1.7) in Chapter 3, and for a *Uniform Linear Array (ULA)* by (3.1.13). Note that in addition to the DOA and the carrier frequency, $\mathbf{z}(\theta)$ also depends on the (fixed) array geometry, i.e. the positions of the sensors, and the propagation medium characteristics such as the propagation velocity. As an example, we briefly consider the situation that all sensors in Fig. 1.7 are distributed uniformly on the horizontal axis, thereby forming a ULA. As depicted in Fig. 1.8, we assume that the array consists of three sensors located at positions $-d, 0$, and d of the horizontal axis denoted by p^1, and that

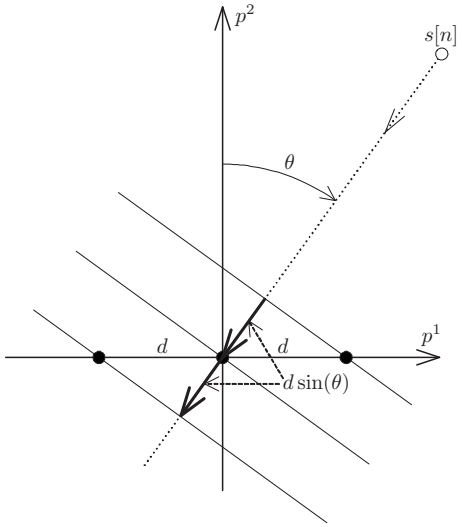

Figure 1.8: *Determination of array response vector for ULA.*

a unit-amplitude complex exponential plane wave coming from DOA θ with the positive vertical axis denoted by p^2 is incident on the array. Then, the propagation time difference of the wavefront for two adjacent sensors is given by $\tau(\theta) \triangleq d\sin(\theta)/c$, where c is the propagation velocity. Due to the single-frequency (narrowband) character of the source signal, the time delay $\tau(\theta)$ corresponds to the phase factor $\exp\left(-\jmath\omega_{\mathrm{c}}\tau(\theta)\right)$, where ω_{c} is the angular frequency of the carrier signal. Taking the middle sensor as a reference, it follows that the array response vector $\mathbf{z}(\theta)$ is given by:

$$\mathbf{z}(\theta) = \begin{bmatrix} \exp\left(-\jmath\omega_{\mathrm{c}}\tau(\theta)\right) \\ 1 \\ \exp\left(\jmath\omega_{\mathrm{c}}\tau(\theta)\right) \end{bmatrix} = \begin{bmatrix} \exp\left(-\jmath\omega_{\mathrm{c}} d\sin(\theta)/c\right) \\ 1 \\ \exp\left(\jmath\omega_{\mathrm{c}} d\sin(\theta)/c\right) \end{bmatrix}.$$

From the right-most expression, we clearly see that apart from the DOA θ and the angular frequency ω_{c} of the carrier, $\mathbf{z}(\theta)$ also depends on the array geometry (through the sensor spacing parameter d) and the propagation medium characteristics (through the propagation velocity parameter c). For more information and details, see Section 3.1.2.

In general, due to the available a priori knowledge about the propagation medium characteristics and sensor positions (see the problem statement), the specific dependence of an array response vector on the source position parameters is known. Since the purpose of ISBSL is to recover the position parameters of all sources from the sensor data, these source position parameters are truly *variables* of the array response vector, whereas the sensor position parameters and propagation medium characteristics should be considered as given *parameters*. As an example that can be generalized easily, we consider the following scenario. Assuming that the sources are located in three-dimensional space and in the *near field*, i.e. the distance from the sources to the array is of the same order as the array aperture, and using a spherical coordinate system, the array response vector can be written as a function of three variables, namely azimuth θ, elevation ζ and range ρ. In addition, the array response vector has the sensor positions $\tilde{\mathbf{p}}_1, \ldots, \tilde{\mathbf{p}}_D$ and the propagation velocity c as its parameters. Hence, it can be written as $\mathbf{z}\big(\theta, \zeta, \rho; \{\tilde{\mathbf{p}}_i\}_{1\leq i\leq D}, c\big)$, where the semicolon separates the variables from the

parameters. However, because the parameters are assumed to be known and fixed, we usually leave them out for the sake of brevity and expedience of notation. Hence, we will write $\mathbf{z}\big(\theta,\zeta,\rho\big)$ instead of $\mathbf{z}\big(\theta,\zeta,\rho;\{\tilde{\mathbf{p}}_i\}_{1\leq i\leq D},c\big)$, and similarly for other scenarios. The ISBSL mixing matrix $\mathbf{A}$ is obtained by stacking the array response vectors corresponding to the different sources along each other. Like an array response vector, it can be written in functional form in terms of its variables. For example, for the 3D near field case, we write:

$$\mathbf{A}\big(\tilde{\boldsymbol{\theta}},\tilde{\boldsymbol{\zeta}},\tilde{\boldsymbol{\rho}}\big)=\big[\mathbf{a}^1\ \cdots\ \mathbf{a}^S\big]=\big[\mathbf{z}\big(\theta^1,\zeta^1,\rho^1\big)\ \cdots\ \mathbf{z}\big(\theta^S,\zeta^S,\rho^S\big)\big]\in\mathbb{C}_D^S\,, \tag{1.1.6}$$

where θ^j,ζ^j,ρ^j and $\mathbf{z}\big(\theta^j,\zeta^j,\rho^j\big)$ are the azimuth, elevation, range, and array response vector respectively corresponding to the j-th source. The row vectors $\tilde{\boldsymbol{\theta}},\tilde{\boldsymbol{\zeta}},\tilde{\boldsymbol{\rho}}\in\mathbb{R}^S$ are defined by stacking the (set of) azimuthal angles $\big\{\theta^j\big\}^{1\leq j\leq S}$, elevation angles $\big\{\zeta^j\big\}^{1\leq j\leq S}$ and ranges $\big\{\rho^j\big\}^{1\leq j\leq S}$ respectively of all sources in row vectors as follows:

$$\tilde{\boldsymbol{\theta}}\triangleq\big[\theta^1\ \cdots\ \theta^S\big]\,,\quad \tilde{\boldsymbol{\zeta}}\triangleq\big[\zeta^1\ \cdots\ \zeta^S\big]\quad\text{and}\quad \tilde{\boldsymbol{\rho}}\triangleq\big[\rho^1\ \cdots\ \rho^S\big]\,. \tag{1.1.7}$$

For obvious reasons, the matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}},\tilde{\boldsymbol{\zeta}},\tilde{\boldsymbol{\rho}}\big)$ in (1.1.6) is called *array response matrix* (note that it is a function of the position parameters of *all* sources). Its extension to more general scenarios with more and/or different source position parameters is straightforward.

From the discussion above, it follows that *ISBSL can be considered as a parameterized version of MIBI because the columns of the mixing matrix have a known functional dependence on the source position parameters*. Mathematically, this is expressed by writing the array response vectors and matrix as a function of the variables that represent the source position parameters/coordinates. Depending on the considered model or application, different parameterizations may be used. Since the purpose of ISBSL is the unique determination of the source position parameters from the sensor data only, and because the sensor data is observed via the array response vectors (see (1.1.1)), the set of position parameters characterizing a source and its corresponding array response vector need to be related bijectively for all possible allowed values of these parameters. For instance, for the scenario in Fig. 1.7 a DOA θ and its corresponding array response vector $\mathbf{z}(\theta)$ must be related bijectively for all possible allowed values of θ in order to allow the unique determination of the S source DOA's $\theta^1,\ldots,\theta^S$ from the sensor data. Because the functional dependence of the array response vectors on the source position parameters is assumed to be known, the ISBSL problem is not fully but semi-blind. As we will see in later chapters, the so-called *subspace techniques* [16, 62, 84, 137, 138, 147, 184] that are commonly employed in the *subspace approach* to source localization problems can be generalized to fully blind problems.

Finally, we note that in practice the narrowband source localization problem often is more complicated than we have sketched in this section. For example, the sensor characteristics may not be as ideal as we have assumed (a possible solution is to calibrate the array, but this might complicate the problem considerably), multipath propagation may occur, and the source signals may not be (fully) statistically independent. In general, when the involved physics concerning the sources, propagation medium, and sensor characteristics cannot be appropriately determined or modeled (for instance, unknown source and sensor directivity patterns, inhomogeneous and isotropic propagation medium), the functional dependence of the array response matrix on the source positions is not known (sufficiently accurate), and thus the problem is truly blind. In this case, we have to solve a more general, e.g. fully blind and/or convolutive, MIBI problem.

Examples

Radio Telescope Array In radio astronomy, the interest is in radio emission from celestial sources [186]. Several radio telescope arrays (see Fig. 1.9) consisting of tens or hundreds of antenna elements, and ranging from hundreds of meters to thousands of kilometers, have been constructed. Amongst the most well-known are the Cambridge 5-km telescope [135], the Very Large Array (VLA) telescope of the National Radio Astronomy Observatory in New Mexico [84, 90, 155], the Westerbork synthesis radio telescope [113], and the Square Kilometer Array Interferometer (SKAI) [136, 142]. The purpose of radio telescope array processing [174, 186] is not restricted to source localization, but also encompasses the complete reconstruction of radio images of the sky, with emphasis on resolution, sensitivity, dynamic range, noise suppression, and interference rejection. The latter point is particularly relevant nowadays due to the heavy use of the electromagnetic spectrum for telecommunication purposes [101]. For more information, see [84, 90].

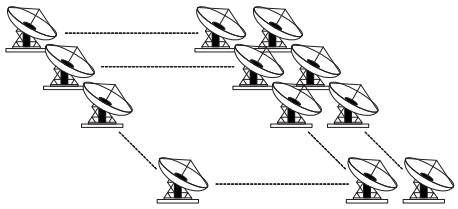

Figure 1.9: *Radio Telescope Array.*

Hydrophone arrays In underwater acoustics, hydrophone arrays are employed in various applications such as geophysical research and detection of hostile submarines and/or ships [84, 120, 121]. See Fig. 1.10 for some possible deployments. For example, cylindrical arrays are typically lowered from a ship or off-shore platform, towed arrays are pulled by a ship or submarine, and many submarines contain a hydrophone array mounted in their hull. For more information, see [84, 90, 120].

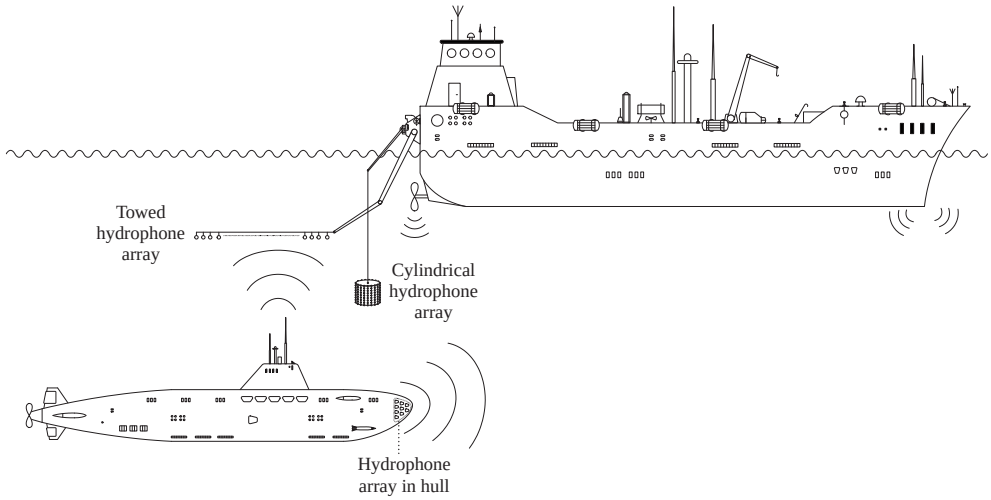

Figure 1.10: *Deployments of hydrophone arrays.*

1.2 Motivations, contributions and problem statement

In this section, we first give a brief overview of the main principles of, and approaches to, blind signal processing problems, which are treated elaborately in Chapter 2. Then, by analyzing these approaches we deduce lacunas in the literature that motivate our work. In addition, we discuss several other motivations extensively. Finally, we formulate our main thesis problem statement and objectives, and describe our main contributions.

1.2.1 Main approaches to blind problems

Many methods and algorithms for performing Multiple-Input Multiple-Output Instantaneous Blind Identification, Instantaneous Blind Signal Separation, and Instantaneous Semi-Blind Source Localization have been developed during the last decades. For example, see [34, 47, 75, 79, 80] and the references therein. Although many of these methods and algorithms are different, their underlying principles can be classified into a few distinctive classes that mainly differ in their assumptions about the spatial and temporal statistical structure and variability of the involved signals. Until now, usually three different principles or approaches have been distinguished, most of which are related to the assumption of statistical independence between the source signals[‡]:

PR1: ***Non-Gaussianity***
This approach assumes that the source signals are mutually statistically independent and that at most one of them is Gaussian. By the Central Limit Theorem (CLT), the sensor/observation signals, which are linear mixtures of the statistically independent source signals, are more Gaussian than the source signals. This property can be exploited for identifying the mixing system and/or recovering the source signals. Since the non-Gaussianity approach exploits the marginal non-Gaussianity of the source signals, it requires the use of Higher Order Statistics (HOS), e.g. in the form of higher order cumulants. See Sections 2.3.1 and 2.7, and references [5, 27, 29, 30, 34, 80] for examples and more information.

PR2: ***Spatial uncorrelatedness and temporal non-whiteness/auto-correlatedness***
This approach assumes that the source signals are mutually spatially and temporally uncorrelated (note that this is less restrictive than being mutually statistically independent), and that each source has non-vanishing temporal auto-correlation (on some Region Of Support) that is sufficiently different from the other sources, i.e. it assumes that the spectra of the source signals are sufficiently different. Under those assumptions Second Order Statistics (SOS) can be exploited to estimate the mixing matrix. See Sections 2.3.2 and 2.8, and references [14, 48, 106, 107, 114] for examples and more information.

PR3: ***Spatial uncorrelatedness and temporal second order non-stationarity***
Like *PR2* this approach assumes that the source signals are mutually spatially and temporally uncorrelated, but instead of exploiting the second order temporal auto-correlation of the sources, it exploits their second order temporal non-stationarity under the assumption that each source signal has a 'non-stationarity pattern' that is sufficiently different from the others. See Sections 2.3.2 and 2.8, and references [43–46, 110, 125] for examples and more information.

[‡] '*PR*' in abbreviations *PR1-PR3* stands for 'PRinciple'; see also the list on page xviii.

1.2.2 Unifying approach for exploiting temporal structure of any kind

Approach *PR1* requires the use of Higher Order Statistics and typically neglects any temporal structure in the data, i.e. only lag-zero higher order cumulants are considered. On the contrary, approaches *PR2* and *PR3* require the presence of temporal structure in the data and typically neglect any HOS in the data. They are based on Second Order Statistics and exploit what we will refer to as the *Second Order Temporal Structure (SOTS)*. *PR2* and *PR3* are often considered and treated separately from each other. However, as is intuitively clear and as we will show in the coming chapters, essentially they are quite similar because both are examples of exploiting spatial uncorrelatedness together with second order temporal structure in the data (or spatial structure in case of images, see Section 1.1.2). This calls for a unifying approach to the exploitation of both temporal non-whiteness and non-stationarity, or any possible kind of temporal structure. The method that will be presented in this thesis complies with this demand, both for SOS and HOS.

1.2.3 Unifying method for exploiting temporal structure of any order

What is lacking from the categories *PR1-PR3* discussed in Section 1.2.1 is an approach that is based on the exploitation of the *Higher Order Temporal Structure (HOTS)* in the data, i.e. the combination of HOS and temporal structure. This has been one of our main motivations for performing research in this direction. Somewhat surprisingly, only very few algorithms that do exploit the HOTS in the data have been developed [40, 190]. Possible reasons might be the difficulty of extending existing methods for the exploitation of SOTS to HOTS, and the difficulty or computational complexity of accurately estimating the involved HOS. As we will show, it is possible to develop methods that exploit the temporal structure in the data of any kind and any order in a unifying way. This approach is complementary to most common approaches, which usually consider SOS and HOS based methods separately and design specific algorithms for each of them. For this reason, many similarities between them are overlooked.

The first problem mentioned above, i.e. the difficulty of extending existing SOTS methods to HOTS, is mainly due to the specific nature of many SOTS based algorithms that requires the use of conventional linear algebra tools. In fact, signal processing problems are commonly formulated more or less deliberately as linear algebra problems. This certainly may have significant advantages because mainly systems of linear equations or (generalized) eigenvalue decomposition problems need to be solved, for which many good off-the-shelf algorithms exist. However, it also has the disadvantage that the underlying more general and possibly nonlinear structure of the problem often is overlooked or neglected. In our opinion, *always (blindly) trying to (re)formulate a problem in linear algebra terms may disguise the true nature or structure of the problem, and impedes the development and use of nonlinear algebra tools in signal processing*. In other research areas nonlinear tools are employed and developed more often. In particular, several approaches have been developed for the problem of finding multiple solutions of a multivariate system of nonlinear equations, which arises in our problem and many other (signal processing) applications. In general, this problem is very difficult because a global zeroing or optimization method that can find multiple or all solutions is required. In practice, often heuristic methods are used that are unable to find all relevant solutions. In this thesis, among other contributions we will 'project' the MIBI problem onto the problem of solving a multivariate system of homogeneous polynomial or

polynomial-like equations (see Section 1.2.8). The vector-valued solutions of this system are the estimations of the columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of the mixing matrix $\mathbf{A}$; in order to obtain them, we will use a method called *homotopy*[‡] [2, 105, 116, 144]. Simply stated, a homotopy method provides a deterministic means for solving a system of nonlinear equations by smoothly deforming the known solutions of a simple start system into the desired solutions of the target system. It is relatively unknown and unused in the signal processing community. Since in our opinion homotopy techniques are very powerful and useful for several signal processing problems, one of our goals in this work is the advocation of homotopy methods for use in the signal processing research area.

The second problem mentioned above, i.e. accurate estimation of HOS from data, is encountered in several applications utilizing estimated HOS. According to Brillinger [22], the sample size needed for the estimation of the l-th order statistics of a stochastic process, under the constraint of prescribed estimation bias and variance, increases almost exponentially with l. Another disadvantage of HOS is that they can be very sensitive to outliers in the data. Under certain conditions these consequences can be alleviated. For example, if the data is ergodic and stationary up to the exploited (cumulant) order, and the MIMO system is time-invariant, sufficient observation samples may be acquired to obtain accurate estimates of the required statistics. Since our focus will be on principles, concepts and insight, those latter issues will not be addressed extensively in this thesis. However, as we will see in later chapters, for many scenarios our theory yields practical algorithms (i.e. algorithms that do not lead to a prohibitively large required number of samples or computational complexity) for exploiting HOTS for orders smaller than five.

1.2.4 More sources than sensors: underdetermined blind problems

With most approaches to MIBI, IBSS and ISBSL, problems arise when more sources than sensors are present. This so-called *underdetermined* or *overcomplete* scenario [18, 19, 29, 56, 61, 96, 102] is typically (considered to be) much more difficult than the conventional case where the number of sources is equal to or smaller than the number of sensors. However, nowadays it is recognized by several researchers that studying underdetermined blind problems becomes more and more important because in many practical applications there are more sources than sensors. For example, in several practical scenarios many interfering and/or noise sources propagating towards the sensor array are present. Such signals can be considered as undesired source signals in our model (1.1.1) because they are not just additive sensor noise. Since the mixing matrix has more columns than rows in the underdetermined case, it does not have a left inverse. Hence, the source signals cannot be recovered by a linear transformation of the sensor signals and the conventional IBSS approaches do not work because these are based on the assumption that there exists an inverse of the mixing system $\mathbf{A}$, viz. the de-mixing system $\mathbf{W}$; see also (1.1.2). Therefore, the common approach to underdetermined IBSS consists of two stages. Firstly, the mixing matrix is identified, and secondly the sources are estimated from the sensor signals and the estimated mixing matrix in as far this is possible according to some measure. Note that for MIBI and ISBSL only the first stage is necessary since their purpose is to estimate (parameters of) the mixing system.

A few methods exist that can handle the first stage for underdetermined instantaneous blind problems. Most of them assume that the source signals are *sparse* [18, 19, 96, 102] or

[‡] See Appendix E for an introduction to homotopy methods.

that their regions of support in some transform domain (where they are non-zero) are disjoint. Simply stated, this means that only a few of the source signals are allowed to differ significantly from zero at any point in some appropriate representation domain. Another category of methods that do not require the sparsity assumption is based on the (approximate) algebraic decomposition of a higher order *tensor*. In this context, a tensor can be considered as a higher-dimensional array, i.e. a generalization of a matrix; see Section 2.7.3.1 for more information. There are several methods that are based on the (approximate) algebraic decomposition of a lag-zero higher order cumulant tensor. For example, see the methods by Cardoso [29], De Lathauwer [61], and Comon and Mourrain [56]. Since these methods do not require the sparsity assumption, they are more generally applicable.

It is commonly believed that HOS are required for dealing with underdetermined scenarios and thus that the SOS based approaches of categories *PR2* and *PR3* in Section 1.2.1 will not work. In this thesis, we will show that by exploiting the temporal structure in the source signals it is possible to deal with underdetermined scenarios with a sufficiently large number of sensors by using SOS only. For example, when using three sensors and second order statistics four mixing matrix columns can be identified. Among other things, the maximum number of columns of the mixing matrix that can be identified correctly for a given number of sensors depends on the order of the exploited temporal structure (assuming that the temporal 'auto-structures' of the source signals are sufficiently diverse for that order). For example, we will demonstrate later on that with two sensors the number of columns of the mixing matrix that can be identified is equal to the order of the exploited statistics. The precise dependence of the maximum number of identifiable columns/sources on the number of sensors and the order of the statistics will be determined in later chapters. This is another motivation for, and contribution of, this work because virtually all MIBI methods that are able to handle the underdetermined problem do not describe this dependence. Due to their inherent parameterization, i.e. the use of more a priori knowledge, for ISBSL problems the maximum number of identifiable columns often exceeds that for the more general MIBI case.

The second stage of noise-free underdetermined IBSS, i.e. the recovery of the source signals for a certain set of time indices Ω from the estimated mixing matrix $\hat{\mathbf{A}}$ and the observations $\{\mathbf{x}[n]\}_{n\in\Omega}$ requires the solution of the system $\{\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]\}_{n\in\Omega}$ for the time-dependent source signal vector $\mathbf{s}[n]$. In Section 2.2.4, we will briefly discuss the most common approach to this problem, but as we have said above in this work we will mainly focus on the matrix estimation stage. For more elaborate information, see [102], [153], and the references therein. Here, we limit ourselves to the following remarks. Since the size of the mixing matrix $\hat{\mathbf{A}}$ is D by S with $S > D$, the solution of $\{\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]\}_{n\in\Omega}$ yields an affine vector space of dimension at least $S - D$. Hence, there is no unique solution and an additional criterion is required for selecting a specific solution. A commonly employed criterion is derived from the maximum-likelihood approach of maximizing the a posterior probability density function $p_{\mathbf{s}[n]|\mathbf{x}[n];\hat{\mathbf{A}}}(\mathbf{s}[n]|\mathbf{x}[n];\hat{\mathbf{A}})$ of $\mathbf{s}[n]$ for all $n \in \Omega$, where $\hat{\mathbf{A}}$ and $\{\mathbf{x}[n]\}_{n\in\Omega}$ are considered to be known [102], [153]:

$$\hat{\mathbf{s}}[n] \;=\; \underset{\mathbf{s}[n] \text{ s.t. } \mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmax}} \; p_{\mathbf{s}[n]|\mathbf{x}[n];\hat{\mathbf{A}}}(\mathbf{s}[n]|\mathbf{x}[n];\hat{\mathbf{A}}) \qquad \forall\, n \in \Omega\,. \tag{1.2.1}$$

This is a linear optimization problem for which several known algorithms exist [115], [108].

1.2.5 Annihilating the influence of noise with 'simple' structure

In many applications/scenarios, it is (or may be) assumed that the noise signals have a 'simpler' temporal structure than the signals of interest, i.e. the source signals. For example, this is the case when the noise signals are white and the source signals colored. Another example is the situation where non-white noise signals have non-zero auto-correlation only on a finite Region Of Support (ROS), e.g. a small set of lags for stationary signals, that is smaller than that of the source signals. We will show in later chapters that under some assumptions, which will be made precise later on, our method can annihilate the effects of additive sensor noise on the estimation of the mixing matrix in such scenarios. For instance, if the source signals have auto-correlation functions that are sufficiently different/diverse, and additive white noise is present at the sensors, the mixing matrix can be estimated in a noise-free manner by using only the sensor correlation values for lags unequal to zero; see Chapters 4 and 5. In general, *the influence of additive sensor noise on the estimation of the mixing system can be annihilated by considering the sensor cumulants functions only for those specific tuples of time, time-lag, or lag indices for which the noise cumulant functions are zero*. This principle is a major advantage of exploiting the temporal structure in the data.

The concept of annihilating the influence of additive sensor noise by considering the sensor cumulant functions only for tuples of time, time-lag, or lag indices for which the noise cumulant functions are zero is similar to another common principle in signal processing, viz. that of suppressing Gaussian noise by using fourth order cumulants; e.g. see [30]. The latter idea is based on the fact that all cumulants of order larger than two are zero for Gaussian random variables (see also *UCP2* on page 425) and it can easily be generalized as follows: *if the noise signals have non-zero cumulants only for orders up to and including $l-1$, the influence of noise on the estimation of the mixing system can be annihilated by only using sensor cumulants of order l and higher.* This principle is an advantage of using HOS. To emphasize the parallel between the ideas in the previous and this paragraph, we conclude this section with the following comparison (where we should keep in mind the generalizations of this parallel). The advantage of using HOS is that Gaussian noise can be suppressed no matter what spatial and temporal structure it has. Note that Gaussian random variables have the most informative probability density function (pdf) in the sense that the Gaussian pdf has the largest entropy[‡] among all random variables with the same variance; see also Section 2.7.2.1. On the other hand, the advantage of using temporal structure is that temporally white noise can be suppressed no matter what spatial structure it has and for what orders it has non-zero statistics. Note that a sequence of statistically independent random variables is the most informative among all sequences of random variables in the sense that there is no mutual information[‡]between successive variables. Obviously, combining the exploitation of HOS and temporal structure has both advantages, thereby providing another motivation for developing a (unifying) method for exploiting the HOTS in the data.

1.2.6 Subspace methods and arranging sensor statistics

Subspace methods [10, 16, 137, 138, 147, 171, 173, 175, 184] are commonly used for solving the semi-blind DOA estimation problem and for several other parameterized MIBI problems. Most of them rely on partitioning the eigen-space of the lag-zero sensor correlation matrix into a subspace associated with the source signals, the so-called *signal subspace*, and

[‡] See Section 2.7.2 for an explanation of information theoretical concepts.

a subspace associated with the noise signals (or, more appropriately, not associated with the source signals), the so-called *noise subspace* [147, 179, 184]. The desired parameters, e.g. the source DOA's, are determined by the intersections between the signal subspace and the array manifold (see Section 1.1.3 and Chapter 3 for more information), or, formulated differently, by the parameter values for which the array response vector function is orthogonal to the noise subspace. Among other differences, the various subspace methods differ in the way the desired parameters are extracted from the signal and/or noise subspace(s). Well-known algorithms are Multiple Signal Classification (MUSIC), which is discussed in Section 3.2.2 and e.g. [16, 137, 138, 147], Root Multiple Signal Classification (ROOT-MUSIC), which is discussed in Section 3.2.3 and e.g. [10, 147]), and Minimum-Norm (MIN-NORM), which is discussed in Section 3.2.4 and e.g. [98, 99, 154].

Although most subspace methods are based on the conventional lag-zero sensor correlation matrix $\mathbf{R}^x \triangleq E\{\mathbf{x}[n]\,\mathbf{x}[n]^H\}$ (note that $E\{\cdot\}$ denotes the ensemble averaging operator, see Appendix B), the underlying concepts can be generalized in different ways. Firstly, HOS can be exploited instead of SOS. For example, a matrix of fourth order sensor cumulants can be used instead of second order correlations. This directly leads us to the second kind of generalization, viz. the way the sensor cumulant samples are arranged in a matrix is arbitrary to a certain extent and thus can be chosen is several manners. In the literature, usually a well-known fixed arrangement is considered and no attention is paid to this issue. In this work we will show that the way of arranging the cumulant samples significantly influences the solution method, the number of identifiable columns, etcetera. Much insight is obtained by considering several arrangements. In a sense that will be made precise later on, our method for exploiting the temporal structure in the data is at one end of a whole range or spectrum of methods, whereas the methods exploiting only lag-zero cumulants are at the other end. Thirdly, subspace methods can be generalized to include temporal information. Finally, although subspace methods are typically used in parameterized array signal processing tasks, we will show that they can also be generalized to unstructured, i.e. fully blind, MIBI problems. Because subspace methods play an important role in this thesis, an elaborate explanation of their main principles is given in Chapter 3.

1.2.7 Conjugation pattern

For complex-valued signals different moment and cumulant functions can be defined corresponding to the pattern in which the arguments of the functions are conjugated. As is explained in Section B.3, for a certain signal scenario the *conjugation pattern* of the arguments of the involved functions can be chosen in several ways, which may or may not be sensible depending on the complex statistical structure of the source signals. Therefore, in practice the conjugation pattern should be chosen in accordance with the characteristics of the involved signals. As an example, consider the two correlation matrices defined by $\mathbf{R}^{x,\circ\circ} \triangleq E\{\mathbf{x}[n]\,\mathbf{x}[n]^T\}$ and $\mathbf{R}^{x,\circ *} \triangleq E\{\mathbf{x}[n]\,\mathbf{x}[n]^H\}$, and assume that $\mathbf{x}[n]$ is a vector containing *complex circular* signals [94, 147] of unit variance. Then, it can easily be shown [94, 147] that $\mathbf{R}^{x,\circ\circ} = \mathbf{0}$ and $\mathbf{R}^{x,\circ *} = \mathbf{I}$. Hence, in this case $\mathbf{R}^{x,\circ\circ}$ does not contain any information. For signals that are not complex circular, this need not be the case and the situation might even be the other way around. Note that for real-valued signals both correlation matrices are the same. See Sections A.5, A.6 and B.3 for more information. The theory that we will develop in this work is general and unifying w.r.t. the order of the exploited temporal structure and the employed conjugation pattern.

1.2.8 Thesis problem statement, objectives and contributions

Inspired by the problem descriptions in Section 1.1 and the motivations given above, we now formulate the main Thesis Problem Statement (TPS) and the main objectives of our work.

Thesis problem statement:
Develop a method for performing MIBI, IBSS and ISBSL based on the temporal structure in the data of arbitrary order l such that:

TPS1: it is unifying w.r.t. the types of temporal statistical variability in the data, such as l-th order non-whiteness and non-stationarity;

TPS2: it is unifying w.r.t. the order l of the exploited temporal structure, i.e. as long as certain conditions on the temporal structure of the source and noise signals are satisfied, l can be chosen arbitrarily;

TPS3: for complex systems/signals it is unifying w.r.t. the conjugation pattern;

TPS4: it allows to annihilate the influence of additive sensor noise that has a simpler temporal structure than the source signals;

TPS5: it allows to make trade-offs between various quantities such as the maximum number of sources or mixing matrix columns that can be identified, number of sensors, exploited order of statistics, exploited type of temporal structure, conjugation pattern, etcetera;

TPS6: it allows us to understand the dependence of the maximum number of identifiable columns/sources on the number of sensors, the order of the statistics, and the conjugation pattern;

TPS7: the use of homotopy methods for solving nonlinear systems of equations in signal processing is advocated.

Since the MIBI method that will be presented in this thesis adheres to the conditions listed here, these conditions at the same time represent the main objectives and contributions of our work. As we will see, our approach provides a lot of insight into the geometric and algebraic structure of the problem. In addition, it provides closed-form solutions for relatively simple scenarios and can deal with underdetermined blind problems.

Based on certain assumptions about the statistical structure and diversity of the signals and mixing system, and using subspace techniques, in this thesis the MIBI problem will be 'projected onto' two different dual mathematical problems. In the first problem, MIBI is projected onto the problem of solving a system of multivariate homogeneous polynomial or polyconjugal (polynomial-like) equations, the solutions of which yield (possibly scaled) estimates of the columns of the mixing matrix. A so-called homotopy method will be presented for solving this system. In the second problem, MIBI is projected onto the problem of solving what we call a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem, whose generalized eigenvectors are related to the columns of the mixing matrix by a linear transformation. Under certain conditions, this problem can be solved using existing mathematical tools, whereas for some other conditions appropriate tools still need to be developed. The sense in which the two different formulations are complementary will be clarified in later chapters. In the course of the thesis, we will provide additional evidence of, and enhanced

insight into, the close relationship between solving systems of polynomial equations on the one hand, and solving (generalized) eigenvalue problems on the other hand [117, 118, 146]. The assumptions underlying the two problem formulations are defined on a certain Region Of Support (ROS) consisting of a set of chosen time, time-lag, or lag index tuples, and mainly serve to ensure that:

MAS1: The source signals are mutually unrelated;

MAS2: Sufficient temporal structure is present in the source signals;

MAS3: The source and noise signals are mutually unrelated;

MAS4: The noise signals have a 'simpler' temporal structure than the source signals.

We will state and discuss these assumptions in more detail in later chapters (the abbreviation '*MAS*' stands for 'Main ASsumption').

We conclude this section by mentioning another contribution of our work. Namely, along the way we identify several unsolved or incompletely solved mathematical problems that are useful for our problem and for signal processing in general. In our view, solutions to those problems are particularly relevant for many signal processing problems and also for other engineering areas; hence they need to be addressed. Among other things, these include the generalization of GEVD algorithms to MMGEVD algorithms, the development of general MMGEVD algorithms that can also handle rectangular matrices, robust approximate MMGEVD algorithms, robust joint approximate diagonalization algorithms for higher order tensors, and robust solution methods for systems of multivariate polynomial/polyconjugal equations with approximated coefficients.

1.3 Publication overview

In [167, 170], we first investigated the principles of exploiting non-whiteness and non-stationarity in the data for achieving IBSS in a unifying manner for the simplest possible case with a 2×2 mixing matrix and SOS. For this scenario, we showed that a specific closed-form solution can be derived. After that, in [161] we formulated the theory for a real-valued square mixing matrix $\mathbf{A} \in \mathbb{R}_D^D$ and SOS in such a way that it can be generalized in several directions. In that paper we first derived the system of multivariate homogeneous polynomial equations referred to above, and its connection with the dual Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem. Moreover, insight into the algebraic structure of the problem was provided by showing that the columns $\mathbf{a}^j \in \mathbb{R}_D$ of $\mathbf{A}$ satisfy a system of D-variate 2-homogeneous polynomial equations. Likewise, insight into the geometric structure of the problem was provided by showing that the zero contour levels of the involved homogeneous polynomials are cones in D-dimensional space and that the columns of the mixing matrix are defined by the intersections between all these cones. In [165], we demonstrated the viability of our concepts and solution approach by presenting a MIBI method based on SOS without conjugations and a homotopy method. As an example, an algorithm specific for the scenario with three sensors and four sources was provided, thereby demonstrating the ability of estimating a mixing matrix for a scenario with more sources than sensors by exploiting SOS only, and the applicability of homotopy methods to solving the non-exact system of polynomial equations. The generalization of this work to a MIBI scenario with an arbitrary

conjugation pair was presented in [166]. In [162], we generalized our MIBI problem formulation to l-th order statistics and a $D \times S$ real-valued mixing matrix $\mathbf{A}$. For this scenario, it was shown that the columns of $\mathbf{A}$ satisfy a system of D-variate l-homogeneous polynomial equations. Likewise, it was shown that the zero contour levels of these homogeneous polynomials are 'l-cones' in D-dimensional space and that the columns of $\mathbf{A}$ are defined by the intersections between all these cones. In addition, a first result on the maximum number of identifiable columns in terms of D and l was presented. All those concepts have been applied to Direction Of Arrival estimation of superimposed statistically independent complex narrowband signals in noise, where complex mixing matrices are involved (see Section 1.1.3). For example, in [164] and [163] we have presented algorithms similar to the conventional MUSIC algorithm for a Uniform Linear Array (ULA) and an arbitrary planar array respectively. In other 'non-MIBI' papers [168, 169], we have shown that non-stationarity of the data can be very useful in other signal processing problems as well.

1.4 Notation

The notation employed in this thesis has been set up in such a way that it allows a uniform description of the developed theory for the general MIBI problem. For instance, for any considered order of the temporal structure, the problem formulation and the involved quantities have the same structure. At first sight, our notation and problem formulation might seem much more convoluted than necessary for some of the problems considered in this work. However, as we will see, they have the significant advantage that they make all kinds of generalizations much easier, and in a sense even trivial. For example, we heavily employ both sub- and superscript indices. This allows us to make a clear distinction between quantities that are essentially different in a natural way, whereas such a distinction would not have been so clear and natural when using e.g. only subscript indices. Most of our notational conventions are introduced in the text and summarized for reference in Appendix A. We suggest the reader to read Sections A.1 and A.2, where we explain our notation regarding vectors, matrices, vector spaces, tuples and sets respectively. See also the list of symbols on page xix and the list of mathematical operator names on page xxii. Here, we emphasize two important notational issues regarding linear vector spaces of discrete-time functions on the one hand, and bijective mappings between functions and vectors, and function-valued vectors and matrices, on the other.

Firstly, throughout the thesis the space of discrete-time functions defined on a given Region Of Support (ROS) will be considered as a linear vector space. Consequently, a discrete-time function $f \equiv f[n]$ defined on a given ROS $\mathcal{T}$ will be considered as a vector in a linear vector space. For convenience of notation, let $\mathbb{C}[\mathcal{T}]$ ($\mathbb{R}[\mathcal{T}]$) denote the set or linear vector space of all complex-valued (real-valued) functions that depend on a certain discrete-time variable and are defined on a certain Region Of Support $\mathcal{T}$. Likewise, let $\mathbb{C}_M[\mathcal{T}]$ ($\mathbb{R}_M[\mathcal{T}]$) denote the set or linear vector space of all 'function-valued' column vectors of length M whose elements belong to $\mathbb{C}[\mathcal{T}]$ ($\mathbb{R}[\mathcal{T}]$). Let $f, g \in \mathbb{C}[\mathcal{T}]$ ($\mathbb{R}[\mathcal{T}]$) and let α be a scalar. Then addition is defined by the rule $(f+g)[n] = f[n] + g[n] \quad \forall\, n \in \mathcal{T}$, and scalar multiplication by $(\alpha f)[n] = \alpha f[n] \quad \forall\, n \in \mathcal{T}$. Equality of vectors is defined as follows: $f = g \iff f[n] = g[n] \quad \forall\, n \in \mathcal{T}$. The null or zero vector is the function that is identically zero on the ROS $\mathcal{T}$ and is denoted by 0 or $0[n]$. Depending on whether the considered functions and scalars are allowed to be real- or complex-valued, the space will also be referred to as the

vector space of real- or complex-valued discrete-time functions on the ROS $\mathcal{T}$. The linear span of a set of functions or vectors $\mathcal{V}$ in a linear vector space is denoted by $\mathcal{L}(\mathcal{V})$, and its dimension by $\dim(\mathcal{L}(\mathcal{V}))$.

Secondly, in addition to using the 'functional notation' described in the previous paragraph, we often use matrix-vector notation to describe the same or similar results. Formulating problems in these two equivalent ways often provides enhanced insight. In order to establish a bijective mapping between functions and vectors, and function-valued vectors and matrices, we need to set up appropriate correspondences as depicted in Fig. 1.11 on the next page for the multivariate case (see further). With each function $f[n]$ a row vector $\tilde{\mathbf{f}}$ is associated that contains the values of f for all $n \in \mathcal{T}$ in some arbitrary, but fixed, order. For example, if $\mathcal{T}$ is given by $\mathcal{T} \triangleq \{n_1, \ldots, n_N\}$, where $N \triangleq |\mathcal{T}|$, the row vector $\tilde{\mathbf{f}} \in \mathbb{C}^N$ $(\mathbb{R}^N)$ associated with $f[n] \in \mathbb{C}[\mathcal{T}]$ $(\mathbb{R}[\mathcal{T}])$ is defined by $\tilde{\mathbf{f}} \triangleq \begin{bmatrix} f[n_1] & \cdots & f[n_N] \end{bmatrix}$. This way, a bijective mapping between a function defined on some ROS $\mathcal{T}$ and its corresponding row vector is established. Similarly, a bijective mapping is established between the function-valued vector $\mathbf{f}[n] \triangleq \begin{bmatrix} f_1[n] & \cdots & f_M[n] \end{bmatrix}^T \in \mathbb{C}_M[\mathcal{T}]$ $(\mathbb{R}_M[\mathcal{T}])$ and the matrix $\mathbf{F} \in \mathbb{C}_M^N$ $(\mathbb{R}_M^N)$ defined by stacking the row vectors $\{\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_M\}$ on top of each other, or column vectors $\{\mathbf{f}[n_1], \ldots, \mathbf{f}[n_N]\}$ along each other, as follows:

$$\mathbf{F} \triangleq \begin{bmatrix} \tilde{\mathbf{f}}_1 \\ \vdots \\ \tilde{\mathbf{f}}_M \end{bmatrix} = \begin{bmatrix} \mathbf{f}[n_1] & \cdots & \mathbf{f}[n_N] \end{bmatrix} \in \mathbb{C}_M^N \, (\mathbb{R}_M^N) \, .$$

Clearly, the associated concepts are related bijectively as well. For example, the linear span of a set of functions $\{f_1[n], \ldots, f_M[n]\}$ defined on $\mathcal{T}$ is mapped bijectively to the linear span of the corresponding row vectors $\{\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_M\}$, and its dimension is equal to the rank of the matrix $\mathbf{F}$ associated with $\mathbf{f}[n]$ defined on $\mathcal{T}$.

Finally, note that we can extend the notation and all concepts described above to discrete multi-variable functions. For example, suppose that f depends on a multi-variable argument $\mathbf{n}$ and is defined on some Region Of Support (ROS) $\boldsymbol{\mathcal{T}}$. Then, we say that $f[\mathbf{n}] \in \mathbb{C}[\boldsymbol{\mathcal{T}}]$ $(\mathbb{R}[\boldsymbol{\mathcal{T}}])$. Furthermore, for $\boldsymbol{\mathcal{T}} \triangleq \{\mathbf{n}_1, \ldots, \mathbf{n}_N\}$ we can define a bijective mapping between $f[\mathbf{n}] \in \mathbb{C}[\boldsymbol{\mathcal{T}}]$ $(\mathbb{R}[\boldsymbol{\mathcal{T}}])$ and the row vector $\tilde{\mathbf{f}} \triangleq \begin{bmatrix} f[\mathbf{n}_1] & \cdots & f[\mathbf{n}_N] \end{bmatrix} \in \mathbb{C}^N$ $(\mathbb{R}^N)$. Likewise, as before we can define a bijective mapping between the function-valued vector $\mathbf{f}[\mathbf{n}] \triangleq \begin{bmatrix} f_1[\mathbf{n}] & \cdots & f_M[\mathbf{n}] \end{bmatrix}^T \in \mathbb{C}_M[\boldsymbol{\mathcal{T}}]$ $(\mathbb{R}_M[\boldsymbol{\mathcal{T}}])$ and the matrix $\mathbf{F} \in \mathbb{C}_M^N$ $(\mathbb{R}_M^N)$. See Fig. 1.11 for an overview of the various bijective mappings.

1.5 Thesis outline

In this section, we provide an outline of the thesis. To start with, in this chapter we have provided an introduction to the blind signal processing topics considered in this work. Secondly, Chapter 2 provides the theoretical background and a literature review of the instantaneous blind signal processing research area, and further contextualizes our work. Likewise, Chapter 3 presents the theoretical background of the subspace approach, which is of paramount importance for the unifying blind identification method that is developed in this thesis. Next, Chapter 4 gives an elaborate introduction to our problem solving approach for the simplest possible MIBI scenario with a 2×2 real-valued mixing system and stationary signals. This chapter explains the rationale behind using subspace techniques for exploiting the second order temporal structure in the source signals. Moreover, it serves to abstract and emphasize

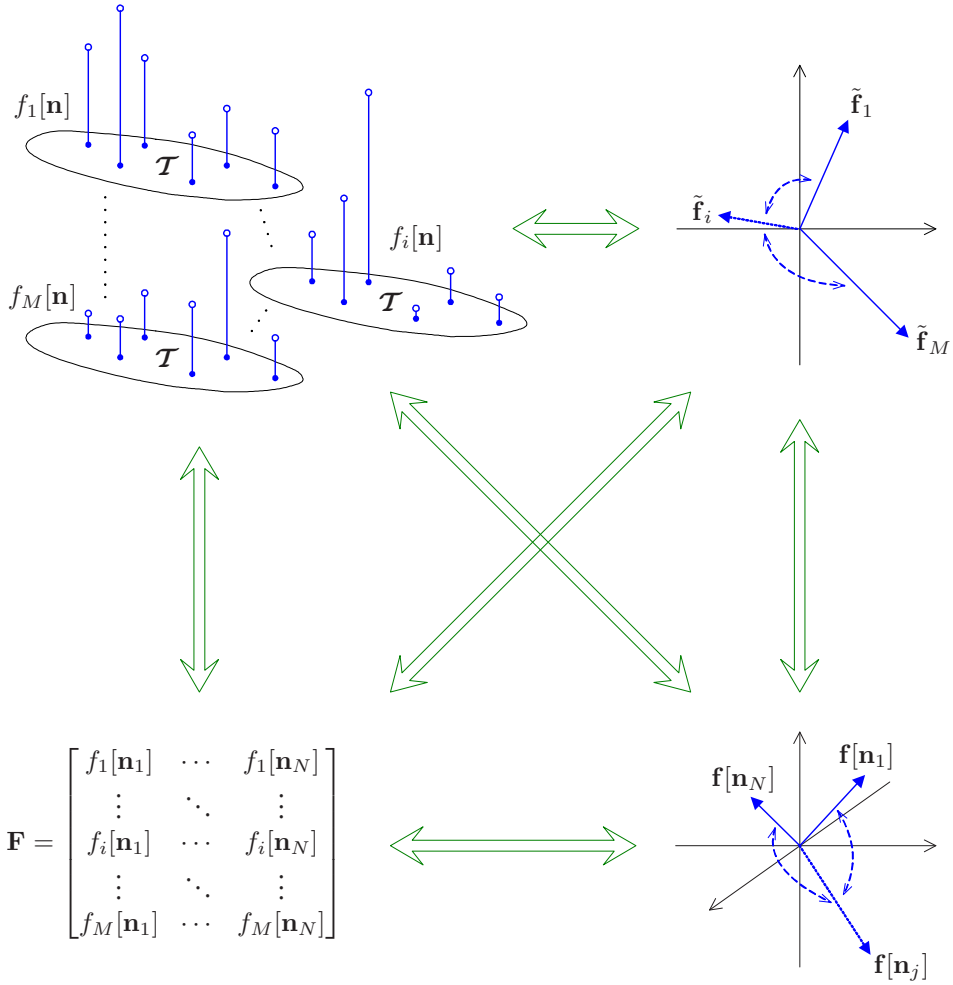

Figure 1.11: *Bijective mappings between functions, row vectors, function-valued vectors and matrices.*

the key points of the method and to introduce a notational device that allows a description of the theory that is uniform in several senses. Chapter 5 generalizes the ideas and framework from Chapter 4 in several directions. A scenario with a complex-valued $D \times S$ mixing system, complex-valued signals, second order statistics, and an arbitrary conjugation pair is considered. Subsequently, in Chapter 6 the sequence of generalizations presented in the previous chapters is completed by providing a unifying framework for exploiting arbitrary order temporal structure present in the source signals for performing MIBI with real- or complex-valued signals and mixing system, and an arbitrary conjugation pattern. This thesis has been set up in such a way that new theory is introduced gradually in the course of the chapters culminating in the most general and complex form of the theory in Chapter 6. In Chapter 7 the developed theory is applied to narrowband source localization, yielding several new algorithms. Furthermore, several issues related to array design are discussed. Finally, in Chapter 8 we discuss trade-offs, conclusions and future research.

1.6 Conclusions

In this chapter, we have introduced, contextualized and motivated the main topics considered in this thesis, viz. Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI), Instantaneous Blind Signal Separation (IBSS), and Instantaneous Semi-Blind Source Localization (ISBSL). In all three problems the source signals and MIMO instantaneous transfer system are unknown, and little a priori information is available. For these reasons, the problems are called 'blind'. Since in ISBSL slightly more prior information is available, viz. the parameterization of the mixing matrix is known, this problem is called 'semi-blind'. With respect to the goals of MIBI, IBSS and ISBSL, we have explained that for MIBI the main interest is in the transfer system, for IBSS in the waveforms of the source signals, and for ISBSL in the positions of the sources. We also have shown that MIBI is a kind of generalization of both IBSS and ISBSL, and we have given several application examples. In order to be able to identify the mixing system and/or recover the source signals, the apparently large lack of information needs to be compensated for by means of strong assumptions on the source signals. The 'main source signal assumption' underlying all blind methods is that the source signals are mutually statistically independent. We have argued that MIMO instantaneous blind or semi-blind problems can be tackled by processing multiple sensor signals because of the different kinds of diversity assumed to be present in the mixing system and source signals, viz. the spatial and statistical diversities respectively. Furthermore, we have stated that in this thesis our main focus will be on the development of a unified blind identification method for exploiting any kind of temporal structure in the source signals of any chosen order, such as (higher order) non-whiteness/correlatedness and non-stationarity. Finally, we have given several motivations for the work presented in this thesis and formulated problem statements pertaining to MIBI, IBSS, ISBSL, and an encompassing thesis problem statement.

CHAPTER 2

Theory, literature review and problem context

This chapter presents the theoretical background and a literature review of instantaneous blind signal processing, thereby allowing us to contextualize the work presented later on in this thesis w.r.t. other work. Different principles and approaches are identified and discussed; see also Chapter 1. Several overviews of the Blind Signal Processing (BSP) research area have already been published in the literature; for example, see [1, 24, 34, 47, 52, 54, 55, 61, 63, 64, 75, 79, 80, 83, 92, 128, 139, 158, 189], and the references therein. Although the known approaches to BSP are all based on the same *guiding principle of statistical independence*, nowadays the field has become so elaborate that it is virtually impossible to cover all details of all approaches in a single reasonably-sized volume, let alone in a single chapter. Therefore, most overviews focus on one specific approach and briefly mention the others. In our review, we also have to skip some approaches, but nevertheless we will try to present a coherent view of the field. Moreover, we attempt to add value to the existing overviews by stressing insight and relationships. Among other things, we hope to achieve this goal by describing the basic ideas in such a way that in principle they can be generalized to more complicated scenarios rather straightforwardly, even when this has not been done yet in the literature. The provided insight will be used during the development of our temporal structure based MIBI method later on in the thesis. We also hope that the presented generalizations and relationships will serve as starting points for further development and research.

The outline of this chapter is as follows. We start our review in Section 2.1 by presenting a brief survey of, and viewpoint on, the development of blind signal processing methods. Then, in Section 2.2 we discuss the most common model assumptions concerning the mixing system, source signals, and additive sensor noise signals. The most important of these will be elaborated on in later sections. Next, in Section 2.3 we categorize the approaches to BSP problems like MIBI, IBSS and ISBSL into two main categories that reflect our point of view and represent a framework for contextualizing our work. The first category consists of *methods that do not exploit any temporal structure* in the data and are based on two main assumptions, viz. mutual spatial statistical independence and non-Gaussianity of the source signals; see Sections 2.3.1 and 2.7.1. This category of methods is typically referred to as *Independent Component Analysis (ICA)*. The second category consists of *methods that do exploit temporal structure* in the data and are also based on two main assumptions, viz. mutual spatial statistical independence and the presence of a certain temporal structure in the data; see Sections 2.3.2 and 2.8. The method that will be presented in later chapters belongs to the latter category. In Section 2.4, two indeterminacies that are inherent to the MIBI data model in (1.1.1) are discussed. No matter what kind of approach or algorithm is used, these indeterminacies can never be resolved without more a priori knowledge than is usually assumed to be available. Then, in Section 2.5 we cover some general topics that are heavily used in BSP, such as objective functions, optimization methods, (stochastic) gradient ascent or descent methods, and standard versus natural gradient. After that, in Section 2.6 a preprocessing

stage that often is employed by BSP methods, viz. the so-called *whitening* stage, is explained. Subsequently, in Section 2.7 we discuss several aspects, identification/separation principles, and algorithms or methods belonging to the category that is based on exploiting spatial independence and non-Gaussianity. In Section 2.8 we do the same for the category of methods that is based on exploiting spatial independence and temporal structure. Finally, conclusions and discussion are presented in Section 2.9.

2.1 Overview of development of blind methods

The development of blind identification or separation methods for problems like MIBI, IBSS, and ISBSL involves different steps. First of all, several assumptions are made on the mixing system, source signals, and sensor noise signals. Different methods or approaches usually adopt different assumptions. Examples of commonly adopted assumptions are that the mixing matrix is square and full rank, that the source signals are statistically independent and non-Gaussian, that the sensor noise is temporally and spatially white, etc. Next, based on the assumptions, an identification or separation criterion is chosen. For example, in IBSS it may be desired to make (the absolute values of) the fourth order cross-cumulants of the estimated source signals as small as possible, or to make their cross-correlations as small as possible for different lags, etc. Departing from such a criterion, essentially two different approaches can be followed for estimating the mixing system, de-mixing system, or source signals. Firstly, an *objective function* (see Section 2.5) can be constructed in such a way that its minima or maxima correspond to a desired solution [52]. Secondly, a system of (nonlinear) equations can be constructed in such a way that its roots correspond to a desired solution. For the first approach, an optimization method or algorithm for optimizing the objective function has to be designed or chosen. Note that we make a clear distinction between an objective function and the method or algorithm used to optimize it. Loosely speaking, the development or composition of blind methods based on this approach can be expressed as follows:

$$\boxed{\text{Method} = \text{Assumptions} + \text{Objective function} + \text{Optimization algorithm}} \quad .$$

For the second approach, a method or algorithm for solving the system of equations has to be designed or chosen. Again, we make a clear distinction between the system of equations and the method or algorithm used to solve it. Loosely speaking, the development of blind methods based on this approach can be expressed as follows:

$$\boxed{\text{Method} = \text{Assumptions} + \text{System of equations} + \text{Root-finding algorithm}} \quad .$$

In case the system of equations (and involved statistics) is known accurately, it might be attempted to solve the system directly. Otherwise, again an objective function must be constructed the optima of which correspond to approximate solutions of the system, and we have a kind of 'indirect form' of the first approach. Hereafter, by MIBI, IBSS or ISBSL 'method' or 'algorithm', we mean all issues involved in a complete working identification or separation method. That is, such a method encompasses all assumptions, the criterion, objective function or system of equations, and the optimization or root-finding algorithm. In the next section, we start by describing several assumptions on the mixing system, source signals, and sensor noise signals, which are adopted by many methods. We will encounter some specific objective functions, systems of equations, optimization, and root-finding algorithms in later sections and chapters, where specific algorithms are discussed.

2.2 Model assumptions

There are many different methods and approaches for blind identification and separation. This is largely due to the different assumptions adopted by different methods. This section gives an overview of the most common assumptions. The most important ones will be elaborated on in later sections. The assumptions are divided into three different categories. The first contains assumptions on the mixing system, the second on the source signals, and the last on the noise signals (see also [54]). MIBI and IBSS methods adopt several assumptions from each category simultaneously. We will discuss each class of assumptions in turn, followed by a discussion of the general issues addressed by the assumptions in that class. *In this chapter, we will always assume that the source and noise signals are statistically independent.* For convenience of discussion, the considered MIBI model (1.1.1) is repeated here:

$$\mathbf{x}[n] = \sum_{j=1}^{S} \mathbf{a}^j \, s_j[n] + \boldsymbol{\nu}[n] = \mathbf{A}\mathbf{s}[n] + \boldsymbol{\nu}[n] \qquad \forall \, n \in \mathbb{Z} \, . \tag{2.2.1}$$

2.2.1 Assumptions on mixing system

The following list is representative of the most common assumptions on the mixing matrix $\mathbf{A}$:

ASM1: $\mathbf{A}$ *is square and invertible;*

ASM2: $\mathbf{A}$ *has more rows than columns, i.e. there are more sensors than sources;*

ASM3: $\mathbf{A}$ *has more columns than rows, i.e. there are more sources than sensors;*

ASM4: $\mathbf{A}$ *has full column rank;*

ASM5: $\mathbf{A}$ *has full row rank;*

ASM6: $\mathbf{A}$ *is allowed to be rank-deficient;*

ASM7: $\mathbf{A}$ *has columns with unit Euclidian norm;*

ASM8: $\mathbf{A}$ *has a known specific structure.*

As can be observed from this list, the assumptions on the mixing system cover issues concerning the size, rank, and possible structure of the mixing matrix $\mathbf{A}$. Assumptions *ASM1-ASM3* concern the size of $\mathbf{A}$, and in particular the relation between the number of sensors D and the number of sources S. In order to highlight some of the notation adopted in this work[‡], we have written $\mathbf{A}$ in terms of its elements, columns and rows respectively:

$$\mathbf{A} = \begin{bmatrix} a_1^1 & \cdots & a_1^S \\ \vdots & \ddots & \vdots \\ a_D^1 & \cdots & a_D^S \end{bmatrix} = \begin{bmatrix} \mathbf{a}^1 & \cdots & \mathbf{a}^S \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{a}}_1 \\ \vdots \\ \tilde{\mathbf{a}}_D \end{bmatrix} \quad \in \quad \mathbb{R}_D^S \left(\mathbb{C}_D^S \right) . \tag{2.2.2}$$

As we have seen already, in this work matrices are denoted by upper case boldface letters, and their elements are denoted by lower case letters with both sub- and superscript indices. The subscript indices correspond to row indices, whereas the superscript indices correspond to column indices. Column vectors are denoted by lower case boldface letters and their elements are indexed by subscript indices. Row vectors are denoted by lower case boldface letters with a tilde and their elements are indexed by superscript indices. Assumptions *ASM1*

[‡] See Appendix A for detailed information about notational issues.

and *ASM4-ASM6* concern the rank of $\mathbf{A}$. Together with the assumptions on the size, to a large extent they determine which MIBI, IBSS or ISBSL problem formulation is most suitable for a given practical problem. Most IBSS methods, where the interest by definition is in the source signals, assume that $\mathbf{A}$ is square and invertible. Hence, in this case it is assumed that there exists a separation matrix $\mathbf{W}$ (e.g. $\mathbf{A}^{-1}$) such that the components of $\mathbf{y}[n] \triangleq \mathbf{W}\mathbf{x}[n] = \mathbf{W}\mathbf{A}\mathbf{s}[n]$ (see Fig. 1.2 on page 5 and (1.1.2)) are estimates of the source signals, possibly up to some ambiguities; see Section 2.4. Practical IBSS algorithms try to estimate $\mathbf{W}$ from the observed sensor signals $\mathbf{x}[n]$ only by imposing the same assumptions on the output signals $\mathbf{y}[n]$ as the ones made on the source signals; see Section 2.2.2. The same approach works when there are more sensors than sources and $\mathbf{A}$ has full column rank because then there exists a left pseudo-inverse $\mathbf{A}^{\dagger}$ of $\mathbf{A}$ by which the source signals can be recovered. For MIBI and ISBSL, where the main interest is in the mixing system, the matrix $\mathbf{A}$ can be estimated directly from the observed sensor signals $\mathbf{x}[n]$ instead of first estimating its inverse $\mathbf{W}$. We note that the vast majority of blind methods deals with the conventional case $D \geq S$, whereas only a few methods consider the underdetermined case $D < S$. The main reason is that the latter case is typically considered to be much more difficult, see Sections 1.2.4 and 2.2.4. Assumption *ASM7* is a normalization constraint that is often imposed in MIBI and IBSS. As will be explained in Section 2.4, this is due to the *scaling indeterminacy* or *ambiguity* inherent to the MIBI model in (2.2.1), which means that the norm of the columns of the mixing matrix and the variances of the source signals cannot be determined from the sensor data. Finally, assumption *ASM8* deals with the case that $\mathbf{A}$ has a certain known specific structure. For example, in narrowband source localization the specific dependence of the mixing matrix, also known as array response matrix in this context, on the source position parameters is known because of the available a priori knowledge about the propagation medium characteristics and sensor positions. See Section 1.1.3 for examples and more information. Note that we call blind problems in which a priori knowledge about the structure of $\mathbf{A}$ is available 'semi-blind'.

2.2.2 Assumptions on source signals

The following list is representative of the most common assumptions made on the source signals $s_1[n], \ldots, s_S[n]$:

- *ASS1:* *The source signals are mutually statistically independent; for approaches based on statistics of order l, independence is understood to be at, or up to, order l only*[‡]*;*
- *ASS2:* *Each source signal is independently and identically distributed (i.i.d.), i.e. it is temporally strongly white and stationary;*
- *ASS3:* *At most one of the source signals is Gaussian;*
- *ASS4:* *The source signals have different kurtosis;*
- *ASS5:* *The source signals are stationary;*
- *ASS6:* *The source signals are non-stationary;*
- *ASS7:* *The source auto-correlation functions are linearly independent;*
- *ASS8:* *The l-th order source auto-cumulant functions are linearly independent;*

[‡] See Section 2.7.3.1 and Appendix B for more information about statistical independence and cross-cumulants.

ASS9: *The source signals have unit variance;*

ASS10: *The source auto-correlation functions are known;*

ASS11: *The source signals have a known l-th order temporal structure, e.g. the l-th order source auto-cumulant functions are known;*

ASS12: *The probability density functions of the source signals are known, or have a known property such as (a)symmetry;*

ASS13: *The source signals have other specific properties such as constant modulus, discrete probability distribution, etc.*

As can be observed from this list, the assumptions on the source signals cover several issues concerning the statistics, temporal structure, specific properties of the sources, etc. *ASS1* is the most fundamental assumption in blind signal processing and is exploited implicitly or explicitly by all identification and separation approaches. It concerns the relation between the different source signals and forms the guiding principle in blind signal processing; see *MAS1* on page 23. Although it is a strong assumption, *it is physically plausible in many practical situations because the source signals often are generated independently of each other by different sources*. Assumptions *ASS2*-*ASS13* concern the statistics of the individual source signals, i.e. the 'auto-statistics', and ensure that the source signals possess sufficient statistical diversity; see the MIBI, IBSS and ISBSL problem statements in Sections 1.1.1, 1.1.2 and 1.1.3 respectively, and *MAS2* on page 23. In addition to *ASS1*, all blind methods require at least one of the assumptions *ASS2*-*ASS8*. Note that *ASS7* is a specific example of *ASS8*. In Section 2.3, we will show that source assumptions in the spirit of *ASS2*-*ASS8* roughly divide the MIBI, IBSS and ISBSL methods into two main categories: one that *does not*, and one that *does* exploit the temporal structure present in the signals. The first category requires assumption *ASS3* to be satisfied, see Section 2.7, whereas the second relies on assumptions like *ASS6* and *ASS7*. Like *ASM7* in the previous section, assumption *ASS9* merely is a normalization constraint that often is imposed in MIBI and IBSS because of the scaling indeterminacy inherent to MIBI. In practical MIBI or IBSS, either *ASM7* or *ASS9* has to be chosen, i.e. either the norms of the columns of $\mathbf{A}$ or the variances of the source signals have to be fixed. Finally, assumptions *ASS10*-*ASS13* specify extra a priori information about the statistical structure of the source signals. For example, in some wireless communication systems, the correlation functions of the transmitted signals could be fixed and known at both the transmitter and receiver sides. Similarly to *ASM8* in the previous section, this kind of prior knowledge makes the identification or separation problem semi-blind.

2.2.3 Assumptions on noise signals

The following list is representative for the most common assumptions on the additive sensor noise signals $\nu_1[n], \ldots, \nu_D[n]$:

ASN1: *The noise signals are mutually statistically independent; the remark about the order l in ASS1 of Section 2.2.2 is also applicable here;*

ASN2: *Each noise signal is i.i.d., i.e. it is temporally strongly white and stationary;*

ASN3: *The noise signals have a 'simpler' temporal structure than the source signals in the sense explained in Section 1.2.5;*

ASN4: *The noise signals are Gaussian;*

ASN5: *The noise signals are stationary;*

ASN6: *The noise signals are non-stationary;*

ASN7: *The noise auto-correlation functions are known;*

ASN8: *The noise signals have a (partially) known l-th order temporal structure, e.g. the l-th order noise auto-cumulant functions are known;*

ASN9: *The noise signals are not mutually statistically independent, but the spatial statistics of the noise signals are (partially) known, e.g. the spatial correlation matrix $E\{\boldsymbol{\nu}[n]\boldsymbol{\nu}^H[n]\}$ is known a priori;*

ASN10: *The probability density functions of the noise signals are known, or have a known property such as (a)symmetry.*

All these assumptions are frequently encountered in many signal processing problems, and many more can be added to the list. On the one hand they concern the relation between different noise signals, and on the other hand their 'auto-statistics'. Some of them are more realistic than others, but the 'simplest' ones are adopted most often because they make a signal processing problem mathematically (more) tractable while still approximating some practical scenario reasonably well. For instance, *ASN1-ASN5* belong to this category. Different noise assumptions usually require different noise elimination or reduction approaches. Several of the assumptions above, e.g. *ASN1-ASN4*, ensure that the noise signals have a certain lack of statistical diversity compared to the source signals. For example, as we have argued in Section 1.2.5, when *ASS3* and *ASN4* are satisfied, the influence of noise on the mixing matrix estimation can be eliminated by using cumulants of order larger than two. Likewise, when *ASS7* and *ASN2* are satisfied, the influence of noise can be eliminated by considering only noise-free time or lag tuples (note that *ASN2* is reminiscent of *MAS4* in Section 1.2.8 and *ASN3*). Finally, assumptions *ASN7-ASN10* specify extra a priori information about the statistical structure of the noise signals that can be exploited.

2.2.4 Underdetermined blind problems

In Section 1.2.4, we have seen that most approaches to MIBI, IBSS and ISBSL cannot deal with underdetermined blind problems in which the sources outnumber the sensors. Only a few methods exist for this underdetermined case, and most of them are based on sparsity [18, 19, 96, 102] or/and non-Gaussianity [30, 56, 61]. Most underdetermined IBSS methods follow the two-stage approach described in Section 1.2.4, i.e. first the mixing matrix is identified and then the source signals are estimated. Different approaches for exploiting the sparsity of the source signals are the maximization of some (a posteriori) likelihood function, finding a sparse representation of the source signals in some proper signal dictionary, using a clustering algorithm, etc. [18, 100, 102, 153]. In this section, our purpose is to explain the main principles of the source recovery stage for underdetermined IBSS based on the sparsity assumption.

In general, sparsity based underdetermined IBSS source recovery methods assume that at most points in some appropriate representation domain the number of sources that differs significantly from zero is smaller than or equal to the number of sensors D. If the sensor data is not sufficiently sparse, sometimes it is possible to transform the mixtures linearly and reversibly into a domain with improved sparsity, in which the resulting signals are separated and then transformed back to the original domain. Sparse sources are usually modeled by

random variables with high *kurtosis*[‡]. This means that the source signals are strongly *Super-Gaussian*, also known as *leptokurtic*. Consequently, they can be modeled by probability density functions with large peaks centered at zero. For instance, a commonly assumed source probability density function (pdf) is the Laplacian pdf. In the sequel of this section, we assume that all source signals have the same *known* pdf $p_s(s)$ for all considered time samples; hence, $p_{s_j}(s_j) = p_s(s) = p_{s[n]}(s[n])\ \forall 1 \leq j \leq S,\ \forall n$ and the source *joint probability density function (jpdf)* $p_{\mathbf{s}[n]}(\mathbf{s}[n]) = p_{\mathbf{s}}(\mathbf{s})$ of the signals $s_1[n], \ldots, s_S[n]$ factorizes. Now, we reconsider the problem described at the end of Section 1.2.4, i.e. recovering the source signals from the estimated mixing matrix and sensor signals for a set of time indices Ω. For simplicity, the noise is neglected, i.e. we consider the system $\{\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]\}_{n\in\Omega}$ instead of $\{\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n] + \boldsymbol{\nu}[n]\}_{n\in\Omega}$. At first, we consider one subsystem $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$ for a specific value of n and leave out the time index, i.e. we consider the linear system of equations $\mathbf{x} = \hat{\mathbf{A}}\mathbf{s}$. As we have said in Section 1.2.4, the solution of $\{\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]\}_{n\in\Omega}$ yields an affine vector space of dimension at least $S - D$ and has no unique solution. Hence, an additional criterion is required for selecting a specific solution. One such a criterion can be derived by maximizing the a posteriori (j)pdf $p_{\mathbf{s}|\mathbf{x};\hat{\mathbf{A}}}(\mathbf{s}|\mathbf{x};\hat{\mathbf{A}})$ of $\mathbf{s}$, where $\hat{\mathbf{A}}$ and $\{\mathbf{x}[n]\}_{n\in\Omega}$ are considered as known parameters (see Section B.2.2 and equation (B.2.9)). From the maximum-likelihood approach and Bayes' theorem, the following method for estimating the source signals at a specific time instant n can be derived (see also Eq. (1.2.1) from Chapter 1, and [102, 153]):

$$\hat{\mathbf{s}} = \underset{\mathbf{s}\text{ s.t. }\mathbf{x}=\hat{\mathbf{A}}\mathbf{s}}{\operatorname{argmax}}\ p_{\mathbf{s}|\mathbf{x};\hat{\mathbf{A}}}(\mathbf{s}|\mathbf{x};\hat{\mathbf{A}}) = \underset{\mathbf{s}\text{ s.t. }\mathbf{x}=\hat{\mathbf{A}}\mathbf{s}}{\operatorname{argmax}}\ p_{\mathbf{x}|\mathbf{s};\hat{\mathbf{A}}}(\mathbf{x}|\mathbf{s};\hat{\mathbf{A}})p_{\mathbf{s}}(\mathbf{s}) = \underset{\mathbf{s}\text{ s.t. }\mathbf{x}=\hat{\mathbf{A}}\mathbf{s}}{\operatorname{argmax}}\ p_{\mathbf{s}}(\mathbf{s})\,. \quad (2.2.3)$$

Hence, the general method for the source-recovery stage in IBSS is the maximization of the jpdf $p_{\mathbf{s}}(\mathbf{s})$ under the constraint $\mathbf{x} = \hat{\mathbf{A}}\mathbf{s}$. This is a linear optimization problem that can be tackled using various optimization algorithms [108, 115]. In order to further interpret this result, we consider several source prior pdf's.

1) Firstly, assume that the source prior pdf is *Laplacian*, i.e.:

$$p_{\mathbf{s}}(\mathbf{s}) = \alpha \exp(-\left\|\mathbf{s}\right\|_1 / \beta)\,, \quad (2.2.4)$$

where α and β are fixed positive constants and $\left\|\mathbf{s}\right\|_1 \triangleq \sum_{j=1}^{S} |s_j|$ is the 1-norm of $\mathbf{s}$. This prior pdf is characteristic of sparse signals. For the general problem with all time samples $n \in \Omega$, using (2.2.4) in (2.2.3) yields:

$$\begin{aligned}\hat{\mathbf{s}}[n] &= \underset{\mathbf{s}[n]\text{ s.t. }\mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmax}}\ p_{\mathbf{s}[n]}(\mathbf{s}[n]) = \underset{\mathbf{s}[n]\text{ s.t. }\mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmax}}\ \exp\big(-\left\|\mathbf{s}[n]\right\|_1\big) \\ &= \underset{\mathbf{s}[n]\text{ s.t. }\mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmin}}\ \left\|\mathbf{s}[n]\right\|_1 \qquad \forall\ n \in \Omega\,. \end{aligned} \quad (2.2.5)$$

Hence, for a Laplacian source prior pdf, estimating the source signals according to (2.2.3) is equivalent to minimizing the 1-norm $\left\|\mathbf{s}[n]\right\|_1$ of $\mathbf{s}[n]$ subject to the constraint(s) $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$. As has been shown in [18] by Bofill and Zibulevsky, this minimization has a nice visual interpretation, which they call *shortest-path decomposition*. If the columns $\{\hat{\mathbf{a}}^j\}^{1\leq j\leq S}$ of $\hat{\mathbf{A}}$ are normalized, the optimal representation of the data point $\mathbf{x}[n]$ in the 'overcomplete basis' (frame) $\{\hat{\mathbf{a}}^j\}^{1\leq j\leq S}$ that minimizes $\left\|\mathbf{s}[n]\right\|_1$ s.t. $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$ will include at most D of the columns corresponding to the *minimal simplex* enclosing the direction of $\mathbf{x}[n]$, thereby obtaining a sparse representation of the source signals [18]. The non-zero components of the

[‡]See Section B.1.3.3 for the definition and meaning of statistical terms like *skewness, kurtosis, Sub-Gaussian, Super-Gaussian, platykurtic, mesokurtic, and leptokurtic.*

optimal decomposition correspond to the shortest path from the origin to the data point when only the directions of the columns of the mixing matrix may be used [18, 152]. Note that $\mathbf{s}[n]$ must have the same sparsity structure for each value of n, i.e. the zero components must be in the same position for all $n \in \Omega$. The problem in (2.2.5) is closely related to overcomplete signal representation and best basis selection problems [96, 102, 133]. It can be shown that the solutions of (2.2.5), i.e. the source estimates, are unique under mild conditions [152].

2) Secondly, assume that the source prior density is *Gaussian*, i.e.:

$$p_{\mathbf{s}}(\mathbf{s}) = \alpha \exp\big(-(\|\mathbf{s}\|_2)^2/\beta\big)\,, \tag{2.2.6}$$

where α and β are fixed positive constants and $\|\mathbf{s}\|_2 \triangleq \sqrt{\sum_{j=1}^{S} |s_j|^2}$ is the 2-norm (Euclidian norm) of $\mathbf{s}$. This prior pdf does not properly represent sparse data, but our purpose for considering it is to obtain insight by using it in (2.2.3). Following the same reasoning as in the previous paragraph, a result similar to (2.2.5) is obtained for the general problem with all time samples $n \in \Omega$:

$$\hat{\mathbf{s}}[n] = \underset{\mathbf{s}[n] \text{ s.t. } \mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmin}} \big(\|\mathbf{s}[n]\|_2\big)^2 = \underset{\mathbf{s}[n] \text{ s.t. } \mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmin}} \|\mathbf{s}[n]\|_2 \qquad \forall\, n \in \Omega\,. \tag{2.2.7}$$

Hence, for a Gaussian source prior pdf, estimating the source signals according to (2.2.3) is equivalent to minimizing the 2-norm $\|\mathbf{s}[n]\|_2$ of $\mathbf{s}[n]$ subject to the constraint(s) $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$. For a single time index n, this corresponds to finding the vector $\mathbf{s}[n]$ with minimal Euclidian length in the S-dimensional Euclidian space from the origin to the hyperplane representing the set of solutions of the system $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$ for $\mathbf{s}[n]$. The solution of this well-known problem is given by $\hat{\mathbf{s}}[n] = \hat{\mathbf{A}}^{\dagger}\mathbf{x}[n]$, where $\hat{\mathbf{A}}^{\dagger} = \hat{\mathbf{A}}^H(\hat{\mathbf{A}}\hat{\mathbf{A}}^H)^{-1}$ is the pseudo-inverse of $\hat{\mathbf{A}}$. Thus, if $\mathbf{A}$ has full rank the solution exists and is unique. Note that contrary to the situation with the Laplacian prior, the recovery of the sources now is a linear operation on the sensor data.

3) Finally, we generalize the previous results to source prior pdf's that are members of the so-called *exponential power family*. For a positive real-valued power p, which determines the type of distribution, the corresponding element of the exponential power family is given by:

$$p_{\mathbf{s}}(\mathbf{s}) = \alpha \exp\big(-(\|\mathbf{s}\|_p)^p/\beta\big)\,, \tag{2.2.8}$$

where α and β are fixed positive constants and $\|\mathbf{s}\|_p \triangleq \big(\sum_{j=1}^{S} |s_j|^p\big)^{\frac{1}{p}}$ is the p-norm of $\mathbf{s}$. Again following the same reasoning as in the previous paragraphs, a result similar to (2.2.5) and (2.2.7) is obtained for the general problem with all time samples $n \in \Omega$:

$$\hat{\mathbf{s}}[n] = \underset{\mathbf{s}[n] \text{ s.t. } \mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmin}} \big(\|\mathbf{s}[n]\|_p\big)^p = \underset{\mathbf{s}[n] \text{ s.t. } \mathbf{x}[n]=\hat{\mathbf{A}}\mathbf{s}[n]}{\operatorname{argmin}} \|\mathbf{s}[n]\|_p \qquad \forall\, n \in \Omega\,. \tag{2.2.9}$$

Hence, *for the exponential source prior pdf* (2.2.8) *with power p, estimating the source signals according to* (2.2.3) *is equivalent to minimizing the p-norm $\|\mathbf{s}[n]\|_p$ of $\mathbf{s}[n]$ subject to the constraint(s)* $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$. For a single time index n, this corresponds to finding the vector $\mathbf{s}[n]$ with minimum p-norm in the S-dimensional Euclidian space from the origin to the hyperplane representing the set of solutions of the system $\mathbf{x}[n] = \hat{\mathbf{A}}\mathbf{s}[n]$ for $\mathbf{s}[n]$. Depending on the considered norm, $\hat{\mathbf{s}}[n]$ may or may not be unique. For instance, for the supremum norm ($p = \infty$), it is non-unique.

2.3 Temporal versus no temporal structure

Categorizing blind approaches

As has been stated in Section 2.2.2, all MIBI methods require *ASS1* about the mutual statistical independence (possibly at, or up to, a specific order l) of the source signals to be satisfied. In addition, at least one of the assumptions *ASS2-ASS8* is required. Depending on whether these require the source signals to posses temporal structure or not, two categories of blind approaches can be distinguished. We will first give an overview of the category that does not exploit any temporal structure in the data and introduce a simplified notation for it without time indices. After that, we consider the category that does exploit the temporal structure. The purpose of this section is to give an overview of both categories; therefore, details are deferred to later sections.

2.3.1 Mutual statistical independence and non-Gaussianity: ICA

Blind methods that are not based on exploiting temporal structure in the data are mainly founded upon *ASS1-ASS5.* They either neglect any temporal structure, or assume it to be absent (e.g. *ASS2*), and at least require *ASS3* in addition to *ASS1*; see also Section 2.7. In other words, the lack or negligence of any temporal structure in the data needs to be compensated for by the assumption that the source signals are non-Gaussian, possibly with the exception of one source. This implies that Higher Order Statistics (HOS) have to be used. Hence, we can conclude that *methods that do not exploit any temporal structure in the data are based on two main assumptions: the mutual (spatial) statistical independence and non-Gaussianity of the source signals*. This category of methods is typically referred to as *Independent Component Analysis (ICA)*. ICA can be considered as a class of methods for performing MIBI; in this context, the source signals are called 'independent components'. Formally, ICA is often defined as the linear decomposition of a random vector into components that are as independent as possible [80].

Since ICA does not exploit any temporal structure in the signals, and the signals are assumed to be stationary, the time dependence of the signals is not important. Therefore, all time indices are usually dropped and any time signal $v[n]$ is considered as a random variable v instead of a time signal or time series; the observed values $v[n]$ are then (temporal or spatial) samples of this random variable. Hence, the ICA equivalent of model (2.2.1), which is the same as (1.1.1), is given by:

$$\mathbf{x} = \sum_{j=1}^{S} \mathbf{a}^j \, s_j + \boldsymbol{\nu} = \mathbf{A}\mathbf{s} + \boldsymbol{\nu} \,, \tag{2.3.1}$$

where $\mathbf{x}$, $\mathbf{s}$ and $\boldsymbol{\nu}$ are considered as random vectors, and $\mathbf{x}[n]$, $\mathbf{s}[n]$ and $\boldsymbol{\nu}[n]$ respectively as samples or realizations of them. The *marginal probability density functions (mpdf's)* of the components $v_1, \ldots, v_G$ of a length-G random vector $\mathbf{v}$ are denoted by $p_{v_1}(v_1)$, …, $p_{v_S}(v_S)$, and their *joint probability density function (jpdf)* by $p_{\mathbf{v}}(\mathbf{v}) = p_{v_1,\ldots,v_S}(v_1, \ldots, v_S)$[‡]. For example, the mpdf's of the source signals are denoted by $p_{s_1}(s_1)$, …, $p_{s_S}(s_S)$ and their jpdf by $p_{\mathbf{s}}(\mathbf{s}) = p_{s_1,\ldots,s_S}(s_1, \ldots, s_S)$. The fundamental assumption of mutual (spatial) statistical

[‡] See Section B.2.1 for the definition and properties of pdf, mpdf, jpdf, mjpdf, etcetera.

independence of the source signals can now be written mathematically as follows:

$$p_{\mathbf{s}}(\mathbf{s}) = \prod_{j=1}^{S} p_{s_j}(s_j)\,, \tag{2.3.2}$$

i.e. the jpdf factorizes into the mpdf's; see (B.2.10). From this condition, several important properties can be derived that can be used for blind identification or separation. In the literature, the goal of ICA often is formulated as follows:

> ***Independent Component Analysis (ICA) problem statement:***
> The purpose of ICA is to find a linear transformation that transforms a given random vector into another random vector the components of which are as statistically independent and non-Gaussian as possible.

In the current context, the purpose of ICA is to find a matrix $\mathbf{W}$ that transforms the random vector $\mathbf{x}$ of observations into the random vector $\mathbf{y} \triangleq \mathbf{W}\mathbf{x}$ the components of which are as statistically independent as possible. Most ICA methods assume that the mixing matrix is square and invertible (see *ASM1*). This implies that there exists a de-mixing matrix $\mathbf{W}$ such that the output vector:

$$\mathbf{y} \triangleq \hat{\mathbf{s}} = \mathbf{W}\mathbf{x} \tag{2.3.3}$$

contains scaled and permuted estimates of the source signals; see also (2.4.4). *ICA methods try to recover/estimate the source signals by imposing the same assumptions on the output (i.e. estimated source) signals as the ones made on the source signals.* For example, because the source signals are assumed to be mutually (spatially) statistically independent, the same independence condition should be imposed on the output signals:

$$p_{\mathbf{y}}(\mathbf{y}) = \prod_{j=1}^{S} p_{y_j}(y_j)\,. \tag{2.3.4}$$

As we have stated in the introduction to this section, we will show in Section 2.7 why methods for performing ICA require the source signals to be non-Gaussian. There, we will also discuss actual ICA methods for exploiting mutual (spatial) independence and non-Gaussianity. See also [26, 27, 29, 38, 47, 50, 51, 55, 75, 80, 91] and the references therein.

2.3.2 Temporal structure

Blind methods that are based on exploiting temporal structure in the data usually are founded upon one or more of assumptions *ASS1* and *ASS5-ASS8.* These assumptions concern certain types of temporal structure, e.g. auto-correlation and non-stationarity, and diversity, e.g. linear independence of source auto-correlation functions, in the data. Many algorithms are based on one or more of them; see Section 2.8 and references [14, 46, 86, 87, 106, 107, 157, 158, 185] for examples. When designing algorithms for blind methods that do exploit temporal structure, the signals involved in data model (2.2.1) usually are modeled as realizations of random processes with certain marginal and joint probability density functions [154]. The marginal and joint probability density functions of the signals should incorporate information about the temporal structure. As an example, consider the S source signals $s_1[n], \ldots, s_S[n]$ at a specific time (index) n. Their marginal joint probability density functions are denoted by $p_{s_1[n]}(s_1[n]), \ldots, p_{s_S[n]}(s_S[n])$. This notation reflects the fact that in general the signals are allowed to be non-stationary, i.e. have statistics that change in time (as we have

seen in the previous section, for stationary signals the time indices could be omitted in principle). The temporal relation between different samples of the same signal can be modelled using a jpdf with different time indices. Another possibility is to use one time index and several 'lag indices' (for stationary signals, lag indices only are sufficient). For example, the jpdf that models the relation between l samples of the j-th source signal is denoted by $p_{s_j[n_1],\ldots,s_j[n_l]}(s_j[n_1],\ldots,s_j[n_l])$, where $n_1,\ldots,n_l$ are arbitrarily chosen time indices. A similar notation is used to model the joint temporal statistics of two or more signals. For example, the jpdf of all sources considered at different times is denoted by $p_{s_1[n_1],\ldots,s_S[n_S]}(s_1[n_1],\ldots,s_S[n_S])$. The fundamental assumption of mutual statistical independence of the source signals now not only is to be understood in the spatial sense, but also in the temporal sense. Hence, similarly to (2.3.2), this assumption can be written as follows:

$$p_{s_1[n_1],\ldots,s_S[n_S]}(s_1[n_1],\ldots,s_S[n_S]) = \prod_{j=1}^{S} p_{s_j[n_j]}(s_j[n_j]) \quad \forall\, n_1,\ldots,n_S \in \mathbb{Z}\,, \tag{2.3.5}$$

i.e. the jpdf of the sources factorizes into the mpdf's of the sources. For any (sub)set of $1 \leq l \leq S$ different indices $1 \leq j_1 \neq j_2 \neq \cdots \neq j_l \leq S$, a similar result holds:

$$p_{s_{j_1}[n_{j_1}],\ldots,s_{j_l}[n_{j_l}]}(s_{j_1}[n_{j_1}],\ldots,s_{j_l}[n_{j_l}]) = \prod_{k=1}^{l} p_{s_{j_k}[n_{j_k}]}(s_{j_k}[n_{j_k}]) \quad \forall\, n_{j_1},\ldots,n_{j_l} \in \mathbb{Z}\,. \tag{2.3.6}$$

This condition is of paramount importance for all Blind Signal Processing (BSP) problems, and several important properties on which many MIBI methods are based can be derived from it; see Appendix B and Section 2.7.3.

In practice, virtually all blind algorithms that exploit temporal structure are based on Second Order Statistics (SOS). For example, see [14, 42, 46, 71, 85–87, 106, 107, 114, 157, 158, 177, 178, 185]. For SOS, assumption *ASS1* of Section 2.2.2 should be interpreted as mutual statistical independence of the sources at order 2 only. In other words, the different source signals are assumed to be uncorrelated with each other. This uncorrelatedness has two different aspects. On the one hand, the source signals are uncorrelated in the *spatial* sense; this emphasizes the fact that different source signals, which are generated at different positions, are uncorrelated. On the other hand, the source signals are uncorrelated in the *temporal* sense; this emphasizes the fact that different sources are uncorrelated for all possible combinations of time indices. These two aspects are intertwined and can be formulated as one statement:[‡]

$$r_{ij}^{s}[m,n] \triangleq E\left\{s_i[m]s_j[n]\right\} = 0 \qquad \forall\, 1 \leq i \neq j \leq S, \quad \forall\, m,n \in \mathbb{Z}\,. \tag{2.3.7}$$

The symbol $r_{ij}^{s}[m,n]$ at the left hand side denotes the correlation between $s_i[m]$ and $s_j[n]$[‡]. The indices i and j in the subscript position represent the components of the involved signal vector between which the correlation is computed. The superscript symbol 's' indicates which signal is under consideration; in the current case it stands for 'source'[‡]. Furthermore, $E\{\cdot\}$ denotes the mathematical expectation operator[§]. Theoretically, $r_{ij}^{s}[m,n]$ can be computed from the marginal joint probability density function $p_{s_i[n],s_j[m]}(s_i[n],s_j[m])$ or the 'full' joint probability density function $p_{s_1[n_1],\ldots,s_S[n_S]}(s_1[n_1],\ldots,s_S[n_S])$. In practice, it is usually estimated directly from the data since this is much easier than first estimating the

[‡] See Appendix A for the notation of correlation and cumulant functions.

[§] See Appendix B for the notation and definition of statistical operators and quantities.

jpdf (which may even be impossible). In Section 2.8, we will see how SOTS property (2.3.7) of the source signals can be exploited to estimate the (de-)mixing system. This property can easily be generalized to HOTS by means of cumulant functions; see Chapter 6 and Appendix B. In spite of this, as we have discussed in Chapter 1, blind algorithms that do exploit the Higher Order Temporal Structure (HOTS) in the data are rare. However, in later chapters we will show that it is possible to do so by means of a unifying method; see also [163] and [164].

2.4 Model indeterminacies

This section discusses the indeterminacies that are inherent to the MIBI data model in (1.1.1) or (2.2.1), and which no matter what kind of approach or algorithm is used can never be resolved without more a priori knowledge than is usually assumed to be available in MIBI, IBSS and ISBSL. Examination of the observation model reveals two *indeterminacies* or ambiguities involved in instantaneous blind identification and separation. Firstly, it is clear that permuting the columns of $\mathbf{A}$ together with an equal permutation of the source signals in $\mathbf{s}[n]$ still yields the same vector of sensor signals $\mathbf{x}[n]$. For example, consider the situation with two source signals, i.e. $S = 2$:

$$\mathbf{x}[n] = \left(\mathbf{a}^1\, s_1[n] + \mathbf{a}^2\, s_2[n]\right) + \boldsymbol{\nu}[n] = \left(\mathbf{a}^2\, s_2[n] + \mathbf{a}^1\, s_1[n]\right) + \boldsymbol{\nu}[n]\,. \tag{2.4.1}$$

This means that without additional prior information, the order of the columns, respectively sources, cannot be determined. This ambiguity is typically called *order indeterminacy*. Secondly, as is also clear from (1.1.1), scaling the columns of $\mathbf{A}$ by arbitrary real- or complex-valued scalars, together with a corresponding inverse scaling of the source signals in $\mathbf{s}[n]$, again yields the same vector of sensor signals $\mathbf{x}[n]$. For example, if the j-th source is multiplied by a scalar $\alpha^j \neq 0$ for all $1 \leq j \leq S$ with $S = 2$, we have:

$$\mathbf{x}[n] = \left(\mathbf{a}^1\, s_1[n] + \mathbf{a}^2\, s_2[n]\right) + \boldsymbol{\nu}[n] = \left(\frac{\mathbf{a}^1}{\alpha^1}\left(\alpha^1 s_1[n]\right) + \frac{\mathbf{a}^2}{\alpha^2}\left(\alpha^2 s_2[n]\right)\right) + \boldsymbol{\nu}[n], \tag{2.4.2}$$

and in the general case:

$$\mathbf{x}[n] = \left(\sum_{j=1}^{S} \mathbf{a}^j\, s_j[n] + \boldsymbol{\nu}[n]\right) + \boldsymbol{\nu}[n] = \left(\sum_{j=1}^{S} \frac{\mathbf{a}^j}{\alpha^j}\left(\alpha^j s_j[n]\right)\right) + \boldsymbol{\nu}[n]\,. \tag{2.4.3}$$

This implies that without additional prior information, the columns and sources can only be recovered up to a scale factor. This ambiguity is typically called *scaling indeterminacy*, or, when the emphasis is on the columns, *norm indeterminacy*.

Taking into account these two indeterminacies, the goal of MIBI and ISBSL is to recover the columns of the mixing system in arbitrary order and with arbitrary non-zero (Euclidian) norms. In terms of matrices this can be formulated as follows. An estimate $\hat{\mathbf{A}}$ of the mixing system should satisfy $\hat{\mathbf{A}} \approx \mathbf{APD}$, where $\mathbf{P} \in \mathbb{R}_S^S$ is some permutation matrix and $\mathbf{D} \in \mathbb{R}_S^S\left(\mathbb{C}_S^S\right)$ some nonsingular diagonal matrix. In practice, either the columns of the estimated mixing system $\hat{\mathbf{A}}$ are normalized to unit Euclidian norm, e.g. see *ASM7* on page 31, or the estimated source signals are normalized to unit variance, e.g. see *ASS9* on page 33. Note that this normalization fixes the variances of the source signals, but not their signs. For most MIBI and ISBSL applications the order and scaling indeterminacies usually do not cause serious problems because the most relevant information is in the 'directions' of the columns rather than in their order or norms. Similar to MIBI and ISBSL, taking into account the two

indeterminacies described above, the goal of IBSS is to recover the source signals in some arbitrary order and with some arbitrary nonsingular scaling or variances. Hence, an estimate $\hat{\mathbf{s}}[n]$ of the source signal vector should satisfy:

$$\hat{\mathbf{s}}[n] \approx \mathbf{PDs}[n]\,, \tag{2.4.4}$$

where $\mathbf{P}$ and $\mathbf{D}$ have the same meaning as above. Such an estimate is said to be a *waveform preserving solution* [156, 157]. Equivalently, the de-mixing matrix $\mathbf{W}$ in (1.1.2) should satisfy:

$$\mathbf{T} \triangleq \mathbf{WA} \approx \mathbf{PD}\,, \tag{2.4.5}$$

where $\mathbf{T}$ denotes the total transfer (matrix) from sources to outputs. Note that choosing $\mathbf{W} = \mathbf{PD}\hat{\mathbf{A}}^{-1}$ for a square mixing system yields a 'correct', i.e. waveform preserving, solution. Also for most IBSS applications the order and scaling indeterminacies usually do not cause serious problems because the most relevant information about the source signals is contained in their waveforms rather than in their order or variances.

2.5 Objective and contrast functions, optimization methods

The assumptions listed in Section 2.2, and in particular the ones in Section 2.2.2, provide clues about the way the source signals and/or (de-)mixing matrix should be recovered in MIBI, IBSS and ISBSL. For IBSS and ICA it is clear that the de-mixing matrix $\mathbf{W}$ should be estimated in such a way that the output signals $\mathbf{y}$ satisfy the same assumptions as the source signals $\mathbf{s}$. This can be done by optimizing a so-called 'objective function'. *An objective function is a real-valued function that is constructed in such a way that it penalizes deviations from the desired properties/assumptions, and attains its optimum(a) for the correct solution(s).* For instance, in ICA often an objective function $J(\mathbf{W}, \{\mathbf{y}\})$ is constructed that is a function of the de-mixing system $\mathbf{W}$ and the sequence $\{\mathbf{y}\}$ of output signal vector samples, and penalizes deviations of the components of $\mathbf{y} = \mathbf{Wx}$ from statistical independence and non-Gaussianity [80]. This function is then optimized as a function of $\mathbf{W}$. Hence, it should be designed in such a way that it attains its minima or maxima at 'separating solutions' of the form $\mathbf{W} = \mathbf{PD}\hat{\mathbf{A}}^{-1}$ with $\mathbf{P} \in \mathcal{P}_S^S$ and $\mathbf{D} \in \mathcal{D}_S^S$, where $\mathcal{P}_S^S$ and $\mathcal{D}_S^S$ represent the sets of all nonsingular permutation and diagonal matrices of size S by S respectively. When the separating solutions are at the minima of the objective function, it is also called *cost function*. In general, objective functions should be insensitive to the model indeterminacies discussed in Section 2.4. Often, for convenience of notation an objective or cost function is written as a function of the output signal $\mathbf{y}$ only. The desired properties of an objective/cost function $J(\mathbf{y})$ suitable for blind signal processing can be formulated as follows [1]:

- *JP1:* $J(\mathbf{y})$ *is invariant to scaling:* $J(\mathbf{y}) = J(\mathbf{Dy})$ *for all* $\mathbf{D} \in \mathcal{D}_S^S$*;*
- *JP2:* $J(\mathbf{y})$ *is invariant to permutation:* $J(\mathbf{y}) = J(\mathbf{Py})$ *for all* $\mathbf{P} \in \mathcal{P}_S^S$*;*
- *JP3:* *If* $\mathbf{y}$ *has statistically independent components, then:* $J(\mathbf{y}) \leq J(\mathbf{By})$ *for all nonsingular* $\mathbf{B} \in \mathbb{R}_S^S$ $\left(\mathbb{C}_S^S\right)$*, with equality if and only if* $\mathbf{B}$ *is of the form* $\mathbf{PD}$ *with* $\mathbf{P} \in \mathcal{P}_S^S$ *and* $\mathbf{D} \in \mathcal{D}_S^S$*.*

If the separating solutions are maxima of $J(\mathbf{y})$, the statement $J(\mathbf{y}) \leq J(\mathbf{By})$ in *JP3* should be replaced by $J(\mathbf{y}) \geq J(\mathbf{By})$. Ideally, the entire set of minima/maxima of $J(\mathbf{y})$ should coincide with the set of separating solutions. However, in practice, this often is not the case and the set of separating solutions forms a proper subset of the set of minima/maxima. This implies that a decision has to be made about which solutions should be selected.

A concept that is closely related to the notion of objective function is the so-called *contrast function*. This is a real-valued function of the joint probability density function $p_{\mathbf{v}}(\mathbf{v})$ of a random vector $\mathbf{v}$ that satisfies certain properties and achieves its maximum when $p_{\mathbf{v}}(\mathbf{v})$ factorizes into marginal pdf's, i.e. when the components of $\mathbf{v}$ are statistically independent; see Appendix B. In the IBSS or ICA context usually $\mathbf{v} = \mathbf{y}$, i.e. the (jpdf $p_{\mathbf{y}}(\mathbf{y})$ of the) output signal vector $\mathbf{y}$, is considered. Contrast functions were introduced into ICA by Pierre Comon who provided a solid mathematical basis for the ICA problem [52]. Essentially, he defined them in the following way:

Definition 2.5.1. Contrast function. Let $\mathbb{E}_G$ be the space of length-G random vectors that admit a jpdf. Then, a contrast function $\Xi(p_{\mathbf{v}})$ is defined as a mapping from the set $\{p_{\mathbf{v}} \mid \mathbf{v} \in \mathbb{E}_G\}$ of jpdf's to $\mathbb{R}$, which satisfies the following three requirements:

CFP1: $\Xi(p_{\mathbf{v}})$ *is invariant to scaling:* $\Xi(p_{\mathbf{v}}) = \Xi(p_{\mathbf{Dv}})$ *for all* $\mathbf{D} \in \mathcal{D}_S^S$*;*

CFP2: $\Xi(p_{\mathbf{v}})$ *is invariant to permutation:* $\Xi(p_{\mathbf{v}}) = \Xi(p_{\mathbf{Pv}})$ *for all* $\mathbf{P} \in \mathcal{P}_S^S$*;*

CFP3: If $\mathbf{v}$ *has statistically independent components, then:* $\Xi(p_{\mathbf{Bv}}) \leq \Xi(p_{\mathbf{v}})$ *for all nonsingular* $\mathbf{B} \in \mathbb{R}_S^S$ $\left(\mathbb{C}_S^S\right)$.

In addition, Comon makes the following definition:

Definition 2.5.2. . A contrast function is said to be *discriminating* over a set $\mathbb{E}_G$ if the equality $\Xi(p_{\mathbf{v}}) = \Xi(p_{\mathbf{Bv}})$, where $\mathbf{v}$ has statistically independent components and $\mathbf{B} \in \mathbb{R}_S^S$ $\left(\mathbb{C}_S^S\right)$ is nonsingular, holds if and only if $\mathbf{B}$ is of the form $\mathbf{PD}$ with $\mathbf{P} \in \mathcal{P}_S^S$ and $\mathbf{D} \in \mathcal{D}_S^S$.

Note that this definition is tantamount to the latter part of property *JP3.* Contrast functions are most frequently encountered in information-theoretic approaches to MIBI and IBSS, see Section 2.7.2.

The actual optimization of an objective function can be performed using known techniques, such as gradient descent or ascent, conjugate gradient, Newton's method, RLS-like algorithms, fixed point methods, and genetic algorithms. We will briefly recapitulate the *gradient descent method* because it is most commonly used in blind signal processing. The other methods will not be treated here, but are described extensively in the literature, for example see [108,109,115]. Suppose that a given cost function $J(\mathbf{v})$ has to be minimized as a function of its vector-valued argument $\mathbf{v}$ (the argument can also be a scalar or matrix, or just a set of arguments). Then, given the current estimate $\mathbf{v}[k]$, the new update $\mathbf{v}[k+1]$ is computed as follows:

$$\mathbf{v}[k+1] = \mathbf{v}[k] - \mu[k] \frac{\partial J(\mathbf{v})}{\partial \mathbf{v}}\Big|_{\mathbf{v}=\mathbf{v}[k]}, \tag{2.5.1}$$

where $\mathbf{v}[0]$ is any initial-parameter vector and $\{\mu[k]\}_{k=0,1,\ldots}$ is a non-negative sequence of step sizes. The vector $-\frac{\partial J(\mathbf{v})}{\partial \mathbf{v}}\Big|_{\mathbf{v}=\mathbf{v}[k]}$ represents the steepest-descent direction of the cost function at $\mathbf{v}[k]$. The gradient *ascent* method used for maximizing an objective function is completely similar to (2.5.1). The only difference is that the minus sign is replaced by a plus sign because now the current estimate should be updated in the direction of steepest-ascent.

In many optimization problems, the underlying parameter space is not Euclidian, but curved, i.e. Riemannian. For instance, this is the case for the space of square matrices subject to some constraint, such as having unit determinant, where another kind of gradient is more suitable for parameter adaptation because it provides a significantly improved convergence. This so-called *natural gradient* is based on differential geometry and employs knowledge of

the Riemannian structure of the parameter space to adjust the gradient search direction. The natural gradient update is performed as follows:

$$\mathbf{v}[k+1] = \mathbf{v}[k] - \mu[k]\mathbf{G}^{-1}(\mathbf{v}[k])\,\frac{\partial J(\mathbf{v})}{\partial \mathbf{v}}\Big|_{\mathbf{v}=\mathbf{v}[k]}, \tag{2.5.2}$$

where $\mathbf{G}^{-1}(\mathbf{v}[k])$ is the Riemannian metric tensor for the parameter vector $\mathbf{v}$, as defined by the manifold of parameters. See [4] for more information and details.

Often, an objective or contrast function is defined in terms of expectations (see Appendix B), and thus in terms of probability density functions. In practice, these expectations are approximated by averaging certain functions of the available data. Since this turns the resulting objective function into a stochastic quantity, the corresponding gradient ascent/descent method is called *stochastic gradient ascent/descent*. The simplest possible approximation is based on using only the current data sample, i.e. by not performing any averaging. In this case, the expectation operator is simply neglected and the resulting objective function is referred to as *instantaneous objective function*. Thus, if $J_{\mathrm{i}}(\mathbf{v})$ is the 'instantaneous version' of some objective function $J(\mathbf{v})$, the equality $J(\mathbf{v}) = E\{J_{\mathrm{i}}(\mathbf{v})\}$ holds. See Sections 2.7.2.7 and 2.7.3.5, for example.

Now that we have discussed most of the general issues involved in blind methods, we will explain some of their specificities in the following sections.

2.6 Whitening the observations

In this section, a frequently used and important preprocessing step for blind methods is explained, the so-called *whitening* or *sphering* stage [34, 36, 47, 52, 64]. For simplicity, we assume that all involved quantities are real-valued, that the mixing and de-mixing matrices are square, i.e. $D = S$, and that no sensor noise is present. For the same reason, and because whitening is employed by the majority of ICA methods, i.e. the blind methods that exploit the mutual statistical independence and non-Gaussianity of the source signals, we focus on ICA. However, the theory can easily be extended to the more general MIBI problem, and thus to IBSS and ISBSL. As has been explained in Section 2.3.1, in the MIBI problem context the purpose of ICA is to find a matrix $\mathbf{W}$ that transforms a random vector of observations $\mathbf{x}$ into the random vector $\mathbf{y} \triangleq \mathbf{W}\mathbf{x}$ the components of which are as statistically independent as possible. In Section B.2.6.8 of Appendix B, it is shown that statistical independence of a set of random variables implies that their cross-cumulants of any order are zero. Hence, $\mathbf{W}$ should be constructed in such a way that the cross-cumulants of $\mathbf{y}$ of all orders are zero. Whitening is a method for partially achieving this goal by zeroing the second order lag-zero cross-cumulants in a separate preprocessing stage. Since second order cumulants of zero mean variables[‡] are equal to correlations, this comes down to de-correlating the output components in $\mathbf{y}$. In addition to zeroing cross-correlations, the variances of the output signals usually are normalized to unity in the preprocessing stage. The total procedure consisting of de-correlation and normalization to unity is called (spatial) whitening or sphering. In order to achieve complete statistical independence, in addition to whitening ICA methods try to zero cross-cumulants for one or more higher orders, whereas temporal structure based methods try to zero cross-correlations or cross-cumulants for one or more times or non-zero lags; see Fig 2.1 and Section 2.6.3. In the sequel, we will first discuss the whitening procedure in detail, and then show how it simplifies ICA methods.

[‡] Everywhere in this thesis all signals are assumed to have zero mean (see Section 1.1).

2.6.1 Whitening transforms and Principal Component Analysis

Formally, the purpose of whitening can be formulated as follows:

> ***Whitening problem statement:***
> The purpose of whitening is to find a linear transformation that transforms a given random vector into another random vector the components of which are uncorrelated and normalized to unit variance.

In the ICA context, the goal of PCA is to find a *whitening transform* or *whitening matrix* $\mathbf{V}$ that transforms the random vector of observations $\mathbf{x}$ into a whitened random vector $\mathbf{y}_{\text{w}} \triangleq \mathbf{V}\mathbf{x}$ the components $y_{\text{w},1}, \ldots, y_{\text{w},S}$ of which are uncorrelated and have unit variance. In 'ICA notation' without time indices, this can be written as follows:

$$r_{ij}^{y_{\text{w}}} \triangleq E\{y_{\text{w},i}\, y_{\text{w},j}\} = \delta_{ij} \qquad \forall\, 1 \leq i,j \leq S\,, \tag{2.6.1}$$

where δ_{ij} denotes a Kronecker delta function (δ_{ij} is 1 iff $i = j$ and 0 elsewhere; see Section A.4). In matrix notation, this condition can be written as:

$$\mathbf{R}^{y_{\text{w}}} \triangleq E\{\mathbf{y}_{\text{w}}\mathbf{y}_{\text{w}}^T\} = \mathbf{I}_S^S\,, \tag{2.6.2}$$

where $\mathbf{I}_S^S$ denotes the S by S identity matrix.

Now, we will explain how a whitening matrix $\mathbf{V}$ can be found. For convenience, we assume real-valued signals and systems. Firstly, a square-root decomposition of the sensor correlation matrix $\mathbf{R}^x \triangleq E\{\mathbf{x}\mathbf{x}^T\}$ is computed [72, 115], i.e. a matrix $\mathbf{L}$ is computed such that $\mathbf{R}^x = \mathbf{L}\mathbf{L}^T$. Then, assuming that $\mathbf{R}^x$ is nonsingular, $\mathbf{V}$ is given by the inverse of $\mathbf{L}$, i.e. $\mathbf{V} = \mathbf{L}^{-1}$. Using the condition in (2.6.2), it follows immediately that the transformed vector $\mathbf{y}_{\text{w}}$ is spatially white:

$$\mathbf{R}^{y_{\text{w}}} = E\{\mathbf{y}_{\text{w}}\mathbf{y}_{\text{w}}^T\} = E\{\mathbf{V}\mathbf{x}\mathbf{x}^T\mathbf{V}^T\} = \mathbf{V}\mathbf{R}^x\mathbf{V}^T = \mathbf{L}^{-1}\mathbf{L}\mathbf{L}^T\mathbf{L}^{-T} = \mathbf{I}\,.$$

The square-root matrix $\mathbf{L}$ can be found in several ways. For example, the Cholesky decomposition [72, 115] can be used. In ICA, a more common approach is based on the Eigenvalue Decomposition (EVD) that will be explained briefly here; for example, see [47]. Let $\mathbf{R}^x = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^T$ be the EVD of $\mathbf{R}^x$, where $\mathbf{E}$ is an orthogonal matrix of eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues $\lambda_1, \ldots, \lambda_D$. Then, a valid square-root decomposition of $\mathbf{R}^x$ is given by $\mathbf{L} = \mathbf{E}\mathbf{\Lambda}^{\frac{1}{2}}$, where $\mathbf{\Lambda}^{\frac{1}{2}} = \text{diag}(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_D})$ (note that under some mild assumptions $\mathbf{R}^x$ is positive definite and thus has positive eigenvalues). Defining $\mathbf{\Lambda}^{-\frac{1}{2}} \triangleq \text{diag}(1/\sqrt{\lambda_1}, \ldots, 1/\sqrt{\lambda_D})$ and substituting $\mathbf{L}^{-1} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^T$ into $\mathbf{V} = \mathbf{L}^{-1}$ yields the following linear whitening transform:

$$\mathbf{V} = \mathbf{L}^{-1} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^T\,. \tag{2.6.3}$$

Using the property $\mathbf{E}\mathbf{E}^T = \mathbf{I}$, it can be verified again that the transformed vector $\mathbf{y}_{\text{w}}$ is spatially white:

$$\mathbf{R}^{y_{\text{w}}} = E\{\mathbf{y}_{\text{w}}\mathbf{y}_{\text{w}}^T\} = \mathbf{V}\mathbf{R}^x\mathbf{V}^T = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^T\mathbf{E}\mathbf{\Lambda}\mathbf{E}^T\mathbf{E}\mathbf{\Lambda}^{-\frac{1}{2}} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{\Lambda}\mathbf{\Lambda}^{-\frac{1}{2}} = \mathbf{I}\,.$$

See [34,36,47,52,64] for more information. In addition to the square-root approach described here, whitening can also be performed using the Singular Value Decomposition (SVD), as described in [156], for example.

A technique that is closely related to whitening is *Principal Component Analysis (PCA)*. Given a length-D random vector $\mathbf{x}$ and an integer $g \leq D$, the goal of PCA is to find a

set of g linearly transformed uncorrelated components in such a way that they explain the maximum possible amount of variance in the data. Our purpose here is twofold; on the one hand, we want to highlight the connection between PCA and whitening, and on the other hand the connection between PCA and ICA. For details about the specificities of PCA we refer the reader to [88] and [80]. It can be shown that the PCA transformation and involved variances are completely determined by the eigenvectors and eigenvalues respectively of the correlation matrix $\mathbf{R}^x$ of the considered random vector $\mathbf{x}$. Let $\mathbf{E}\mathbf{\Lambda}\mathbf{E}^T$ be the EVD of $\mathbf{R}^x$, where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues ordered in decreasing value along the main diagonal, and $\mathbf{E}$ a matrix containing the corresponding eigenvectors. Then, the PCA transformation that yields g uncorrelated components that explain the maximum possible amount of variance in g linearly transformed variables is given by $(\mathbf{E}^g)^T$, where $\mathbf{E}^g$ contains the first g columns of $\mathbf{E}$. The full PCA transformation $\mathbf{V}_{\text{PCA}}$, i.e. the one using all eigenvectors, is given by $\mathbf{V}_{\text{PCA}} = \mathbf{E}^T$. Hence, the vector $\mathbf{V}_{\text{PCA}}\mathbf{x} = \mathbf{E}^T\mathbf{x}$ has uncorrelated components whose variances decrease from the first to the last component. Now, the relation $\mathbf{V} = \mathbf{L}^{-1} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^T = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}_{\text{PCA}}$ in (2.6.3), which defines a proper whitening transformation, shows that whitening can be considered as PCA followed by a scaling by the square roots of the eigenvalues [80]. Hence, whitening can be accomplished by PCA together with post-scaling. Since PCA only zeroes the lag-zero cross-correlations, i.e. second order cross-cumulants, between the output signals, it can be seen as a weak version of ICA because the latter tries to zero the cross-cumulants for all orders larger than one. This also follows immediately from the fact that statistical independence implies uncorrelatedness but not vice versa; see also the next section and Appendix B. Finally, note that *PCA can also be used as a data reduction technique*, which may be useful in ICA problems with more sensors than sources.

2.6.2 ICA complexity reduction by whitening

In this section, we will show that whitening is not sufficient for performing ICA, but reduces its complexity [34, 47, 80]. Consider an arbitrary square orthogonal matrix $\mathbf{U}$, i.e. a matrix that satisfies $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}$. Then, for a given whitening matrix $\mathbf{V}$ *any* matrix $\mathbf{U}\mathbf{V}$ is also a whitening matrix:

$$E\{\mathbf{U}\mathbf{V}\mathbf{x}\mathbf{x}^T\mathbf{V}^T\mathbf{U}^T\} = \mathbf{U}(\mathbf{V}\mathbf{R}^x\mathbf{V}^T)\mathbf{U}^T = \mathbf{U}\mathbf{I}\mathbf{U}^T = \mathbf{I}\,.$$

Since $\mathbf{U}$ may be any orthogonal matrix, the whitening matrix $\mathbf{V}$ is non-unique and gives the independent components up to an orthogonal transformation only. Hence, *whitening is not sufficient to perform ICA.* Obviously, this was to be expected since statistical independence implies uncorrelatedness but not vice versa. Now, consider the relation between the whitened output vector $\mathbf{y}_{\text{w}}$ and the source signal vector $\mathbf{s}$ (recall that the noise is neglected for demonstration purposes):

$$\mathbf{y}_{\text{w}} = \mathbf{V}\mathbf{x} = (\mathbf{V}\mathbf{A})\mathbf{s}\,. \tag{2.6.4}$$

Without loss of generality, assume that the variance of the source signals equals one (this is allowed due to the scaling indeterminacy discussed in Section 2.4), i.e. the source correlation matrix $\mathbf{R}^s \triangleq E\{\mathbf{s}\mathbf{s}^T\}$ is given by $\mathbf{R}^s = \mathbf{I}$. Then, substituting (2.6.4) into (2.6.2) yields:

$$\mathbf{R}^{y_{\text{w}}} = E\{\mathbf{y}_{\text{w}}\mathbf{y}_{\text{w}}^T\} = (\mathbf{V}\mathbf{A})\mathbf{R}^s(\mathbf{V}\mathbf{A})^T = (\mathbf{V}\mathbf{A})(\mathbf{V}\mathbf{A})^T = \mathbf{I}\,.$$

Hence, the matrix $\mathbf{Q} \triangleq \mathbf{V}\mathbf{A}$ that relates the sources to the whitened outputs is orthogonal and the whitened output signal vector is a rotated version of the source vector:

$$\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s} \quad \text{with} \quad \mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^T\mathbf{Q} = \mathbf{I}\,. \tag{2.6.5}$$

In other words, after whitening the ICA problem becomes the problem of finding the unique orthogonal matrix $\mathbf{Q}^T = \mathbf{Q}^{-1}$ that yields statistically independent output components with unit variance; see also (2.6.6). Since an orthogonal matrix has less degrees of freedom than an arbitrary matrix, this problem is simpler than solving the original problem for $\mathbf{A}$. This is one of the reasons for whitening the observation data. After estimating $\mathbf{Q}$, the source signals can be recovered/estimated by computing the output vector $\mathbf{y}$ as follows:

$$\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^{-1}\mathbf{y}_{\mathrm{w}} = \mathbf{Q}^T\mathbf{y}_{\mathrm{w}} = \mathbf{W}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}}\,, \tag{2.6.6}$$

where for convenience we have denoted the orthogonal part $\mathbf{Q}^T$ of the total de-mixing system $\mathbf{W}$ in the original ICA problem by $\mathbf{W}_{\mathrm{o}}$. The letter 'o' in the subscript position of $\mathbf{W}_{\mathrm{o}}$ stands for 'orthogonal'. Hence, $\mathbf{W}$ can be decomposed as $\mathbf{W} = \mathbf{Q}^T\mathbf{V} = \mathbf{W}_{\mathrm{o}}\mathbf{V}$, where $\mathbf{W}_{\mathrm{o}}$ and $\mathbf{V}$ are orthogonal and whitening matrices respectively.

2.6.3 Two-stage approach to blind identification and separation

In the previous sections it was shown that whitening reduces the standard ICA problem with an *arbitrary* (square) mixing matrix to an ICA problem with an *orthogonal* mixing matrix. The estimation of the remaining orthogonal mixing system $\mathbf{Q}$ can be accomplished in two essentially different ways: either by exploiting the Higher Order Statistics in the data (see Section 2.3.1), or by exploiting the temporal structure (see Section 2.3.2). In the first case, $\mathbf{Q}$ is determined by requiring the output signals to be higher order statistically independent and non-Gaussian. For example, this can be achieved by requiring that the lag-zero cross-cumulants are zero for all orders larger than one. The principles of this approach are discussed in Section 2.7. In the second case, $\mathbf{Q}$ is determined by requiring the output signals to be statistically independent for different times or lags. For example, this can be achieved by requiring that all cross-correlation functions between different output components are zero for all lags, i.e. $r_{ij}^{y}[m,n] = E\{y_i[m]\,y_j[n]\} = 0^{\ddagger}$ for all $1 \leq i \neq j \leq S$ and $m,n \in \mathbb{Z}$. The same goal can also be achieved by requiring that all cross-cumulant functions between output components of one or more orders are zero for all time or lag tuples. The principles of this approach are discussed in Section 2.8. As we will show in later chapters, one of the contributions of this thesis is in the latter area. In summary, the following two-stage approach to instantaneous blind problems, depicted in Fig 2.1, is often considered in the literature:

ST1: *Spatially whiten the sensor/observation data;*

ST2: *Determine the remaining orthogonal (de-)mixing system by using one of the following approaches:*

- *Exploit spatial statistical independence and non-Gaussianity;*
- *Exploit temporal structure.*

We conclude this section by pointing out some important advantages and disadvantages of whitening [34, 80, 189]. The main advantage is that blind problems are reduced to simpler ones. In addition, Second Order Statistics can be estimated more accurately than Higher Order Statistics. Therefore, it seems reasonable to first exploit all SOS information conveyed by the data. However, depending on the specific application and available data, whitening may severely limit the performance of the whole separation system in the sense that errors in the whitening stage cannot be corrected in the following orthogonal separation stage [31].

[‡] See Appendices A and B for the definition and notation of correlation and cumulant functions.

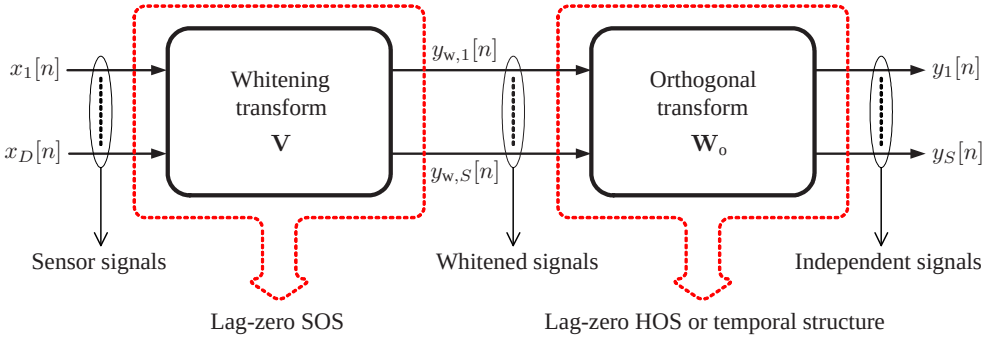

Figure 2.1: *Two-stage approach to ICA.*

Another drawback is that sensor noise has a large influence on the performance of the whitening stage. In this work, we will not adopt the two-stage procedure but estimate the mixing matrix directly.

Now that we have discussed the most important generalities and preprocessing stages of blind methods, we will explain the rationale behind each of the two main categories distinguished in Section 2.3. Section 2.7 deals with the ICA category, i.e. the category that is based on exploiting mutual (spatial) statistical independence and non-Gaussianity, and neglects any temporal structure in the data. Different aspects, principles as well as algorithms from the literature will be discussed. Section 2.8 deals with a subclass of the second category, viz. the category that is based on exploiting mutual (spatial) uncorrelatedness and Second Order Temporal Structure (SOTS). Again, different aspects, principles and algorithms from the literature will be discussed.

2.7 Exploiting spatial independence and non-Gaussianity

This section discusses several aspects, identification/separation principles and algorithms from the literature for the ICA category of methods. Based on the fundamental assumptions of mutual statistical independence and non-Gaussianity of the source signals, several useful criteria and/or measures for developing ICA methods are derived. The first fundamental assumption, which is of paramount importance to all blind methods, has been discussed and formulated mathematically in Section 2.3, see condition (2.3.2). In the next section we will explain why the second assumption is also crucial for ICA methods. The subsequent sections will cover different approaches for measuring mutual statistical independence and non-Gaussianity, and present several practical algorithms.

2.7.1 Non-Gaussianity

In this section, we will give an intuitive explanation for the source non-Gaussianity requirement of Independent Component Analysis (ICA) in the separation context [80]. For simplicity and focus on principles, all source signals are assumed to have the same pdf and the sensor noise is neglected for the moment. Hence, each sensor signal is a linear combination of the

source signals only, see (2.3.1):

$$x_i = \sum_{j=1}^{S} a_i^j s_j \qquad \forall\, 1 \leq i \leq D\,.$$

The well-known Central Limit Theorem (CLT), discussed in Section B.2.4, states that the sum of several independent random variables with the same pdf has a pdf that is closer to a Gaussian distribution than the pdf of the individual variables. Hence, in general the sensor signals, which are linear combinations of statistically independent source signals, are more Gaussian than the source signals. From this, an intuitive separation principle follows directly: design the de-mixing matrix $\mathbf{W}$ in (1.1.2) in such a way that the non-Gaussianity of the output signals is maximized (also a single source can be estimated by estimating one row of $\mathbf{W}$ according to this principle). Formulated differently, *because a 'general' mixing matrix* $\mathbf{A}$*, i.e. one that cannot be decomposed in terms of permutation and scaling matrices, makes the observed signals more Gaussian than the source signals, the de-mixing matrix* $\mathbf{W}$ *should try to undo this effect by making the output signals (i.e. the estimated source signals) less Gaussian (i.e. more non-Gaussian).* Consequently, in order to find the de-mixing matrix, a measure of non-Gaussianity is needed. In Appendix B, it is shown that the non-Gaussianity of random variables can be measured by their cumulants of order larger than two because in theory these are identically zero only for Gaussian random variables (see *UCP2* on page 425). Hence, higher order cumulants can be used for this purpose, implying the use of Higher Order Statistics (HOS). Now it is also clear why the source signals are required to be non-Gaussian, or, more precisely, why at most one source signal is allowed to be Gaussian [52, 80]. If the source signals are already Gaussian, their higher order cumulants are zero and thus there is no point in maximizing the non-Gaussianity of the source estimates. Summarizing, in the ICA context statistical independence and non-Gaussianity are closely related. In fact, maximum non-Gaussianity means statistical independence and vice versa. Hence, *ICA estimation by minimizing statistical dependence between source signals is equivalent to maximizing their individual non-Gaussianities*. See Section 2.7.2.6, [80] and the references therein for a more rigorous justification. A consequence of performing ICA by maximizing non-Gaussianity is that the signals can be extracted one at a time [80]. This procedure is usually called *Blind Signal Extraction (BSE)* [47] and comes down to estimating one row of the de-mixing matrix at a time. Algorithms that estimate one component at a time are typically called one-unit algorithms [47, 80]. In order to recover more components by running a one-unit algorithm several times, a proper kind of deflation is necessary [81].

Practical measures of statistical independence and non-Gaussianity

Together, the mutual statistical independence and non-Gaussianity assumptions form the basis for designing identification and separation criteria for tackling BSP problems. In ICA, there are mainly two approaches to measuring mutual statistical independence and non-Gaussianity. One is based on *information-theoretic concepts* such as entropy and mutual information. The theoretical measures based on these concepts typically require a priori knowledge of the involved probability density functions, which usually is difficult or impossible to obtain. In order to turn these concepts into useful separation criteria, more practical measures are required that are expressed in terms of (higher order) cumulants and moments. In principle, the latter statistics are also defined in terms of pdf's, but in practice they usually can be estimated directly from the data. Many differences between information-theoretic ICA approaches presented in the literature essentially are only due to the differences in the ways

the information-theoretic measures are approximated by cumulants and moments. Despite all these differences, the various approaches are closely related. Moreover, since ICA estimation by minimizing statistical dependence is equivalent to maximizing non-Gaussianity (see previous paragraph), measures for non-Gaussianity are also measures of mutual statistical independence. This issue will be highlighted in Section 2.7.2.6. The second approach to measuring mutual statistical independence and non-Gaussianity is based on *traditional 'signal processing oriented' concepts* such as (nonlinear) correlations, cumulants, etc. Most methods based on this approach use measures that can directly be derived from the statistical independence assumption (see Section 2.3 and Appendix B) and are formulated in terms of cross-cumulants. In fact, it can be shown that this category may be viewed as a subclass of the first one, i.e. the one based on information-theoretic concepts. Many differences between signal processing based approaches presented in the literature are due to the differences in the order of the exploited statistics, the kind of temporal information that is exploited (e.g. non-whiteness and/or non-stationarity, see Section 1.2.2), the way the statistical information is arranged in matrices (see Section 1.2.6), etcetera.

In the next sections, we will discuss the above two classes of non-Gaussianity measures/ approaches, focussing on principles and insight instead of complete derivations. Section 2.7.2 discusses the category of information-theoretic non-Gaussianity measures and presents a practical algorithm based on them. Then, Section 2.7.3 discusses the category of signal processing oriented non-Gaussianity measures, presenting several practical algorithms along the way. Because the MIBI method that we will present in later chapters uses a cross-cumulant based measure of statistical independence belonging to the signal processing oriented category of measures, we elaborate on the most important principles of cross-cumulant based algorithms in a dedicated section, viz. Section 2.7.4.

2.7.2 Information-theoretic measures of statistical independence and non-Gaussianity

This section discusses the information-theoretic concepts and their approximations referred to in the previous section. It also provides the theoretical background for the application of these concepts to the ICA problem, as well as a practical algorithm.

2.7.2.1 Differential entropy

The *differential entropy* $H(\mathbf{v})$ of a random vector $\mathbf{v}$ with jpdf $p_{\mathbf{v}}(\mathbf{v})$ is defined as follows [58], [122]:

$$H(\mathbf{v}) \triangleq -E\left\{\log(\mathbf{v})\right\} = -\int_{-\infty}^{\infty} \log(p_{\mathbf{v}}(\mathbf{v}))\, p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v}\,, \tag{2.7.1}$$

where $\log(\cdot)$ usually denotes the natural logarithm. Hereafter, differential entropy will simply be referred to as 'entropy'. The entropy of a random variable or vector is a measure of the amount of information that the observation of the variable respectively vector provides. The more random the variable or vector, the larger its entropy, and vice versa. A fundamental result in information theory states that *the Gaussian distribution has maximum entropy among all distributions with a given covariance matrix*. This means that entropy can be used as a measure of non-Gaussianity. A measure that is commonly used in ICA/IBSS is a normalized (w.r.t. a Gaussian distribution) version of entropy, the so-called negentropy.

2.7.2.2 Negentropy

The *negentropy* $H_{\text{neg}}(\mathbf{v})$ of $\mathbf{v}$ is defined as follows [80]:

$$H_{\text{neg}}(\mathbf{v}) \triangleq H(\mathbf{v}_{\text{gauss}}) - H(\mathbf{v})\,, \tag{2.7.2}$$

where $\mathbf{v}_{\text{gauss}}$ is a Gaussian random vector with the same covariance matrix as $\mathbf{v}$. Due to the maximality property mentioned in Section 2.7.2.1, negentropy is always nonnegative; moreover, the larger it is, the more non-Gaussian $\mathbf{v}$ is. Moreover, it can be shown from the uniqueness of the maximum entropy distribution that it is zero if and only if $\mathbf{v}$ is Gaussian. Hence, algorithms based on negentropy should ***maximize the negentropy*** of the components of $\mathbf{y}$ in (2.3.3) as a function of $\mathbf{W}$.

2.7.2.3 Mutual information

The *mutual information* $I(\mathbf{v})$ between the random variables $v_1, \ldots, v_G$ of the random vector $\mathbf{v}$ is defined as follows [58]:

$$I(\mathbf{v}) \triangleq \sum_{j=1}^{G} H(v_j) - H(\mathbf{v})\,, \tag{2.7.3}$$

i.e. it is the sum of the individual/marginal entropies minus the joint entropy. It can be shown that mutual information is always nonnegative, and zero if and only if the components of $\mathbf{v}$ are statistically independent. Hence, *mutual information is a natural measure of the independence of random variables*. Algorithms based on mutual information should ***minimize the mutual information*** of the components of $\mathbf{y} = \mathbf{W}\mathbf{x}$ in (2.3.3) as a function of $\mathbf{W}$. Since mutual information is completely determined by a (joint) probability density function (like differential entropy and negentropy), with abuse of notation we can write $I(p_{\mathbf{v}}(\mathbf{v}))$. In [52], it is shown that the negative mutual information $\Xi(p_{\mathbf{v}}(\mathbf{v})) \triangleq -I(p_{\mathbf{v}}(\mathbf{v})$ satisfies properties *CFP1-CPP3* on page 42 and thus is a contrast function; see also Section 2.5. The mutual information $I(\mathbf{y})$ of $\mathbf{y}$ can be expressed as a function of the de-mixing matrix $\mathbf{W}$ and the entropy $H(\mathbf{x})$ of the observations as follows (recall that we are assuming that $S = D$):

$$I(\mathbf{y}) = \sum_{j=1}^{D} H(y_j) - H(\mathbf{x}) - \log(\det(\mathbf{W}))\,. \tag{2.7.4}$$

This expression is very important for ICA and IBSS because many algorithms are inspired by, or based on, it. For whitened data (see Section 2.6), (2.7.4) becomes:

$$I(\mathbf{y}) = \sum_{j=1}^{D} H(y_j) - H(\mathbf{y}_{\text{w}}) \tag{2.7.5}$$

because $\log(\det(\mathbf{Q})) = \log(1) = 0$ due to the fact that $\mathbf{Q}$ is orthogonal. As is shown in [52], $I(\mathbf{y})$ can also be expressed in terms of negentropies and correlation values as follows:

$$I(\mathbf{y}) = H_{\text{neg}}(\mathbf{y}) - \sum_{j=1}^{D} H_{\text{neg}}(y_j) + \frac{1}{2}\log\left(\frac{\prod_j (\sigma_j^y)^2}{\det(\mathbf{R}^y)}\right)\,, \tag{2.7.6}$$

where the variances $(\sigma_j^y)^2$ are given by the diagonal elements of the correlation matrix $\mathbf{R}^y \triangleq E\{\mathbf{y}\mathbf{y}^T\}$. It can easily be seen that if the components of $\mathbf{y}$ are uncorrelated, this expression

becomes:

$$I(\mathbf{y}) = H_{\text{neg}}(\mathbf{y}) - \sum_{j=1}^{D} H_{\text{neg}}(y_j)\,. \tag{2.7.7}$$

2.7.2.4 Kullback-Leibner divergence

The *Kullback-Leibner divergence*, often abbreviated as KL-divergence, between two probability density functions $f_{\mathbf{v}}(\mathbf{v})$ and $g_{\mathbf{v}}(\mathbf{v})$ is defined as follows:

$$K(f_{\mathbf{v}} \parallel g_{\mathbf{v}}) \triangleq \int_{-\infty}^{\infty} \log\left(\frac{f_{\mathbf{v}}(\mathbf{v})}{g_{\mathbf{v}}(\mathbf{v})}\right) f_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v}\,. \tag{2.7.8}$$

It can be shown that $K(f_{\mathbf{v}} \parallel g_{\mathbf{v}}) \geq 0$ with equality if and only if the two distributions $f_{\mathbf{v}}(\mathbf{v})$ and $g_{\mathbf{v}}(\mathbf{v})$ are equal. The KL-divergence is a kind of distance between two probability density functions. However, it is not a proper distance function because it is asymmetric in its arguments, i.e. $K(f_{\mathbf{v}} \parallel g_{\mathbf{v}}) \neq K(g_{\mathbf{v}} \parallel f_{\mathbf{v}})$. *The KL-divergence can be used as a measure of statistical independence* of the components of the random vector $\mathbf{y}$. This can be done by considering the divergence between the joint probability density function $p_{\mathbf{y}}(\mathbf{y})$ of the output signals and the product of the marginal probability density functions $p_{y_1}(y_1), \ldots, p_{y_D}(y_D)$ derived from $p_{\mathbf{y}}(\mathbf{y})$ (see Section B.2.1 and equation (B.2.8)):

$$K\left(p_{\mathbf{y}}(\mathbf{y}) \,\middle\|\, \prod_{j=1}^{D} p_{y_j}(y_j)\right) \triangleq \int_{-\infty}^{\infty} \log\left(\frac{p_{\mathbf{y}}(\mathbf{y})}{\prod_{j=1}^{D} p_{y_j}(y_j)}\right) p_{\mathbf{y}}(\mathbf{y})\, d\mathbf{y}\,. \tag{2.7.9}$$

For a factorizable jpdf, which is reminiscent of statistically independent components, the logarithm in (2.7.9) yields zero. Hence, the Kullback-Leibner divergence between a jpdf, and its version factorized in terms of the mpdf's, can be used as a measure of statistical independence. This is also evident from the fact that $K\big(p_{\mathbf{y}}(\mathbf{y}) \| \prod_{j=1}^{D} p_{y_j}(y_j)\big)$ is equal to the mutual information of the random variables $y_1, \ldots, y_D$; see Section 2.7.2.3 . This can easily be seen as follows:

$$\begin{aligned} K\left(p_{\mathbf{y}}(\mathbf{y}) \,\middle\|\, \prod_{j=1}^{D} p_{y_j}(y_j)\right) &= -\sum_{j=1}^{D} \int_{-\infty}^{\infty} \log\left(p_{y_j}(y_j)\right) p_{\mathbf{y}}(\mathbf{y})\, d\mathbf{y} \\ &\quad + \int_{-\infty}^{\infty} \log\left(p_{\mathbf{y}}(\mathbf{y})\right) p_{\mathbf{y}}(\mathbf{y})\, d\mathbf{y} \overset{(2.7.1)}{=} \sum_{j=1}^{D} H(y_j) - H(\mathbf{y}) \overset{(2.7.3)}{=} I(\mathbf{y})\,. \end{aligned} \tag{2.7.10}$$

2.7.2.5 Approximating entropy and related quantities

The problem with using entropy and related quantities such as negentropy and mutual information is that they are difficult to estimate in practice because the complete (j)pdf's have to be known and difficult integrations have to be performed. It is well known that in general the parametric or non-parametric estimation of probability density functions is a difficult problem that is very prone to errors. For these reasons, entropy mainly remains a theoretical

quantity and approximations are used in practice. There are several methods for approximating entropy and related quantities by expressing it in terms of cumulants or more general expectations; see Appendix B and [82], [81], [78].

The classic method uses polynomial expansions of the probability density functions around a Gaussian pdf. Two common expansions are the Gram-Charlier and Edgeworth expansions [34, 52, 111]. Using the first few terms of such an expansion (which form an approximation of the pdf to a Gaussian pdf), together with the first few terms of the Taylor expansion of the logarithm in (2.7.1), an approximation of entropy is obtained. For example, for a zero mean univariate random variable v, the Gram-Charlier expansion results in the following approximation of entropy [80]:

$$H(v) \approx H(v_{\text{gauss}}) - \frac{1}{12}\big(\kappa^{v,3}\big)^2 - \frac{1}{48}\big(\kappa^{v,4}\big)^2, \tag{2.7.11}$$

where $H(v_{\text{gauss}})$ denotes the entropy of a standardized Gaussian variable, and $\kappa^{v,l}$ denotes the l-th order cumulant of v[‡]. From (2.7.2), it then follows that:

$$H_{\text{neg}}(v) \approx \frac{1}{12}\big(\kappa^{v,3}\big)^2 + \frac{1}{48}\big(\kappa^{v,4}\big)^2. \tag{2.7.12}$$

Similar (generalized) approximations can be obtained for random vectors. Likewise, owing to relations like (2.7.6) and (2.7.7), approximations of the Kullback-Leibner divergence and mutual information can also be found. For more information on approximations of entropy and related quantities, see [89], [80], [82], [75], [6] and [52] and the references therein.

2.7.2.6 The relation between statistical independence and non-Gaussianity

In this section, we briefly reconsider the important relation between statistical independence and non-Gaussianity (see Section 2.7.1) by examining the relation between the corresponding information-theoretic measures, viz. mutual information and negentropy respectively. First note that from (2.7.10) it follows that mutual information and the Kullback-Leibner divergence are equivalent measures of statistical independence. Likewise, from (2.7.2) it follows that negentropy and differential entropy are similar or equivalent measures of non-Gaussianity. Hence, examining the link between mutual information and negentropy (i.e. statistical independence and non-Gaussianity) is equivalent to examining the link between Kullback-Leibner divergence and differential entropy respectively, or between the Kullback-Leibner divergence and negentropy. For convenience, here we consider the relation between mutual information and negentropy.

For the sake of clarity, we assume that the observation signal vector $\mathbf{x}$ of our MIBI/ICA model (2.3.1) has been whitened. As in Section 2.6, the resulting whitened signal vector is denoted $\mathbf{y}_{\text{w}}$ and the remaining problem is the determination of the orthogonal matrix $\mathbf{Q}$ in (2.6.6) from the samples of $\mathbf{y}_{\text{w}}$; see Section 2.6.2. Equality (2.7.7) shows that the mutual information $I(\mathbf{y})$ of $\mathbf{y}$ can be expressed in terms of the joint negentropy of $\mathbf{y}$ and the individual negentropies of its components. Furthermore, in [52] and [80] it is shown that *negentropy is invariant for invertible linear transformations*. Now, from (2.6.6) it follows that $H_{\text{neg}}(\mathbf{y}) = H_{\text{neg}}(\mathbf{Q}^T\mathbf{y}_{\text{w}}) = H_{\text{neg}}(\mathbf{y}_{\text{w}})$. Substituting this result into (2.7.7) finally gives:

$$I(\mathbf{y}) = H_{\text{neg}}(\mathbf{y}_{\text{w}}) - \sum_{j=1}^{D} H_{\text{neg}}(y_j)\,. \tag{2.7.13}$$

[‡]In terms of the multivariate cumulant notation discussed in Section B.2.6.5, $\kappa^{v,l} \triangleq \text{cum}(v,\ldots,v)$.

Since the negentropy $H_{\text{neg}}(\mathbf{y}_{\text{w}})$ of the whitened data is not influenced by the final orthogonal transformation $\mathbf{y} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ to statistically independent components, the minimization of $I(\mathbf{y})$ as a function of $\mathbf{Q}$ (or $\mathbf{Q}^T$) is equivalent to maximizing the sum $\sum_{j=1}^{D} H_{\text{neg}}(y_j)$. In other words, ICA estimation by minimizing the mutual information of the estimated source signals is equivalent to maximizing the sum of their individual negentropies. Hence, because of the links emphasized at the beginning of this section, *ICA estimation by minimizing the statistical independence of the estimated source signals is equivalent to maximizing the sum of their non-Gaussianities*. This is exactly what we claimed in Section 2.7.1. For more information, see Chapters 10 and 14 of [80].

2.7.2.7 Algorithm based on information-theoretic measure(s)

Since most algorithms based on information-theoretic concepts are closely related, and because of space limitations, we only discuss the most common algorithm based on minimizing mutual information [5–7, 12, 47, 65]. This algorithm tries to find the de-mixing matrix $\mathbf{W}$ by minimizing the mutual information of the estimated source signals by a gradient descent method (see Section 2.5). Naturally, a cost function that is equal to the mutual information $I(\mathbf{y})$ in (2.7.4) is chosen. We briefly sketch a possible intuitive derivation; see the references above for more information. According to the standard gradient descent method, the cost function defined as $I(\mathbf{y})$ is minimized by iteratively updating some initial value $\mathbf{W}[0]$ of $\mathbf{W}$ as follows:

$$\mathbf{W}[k+1] = \mathbf{W}[k] - \mu[k]\frac{\partial I(\mathbf{y}(\mathbf{W}))}{\partial \mathbf{W}}\bigg|_{\mathbf{W}[k]} . \tag{2.7.14}$$

The derivative of $I(\mathbf{y}) = I(\mathbf{y}(\mathbf{W}))$ w.r.t.$\mathbf{W}$ is given by (for example, see [5, 7, 47, 65]):

$$\frac{\partial I(\mathbf{y})}{\partial \mathbf{W}} = -\mathbf{W}^{-T} + \sum_{j=1}^{D}\frac{\partial H(y_j)}{\partial \mathbf{W}} = -\mathbf{W}^{-T} - \sum_{j=1}^{D}\frac{\partial E\{\log(y_j)\}}{\partial \mathbf{W}} . \tag{2.7.15}$$

It can be shown that the second term at the right hand side can be written as follows [5, 47]:

$$-\sum_{j=1}^{D}\frac{\partial E\{\log(y_j)\}}{\partial \mathbf{W}} = E\{\mathbf{f}(\mathbf{y})\mathbf{x}^T\} , \tag{2.7.16}$$

where

$$\mathbf{f}(\mathbf{y}) \triangleq \begin{bmatrix} f_1(y_1) \\ \vdots \\ f_D(y_D) \end{bmatrix} \qquad \text{with} \qquad f_j(y_j) \triangleq -\frac{\partial \log(p_{y_j}(y_j))}{\partial y_j} \quad \forall\, 1 \le j \le D .$$

The nonlinear functions $f_j(\cdot)$ are typically called *score* or *activation functions* [5]. The ideal form of the functions is determined by the source pdf's, i.e. by substituting $p_{y_j}(y_j) = p_{s_j}(y_j)$ in the expression above for $f_j(y_j)$. However, in (fully) blind signal processing the source densities are generally unknown; otherwise the problem would be semi-blind. Therefore, most researchers assume the same arbitrary fixed form for all functions, i.e. $f_1(y) = \cdots = f_D(y) \triangleq f(y)$, where typical examples of $f(y)$ are $\text{sign}(y), \tanh(y), (y)^3$, etc. Another possibility is to estimate the score function by approximating the probability density functions. This can be done by using Gram-Chalier or Edgeworth expansions (see Section 2.7.2.5). Another way is to parameterize the score function(s) or source pdf(s) and try to estimate the

parameters adaptively from the output signals. We will not address these issues here, but assume that the score functions are known or can be approximated appropriately.

Using the instantaneous version of gradient (2.7.15) in update rule (2.7.14), the following on-line stochastic gradient descent algorithm is obtained (See Section 2.5):

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k] \left(\mathbf{W}^{-T}[k] - \mathbf{f}(\mathbf{y}[k]) \mathbf{x}^T[k] \right) . \tag{2.7.17}$$

Note that the index k iterates over the available samples. For example, in the case of time series data it becomes a time index. Using the relation $\mathbf{y}[k] = \mathbf{W}[k]\mathbf{x}[k]$, (2.7.17) can be written as follows:

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k] \left(\mathbf{I} - \mathbf{f}(\mathbf{y}[k]) \mathbf{y}^T[k] \right) \mathbf{W}^{-T}[k] . \tag{2.7.18}$$

This learning rule is of fundamental importance in ICA. Many other approaches based on information-theoretic concepts arrive at the same or a slightly different rule from different viewpoints. In [4,7,47], it is shown that the parameter space consisting of nonsingular square matrices has a Riemannian, i.e. non-Euclidian structure. Therefore, the natural gradient approach is more suitable to the problem and it provides a significantly improved convergence; see also Section 2.5. As shown in [4, 7, 50, 75] and [47], the natural gradient is obtained by post-multiplying the standard gradient by $\mathbf{W}^T\mathbf{W}$, resulting in the following learning rule:

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k] \left(\mathbf{I} - \mathbf{f}(\mathbf{y}[k]) \mathbf{y}^T[k] \right) \mathbf{W}[k] . \tag{2.7.19}$$

In addition to providing enhanced performance, this rule is also easier to compute than (2.7.18) because the matrix inversion is no longer required. We summarize the results of this section in the following algorithm for performing IBSS based on the minimization of mutual information:

Algorithm 2.1 IBSS based on minimizing mutual information.

Initialize $\mathbf{W}[k]$*, then iterate over available samples:*

1: *Compute output signals by* $\mathbf{y}[k] = \mathbf{W}[k]\mathbf{x}[k]$*;*

2: *Update* $\mathbf{W}[k]$ *according to learning rule* (2.7.18)*:*

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k] \left(\mathbf{I} - \mathbf{f}(\mathbf{y}[k]) \mathbf{y}^T[k] \right) \mathbf{W}^{-T}[k]$$

or (2.7.19)*:*

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k] \left(\mathbf{I} - \mathbf{f}(\mathbf{y}[k]) \mathbf{y}^T[k] \right) \mathbf{W}[k] .$$

Finally, we note that contrary to the development above, where we started with the ideal (i.e. 'unapproximated') mutual information and introduced approximations later on during the gradient estimation, it is also possible to start directly from some approximation of the mutual information (see Section 2.7.2.5). The latter approach yields learning rules that are the same as, or very similar to, the ones derived above.

2.7.3 Signal processing oriented measures of statistical independence and non-Gaussianity

This section discusses several signal processing oriented measures of statistical independence and non-Gaussianity, such as joint cumulants, joint moments, and other types of (nonlinear) joint expectations. Like Section 2.7.2, it also provides the theoretical background for the application of these concepts to the ICA problem. In principle, similarly to the information-theoretic measures the signal processing measures are also specified in terms of (joint) probability density functions, but in practice they usually can be estimated directly from the data. We will make use of several well-known results of statistics that are summarized in Appendix B; for details we refer the reader to [111, 112, 122, 154]. Along the way, several practical algorithms are discussed.

2.7.3.1 Measuring statistical independence by cross-cumulants

In this section, we will consider the use of joint cumulants for measuring statistical independence. This is justified by a theorem that can be derived from the statistical independence criterion formulated in Section B.2.3 (for an example in terms of the source signals, see (2.3.2) in Section 2.3.1). In Section B.2.6.8, this theorem is formulated and proven as Theorem B.2.3; we repeat it here for convenience:

Theorem 2.7.1. Statistical independence criterion based on joint cumulants. *The random variables $v_1, \ldots, v_G$ are mutually statistically independent if and only if their cross-cumulants of all possible orders are zero:*

$$\operatorname{cum}\left(v_{i_1}, \ldots, v_{i_l}\right) = 0 \qquad \forall\, (i_1, \ldots, i_l) \in \mathcal{I}^l_{\mathrm{c},G}\,, \quad \forall\, l \in \mathbb{N}\,. \tag{2.7.20}$$

The notation $\operatorname{cum}(\cdot, \cdots, \cdot)$ denotes the cumulant function[‡] and $\mathcal{I}^l_{\mathrm{c},G}$ denotes the set of all length-l 'cross-tuples' of indices[§], i.e. all possible tuples with $1 \leq i_1, \ldots, i_l \leq G$ except for those with all indices equal $i_1 = \cdots = i_l$.

Section B.2.6.8 also provides two theorems that are simplified versions of this theorem, and which are very useful in practice. We will now briefly repeat these theorems, and then give a concrete example.

Definition 2.7.2. Statistical independence up to order l. The random variables $v_1, \ldots, v_G$ are said to be *mutually statistically independent up to order l* if and only if all their cross-cumulants of orders smaller than or equal to l are zero:

$$\operatorname{cum}\left(v_{i_1}, \ldots, v_{i_k}\right) = 0 \qquad \forall\, (i_1, \ldots, i_k) \in \mathcal{I}^k_{\mathrm{c},G}\,, \quad \forall\, k \leq l\,. \tag{2.7.21}$$

In words, this theorem states that $v_1, \ldots, v_G$ are mutually statistically independent up to, and including, order l if and only if all their possible cross-cumulants of orders smaller than or equal to l are zero. Note that the set of conditions in (2.7.21) is weaker than that in (2.7.20).

Definition 2.7.3. Statistical independence at order l. The random variables $v_1, \ldots, v_G$ are said to be *mutually statistically independent at order l* if and only if all their cross-cumulants

[‡] See Sections B.2.6.2 and B.2.6.5 of Appendix B for the definition of the moment and cumulant functions $\operatorname{mom}(\cdot, \cdots, \cdot)$ and $\operatorname{cum}(\cdot, \cdots, \cdot)$ respectively.

[§] See Appendix A for information about tuple and set notations like $(i_1, \ldots, i_l)$ and $\mathcal{I}^l_{\mathrm{c},G}$ respectively.

of order l are zero:

$$\operatorname{cum}\left(v_{i_1},\ldots,v_{i_l}\right)=0 \qquad \forall\,(i_1,\ldots,i_l)\in\mathcal{I}_{\mathrm{c},G}^{l}\,. \tag{2.7.22}$$

This theorem states that $v_1,\ldots,v_G$ are mutually statistically independent at order l if and only if all their possible cross-cumulants of order l are zero. Note that the set of conditions in (2.7.20) and (2.7.22) is weaker than that in (2.7.21).

Now, in order to make these theorems and associated notation more tangible, we consider Definition 2.7.3 for a specific case, viz. the statistical independence at order two, i.e. $l=2$, of three real-valued zero mean random variables v_1, v_2 and v_3, i.e. $G=3$. First note that for zero mean variables, second order cumulants are equal to correlations (see Section B.2.6.6, and in particular (B.2.44b) on page 442). Furthermore, according to the shorthand notation explained in Section A.2, the set $\mathcal{I}_{\mathrm{c},3}^{2}$ is defined as the set of all pairs of indices ranging from 1 to 3 for which the indices are different, i.e. $\mathcal{I}_{\mathrm{c},3}^{2}\triangleq\{(i_1,i_2)\mid 1\leq i_1,i_2\leq 3, i_1\neq i_2\}=\{(1,2),(2,1),(1,3),(3,1),(2,3),(3,2)\}$. Since $\operatorname{cum}\left(v_{i_1},v_{i_2}\right)=E\{v_{i_1}v_{i_2}\}=E\{v_{i_2}v_{i_1}\}=\operatorname{cum}\left(v_{i_2},v_{i_1}\right)$ (see Sections B.1.3 and B.2.6 respectively for the definitions of uni- and multivariate expectations), half of the constraints in (2.7.22) is redundant and can be omitted. Hence, (2.7.22) yields the following set of conditions:

$$\operatorname{cum}\left(v_{i_1},v_{i_2}\right)=E\{v_{i_1}v_{i_2}\}=0 \qquad \forall\,(i_1,i_2)\in\{(1,2),(1,3),(2,3)\}\,.$$

Therefore, for this specific example in words Definition 2.7.3 states that v_1, v_2 and v_3 are mutually statistically independent *at* order 2, i.e. uncorrelated, if and only if the cross-correlations r_{12}^{v}, r_{13}^{v} and r_{23}^{v} are all zero.

The theorem and definitions above can be applied directly to MIBI and related problems because these usually assume that the source signals $s_1,\ldots,s_S$ (in MIBI and ISBSL) and/or the output signals $y_1,\ldots,y_D$ (in IBSS/ICA) are statistically independent at, or up to, some order l. For example, in Section 2.2.2 we have seen that it is commonly assumed that certain cross-cumulants of the source signals are zero. In practice, cumulants and moments can only be estimated with sufficient accuracy for orders smaller than or equal to four or five. For this reason, in practice we are more or less forced to employ Definition 2.7.2 or 2.7.3 instead of Theorem 2.7.1, which is mainly of theoretical importance. Most blind approaches based on exploiting temporal structure use second order statistics, whereas approaches based on exploiting statistical independence and non-Gaussianity require the order of the statistics to be at least 3 because of *ASS3*. Most practical algorithms in the latter category use third order statistics for signals with asymmetric pdf's and fourth order statistics for signals with symmetric pdf's. Depending on the assumptions adopted by a certain method, considering a single order might be sufficient for identifying the mixing system or recovering the sources. Later on we will show that the MIBI method that we will develop in later chapters allows one to exploit several orders simultaneously, which may be advantageous for some scenarios. As we have indicated at the end of Section 2.7.1, because this method employs the cross-cumulant based measure of statistical independence discussed above, we will discuss cross-cumulant based algorithms described in the literature along with their main principles in a dedicated section, viz. Section 2.7.4. In the next section, we discuss another signal processing oriented approach for measuring statistical independence that is based on (nonlinear) joint expectations. Two practical algorithms based on this approach are discussed in Section 2.7.3.3.

2.7.3.2 Measuring statistical independence by (nonlinear) joint expectations

In this section, we consider a basic property of statistically independent random variables for measuring statistical independence. As is shown in Section B.2.3, this property is expressed in terms of mathematical expectations and can be deduced very simply from the statistical independence criterion. It is formulated in Theorem B.2.1, which is repeated here for convenience:

Theorem 2.7.4. Nonlinear joint expectations. *Let* $\mathbf{v}$ *be a length-*G *random vector with statistically independent components and let* $(i_1, \ldots, i_l)$ *be a length-*l *tuple of indices. Consider a subset* $\{v_{i_1}, \ldots, v_{i_l}\}$ *of* l *components of* $\mathbf{v}$*, where each index* $1 \leq i_k \leq G$ *indexes a component of* $\mathbf{v}$*. In addition, let* $g_1(v_{i_1}), \ldots, g_l(v_{i_l})$ *be a set of* $l \leq G$ *smooth functions from* $\mathbb{R}\,(\mathbb{C})$ *to* $\mathbb{R}\,(\mathbb{C})$ *of which the expectations exist. Then, the following property is satisfied:*

$$E\left\{\prod_{m=1}^{l} g_m\big(v_{i_m}\big)\right\} = \prod_{m=1}^{l} E\left\{g_m\big(v_{i_m}\big)\right\} \qquad \forall\,(i_1, \ldots, i_l) \in \mathcal{I}_{\mathrm{d},G}^{l}\,, \quad \forall\, l \in \mathbb{N}\,. \tag{2.7.23}$$

The notation $\mathcal{I}_{\mathrm{d},G}^{l}$ *denotes the set of all length-*l *tuples of indices with* $1 \leq i_1, \ldots, i_l \leq G$ *that are all different* $\big($*hence the subscript 'd' in* $\mathcal{I}_{\mathrm{d},G}^{p}\big)$*, e.g.* $(i_1, \ldots, i_l) = (1, 2, \ldots, l)$*.*

For properly chosen (sets of) functions $g_1(\cdot), \ldots, g_l(\cdot)$, this criterion can serve as a useful characterization of statistical independence in MIBI and related problems. Virtually all applications described in the literature that are based on Theorem 2.7.4 only consider the simplest possible case, viz. $l = 2$; see [49, 50, 91] and [51], for example. Because one of our main purposes in this thesis is providing conceptual insight, we consider and formulate some ideas that are more general than the ones described in the literature. Note that the left hand side of (2.7.23) can be regarded as a kind of *nonlinear moment* if some or all of the involved functions are nonlinear. In the sequel of this section, we make Theorem 2.7.4 more corporeal by considering some possible choices for the functions $g_1(\cdot), \ldots, g_l(\cdot)$. In addition, our purpose is to give some feeling for how these functions should be chosen properly for blind problems. Then, in the next section we will explain two algorithms that are based on the theorem.

To start with, assume that $g_1(v) = \cdots = g_l(v) = v$, i.e. all functions are the same and equal to the odd linear function $g(v) = v$; then, (2.7.23) becomes:

$$E\left\{\prod_{m=1}^{l} v_{i_m}\right\} = \prod_{m=1}^{l} E\left\{v_{i_m}\right\} \qquad \forall\,(i_1, \ldots, i_l) \in \mathcal{I}_{\mathrm{d},G}^{l}\,, \quad \forall\, l \in \mathbb{N}\,. \tag{2.7.24}$$

The term at the left hand side is precisely the joint moment $r^{v}_{i_1 \cdots i_l} = \mathrm{mom}\big(v_{i_1}, \ldots, v_{i_l}\big) \triangleq E\{v_{i_1} \cdots v_{i_l}\}$ of $v_{i_1}, \ldots, v_{i_l}$ defined in (B.2.28) on page 438. If we also adopt the assumption that at least one of the random variables $v_{i_1}, \ldots, v_{i_l}$ has zero mean, which is common in many signal processing applications, it is clear that the right hand side of (2.7.24) equals zero. Hence, in this case characterization (2.7.24) states that $\mathrm{mom}\big(v_{i_1}, \ldots, v_{i_l}\big) = 0$ for all tuples $(i_1, \ldots, i_l)$ with different indices. This result is very similar to (2.7.22), but in this case the set of allowed index tuples is much smaller. This is one of the reasons that employing joint cumulants usually is preferred over employing joint moments. However, by allowing equal indices in Theorem 2.7.4, a more general result can be obtained: we will show that $\mathrm{mom}\big(v_{i_1}, \ldots, v_{i_l}\big) = 0$ for all index tuples that contain at least one index value that occurs an odd number of times. For example, if $(i_1, i_2, i_3, i_4) = (4, 2, 2, 3)$, $\mathrm{mom}\big(v_4, v_2, v_2, v_3\big) = E\{v_4\}(E\{v_2\})^2 E\{v_3\} = 0$ because $E\{v_4\}$ and $E\{v_3\}$ are assumed to be zero. On the other

hand, when $(i_1, i_2, i_3, i_4) = (4, 2, 2, 4)$, $\operatorname{mom}\big(v_4, v_2, v_2, v_4\big) = (E\{v_2\})^2 (E\{v_4\})^2 \neq 0$ because the variances $(E\{v_2\})^2$ and $(E\{v_4\})^2$ are non-zero. Denoting by $\mathcal{I}^l_{\text{o},G}$ (see (A.2.19)) the set of all index tuples contained in $\mathcal{I}^l_{\text{t},G} \triangleq \{(i_1, \ldots, i_l) \mid 1 \leq i_1, \ldots, i_l \leq G\}$ (see (A.2.14)) that have at least one index value that occurs an odd number of times, we have the following result:

$$\operatorname{mom}\big(v_{i_1}, \ldots, v_{i_l}\big) = 0 \qquad \forall\, (i_1, \ldots, i_l) \in \mathcal{I}^l_{\text{o},G}\,. \tag{2.7.25}$$

The similarity to (2.7.22) is now obvious. As explained in Section B.2.6.7, both in theory and practice it is preferable to work with (2.7.22) than with (2.7.25) because cumulants have many nice properties not shared by moments.

Secondly, we consider the scenario where at least one of the functions $g_1(\cdot), \ldots, g_l(\cdot)$ in (2.7.23), say the k-th, is odd. In addition, we adopt the assumption that the pdf of the corresponding variable v_{i_k} is even, which is another common assumption in signal processing. Then, $E\left\{g_k\big(v_{i_k}\big)\right\} = 0$ and the right hand side of (2.7.23) is zero. Hence, (2.7.23) becomes:

$$E\left\{\prod_{m=1}^{P} g_m\big(v_{i_m}\big)\right\} = 0 \qquad \forall\, (i_1, \ldots, i_l) \in \mathcal{I}^l_{\text{d},G}\,, \quad \forall\, l \in \mathbb{N}\,. \tag{2.7.26}$$

Like properties (2.7.23)-(2.7.25), this result is also a characterization of statistical independence, but does not necessarily imply it. However, it can be shown that under some additional assumptions, satisfaction of this property or of (2.7.23)-(2.7.25) almost surely guarantees statistical independence [69]. We will not pursue this issue here, but briefly highlight the application of (2.7.26) to MIBI and related problems. Considering the IBSS problem for example, we know that the source signals satisfy properties (2.7.23)-(2.7.26) for properly chosen functions because they are statistically independent. Therefore, we also know that the de-mixing matrix $\mathbf{W}$ should be designed in such a way that the components of the output vector $\mathbf{y}$ satisfy the same properties. A specific example for $l = 2$ of the application of (2.7.26) to IBSS is described in the literature, see [49, 51, 91] and [50]. Usually, the two functions $g_1(\cdot)$ and $g_2(\cdot)$ in (2.7.26) are called $f(\cdot)$ and $g(\cdot)$ respectively, i.e. $g_1(\cdot) = f(\cdot)$ and $g_2(\cdot) = g(\cdot)$, and at least one of them is nonlinear. Property (2.7.26) can now be written as:

$$E\left\{f\big(y_i\big)g\big(y_j\big)\right\} = 0 \qquad \forall\, 1 \leq i \neq j \leq S\,. \tag{2.7.27}$$

For obvious reasons, the quantity on the left hand side is often called *nonlinear correlation*. In general, it is hoped and/or expected that by zeroing the nonlinear correlations between each possible pair of different output signals statistical independence is obtained. This is reasonable because it can easily be shown that nonlinear correlation implicitly involves HOS due to the nonlinearity of the functions. See [80] and [91] for considerations and explanations of this problem, and [49, 51, 91] and [50] for possible choices of the functions. In the next section, we will discuss a few approaches from the literature to the actual solution of the system of equations (2.7.27).

We conclude this section by briefly sketching an idea for generalizing the approach in the previous paragraph, which to our knowledge is not described in the literature. In principle, we can exploit (2.7.23) by minimizing the (squared) absolute difference between both sides as a function of the elements of the de-mixing matrix $\mathbf{W}$:

$$\hat{\mathbf{W}} = \operatorname*{argmin}_{\mathbf{W}} \left| E\left\{\prod_{m=1}^{l} g_m\big(v_{i_m}\big)\right\} - \prod_{m=1}^{l} E\left\{g_m\big(v_{i_m}\big)\right\} \right|.$$

As with any method involving nonlinear functions, it may be very difficult in practice to find the estimate $\hat{\mathbf{W}}$ of $\mathbf{W}$ for this problem. The most obvious method for minimization is to use a gradient descent algorithm, which is easy to derive in this case. However, it usually cannot be guaranteed that the algorithm will not converge to a local minimum.

2.7.3.3 Algorithms based on nonlinear de-correlation

2.7.3.3.1 The Hérault-Jutten algorithm Hérault and Jutten were among the first researchers that provided a solution for the instantaneous blind signal separation problem [55, 91]. Inspired by certain capabilities of the nervous system, they designed the neural feedback system for blind signal separation depicted in Fig. 2.2. Their approach is based on nonlinear de-correlation and works for square mixing systems only. For the sake of clarity, we assume that the mixing system is real-valued and that no sensor noise is present. The output signal is defined as $\mathbf{y} = \mathbf{x} - \mathbf{C}\mathbf{y}$, where $\mathbf{C}$ is a matrix with zero diagonal elements. Assuming that the involved inverse exists, the input-output mapping of the neural network is given by:

$$\mathbf{y} = (\mathbf{I} + \mathbf{C})^{-1}\mathbf{x}\,.$$

Hence, in terms of our notation the de-mixing system $\mathbf{W}$ is defined as $\mathbf{W} \triangleq (\mathbf{I} + \mathbf{C})^{-1}$. If $\mathbf{W}$ equals the ideal solution $\mathbf{A}^{-1}$, the equality $\mathbf{I} + \mathbf{C} = \mathbf{A}$ holds. Since $\mathbf{C}$ has zero diagonal elements, the diagonal elements of $\mathbf{A}$ (or a scaled and permuted version of it) are implicitly assumed to be one, thereby handling the scaling indeterminacy discussed in Section 2.4. In order to obtain a separating solution, Hérault and Jutten devised a rule that tried to adapt the off-diagonal elements of $\mathbf{C}$ in such a way that the non-linear correlation between the output signals becomes zero for certain functions $f(\cdot)$ and $g(\cdot)$ as formulated in (2.7.27) of the previous section. They proposed the following separation learning rule for achieving this:

$$c_i^j[k+1] = c_i^j[k] + \mu[k] f\big(y_i[k]\big) g\big(y_j[k]\big) \qquad \forall\, 1 \leq i \neq j \leq S\,. \tag{2.7.28}$$

where $f(\cdot)$ and $g(\cdot)$ are odd nonlinear functions usually chosen as $f(y) = (y)^3$ and $g(y) = \arctan(y)$. These functions have to be different, otherwise the matrix $\mathbf{C}$ would be constrained to be symmetric. Because the source signals are assumed to be zero mean and symmetrically distributed, and $f(\cdot)$ and $g(\cdot)$ are odd, $E\{f\big(y_i[k]\big)\} = 0$ and $E\{g\big(y_j[k]\big)\} = 0$ (see also the similar discussion in the previous section). This ensures that the adaptation rule in (2.7.28) does not change the coefficients when the signals have become independent because for independent output signals the following property is satisfied:

$$E\left\{f\big(y_i\big) g\big(y_j\big)\right\} = E\{f\big(y_i[k]\big)\} E\{g\big(y_j[k]\big)\} = 0 \qquad \forall\, 1 \leq i \neq j \leq S\,. \tag{2.7.29}$$

However, property (2.7.29) itself does not imply that an algorithm using learning rule (2.7.28) will actually converge to a solution. Indeed, several convergence problems have been reported for this algorithm [145]. Supposing for the moment that the rule does converge, and assuming that the source signals are non-Gaussian, it is reasonable to expect a certain degree of

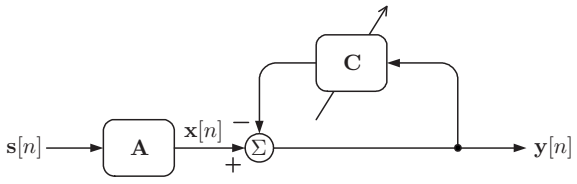

Figure 2.2: *Feedback system for Hérault-Jutten algorithm.*

independence between the signals when this equation is satisfied. This is the rationale behind the Hérault-Jutten algorithm. For more details, see the previous section and [55,91,145]. The Hérault-Jutten algorithm for performing IBSS based on nonlinear de-correlation is summarized in the following Alg. 2.2.

Algorithm 2.2 Hérault-Jutten algorithm for IBSS based on zeroing nonlinear de-correlation.

Initialize $\mathbf{C}[k]$*, then iterate over available samples:*

1: *Compute output signals by* $\mathbf{y}[k] = \mathbf{x}[k] - \mathbf{C}[k]\mathbf{y}[k]$*;*

2: *Update the elements* $\mathbf{C}[k]$ *according to learning rule* (2.7.28)*:*

$$c_i^j[k+1] = c_i^j[k] + \mu[k] f\big(y_i[k]\big) g\big(y_j[k]\big) \qquad \forall\, 1 \leq i \neq j \leq S\,.$$

2.7.3.3.2 Cichocki-Unbehauen algorithm Based on the Hérault-Jutten algorithm, Cichocki, Unbehauen, et. al. [49], [51] and [50] proposed an extension that yielded a significantly enhanced performance. Contrary to Hérault and Jutten, they adapted the conventional *feedforward* de-mixing system $\mathbf{W}$ directly with the following learning rule:

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k]\Big(\mathbf{\Lambda} - \mathbf{f}\big(\mathbf{y}[k]\big)\mathbf{g}^T\big(\mathbf{y}[k]\big)\Big)\mathbf{W}[k]\,, \tag{2.7.30}$$

where

$$\mathbf{f}(\mathbf{y}) \triangleq \begin{bmatrix} f(y_1) \\ \vdots \\ f(y_D) \end{bmatrix} \qquad \text{and} \qquad \mathbf{g}(\mathbf{y}) \triangleq \begin{bmatrix} g(y_1) \\ \vdots \\ g(y_D) \end{bmatrix}.$$

Here, $\mathbf{\Lambda}$ is a diagonal matrix (often chosen as the identify matrix) whose elements determine the amplitude scaling of the output signals, and the functions $f(\cdot)$ and $g(\cdot)$ are two nonlinear scalar functions. As for the Hérault-Jutten algorithm, the rationale behind this algorithm is based on nonlinear correlations as measures for independence. It can easily be shown that the output signals are nonlinearly de-correlated at the stationary points of the learning rule. Since after convergence the update part must be zero on average, it follows that:

$$E\Big\{\mathbf{\Lambda} - \mathbf{f}\big(\mathbf{y}[k]\big)\mathbf{g}^T\big(\mathbf{y}[k]\big)\Big\} = \mathbf{\Lambda} - E\Big\{\mathbf{f}\big(\mathbf{y}[k]\big)\mathbf{g}^T\big(\mathbf{y}[k]\big)\Big\} = 0\,,$$

where the trivial solution $\mathbf{W} = \mathbf{0}$ is excluded. This equation directly implies (2.7.27) for the off-diagonal elements. Hence, when the algorithm has converged the output signals are nonlinearly de-correlated. Again, this does not imply the reverse, viz. that the algorithm will actually converge. Finally, note the striking resemblance of (2.7.30) to (2.7.19). In fact, setting $f_1(y) = \cdots = f_D(y) \triangleq f(y)$ in (2.7.19), $g(y) = y$, and $\mathbf{\Lambda} = \mathbf{I}$ in (2.7.30), the rules are exactly the same. This is reminiscent of the fact that both algorithms are based on the same assumptions of statistical independence and non-Gaussianity. The Cichocki-Unbehauen algorithm for performing IBSS based on nonlinear de-correlation is summarized in the following Alg. 2.3.

Algorithm 2.3 Cichocki-Unbehauen algorithm for IBSS based on zeroing nonlinear de-correlation.

Initialize $\mathbf{W}[k]$*, then iterate over available samples:*

1: *Compute output signals by* $\mathbf{y}[k] = \mathbf{W}[k]\mathbf{x}[k]$*;*

2: *Update* $\mathbf{W}[k]$ *according to learning rule* (2.7.30)*:*

$$\mathbf{W}[k+1] = \mathbf{W}[k] + \mu[k]\Big(\mathbf{\Lambda} - \mathbf{f}\big(\mathbf{y}[k]\big)\mathbf{g}^T\big(\mathbf{y}[k]\big)\Big)\mathbf{W}[k]\,.$$

2.7.3.4 Measuring non-Gaussianity by kurtosis

In this section we consider the use of a statistical characterization called *normalized kurtosis*, or simply *kurtosis*, for measuring non-Gaussianity in IBSS/ICA problems [47, 79, 80, 83]. It is well known that the higher order cumulants of a Gaussian random variable are zero (see *UCP2* on page 425). Hence, (a combination of) the absolute values of one or more higher order cumulants can be used as a measure of non-Gaussianity. In order to recover one source at a time (this is called Blind Signal Extraction [47], see Section 2.7.1), a row vector $\tilde{\mathbf{w}}$ should be designed is such a way that the absolute values of one or more higher order cumulants of $y \triangleq \tilde{\mathbf{w}}\mathbf{x}$ are maximized. Often, only one cumulant order is considered, in particular the fourth. Instead of directly using the fourth order cumulant, a closely related quantity called normalized kurtosis is used. The normalized kurtosis of a random variable v is denoted by $\mathrm{kurt}(v)$ and defined as follows (B.1.17):

$$\mathrm{kurt}(v) \triangleq E\left\{\left(\frac{v - E\{v\}}{\sigma^v}\right)^4\right\} - 3\,. \tag{2.7.31}$$

In the following, we will use the terms kurtosis and normalized kurtosis interchangeably. As is explained in Section B.1.3.3, *the kurtosis is a measure of the non-Gaussianity of a random variable and measures the flatness or peakedness of a distribution w.r.t. a Gaussian distribution*. It is normalized w.r.t. the Gaussian distribution in the sense that it is zero for a Gaussian distribution only.

As has been explained in Section 2.7.1, several IBSS/ICA methods are based on the maximization of non-Gaussianity. Many of these methods work with whitened data [47]. Therefore, in the remainder of this section we will assume that the sensor signals have been whitened. See Section 2.6 for details and associated notation. In order to recover the source signals, the non-Gaussianities of the components of $\mathbf{y} = \mathbf{W}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}}$ have to be maximized, see (2.6.6). As has been mentioned in Section 2.7.1, with this approach it is possible to extract the source signals one at a time (BSE). To get an idea of the rationale behind such methods, consider the kurtosis of one output signal $y \triangleq \tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}}$, where $\tilde{\mathbf{w}}_{\mathrm{o}}$ is one row of $\mathbf{W}_{\mathrm{o}}$ and $\mathbf{y}_{\mathrm{w}}$ contains the whitened sensor signals. Because $\|\tilde{\mathbf{w}}_{\mathrm{o}}\| = 1$, the variance of y is also 1 and the kurtosis of y is given by:

$$\mathrm{kurt}(y) \triangleq E\big\{(y)^4\big\} - 3 = E\big\{(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})^4\big\} - 3\,. \tag{2.7.32}$$

Based on this, a valid objective function can be defined as follows:

$$J(\tilde{\mathbf{w}}_{\mathrm{o}}) \triangleq |\mathrm{kurt}(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})| = \left|E\big\{(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})^4\big\} - 3\right|\,. \tag{2.7.33}$$

Maximizing (2.7.33) yields the solution for $\tilde{\mathbf{w}}_{\mathrm{o}}$ that provides the maximum non-Gaussian signal y. A practical gradient algorithm for performing this maximization is described in the next section. The other source signals can be extracted iteratively after removing the contributions due to y from the sensor signals and/or statistics by a deflation method, see Section 2.7.1 and [81].

2.7.3.5 Gradient algorithm based on kurtosis

A stochastic gradient algorithm that tries to find the row vector $\tilde{\mathbf{w}}_{\mathrm{o}}$ that maximizes the objective function in (2.7.33) is presented in [80] . As in Section 2.7.2.7, we first compute the derivative of objective function (2.7.33):

$$\frac{\partial J(\tilde{\mathbf{w}}_{\mathrm{o}})}{\partial \tilde{\mathbf{w}}_{\mathrm{o}}} = \frac{\partial \left|\mathrm{kurt}(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})\right|}{\partial \tilde{\mathbf{w}}_{\mathrm{o}}} = 4\,\mathrm{sign}(\mathrm{kurt}(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}}))\, E\big\{(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})^3\, \mathbf{y}_{\mathrm{w}}^T\big\}\,. \tag{2.7.34}$$

Hence, the following gradient ascent algorithm is obtained:

Algorithm 2.4 Gradient ascent algorithm for IBSS based on kurtosis.

Initialize $\tilde{\mathbf{w}}_{\mathrm{o}}[k]$, then iterate over available samples of whitened signal vector $\mathbf{y}_{\mathrm{w}}$:

1: *Compute output signal by $y[k] = \tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k]$;*

2: *Update $\tilde{\mathbf{w}}_{\mathrm{o}}[k]$ according to:*

- $\tilde{\mathbf{w}}_{\mathrm{o}}[k+1] = \tilde{\mathbf{w}}_{\mathrm{o}}[k] + \mu[k]\,\mathrm{sign}\big(\mathrm{kurt}(\tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k])\big)\, E\big\{(\tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k])^3\, \mathbf{y}_{\mathrm{w}}^T[k]\big\}$;
- $\tilde{\mathbf{w}}_{\mathrm{o}}[k+1] = \dfrac{\tilde{\mathbf{w}}_{\mathrm{o}}[k+1]}{\|\tilde{\mathbf{w}}_{\mathrm{o}}[k+1]\|}$.

The normalization step ensures that the unit-norm property of $\tilde{\mathbf{w}}_{\mathrm{o}}$ is preserved. As we have explained in Section 2.5, a practical online version of this algorithm can be obtained by disregarding the right-most expectation operator in the update term. According to [80], we may not simply use the instantaneous value of the kurtosis $\mathrm{kurt}(\tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k])$ in the algorithm above, but a more reliable estimate $\widehat{\mathrm{kurt}}(\tilde{\mathbf{w}}_{\mathrm{o}}\mathbf{y}_{\mathrm{w}})$ that can be obtained by averaging over different samples; this can be done under certain (quasi-)stationarity and ergodicity assumptions (see Sections B.1.3.6 and B.2.6.9). Substituting this estimate for the argument of the sign function, the update step of the algorithm above becomes:

$$\tilde{\mathbf{w}}_{\mathrm{o}}[k+1] = \tilde{\mathbf{w}}_{\mathrm{o}}[k] + \mu[k]\,\mathrm{sign}\Big(\widehat{\mathrm{kurt}}(\tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k])\Big)(\tilde{\mathbf{w}}_{\mathrm{o}}[k]\mathbf{y}_{\mathrm{w}}[k])^3\, \mathbf{y}_{\mathrm{w}}^T[k]\,, \tag{2.7.35}$$

See [79, 80, 83] and [47] for more information.

2.7.4 Algebraic and iterative algorithms based on cross-cumulants

In this section, we will elaborate on the main principles of using joint cross-cumulants as measures of statistical independence (see Section 2.7.3.1), and discuss several practical algorithms. We confine ourselves to fourth order cumulants because these are employed by most methods described in the literature. Some methods are based on 'purely' algebraic approaches that are formulated in terms of (generalized) eigendecompositions, tensor decompositions, etc. Others are based on iterative approaches, where some objective function is optimized iteratively. Methods in this category typically use a cost function that penalizes the cross-cumulants between the output signals.

2.7.4.1 Special structure of fourth order sensor cumulants

All methods based on cumulants implicitly or explicitly exploit the special algebraic structure of the sensor cumulants [26,28,32,36,53]. This structure can be derived from the fact that all fourth order cross-cumulants of the source signals are zero (see Definition 2.7.3 with $l = 4$) by employing some convenient properties of cumulants (see Section B.2.6.7 in Appendix B and [111]). For clarity, we will consider real-valued systems and signals here; the derivation for the complex case proceeds similarly (see Section 3.3, for example). In addition, we assume that the source auto-cumulants are non-zero and that there is no sensor noise. Using the notation explained in Section B.2.6.5 and Appendix A, the assumptions on the lag-zero source cross- and auto-cumulants can be written as follows:

$$\kappa^s_{\mathbf{j}_4} \equiv \kappa^s_{j_1 j_2 j_3 j_4} \triangleq \mathrm{cum}\big(s_{j_1}, s_{j_2}, s_{j_3}, s_{j_4}\big) = 0 \qquad \forall\, (j_1, j_2, j_3, j_4) \in \mathcal{J}^4_{\mathrm{c},D}\,, \tag{2.7.36}$$

and

$$\kappa^s_{(j)_4} \equiv \kappa^s_{jjjj} \triangleq \mathrm{cum}\big(s_j, s_j, s_j, s_j\big) \neq 0 \qquad \forall\, 1 \leq j \leq D \tag{2.7.37}$$

respectively. The notation $\mathbf{j}_4$ in (2.7.36) denotes the tuple (j_1, j_2, j_3, j_4) and $\kappa^s_{\mathbf{j}_4} \equiv \kappa^s_{j_1 j_2 j_3 j_4}$ denotes the fourth order joint cross-cumulant of the j_1-th, j_2-th, j_3-th and j_4-th source signals respectively. Furthermore, $\mathcal{J}^4_{\mathrm{c},D}$ denotes the set of all possible length-4 'cross-tuples' of indices ranging from 1 till D (A.2.18). Likewise, the notation $(j)_4$ in (2.7.37) denotes the tuple (j, j, j, j) and $\kappa^s_{(j)_4} \equiv \kappa^s_{jjjj}$ denotes the fourth order auto-cumulant of the j-th source signal. Note that $(j)_4$ can be seen as the fourth order 'auto-tuple' defined by the index j; see also (A.2.2). Using (2.7.36) and cumulant property *CP7* on page 443, all fourth order sensor cumulants $\{\kappa^x_{\mathbf{i}_4}\}_{\mathbf{i}_4 \in \mathcal{I}^4_{\mathrm{t},D}} = \{\kappa^x_{i_1 i_2 i_3 i_4}\}_{(i_1,i_2,i_3,i_4) \in \mathcal{I}^4_{\mathrm{t},D}}$ (the 't' in $\mathcal{I}^4_{\mathrm{t},D}$ stands for 'total', see (A.2.14) and Appendix A for the definition and more information respectively) can now be expressed in terms of the elements of the mixing matrix and the fourth order source auto-cumulants as follows:

$$\kappa^x_{i_1 i_2 i_3 i_4} = \sum_{j=1}^{D} a^j_{i_1} a^j_{i_2} a^j_{i_3} a^j_{i_4} \kappa^s_{(j)_4} \qquad \forall\, (i_1, i_2, i_3, i_4) \in \mathcal{I}^4_{\mathrm{t},D}\,, \tag{2.7.38}$$

or, equivalently:

$$\mathrm{cum}\big(x_{i_1}, x_{i_2}, x_{i_3}, x_{i_4}\big) = \sum_{j=1}^{D} a^j_{i_1} a^j_{i_2} a^j_{i_3} a^j_{i_4}\, \mathrm{cum}\big(s_j, s_j, s_j, s_j\big) \qquad \forall\, (i_1, i_2, i_3, i_4) \in \mathcal{I}^4_{\mathrm{t},D}\,. \tag{2.7.39}$$

The product terms of the form $a^j_{i_1} a^j_{i_2} a^j_{i_3} a^j_{i_4}$ are made up from a single vector, viz. the j-th column $\mathbf{a}^j$ of $\mathbf{A}$, and are usually called *tetrads*[‡]. All possible fourth order sensor cumulants can be stored in a four-dimensional array indexed by four the indices i_1, i_2, i_3 and i_4. Such a higher-dimensional array, which is a generalization of a matrix, is often called *tensor*. Accordingly, the right hand side of (2.7.38) represents a (tetradic) *tensor decomposition*. More specifically, the fourth order tensor decomposition in (2.7.38) is called *D-tetradic decomposition* because it consists of a linear combination of D tetrads [32, 32, 36, 53].

Although tables with more than two dimensions are usually called tensors in signal processing, mathematically speaking this is not correct. The correct name for an array with an arbitrary number of dimensions is *holor* [60]. For example, a matrix is a two-dimensional holor.

[‡] A *tetrad* is a kind of fourth order outer product of a vector with itself; this is in analogy with a *dyad*, which is the outer product of a vector with itself and defines a rank-1 matrix.

A tensor is defined as a mapping or function that is linear in each of its arguments. It satisfies a multilinear transformation law when a linear transform is applied to any vector of the underlying 'input vector space' [60, 111], exactly as in *CP7* on page 443. The coefficients of this multilinear transformation w.r.t. a certain basis are usually stored in a holor. Hence, a holor represents a multilinear transformation w.r.t. a certain basis in exactly the same way as a matrix represents a linear transformation w.r.t. a certain basis. For the purpose of this work, we will not need to make the distinction between a tensor and a holor. Therefore, we will adopt the signal processing convention and refer to a multi-dimensional array by the word 'tensor'. We will use the same notation for tensors as for matrices (see Appendix A), i.e. they are denoted by boldface capitalized letters that correspond to the letters used for their coefficients. For example, the fourth order tensor with coefficients $\{v_{i_1 i_2 i_3 i_4}\}_{1 \leq i_1, i_2, i_3, i_4 \leq P}$ of size $P \times P \times P \times P$ is denoted by $\mathbf{V}$. A tensor defined by cumulants will be called *cumulant tensor* in the sequel [36, 38]. Note that the sensor cumulant tensor $\mathbf{K}^x$ defined by the coefficients $\{\kappa^x_{i_1 i_2 i_3 i_4}\}_{1 \leq i_1, i_2, i_3, i_4 \leq D}$ in (2.7.38) is invariant to permutations of the indices. Consequently, it is called symmetric. In the next sections, several algebraic and iterative methods that exploit the structure revealed in (2.7.38) are discussed. We start by discussing some general concepts on which many of these methods are based.

2.7.4.2 Concepts related to fourth order cumulants

Several approaches to ICA that are based on fourth order cumulants have been developed by Cardoso and Comon, who are among the main pioneers of and contributors to ICA [26–30,32, 36,38,52,56]. In this section, we discuss some concepts that have been introduced by Cardoso and that form the basis of several algorithms. In the following sections, the application of these concepts to different algorithms is highlighted. We adopt the same assumptions as in the previous section. Furthermore, it is assumed that the sensor cumulants are known or can be estimated from the available data.

Any 'tetradic' four-dimensional tensor, say $\mathbf{C} = \{c_{i_1 i_2 i_3 i_4}\}_{1 \leq i_1, i_2, i_3, i_4 \leq D}$, can be used to define (element-wise) a linear matrix-to-matrix mapping $\mathbf{T}^c$ by the following contraction operation [36, 38]:

$$[\mathbf{T}^c(\mathbf{M})]^j_i = \sum_{k=1}^{D} \sum_{l=1}^{D} c_{ijkl} m^k_l \qquad \forall\, 1 \leq i, j \leq D, \quad \forall\, \mathbf{M} \in \mathbb{R}^D_D \,. \tag{2.7.40}$$

The coefficients $[\mathbf{T}^c(\mathbf{M})]^j_i$ are the coefficients of the transformation of $\mathbf{M}$ by $\mathbf{T}^c$ w.r.t. the standard orthonormal basis for the linear space of D by D matrices. Note that if the tensor $\mathbf{C}$ is not invariant under permutations of the indices, different linear operators can be defined by contracting over different indices. However, for our purposes it is sufficient to consider only the definition in (2.7.40). The range of the linear mapping is defined as usual:

$$\mathcal{R}\left(\mathbf{T}^c\right) \triangleq \{\mathbf{T}^c(\mathbf{M}) \mid \mathbf{M} \in \mathbb{R}^D_D\}^{\ddagger} \,. \tag{2.7.41}$$

The rank of $\mathbf{T}^c$ is defined as the dimension of its range:

$$\operatorname{rank}(\mathbf{T}^c) \triangleq \dim\big(\mathcal{R}\left(\mathbf{T}^c\right)\big) \,. \tag{2.7.42}$$

Again, other definitions are possible, but this one is sufficient for our purposes.

[‡] See Appendix A and the list of symbols on page ?? for the meaning of the 'range symbol' $\mathcal{R}\left(\cdot\right)$.

Substituting the sensor cumulant tensor $\mathbf{K}^x$ for $\mathbf{C}$ in (2.7.40), i.e. setting $c_{i_1 i_2 i_3 i_4} = \kappa^x_{i_1 i_2 i_3 i_4}$ for all $1 \leq i_1, i_2, i_3, i_4 \leq D$, yields:

$$[\mathbf{T}^x(\mathbf{M})]^j_i = \sum_{k=1}^{D}\sum_{l=1}^{D} \kappa^x_{ijkl} m^k_l = \sum_{k=1}^{D}\sum_{l=1}^{D} \mathrm{cum}\big(x_i, x_j, x_k, x_l\big)\, m^k_l \quad \forall\, 1 \leq i, j \leq D\,. \tag{2.7.43}$$

In the ICA context, the transformed matrix $\mathbf{T}^x(\mathbf{M})$ for an arbitrary matrix $\mathbf{M}$ is often called *cumulant matrix* [36, 38]. Note that $\mathbf{T}^x(\mathbf{M})$ is symmetric due to the symmetry of the sensor cumulant tensor. The structure of $\mathbf{T}^x(\mathbf{M})$ is easily deduced from (2.7.38):

$$[\mathbf{T}^x(\mathbf{M})]^j_i = \sum_{k=1}^{D}\sum_{l=1}^{D}\sum_{p=1}^{D} a^p_i a^p_j a^p_k a^p_l \kappa^s_{(p)_4} m^k_l = \sum_{p=1}^{D}\left(\kappa^s_{(p)_4}\sum_{k=1}^{D}\sum_{l=1}^{D} a^p_k a^p_l m^k_l\right) a^p_i a^p_j$$

$$\forall\, 1 \leq i, j \leq D, \quad \forall\, \mathbf{M} \in \mathbb{R}^D_D\,.$$

In matrix notation, this equation reads:

$$\mathbf{T}^x(\mathbf{M}) = \mathbf{A}\,\mathbf{\Lambda}(\mathbf{M})\mathbf{A}^T \qquad \forall\, \mathbf{M} \in \mathbb{R}^D_D\,, \tag{2.7.44}$$

where

$$\mathbf{\Lambda}(\mathbf{M}) \triangleq \mathrm{diag}\big(\lambda_1(\mathbf{M}), \ldots, \lambda_D(\mathbf{M})\big) \quad \text{and} \quad \lambda_p(\mathbf{M}) \triangleq \kappa^s_{(p)^4}(\mathbf{a}^p)^T\mathbf{M}\mathbf{a}^p\,. \tag{2.7.45}$$

Hence, assuming that $\kappa^s_{(p)^4} \neq 0$ and $\mathbf{a}^p \neq \mathbf{0}$ for all $1 \leq p \leq D$, the range of $\mathbf{T}^x$ is the set of all matrices in the form $\mathbf{A}\mathbf{D}\mathbf{A}^T$ with $\mathbf{D} \in \mathbb{R}^D_D$ an arbitrary diagonal matrix:

$$\begin{aligned}\mathcal{R}\left(\mathbf{T}^x\right) &\triangleq \left\{\mathbf{A}\mathbf{D}\mathbf{A}^T \middle| \mathbf{D} = \mathrm{diag}(\mathbf{D})\right\} = \left\{\sum_{j=1}^{D} d_j \mathbf{a}^j(\mathbf{a}^j)^T \,\middle|\, d_1, \ldots, d_D \in \mathbb{R}\right\} \\ &= \mathcal{L}\left(\mathbf{a}^1(\mathbf{a}^1)^T, \ldots, \mathbf{a}^D(\mathbf{a}^D)^T\right) = \mathcal{L}\left(\{\mathbf{a}^j(\mathbf{a}^j)^T\}^{1\leq j\leq D}\right)\,. \end{aligned} \tag{2.7.46}$$

The j-th element $\mathbf{a}^j(\mathbf{a}^j)^T$ of the set of 'vectors' $\{\mathbf{a}^j(\mathbf{a}^j)^T\}^{1\leq j\leq D}$ that span $\mathcal{R}\left(\mathbf{T}^x\right)$ is the orthogonal projector $\mathbf{\Pi}^j \triangleq \mathbf{a}^j(\mathbf{a}^j)^T$ onto the one-dimensional subspace spanned by the j-th column $\mathbf{a}^j$ of the mixing matrix. The range $\mathcal{R}\left(\mathbf{T}^x\right)$ is termed *Fourth Order Signal Subspace (FOSS)* by Cardoso [28, 30, 36]. From (2.7.46) it easily follows that if the projectors $\mathbf{\Pi}^1, \ldots, \mathbf{\Pi}^D$ are linearly independent, the dimension of the FOSS is D, i.e. $\dim(\mathcal{R}\left(\mathbf{T}^x\right)) = D$. The latter condition will be assumed in the rest of this section.

An orthonormal basis for the range of $\mathbf{T}^x$ can be obtained from an Eigenvalue Decomposition (EVD) of $\mathbf{T}^x$ [27, 30, 36]. Note that $\mathbf{T}^x$ has an orthonormal EVD because it is a linear symmetric operator. By definition, an eigenmatrix of $\mathbf{T}^x$ is a matrix $\mathbf{E}$ such that:

$$\mathbf{T}^x(\mathbf{E}) = \lambda\mathbf{E}\,, \tag{2.7.47}$$

where λ is a real-valued scalar eigenvalue. It can be shown [27] that there exists a system of $(D)^2$ symmetric eigenmatrices $\mathbf{E}^1, \ldots, \mathbf{E}^{(D)^2}$ of size $D \times D$, and a set of corresponding real-valued eigenvalues $\lambda_1, \ldots, \lambda_{(D)^2}$, satisfying:

$$\begin{aligned} &\bullet\ \mathbf{T}^x(\mathbf{E}^k) = \lambda_k\mathbf{E}^k \quad \forall\, 1 \leq k \leq (D)^2\,; \\ &\bullet\ \mathrm{tr}(\mathbf{E}^k(\mathbf{E}^l)^T) = \delta^k_l \quad \forall\, 1 \leq k, l \leq (D)^2\,, \end{aligned} \tag{2.7.48}$$

where $\mathrm{tr}(\mathbf{B}\mathbf{C}^T)$ denotes the standard inner product between the matrices $\mathbf{B}$ and $\mathbf{C}$ defined in Appendix A using the matrix 'trace' operator $\mathrm{tr}(\cdot)$. In practice, the EVD of $\mathbf{T}^x$ can be computed by considering the tensor $\mathbf{K}^x = \{\kappa^x_{i_1 i_2 i_3 i_4}\}_{1\leq i_1, i_2, i_3, i_4 \leq D}$ as a matrix in the space

of $D \times D$ matrices, and vectorizing the involved matrices by stacking their columns on top of each other. This way, the tensor is represented by a $(D)^2 \times (D)^2$ matrix, whose eigenvectors and eigenvalues can be computed by conventional tools. By unstacking the eigenvectors into matrices, the desired eigenmatrices are obtained. Since $\dim(\mathcal{R}(\mathbf{T}^x)) = D$, only D out of $(D)^2$ eigenvalues are non-zero. Arranged in decreasing order (as is conventional), the eigenvalues are denoted by $\lambda_1, \ldots, \lambda_D$. Hence, the corresponding eigenmatrices form an orthonormal basis of the FOSS:

$$\text{FOSS} = \mathcal{R}(\mathbf{T}^x) = \mathcal{L}\left(\mathbf{a}^1(\mathbf{a}^1)^T, \ldots, \mathbf{a}^D(\mathbf{a}^D)^T\right) = \mathcal{L}\left(\mathbf{E}^1, \ldots, \mathbf{E}^D\right). \tag{2.7.49}$$

The splitting of the FOSS into two subspaces, viz. one with the D largest eigenvalues and one with the $(D)^2 - D$ smallest eigenvalues, is reminiscent of the *subspace approach*, which plays a major role in this thesis and is discussed in Chapter 3. The structure and properties of the FOSS can be exploited in different ways for estimating the mixing matrix $\mathbf{A}$ and/or the source signals, some of which are illustrated in the next sections.

2.7.4.3 Approach based on Eigenvalue Decomposition

In this section, we discuss a simple approach that is based on the whitened observations $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$, where $\mathbf{Q} \in \mathbb{R}_D^D$ (2.6.5). As has been explained in Section 2.6.2, after whitening the blind problem comes down to determining the unique orthogonal matrix $\mathbf{Q} = [\mathbf{q}^1 \cdots \mathbf{q}^D]$ in (2.6.5) and (2.6.6) (or its inverse) that relates the sources to the whitened observations. In [38], Cardoso showed that in principle this can be done by performing an Eigenvalue Decomposition (EVD) of the cumulant matrix $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$ of $\mathbf{y}_{\text{w}}$ for some properly chosen $\mathbf{M}$. Indeed, by considering $\mathbf{y}_{\text{w}}$ instead of $\mathbf{x}$ and $\mathbf{Q}$ instead of $\mathbf{A}$ in (2.7.44), it directly follows that:

$$\mathbf{T}^{y_{\text{w}}}(\mathbf{M}) = \mathbf{Q}\,\mathbf{\Lambda}(\mathbf{M})\mathbf{Q}^T \qquad \forall\, \mathbf{M} \in \mathbb{R}_D^D, \tag{2.7.50}$$

where

$$\mathbf{\Lambda}(\mathbf{M}) \triangleq \text{diag}\big(\lambda_1(\mathbf{M}), \ldots, \lambda_D(\mathbf{M})\big) \quad \text{and} \quad \lambda_p(\mathbf{M}) \triangleq \kappa^s_{(p)^4}(\mathbf{q}^p)^T\mathbf{M}\mathbf{q}^p\,. \tag{2.7.51}$$

Since $\mathbf{Q}$ is orthogonal, the right hand side of (2.7.50) represents the EVD of $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$. This reveals the basic idea of EVD-based identification of $\mathbf{Q}$: compute the EVD of $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$ for some randomly chosen $\mathbf{M}$; then the resulting eigenvectors correspond to the columns of $\mathbf{Q}$ if the eigenvalues are different. If the eigenvalue decomposition is degenerate, i.e. if the eigenvalues are equal, a different $\mathbf{M}$ has to be chosen. Although in practice this is very unlikely to happen, $\mathbf{M}$ might be such that the problem is ill-conditioned. However, for the moment we assume that it is properly chosen. Equation (2.7.50) states that $\mathbf{Q}^T$ is given by the (unique) *orthogonal diagonalizer* of $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$ because $\mathbf{Q}^T\mathbf{T}^{y_{\text{w}}}(\mathbf{M})\mathbf{Q} = \mathbf{\Lambda}(\mathbf{M})$ is diagonal[‡]. The algorithm for performing IBSS based on the eigenvalue decomposition of the sensor cumulant matrix $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$ of the whitened data is summarized in Alg. 2.5. It should be noted that $\mathbf{A}$ is determined 'almost surely' uniquely by $\mathbf{M}$ if the latter matrix is drawn from a continuous probability density function. In other words, although in theory it is possible to choose an $\mathbf{M}$ that yields equal eigenvalues, the choices for $\mathbf{M}$ that yield such degenerate EVD problems have very small probability.

[‡]Sometimes $\mathbf{Q}$ instead of $\mathbf{Q}^T$ is called the orthogonal diagonalizer, but this is just a matter of convention.

Algorithm 2.5 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ based on EVD of cumulant matrix of whitened data.

1: *Compute/estimate sensor correlation matrix* $\mathbf{R}^x \triangleq E\{\mathbf{x}\mathbf{x}^T\}$*;*
2: *Compute whitening matrix* $\mathbf{V}$ *from a square-root decomposition of* $\mathbf{R}^x$*;*
3: *Compute whitened observation vector* $\mathbf{y}_{\text{w}} \triangleq \mathbf{V}\mathbf{x}$ *with* $\mathbf{R}^{y_{\text{w}}} \triangleq E\{\mathbf{y}_{\text{w}}\mathbf{y}_{\text{w}}^T\} = \mathbf{I}$*;*
4: *Compute/estimate all fourth order cumulants of* $\mathbf{y}_{\text{w}}$*, i.e.the set* $\{\kappa^{y_{\text{w}}}_{\mathbf{i}_4}\}_{\mathbf{i}_4 \in \mathcal{I}^4_{\text{t},D}}$*;*
5: *Compute* $\mathbf{T}^{y_{\text{w}}}(\mathbf{M})$ *for some randomly chosen* $\mathbf{M}$*;*
6: *Compute EVD of* $\mathbf{T}^{y_{\text{w}}}(\mathbf{M}) = \mathbf{Q}\,\mathbf{\Lambda}(\mathbf{M})\mathbf{Q}^T$*;*
7: *Compute estimated source signals by* $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ *.*

2.7.4.4 Joint Approximate Diagonalization of Eigenmatrices (JADE)

The idea in the previous section can be extended to multiple matrices [38]. The rationale is to estimate $\mathbf{Q}^T$ as the *joint (orthogonal) diagonalizer* of a set of cumulant matrices $\mathcal{T} \triangleq \{\mathbf{T}^{y_{\text{w}}}(\mathbf{M}_k)\}_{1\le k\le K}$ for K randomly generated matrices in the set $\mathcal{M} \triangleq \{\mathbf{M}_k\}_{1\le k\le K}$. In other words, we want to find $\mathbf{Q}$ or $\mathbf{Q}^T$ such that the matrices $\{\mathbf{Q}^T\mathbf{T}^{y_{\text{w}}}(\mathbf{M}_k)\mathbf{Q}\}_{1\le k\le K}$ are diagonal. This is exactly the idea behind Cardoso's well-known Joint Approximate Diagonalization of Eigenmatrices (JADE) algorithm [38]. The advantages of this approach are that the problem usually becomes better conditioned and that the chance of degenerate or ill-conditioned solutions due to equal eigenvalues (see previous section) is reduced. Since the matrices in the set $\mathcal{T}$ are constructed from sample cumulants (i.e. estimated from data) and thus contain errors, they cannot be exactly jointly diagonalized. For this reason, JADE is based on the approximate joint orthogonal diagonalization of $\mathcal{T}$, which is discussed in the next paragraph.

A natural measure for the approximate diagonality of a matrix is the ratio of the sum of squares of its diagonal elements to the sum of the squares of its off-diagonal elements [38,80]. Since an orthogonal matrix does not change the total sum of squares of a matrix, we have the following result. *If* $\mathbf{Q}$ *is orthogonal (as is assumed), the minimization of the sum of squares of off-diagonal elements of the matrix* $\mathbf{Q}^T\mathbf{T}\mathbf{Q}$ *is equivalent to maximization of the sum of squares of its diagonal elements*. This is the rationale behind the following definition of the objective function used by the JADE algorithm. A joint orthogonal diagonalizer of the set $\mathcal{T}$ is an orthogonal matrix $\mathbf{Q}^T$ (or $\mathbf{Q}$) that maximizes the following objective function:

$$J_{\text{JADE}}(\mathbf{Q};\mathcal{T}) = \sum_{\mathbf{T}\in\mathcal{T}} \left\|\text{diag}\left(\mathbf{Q}^T\mathbf{T}\mathbf{Q}\right)\right\|^2 = \sum_{k=1}^{K} \left\|\text{diag}\left(\mathbf{Q}^T\mathbf{T}^{y_{\text{w}}}(\mathbf{M}_k)\mathbf{Q}\right)\right\|^2. \qquad (2.7.52)$$

Since each term $\left\|\text{diag}\left(\mathbf{Q}^T\mathbf{T}\mathbf{Q}\right)\right\|^2$ is the sum of the squared diagonal elements of $\mathbf{Q}^T\mathbf{T}\mathbf{Q}$, this amounts to simultaneously maximizing the diagonality of all matrices $\mathbf{T} \in \mathcal{T}$. In [38], the Jacobi technique [72] for diagonalizing a single symmetric matrix is extended to the joint approximate diagonalization problem. As is reminiscent of Jacobi methods, (2.7.52) is maximized by Givens rotations; see [38] for details. Since these rotations are applied in an iterative fashion, JADE is an iterative algorithm.

An important issue related to JADE is the method for determining which set of matrices $\mathcal{M} \triangleq \{\mathbf{M}_k\}_{1\le k\le K}$ should be used. This set can be chosen randomly, but there exist more

structured approaches. Since the cumulant matrices $\mathcal{T} \triangleq \{\mathbf{T}^{y_w}(\mathbf{M}_k)\}_{1\leq k\leq K}$ are contracted versions of the matrices in $\mathcal{M}$ (see the definition in (2.7.40)), intuitively it is clear that there is a loss of information if K is chosen too small. A suitable set of $(D)^2$ matrices is able to encompass all fourth order information, but for computational reasons often a smaller set is desirable. In [38], it is shown that a suitable set $\mathcal{M}$ is given by the D eigenmatrices of the operator $\mathbf{T}^{y_w}$ that correspond to non-zero eigenvalues. The actual computation of this so-called eigen-set proceeds similarly to the description at the end of Section 2.7.4.2. In summary, we have the following JADE (batch) algorithm:

Algorithm 2.6 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_w = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_w$ based on JADE.

1-4: Whitening: see steps 1-4 of Alg. 2.5;

5: Determine a suitable set $\mathcal{M} \triangleq \{\mathbf{M}_k\}_{1\leq k\leq K}$*;*

6: Jointly diagonalize the set $\mathcal{T} \triangleq \{\mathbf{T}^{y_w}(\mathbf{M}_k)\}_{1\leq k\leq K}$ *by an orthogonal matrix* $\mathbf{Q}^T$*, i.e. maximize* $J_{\text{JADE}}(\mathbf{Q};\mathcal{T})$ *in* (2.7.52)*;*

7: Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_w$ *.*

There exist several other algorithms that are closely related to JADE, e.g. Fourth Order Blind Identification (FOBI) [26] and Extended Fourth Order Blind Identification (EFOBI) [157]. These are outside the scope of this review. Finally, we note that JADE cannot handle the case of more sources than sensors. This is due to the fact that whitening limits the number of identifiable columns of the mixing matrix to the number of sensors.

2.7.4.5 Approaches based on iteratively optimizing cumulant criteria

Several iterative approaches are based on whitened data and the optimization of an objective function that is defined in terms of the auto- or cross-cumulants of the output signal vector $\mathbf{y} = \mathbf{Q}^T\mathbf{y}_w$, where $\mathbf{Q}$ is unitary (i.e. $\mathbf{Q}^H\mathbf{Q} = \mathbf{Q}\mathbf{Q}^H = \mathbf{I}$). We briefly consider some possible objective functions for complex data without discussing the actual iterative optimization methods. For example, based on a maximum likelihood approach, it has been proposed in [70] to determine $\mathbf{Q}$ as the unitary maximizer of the following objective function:

$$J_a(\mathbf{Q}) \triangleq \sum_{j=1}^{D} \left| \operatorname{cum}\Big(y_j, (y_j)^*, y_j, (y_j)^*\Big)\right|^2, \tag{2.7.53}$$

where '*' denotes complex conjugation. Similarly to the previous section, where $\mathbf{Q}$ was assumed to be orthogonal, the sum of the squared absolute values of the auto-cumulants and cross-cumulants of $\mathbf{y}$ does not depend on $\mathbf{Q}$ as long as $\mathbf{Q}$ is unitary (which is ensured by a complex whitening stage). Therefore, the maximization of (2.7.53) is equivalent to the minimization of the following objective function:

$$J_c(\mathbf{Q}) \triangleq \sum_{\mathbf{j}_4\in\mathcal{J}^4_{c,D}} \left| \operatorname{cum}\Big(y_{j_1}, (y_{j_2})^*, y_{j_3}, (y_{j_4})^*\Big)\right|^2. \tag{2.7.54}$$

Obviously, this cost function penalizes all possible cross-cumulants between outputs. In the light of (2.7.36), this is a very natural and intuitive criterion. Cardoso proposed to determine

$\mathbf{Q}$ as the unitary maximizer of the following function [38]:

$$J(\mathbf{Q}) \triangleq \sum_{j_1=1}^{D}\sum_{j_2=1}^{D}\sum_{j_3=1}^{D}\left|\operatorname{cum}\Big(y_{j_1},(y_{j_1})^*,y_{j_2},(y_{j_3})^*\Big)\right|^2. \tag{2.7.55}$$

This criterion is closely related to the JADE criterion $J_{\text{JADE}}(\mathbf{Q};\mathcal{T})$. In fact, Cardoso shows that for the 'parallel set' of cumulant matrices $\mathcal{T}_p \triangleq \{\mathbf{T}^{y_{\text{w}}}(\mathbf{e}^i(\mathbf{e}^j)^H) \mid 1 \leq i,j \leq D\}$ (see Appendix A for the definition of $\mathbf{e}^i$) the two functions are equal, i.e. $J(\mathbf{Q}) = J_{\text{JADE}}(\mathbf{Q};\mathcal{T}_p)$. The fact that the first two indices in the arguments of the $\operatorname{cum}(\cdot)$ function in (2.7.55) are the same allows an interpretation in terms of an underlying eigenstructure. The algorithm for performing IBSS based on the optimization of objective functions incorporating cumulant criteria is summarized in the following Alg. 2.7.

Algorithm 2.7 Blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\text{w}} = \mathbf{Q}\,\mathbf{s}$ and $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$ based on optimizing objective functions incorporating cumulant criteria.

1-4: Whitening: see steps 1-4 of Alg. 2.5;

5: Find $\mathbf{Q}$ *by optimizing involved objective function:*

$$\hat{\mathbf{Q}} = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmax}}\, J_{\text{a}}(\mathbf{Q}) = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmax}} \sum_{j=1}^{D}\left|\operatorname{cum}\Big(y_j,(y_j)^*,y_j,(y_j)^*\Big)\right|^2,$$

or:

$$\hat{\mathbf{Q}} = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmin}}\, J_{\text{c}}(\mathbf{Q}) = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmin}} \sum_{\mathbf{j}_4\in\mathcal{J}_{\text{c},D}^4}\left|\operatorname{cum}\Big(y_{j_1},(y_{j_2})^*,y_{j_3},(y_{j_4})^*\Big)\right|^2,$$

or:

$$\hat{\mathbf{Q}} = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmax}}\, J(\mathbf{Q}) = \underset{\text{unitary}\mathbf{Q}}{\operatorname{argmax}} \sum_{\mathbf{j}_3\in\mathcal{J}_{\text{c},D}^3}\left|\operatorname{cum}\Big(y_{j_1},(y_{j_1})^*,y_{j_2},(y_{j_3})^*\Big)\right|^2;$$

6: Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^T\mathbf{y}_{\text{w}}$.

2.7.4.6 Approach based on Generalized Eigenvalue Decomposition

It is also possible to develop approaches that estimate the mixing matrix $\mathbf{A}$ directly from the sensor data, i.e. without whitening. A straightforward technique based on cumulant matrices was developed by Cardoso and Comon [36] proceeds as follows. Firstly, $\mathbf{T}^x(\mathbf{M}_1)$ and $\mathbf{T}^x(\mathbf{M}_2)$ are computed for two randomly generated matrices $\mathbf{M}_1$ and $\mathbf{M}_2$. Then, (2.7.44) yields the following system:

$$\begin{cases}\mathbf{T}^x(\mathbf{M}_1) = \mathbf{A}\,\mathbf{\Lambda}(\mathbf{M}_1)\mathbf{A}^T;\\ \mathbf{T}^x(\mathbf{M}_2) = \mathbf{A}\,\mathbf{\Lambda}(\mathbf{M}_2)\mathbf{A}^T.\end{cases} \tag{2.7.56}$$

This system can be solved for $\mathbf{A}$ by using the Generalized Eigenvalue Decomposition (GEVD); see [115], [72] and Section D.3. Write the GEVD of $\mathbf{T}^x(\mathbf{M}_1)$ and $\mathbf{T}^x(\mathbf{M}_2)$ defined in

(2.7.56) in the same form as (D.3.2):

$$\mathbf{T}^x(\mathbf{M}_1)\mathbf{U} = \mathbf{T}^x(\mathbf{M}_2)\mathbf{U}\mathbf{\Sigma}\,, \tag{2.7.57}$$

i.e. $\mathbf{U}$ is the matrix of generalized eigenvectors of the *matrix pencil* $\{\mathbf{T}^x(\mathbf{M}_1) - \sigma\mathbf{T}^x(\mathbf{M}_2) \mid \sigma \in \mathbb{R}\}$, and $\mathbf{\Sigma}$ is the diagonal matrix that contains the corresponding generalized eigenvalues $\sigma(\mathbf{T}^x(\mathbf{M}_1), \mathbf{T}^x(\mathbf{M}_2))$. We will now show that once $\mathbf{U}$ (and $\mathbf{\Sigma}$) has been computed, an estimate of $\mathbf{A}$ can be derived from it. It can easily be verified that a valid solution of (2.7.57) is given by $\mathbf{U} = \mathbf{A}^{-T}, \mathbf{\Sigma} = \mathbf{\Lambda}(\mathbf{M}_2)^{-1}\mathbf{\Lambda}(\mathbf{M}_1)$. Hence, it follows that an estimate of $\mathbf{A}$ is given by $\mathbf{U}^{-T}$. Since this solution is unique up to the indeterminacies inherent to MIBI (see Section 2.4), this is a correct, i.e. separating and waveform-preserving, solution. Note that $\mathbf{A}$ is determined 'almost surely' uniquely by $\mathbf{M}_1$ and $\mathbf{M}_2$ if the latter matrices are drawn from continuous probability density functions. In other words, the choices for $\mathbf{M}_1$ and $\mathbf{M}_2$ that yield a degenerate GEVD problem have zero probability. Also see the comments about this issue in Sections 2.7.4.3 and 2.7.4.4. Summarizing, we have the following simple high-level batch algorithm:

Algorithm 2.8 Blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on the GEVD of two cumulant matrices.

1: *Compute/estimate all fourth order sensor cumulants of* $\mathbf{x}$*, i.e. the set* $\{\kappa^x_{\mathbf{i}_4}\}_{\mathbf{i}_4 \in \mathcal{I}^4_{\mathrm{t},D}}$*;*

2: *Compute* $\mathbf{T}^x(\mathbf{M}_1)$ *and* $\mathbf{T}^x(\mathbf{M}_2)$ *for two randomly chosen matrices* $\mathbf{M}_1$ *and* $\mathbf{M}_2$*;*

3: *Compute* $[\mathbf{U}, \mathbf{\Sigma}] = \mathrm{gevd}(\mathbf{T}^x(\mathbf{M}_1), \mathbf{T}^x(\mathbf{M}_2))$*, then* $\hat{\mathbf{A}} = \mathbf{U}^{-T}$*;*

4: *Compute estimated source signals by* $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{U}^T\mathbf{x}$ *.*

An insightful interpretation of the 'GEVD-method' can be given as follows. From (2.7.44) it follows directly that:

$$\mathbf{A}^{-1}\mathbf{T}^x(\mathbf{M})\mathbf{A}^{-T} = \mathbf{\Lambda}(\mathbf{M}) \qquad \forall\, \mathbf{M} \in \mathbb{R}^D_D\,. \tag{2.7.58}$$

Formulated differently, $\mathbf{A}^{-1}$ is the unique joint (generally) *non-orthogonal diagonalizer* of the set of matrices $\{\mathbf{T}^x(\mathbf{M})\}_{\mathbf{M}\in\mathbb{R}^D_D}$ because the matrices in the set $\{\mathbf{A}^{-1}\mathbf{T}^x(\mathbf{M})\mathbf{A}^{-T}\}_{\mathbf{M}\in\mathbb{R}^D_D}$ are diagonal. This is reminiscent of the diagonalizer discussed in Sections 2.7.4.3 and 2.7.4.4, where $\mathbf{Q}^T$ is the unique *orthogonal diagonalizer* of the set of matrices $\{\mathbf{T}^{y_\mathrm{w}}(\mathbf{M})\}_{\mathbf{M}\in\mathbb{R}^D_D}$ because $\mathbf{Q}^T\mathbf{T}^{y_\mathrm{w}}(\mathbf{M})\mathbf{Q} = \mathbf{\Lambda}(\mathbf{M})$ for all $\mathbf{M} \in \mathbb{R}^D_D$. Hence, $\mathbf{A}^{-1}$ plays a 'non-orthogonal role' that is similar to the 'orthogonal role' of the matrix $\mathbf{Q}^T$ in Sections 2.7.4.3 and 2.7.4.4. This becomes even more obvious by realizing that $\mathbf{Q}^T = \mathbf{Q}^{-1}$. Now, if only one matrix $\mathbf{M}$ is considered, as in Section 2.7.4.3, the diagonalizer of the corresponding cumulant matrix $\mathbf{T}^x(\mathbf{M})$ is non-unique because there are at least two solutions. One is given by $\mathbf{A}^{-1}$ and another by the orthogonal matrix containing the eigenvectors of $\mathbf{T}^x(\mathbf{M})$. The reason that using only one matrix $\mathbf{M}$ worked in Section 2.7.4.3 was the fact that the data had already been whitened, and thus only an orthogonal transform was left to be determined. In the current case, using only one matrix yields an underdetermined system of equations. Therefore, we need at least two suitable matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ in (2.7.58) to determine $\mathbf{A}^{-1}$ uniquely, as was shown above. This results in the following system that is equivalent to (2.7.56):

$$\begin{cases} \mathbf{A}^{-1}\mathbf{T}^x(\mathbf{M}_1)\mathbf{A}^{-T} = \mathbf{\Lambda}(\mathbf{M}_1)\,; \\ \mathbf{A}^{-1}\mathbf{T}^x(\mathbf{M}_2)\mathbf{A}^{-T} = \mathbf{\Lambda}(\mathbf{M}_2)\,. \end{cases} \tag{2.7.59}$$

Formulated differently, the task is to find the unique joint diagonalizer $\mathbf{W}$ of $\mathbf{T}^x(\mathbf{M}_1)$ and $\mathbf{T}^x(\mathbf{M}_2)$ that solves the following system:

$$\begin{cases} \mathbf{W}\mathbf{T}^x(\mathbf{M}_1)\mathbf{W}^T = \mathbf{\Sigma}_1; \\ \mathbf{W}\mathbf{T}^x(\mathbf{M}_2)\mathbf{W}^T = \mathbf{\Sigma}_2\,, \end{cases} \tag{2.7.60}$$

where $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are diagonal matrices. The solution $\mathbf{W}$ is related to the GEVD-solution of (2.7.57) by $\mathbf{W} = \mathbf{U}^T$. Note that a solution to (2.7.60) or a generalization of it might can be computed by any method for jointly diagonalizing two or more matrices. See Section 2.8.6 for more information on joint diagonalization algorithms.

Similarly to the JADE algorithm, one might consider the extension of these ideas to multiple matrices. Obviously, $\mathbf{A}^{-1}$ should be estimated as the joint non-orthogonal diagonalizer of a set of cumulant matrices $\mathcal{T} \triangleq \{\mathbf{T}^x(\mathbf{M}_k)\}_{1\leq k\leq K}$ for K randomly generated matrices in the set $\mathcal{M} \triangleq \{\mathbf{M}_k\}_{1\leq k\leq K}$. In other words, we want to find a matrix $\mathbf{W} = \mathbf{A}^{-1}$ such that the matrices $\{\mathbf{W}\mathbf{T}^x(\mathbf{M}_k)\mathbf{W}^T\}_{1\leq k\leq K}$ are as diagonal as possible. One of the advantages of this approach is that the problem usually becomes better conditioned and that the chance of badly conditioned or degenerate solutions due to equal eigenvalues is reduced (see previous sections). As with JADE, since the matrices in $\mathcal{T}$ are constructed from sample cumulants, they cannot be exactly jointly diagonalized, and a *Joint Approximate Non-Orthogonal Diagonalization (JAND)* technique is needed. For an overview of JAND methods, see Section 2.8.6.

2.8 Exploiting spatial uncorrelatedness and second order temporal structure

This section discusses several aspects, identification/separation principles, and algorithms from the literature for the category of blind methods that is based on exploiting the Second Order Temporal Structure (SOTS) in the data [14,42,46,85–87,106,114,157,158,185]. As we have explained in Section 2.3.2, SOTS methods are based on the simultaneous exploitation of the second order spatial independence of the source signals, i.e. spatial uncorrelatedness, and their second order temporal structure. In other words, they exploit the fact that the source signals are mutually statistically independent at order two for all possible combinations of time/lag indices. Recall from Chapter 1 and Section 2.2.2 that *by Second Order Temporal Structure we mean that the source signals possess a certain (temporal) auto-correlation structure, second order non-stationarity, or a combination of both.* We will discuss several possible approaches for exploiting SOTS. Like the ICA methods that are based on exploiting spatial independence and non-Gaussianity, some of them are based on whitened sensor signals, but most of them estimate the (de-)mixing matrix directly from the sensor signals. Since virtually all blind algorithms that are based on exploiting temporal structure employ Second Order Statistics (SOS) only, in this section we will not deal with methods based on Higher Order Temporal Structure (HOTS).

2.8.1 Second order temporal structure

In this section, we justify the exploitation of temporal structure in the data for MIBI. For simplicity, it is assumed that the system and signals are real-valued, and that $\mathbf{A}$ is square and invertible. If sensor noise is present, we assume that it is uncorrelated with the source signals. However, often we will assume that there is no sensor noise. As we have explained

in the previous sections, 'full statistical independence' means that different source signals are statistically independent for all possible orders and times; see Section 2.3.2 and in particular Eq. (2.3.5) and (2.3.6). In Section 2.6, we have seen that whitening the sensor data is not sufficient for performing blind identification or separation. This follows directly from the fact that statistical independence implies spatial and temporal uncorrelatedness but not vice versa. In principle, it means that by exploiting the 'zero-ness' of the source cross-correlation functions for lag zero only, we do not have sufficient equations to determine the mixing system $\mathbf{A}$ uniquely (up to the two indeterminacies discussed in Section 2.4). Formulated differently, *by considering the sensor correlation matrix for lag zero only, we have an underdetermined system of equations* in the (coefficients of) the (de-)mixing system. As we have explained in Section 2.6.2, after whitening only an orthogonal mixing matrix $\mathbf{Q}$ is left to be estimated. This can be accomplished in two essentially different ways (see Section 2.6.3 and Fig. 2.1): either by exploiting Higher Order Statistics, or by exploiting temporal structure. If the temporal structure of the source signals is sufficiently rich/diverse, we can gather more equations for determining $\mathbf{A}$ by exploiting the 'zero-ness' of the source cross-correlation functions for different times and/or lags. Examples of conditions for sufficient temporal diversity can be found in Section 2.2.2, e.g. the source auto-correlation functions may be assumed to be linearly independent for a certain set of lags. Note that for Gaussian data, recovering the mixing system is only possible by exploiting temporal structure. In addition, although the contrary is commonly believed, it is possible to recover the mixing matrix for more sources than sensors by using only second order statistics; we will show in later chapters that our approach can handle such scenarios.

Obviously, a useful uncorrelatedness measure is the conventional correlation function, possibly with the ensemble averaging operator $E\{\cdot\}$ replaced by a proper form of (block) time averaging operator. The source uncorrelatedness assumption is expressed mathematically in (2.3.7), which we repeat here for convenience:

$$r_{ij}^{s}[m,n] \triangleq E\left\{s_i[m]s_j[n]\right\} = 0 \qquad \forall\, 1 \leq i \neq j \leq S, \quad \forall\, m,n \in \mathbb{Z}\,. \tag{2.8.1}$$

All possible approaches for exploiting SOTS are based on this set of conditions. Most of them are formulated in terms of the sensor correlation matrix:

$$\mathbf{R}^{x}[m,n] \triangleq E\{\mathbf{x}[m]\mathbf{x}[n]^T\} \overset{(2.2.1)}{=} \mathbf{A}\mathbf{R}^{s}[m,n]\mathbf{A}^T + \mathbf{R}^{\nu}[m,n]\,, \tag{2.8.2}$$

where $\mathbf{R}^{s}[m,n] \triangleq E\{\mathbf{s}[m]\mathbf{s}[n]^T\}$ and $\mathbf{R}^{\nu}[m,n] \triangleq E\{\boldsymbol{\nu}[m]\boldsymbol{\nu}[n]^T\}$ are the source and noise correlation matrices respectively for time pair (m,n). This is the most general formulation of the sensor correlation matrix for the real-valued case. If stationary signals are assumed, with abuse of notation we will write $\mathbf{R}^{x}[k]$ instead of $\mathbf{R}^{x}[m,n]$, i.e. $\mathbf{R}^{x}[k] = \mathbf{R}^{x}[n,n-k] \triangleq E\{\mathbf{x}[n]\mathbf{x}[n-k]^T\}$, where k denotes a time lag. Similarly, in case quasi-stationary signals are assumed, with abuse of notation we will write $\mathbf{R}^{x}[n;k]$ instead of $\mathbf{R}^{x}[m,n]$, i.e. $\mathbf{R}^{x}[n\,;k] = \mathbf{R}^{x}[n,n-k] \triangleq E\{\mathbf{x}[n]\mathbf{x}[n-k]^T\}$, where n denotes a time or block index and k again denotes a lag. Note that in this case the quantity $\mathbf{R}^{x}[n,n-k]$ is time- or block-dependent, whereas in the stationary case it is time-independent.

In the coming sections we will discuss several well-known algorithms from the literature that demonstrate the application of the principles and rationale that we have just discussed to IBSS. They all exploit the specific structure of the sensor correlation matrix $\mathbf{R}^{x}[m,n]$ in (2.8.2) for different time, lag, or time-lag pairs. It should be noted that this latter point is important from the viewpoint that the form of possible solutions are restricted because only entities like (2.8.2) are considered. In fact, as we will see later on, our method is based on arranging and exploiting the same known statistics in a different way.

2.8.2 Work of Molgedey and Schuster

Molgedey and Schuster were among the first researchers that investigated the exploitation of temporal correlations (non-whiteness) for IBSS [114]. They assumed noise-free stationary signals and a square invertible mixing matrix $\mathbf{A} \in \mathbb{R}_D^D$. Hence, with abuse of notation for $r_{ij}^s[m,n]$, the stationary version of the uncorrelatedness assumption in (2.8.1) becomes:

$$r_{ij}^s[k] \triangleq E\{s_i[n]s_j[n-k]\} = 0 \qquad \forall\, 1 \leq i \neq j \leq D, \quad \forall\, n,k \in \mathbb{Z}\,. \tag{2.8.3}$$

By considering the observation correlation matrix $\mathbf{R}^x[k]$ for two different lags, viz. lag zero and some non-zero lag k, the following system of equations is obtained from (2.8.2):

$$\begin{cases} \mathbf{R}^x[0] = \mathbf{A}\mathbf{R}^s[0]\mathbf{A}^T; \\ \mathbf{R}^x[k] = \mathbf{A}\mathbf{R}^s[k]\mathbf{A}^T, \end{cases} \tag{2.8.4}$$

where the unknowns are $\mathbf{A}$, $\mathbf{R}^s[0]$ and $\mathbf{R}^s[k]$. Only the unknown $\mathbf{A}$ is interesting, and $\mathbf{R}^s[0]$ and $\mathbf{R}^s[k]$ can be considered as nuisance parameters. It is noted in [114] that this system immediately leads to a conventional eigenvalue problem. This can be shown by considering the matrix defined by $\mathbf{R}^x[0](\mathbf{R}^x[k])^{-1}$:

$$\mathbf{R}^x[0](\mathbf{R}^x[k])^{-1} = \mathbf{A}\mathbf{R}^s[0]\mathbf{A}^T\mathbf{A}^{-T}(\mathbf{R}^s[k])^{-1}\mathbf{A}^{-1} = \mathbf{A}\mathbf{R}^s[0](\mathbf{R}^s[k])^{-1}\mathbf{A}^{-1}. \tag{2.8.5}$$

Indeed, multiplying all terms in (2.8.5) by $\mathbf{A}^{-1}$ from the left and $\mathbf{A}$ from the right directly reveals the eigenstructure of the problem:

$$\mathbf{A}^{-1}\mathbf{R}^x[0](\mathbf{R}^x[k])^{-1}\mathbf{A} = \mathbf{R}^s[0](\mathbf{R}^s[k])^{-1}. \tag{2.8.6}$$

Note that the matrix $\mathbf{R}^s[0](\mathbf{R}^s[k])^{-1}$ is diagonal because of (2.8.3). The Eigenvalue Decomposition (EVD) problem in (2.8.6) can be solved by conventional EVD techniques. Another way to see that (2.8.5) defines a conventional EVD problem is to (only) multiply it by $\mathbf{A}$ from the right. Then, denoting the j-th column of $\mathbf{A}$ by $\mathbf{a}^j$ and the j-th diagonal element of $\mathbf{R}^s[0](\mathbf{R}^s[k])^{-1}$ by λ_j, this equation reads $\mathbf{R}^x[0](\mathbf{R}^x[k])^{-1}\mathbf{a}^j = \lambda_j\mathbf{a}^j \;\forall\; 1 \leq j \leq D$, thereby also clearly exhibiting the eigenproblem. Obviously, in order to identify the columns of $\mathbf{A}$ uniquely (up to the inherent indeterminacies), it is necessary that the D eigenvalues of $\mathbf{R}^x[0](\mathbf{R}^x[k])^{-1}$ are unique. If this is not the case, another lag has to be tried. Molgedey and Schuster proposed a method for actually computing the output signals that was based on a neural network similar to the one proposed by Hérault and Jutten [91].

Since the structure of system (2.8.4) is completely similar to that of (2.7.56), the problem can be written is the same way(s) as in Section 2.7.4.6. Thus, one way of viewing the problem is by considering it as the following GEVD problem that is similar to system (2.7.57) (see also Section D.3 and [72]):

$$\mathbf{R}^x[0]\mathbf{U} = \mathbf{R}^x[k]\mathbf{U}\mathbf{\Sigma}\,, \tag{2.8.7}$$

where $\mathbf{U}$ is the matrix of generalized eigenvectors of the matrix pencil $\{\mathbf{R}^x[0] - \sigma\mathbf{R}^x[k] \mid \sigma \in \mathbb{R}\}$, and $\mathbf{\Sigma}$ is the diagonal matrix that contains the corresponding eigenvalues $\sigma(\mathbf{R}^x[0], \mathbf{R}^x[k])$. As in Section 2.7.4.6, an estimate of $\mathbf{A}$ is given by $\mathbf{A} = \mathbf{U}^{-T}$. See also the other comments made in Section 2.7.4.6 w.r.t. the GEVD. The key steps of the method of Molgedey and Schuster are summarized in Alg. 2.9 on the next page. For other methods based on matrix pencils, see [41] and [46].

Algorithm 2.9 Molgedey and Schuster algorithm for blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on GEVD of two correlation matrices.

1: *Find a lag k such that the generalized eigenvalues $\sigma(\mathbf{R}^x[0], \mathbf{R}^x[k])$ are unique;*

2: *Compute $[\mathbf{U}, \mathbf{\Sigma}] = \mathrm{gevd}(\mathbf{R}^x[0], \mathbf{R}^x[k])$, then $\mathbf{A} = \mathbf{U}^{-T}$;*

3: *Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{U}^T\mathbf{x}$.*

Yet another way of viewing the problem is by considering it as the following (generally) non-orthogonal joint diagonalization problem that is similar to system (2.7.59):

$$\begin{cases} \mathbf{A}^{-1}\mathbf{R}^x[0]\mathbf{A}^{-T} = \mathbf{R}^s[0]\,; \\ \mathbf{A}^{-1}\mathbf{R}^x[k]\mathbf{A}^{-T} = \mathbf{R}^s[k]\,. \end{cases} \tag{2.8.8}$$

Hence, $\mathbf{A}^{-1}$ is the unique joint (generally) non-orthogonal diagonalizer of $\mathbf{R}^x[0]$ and $\mathbf{R}^x[k]$. For information about joint diagonalization algorithms, see Section 2.8.6.

2.8.3 AMUSE: an approach based on whitened observations and EVD

Another simple approach that exploits non-whiteness is due to Tong et al. [157, 158] and is based on the whitened observations $\mathbf{y}_{\mathrm{w}} = \mathbf{Q}\,\mathbf{s}$, where $\mathbf{Q} \in \mathbb{R}_D^D$ is a unitary matrix; see Section 2.6. The resulting algorithm is called Algorithm for Multiple Unknown Signals Extraction (AMUSE) and we will highlight its principles now. As has been explained in Section 2.6, after whitening the problem is to determine the unique orthogonal matrix $\mathbf{Q}$ that relates the sources to the whitened observations. As was noted in Section 2.6.3, instead of resorting to HOS, this can also be achieved by exploiting the temporal structure in the data; see also Fig. 2.1. More specifically, for second order statistics, it is required that the output signals are uncorrelated for some lag $k \neq 0$. For noise-free signals, this approach amounts to solving the following system of equations for $\mathbf{Q}$:

$$\begin{cases} \mathbf{R}^{y_{\mathrm{w}}}[0] = \mathbf{Q}\mathbf{R}^s[0]\mathbf{Q}^T = \mathbf{Q}\mathbf{Q}^T = \mathbf{I}\,; \\ \mathbf{R}^{y_{\mathrm{w}}}[k] = \mathbf{Q}\mathbf{R}^s[k]\mathbf{Q}^T. \end{cases}$$

Since $\mathbf{Q}$ is orthogonal, it can be computed directly by performing the EVD of $\mathbf{R}^{y_{\mathrm{w}}}[k]$. The same restriction as for other (G)EVD based approaches applies here: the lag k must be chosen such that the eigenvalues of $\mathbf{R}^{y_{\mathrm{w}}}[k]$ are unique (see Sections 2.7.4.3-2.7.4.5 and 2.8.2).

In fact, AMUSE is slightly more sophisticated than the procedure outlined here in two senses. Firstly, it can deal (to a certain extent) with sensor noise that is spatially and temporally white when the number of sensors D is larger than the number of sources S. Secondly, instead of $\mathbf{R}^{y_{\mathrm{w}}}[k]$, the matrix $\frac{1}{2}(\mathbf{R}^{y_{\mathrm{w}}}[k] + \mathbf{R}^{y_{\mathrm{w}}}[-k])$ is used. For sample correlation estimates, the latter matrix is guaranteed to be symmetric whereas the former is not. This implies that the EVD is simpler to compute. The key steps of the AMUSE method are summarized in Alg. 2.10 for noise-free signals and a square mixing matrix. For the most elaborated version that can deal with white sensor noise and the more sensors than sources, see [157] and [158].

Algorithm 2.10 AMUSE algorithm for blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\mathrm{w}} = \mathbf{Q}\,\mathbf{s}$ based on temporal correlation structure.

1-4: Whitening: see steps 1-4 of Alg. 2.5;

5: Find a lag k such that $\frac{1}{2}(\mathbf{R}^{y_{\mathrm{w}}}[k] + \mathbf{R}^{y_{\mathrm{w}}}[-k])$ has distinct eigenvalues;

6: Compute EVD of $\frac{1}{2}(\mathbf{R}^{y_{\mathrm{w}}}[k] + \mathbf{R}^{y_{\mathrm{w}}}[-k]) = \mathbf{Q}\mathbf{R}^{s}[k]\mathbf{Q}^{T}$;

7: Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^{T}\mathbf{y}_{\mathrm{w}}$.

2.8.4 Extension of AMUSE to several time lags: SOBI

The idea in the previous section can be extended to multiple matrices in the same spirit as the JADE algorithm discussed in Section 2.7.4.4 by considering a set of correlation matrices for different lags:

$$\mathbf{R}^{y_{\mathrm{w}}}[k] = \mathbf{Q}\mathbf{R}^{s}[k]\mathbf{Q}^{T} \qquad \forall\, k \in \mathcal{L}_k\,, \tag{2.8.9}$$

where $\mathcal{L}_k$ is a set of suitable time lags. In this case, $\mathbf{Q}$ should be estimated as the joint orthogonal diagonalizer of the set $\mathcal{R}^{y_{\mathrm{w}}} \triangleq \{\mathbf{R}^{y_{\mathrm{w}}}[k]\}_{k\in\mathcal{L}_k}$ of correlation matrices. In other words, we want to find $\mathbf{Q}$ such that the matrices $\{\mathbf{Q}^{T}\mathbf{R}^{y_{\mathrm{w}}}[k]\mathbf{Q}\}_{k\in\mathcal{L}_k}$ are diagonal. Since in practice the matrices in the set $\mathcal{R}^{y_{\mathrm{w}}}$ are estimated from data, i.e. they consist of sample correlations, they cannot be exactly jointly diagonalized. For this reason, an approximate joint orthogonal diagonalization algorithm is required. This approach is completely similar to the Joint Approximate Diagonalization (JAD) part of the JADE algorithm [38] and thus the same remarks apply. As with JADE, the advantages of the joint diagonalization approach are that the problem usually becomes better conditioned and that the chance of degenerate solutions due to equal eigenvalues is reduced. Also, the same Joint Approximate Orthogonal Diagonalization (JAOD) algorithm as in JADE may be used for the estimation of $\mathbf{Q}$. However, nowadays several other joint approximate diagonalization algorithms have been developed. Section 2.8.6 gives an overview of the most important ones. An algorithm that is based on the principles described above is Second Order Blind Identification (SOBI), which has been developed by Belouchrani et al. in [14]. In summary, we have the following SOBI algorithm:

Algorithm 2.11 SOBI algorithm for blind identification of $\mathbf{Q}$ and $\mathbf{s}$ in $\mathbf{y}_{\mathrm{w}} = \mathbf{Q}\,\mathbf{s}$ based on JAD of correlation matrices.

1-4: Whitening: see steps 1-4 of Alg. 2.5;

5: Determine a suitable set $\mathcal{L}_k$ of time lags;

6: Perform joint approximate diagonalization of the set $\mathcal{R}^{y_{\mathrm{w}}} \triangleq \{\mathbf{R}^{y_{\mathrm{w}}}[k]\}_{k\in\mathcal{L}_k}$ by an orthogonal matrix $\mathbf{Q}$ using one of the available numerical JAOD algorithms;

7: Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{Q}^{T}\mathbf{y}_{\mathrm{w}}$.

2.8.5 The general case: joint approximate diagonalization of several time- and lag-dependent correlation matrices

Now that we have seen some basic principles underlying SOTS based blind identification and separation using correlation matrices, a unifying framework is presented to which all approaches are related directly or indirectly. For simplicity, the noise-free case is considered with a square invertible mixing matrix.

All approaches either implicitly or explicitly exploit the special structure of the sensor (observation) correlation matrix, which is given by $\mathbf{R}^x[m,n] = \mathbf{A}\mathbf{R}^s[m,n]\mathbf{A}^T$. Obviously, in order to exploit the information in the sensor correlations, several pairs $(m,n) \in \mathbb{Z}^2$ of time indices have to be considered. Let $\mathcal{T}$ denote a suitable set of such time pairs and consider the correlation matrices in the set $\mathcal{R}^x \triangleq \left\{\mathbf{R}^x[m,n]\right\}_{(m,n)\in\mathcal{T}}$:

$$\mathbf{R}^x[m,n] = \mathbf{A}\mathbf{R}^s[m,n]\mathbf{A}^T \qquad \forall\ (m,n) \in \mathcal{T}. \tag{2.8.10}$$

Similarly to (2.7.58) and (2.8.8), under the assumption that $\mathbf{A}$ is square and invertible, this system can be written equivalently as follows:

$$\mathbf{A}^{-1}\mathbf{R}^x[m,n]\mathbf{A}^{-T} = \mathbf{R}^s[m,n] \qquad \forall\ (m,n) \in \mathcal{T}. \tag{2.8.11}$$

Hence, $\mathbf{A}^{-1}$ is the joint (generally) non-orthogonal diagonalizer of (all matrices in the set) $\mathcal{R}^x$. Note that the matrices $\left\{\mathbf{R}^s[m,n]\right\}_{(m,n)\in\mathcal{T}}$ at the right hand side of (2.8.11) are also unknown. Mathematically, the problem may be formulated as finding the de-mixing matrix $\mathbf{W}$ that is the joint diagonalizer of $\mathcal{R}^x$ (note the similarity to (2.7.60)):

$$\mathbf{W}\mathbf{R}^x[m,n]\mathbf{W}^T = \mathbf{\Sigma}[m,n] \qquad \forall\ (m,n) \in \mathcal{T}, \tag{2.8.12}$$

where $\left\{\mathbf{\Sigma}[m,n]\right\}_{(m,n)\in\mathcal{T}}$ are arbitrary diagonal matrices.

As has been explained above, since in practice the correlation matrices are only known approximately due to (finite sample) estimation errors, the diagonalization cannot be performed exactly. For this reason, a joint approximate diagonalization (JAD) technique is required. Section 2.8.6 describes several possible approaches to this problem. Summarizing, we have the following high-level algorithm:

Algorithm 2.12 Blind identification of $\mathbf{A}$ and $\mathbf{s}$ based on JAD of correlation matrices.

1: *Determine a suitable set $\mathcal{T}$ of time pairs (m,n);*
2: *Perform joint approximate diagonalization of the set $\mathcal{R}^x \triangleq \left\{\mathbf{R}^x[m,n]\right\}_{(m,n)\in\mathcal{T}}$ by a de-mixing matrix $\mathbf{W}$ using any of the available numerical JAD algorithms (see Section 2.8.6);*
3: *Compute estimated source signals by $\mathbf{y} = \hat{\mathbf{s}} = \mathbf{W}\mathbf{x}$.*

Note that for stationary and whitened signals, this algorithm reduces to Alg. 2.11. In the literature, several algorithms are described that are based on the following types of temporal information:

- Only non-whiteness is exploited; in this case the set $\mathcal{R}^x$ consists of several correlation matrices $\mathbf{R}^x[k]$, where k is a time lag;
- Only non-stationarity or quasi-stationarity is exploited; in this case the set $\mathcal{R}^x$ consists of several correlation matrices $\mathbf{R}^x[n\,;k]$, where n is a time index and k is a *fixed* time lag;
- Both non-whiteness and non-stationarity are exploited; in this case the set $\mathcal{R}^x$ consists of several correlation matrices $\mathbf{R}^x[n\,;k]$, where n is a time index and k is a time lag.

For examples, see [42], [46], [85], [86], [87], [185], and the references therein.

2.8.6 Joint approximate diagonalization methods

In this section, we consider the mathematical problem of finding a matrix that achieves the Joint Approximate Diagonalization (JAD) of a given set of matrices. As we have seen, this problem arises in various ways in blind signal processing. Firstly, consider the (exact) joint diagonalization problem. Suppose that a set $\mathcal{C} \triangleq \{\mathbf{C}_k\}_{k=1}^K$ of K possibly complex matrices of size D by D is given, then the goal of joint diagonalization is to find a transformation matrix $\mathbf{W}$ such that the matrices $\{\mathbf{F}_k \triangleq \mathbf{W}\mathbf{C}_k\mathbf{W}^H\}_{k=1}^K$ are diagonal:

$$\mathrm{offd}(\mathbf{F}_k) = \mathrm{offd}(\mathbf{W}\mathbf{C}_k\mathbf{W}^H) = \mathbf{0} \qquad \forall\, 1 \leq k \leq K\,. \tag{2.8.13}$$

Note that $\mathrm{offd}(\mathbf{F}_k)$ denotes the matrix $\mathbf{F}_k$ with zeroed diagonal elements; see Appendix A. It is well known that exact joint diagonalization is in general only possible for two matrices. Extensive literature exists on this subject, e.g. see [176], [72], [9], [25], and the references therein. For more than two matrices, exact diagonalization is only possible if the matrices possess a certain common structure. This structure is naturally revealed in the blind signal processing context, where the matrices to be diagonalized are assumed to have the following form (in the ideal case with no noise and estimation errors in the involved statistics):

$$\mathbf{C}_k \triangleq \mathbf{A}\mathbf{\Lambda}_k\mathbf{A}^H \qquad \forall\, 1 \leq k \leq K\,, \tag{2.8.14}$$

where the $\mathbf{\Lambda}_k$'s are diagonal matrices. For example, for stationary complex data the sensor correlation matrix is usually defined as $\mathbf{R}^x[k] \triangleq E\{\mathbf{x}[n]\mathbf{x}^H[n-k]\}$, which becomes $\mathbf{R}[k]^x = \mathbf{A}\mathbf{R}^s[k]\mathbf{A}^H$ in the noise-free case, and $\mathbf{R}^s[k] \triangleq E\{\mathbf{s}[n]\mathbf{s}^H[n-k]$ is diagonal due to the independence assumption. Clearly, $\mathbf{W} = \mathbf{A}^{-1}$ is a joint diagonalizer for (2.8.14) because:

$$\mathbf{W}\mathbf{A}\mathbf{\Lambda}_k\mathbf{A}^H\mathbf{W}^H = \mathbf{\Lambda}_k \qquad \forall\, 1 \leq k \leq K\,.$$

As has been explained above, the set $\mathcal{C}$ consists of matrix-valued statistics that are unknown in practice and have to be estimated from the data. Instead of a matrix $\mathbf{C}_k$, we have to consider its estimate $\widehat{\mathbf{C}}_k = \mathbf{C}_k + \Delta\mathbf{C}_k$, where $\Delta\mathbf{C}_k$ denotes the estimation error. The smaller the number of samples used for computing such a matrix, the larger the estimation error $\Delta\mathbf{C}_k$, and vice versa. Denote the estimate of the set $\mathcal{C}$ by $\hat{\mathcal{C}}$, i.e. $\hat{\mathcal{C}} \triangleq \{\widehat{\mathbf{C}}_k\}_{k=1}^K$. Since the matrices in $\hat{\mathcal{C}}$ do not exactly have the structure in (2.8.14), they cannot be exactly jointly diagonalized. However, it is still possible to define a kind of joint approximate diagonalization (JAD) problem. This can be done in several ways, the most important of which will be listed in Section 2.8.6.1. In Section 2.8.6.2, we briefly describe the main solution approaches to optimize the JAD criteria presented in Section 2.8.6.1.

2.8.6.1 Joint approximate diagonalization formulations

In this section, the most relevant definitions of joint approximate diagonalization are listed, along with some comments (also see [191]). This is done by defining appropriate criteria:

JAD1. Frobenius norm formulation:

Let $\widehat{\mathbf{F}}_k \triangleq \mathbf{W}\widehat{\mathbf{C}}_k\mathbf{W}^H$ for all $1 \leq k \leq K$. Then, the JAD of the set $\hat{\mathcal{C}}$ is defined as the minimization of the following cost function (see [86], [87] and [185], for example):

$$J_{\text{F}}(\mathbf{W};\hat{\mathcal{C}}) \triangleq \sum_{k=1}^{K} \beta^k \operatorname{off}(\widehat{\mathbf{F}}_k) = \sum_{k=1}^{K} \beta^k \left\|\operatorname{offd}(\widehat{\mathbf{F}}_k)\right\|_{\text{F}}, \tag{2.8.15}$$

where the coefficients $\{\beta^k\}_{k=1}^K$ are pre-specified positive weight factors, and $\operatorname{off}(\widehat{\mathbf{F}}_k)$ is the Frobenius norm of the off-diagonal elements of $\widehat{\mathbf{F}}_k$. According to this formulation, the goal of JAD is to find a transformation matrix $\mathbf{W}$ such that the matrices $\{\widehat{\mathbf{F}}_k\}_{k=1}^K$ are as diagonal as possible as measured by (2.8.15). Hence, the optimal diagonalization matrix $\mathbf{W}_{\text{opt}}$ is given by:

$$\mathbf{W}_{\text{opt}} = \underset{\mathbf{W}}{\operatorname{argmin}}\, J_{\text{F}}(\mathbf{W};\hat{\mathcal{C}})\,. \tag{2.8.16}$$

Since the trivial solution $\mathbf{W} = \mathbf{0}$ minimizes the cost function, an additional assumption has to be made on $\mathbf{W}$. For example, it may be assumed to be full rank, to be orthonormal, or have unit-norm columns. Different methods for minimizing (2.8.15) vary also in these assumptions.

JAD2. Positive definite formulation:

The criterion used in this formulation is based on the assumption that the matrices in the set $\hat{\mathcal{C}}$ (or $\mathcal{C}$) are positive-definite. For example, in the IBSS context this assumption is justified when only the non-stationarity structure of the data for lag zero is used while neglecting the temporal correlation structure. Hence, sensor correlation matrices of the form $\mathbf{R}^x[n\,;k]$, where n is a time index and k a time lag, are considered, where $k = 0$ (see previous section). For this case, Matsuaoka et at. [110] and Pham [124] proposed the following cost function:

$$J_{\text{PD}}(\mathbf{W};\hat{\mathcal{C}}) \triangleq \sum_{k=1}^{K} \Big(\log\det\big(\operatorname{ddiag}(\widehat{\mathbf{F}}_k)\big) - \log\det\big(\widehat{\mathbf{F}}_k\big)\Big)\,. \tag{2.8.17}$$

Obviously, this criterion yields zero when all $\widehat{\mathbf{F}}_k$'s are diagonal. For time-delayed correlation matrices, this criterion is not suitable since these matrices are not guaranteed to be positive-definite. The optimal diagonalization matrix $\mathbf{W}_{\text{opt}}$ is given by an expression similar to (2.8.16), where $J_{\text{F}}(\mathbf{W};\hat{\mathcal{C}})$ is replaced by $J_{\text{PD}}(\mathbf{W};\hat{\mathcal{C}})$.

JAD3. Model fitting formulation:

This approach is based on fitting the ideal model of the matrices in $\mathcal{C}$, as specified by (2.8.14), to the estimated matrices in $\hat{\mathcal{C}}$. Hence, a matrix $\hat{\mathbf{A}}$ and a set of diagonal matrices $\{\hat{\mathbf{\Lambda}}_k\}_{k=1}^K$ are sought such that the matrices $\{\hat{\mathbf{A}}\hat{\mathbf{\Lambda}}_k\hat{\mathbf{A}}^H\}_{k=1}^K$ are approximately equal to the matrices $\{\widehat{\mathbf{C}}_k\}_{k=1}^K$ in some sense (note that $\widehat{\mathbf{C}}_k = \mathbf{C}_k + \Delta\mathbf{C}_k = \mathbf{A}\mathbf{\Lambda}_k\mathbf{A}^H + \Delta\mathbf{C}_k$ for the 'ideal' values of $\hat{\mathbf{A}}$ and $\{\hat{\mathbf{\Lambda}}_k\}_{k=1}^K$):

$$\widehat{\mathbf{C}}_k \approx \hat{\mathbf{A}}\hat{\mathbf{\Lambda}}_k\hat{\mathbf{A}}^H \qquad \forall\, 1 \leq k \leq K\,.$$

A commonly used measure of the fit between both sides of this equation is the Frobenius norm of their difference, as specified by the following cost function:

$$J_{\text{fit}}\big(\mathbf{A}, \{\mathbf{\Lambda}_k\}_{k=1}^K; \hat{\mathcal{C}}\,\big) \triangleq \sum_{k=1}^{K} \beta^k \big\| \widehat{\mathbf{C}}_k - \mathbf{A}\mathbf{\Lambda}_k\mathbf{A}^H \big\|_{\text{F}}^2 \,, \tag{2.8.18}$$

where the coefficients $\{\beta^k\}_{k=1}^K$ are pre-specified positive weight factors. See [188] and [172]. The optimal solution, i.e. the optimal values of the model unknowns, are given by:

$$\big\{\hat{\mathbf{A}}, \{\hat{\mathbf{\Lambda}}^k\}_{k=1}^K\big\} = \underset{\mathbf{A}, \{\mathbf{\Lambda}_k\}_{k=1}^K}{\operatorname{argmin}} \; J_{\text{fit}}\big(\mathbf{A}, \{\mathbf{\Lambda}_k\}_{k=1}^K; \hat{\mathcal{C}}\,\big) \,. \tag{2.8.19}$$

This formulation is reminiscent of the so-called *(weighted) subspace fitting* problem. Compared to the other two approaches, model fitting algorithms have some very important advantages. Firstly, it is not needed to make assumptions like orthonormality, positive-definiteness, (Hermitian) symmetry, etc. on $\mathbf{A}$ and $\{\mathbf{C}_k\}_{k=1}^K$. Secondly, it is possible to handle non-square matrices. In the MIBI context, this allows one to determine more columns than sensors, which is an important advantage for the work in this thesis. Note that the comparison between model fitting on the one hand, and finding a diagonalizing transformation matrix on the other, is similar in a certain sense to that between MIBI and IBSS. Also note that we can see the model fitting approach as a kind of 'approximate average matrix decomposition'.

2.8.6.2 Solution approaches for optimization of JAD criteria

Now that we have seen the major formulations of the joint approximate diagonalization in terms of optimization problems, we will give a brief overview of the solution approaches that are commonly used in this context. There are several ways of categorizing solution methods. In the current context, it makes sense to distinguish the following main categories: Jacobi-like methods and iterative methods for cost function minimization such as gradient descent, conjugate gradient, Newton's method, etc. The methods in the latter category are often based on some approximation of the cost function.

The first class of methods for achieving JAD is based on the so-called *Jacobi method*. The classic Jacobi method for diagonalizing a single Hermitian (or symmetric) matrix $\mathbf{A}$ is based on iteratively updating $\mathbf{A}$ by orthogonal plane similarity transformations in such a way that each new update is more diagonal than the previous one, as measured by $\operatorname{off}(\mathbf{A}) = \sum_i \sum_{j\neq i} |a_i^j|^2$ [72]. That is, $\mathbf{A}$ is updated according to $\mathbf{A} \leftarrow \mathbf{Q}^H\mathbf{A}\mathbf{Q}$, where $\mathbf{Q}$ is a *Jacobi rotation* (also called Givens rotation). Such a Jacobi rotation is a plane rotation in the sense that it alters only two rows and two columns. This technique has been extended by several authors to the joint diagonalization of multiple matrices, where the criterion in (2.8.15) was used as a measure of diagonality. We will not discuss the specifics of the different methods, but refer the reader to the literature: see [25], [38] and [39] for examples. Most Jacobi-like algorithms for JAD are restricted to finding an orthonormal diagonalizing matrix (see JADE in Section 2.7.4.4, for example). Hence, such methods are only suitable for whitened data. However, as was noted in Section 2.6.3, whitening can severely limit the performance of the separation in the sense that bias or errors in the whitening stage cannot be corrected in the following orthogonal separation stage [31]. The whitening approach favours the exact diagonalization of one specific matrix, namely the lag zero correlation matrix, possibly and

probably at the expense of a poor diagonalization of the other matrices. Therefore, for MIBI or IBSS based on JAD, it is in general preferable to work with non-whitened data. Consequently, methods that are able to find (generally) non-orthogonal diagonalizers are preferred. Pham [124] has proposed an algorithm that follows the classic Jacobi approach of making successive transformations on pairs of rows and columns of the matrices to be diagonalized, but with elementary transformations that are different from the traditional Jacobi rotations. However, this approach only works for positive definite matrices (see the second point *JAD2* in the list above).

The second class of methods for achieving JAD consists of more conventional iterative methods for cost function minimization such as gradient descent, conjugate gradient, and Newton's method. This class has recently become more popular; one of the reasons is that it can more easily handle non-orthogonal diagonalization matrices. The most commonly used iterative methods are gradient descent and Newton's method. As an example, we will briefly discuss the gradient based approach (see Section 2.5). According to the standard gradient descent method, the cost function $J_{\mathrm{F}}(\mathbf{W};\hat{\mathcal{C}})$ in (2.8.15) is minimized by iteratively updating some initial value $\mathbf{W}[0]$ of $\mathbf{W}$ as follows:

$$\mathbf{W}[k+1] = \mathbf{W}[k] - \mu[k]\nabla_{\mathbf{W}} J_{\mathrm{F}}(\mathbf{W}[k];\hat{\mathcal{C}})\,. \tag{2.8.20}$$

Note that $J_{\mathrm{F}}(\mathbf{W};\hat{\mathcal{C}})$ is a real-valued scalar function and its argument $\mathbf{W}$ can be real- or complex-valued. In case $\mathbf{W}$ is complex, the gradient $\nabla_{\mathbf{W}}$ is a so-called complex gradient. See [20] for a rigorous treatment of complex gradients.

As has been noted in the first point *JAD1* of the list above, $J_{\mathrm{F}}(\mathbf{W};\hat{\mathcal{C}})$ is minimized by the trivial solution $\mathbf{W} = \mathbf{0}$. For this reason, an additional constraint needs to be imposed on $\mathbf{W}$. For example, possible useful constraints on $\mathbf{W}$ are that it has some fixed Frobenius norm, unit-norm rows or columns, unit diagonal elements, or a determinant with absolute value one. These constraints are all related to, and (intuitively) justified by the scaling indeterminacy discussed in Section 2.4 (also see Section 2.2.2). In the blind signal processing context, this amounts to fixing the scaling ambiguity by imposing constraints on the de-mixing matrix $\mathbf{W}$ instead of the mixing matrix $\mathbf{A}$. If whitened data are used, $\mathbf{W}$ should be constrained to be unitary ($\mathbf{W}\mathbf{W}^H = \mathbf{W}^H\mathbf{W} = \mathbf{I}$). In other words, $\mathbf{W}$ must belong to the space of unitary matrices, the so-called (complex) Stiefel manifold. Mathematically, the constraints mentioned above take the following form:

$$\|\mathbf{W}\|_{\mathrm{F}} = w_{\mathrm{F}}, \quad \mathrm{ddiag}(\mathbf{W}\mathbf{W}^H) = \mathbf{I}, \quad \mathrm{diag}(\mathbf{W}) = \mathbf{I}, \quad |\det(\mathbf{W})| = 1\,, \tag{2.8.21}$$

where w_{F} is some positive scalar.

Essentially, there are only a few practical ways to impose the additional constraints. Firstly, *projection onto the constraint set* can be used [109], [115], [80]. In this case, after each iteration step of the unconstrained iterative method that is used for minimization of the cost function, for example the gradient descent rule in (2.8.20), Newton's method, etc., the updated solution is projected onto the constraint set such that is satisfies the constraints. E.g., after using update rule (2.8.20) and adopting the first assumption in (2.8.21), $\mathbf{W}$ should be projected as $\mathbf{W} \leftarrow w_{\mathrm{F}} \cdot \mathbf{W} / \|\mathbf{W}\|_{\mathrm{F}}$. Obviously, the initial value $\mathbf{W}[0]$ should also be normalized like this. Instead of projecting the updated value of $\mathbf{W}$, it is also possible to project $\Delta\mathbf{W}$, i.e. the change in $\mathbf{W}$ made by the iterative method (for example, $\Delta\mathbf{W} = -\mu[k]\nabla_{\mathbf{W}} J_{\mathrm{F}}(\mathbf{W}[k];\hat{\mathcal{C}})$ in (2.8.20)) onto the constraint set. For example, the third assumption in (2.8.21) can easily be applied by starting with an initial value $\mathbf{W}[0]$ that satisfies the constraint, i.e. $\mathrm{diag}(\mathbf{W}[0]) = \mathbf{I}$, and zeroing the diagonal elements of $\Delta\mathbf{W}$ before

applying each update. An alternative method for incorporating a constraint is by adding a *penalty term* $J_{\text{pen}}(\mathbf{W})$ to the main cost function that has to be minimized [85], i.e.:

$$J_{\text{ova}}(\mathbf{W}) = J_{\text{main}}(\mathbf{W}) + w_{\text{pen}} J_{\text{pen}}(\mathbf{W}) \,, \tag{2.8.22}$$

where $J_{\text{ova}}(\mathbf{W})$ denotes the resulting 'overall' cost function, $J_{\text{main}}(\mathbf{W})$ the cost term that represents the main diagonalization criterion such as in (2.8.15), and $J_{\text{pen}}(\mathbf{W})$ the penalty term that penalizes the deviation of $\mathbf{W}$ from the constraint. The penalty term can be weighted by the positive scalar w_{pen}. When using the gradient descent rule (2.8.20), the gradient $\nabla_{\mathbf{W}} J_{\text{F}}(\mathbf{W}[k]; \hat{\mathcal{C}})$ should be replaced by $\nabla_{\mathbf{W}} J_{\text{ova}}(\mathbf{W}) = \nabla_{\mathbf{W}} J_{\text{F}}(\mathbf{W}[k]; \hat{\mathcal{C}}) + w_{\text{pen}} \nabla_{\mathbf{W}} J_{\text{pen}}(\mathbf{W})$.

Convenient formulations of constraints are usually defined in terms of squared norms. For example, the penalty terms corresponding to the constraints in (2.8.21) could be defined in a rather natural way by respectively:

$$\begin{aligned} J_{\text{pen}}(\mathbf{W}) &\triangleq (\|\mathbf{W}\|_{\text{F}} - w_{\text{F}})^2, \qquad J_{\text{pen}}(\mathbf{W}) \triangleq \left\|\text{ddiag}(\mathbf{W}\mathbf{W}^H) - \mathbf{I}\right\|_{\text{F}}^2, \\ J_{\text{pen}}(\mathbf{W}) &\triangleq \|\text{diag}(\mathbf{W}) - \mathbf{I}\|_{\text{F}}^2, \quad J_{\text{pen}}(\mathbf{W}) \triangleq |\log(|\det(\mathbf{W})|)|^2 \,. \end{aligned} \tag{2.8.23}$$

See [85] for several examples of penalty terms and their gradients. The iterative methods described above often are based on some approximation of the (main) cost function, usually in terms of a Taylor expansion. A nice and efficient method is presented in [191]. See [87] for a Newton method based on a second order cost function approximation.

The cost function involved in the model fitting formulation *JAD3* in the list above can be minimized using conventional techniques, for example, by a gradient descent method. Note however, that in this case more parameters are involved: not only the matrix $\mathbf{A}$ has to be estimated, but also the diagonal matrices $\{\mathbf{\Lambda}_k\}_{k=1}^K$. Usually, the gradient descent method is not applied directly, but special techniques have been developed for the model fitting problem. One approach follows the conventional approach of (weighted) subspace fitting, see [172] and [184]. Problem (2.8.19) is then reformulated in such a way that the problem becomes separable in the sense that the parameters of the diagonal matrices $\{\mathbf{\Lambda}_k\}_{k=1}^K$ are eliminated. The remaining problem then 'only' has to be solved for $\mathbf{A}$. This can be done by any of the iterative methods mentioned above: gradient descent, Newton's method, etc. See [172] for an example of applying subspace fitting techniques to the JAD problem. Another approach to estimate $\mathbf{A}$ and $\{\mathbf{\Lambda}_k\}_{k=1}^K$ has been proposed by Yeredor in [187] and [188]. His algorithm bears the acronym AC-DC because it alternates between two minimization schemes:

- The AC ('Alternating Columns') phase minimizes $J_{\text{fit}}\big(\mathbf{A}, \{\mathbf{\Lambda}_k\}_{k=1}^K; \hat{\mathcal{C}}\,\big)$ in (2.8.18) w.r.t. a selected column of $\mathbf{A}$ while keeping the other columns, as well as the diagonal matrices $\{\mathbf{\Lambda}_k\}_{k=1}^K$, fixed;

- The DC ('Diagonal Centers') phase minimizes $J_{\text{fit}}\big(\mathbf{A}, \{\mathbf{\Lambda}_k\}_{k=1}^K; \hat{\mathcal{C}}\,\big)$ w.r.t. the diagonal matrices $\{\mathbf{\Lambda}_k\}_{k=1}^K$ while keeping $\mathbf{A}$ fixed.

The ideas of this algorithm are important because they can be generalized to more complicated and general problems.

2.9 Conclusions and discussion

In this chapter, we have presented the theoretical background and a literature review of instantaneous Blind Signal Processing (BSP). Among other things, this allows us to contextualize our work w.r.t. the existing literature. We have attempted to provide a coherent overview of the field and to add value to the existing overviews and literature by stressing insight and relationships. One of our contributions is that the ideas underlying BSP have been formulated in such a way that in theory they can be generalized quite naturally to more complicated scenarios. We have seen that the development of a blind identification method involves several important steps. First of all, several fundamental assumptions are made on the mixing system, source signals, and noise signals. All blind methods have in common that they are based on the same guiding principle of statistical independence of the source signals. Secondly, based on the assumptions an identification or separation criterion is chosen. Finally, the mixing system, de-mixing system, or source signals are estimated by optimizing this criterion. We have distinguished two main approaches to BSP problems that reflect our point of view. The first category consists of methods that do not exploit any temporal structure in the data and are based on two main assumptions, viz. mutual spatial statistical independence and non-Gaussianity of the source signals. This category of methods is typically referred to as Independent Component Analysis (ICA) and it requires the use of HOS. The second category consists of methods that do exploit temporal structure; the methods belonging to it are also based on two main assumptions, viz. mutual spatial statistical independence and the presence of a certain temporal structure in the data. Although in principle algorithms from this category can use both second and higher order statistics, in practice virtually only second order statistics are exploited. Although many blind algorithms and techniques that are presented in the literature for the two different categories of blind methods seem different at first sight, we have shown that often they can be formulated in very similar ways. For instance, we have highlighted the resemblance between eigenstructure based algorithms for lag-zero Higher Order Statistics (HOS) based methods from the ICA category and Second Order Temporal Structure (SOTS) based methods from the temporal category. Furthermore, in this chapter we have considered several measures of statistical independence, which we also classified into two classes, viz. information-theoretic and signal processing oriented measures. The first class consists of measures like differential entropy, negentropy, mutual information, and Kullback-Leibner divergence. Such measures are typically used by blind methods from the ICA category. The second class consists of measures like cross-correlations, cross-cumulants, nonlinear cross-correlations and mathematical expectations, and kurtosis. These measures are used by both the ICA and the temporal categories. The MIBI method that we will present in later chapters belongs to the temporal category and will be developed in such a way that it is general and unifying w.r.t. the order and type of the exploited statistics. It heavily relies on the so-called subspace techniques that will be explained in the next chapter.

CHAPTER 3

Subspace methods for DOA identification

An approach that is commonly followed for solving instantaneous semi-blind array signal processing problems, such as Direction Of Arrival (DOA) estimation, is the so-called *subspace approach* [16, 137, 147, 148, 154, 171, 175, 179, 183]. Together with the rationale behind exploiting temporal structure, the rationale behind the subspace approach is of paramount importance for the blind identification method that we will develop in the chapters ahead. For this reason, it is discussed in depth in this chapter. *Subspace methods* are based on the decomposition of the observation/sensor vector space into two subspaces, viz. one that is associated with the source signals, the so-called *signal subspace*, and one that is associated with the noise signals, the so-called *noise subspace* [147, 154]. Although the concepts involved in subspace methods are more general, most approaches rely on the eigen-decomposition of the lag-zero covariance matrix of the sensor snapshots, i.e. they only employ lag-zero Second Order Statistics (SOS); see also Section 1.2.6. Most of the SOS based subspace theory described in the first part of this chapter is well-known. Its development started with the works of Schmidt [137], and Bienvenu and Kopp [16], and has been steadily proceeding until now. The literature on Higher Order Statistics (HOS) based subspace methods [28, 40, 129, 140, 190] is quite modest and less known compared to SOS (see Chapter 1 for possible reasons). One of our goals in this chapter is to contribute to the 'subspace research area' by providing a unifying overview that allows natural generalizations and gives enhanced insight. Among other things, we show that the principles underlying SOS based subspace methods can easily be generalized to higher order and temporal statistics. We abstract and emphasize key ideas, such as the way in which the employed sensor statistics are arranged in a matrix, the rank of that matrix, etcetera. In fact, we deduce a kind of 'algorithm for producing subspace based algorithms' employing any kind of statistics. The techniques and generalizations that are presented in this chapter, as well as the notation used to describe them, will be employed intensively in the rest of the thesis. The notation has been set up in such a way it allows us to denote all kinds of quantities in a unifying manner. Many parameter estimation applications in Array Signal Processing (ASP), such as the various Source Localization applications considered in Section 1.1.3, typically employ subspace techniques. Because most of the literature on the topic deals with the conventional DOA estimation problem, and this subject lends itself very well for explaining subspace methods, here we will also restrict our attention to this research area. In later chapters, we will present a way to employ and generalize subspace techniques for solving the more difficult and general MIBI/IBSS problems.

In Section 3.1, we explain the considered DOA model(s) in detail. After that, in Section 3.2 we highlight the principles behind conventional lag-zero Second Order Statistics (SOS) based subspace methods. Next, a possible generalization of subspace techniques to Higher Order Statistics (HOS) is discussed in Section 3.3. Then, we have obtained sufficient insight for abstracting the key ideas of subspace methods in a retrospective analysis, which is presented in Section 3.4. Finally, conclusions and discussion are presented in Section 3.5.

3.1 Direction Of Arrival observation model

As in Section 1.1.3, it is assumed that S statistically independent complex narrowband signals [94,147,179,183] emitted by *far field* sources are incident on a planar array of D sensors. In addition, it is assumed that the propagation medium is linear, homogeneous and isotropic, and that all sensors are identical and have omnidirectional unit transfer. For simplicity, we will mainly consider the (co)planar case depicted in Fig. 3.1, where the sources and sensors are located in the same two-dimensional plane. We will refer to this scenario as 2*D-scenario* because it involves two spatial dimensions. Most concepts discussed in this section can be generalized in a rather straightforward manner to more general scenarios such as the 3D-scenario. In Section 3.1.1, first the observation model for an arbitrary planar array, i.e. an array with arbitrary 2D sensor positions, is discussed. Then, in Section 3.1.2 this model is specialized to a so-called Uniform Linear Array (ULA). Finally, in Section 3.1.3 the geometrical properties of the general model are highlighted.

3.1.1 Arbitrary sensor positions for 2D-scenario

In the plane containing the sources and sensors an orthogonal coordinate system is defined with coordinate axes denoted by p^1 and p^2. The problem setup is depicted in Fig. 3.1. As in Fig. 1.7, the sensor locations are indicated by black dots and the source locations by white dots. The position of the i-th sensor is denoted by the row vector $\tilde{\mathbf{p}}_i = \begin{bmatrix} p_i^1 & p_i^2 \end{bmatrix}$, i.e. the first coordinate of the i-th sensor position is p_i^1 and that of the second is p_i^2. Let $\mathbf{z}(\theta)$ be

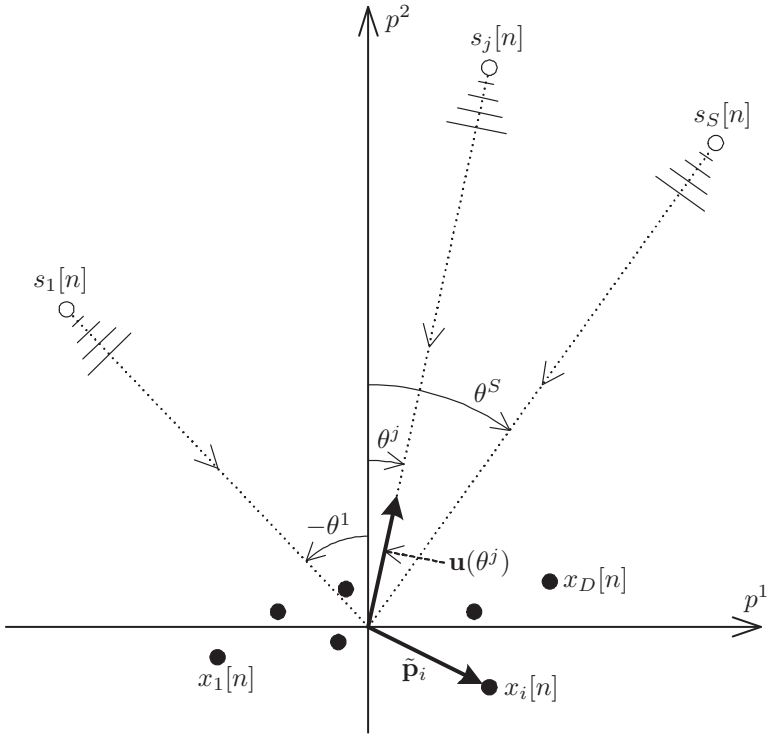

Figure 3.1: *Coplanar DOA estimation problem setup.*

the noise-free complex *array response (vector)* (see also Section 1.1.3 and [97, 147, 183]) to a unit-amplitude complex plane wave with Direction Of Arrival (DOA) θ that is measured clockwise w.r.t. the positive p^2-axis (the functional dependence of $\mathbf{z}(\theta)$ on θ will be derived shortly). Then, the complex sensor array observation vector $\mathbf{x}[n] \in \mathbb{C}_D$ at discrete time n, called the n-th *snapshot*, can be modeled as in (1.1.1) with parameterized mixing matrix columns:

$$\mathbf{x}[n] = \sum_{j=1}^{S} \mathbf{z}(\theta^j)\, s_j[n] + \boldsymbol{\nu}[n] = \mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\, \mathbf{s}[n] + \boldsymbol{\nu}[n] \qquad \forall\, n \in \mathbb{Z}\,. \tag{3.1.1}$$

Recall that the array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \in \mathbb{C}_D^S$ is defined as $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \triangleq \big[\mathbf{z}(\theta^1) \;\cdots\; \mathbf{z}(\theta^S)\big]$, where $\tilde{\boldsymbol{\theta}} \triangleq \big[\theta^1 \;\cdots\; \theta^S\big] \in \mathbb{R}^S$. The i-th element $a_i^j \triangleq z_i(\theta^j)$ of the j-th array response vector $\mathbf{a}^j \triangleq \mathbf{z}(\theta^j)$ denotes the complex instantaneous transfer from the j-th source to the i-th sensor. Let $\mathbf{u}(\theta)$ be the unit column vector pointing in DOA θ, i.e. $\mathbf{u}(\theta) = \big[\sin(\theta) \;\; \cos(\theta)\big]^T$. Furthermore, let the frequency, angular frequency, and wavelength of the carrier be denoted by f_c, ω_c, and λ_c respectively. Finally, let the signal propagation velocity be denoted by c. Then, the propagation time difference $\tau_i(\theta)$ between the arrival at the origin of the p^1-p^2 coordinate system and the arrival at the i-th sensor for a signal impinging from direction θ is given by:

$$\tau_i(\theta) = \frac{\tilde{\mathbf{p}}_i \mathbf{u}(\theta)}{c} \qquad \forall\, 1 \le i \le D\,. \tag{3.1.2}$$

Due to the narrowband character of the source signals, the corresponding complex transfer coefficient $z_i(\theta)$ is given by the phase factor $\exp\big(\jmath\omega_\mathrm{c}\tau_i(\theta)\big)$ [97, 147, 179]. The product $\phi_i(\theta) \triangleq \omega_\mathrm{c}\tau_i(\theta)$ in the exponent, i.e. the phase of $z_i(\theta)$, can be written as follows:

$$\begin{aligned}\phi_i(\theta) = \omega_\mathrm{c}\tau_i(\theta) &= \frac{\omega_\mathrm{c}}{c}\tilde{\mathbf{p}}_i\mathbf{u}(\theta) = \frac{2\pi f_\mathrm{c}}{c}\tilde{\mathbf{p}}_i\mathbf{u}(\theta) = \frac{2\pi}{\lambda_\mathrm{c}}\tilde{\mathbf{p}}_i\mathbf{u}(\theta)\\ &= 2\pi\tilde{\mathbf{p}}_{i,\mathrm{n}}\mathbf{u}(\theta) \qquad \forall\,\theta \in \Theta, \quad \forall\, 1 \le i \le D\,,\end{aligned} \tag{3.1.3}$$

where Θ is the *parameter space* of interest, i.e. the set in which θ is known, allowed, or chosen to vary, and $\tilde{\mathbf{p}}_{i,\mathrm{n}}$ is the position vector of the i-th sensor that is normalized w.r.t. the carrier wavelength λ_c, i.e.:

$$\tilde{\mathbf{p}}_{i,\mathrm{n}} \triangleq \frac{\tilde{\mathbf{p}}_i}{\lambda_\mathrm{c}} \qquad \forall\, 1 \le i \le D\,. \tag{3.1.4}$$

By using normalized sensor positions, i.e. by measuring all sensor positions in units of carrier wavelength λ_c, the model and the results that follow from it do not depend on absolute array apertures and sensor positions. Now, the complex instantaneous transfer $z_i(\theta)$ from a source at DOA θ to the i-th sensor can be written as:

$$z_i(\theta) = \exp\big(\jmath\,\omega_\mathrm{c}\tau_i(\theta)\big) = \exp\big(\jmath\, 2\pi\tilde{\mathbf{p}}_{i,\mathrm{n}}\mathbf{u}(\theta)\big) = \exp\big(\jmath\,\phi_i(\theta)\big) \quad \forall\,\theta \in \Theta, \quad \forall\, 1 \le i \le D\,, \tag{3.1.5}$$

where the phase $\phi_i(\theta)$ is given by (3.1.3). In particular, the instantaneous transfer a_i^j from the j-th source to the i-th sensor can be written as follows:

$$a_i^j = z_i(\theta^j) = \exp\big(\jmath\, 2\pi\tilde{\mathbf{p}}_{i,\mathrm{n}}\mathbf{u}(\theta^j)\big) \qquad \forall\, 1 \le i \le D, \quad \forall\, 1 \le j \le S\,. \tag{3.1.6}$$

From (3.1.5), it is clear that the array response vector $\mathbf{z}(\theta)$ for a source at DOA θ is given by:

$$\mathbf{z}(\theta) = \begin{bmatrix} z_1(\theta) \\ \vdots \\ z_D(\theta) \end{bmatrix} = \begin{bmatrix} \exp\big(\jmath\,\phi_1(\theta)\big) \\ \vdots \\ \exp\big(\jmath\,\phi_D(\theta)\big) \end{bmatrix} = \begin{bmatrix} \exp\big(\jmath\, 2\pi\tilde{\mathbf{p}}_{1,\mathrm{n}}\mathbf{u}(\theta)\big) \\ \vdots \\ \exp\big(\jmath\, 2\pi\tilde{\mathbf{p}}_{D,\mathrm{n}}\mathbf{u}(\theta)\big) \end{bmatrix} \quad \forall\,\theta \in \Theta\,. \tag{3.1.7}$$

The collection of array response vectors $\mathbf{z}(\theta)$ formed by varying the DOA θ over the parameter space of interest Θ is called the *array manifold* $\boldsymbol{\mathcal{A}}(\Theta)$ [97]:

$$\boldsymbol{\mathcal{A}}(\Theta) \triangleq \left\{ \mathbf{z}(\theta) \mid \theta \in \Theta \right\}. \tag{3.1.8}$$

Since there is only one parameter θ in the currently considered model, the array manifold is a one-dimensional curve in the complex D-dimensional Euclidian space. In the sequel *it is assumed that the parameterization of the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$, i.e. the functional dependence of $\mathbf{z}(\theta)$ on θ for all $\theta \in \Theta$, is known*. Since the purpose of DOA estimation is the extraction of the source DOA's from the sensor data, and since the sensor data is observed via the array response vectors (see (3.1.1)), a DOA θ and its associated array response vector $\mathbf{z}(\theta)$ must be related one-to-one for all $\theta \in \Theta$ in order to allow the unique determination of the source DOA's from the sensor data. This condition is necessary but not sufficient because the DOA determination also depends on other quantities and/or parameters such as the number of sensors in relation to the number of sources, the statistical properties of the signals, the order of the exploited statistics, etc.

Definition 3.1.1. Unambiguous array response vector or unambiguous array. An array response vector $\mathbf{z}(\theta) \in \boldsymbol{\mathcal{A}}(\Theta)$ is said to be unambiguous over Θ if the DOA θ can be determined uniquely from $\mathbf{z}(\theta)$, i.e. if θ and $\mathbf{z}(\theta)$ are related one-to-one for all $\theta \in \Theta$. In other words, $\mathbf{z}(\theta) \in \boldsymbol{\mathcal{A}}(\Theta)$ is unambiguous if the mapping $\mathbf{z}(\theta)$ from Θ to $\boldsymbol{\mathcal{A}}(\Theta)$ is injective. In addition, if the array response vector $\mathbf{z}(\theta) \in \boldsymbol{\mathcal{A}}(\Theta)$ is unambiguous the array itself is also said to be unambiguous.

As is clear intuitively from (3.1.7), a one-to-one relation between θ and $\mathbf{z}(\theta)$ for all $\theta \in \Theta$ can be established by choosing the normalized sensor positions $\{\tilde{\mathbf{p}}_{i,\mathrm{n}}\}_{1 \leq i \leq D}$ properly. An example of such a choice will be given in the next section for the simple case of a so-called *Uniform Linear Array (ULA)*. In general, the purpose of Source Localization is to extract (a subset of) the parameters characterizing each source position, e.g. Cartesian, polar or spherical coordinates, from the sensor data [179, 183]. Similarly to the DOA estimation scenario considered above, *in the general ISBSL scenario the unique determination of the source positions from the sensor data requires that the set of parameters characterizing a source position and its associated array response vector are related one-to-one for all possible values of the parameters* (see also Section 1.1.3). Similarly to the ULA case, this can be achieved by placing the sensors properly.

Intermezzo I: Usefulness of using both sub- and superscripts

Note from (3.1.6) that the subscript position of the index i in the component $\tilde{\mathbf{p}}_{i,\mathrm{n}}$ of $a_i^j = z_i(\theta^j)$, and in a_i^j itself, immediately indicates that the row vector $\tilde{\mathbf{p}}_{i,\mathrm{n}}$ is related to the 'sensor part' of the mixing matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ because a specific, viz. the i-th, row $\tilde{\mathbf{a}}_i\big(\tilde{\boldsymbol{\theta}}\big)$ of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ corresponds to a specific, viz. the i-th, sensor and is indexed by a subscript index; see (3.1.1) and (3.1.7). Likewise, the superscript position of the index j in the component $\mathbf{u}(\theta^j)$ of a_i^j, and in a_i^j itself, indicates that the column vector $\mathbf{u}(\theta^j)$ is related to the 'source part' of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ because a specific, viz. the j-th, column $\mathbf{a}^j = \mathbf{z}(\theta^j)$ of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ is associated with a specific, viz. the j-th, source and is indexed by a superscript index. In addition, note that the summation in (3.1.1) is reminiscent of the notational convention adopted in our work which states that summations are typically performed over a pair of quantities, one of which has a subscript index and the other a superscript index, or can be interpreted that way. The summation in (3.1.1) clearly

shows that the j-th array response vector $\mathbf{z}(\theta^j)$ is associated with the j-th source signal $s_j[n]$, which is also evident physically. We summarize these notational conventions as follows (see also Appendix A):

> A subscript index of a transfer/propagation factor indicates a direct relation to a specific sensor, whereas a superscript index indicates a direct relation to a specific source.

As we will see in the sequel, this also holds for most statistical quantities. The distinctions made above would not have been clear if all vectors were denoted the same way, e.g. by choosing all vectors to be column vectors and obtaining row vectors by (conjugate) transposing column vectors, and vice versa. For example, if the DOA of the j-th source were denoted by θ_j instead of θ^j, and thus the associated unit vector by $\mathbf{u}(\theta_j)$ instead of $\mathbf{u}(\theta^j)$, and if there were no distinction between row and column vectors, it would not immediately be clear that the vector $\mathbf{p}_{i,\mathrm{n}}$ in the inner product $\mathbf{p}_{i,\mathrm{n}}^T\mathbf{u}(\theta_j) = \mathbf{u}(\theta_j)^T\mathbf{p}_{i,\mathrm{n}}$ is related to a specific, viz. the i-th, sensor and the vector $\mathbf{u}(\theta_j)$ to a specific, viz. the j-th, source. In general, from our notation it is clear immediately that v_i is an element of a column vector and v^j an element of a row vector.

3.1.2 Example: Uniform Linear Array (ULA)

Most literature on DOA estimation considers an array with its sensors distributed uniformly on a line. Such an array is commonly referred to as a *Uniform Linear Array (ULA)* [84, 119, 126, 147, 179, 183]. See Fig. 1.8 in Chapter 1 for an example. A ULA can be considered as a special case of the planar array in the previous section because it is a kind of 'degenerate version' of it. Assume that the sensors are located on the positive p^1-axis and that the first sensor is positioned in the origin. Then, the position $\tilde{\mathbf{p}}_i$ of the i-th sensor ($1 \leq i \leq D$) is given by $\tilde{\mathbf{p}}_i = \begin{bmatrix} p_i^1 & p_i^2 \end{bmatrix}$, where $p_i^1 = (i-1)d$ with d the sensor spacing, and $p_i^2 = 0$, i.e. $\tilde{\mathbf{p}}_i = \begin{bmatrix} (i-1)d & 0 \end{bmatrix}$. Now, the phase $\phi_i(\theta) \triangleq \omega_{\mathrm{c}}\tau_i(\theta)$ in (3.1.3) can be written as follows:

$$\begin{aligned}\phi_i(\theta) = \omega_{\mathrm{c}}\tau_i(\theta) = \frac{\omega_{\mathrm{c}}}{c}\tilde{\mathbf{p}}_i\mathbf{u}(\theta) = \frac{\omega_{\mathrm{c}}}{c}(i-1)d\sin(\theta) = (i-1)\,2\pi\frac{d}{\lambda_{\mathrm{c}}}\sin(\theta)\\ = (i-1)\,2\pi d_{\mathrm{n}}\sin(\theta) = (i-1)\xi(\theta) \qquad \forall\,\theta\in\Theta, \quad \forall\,1\leq i\leq D\,,\end{aligned} \tag{3.1.9}$$

where:

$$\xi(\theta) \triangleq 2\pi\frac{d}{\lambda_{\mathrm{c}}}\sin(\theta) = 2\pi d_{\mathrm{n}}\sin(\theta) \qquad \forall\,\theta\in\Theta\,, \tag{3.1.10}$$

and d_{n} is the sensor spacing that is normalized w.r.t. the carrier wavelength λ_{c}, i.e.:

$$d_{\mathrm{n}} \triangleq d/\lambda_{\mathrm{c}}\,. \tag{3.1.11}$$

Substituting (3.1.9) into (3.1.5) gives the complex instantaneous transfer $z_i(\theta)$ from a source at DOA θ to the i-th sensor for a ULA:

$$z_i(\theta) = \exp\big(\jmath\,\phi_i(\theta)\big) = \exp\big(\jmath\,(i-1)\xi(\theta)\big) \qquad \forall\,\theta\in\Theta, \quad \forall\,1\leq i\leq D\,. \tag{3.1.12}$$

Hence, from (3.1.7) it follows that the resulting array response vector $\mathbf{z}(\theta)$ is given by:

$$\mathbf{z}(\theta) = \begin{bmatrix} z_1(\theta)\\ z_2(\theta)\\ \vdots\\ z_D(\theta)\end{bmatrix} = \begin{bmatrix} 1\\ \exp\big(\jmath\,\xi(\theta)\big)\\ \vdots\\ \exp\big(\jmath\,(D-1)\xi(\theta)\big)\end{bmatrix} = \begin{bmatrix} 1\\ \exp\big(\jmath\,2\pi d_{\mathrm{n}}\sin(\theta)\big)\\ \vdots\\ \exp\big(\jmath\,(D-1)2\pi d_{\mathrm{n}}\sin(\theta)\big)\end{bmatrix} \quad \forall\,\theta\in\Theta\,. \tag{3.1.13}$$

As has been stated in the previous section, in order to allow the unique determination of the source DOA's from the sensor data, θ and $\mathbf{z}(\theta)$ must be related one-to-one for all $\theta \in \Theta$. Before explaining how the (normalized) sensor positions (or just the sensor spacing in the ULA case) can be chosen in such a way that this condition is satisfied, we first deduce a 'sensible range' Θ for θ by pointing out two kinds of ambiguities which are due to the properties of the sine function. Firstly, $\mathbf{z}(\theta)$ is periodic with period 2π because of the periodicity of the function $\sin(\theta)$. Hence, $\mathbf{z}(\theta) = \mathbf{z}(\theta + k2\pi)$ for all $\theta \in \Theta$ and $k \in \mathbb{Z}$. Secondly, $\mathbf{z}(\theta) = \mathbf{z}(\pi - \theta)$ for all $\theta \in \Theta$ because $\sin(\theta) = \sin(\pi - \theta)$ for all $\theta \in \Theta$. This latter result also follows from the geometry of the problem: two sources at locations symmetric w.r.t. the array (line), viz. one at θ and the other at $\pi - \theta$, yield identical sets of propagation time differences, i.e. $\tau_i(\theta) = \tau_i(\pi - \theta)$ for all $1 \leq i \leq D$, and thus cannot be distinguished. These two ambiguities imply that DOA's can never be determined uniquely for more than 'one side' of the ULA. Therefore, the parameter space of interest for a ULA is often chosen to be $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$. Note that this interval can also immediately be deduced from the fact that the sine function is injective on it. As we have remarked in the previous section, it also depends on the normalized sensor positions whether a DOA θ and its associated array response vector $\mathbf{z}(\theta)$ are actually related one-to-one for $\theta \in \Theta$. Choosing the parameter interval $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$ is necessary but not sufficient to guarantee this. This can be seen as follows. From (3.1.10) it is clear that $\xi(\theta)$ is a bijective function on $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$ ranging from $-2\pi d_{\text{n}}$ to $2\pi d_{\text{n}}$. Furthermore, from (3.1.12) and (3.1.13) it can be seen that $\mathbf{z}(\theta)$ is unique on Θ only if $|\xi(\theta)| \leq \pi$. Note that $\xi(\theta)$ can be considered as a kind of normalized spatial frequency. Now, since the condition $|\xi(\theta)| = 2\pi d_{\text{n}} |\sin(\theta)| \leq \pi$ implies $d_{\text{n}} |\sin(\theta)| \leq \frac{1}{2}$, and $\max_{\theta \in \Theta} |\sin(\theta)| = 1$, the condition $d_{\text{n}} \leq \frac{1}{2}$ follows. Hence, we can conclude that $\mathbf{z}(\theta)$ is an injective function-valued vector/mapping from Θ to $\boldsymbol{\mathcal{A}}(\Theta)$ iff the normalized sensor spacing d_{n} is smaller than or equal to $\frac{1}{2}$. This implies that θ can be determined uniquely from $\mathbf{z}(\theta)$ iff $d_{\text{n}} \leq \frac{1}{2}$, and the array is said to be unambiguous (see Section 3.1.1). Note that the condition $d_{\text{n}} \leq \frac{1}{2}$ is equivalent to $d \leq \frac{\lambda_{\text{c}}}{2}$. Since d can be considered as the spatial sampling distance of the ULA that samples a wavefield uniformly in space, this condition can be interpreted as the spatial version of Shannon's sampling theorem [97,179]. It guarantees that there is no *spatial aliasing* whenever it is satisfied.

3.1.3 Geometrical properties of DOA observation model

The geometry of the observation model (3.1.1) is of great interest because it provides a lot of insight [97, 147, 154, 179, 183]. The 'informative' part of the (generally) length-D observation vector $\mathbf{x}[n]$, i.e. the (noise-free) part related to the source signals, is a linear combination of the S array response vectors $\mathbf{z}(\theta^1), \ldots, \mathbf{z}(\theta^S)$ with coefficients $s_j[n]$. Hence, this part is confined to an S-dimensional subspace of the complex D-dimensional space $\mathcal{L}\left(\mathbf{z}(\theta^1), \ldots, \mathbf{z}(\theta^S)\right) = \mathcal{R}_{\text{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$[‡]. Due to the noise, a part of $\mathbf{x}[n]$ extends into the orthogonal complement $\mathcal{R}_{\text{c}}^{\perp}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ of $\mathcal{R}_{\text{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$.

Since the array response vectors $\mathbf{z}(\theta^1), \ldots, \mathbf{z}(\theta^S)$ are elements of both the subspace $\mathcal{R}_{\text{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ and the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$, they lie in the intersection $\mathcal{R}_{\text{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big) \cap \boldsymbol{\mathcal{A}}(\Theta)$. In fact, it can be shown that under some common assumptions (e.g. see Section 3.2.1.1) the source array response vectors are determined uniquely by the intersections between $\mathcal{R}_{\text{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$

[‡]See Appendix A and Definition C.2.1 for the definition and meaning of the 'linear span' symbol $\mathcal{L}(\cdot)$ and the 'column range symbol' $\mathcal{R}_{\text{c}}(\cdot)$; see also the list of symbols on page ??.

and $\boldsymbol{\mathcal{A}}(\Theta)$. Hence, if the array is unambiguous (see Section 3.1.1), i.e. if the mapping $\mathbf{z}(\theta)$ from Θ to $\boldsymbol{\mathcal{A}}(\Theta)$ is *injective*, these intersections determine the source DOA's uniquely. This implies that the considered array response matrix is unambiguous according to the following definition.

Definition 3.1.2. Unambiguous array response matrix. An array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ is said to be unambiguous over Θ if for all $\tilde{\boldsymbol{\theta}}$ with $\theta^1, \ldots, \theta^S \in \Theta$ the source DOA's can be determined uniquely from the intersections between $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ and the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$.

Note that this definition implies that $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ must have full rank for any collection of S distinct DOA's in $\tilde{\boldsymbol{\theta}}$ with $\theta^1, \ldots, \theta^S \in \Theta$, and that a necessary condition for the array response matrix to be unambiguous is that the array must be unambiguous as well. As an example, the array response matrix associated with a ULA with $d_{\mathrm{n}} \leq \frac{1}{2}$ and $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$ is unambiguous because it can be shown that $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ has full rank whenever $D \geq S$ and all angles in $\tilde{\boldsymbol{\theta}}$ are different [147]. Summarizing, for an unambiguous array response matrix the intersections between the subspace $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ and the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$ *are* the array response vectors and determine the source DOA's uniquely.

3.2 Subspace Methods for DOA estimation based on Second Order Statistics

In this section, we discuss the main principles of estimating DOA's by subspace methods that are based on Second Order Statistics [16, 97, 115, 127, 137, 138, 147, 179, 183]. Most subspace methods rely on the partitioning of the eigen-space of the lag-zero sensor correlation matrix $\mathbf{R}^x$ into a subspace associated with the source signals, the so-called *signal subspace*, and a subspace associated with the noise signals, the so-called *noise subspace*. As has been explained in Section 3.1.3, the desired parameters, e.g. the source DOA's, are determined by the intersections between the signal subspace and the array manifold and can be extracted in various ways. In Section 3.2.1, we start by examining the structure of the covariance matrix and its decomposition into the signal and noise subspaces. Then, in Section 3.2.2, the well-known MUSIC method [16, 127, 137, 138, 147, 179] for the extraction of the DOA's is discussed, followed in Section 3.2.3 by the so-called ROOT-MUSIC method [10, 147, 179]. Finally, the min-norm method [98, 99, 147, 179] is discussed in Section 3.2.4.

3.2.1 Sensor correlation structure, signal and noise subspaces

Firstly, in Section 3.2.1.1 we will formulate the commonly adopted assumptions underlying SOS based subspace methods. Then, in Section 3.2.1.2 the structure of the lag-zero sensor correlation matrix $\mathbf{R}^x$ is derived and analyzed. Subsequently, in Section 3.2.1.3 a signal and a noise subspace are defined in terms of the eigenstructure of $\mathbf{R}^x$. After that, in Section 3.2.1.4 several important conclusions are drawn about the relationships between the array response vectors and the signal and noise subspaces. Based on these conclusions, a system of equations satisfied by the source DOA's is formulated. Finally, in Section 3.2.1.5 we highlight some consequences of, and techniques for, working with estimated statistics in practice.

3.2.1.1 Assumptions of SOS based subspace methods

The structure of the lag-zero sensor correlation (or covariance) matrix $\mathbf{R}^x \triangleq E\{\mathbf{x}[n]\mathbf{x}[n]^H\}$ and the related subspace principles are based on several commonly adopted assumptions [16, 97, 119, 137, 138, 147, 179] that are summarized in the list at the end of this section. Firstly, it is assumed that all involved signals are realizations of zero mean, stationary, and ergodic random processes (*AS1*). Since conventional subspace methods are based on the lag-zero sensor correlation matrix, they neglect any temporal structure that might be present in the data (compare this to the ICA category of blind methods discussed in the previous chapter). As in Section 2.3.1, for convenience of notation all time indices are dropped and any time signal $v[n]$ is considered as a random variable v instead of a time series; the observed values $v[n]$ are then samples of this random variable. For example, (3.1.1) is written as $\mathbf{x} = \sum_{j=1}^{S} \mathbf{z}(\theta^j)\, s_j + \boldsymbol{\nu} = \mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\, \mathbf{s} + \boldsymbol{\nu}$, where $\mathbf{x}$, $\mathbf{s}$ and $\boldsymbol{\nu}$ are considered as random vectors with samples $\mathbf{x}[n]$, $\mathbf{s}[n]$ and $\boldsymbol{\nu}[n]$ respectively. Furthermore, the sensor correlation matrix $\mathbf{R}^x$ can now be defined as $\mathbf{R}^x \triangleq E\{\mathbf{x}\mathbf{x}^H\}$. Secondly, it is assumed that the source signals are mutually uncorrelated with non-zero variances $\{(\sigma_j^s)^2\}_{1\leq j\leq S}$ (*AS2*), i.e. $\mathbf{R}^s \triangleq E\{\mathbf{s}\mathbf{s}^H\} = \mathrm{diag}\big((\sigma_1^s)^2, \ldots, (\sigma_S^s)^2\big)$ (other, more general, scenarios assume that $\mathbf{R}^s$ is nonnegative definite). For completeness and clarity, we explicitly assume that the source DOA's are different (*AS3*). Next, it is assumed that the source and sensor noise signals are mutually uncorrelated, i.e. $\mathbf{R}^{s\nu} \triangleq E\{\mathbf{s}\boldsymbol{\nu}^H\} = \mathbf{0}$ (*AS4*). For simplicity it is assumed that the sensor noise is spatially white, i.e. $\mathbf{R}^{\nu} \triangleq E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\} = (\sigma^{\nu})^2\mathbf{I}$ (*AS5*). If the correlation structure is known it is also possible to deal with spatially non-white sensor noise. Note that any temporal structure that might be present in the source and/or noise signals is irrelevant for the rationale behind the conventional subspace approach because it is ignored. Finally, some very important assumptions regarding the properties of the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$ and the array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ are adopted. One of the most important ones is that the functional dependence of the array response vector $\mathbf{z}(\theta)$ on the source DOA θ is known a priori for all $\theta \in \Theta$ (*AS6*). Also, it is assumed that the number of sensors is larger than the number of sources, i.e. $D > S$ (*AS7*). As we will see, this assumption implies that $\mathbf{R}^x$ has a non-trivial noise subspace, which is a prerequisite for all subspace methods. Finally, it is assumed that the array response matrix is unambiguous (*AS8*). As we have discussed in Section 3.1.3, this means that the source DOA's can be determined uniquely from the intersections between the column range $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ and the known array manifold $\boldsymbol{\mathcal{A}}(\Theta)$. Summarizing, lag-zero SOS based subspace methods exploit the following assumptions [16, 97, 119, 137, 138, 147, 179, 183]:

AS1: All involved signals are realizations of zero-mean, stationary, and ergodic random processes;

AS2: The source signals have non-zero variance and are mutually uncorrelated: $\mathbf{R}^s = \mathrm{diag}\big((\sigma_1^s)^2, \ldots, (\sigma_S^s)^2\big)$ *with* $(\sigma_j^s)^2 \neq 0 \quad \forall\, 1 \leq j \leq S$;

AS3: The source DOA's are different: $\theta^i \neq \theta^j \quad \forall\, 1 \leq i \neq j \leq S$;

AS4: The source and sensor noise signals are mutually uncorrelated: $\mathbf{R}^{s\nu} = \mathbf{0}$;

AS5: The sensor noise is spatially white, i.e. $\mathbf{R}^{\nu} = (\sigma^{\nu})^2\mathbf{I}$;

AS6: The parameterization of the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$*, i.e. the functional dependence of* $\mathbf{z}(\theta)$ *on* θ *for all* $\theta \in \Theta$*, is known a priori;*

AS7: The number of sensors is larger than the number of sources, i.e. $D > S$;

AS8: The array response matrix $\mathbf{A}(\tilde{\boldsymbol{\theta}})$ is unambiguous over the parameter space Θ, i.e. the source DOA's can be determined uniquely from $\mathcal{R}_{\mathrm{c}}(\mathbf{A}(\tilde{\boldsymbol{\theta}})) \cap \boldsymbol{\mathcal{A}}(\Theta)$.

3.2.1.2 Structure of sensor correlation matrix $\mathbf{R}^x$

The lag-zero sensor correlation matrix $\mathbf{R}^x \in \mathbb{C}_D^D$ easily follows from (3.1.1) and assumptions *AS1-AS8* on the facing page:

$$\mathbf{R}^x = \sum_{j=1}^{S} (\sigma_j^s)^2 \mathbf{z}(\theta^j)\mathbf{z}(\theta^j)^H + \mathbf{R}^\nu = \underbrace{\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H}_{\text{Due to source signals}} + \underbrace{(\sigma^\nu)^2\,\mathbf{I}_D^D}_{\text{Due to noise signals}} \quad . \tag{3.2.1}$$

Hence, $\mathbf{R}^x$ can be written as the sum of two auto-correlation matrices. The first one, $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$, is due to the source signals and involves the source auto-correlation matrix $\mathbf{R}^s$ and the array response matrix $\mathbf{A}(\tilde{\boldsymbol{\theta}})$. The second one, $\mathbf{R}^\nu = (\sigma^\nu)^2\mathbf{I}$, is the noise correlation matrix. Because $\mathbf{A}(\tilde{\boldsymbol{\theta}})$ has full rank S owing to *AS8*, and $\mathbf{R}^s$ has rank S owing to *AS2*, the part of $\mathbf{R}^x$ that is due to the source signals also has rank S [115, 147]:

$$\operatorname{rank}\Big(\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H\Big) = S\,. \tag{3.2.2}$$

This property implies that if the number of sources S is unknown, it can be computed/estimated as the (effective) rank of $\mathbf{R}^x$. Obviously, the part of $\mathbf{R}^x$ that is due to the noise signals has rank D whenever $(\sigma^\nu)^2 \neq 0$ because then $\operatorname{rank}(\mathbf{R}^\nu) = \operatorname{rank}(\mathbf{I}_D^D) = D$ (note that $\mathbf{I}_D^D$ is the identity matrix of size D by D, see Appendix A). Hence, in general:

$$\operatorname{rank}(\mathbf{R}^x) = D\,. \tag{3.2.3}$$

3.2.1.3 Eigenstructure of $\mathbf{R}^x$; signal and noise subspaces

In addition to performing a decomposition of $\mathbf{R}^x$ into a sum of two auto-correlation matrices, as in (3.2.1), it is also possible and useful to perform an *Eigenvalue Decomposition (EVD)* [72,115] of $\mathbf{R}^x$ and split the result into two parts [16,97,137,138,147,154]. In fact, as we will see in the sequel, each term of either decomposition is closely linked to the corresponding term in the other decomposition. In order to gain insight into the properties of the EVD of $\mathbf{R}^x$, we first consider the EVD of the 'source signal part' $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$, i.e. the first term at the right hand side of (3.2.1), and then consider the influence of the 'sensor noise part' $(\sigma^\nu)^2\mathbf{I}$ on the eigenstructure of $\mathbf{R}^x$. Let $\{\lambda_i^s\}^{1\leq i\leq D}$ and $\{\mathbf{u}^i\}^{1\leq i\leq D}$ be the eigenvalues and eigenvectors respectively of $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$ with the eigenvalues arranged in decreasing order, i.e.:

$$\Big(\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H\Big)\mathbf{u}^i = \lambda_i^s\mathbf{u}^i \quad \forall\, 1 \leq i \leq D \quad \text{with} \quad \lambda_1^s \geq \lambda_2^s \geq \cdots \geq \lambda_D^s\,. \tag{3.2.4}$$

From (3.2.2), it follows that the last $D - S$ eigenvalues $\lambda_{S+1}^s, \ldots, \lambda_D^s$ are zero. Because the considered matrix is Hermitian and positive definite, the first S eigenvalues $\lambda_1^s, \ldots, \lambda_S^s$ are real-valued and greater than zero [76]. In addition, all eigenvectors can be computed in such a way that they are orthonormal [150]. Therefore, from now on we assume that the eigenvectors $\{\mathbf{u}^i\}^{1\leq i\leq D}$ form an orthonormal set. Writing the equalities for $1 \leq i \leq S$ in (3.2.4) simultaneously in one matrix equation yields:

$$\Big(\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H\Big)\mathbf{U}^s = \mathbf{U}^s\boldsymbol{\Lambda}^s, \tag{3.2.5}$$

where:

$$\mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^S\end{bmatrix} \in \mathbb{C}_D^S \quad \text{and} \quad \mathbf{\Lambda}^s \triangleq \operatorname{diag}(\lambda_1^s, \ldots, \lambda_S^s) \in \mathbb{R}_S^S \tag{3.2.6}$$

are matrices containing the so-called *signal eigenvectors* and *signal eigenvalues* respectively. Note that $\mathbf{U}^s$ and $\mathbf{\Lambda}^s$ depend on the actual source DOA's in $\tilde{\boldsymbol{\theta}}$ because they are determined by the sensor correlation matrix $\mathbf{R}^x$, which is a function of $\tilde{\boldsymbol{\theta}}$, as is evident from (3.2.1).

Now that we have considered the eigenstructure of the 'source signal part' $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$ of $\mathbf{R}^x$, we will examine the influence of the 'sensor noise part' $(\sigma^\nu)^2\mathbf{I}$ on the eigenstructure of $\mathbf{R}^x$. Consider the matrix-vector product $\mathbf{R}^x\mathbf{u}^i$, i.e. the product of $\mathbf{R}^x$ with the i-th eigenvector of $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$:

$$\mathbf{R}^x\mathbf{u}^i \overset{(3.2.1)}{=} \left(\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H\right)\mathbf{u}^i + (\sigma^\nu)^2\mathbf{I}\mathbf{u}^i \overset{(3.2.4)}{=} \left(\lambda_i^s + (\sigma^\nu)^2\right)\mathbf{u}^i \quad \forall\, 1 \le i \le D\,. \tag{3.2.7}$$

Hence, the eigenvectors of $\mathbf{R}^x$ are the same as those of $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$, and the eigenvalues equal the eigenvalues of $\mathbf{A}(\tilde{\boldsymbol{\theta}})\mathbf{R}^s\mathbf{A}(\tilde{\boldsymbol{\theta}})^H$ increased by the noise variance $(\sigma^\nu)^2$. Thus, denoting the eigenvalues of $\mathbf{R}^x$ by $\lambda_1^x, \ldots, \lambda_D^x$, it is clear that $\lambda_i^x = \lambda_i^s + (\sigma^\nu)^2$ for all $1 \le i \le D$, and, more specifically:

$$\lambda_i^x = \begin{cases} \lambda_i^s + (\sigma^\nu)^2, & i = 1, \ldots, S\,; \\ (\sigma^\nu)^2, & i = S+1, \ldots, D\,. \end{cases} \tag{3.2.8}$$

Hence, the eigenvalues can be divided into two groups. The eigenvalues $\{\lambda_i^x\}_{1\le i\le S} = \{\lambda_i^s + (\sigma^\nu)^2\}_{1\le i\le S}$ in the first group are called *signal eigenvalues* because they are due to the source signal part of $\mathbf{R}^x$; more appropriately, they are due to the source and sensor noise correlation matrices (and the array response matrix, of course). The eigenvalues $\{\lambda_i^x\}_{S+1\le i\le D} = \{(\sigma^\nu)^2\}_{S+1\le i\le D}$ in the second group are called *noise eigenvalues* because they are due to the noise signals. Likewise, the corresponding eigenvectors can be divided into two groups. The eigenvectors $\{\mathbf{u}^i\}^{1\le i\le S}$ corresponding to the signal eigenvalues are called *signal eigenvectors* and span an S-dimensional subspace $\mathcal{L}\left(\{\mathbf{u}^i\}^{1\le i\le S}\right)$ called *signal subspace*. The eigenvectors $\{\mathbf{u}^i\}^{S+1\le i\le D}$ corresponding to the noise eigenvalues are called *noise eigenvectors* and span a $(D-S)$-dimensional subspace $\mathcal{L}\left(\{\mathbf{u}^i\}^{S+1\le i\le D}\right)$ called *noise subspace*. Note that the source array response vectors lie completely in the signal subspace, whereas the noise vector $\boldsymbol{\nu}$ has components in both subspaces. Summarizing, the EVD of $\mathbf{R}^x$ can be expressed as follows:

$$\mathbf{R}^x = \sum_{i=1}^{D} \lambda_i^x \mathbf{u}^i(\mathbf{u}^i)^H = \sum_{i=1}^{S} \left(\lambda_i^s + (\sigma^\nu)^2\right)\mathbf{u}^i(\mathbf{u}^i)^H + \sum_{i=S+1}^{D} (\sigma^\nu)^2\mathbf{u}^i(\mathbf{u}^i)^H. \tag{3.2.9}$$

In matrix notation, this decomposition can be written as:

$$\boxed{\mathbf{R}^x = \mathbf{U}^x\mathbf{\Lambda}^x(\mathbf{U}^x)^H = \mathbf{U}^s\mathbf{\Lambda}^s(\mathbf{U}^s)^H + \mathbf{U}^\nu\mathbf{\Lambda}^\nu(\mathbf{U}^\nu)^H\,,} \tag{3.2.10}$$

where:

$\mathbf{U}^x \triangleq \begin{bmatrix}\mathbf{u}^1 \cdots \mathbf{u}^D\end{bmatrix} \in \mathbb{C}_D^D$, $\mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 \cdots \mathbf{u}^S\end{bmatrix} \in \mathbb{C}_D^S$ and $\mathbf{U}^\nu \triangleq \begin{bmatrix}\mathbf{u}^{S+1} \cdots \mathbf{u}^D\end{bmatrix} \in \mathbb{C}_D^{D-S}$ are matrices containing the 'sensor', 'signal' and 'noise' eigenvectors respectively, and

$$\mathbf{\Lambda}^x \triangleq \operatorname{diag}(\lambda_1^x, \ldots, \lambda_D^x) \in \mathbb{R}_D^D, \quad \mathbf{\Lambda}^s \triangleq \operatorname{diag}\left(\lambda_1^s + (\sigma^\nu)^2, \ldots, \lambda_S^s + (\sigma^\nu)^2\right) \in \mathbb{R}_S^S,$$
$$\text{and} \quad \mathbf{\Lambda}^\nu \triangleq \operatorname{diag}\left((\sigma^\nu)^2, \ldots, (\sigma^\nu)^2\right) \in \mathbb{R}_{D-S}^{D-S} \tag{3.2.11}$$

are diagonal matrices containing the sensor, (source) signal (plus noise), and noise eigenvalues respectively. Once again, notice that all matrices $\mathbf{U}^x$, $\mathbf{\Lambda}^x$, $\mathbf{U}^s$, $\mathbf{\Lambda}^s$, $\mathbf{U}^\nu$ and $\mathbf{\Lambda}^\nu$ depend on $\tilde{\boldsymbol{\theta}}$ because they are determined by the sensor correlation matrix $\mathbf{R}^x$, which is a function of $\tilde{\boldsymbol{\theta}}$. The matrices resulting from the EVD satisfy several properties. Firstly, note that:

$$\mathbf{U}^x = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix} \text{ with } \mathbf{U}^x(\mathbf{U}^x)^H = (\mathbf{U}^x)^H\mathbf{U}^x = \mathbf{I}_D^D, \text{ and } \mathbf{\Lambda}^x = \begin{bmatrix}\mathbf{\Lambda}^s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}^\nu\end{bmatrix}. \quad (3.2.12)$$

Because the eigenvectors of $\mathbf{R}^x$ are orthonormal, the signal and noise subspaces are orthogonal. In other words, the signal subspace is the orthogonal complement[‡] of the noise subspace and vice versa. For reference purposes, we formulate this mathematically in several equivalent ways:

$$\begin{aligned}\mathbf{U}^s \perp \mathbf{U}^\nu \quad &\equiv \quad (\mathbf{U}^s)^H\mathbf{U}^\nu = \mathbf{0}_S^{D-S} \quad \equiv \quad (\mathbf{U}^\nu)^H\mathbf{U}^s = \mathbf{0}_{D-S}^S \\ \text{or} \quad &\langle \mathbf{v}, \mathbf{w}\rangle \triangleq \mathbf{v}^H\mathbf{w} = 0 \quad \forall\, \mathbf{v} \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right),\ \forall\, \mathbf{w} \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right). \end{aligned} \quad (3.2.13)$$

In addition, in terms of the fundamental subspaces considered in linear algebra [115,150] and discussed in Appendix C, we can summarize these important properties as follows:

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^s\big) = \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right)\right)^T \doteqdot \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right); \quad (3.2.14)$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^\nu\big) = \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right)\right)^T \doteqdot \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right), \quad (3.2.15)$$

where the symbol '$\doteqdot$' denotes *isomorphism*. The results obtained above imply that $\mathbf{R}^x\mathbf{U}^x$ can be decomposed as follows:

$$\mathbf{R}^x\mathbf{U}^x = \mathbf{U}^x\mathbf{\Lambda}^x = \mathbf{R}^x\begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix} = \begin{bmatrix}\mathbf{R}^x\mathbf{U}^s & \mathbf{R}^x\mathbf{U}^\nu\end{bmatrix} = \begin{bmatrix}\mathbf{U}^s\mathbf{\Lambda}^s & \mathbf{U}^\nu\mathbf{\Lambda}^\nu\end{bmatrix}. \quad (3.2.16)$$

For later use, we define two orthogonal projection matrices $\mathbf{P}^s$ and $\mathbf{P}^\nu$ for projecting a vector onto the signal and noise subspaces respectively:

$$\mathbf{P}^s \triangleq \mathbf{U}^s(\mathbf{U}^s)^H, \quad \mathbf{P}^\nu \triangleq \mathbf{U}^\nu(\mathbf{U}^\nu)^H. \quad (3.2.17)$$

Note that because of the 'resolution of identity' property of orthogonal projection matrices [115], the following equality is satisfied:

$$\mathbf{I} = \mathbf{P}^s + \mathbf{P}^\nu = \mathbf{U}^s(\mathbf{U}^s)^H + \mathbf{U}^\nu(\mathbf{U}^\nu)^H. \quad (3.2.18)$$

3.2.1.4 Relation between array response vectors, signal and noise subspaces; obtaining a system of equations satisfied by source DOA's

Now we are in a position to draw several important conclusions about the relation between the array response vectors and the signal and noise subspaces [147,154]. Firstly, we show that the linear span $\mathcal{L}\left(\{\mathbf{u}^i\}^{1\le i\le S}\right)$, i.e. the signal subspace spanned by the signal eigenvectors $\mathbf{u}^1, \ldots, \mathbf{u}^S$, is equal to the linear span $\mathcal{L}\left(\{\mathbf{z}(\theta^j)\}^{1\le j\le S}\right)$ of the array response vectors whenever $\mathbf{R}^s$ has full rank (which is the case in our problem definition, see *AS2* on page 91) [147]. In matrix language, this reads $\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$. For reference purposes, we summarize these results explicitly in the following theorem.

Theorem 3.2.1. Linear span of signal eigenvectors equals that of array response vectors. *If the source auto-correlation matrix $\mathbf{R}^s$ has full rank, the linear span of the signal eigenvectors equals the linear span of the array response vectors:*

$$\boxed{\mathcal{L}\left(\{\mathbf{u}^i\}^{1\le i\le S}\right) = \mathcal{L}\left(\{\mathbf{z}(\theta^j)\}^{1\le j\le S}\right) \quad \equiv \quad \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big).} \quad (3.2.19)$$

[‡]See Appendix C for a brief summary of important linear algebra related concepts and definitions, such as fundamental subspaces, column and row spaces, orthogonality, orthogonal complement, and so on.

Proof. We prove this assertion using (3.2.5). Noting that $\mathbf{\Lambda}^s$ is invertible, and multiplying both sides of (3.2.5) from the right by $(\mathbf{\Lambda}^s)^{-1}$, yields the equality $\mathbf{U}^s = \Big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\mathbf{R}^s\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H\Big)\mathbf{U}^s(\mathbf{\Lambda}^s)^{-1}$. This shows that each column of $\mathbf{U}^s$ belongs to the column space of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$, i.e.:

$$\mathbf{u}^i \in \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big) \qquad \forall\, 1 \leq i \leq S\,.$$

Likewise, noting that $\mathbf{R}^s\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H\mathbf{U}^s$ is invertible because $\mathbf{R}^s$, $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ and $\mathbf{U}^s$ all have full rank, and multiplying both sides of (3.2.5) from the right by $\big(\mathbf{R}^s\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H\mathbf{U}^s\big)^{-1}$, yields the equality $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) = \mathbf{U}^s\mathbf{\Lambda}^s\big(\mathbf{R}^s\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H\mathbf{U}^s\big)^{-1}$. This shows that each column of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ belongs to the column space of $\mathbf{U}^s$, i.e.:

$$\mathbf{z}(\theta^j) \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) \qquad \forall\, 1 \leq j \leq S\,.$$

Hence, because each column of $\mathbf{U}^s$ can be expressed as a linear combination of the columns of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ and vice versa, we conclude that $\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$. □

Corollary 3.2.2. Noise subspace is isomorphic to left null space of $\mathbf{A}\big(\tilde{\theta}\big)$***.***
Under the assumption(s) of Theorem 3.2.1 the noise subspace is isomorphic to the left null space of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$*:*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right) \doteqdot \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big) \qquad \equiv \qquad \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right) = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)\Big)^T\,.} \tag{3.2.20}$$

Proof. Property *FSP1* on page 450 states that the orthogonal complement of the column space of a matrix is isomorphic to its left null space. Applying this property to both sides of (3.2.19) yields:

$$\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^s\big)\Big)^\perp = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)\Big)^\perp \quad \Longrightarrow \quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^s\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)\,.$$

Combining this result with (3.2.14) directly implies the corollary. □

Corollary 3.2.3. Noise subspace equals (right) null space of Hermitian of $\mathbf{A}\big(\tilde{\theta}\big)$***.***
Under the assumption(s) of Theorem 3.2.1 the noise subspace equals the (right) null space of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H$*:*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right) = \mathcal{N}_{\mathrm{r}}\Big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H\Big)\,.} \tag{3.2.21}$$

Proof. Property *FSP4* on page 450 states that the left null space of a matrix is isomorphic to the right null space of its Hermitian. Applying this property to (3.2.20) yields the corollary. □

Using the results that we have developed so far, from the knowledge of $\mathbf{U}^\nu$ (recall that we can consider the EVD matrices $\mathbf{U}^s$, $\mathbf{U}^\nu$, etc. to be known because $\mathbf{R}^x$ is known or can be estimated) we can derive a system of equations that is satisfied by the source DOA's. From Theorem 3.2.1 and Corollaries 3.2.2 and 3.2.3, it follows immediately that:

$$\boxed{(\mathbf{U}^\nu)^H\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) = \mathbf{0}_{D-S}^S \quad \equiv \quad (\mathbf{u}^i)^H\mathbf{z}(\theta^j) = 0 \quad \forall\, S+1 \leq i \leq D,\ \forall\, 1 \leq j \leq S\,.} \tag{3.2.22}$$

Hence, considering the array response vector $\mathbf{z}(\theta)$ as a function-valued vector, we have:

$$\boxed{\langle \mathbf{u}, \mathbf{z}(\theta) \rangle \big|_{\theta=\theta^j} = 0 \qquad \forall\, 1 \leq j \leq S, \quad \forall\, \mathbf{u} \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{\nu}\right)\ ,} \tag{3.2.23}$$

where the inner product denoted by $\langle \cdot, \cdot \rangle$ is a function from $\mathbb{C}_D \times \mathbb{C}_D$ to $\mathbb{C}$ defined by $\langle \mathbf{v}, \mathbf{w} \rangle \triangleq \mathbf{v}^H \mathbf{w} = \sum_{i=1}^{D} (v_i)^* \, w_i \quad \forall\ \mathbf{v}, \mathbf{w} \in \mathbb{C}_D$ (see Appendices A and D). Equation (3.2.23) is of fundamental importance for SOS based subspace methods. It shows that *the desired source DOA's can be obtained/estimated by solving θ from a system of equations of the form $\langle \mathbf{u}, \mathbf{z}(\theta) \rangle = 0$ (or $\mathbf{u}^H \mathbf{z}(\theta) = 0$, which is the same) for a 'sufficient' number of vectors $\mathbf{u}$ belonging to the noise subspace*[‡]. Obviously, in practice such a system can only be solved approximately because only an approximate measured version of the sensor correlation matrix is available; see the next section. Another straightforward and enlightening derivation of the results in (3.2.22) and (3.2.23) can be obtained by considering the product $\mathbf{R}^x \mathbf{U}^{\nu}$ [147]:

$$\mathbf{R}^x \mathbf{U}^{\nu} = \mathbf{U}^{\nu} \mathbf{\Lambda}^{\nu} = (\sigma^{\nu})^2 \mathbf{U}^{\nu} = \mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \mathbf{R}^s \mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H \mathbf{U}^{\nu} + (\sigma^{\nu})^2 \mathbf{U}^{\nu}.$$

The first equality follows from the definition of $\mathbf{U}^{\nu}$ and $\mathbf{\Lambda}^{\nu}$ (see (3.2.7), (3.2.8) and (3.2.10)), the second from (3.2.8) and (3.2.11), and the third from (3.2.1). The last equality implies that $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \mathbf{R}^s \mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H \mathbf{U}^{\nu} = \mathbf{0}$. Since $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \mathbf{R}^s$ has full column rank, this in turn implies that $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)^H \mathbf{U}^{\nu} = \mathbf{0}$, thereby yielding the same result as in (3.2.22) and (3.2.23).

Summarizing, from (3.2.23) it follows that the source DOA's can be found by solving the following system of nonlinear equations for θ:

$$\boxed{\langle \mathbf{u}, \mathbf{z}(\theta) \rangle = 0 \qquad \forall\, \mathbf{u} \in \mathcal{U}^{\nu}\ ,} \tag{3.2.24}$$

where $\mathcal{U}^{\nu}$ is an appropriate set of noise subspace vectors. A set $\mathcal{U}^{\nu}$ is considered appropriate if the associated system (3.2.24) determines the source DOA's uniquely, i.e. if the only solutions to the system are the (true) source DOA's. As we will see in the next sections, the particular choice of such a set is one of the issues that distinguishes one category of subspace methods from another. For example, one way of choosing $\mathcal{U}^{\nu}$ is to exploit the full noise subspace in (3.2.24), e.g. by using all noise eigenvectors $\mathbf{u}^{S+1}, \ldots, \mathbf{u}^{D}$ that are stacked in $\mathbf{U}^{\nu}$. This approach is followed by the so-called Multiple Signal Classification (MUSIC) (see [137] and Section 3.2.2) and Root Multiple Signal Classification (ROOT-MUSIC) (see [10] and Section 3.2.3) subspace algorithms. Hence, in this case $\mathcal{U}^{\nu} = \left\{\mathbf{u}^{S+1}, \ldots, \mathbf{u}^{D}\right\}$ and the source DOA's are given by the solution to the following system of equations:

$$\boxed{(\mathbf{U}^{\nu})^H \mathbf{z}(\theta) = \mathbf{0} \qquad \equiv \qquad \mathbf{z}(\theta)^H \mathbf{U}^{\nu} = \tilde{\mathbf{0}}\ .} \tag{3.2.25}$$

It can be proven [147] that the solution of (3.2.25) is unique (using among other things the fact that the array manifold is assumed to be unambiguous). Another way of choosing $\mathcal{U}^{\nu}$ is to use a single 'special' vector $\mathbf{u} \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{\nu}\right)$ that is chosen such that the single equation $\mathbf{z}(\theta)^H \mathbf{u} = 0$ determines the source DOA's uniquely. This approach is followed by the so-called Minimum-Norm (MIN-NORM) method (see [98, 99] and Section 3.2.4). Since $\mathbf{z}(\theta)$ depends on θ in a nonlinear way, for any choice of $\mathcal{U}^{\nu}$ the solution of system (3.2.24) requires a technique for solving a system of nonlinear equations. This problem is typically considered to be very difficult; we will return to it later on.

[‡] We will address the issue of a 'sufficient' number of noise subspace vectors in later sections.

3.2.1.5 Conventional approach for dealing with estimated statistics

In practice, we have to work with an estimate $\hat{\mathbf{R}}^x$ of the correlation matrix $\mathbf{R}^x \triangleq E\{\mathbf{x}\mathbf{x}^H\}$. In view of the assumptions of stationarity and ergodicity, this estimate is usually computed as $\hat{\mathbf{R}}^x_T \triangleq \frac{1}{T}\sum_{n=1}^{T} \mathbf{x}[n]\mathbf{x}[n]^H$, where $\{\mathbf{x}[n]\}_{n=1}^{T}$ are T (time) samples of the random vector $\mathbf{x}$ (because of the ergodicity assumption, ensemble-averaged quantities obtained from mathematical expectations are equal to the corresponding time-averaged quantities; see also [122, 130, 154], and Sections B.1.3.6 and B.2.6.9). Because $\lim_{T\to\infty} \hat{\mathbf{R}}^x_T = \mathbf{R}^x$, this estimator is *statistically consistent*. If T is finite, $\hat{\mathbf{R}}^x_T$ is only an approximation of $\mathbf{R}^x$ and does not exactly have the same structure and nice properties of $\mathbf{R}^x$. When convenient and not confusing, we will often omit the subscript T from $\hat{\mathbf{R}}^x_T$ in the sequel. Because $\hat{\mathbf{R}}^x$ is an approximation of $\mathbf{R}^x$ the matrices $\hat{\mathbf{U}}^x$, $\hat{\mathbf{\Lambda}}^x$, $\hat{\mathbf{U}}^s$, $\hat{\mathbf{\Lambda}}^s$, $\hat{\mathbf{U}}^\nu$ and $\hat{\mathbf{\Lambda}}^\nu$ in the eigendecomposition:

$$\hat{\mathbf{R}}^x = \hat{\mathbf{U}}^x\hat{\mathbf{\Lambda}}^x(\hat{\mathbf{U}}^x)^H = \hat{\mathbf{U}}^s\hat{\mathbf{\Lambda}}^s(\hat{\mathbf{U}}^s)^H + \hat{\mathbf{U}}^\nu\hat{\mathbf{\Lambda}}^\nu(\hat{\mathbf{U}}^\nu)^H \tag{3.2.26}$$

of $\hat{\mathbf{R}}^x$ deviate from their true values $\mathbf{U}^x$, $\mathbf{\Lambda}^x$, $\mathbf{U}^s$, $\mathbf{\Lambda}^s$, $\mathbf{U}^\nu$ and $\mathbf{\Lambda}^\nu$ respectively. Among other things, this implies that (3.2.24) and (3.2.25) only hold approximately when $\mathbf{U}^\nu$ is replaced by $\hat{\mathbf{U}}^\nu$. Hence, the array response vectors are not completely orthogonal to the estimated noise subspace. Consequently, the system $\{\langle \mathbf{u}, \mathbf{z}(\theta)\rangle = 0\}_{\mathbf{u}\in\mathcal{U}^\nu}$ in (3.2.24) most probably does not contain exact solutions since the equations are only approximations of the 'true equations'. The conventional approach to this type of problem is to define a cost function (see Section 2.5) the minima of which are located exactly at the source DOA's for exactly known statistics, and approximately for estimated or approximately known statistics. A commonly used natural cost function for DOA estimation is the following:

$$\begin{aligned} J_{\text{ss}}(\theta) &\triangleq \sum_{\mathbf{u}\in\mathcal{U}^\nu} \left|\langle \mathbf{u}, \mathbf{z}(\theta)\rangle\right|^2 = \sum_{\mathbf{u}\in\mathcal{U}^\nu} \left|\mathbf{u}^H\,\mathbf{z}(\theta)\right|^2 = \sum_{\mathbf{u}\in\mathcal{U}^\nu} \left|\mathbf{z}(\theta)^H\,\mathbf{u}\right|^2 \\ &= \mathbf{z}(\theta)^H\left(\sum_{\mathbf{u}\in\mathcal{U}^\nu}\mathbf{u}\mathbf{u}^H\right)\mathbf{z}(\theta) = \mathbf{z}(\theta)^H\,\mathbf{F}(\mathcal{U}^\nu)\,\mathbf{z}(\theta)\,, \end{aligned} \tag{3.2.27}$$

where $\mathbf{F}(\mathcal{U}^\nu) \triangleq \sum_{\mathbf{u}\in\mathcal{U}^\nu}\mathbf{u}\mathbf{u}^H$ is a so-called (sub-)frame operator [123]. The subscript 'ss' in $J_{\text{ss}}(\theta)$ stands for 'subspace'. Note that if the vectors $\mathbf{u} \in \mathcal{U}^\nu$ are mutually orthonormal, the frame operator reduces to the standard orthogonal projection operator $\mathbf{P}^\nu$ onto the noise subspace; see (3.2.17) and the next section. In this case, an expression like $\mathbf{z}(\theta)^H\,\mathbf{F}(\mathcal{U}^\nu)\,\mathbf{z}(\theta) = \mathbf{z}(\theta)^H\,\mathbf{P}^\nu\,\mathbf{z}(\theta)$ in (3.2.27) is often called *null-spectrum* [147]. Naturally, we will also use this name for the general expression $\mathbf{z}(\theta)^H\,\mathbf{F}(\mathcal{U}^\nu)\,\mathbf{z}(\theta)$ in (3.2.27), where the vectors $\mathbf{u} \in \mathcal{U}^\nu$ are not necessarily orthonormal. If the estimate $\hat{\mathbf{R}}^x$ of $\mathbf{R}^x$ is known sufficiently accurately, i.e. if $\hat{\mathbf{R}}^x \approx \mathbf{R}^x$ (which usually implies that $\hat{\mathbf{U}}^\nu \approx \mathbf{U}^\nu$), the cost function in (3.2.27) approximately equals zero for the true source DOA's:

$$J_{\text{ss}}(\theta)\big|_{\theta=\theta^j} \to 0 \qquad \forall\, 1 \le j \le S \quad \text{if} \quad \hat{\mathbf{R}}^x \to \mathbf{R}^x\,. \tag{3.2.28}$$

Hence, the source DOA's can be estimated by locating the minima or approximate zeros of (3.2.27). Obviously, a DOA estimator that is based on this principle is statistically consistent because $\hat{\mathbf{R}}^x \to \mathbf{R}^x$ when the number of samples T goes to infinity. In principle, the zero-finding problem could be solved by global optimization methods such as simulated annealing, genetic algorithms, clustering methods, and dynamic programming [115]. However, most subspace algorithms do not follow such an approach but convert the problem into another one, as is demonstrated in the next sections.

3.2.2 SPECTRAL-MUSIC

A famous algorithm that is based on the subspace principles explained in the previous sections is *Multiple Signal Classification (MUSIC)*. The acronym MUSIC was first coined by Schmidt, the developer of the algorithm [137, 138]. Independently from Schmidt, Bienvenu and Kopp invented a similar algorithm [16]. MUSIC is a generalization of the so-called *Pisarenko Harmonic Decomposition (PHD)* method developed by Pisarenko [127]. The PHD algorithm was the first algorithm for the identification of complex exponentials (harmonics) that was based on subspace decomposition principles. The developments in Section 3.2.1 represent the major part of the MUSIC method. MUSIC exploits the full noise subspace by considering the set $\mathcal{U}^\nu \triangleq \{\mathbf{u}^{S+1}, \ldots, \mathbf{u}^D\}$ in (3.2.24). If the correlation matrix $\mathbf{R}^x$ is known exactly, this comes down to solving the nonlinear system in (3.2.25) (recall that the solution is unique for the chosen set $\mathcal{U}^\nu$). However, if only an approximation $\hat{\mathbf{R}}^x$ is available, the minima of cost function (3.2.27) have to be found. For $\mathcal{U}^\nu \triangleq \{\mathbf{u}^{S+1}, \ldots, \mathbf{u}^D\}$, (3.2.27) becomes:

$$J_{\text{ss}}(\theta) \triangleq \sum_{i=S+1}^{D} \left| \langle \mathbf{u}^i, \mathbf{z}(\theta) \rangle \right|^2 = \mathbf{z}(\theta)^H \left(\sum_{i=S+1}^{D} \mathbf{u}^i (\mathbf{u}^i)^H \right) \mathbf{z}(\theta) = \mathbf{z}(\theta)^H \hat{\mathbf{U}}^\nu (\hat{\mathbf{U}}^\nu)^H \mathbf{z}(\theta)$$
$$\overset{(3.2.17)}{=} \mathbf{z}(\theta)^H \hat{\mathbf{P}}^\nu \mathbf{z}(\theta) \overset{(3.2.18)}{=} \mathbf{z}(\theta)^H (\mathbf{I} - \hat{\mathbf{P}}^s) \mathbf{z}(\theta) \quad \forall\, \theta \in \Theta\,. \tag{3.2.29}$$

Note that due to the orthonormality of the vectors $\mathbf{u} \in \mathcal{U}^\nu$ the frame operator $\mathbf{F}(\mathcal{U}^\nu)$ in (3.2.27) reduces to the orthogonal projection operator $\mathbf{P}^\nu$; see also the remarks just below (3.2.27). The result in (3.2.29) can be interpreted as follows (for convenience, we 'speak' in terms of ideal quantities, e.g. $\mathbf{P}^\nu$ instead of $\hat{\mathbf{P}}^\nu$). The equality $J_{\text{ss}}(\theta) = \mathbf{z}(\theta)^H \mathbf{P}^\nu \mathbf{z}(\theta) = \langle \mathbf{z}(\theta), \mathbf{P}^\nu \mathbf{z}(\theta) \rangle$ shows that $J_{\text{ss}}(\theta)$ is defined as the inner product between the array response vector $\mathbf{z}(\theta)$ and its projection onto the noise subspace. Ideally, this inner product between $\mathbf{z}(\theta)$ and its component that lies in the noise subspace is zero only for the true source DOA's; this is what is expected and desired. Due to errors in the estimated sensor statistics, this holds only approximately in the estimated case (or not at all, in extreme cases). Now we return to the specificities of MUSIC. Instead of searching for the minima of $J_{\text{ss}}(\theta)$, the idea behind the 'spectral version' of MUSIC is to find the maxima of the so-called (spatial) *pseudo-spectrum* defined as follows:

$$P_{\text{SM}}(\theta) \triangleq \frac{\|\mathbf{z}(\theta)\|^2}{J_{\text{ss}}(\theta)} = \frac{\|\mathbf{z}(\theta)\|^2}{\sum_{i=S+1}^{D} \left| \langle \mathbf{u}^i, \mathbf{z}(\theta) \rangle \right|^2} = \frac{\|\mathbf{z}(\theta)\|^2}{\mathbf{z}(\theta)^H \hat{\mathbf{U}}^\nu (\hat{\mathbf{U}}^\nu)^H \mathbf{z}(\theta)}$$
$$= \frac{D}{\mathbf{z}(\theta)^H \hat{\mathbf{P}}^\nu \mathbf{z}(\theta)} = \frac{D}{\mathbf{z}(\theta)^H (\mathbf{I} - \hat{\mathbf{P}}^s) \mathbf{z}(\theta)} \quad \forall\, \theta \in \Theta\,. \tag{3.2.30}$$

For obvious reasons, we call this spatial spectrum *'SPECTRAL-MUSIC pseudo-spectrum'*, and the algorithm that computes it *SPECTRAL-MUSIC*. In the next section, we will see that for a ULA there exists another version of MUSIC that is called *ROOT-MUSIC* and provides a means of directly computing the estimated source DOA's, i.e. without computing a pseudo-spectrum and locating its maxima. The nominator $\|\mathbf{z}(\theta)\|^2$ of $P_{\text{SM}}(\theta)$ just serves as a normalization: from (3.1.7) it follows directly that $\|\mathbf{z}(\theta)\|^2 = \mathbf{z}(\theta)^H \mathbf{z}(\theta) = D$. Equations (3.2.27) and (3.2.28) imply that the denominator of $P_{\text{SM}}(\theta)$ is approximately zero at the true source DOA's if $\hat{\mathbf{R}}^x \approx \mathbf{R}^x$. Hence, in this case the pseudo-spectrum exhibits sharp peaks in the vicinity of the source DOA's $\{\theta^j\}^{1 \le j \le S}$; if $\mathbf{R}^x$ is known exactly, theoretically the peaks are infinitely high. Consequently, the source DOA's can be estimated by locating the S highest and/or sharpest peaks of the pseudo-spectrum. This also follows immediately by observing

Algorithm 3.1 SPECTRAL-MUSIC.

1: Compute/estimate lag-zero sensor correlation matrix $\mathbf{R}^x$*;*

2: Compute EVD of $\mathbf{R}^x$ *and split the result into signal and noise subspace parts:* $\mathbf{R}^x = \mathbf{U}^x\mathbf{\Lambda}^x(\mathbf{U}^x)^H = \mathbf{U}^s\mathbf{\Lambda}^s(\mathbf{U}^s)^H + \mathbf{U}^\nu\mathbf{\Lambda}^\nu(\mathbf{U}^\nu)^H$*;*

3: Compute matrix for projection onto noise subspace: $\mathbf{P}^\nu \triangleq \mathbf{U}^\nu(\mathbf{U}^\nu)^H$*;*

4: Evaluate SPECTRAL-MUSIC pseudo-spectrum on fine grid in θ*-domain:*

$$P_{\text{SM}}(\theta) = \frac{D}{\mathbf{z}(\theta)^H\mathbf{P}^\nu\mathbf{z}(\theta)} = \frac{D}{\mathbf{z}(\theta)^H(\mathbf{I}-\mathbf{P}^s)\mathbf{z}(\theta)} \qquad \forall\,\theta\in\Theta\,;$$

5: Determine source DOA's by locating the S *sharpest peaks of* $P_{\text{SM}}(\theta)$*, e.g. by grid search method.*

that locating the minima of $J_{\text{ss}}(\theta)$ is equivalent to locating the maxima of $1/J_{\text{ss}}(\theta)$. In fact, both problems are equally difficult or easy. The term 'pseudo-spectrum' emphasizes the fact that (3.2.30) is not a true spectrum because it does not contain information about the power of the signals and/or the variance of the noise background; it only serves to locate the source DOA's. Most existing methods for computing the maxima of a pseudo-spectrum rely on exhaustive grid search. However, practice has shown that grid search is often problematic, time consuming and prone to errors. An alternative is the so-called roller-coaster algorithm [159] or one of the global optimization methods mentioned at the end of the previous section. For the moment, we are mainly interested in the way the cost function in (3.2.27) and the pseudo-spectrum in (3.2.30) are obtained, and in the properties of these functions. Therefore, we will not elaborate now on finding the extrema of these functions, but provide more insight into SPECTRAL-MUSIC by giving some examples. The different steps of the SPECTRAL-MUSIC algorithm are summarized in Alg. 3.1. For convenience the algorithm is formulated in terms of ideal statistics, but it is completely the same for estimated statistics.

Examples and discussion

In order to demonstrate how SPECTRAL-MUSIC works and to highlight some of its most important properties, we now give some simulated examples for a Uniform Linear Array (ULA) with sensor spacing $d = \frac{\lambda_c}{2}$ (see Section 3.1.2). From (3.1.13), it follows that the array response vector $\mathbf{z}(\theta)$ for this case is given by:

$$\mathbf{z}(\theta) = \begin{bmatrix} z_1(\theta) \\ z_2(\theta) \\ \vdots \\ z_D(\theta) \end{bmatrix} = \begin{bmatrix} 1 \\ \exp\big(\jmath\,\pi\sin(\theta)\big) \\ \vdots \\ \exp\big(\jmath\,(D-1)\pi\sin(\theta)\big) \end{bmatrix} \qquad \forall\,\theta\in\Theta\,. \tag{3.2.31}$$

Two different scenarios are considered for the same number of sources $S = 3$, viz. one with $D = 4$ sensors and one with $D = 5$ sensors. For each scenario, two pseudo-spectra are computed: one using the ideal sensor correlation matrix $\mathbf{R}^x$, and the other using its estimate $\hat{\mathbf{R}}^x$ obtained by using $T = 900$ samples of $\mathbf{x}$ that are generated according to the model in (3.1.1). The DOA's of the three sources are given by $\tilde{\boldsymbol{\theta}} = \big[\theta^1 \;\; \theta^2 \;\; \theta^3\big] = [-60, 10, 17\,]$ degrees. The signals and system are generated in such a way that the assumptions listed in Section 3.2.1

are satisfied. All source and noise signals are white *complex circular Gaussian* [94, 147] signals of unit variance, i.e. $E\{\mathbf{s}[m]\mathbf{s}[n]^H\} = E\{\boldsymbol{\nu}[m]\boldsymbol{\nu}[n]^H\} = \delta_{mn}\mathbf{I}$ and $E\{\mathbf{s}[m]\mathbf{s}[n]^T\} = E\{\boldsymbol{\nu}[m]\boldsymbol{\nu}[n]^T\} = \mathbf{0}$, where δ_{mn} denotes a Kronecker delta function (i.e. δ_{mn} is 1 iff $m = n$ and 0 elsewhere; see Section A.4). For example, a complex circular Gaussian source signal $s[n] = s_{\mathrm{r}}[n] + \jmath s_{\mathrm{i}}[n]$ can easily be constructed by adding two statistically independent stationary white Gaussian sequences $s_{\mathrm{r}}[n]$ and $s_{\mathrm{i}}[n]$, which represent the real and imaginary parts of $s[n]$ respectively, with variance $\frac{1}{2}$ (the variance of a complex signal $v = v_{\mathrm{r}} + \jmath v_{\mathrm{i}}$ is defined as $(\sigma^v)^2 \triangleq E\{v\,(v)^*\}$; see Appendix B). Note that the source and the noise signals have the same variance, hence the 'input SNR' equals 0 dB. The pseudo-spectra are computed by means of Alg. 3.1. First, $\mathbf{R}^x$ $\left(\hat{\mathbf{R}}^x\right)$ is computed (estimated). Then the EVD of this matrix is computed, e.g. using the eig or $\mathrm{Eigensystem}$ functions of Matlab and Mathematica respectively. Subsequently, the $D-S$ eigenvectors corresponding to the $D-S$ smallest eigenvalues are used to construct $\mathbf{U}^\nu$ and the noise projection matrix $\mathbf{P}^\nu$ according to (3.2.17). Finally, the pseudo-spectrum is obtained by evaluating (3.2.30) on a fine grid in the θ-domain for $\theta \in \Theta$, where $\Theta \triangleq [-\pi/2, \pi/2]$ (in radians). The resulting spectra expressed in decibels [dB] are plotted as a function of the DOA θ expressed in degrees in Figures 3.2 and 3.3 on the following page for $D = 4$ and $D = 5$ respectively. The ideal spectra obtained from $\mathbf{R}^x$ are represented by solid lines, whereas the spectra obtained from $\hat{\mathbf{R}}^x$ are represented by dotted lines. The actual source DOA's are indicated by dashed vertical lines.

In each of the figures, the ideal spectrum exhibits very sharp peaks at the source DOA's. Moreover, the resolution is sufficiently high to resolve the sources at θ_2 and θ_3, which differ by 7 degrees only. Also, no *spurious peaks*, i.e. peaks that do not correspond to true source DOA's, are present. All these properties are very important and desirable for any pseudo-spectrum. Therefore, we summarize them here for later use.

Desired properties of MUSIC pseudo-spectrum:

- *DPS1: Sharp peaks are exhibited at the source DOA's;*
- *DPS2: High spectral resolution that allows to resolve closely spaced sources;*
- *DPS3: No spurious peaks are present.*

These properties imply that the source DOA's can be determined uniquely from the ideal pseudo-spectra. The dotted lines in Figures 3.2 and 3.3 show that the pseudo-spectra computed from $\hat{\mathbf{R}}^x$ resemble the ideal ones computed from $\mathbf{R}^x$, but may differ significantly in the neighborhood of the source DOA's. This is mainly due to two closely related reasons, both of which have to do with the inexact estimation of $\mathbf{R}^x$.

Firstly, in practice the number of samples T used for estimating $\mathbf{R}^x$ plays an important role in determining the shape of the spectrum (see also Section 3.2.1). Due to the limited amount of data available for the computation of $\hat{\mathbf{R}}^x$, the noise eigenvectors deviate from their ideal values. It is well-known that eigenvalues and eigenvectors are quite sensitive to small perturbations of a matrix [72]. Since estimated noise eigenvectors are used for the construction of the matrix $\hat{\mathbf{P}}^\nu$ in the denominator $\mathbf{z}(\theta)^H\hat{\mathbf{P}}^\nu\mathbf{z}(\theta)$ of $P_{\mathrm{SM}}(\theta)$, this denominator also deviates from the ideal denominator $\mathbf{z}(\theta)^H\mathbf{P}^\nu\mathbf{z}(\theta)$. This has a number of consequences. First of all, the locations of the peaks may differ from the true source DOA's. For example, this is clearly demonstrated in Fig. 3.2 for θ^3. In addition, as can be seen clearly in Fig. 3.3, the peaks are smeared out, i.e. smoother, wider and lower, compared to the ideal case. Obviously, this is an undesired effect because it makes it more difficult to locate the maxima accurately and to resolve closely spaced sources. Finally, spurious peaks may be introduced. This problem

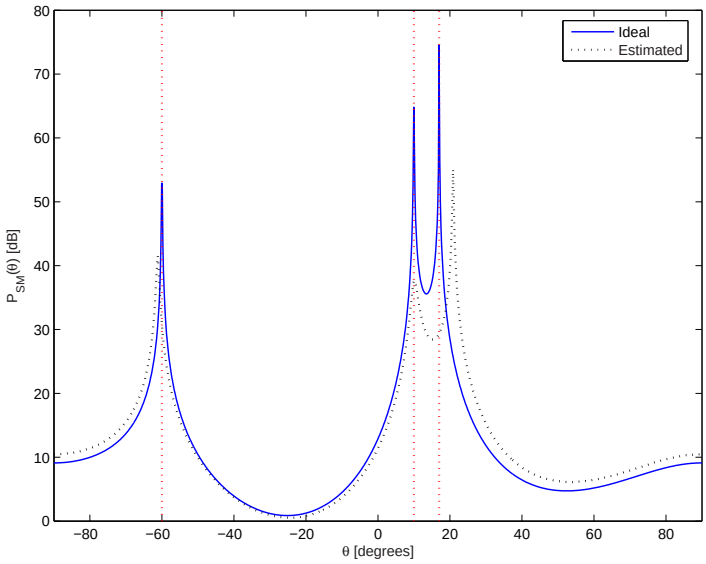

Figure 3.2: *Ideal and estimated MUSIC spectra $P_{\mathrm{SM}}(\theta)$ for $D = 4$ and $S = 3$.*

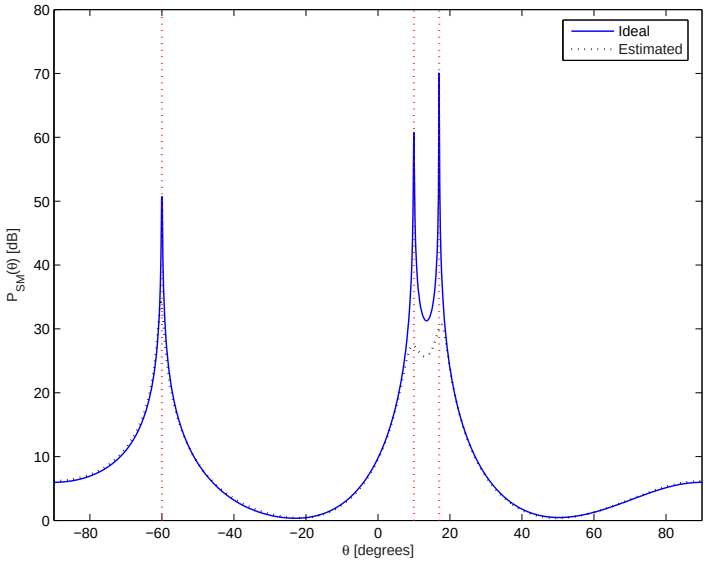

Figure 3.3: *Ideal and estimated MUSIC spectra $P_{\mathrm{SM}}(\theta)$ for $D = 5$ and $S = 3$.*

does not occur in the current example, but may arise for different scenarios and a different number of samples. These three problems may render the exact and unique determination of the true source DOA's difficult or even impossible. For reference and clarity we summarize them in the following list.

Possible influences on MUSIC pseudo-spectrum $P_{\text{SM}}(\theta)$ of using limited amount of data for estimating sensor correlation matrix $\hat{\mathbf{R}}^x$:

PIF1: Locations of peaks differ from true source DOA's;

PIF2: Peaks are smeared out;

PIF3: Spurious peaks are introduced.

Obviously, since the estimation of $\mathbf{R}^x$ is statistically consistent, the estimated spectra converge to the ideal ones if the number of samples T goes to infinity, and problems *PIF1-PIF3* disappear. In other words, the pseudo-spectrum is statistically consistent.

Secondly, in addition to the number of samples T, the number of sensors D in relation to the number of sources S plays an important role in determining the shape of the spectrum, in the ideal as well as in the estimated case. Comparing Figures 3.2 and 3.3, we see that the minima of the ideal pseudo-spectrum in Fig. 3.3, which exploits two noise eigenvectors, are deeper than that of the ideal spectrum in Fig. 3.2, which exploits only one noise eigenvector[‡]. This can be seen, for example, by comparing the values in both figures at -90, 50 and 90 degrees, and the minimum between the second and third peaks. Since this is a desired property, it demonstrates that using several noise eigenvectors may be advantageous. Now, we consider the non-ideal case. Suppose that the number of samples T is fixed and finite, i.e. only an estimate $\hat{\mathbf{R}}^x$ of the sensor correlation matrix is available. From (3.2.29) and (3.2.30), it is clear that the denominator $\sum_{i=S+1}^{D} \left| \langle \mathbf{u}^i, \mathbf{z}(\theta) \rangle \right|^2$ of $P_{\text{SM}}(\theta)$ is the (scaled) average of the squared absolute values of the inner products between all $D-S$ noise eigenvectors and the array response vector $\mathbf{z}(\theta)$. As stated above, eigenvalues and eigenvectors are quite sensitive to small perturbations of a matrix. In fact, in practice each of the estimated noise eigenvectors usually introduces spurious peaks if T is 'small'. Since the spurious peaks introduced by different noise eigenvectors tend to occur at different DOA's, it is advantageous to use several noise eigenvectors because then the effects of the different eigenvectors on the spectrum are averaged out. Moreover, the locations of the peaks corresponding to the source DOA's are closer to their true values because the deviations are averaged out as well. Hence, the more noise eigenvectors are used, i.e. the larger the number of sensors D in comparison with the number of sources S, the less sensitive the pseudo-spectrum is to errors in the estimation of $\mathbf{R}^x$. Indeed, by comparing Figures 3.2 and 3.3 we see that the estimated pseudo-spectrum in Fig. 3.3, which exploits two noise eigenvectors, is much closer to its corresponding ideal spectrum for almost all $\theta \in \Theta$ than the one in Fig. 3.2, which exploits only one noise eigenvector. Moreover, we see that the locations of the peaks in Fig. 3.3 coincide with the true source DOA's, whereas those in Fig. 3.2 do not. On the other hand, due to the averaging the peaks in Fig. 3.3 are less sharp, i.e. lower and wider, than those in Fig. 3.2. Hence, the averaging also results in a loss of resolution of the MUSIC pseudo-spectrum; consequently, a compromise has to be made.

Statistical performance analyses of the MUSIC algorithm can be found in [93, 148, 149] and [104]. We note that although we have considered examples for a ULA only, the SPECTRAL-MUSIC method can equally well be applied to arrays with an arbitrary geometry. The next section discusses the 'ROOT-version' of MUSIC and provides additional insight into the properties of SPECTRAL-MUSIC.

[‡]If the noise subspace is one-dimensional, MUSIC reduces to the well-known Pisarenko Harmonic Decomposition (PHD) algorithm [127].

3.2.3 ROOT-MUSIC

For a Uniform Linear Array (ULA), another common approach for determining the source DOA's is followed by the so-called *ROOT-MUSIC* algorithm [10]. To a large extent this algorithm is the same as SPECTRAL-MUSIC. However, instead of determining the source DOA's by searching a pseudo-spectrum for its maxima, the ROOT-MUSIC algorithm is based on finding the roots of a polynomial that is closely related to the denominator $J_{\text{ss}}(\theta)$ of $P_{\text{SM}}(\theta)$ in (3.2.30). That is, ROOT-MUSIC first computes the roots of this polynomial and then derives the source DOA's from these roots. As a starting point, we consider the ideal version of the function $J_{\text{ss}}(\theta)$ in (3.2.29):

$$\begin{aligned} J_{\text{ss}}(\theta) &= \mathbf{z}(\theta)^H \mathbf{P}^\nu \mathbf{z}(\theta) = \mathbf{z}(\theta)^H \mathbf{U}^\nu (\mathbf{U}^\nu)^H \mathbf{z}(\theta) = \left|(\mathbf{U}^\nu)^H \mathbf{z}(\theta)\right|^2 \\ &= \sum_{i=S+1}^{D} \left|\left\langle \mathbf{u}^i, \mathbf{z}(\theta)\right\rangle\right|^2 = \sum_{i=S+1}^{D} \Big(\mathbf{z}(\theta)^H \mathbf{u}^i\Big)\Big(\mathbf{z}(\theta)^H \mathbf{u}^i\Big)^*. \end{aligned} \tag{3.2.32}$$

Now consider a term of the form $\mathbf{z}(\theta)^H \mathbf{u}^i$, i.e. the complex conjugate of the inner product $\left\langle \mathbf{u}^i, \mathbf{z}(\theta)\right\rangle$. From (3.1.12) and (3.1.13) it follows that this term can be written as follows (note that u_k^i is the k-th element of the column vector $\mathbf{u}^i$):

$$\mathbf{z}(\theta)^H \mathbf{u}^i = \sum_{k=1}^{D} \big(z_k(\theta)\big)^* u_k^i \overset{(3.1.12)}{=} \sum_{k=1}^{D} (e)^{-\jmath\,(k-1)\xi(\theta)} u_k^i = \sum_{k=0}^{D-1} (e)^{-\jmath\, k\xi(\theta)} u_{k+1}^i$$

(recall that $(a)^b$ means a to the power b; see Appendix A). Hence, defining:

$$U^i\big(z\big) \triangleq \sum_{k=0}^{D-1} u_{k+1}^i \,(z)^{-k}\,, \tag{3.2.33}$$

the term $\mathbf{z}(\theta)^H \mathbf{u}^i$ can be written as follows:

$$\mathbf{z}(\theta)^H \mathbf{u}^i = U^i\big(z\big)\big|_{z=(e)^{\jmath\xi(\theta)}} = U^i\big((e)^{\jmath\xi(\theta)}\big)\,. \tag{3.2.34}$$

Note that $U^i\big(z\big)$ is the z-transform of the sequence $u_1^i, \ldots, u_D^i$, i.e. of the i-th eigenvector $\mathbf{u}^i$ of $\mathbf{R}^x$. For this reason, a function like $U^i\big(z\big)$ is often called *eigenfilter* [115, 154]. Using (3.2.33) and (3.2.34), (3.2.32) can now be written as:

$$J_{\text{ss}}(\theta) = \sum_{i=S+1}^{D} U^i\big((e)^{\jmath\xi(\theta)}\big)\Big(U^i\big((e)^{\jmath\xi(\theta)}\big)\Big)^* = \sum_{i=S+1}^{D} U^i\big(z\big)\Big(U^i\big(1/z^*\big)\Big)^*\Big|_{z=(e)^{\jmath\xi(\theta)}}, \tag{3.2.35}$$

where the latter equality holds because $z = \frac{1}{z^*}$ for z on the unit circle. With abuse of notation we write:

$$J_{\text{ss}}(z) = \sum_{i=S+1}^{D} U^i\big(z\big)\Big(U^i\big(1/z^*\big)\Big)^*. \tag{3.2.36}$$

As we will see soon, the source DOA's can be determined from the phases of the roots of $J_{\text{ss}}(z)$ that lie on the unit circle. From (3.2.33), it follows that $\big(U^i(1/z^*)\big)^* = \sum_{k=0}^{D-1} (z)^k \left(u_{k+1}^i\right)^*$. This implies that $J_{\text{ss}}(z)$ only contains terms with negative and positive powers of z, and no terms with powers of z^*. It is this property of $J_{\text{ss}}(z)$ that makes it easy to compute its roots. Since the powers of z in $J_{\text{ss}}(z)$ range from $-(D-1)$ till $(D-1)$,

multiplying $J_{\text{ss}}(z)$ by $(z)^{D-1}$ yields the following *'ROOT-MUSIC polynomial'* $p_{\text{RM}}(z)$ of degree $2(D-1)$:

$$p_{\text{RM}}(z) \triangleq (z)^{D-1} J_{\text{ss}}(z) = (z)^{D-1} \sum_{i=S+1}^{D} U^i(z) \Big(U^i\big(1/z^*\big) \Big)^* . \tag{3.2.37}$$

As we have just indicated, the reason for writing $J_{\text{ss}}(z) = \sum_{i=S+1}^{D} U^i(z) \big(U^i(1/z^*) \big)^*$ instead of $J_{\text{ss}}(z) = \sum_{i=S+1}^{D} U^i(z) \big(U^i(z) \big)^*$ is that it is easy to compute the roots of a polynomial, but difficult (in general) to compute the roots of a 'polynomial-like' expression involving terms with powers of both z and z^*.

As is clear from the developments in Section 3.2.1, and in particular equation (3.2.28), $p_{\text{RM}}(z)$ is zero at the locations $z^j = (e)^{\jmath\xi(\theta^j)}$ $\forall\, 1 \leq j \leq S$ in the z-plane that correspond to the source DOA's:

$$p_{\text{RM}}(z)\big|_{z=(e)^{\jmath\xi(\theta^j)}} = p_{\text{RM}}\big((e)^{\jmath\xi(\theta^j)}\big) = 0 \qquad \forall\, 1 \leq j \leq S\,. \tag{3.2.38}$$

For obvious reasons, we will call the roots $\{z^j \triangleq (e)^{\jmath\xi(\theta^j)}\}^{1\leq j\leq S}$ *source roots*. Since the function $J_{\text{ss}}(z)$ in (3.2.36) is self-reciprocal, i.e. $J_{\text{ss}}(z) = \big(J_{\text{ss}}(1/z^*)\big)^*$, its roots appear in conjugate reciprocal pairs $(z, 1/z^*)$. This implies that each root on the unit circle is double because $1/z^* = z$ for $z = (e)^{\jmath\beta}$ and $\beta \in \mathbb{R}$. Hence, all source roots are double roots. Since $p_{\text{RM}}(z)$ is of degree $2(D-1)$, it has $2(D-1)$ roots, $2S$ of which are the double source roots. Consequently, there are $2(D-S-1)$ other roots that in general do not lie on the unit circle. These roots are called *spurious roots* or *noise roots*. There are essentially two different kinds of spurious roots. Firstly, spurious roots are always present if $D - S > 1$, i.e. if the dimension of the noise subspace is larger than one. If they lie close to the unit circle such noise roots may give rise to spurious peaks in a SPECTRAL-MUSIC pseudo-spectrum, or may cause confusion between source and noise roots in ROOT-MUSIC. Especially if an estimated correlation matrix $\hat{\mathbf{R}}^x$ is used, the noise roots may move close to the unit circle, thereby causing this kind of problem. If all source roots are closer to the unit circle than the noise roots, the latter play no particular role in determining the source DOA's for the ROOT-MUSIC algorithm. Secondly, spurious roots close to or on the unit circle are present if the array is ambiguous. For example, for a ULA this happens if the sensor spacing is larger than half the carrier wavelength, i.e. if the spatial equivalent of Shannon's sampling theorem is not satisfied (see Section 3.1.2). Obviously, this kind of spurious roots always causes confusion of source and noise roots. Therefore, the array should always be designed in such a way that it is unambiguous. The roots of $p_{\text{RM}}(z)$ in (3.2.37), and thus of $J_{\text{ss}}(z)$, can easily be obtained by any polynomial rooting method, e.g. the roots or Solve functions of Matlab and Mathematica respectively. Once these roots have been obtained, in the ideal case the S unique (double) roots that lie on the unit circle are selected, whereas in the estimated case the ones closest to the unit circle are selected. From these roots, the source DOA's can easily be recovered by inverting the function $z(\theta) = \exp(\jmath\xi(\theta))$, where the 'coordinate transformation' $\xi(\theta)$ is given by (3.1.10). Hence, supposing that z is a source root of $p_{\text{RM}}(z)$, the corresponding DOA θ can be computed as follows:

$$\theta = \arcsin\Big(\tfrac{\lambda_{\text{c}}}{2\pi d} \arg(z) \Big)\,. \tag{3.2.39}$$

The different steps of the ROOT-MUSIC algorithm are summarized in Alg. 3.2. It is formulated in terms of ideal statistics, but it is completely the same for estimated statistics. Although typically noise eigenvectors are employed in step 3 of the algorithm, in principle any set $\mathcal{U}^\nu$ (see (3.2.24)) of noise subspace vectors can be chosen.

Algorithm 3.2 ROOT-MUSIC.

1-2: See steps 1 and 2 of Alg. 3.1;

3: Use noise eigenvectors $\mathbf{u}^{S+1}, \ldots, \mathbf{u}^{D}$ *in matrix* $\mathbf{U}^{\nu}$ *to construct eigenfilters for* $S+1 \leq i \leq D$*:*

$$U^i(z) \triangleq \sum_{k=0}^{D-1} u_{k+1}^i (z)^{-k} ;$$

4: Construct ROOT-MUSIC polynomial:

$$p_{\text{RM}}(z) = (z)^{D-1} \sum_{i=S+1}^{D} U^i(z) \Big(U^i\big(1/z^*\big) \Big)^* ;$$

5: Compute roots of $p_{\text{RM}}(z)$*;*

6: Select S *unique roots of* $p_{\text{RM}}(z)$ *that lie on or closest to unit circle;*

7: For each source root z*, compute corresponding source DOA:*

$$\theta = \arcsin\Big(\tfrac{\lambda_c}{2\pi d} \arg(z) \Big) .$$

Examples and discussion

In order to demonstrate how ROOT-MUSIC works and to highlight some of its most important properties, we apply the ROOT-MUSIC method described in Alg. 3.2 to the examples presented in Section 3.2.2. Figures 3.4 and 3.5 show the locations in the z-plane of the roots of the ROOT-MUSIC polynomial $p_{\text{RM}}(z)$ obtained in the fifth step of the algorithm for the examples with $D = 4$ and $D = 5$ respectively. The diamonds ($\diamond$) indicate the roots for the ideal case, i.e. if $\mathbf{R}^x$ is known exactly, whereas the crosses ($\times$) indicate the roots for the estimated case, i.e. if only an estimate $\hat{\mathbf{R}}^x$ is available. The horizontal axis represents the real part of z, whereas the vertical axis represents the imaginary part of z. For reference, also the unit circle is depicted. The rays emanating from the origin represent the phases $\xi(\theta^1), \xi(\theta^2)$ and $\xi(\theta^3)$ corresponding to the actual source DOA's θ^1, θ^2 and θ^3 respectively via relation (3.1.10).

Firstly, we consider Fig. 3.4, which depicts the roots of $p_{\text{RM}}(z)$ for the ideal and estimated cases with $D = 4$ sensors. In the ideal case, there are $S = 3$ double roots that lie exactly on the unit circle; these source roots correspond to the true source DOA's $\theta^1 = -60$, $\theta^2 = 10$ and $\theta^3 = 17$ degrees. The corresponding phases are given by $\xi(-60) = -155.9$, $\xi(10) = 31.3$ and $\xi(17) = 52.6$ degrees. Since $2(D - S - 1) = 2(4 - 3 - 1) = 0$, there are no spurious/noise roots. For the estimated case, the source roots produced by the fifth step of Alg. 3.2 are still close to the unit circle but deviate from their ideal positions. The estimated DOA's corresponding to these roots are $\hat{\theta}^1 = -61.2$, $\hat{\theta}^2 = 10.1$ and $\hat{\theta}^3 = 20.9$ degrees; obviously, these values deviate from the true source DOA's. The phases corresponding to the estimated DOA's are given by $\xi(-61.2) = -157.7$, $\xi(10.1) = 31.5$ and $\xi(20.9) = 64.1$ degrees respectively. Hence, the first two estimated phases and DOA's are reasonably good estimates, whereas the phase and DOA corresponding tot the third source deviate the most from their ideal values. This is also evident from the SPECTRAL-MUSIC pseudo-spectrum in Fig. 3.2 (recall that the roots of the denominator in the SPECTRAL-MUSIC pseudo-spectrum definition equal those of the ROOT-MUSIC polynomial). The causes of these deviations have been discussed in Section 3.2.2.

Secondly, we consider Fig. 3.5, which depicts the roots of $p_{\text{RM}}(z)$ for the ideal and estimated cases with $D = 5$ sensors. The left part of the figure provides an overview of the locations of all roots of $p_{\text{RM}}(z)$, whereas the right part zooms in on the area containing the unit circle. In the ideal case there are $S = 3$ double source roots that lie exactly on the unit circle and correspond to the true source DOA's. Since $2(D - S - 1) = 2(5 - 3 - 1) = 2$, now there are two spurious/noise roots. These are given by $0.0139 - 4.0018\jmath$ (see left part of Fig. 3.5) and $0.0009 - 0.2499\jmath$ (see both parts of the figure). As has been stated above, in the ideal case these roots play no particular role in determining the source DOA's. However, in the estimated case for some scenarios they might move so close to the unit circle that they might be confused with source roots, but this does not happen in our example. As can be seen in Fig. 3.5, the estimated roots for $D = 5$ deviate from their ideal positions. The estimated DOA's corresponding to the three source roots are $\hat{\theta}^1 = -60.3$, $\hat{\theta}^2 = 9.1$ and $\hat{\theta}^3 = 17.8$ degrees, which are reasonably good estimates. The corresponding phases are given by $\xi(-60.3) = -156.3$, $\xi(9.1) = 28.5$ and $\xi(17.8) = 55.1$ degrees respectively.

Finally, we compare the examples corresponding to Figures 3.4 and 3.5; in addition, we relate them to the examples corresponding to Figures 3.2 and 3.3. From the numerical values of the estimated source DOA's, we can conclude that the performance of ROOT-MUSIC with five sensors is better than with four (on average). However, we cannot say in general that using more sensors is always better because this depends on several other things, such as the signal scenario, the number of samples T used for estimating $\mathbf{R}^x$, etc. Now, we provide some important points that should be taken into account when comparing performances for different numbers of sensors. Firstly, by comparing Figures 3.4 and 3.5 and recalling the theory above, we see that in contrast to the case $D > S + 1$, for $D = S + 1$ there are no spurious/noise roots (this is the 'Pisarenko case', where the number of sensors is one larger than the number of sources; see the footnote on page 101). Secondly, by comparing the estimated cases in Figures 3.4 and 3.5, we see that there are different kinds of deviations of the roots of the respective ROOT-MUSIC polynomials from their ideal values. On the one hand, for $D = 4$ the deviation of the roots from their ideal values on average is larger in the angular direction than for $D = 5$. On the other hand, for $D = 5$ the deviation of the roots from their ideal values on average is larger in the radial direction than for $D = 4$. This results in smoothing. Since our interest is in the angles of the roots rather than in their radii, we can conclude that in this particular example the performance of ROOT-MUSIC is better for $D = 5$ than for $D = 4$. Since ROOT-MUSIC is based on the denominator of the MUSIC pseudo-spectrum, each kind of deviation has a corresponding deviation in the pseudo-spectrum (several reasons for deviation of an estimated pseudo-spectrum from the ideal one have been discussed in Section 3.2.2). Firstly, a deviation of a ROOT-MUSIC source root in the angular direction corresponds to a deviation of a peak location from its ideal position in the MUSIC pseudo-spectrum. Hence, as in the ROOT-MUSIC case, for $D = 4$ the deviation of the estimated peak locations in the MUSIC pseudo-spectrum on average is larger than for $D = 5$; compare Fig. 3.2 to Fig. 3.3. Secondly, a deviation of a ROOT-MUSIC source root in the radial direction corresponds to the smoothing of a peak in the MUSIC pseudo-spectrum.

Statistical performance analyses of the ROOT-MUSIC algorithm and related issues can be found in [93, 104, 131, 148, 149]. Finally, note that in many cases ROOT-MUSIC is more efficient compared to SPECTRAL-MUSIC because a rooting method is more efficient than the computation of a pseudo-spectrum on a very fine grid followed by a grid search.

3.2.4 The Minimum-Norm subspace method

For Uniform Linear Arrays, another subspace based method called *Minimum-Norm method*, or *Min-Norm method*, has been developed [98, 99, 147, 179]. Similarly to MUSIC, we can distinguish two 'versions' of the Min-Norm method, viz. *Spectral-Min-Norm* and *Root-Min-Norm*. Like SPECTRAL-MUSIC and ROOT-MUSIC, these methods are based on the construction of a pseudo-spectrum and a polynomial respectively. As has been explained in the previous sections, SPECTRAL-MUSIC and ROOT-MUSIC achieve some degree of robustness against estimation errors in the sensor correlation matrix by using multiple noise eigenvectors and performing a kind of averaging. Instead of doing this, the Minimum-Norm method tries to achieve robustness by using a single appropriately chosen vector $\mathbf{u} \triangleq [u_1, \ldots, u_D]^T$, which is constrained to lie in the noise subspace. Hence, in contrast to MUSIC, the Min-Norm method does not exploit the full noise subspace, but uses the set $\mathcal{U}^\nu \triangleq \{\mathbf{u}\}$ in (3.2.24) with cardinality one for some appropriately chosen noise subspace vector $\mathbf{u} \in \mathcal{R}_{\mathrm{c}}(\mathbf{U}^\nu)$. Before discussing how this vector is chosen, we show how a pseudo-spectrum and a polynomial in the same spirit as SPECTRAL-MUSIC and ROOT-MUSIC respectively can be defined from $\mathbf{u}$.

Firstly, similarly to SPECTRAL-MUSIC (3.2.30) the vector $\mathbf{u}$ can be used to define the *'Spectral-Min-Norm pseudo-spectrum'*:

$$P_{\mathrm{SMN}}(\theta) \triangleq \frac{\|\mathbf{z}(\theta)\|^2}{|\langle \mathbf{u}, \mathbf{z}(\theta)\rangle|^2} = \frac{D}{|\mathbf{z}(\theta)^H \mathbf{u}|^2} = \frac{D}{|\mathbf{z}(\theta)^H \mathbf{u}|^2} \qquad \forall\, \theta \in \Theta\,, \tag{3.2.40}$$

where $\mathbf{z}(\theta)$ is given by (3.1.13) and the nominator serves as a normalization. As with SPECTRAL-MUSIC, the idea is to estimate the source DOA's by locating the S highest and/or sharpest peaks of this pseudo-spectrum. For obvious reasons, we will call this version of the Min-Norm algorithm *Spectral-Min-Norm*.

Secondly, similarly to ROOT-MUSIC the vector $\mathbf{u}$ can be used to define a *'ROOT-Min-Norm' polynomial* $p_{\mathrm{RMN}}(z)$ as follows:

$$p_{\mathrm{RMN}}(z) \triangleq U\big(z\big) \triangleq \sum_{k=0}^{D-1} u_{k+1}\,(z)^{-k}\,, \tag{3.2.41}$$

where $U\big(z\big)$ is the z-transform of the sequence $u_1, \ldots, u_D$; see also (3.2.33) and (3.2.34). As with ROOT-MUSIC, the idea is to estimate the source DOA's from the S roots of $p_{\mathrm{RMN}}(z)$ that are closest to or on the unit circle (using (3.2.39)). For obvious reasons, we will call this version of the Min-Norm algorithm *Root-Min-Norm*. Note that with definition (3.2.41), the spectrum $P_{\mathrm{SMN}}(\theta)$ can be expressed as follows:

$$P_{\mathrm{SMN}}(\theta) = \frac{D}{\big|p_{\mathrm{RMN}}\big(z\big)\big|^2}\bigg|_{z=(e)^{\jmath\xi(\theta)}} = \frac{D}{\big|p_{\mathrm{RMN}}\big((e)^{\jmath\xi(\theta)}\big)\big|^2} \qquad \forall\, \theta \in \Theta\,. \tag{3.2.42}$$

Since $\mathbf{u}$ is constrained to lie in the noise subspace, i.e. $\mathbf{u} \in \mathcal{R}_{\mathrm{c}}(\mathbf{U}^\nu)$, it follows from (3.2.23) that $p_{\mathrm{RMN}}(z)$ has at least S source roots at the locations $z^j = (e)^{\jmath\xi(\theta^j)} \quad \forall\, 1 \leq j \leq S$ on the unit circle (supposing for the moment that $\mathbf{R}^x$ is known exactly):

$$p_{\mathrm{RMN}}(z)\big|_{z=(e)^{\jmath\xi(\theta^j)}} = p_{\mathrm{RMN}}\big((e)^{\jmath\xi(\theta^j)}\big) = 0 \qquad \forall\, 1 \leq j \leq S\,. \tag{3.2.43}$$

The polynomial $p_{\mathrm{RMN}}(z)$ has $D-1$ roots because it is of degree $D-1$. Hence, in addition to the S source roots, it has $D-S-1$ noise or spurious roots (see Section 3.2.3 for the

definitions of source and noise roots). Depending on the chosen vector $\mathbf{u}$ and the accuracy of the estimated sensor correlation matrix, the noise roots might be closer to the unit circle than the source roots. In such a case, it is impossible to distinguish noise and source roots (see also Section 3.2.3) and wrong or inaccurate source DOA estimates may be obtained.

The similarities to SPECTRAL- and ROOT-MUSIC are evident. The essential difference, which at the same time is the Min-Norm design problem, is the design/choice of the noise subspace vector $\mathbf{u}$ in such a way that the adverse effects of estimation errors in the sensor correlation matrix $\mathbf{R}^x$, in particular spurious roots, are attenuated. The Min-Norm method, as its name implies, tries to achieve this goal by minimizing the squared norm $\|\mathbf{u}\|^2 = \mathbf{u}^H\mathbf{u}$ of $\mathbf{u} \in \mathcal{R}_c\left(\mathbf{U}^\nu\right)$ subject to the constraint that the first component of $\mathbf{u}$ equals one [98,99,147, 154, 179]. Summarizing, the optimal vector $\mathbf{u}$ computed by the Min-Norm algorithm has to satisfy the following three constraints:

MNC1: The vector $\mathbf{u}$ *has minimum norm;*

MNC2: The vector $\mathbf{u}$ *lies in the noise subspace;*

MNC3: The first element of $\mathbf{u}$ *equals one.*

These conditions ensure respectively that the spurious roots of $p_{\text{RMN}}(z)$ lie inside the unit circle, that the S source roots of $p_{\text{RMN}}(z)$ lie on (or close to, for estimated statistics) the unit circle, and that the minimum norm solution is not the (trivial) zero vector [98, 99]. It can also be shown that the spurious roots are approximately distributed uniformly around the unit circle in the angular sectors where the source roots are not present [98]. Hence, the spurious roots tend to be well separated from the source roots, which is a very desirable property for attenuating the adverse effects of spurious roots on the estimation of the source DOA's. The problem of finding the vector $\mathbf{u}$ satisfying conditions *MNC1-MNC3* can be formulated mathematically as follows:

$$\hat{\mathbf{u}} = \underset{\mathbf{u}}{\operatorname{argmin}}\, \mathbf{u}^H\mathbf{u} \qquad \text{subject to} \qquad \mathbf{P}^\nu\mathbf{u} = \mathbf{u} \quad \text{and} \quad u_1 = 1\,,$$

where u_1 is the first element of $\mathbf{u}$. Using Lagrange multipliers [108, 115], the solution [154, 179] can be obtained and is given by:

$$\mathbf{u} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H\|^2} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\tilde{\mathbf{u}}_1^\nu(\tilde{\mathbf{u}}_1^\nu)^H} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\tilde{\mathbf{u}}_1^\nu\|^2}\,,$$

where $\tilde{\mathbf{u}}_1^\nu$ is the first row of $\mathbf{U}^\nu$. Of course, once $\mathbf{u}$ has been derived according to the criteria and procedure above, we are free to re-normalize it because this does not influence the shape of the pseudo-spectrum used in the Spectral-Min-Norm algorithm nor the location of the roots of the Root-Min-Norm polynomial. Therefore, we normalize $\mathbf{u}$ to norm one:

$$\mathbf{u} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H\|} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\tilde{\mathbf{u}}_1^\nu\|}\,. \tag{3.2.44}$$

Now, we are in a position to formulate the Spectral-Min-Norm and Root-Min-Norm algorithms. Again, the algorithms are formulated in terms of ideal statistics. The different steps of the Spectral-Min-Norm and Root-Min-Norm algorithms are summarized in Alg. 3.3 and Alg. 3.4 respectively (compare them with Alg. 3.1 and Alg. 3.2 respectively).

Algorithm 3.3 Spectral-Min-Norm.

1-2: See steps 1 and 2 of Alg. 3.1;

3: Use matrix $\mathbf{U}^\nu$ *of noise eigenvectors to construct minimum norm vector* $\mathbf{u}$*:*

$$\mathbf{u} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H\|} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\tilde{\mathbf{u}}_1^\nu\|} \; ;$$

4: Evaluate Min-Norm pseudo-spectrum on fine grid in θ*-domain:*

$$P_{\text{SMN}}(\theta) \triangleq \frac{\|\mathbf{z}(\theta)\|^2}{|\langle \mathbf{u}, \mathbf{z}(\theta)\rangle|^2} = \frac{D}{|\mathbf{z}(\theta)^H\mathbf{u}|^2} \quad \forall \quad \theta \in \Theta \; ;$$

5: Determine source DOA's by locating S *sharpest peaks of* $P_{\text{SMN}}(\theta)$*.*

Algorithm 3.4 Root-Min-Norm.

1-2: See steps 1 and 2 of Alg. 3.1;

3: Use matrix $\mathbf{U}^\nu$ *of noise eigenvectors to construct minimum norm vector* $\mathbf{u}$*:*

$$\mathbf{u} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H\|} = \frac{\mathbf{U}^\nu(\tilde{\mathbf{u}}_1^\nu)^H}{\|\tilde{\mathbf{u}}_1^\nu\|} \; ;$$

4: Construct Root-Min-Norm polynomial:

$$p_{\,\text{RMN}}(z) \triangleq U\big(z\big) \triangleq \sum_{k=0}^{D-1} u_{k+1}\,(z)^{-k} \; ;$$

5: Compute roots of $p_{\,\text{RMN}}(z)$*;*

6: Select S *unique roots of* $p_{\,\text{RMN}}(z)$ *that lie on or closest to unit circle;*

7: For each source root z*, compute corresponding source DOA (see* (3.2.39)*):*

$$\theta = \arcsin\Big(\tfrac{\lambda_c}{2\pi d}\arg(z)\Big).$$

Examples and discussion

The results of applying the Spectral- and Root-Min-Norm methods in Alg. 3.3 and Alg. 3.4 respectively to the same examples that were presented earlier are shown in Figures 3.6–3.9. Because for $D = 4$ and $S = 3$ the noise subspace is one-dimensional, the pseudo-spectra of SPECTRAL-MUSIC and Spectral-Min-Norm are exactly the same. This can be verified by comparing Fig. 3.2 to Fig. 3.6. Likewise, half of the roots of the ROOT-MUSIC polynomial are equal to the roots of the Root-Min-Norm polynomial; this can be verified by comparing Fig. 3.4 to Fig. 3.7. Since the interpretation of the results in these figures follows exactly the same lines as in Sections 3.2.2 and 3.2.3, we leave that, and the comparison of Min-Norm to Spectral- and ROOT-MUSIC, to the reader. Statistical performance analyses of the Min-Norm algorithm and related issues can be found in [93, 103, 132].

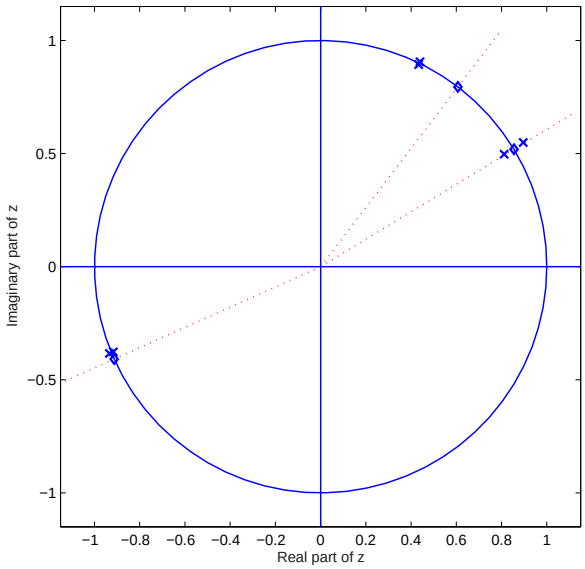

Figure 3.4: *Ideal ($\diamond$) and estimated ($\times$) roots of the ROOT-MUSIC polynomial $p_{\mathrm{RM}}(z)$ for $D = 4$ and $S = 3$.*

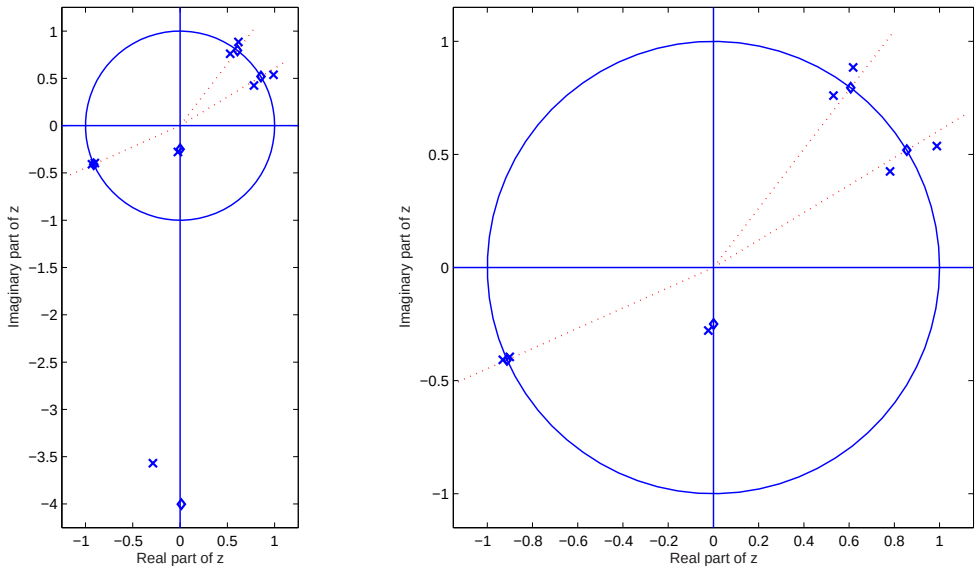

Figure 3.5: *Ideal ($\diamond$) and estimated ($\times$) roots of the ROOT-MUSIC polynomial $p_{\mathrm{RM}}(z)$ for $D = 5$ and $S = 3$; left: all roots, right: area around unit circle enlarged.*

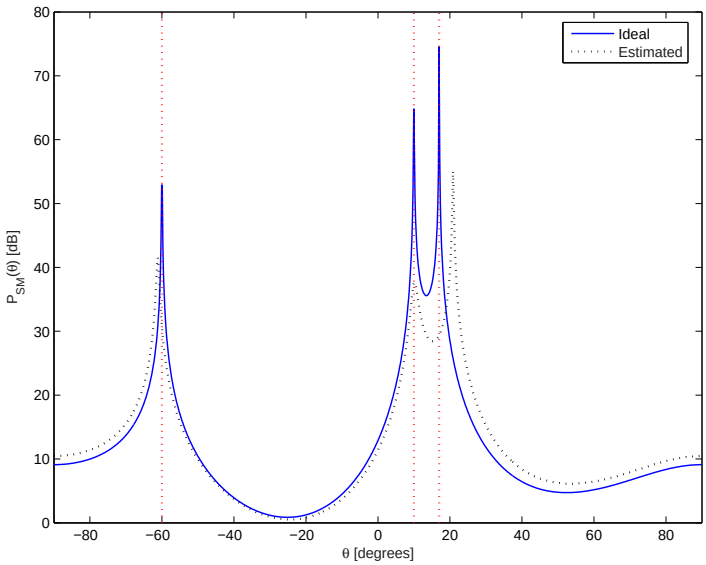

Figure 3.6: *Ideal and estimated Min-Norm spectra* $P_{\text{SMN}}(\theta)$ *for* $D = 4$ *and* $S = 3$.

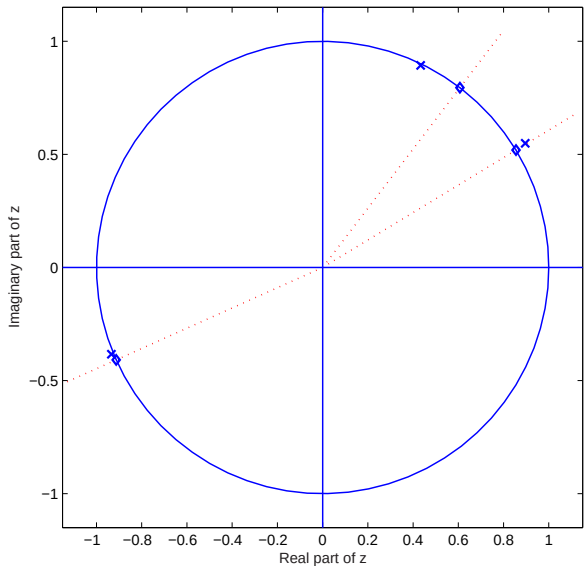

Figure 3.7: *Ideal* ($\diamond$) *and estimated* ($\times$) *roots of the Root-Min-Norm polynomial* $p_{\text{RMN}}(z)$ *for* $D = 4$ *and* $S = 3$.

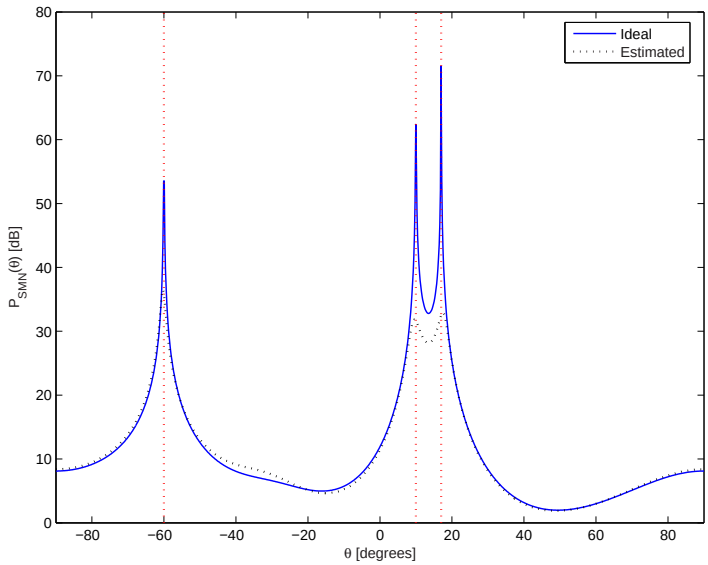

Figure 3.8: *Ideal and estimated Min-Norm spectra* $P_{\text{SMN}}(\theta)$ *for* $D = 5$ *and* $S = 3$.

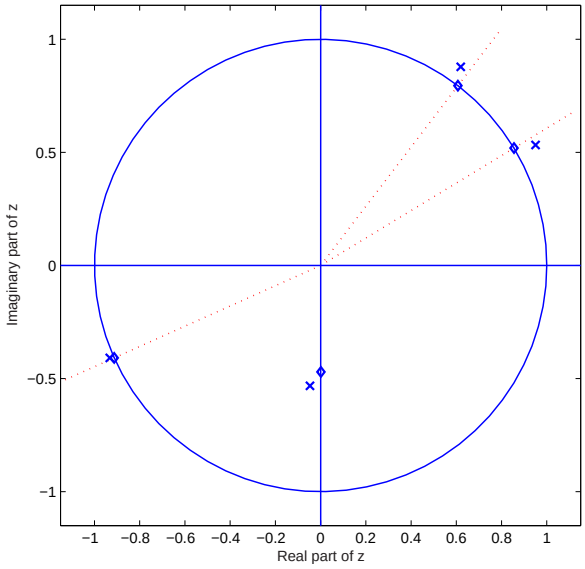

Figure 3.9: *Ideal* ($\diamond$) *and estimated* ($\times$) *roots of the Root-Min-Norm polynomial* $p_{\text{RMN}}(z)$ *for* $D = 5$ *and* $S = 3$.

3.3 Subspace Methods for DOA estimation based on Higher Order Statistics

In this section, we generalize the key principles of the subspace techniques explained in Section 3.2 for SOS to HOS. Although several 'different' algorithms based on HOS and subspace methods have been presented in the literature [28, 40, 129, 140, 190], basically they all rely on the same simple subspace decomposition principles and on the a priori knowledge of the functional form of the array response vectors. In fact, the main differences between different HOS based subspace approaches concern the manner in which a matrix containing (lag-zero or temporal higher order) sensor statistics for subspace analysis is formed. The specific choice for arranging the sensor statistics in this matrix has a large influence on the maximum number of DOA's that can be identified, and on the 'computational convenience' of a certain algorithm. See Section 1.2.6, where we stressed already the importance of the arrangement of the exploited sensor statistics. See also Chapter 4, where SOS based subspace techniques are employed for solving a relatively simple MIBI problem; it is concluded there that the specific arrangement of certain sensor correlation values in a 'subspace matrix' is crucial. Because not the specific type of exploited sensor statistics determines whether or not subspace techniques can be used, but rather their possible 'matrix arrangements', it follows directly that subspace methods can be developed for any type of statistical variability in the data, i.e. arbitrary order lag-zero or temporal statistics, by considering a properly defined matrix. In order to demonstrate and explain these remarks, we will describe a representative lag-zero Higher Order Statistics based subspace method in the next sections. More specifically, we will explain the rationale behind a DOA estimation algorithm that is based on the lag-zero fourth order cumulants of the sensor array data, viz. the algorithm of Porat and Friedlander presented in [129]. Similarly to the SOS based subspace methods discussed in Section 3.2, the approach in [129] does not exploit any temporal structure; therefore, (again) all time indices are dropped and time signals are considered as random variables instead of time series. Basically, we will follow the key steps of the derivation in Section 3.2.1. Along the way, we introduce additional notation that is convenient for handling HOS and that we will use later on in the thesis.

3.3.1 Assumptions on source signals, noise signals, and mixing matrix

As in Section 3.2.1.1, we start by summarizing the main assumptions on which the algorithm of Porat and Friedlander [129] relies; see the list at the end of this section. We will modify some of the assumptions made in [129] in order to make them suitable for our purposes, and also add some to make things more explicit. Firstly, it is assumed that all source signals are realizations of zero mean, white, stationary, ergodic and non-Gaussian random processes (*AS1*).

Secondly, we assume that the source signals are mutually statistically independent. Although the algorithm presented in [129] adopts a more general assumption that allows the source signals to be statistically dependent, here we make this stronger assumption for convenience of comparison with our previous explanations. In fact, the algorithm we will present only needs statistical independence at the fourth order (see Definition 2.7.2 in Section 2.7.3.1, and Section B.2.6.8); therefore, instead of full statistical independence we will assume here that all fourth order cross-cumulants of the source signals are zero (*AS2*). Using the notation explained in Sections A.5, A.6 and B.2.6.5, this assumption is written as:

$$\kappa_{\mathbf{j}_4}^{s,\mathbf{c}_4} \equiv \kappa_{j_1 j_2 j_3 j_4}^{s,c_1 c_2 c_3 c_4} \triangleq \mathrm{cum}\big(s_{j_1}, (s_{j_2})^*, (s_{j_3})^*, s_{j_4}\big) = 0 \quad \forall\, (j_1, j_2, j_3, j_4) \in \mathcal{J}_{\mathrm{c},S}^4\,, \tag{3.3.1}$$

where $\mathrm{cum}(\cdot, \cdots, \cdot)$ is the cumulant function defined in Section B.2.6.5 and $\mathbf{c}_4$ is a fixed *conjugation tuple* chosen as $\mathbf{c}_4 \triangleq (c_1, c_2, c_3, c_4) = (\circ, *, *, \circ)$, where '$*$' denotes conjugation and '$\circ$' no conjugation; see Appendix A (in particular Sections A.5 and A.6) and Section 2.7.4.1. Clearly, we also need to assume that the source auto-cumulants are non-zero:

$$\kappa_j^{s,\mathbf{c}_4} \equiv \kappa_{jjjj}^{s,\mathbf{c}_4} \triangleq \mathrm{cum}\big(s_j, (s_j)^*, (s_j)^*, s_j\big) \neq 0 \qquad \forall\, 1 \leq j \leq S\,. \tag{3.3.2}$$

Combining (3.3.1) and (3.3.2) yields the following property, formulated in concise notation:

$$\kappa_{\mathbf{j}_4}^{s,\mathbf{c}_4} = \delta_{\mathbf{j}_4} \kappa_{\mathbf{j}_4}^{s,\mathbf{c}_4} \qquad \forall\, \mathbf{j}_4 \in \mathcal{J}_{\mathrm{t},S}^4\,, \tag{3.3.3}$$

where $\delta_{\mathbf{j}_4} \triangleq \delta_{j_1 j_2 j_3 j_4}$ denotes a Kronecker delta function, which equals one iff all indices in $\mathbf{j}_4 = (j_1, j_2, j_3, j_4)$ are equal, i.e. $j_1 = j_2 = j_3 = j_4$, and zero otherwise (see Section A.4). For completeness and clarity, as in Section 3.2.1.1 we explicitly assume that the source DOA's are different (*AS3*).

Next, it is assumed that the source and sensor noise signals are mutually statistically independent (*AS4*). In fact, we do not strictly need this assumption because of the next assumption, i.e. (*AS5*), which states that the stochastic process behind the sensor noise is complex circular Gaussian, implying that $\mathrm{cum}\big(\nu_{i_1}, (\nu_{i_2})^*, (\nu_{i_3})^*, \nu_{i_4}\big) = 0$ for all $1 \leq i_1, \ldots, i_4 \leq D$. The spatial noise covariance matrix $\mathbf{R}^\nu \triangleq E\{\boldsymbol{\nu}\boldsymbol{\nu}^H\}$ may be anything as long as this requirement is satisfied. In addition, note that the noise is allowed to have any temporal structure because this is ignored anyway. Hence, the fourth order noise cumulants are zero (due to the 'complex version' of *UCP2* on page 425) and do not contribute to the fourth order sensor cumulants. *AS4* is only mentioned here because it is a very natural and common one, and for consistency and comparison with the assumptions in Section 3.2.1.1.

Assumptions *AS6* and *AS7* concerning the properties of the array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ are the analogues of *AS6* and *AS7* in Section 3.2.1.1. Although Porat and Friedlander adopted *AS7* explicitly, we will show that it is not required because the algorithm can also 'discriminate' the DOA's of statistically independent sources when there are more sources than sensors, i.e. $S > D$, up to a certain maximum, of course. Because this was not noticed by Porat and Friedlander, they assumed that $D > S$ in [129]. In Chapter 7, we will prove that the maximum number of source DOA's that can be estimated using the algorithm that we will present in Section 3.3.6, viz. Alg. 3.5, in theory equals $S_{\max} = 2(D-1)$. Hence, with $2, 3, 4, 5$, etc. sensors respectively, $2, 4, 6, 8$, etc. source DOA's can be identified. Similarly to *AS4*, assumption *AS7* is not required here but it is listed for consistency and comparison with the assumptions in Section 3.2.1.1. Instead, we adopt assumption *AS7'*.

Finally, the important assumption *AS8* has replaced *AS8* of Section 3.2.1.1: instead of requiring that the source DOA's can be determined uniquely from $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)\big)$, we now require that they can be determined uniquely from $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)$, where $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big) \in \mathbb{C}^S_{(D)^2}$ is the so-called second order Khatri-Rao product with conjugation pair $(c_1, c_2) = (\circ, *)$; see (3.3.12), Chapter 5, Section A.9 and in particular equations (A.9.4) and (A.9.5) for the precise definition of $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$. The reason why this assumption is needed will become clear in the following sections. Summarizing, the lag-zero HOS based subspace method in [129] exploits the following (modified) assumptions:

AS1: The source signals are realizations of zero mean, white, stationary, ergodic and non-Gaussian random processes;

AS2: All fourth order source cross-cumulants are zero, whereas all auto-cumulants are non-zero: $\mathrm{cum}\big(s_{j_1},(s_{j_2})^*,(s_{j_3})^*,s_{j_4}\big)=\delta_{j_1j_2j_3j_4}\,\mathrm{cum}\big(s_{j_1},(s_{j_2})^*,(s_{j_3})^*,s_{j_4}\big)$ *for all* $1\leq j_1,j_2,j_3,j_4\leq S$ *with* $\mathrm{cum}\big(s_j,(s_j)^*,(s_j)^*,s_j\big)\neq 0 \quad \forall\, 1\leq j\leq S$*;*

AS3: The source DOA's are different: $\theta^i\neq\theta^j \quad \forall\, 1\leq i\neq j\leq S$*;*

AS4: The source and sensor noise signals are mutually statistically independent;

AS5: The sensor noise signals are realizations of complex circular Gaussian processes: $\mathrm{cum}\big(\nu_{i_1},(\nu_{i_2})^*,(\nu_{i_3})^*,\nu_{i_4}\big)=0$ *for all* $1\leq i_1,\ldots,i_4\leq D$*;*

AS6: The parameterization of the array manifold $\boldsymbol{\mathcal{A}}(\Theta)$*, i.e. the functional dependence of* $\mathbf{z}(\theta)$ *on* θ *for all* $\theta\in\Theta$*, is known a priori;*

AS7: The number of sources is smaller than the number of sensors, i.e. $S<D$*;*

AS7': The number of sources is smaller than $2(D-1)+1$*, i.e.* $S<2(D-1)+1$*;*

AS8: The second order Khatri-Rao product $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ *of* $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ *is unambiguous over the parameter space* Θ*, i.e. the source DOA's can be determined uniquely from* $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ *and the generalized array manifold* $\boldsymbol{\mathcal{A}}^{\circ *}(\Theta)$ *defined in Definition 3.3.1 on page 117.*

3.3.2 Structure and matrix of fourth order sensor cumulants

Firstly, we will consider the structure of the fourth order sensor cumulants, viz. the manner in which they are expressed in terms of the mixing matrix elements and the fourth order source auto-cumulants. Then, we will highlight their specific arrangement in a 'sensor cumulant matrix' $\mathbf{C}^x$ according to [129]. We will employ our own way and notation for deriving and describing this matrix (which are different from the one of Porat and Friedlander) since in our opinion these are more straightforward. Moreover, our derivation and notation suggest several generalizations in natural and intuitive ways, which is one of the contributions of this thesis. First note that according to (3.1.1), the i-th sensor signal $x_i[n]$ is given by:

$$x_i[n]=\sum_{j=1}^{S} z_i(\theta^j)s_j[n]+\nu_i[n] \qquad \forall\, 1\leq i\leq D,\quad \forall\, n\in\mathbb{Z}\,, \tag{3.3.4}$$

where $z_i(\theta^j)$ is the instantaneous transfer from the j-th source to the i-th sensor. For convenience of expressing the fourth order sensor cumulants in terms of the mixing matrix elements (and the fourth order source auto-cumulants), we will momentarily write a_i^j and $\mathbf{a}^j$ instead of $z_i(\theta^j)$ and $\mathbf{z}(\theta^j)$ respectively, and 're-denote' the explicit dependence on the source DOA's later on. In addition, as we said earlier, we will omit the time indices. Hence, for the moment we write x_i as follows (see also (2.3.1)):

$$x_i=\sum_{j=1}^{S} a_i^j s_j+\nu_i \qquad \forall\, 1\leq i\leq D\,. \tag{3.3.5}$$

From (B.2.44d) with $v_{i_1}=x_{i_1}$, $v_{i_2}=(x_{i_2})^*$, $v_{i_3}=(x_{i_3})^*$, $v_{i_4}=x_{i_4}$, it is clear that the (i_1,i_2,i_3,i_4)-th fourth order sensor cumulant $\kappa_{i_1i_2i_3i_4}^{x,c_1c_2c_3c_4}$ with conjugation tuple $(c_1,c_2,c_3,c_4)=(\circ,*,*,\circ)$ is defined as follows:

$$\begin{aligned}
\kappa_{\mathbf{i}_4}^{x,\mathbf{c}_4} &= \kappa_{i_1 i_2 i_3 i_4}^{x,c_1 c_2 c_3 c_4} \triangleq \operatorname{cum}\big(x_{i_1},(x_{i_2})^*,(x_{i_3})^*,x_{i_4}\big) \\
&= E\big\{x_{i_1}\,(x_{i_2})^*\,(x_{i_3})^*\,x_{i_4}\big\} - E\big\{x_{i_1}\,(x_{i_2})^*\big\}E\big\{(x_{i_3})^*\,x_{i_4}\big\} - E\big\{x_{i_1}\,(x_{i_3})^*\big\}E\big\{(x_{i_2})^*\,x_{i_4}\big\} \\
&\quad - E\big\{x_{i_1}\,x_{i_4}\big\}E\big\{(x_{i_2})^*\,(x_{i_3})^*\big\} \qquad \forall\,(i_1,i_2,i_3,i_4)\in\mathcal{I}_{\mathrm{t},D}^4\,. \qquad (3.3.6)
\end{aligned}$$

Using (3.3.3) and properties *CP1-CP8* of joint cumulants listed in Section B.2.6.7, all fourth order sensor cumulants $\{\kappa_{\mathbf{i}_4}^{x,\mathbf{c}_4}\}_{\mathbf{i}_4\in\mathcal{I}_{\mathrm{t},D}^4} = \{\kappa_{i_1 i_2 i_3 i_4}^{x,c_1 c_2 c_3 c_4}\}_{(i_1,i_2,i_3,i_4)\in\mathcal{I}_{\mathrm{t},D}^4}$ can now be expressed in terms of the elements of the mixing matrix and the fourth order source auto-cumulants in the following manner:

$$\begin{aligned}
&\kappa_{i_1 i_2 i_3 i_4}^{x,c_1 c_2 c_3 c_4} \triangleq \operatorname{cum}\big(x_{i_1},(x_{i_2})^*,(x_{i_3})^*,x_{i_4}\big) \\
&= \operatorname{cum}\left(\sum_{j_1=1}^{S} a_{i_1}^{j_1}s_{j_1}+\nu_{i_1},\left(\sum_{j_2=1}^{S} a_{i_2}^{j_2}s_{j_2}+\nu_{i_2}\right)^*,\left(\sum_{j_3=1}^{S} a_{i_3}^{j_3}s_{j_3}+\nu_{i_3}\right)^*,\sum_{j_4=1}^{S} a_{i_4}^{j_4}s_{j_4}+\nu_{i_4},\right) \\
&= \sum_{j_1=1}^{S}\sum_{j_2=1}^{S}\sum_{j_3=1}^{S}\sum_{j_4=1}^{S} a_{i_1}^{j_1}\big(a_{i_2}^{j_2}\big)^*\big(a_{i_3}^{j_3}\big)^* a_{i_4}^{j_4}\ \operatorname{cum}\big(s_{j_1},(s_{j_2})^*,(s_{j_3})^*,s_{j_4}\big) \\
&\qquad + \operatorname{cum}\big(\nu_{i_1},(\nu_{i_2})^*,(\nu_{i_3})^*,\nu_{i_4}\big) \\
&= \sum_{j=1}^{S} a_{i_1}^{j}\big(a_{i_2}^{j}\big)^*\big(a_{i_3}^{j}\big)^* a_{i_4}^{j}\ \operatorname{cum}\big(s_j,(s_j)^*,(s_j)^*,s_j\big) \quad + \quad 0 \\
&= \sum_{j=1}^{S} a_{i_1}^{j}\big(a_{i_2}^{j}\big)^*\big(a_{i_3}^{j}\big)^* a_{i_4}^{j}\ \kappa_{jjjj}^{s,c_1 c_2 c_3 c_4} \qquad \forall\,(i_1,i_2,i_3,i_4)\in\mathcal{I}_{\mathrm{t},D}^4\,. \qquad (3.3.7)
\end{aligned}$$

Using our concise notation explained in Appendix A, this reads:

$$\kappa_{\mathbf{i}_4}^{x,\mathbf{c}_4} = \sum_{j=1}^{S} \breve{a}_{\mathbf{i}_4}^{j,\mathbf{c}_4}\ \kappa_j^{s,\mathbf{c}_4} \qquad \forall\,\mathbf{i}_4\in\mathcal{I}_{\mathrm{t},D}^4\,, \qquad (3.3.8)$$

where $\breve{a}_{\mathbf{i}_4}^{j,\mathbf{c}_4} \triangleq a_{i_1}^{j}\big(a_{i_2}^{j}\big)^*\big(a_{i_3}^{j}\big)^* a_{i_4}^{j}$ (see (A.2.13)). Note that these sensor cumulants are noise-free due to *AS5.* Similarly to (2.7.38), the product terms of the form $\breve{a}_{\mathbf{i}_4}^{j,\mathbf{c}_4} = a_{i_1}^{j}\big(a_{i_2}^{j}\big)^*\big(a_{i_3}^{j}\big)^* a_{i_4}^{j}$ are constructed from a single vector, viz. the j-th column $\mathbf{a}^j$ of $\mathbf{A}$ and define a kind of *tetrads with conjugation tuple* $\mathbf{c}_4$. As in Section 2.7.4.1, all possible fourth order sensor cumulants are stored in a four-dimensional *cumulant tensor* or 'tensor-array' (holor) denoted by $\mathbf{K}^{x,\mathbf{c}_4}$ and indexed by the four indices i_1, i_2, i_3 and i_4, i.e. $\mathbf{K}^{x,\mathbf{c}_4} \triangleq \{\kappa_{\mathbf{i}_4}^{x,\mathbf{c}_4}\}_{\mathbf{i}_4\in\mathcal{I}_{\mathrm{t},D}^4}$; see also the other remarks concerning tensors in Section 2.7.4.1 below (2.7.38).

Now we are ready to address the important issue of arranging the sensor cumulants in a matrix, which will be denoted by $\mathbf{C}^x$ and is of size $(D)^2\times(D)^2$. Using the Kronecker product notation and rules (see Section A.8 and [21]), this can be done in a quite natural way. Firstly, consider the following matrix of size $(D)^2\times(D)^2$ that contains certain fourth order expectations (moments) of the sensor signals:

$$\begin{aligned}
E\left\{\big[\mathbf{x}\otimes\mathbf{x}^*\big]\big[\mathbf{x}\otimes(\mathbf{x})^*\big]^H\right\} &= E\left\{\big(\mathbf{x}\mathbf{x}^H\big)\otimes\big(\mathbf{x}\mathbf{x}^H\big)^*\right\} \quad \text{(see } \textit{KP4} \text{ in Section A.8)} \\
&= E\left\{\begin{bmatrix} x_1\cdot(\mathbf{x})^* \\ \vdots \\ x_D\cdot(\mathbf{x})^*\end{bmatrix}\begin{bmatrix}(x_1)^*\cdot\mathbf{x}^T & \cdots & (x_D)^*\cdot\mathbf{x}^T\end{bmatrix}\right\}
\end{aligned}$$

$$= E\begin{bmatrix} x_1(x_1)^*\cdot(\mathbf{x}\mathbf{x}^H)^* & x_1(x_2)^*\cdot(\mathbf{x}\mathbf{x}^H)^* & \cdots & x_1(x_D)^*\cdot(\mathbf{x}\mathbf{x}^H)^* \\ \vdots & \vdots & \ddots & \vdots \\ x_D(x_1)^*\cdot(\mathbf{x}\mathbf{x}^H)^* & x_D(x_2)^*\cdot(\mathbf{x}\mathbf{x}^H)^* & \cdots & x_D(x_D)^*\cdot(\mathbf{x}\mathbf{x}^H)^* \end{bmatrix}$$

$$= \begin{bmatrix} E\{x_1(x_1)^*(x_1)^*x_1\} & \cdots & E\{x_1(x_D)^*(x_1)^*x_D\} \\ \vdots & \ddots & \vdots \\ E\{x_D(x_1)^*(x_D)^*x_1\} & \cdots & E\{x_D(x_D)^*(x_D)^*x_D\} \end{bmatrix}. \tag{3.3.9}$$

Hence, $E\{x_{i_1}(x_{i_2})^*(x_{i_3})^*x_{i_4}\}$ is the element positioned at the $\big((i_1-1)D+i_2\big)$-th row and $\big((i_3-1)D+i_4\big)$-th column of this matrix. The sensor cumulant matrix $\mathbf{C}^x \in \mathbb{C}^{(D)^2}_{(D)^2}$ of Porat and Friedlander is defined similarly: it is obtained by replacing $E\{x_{i_1}(x_{i_2})^*(x_{i_3})^*x_{i_4}\}$ in (3.3.9) with $\mathrm{cum}\,(x_{i_1},(x_{i_2})^*,(x_{i_4})^*,x_{i_4})$ for all $1 \le i_1,i_2,i_3,i_4 \le D$. Using our notation, we can also define it quite naturally and directly as follows:

$$\begin{aligned}\mathbf{C}^x &\triangleq \mathrm{cum}\big(\mathbf{x},(\mathbf{x})^*,\mathbf{x}^H,(\mathbf{x})^T\big) \\ &= \begin{bmatrix} \mathrm{cum}\big(x_1,(x_1)^*,(x_1)^*,x_1\big) & \cdots & \mathrm{cum}\big(x_1,(x_D)^*,(x_1)^*,x_D\big) \\ \vdots & \ddots & \vdots \\ \mathrm{cum}\big(x_D,(x_1)^*,(x_D)^*,x_1\big) & \cdots & \mathrm{cum}\big(x_D,(x_D)^*,(x_D)^*,x_D\big) \end{bmatrix}. \end{aligned} \tag{3.3.10}$$

Since subspace methods are based on the way a matrix like this is expressed in the source array response vectors, i.e. the columns of the mixing matrix, and source (plus noise, in general) statistics, we need to insert (3.3.7) or (3.3.8) into (3.3.10), yielding:

$$\begin{aligned}\mathbf{C}^x &= \sum_{j=1}^{S}\begin{bmatrix} a_1^j\big(a_1^j\big)^*\big(a_1^j\big)^*a_1^j & \cdots & a_1^j\big(a_D^j\big)^*\big(a_1^j\big)^*a_D^j \\ \vdots & \ddots & \vdots \\ a_D^j\big(a_1^j\big)^*\big(a_D^j\big)^*a_1^j & \cdots & a_D^j\big(a_D^j\big)^*\big(a_D^j\big)^*a_D^j \end{bmatrix}\mathrm{cum}\big(s_j,(s_j)^*,(s_j)^*,s_j\big) \\ &= \sum_{j=1}^{S}\big[\mathbf{a}^j\otimes\big(\mathbf{a}^j\big)^*\big]\big[\mathbf{a}^j\otimes\big(\mathbf{a}^j\big)^*\big]^H\kappa_j^{\mathbf{s},\mathbf{c}_4} \\ &= \Big[\mathbf{a}^1\otimes\big(\mathbf{a}^1\big)^* \;\cdots\; \mathbf{a}^S\otimes\big(\mathbf{a}^S\big)^*\Big]\begin{bmatrix}\kappa_1^{\mathbf{s},\mathbf{c}_4} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \kappa_S^{\mathbf{s},\mathbf{c}_4}\end{bmatrix}\begin{bmatrix}\big[\mathbf{a}^1\otimes\big(\mathbf{a}^1\big)^*\big]^H \\ \vdots \\ \big[\mathbf{a}^S\otimes\big(\mathbf{a}^S\big)^*\big]^H\end{bmatrix} \\ &\overset{(A.9.5)}{=} \mathbf{A}_\diamond^{\circ *}\cdot\mathrm{diag}\left(\kappa_1^{\mathbf{s},\mathbf{c}_4},\ldots,\kappa_S^{\mathbf{s},\mathbf{c}_4}\right)\cdot\big(\mathbf{A}_\diamond^{\circ *}\big)^H, \end{aligned} \tag{3.3.11}$$

where in the last line we have used the compact notation:

$$\mathbf{A}_\diamond^{c_1c_2} = \mathbf{A}_\diamond^{\circ *} \triangleq \Big[\mathbf{a}^1\otimes\big(\mathbf{a}^1\big)^* \;\cdots\; \mathbf{a}^S\otimes\big(\mathbf{a}^S\big)^*\Big], \tag{3.3.12}$$

which is the so-called second order Khatri-Rao product with conjugation pair $(c_1,c_2) = (\circ,*)$ that is defined and explained in Section A.9 (in particular, see equations (A.9.4) and (A.9.5)). The transition from the first line to the second easily follows by observing the similarity to (3.3.9). Now, by explicitly denoting the dependence on the source DOA's again, we can write (3.3.11) concisely as follows:

$$\mathbf{C}^x = \sum_{j=1}^{S}\big[\mathbf{z}(\theta^j)\otimes\big(\mathbf{z}(\theta^j)\big)^*\big]\big[\mathbf{z}(\theta^j)\otimes\big(\mathbf{z}(\theta^j)\big)^*\big]^H\kappa_j^{\mathbf{s},\mathbf{c}_4} = \mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\,\mathbf{C}^s\,\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H, \tag{3.3.13}$$

where:

$$\begin{aligned}\mathbf{C}^s &\triangleq \operatorname{diag}\left(\kappa_1^{s,\mathbf{c}_4},\ldots,\kappa_S^{s,\mathbf{c}_4}\right)\\ &= \operatorname{diag}\Big(\operatorname{cum}\big(s_1,(s_1)^*,(s_1)^*,s_1\big),\ldots,\operatorname{cum}\big(s_S,(s_S)^*,(s_S)^*,s_S\big)\Big)\,. \end{aligned} \tag{3.3.14}$$

The similarity of $\mathbf{C}^x$ in (3.3.13) to $\mathbf{R}^x$ in (3.2.1) is now obvious. As we have explained before, $\mathbf{C}^x$ does not contain a noise term similar to $(\sigma^\nu)^2\,\mathbf{I}_D^D$ in (3.2.1) because we are using fourth order cumulants that are zero for complex circular Gaussian noise signals. Note that we can view vectors of the form $\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^*$ as a kind of *generalized array response vectors*. To provide additional insight, we note that our notation allows us to derive (3.3.13) from (3.3.10) and (3.1.1) (without time indices) in a direct and intuitive manner as follows:

$$\begin{aligned}&\mathbf{C}^x \triangleq \operatorname{cum}\big(\mathbf{x},(\mathbf{x})^*,\mathbf{x}^H,(\mathbf{x})^T\big)\\ &= \operatorname{cum}\left(\sum_{j_1=1}^{S}\mathbf{a}^{j_1}s_{j_1}+\boldsymbol{\nu},\left(\sum_{j_2=1}^{S}\mathbf{a}^{j_2}s_{j_2}+\boldsymbol{\nu}\right)^*,\left(\sum_{j_3=1}^{S}\mathbf{a}^{j_3}s_{j_3}+\boldsymbol{\nu}\right)^H,\left(\sum_{j_4=1}^{S}\mathbf{a}^{j_4}s_{j_4}+\boldsymbol{\nu}\right)^T\right)\\ &= \sum_{j_1=1}^{S}\sum_{j_2=1}^{S}\sum_{j_3=1}^{S}\sum_{j_4=1}^{S}\left[\mathbf{a}^{j_1}\otimes\left(\mathbf{a}^{j_2}\right)^*\right]\cdot\operatorname{cum}\big(s_{j_1},(s_{j_2})^*,(s_{j_3})^*,s_{j_4}\big)\cdot\left[\mathbf{a}^{j_3}\otimes\left(\mathbf{a}^{j_4}\right)^*\right]^H\\ &= \sum_{j=1}^{S}\left[\mathbf{a}^{j}\otimes\left(\mathbf{a}^{j}\right)^*\right]\left[\mathbf{a}^{j}\otimes\left(\mathbf{a}^{j}\right)^*\right]^H\kappa_j^{s,\mathbf{c}_4}=\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\;\mathbf{C}^s\;\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H\;\in\;\mathbb{C}_{(D)^2}^{(D)^2}\,. \end{aligned} \tag{3.3.15}$$

Based on the expressions and results developed above, we can now derive subspace techniques for DOA estimation based on the structure of $\mathbf{C}^x$ by following the same reasoning as in Sections 3.2.1.2 through 3.2.1.5, and also Sections 3.2.2 through 3.2.4. For example, the equivalent of property (3.2.2) is given by:

$$\operatorname{rank}\big(\mathbf{C}^x\big)=\operatorname{rank}\Big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\;\mathbf{C}^s\;\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H\Big)=S\,. \tag{3.3.16}$$

This property holds owing to assumptions *AS2* and *AS8*: $\operatorname{rank}(\mathbf{C}^s)=S$ because of *AS2*, and $\operatorname{rank}\big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)=S$ because of *AS8*. Similarly to the definition of unambiguity of the array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ in Definition 3.1.2 of Section 3.1.3, we can define the unambiguity of the second order Khatri-Rao product $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$, which can be seen as a kind of *generalized array response matrix*, as follows:

Definition 3.3.1. Unambiguous second order Khatri-Rao product with conjugation pair $(\circ,*)$**.** The second order Khatri-Rao product $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ of $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big)$ with conjugation pair $(c_1,c_2)=(\circ,*)$ is said to be unambiguous over the parameter space Θ if for all $\tilde{\boldsymbol{\theta}}$ with $\theta^1,\ldots,\theta^S\in\Theta$ the source DOA's can be determined uniquely from the intersections between $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ and the *generalized array manifold* $\boldsymbol{\mathcal{A}}^{\circ *}(\Theta)$ with conjugation pair $(c_1,c_2)=(\circ,*)$ that is defined by:

$$\boldsymbol{\mathcal{A}}^{\circ *}(\Theta)\triangleq\Big\{\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^*\,\Big|\,\theta\in\Theta\Big\}\,. \tag{3.3.17}$$

Note that this definition implies that $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ must have full rank for any collection of S distinct DOA's in $\tilde{\boldsymbol{\theta}}$ with $\theta^1,\ldots,\theta^S\in\Theta$. If $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ is unambiguous over Θ, the intersections between the subspace $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)$ and the generalized array manifold $\boldsymbol{\mathcal{A}}^{\circ *}(\Theta)$ *are* the generalized source array response vectors, which determine the source DOA's uniquely. Note

that similarly to (3.1.8), (3.3.17) describes the collection of generalized array response vectors formed by varying θ over the parameter space of interest Θ, and defines a one-dimensional curve in the complex $(D)^2$-dimensional Euclidian space. We conclude this section by noting that property (3.3.16) directly implies that if the number of sources is unknown, it can be computed or estimated as the (effective) rank of $\mathbf{C}^x$ or $\hat{\mathbf{C}}^x$. In the next sections, we will briefly provide the equivalents of the main results from Sections 3.2.1.2 through 3.2.1.5. Because all results have been formulated in such a way that they can easily be derived by reasoning by analogy (in particular, this is made intuitive by our notation), we leave the details to the reader.

3.3.3 Eigenstructure of $\mathbf{C}^x$, signal and noise subspaces

Most results in this section follow immediately from the corresponding ones in Section 3.2.1.3 by analogy. Let $\{\lambda_i^s\}_{1\leq i\leq (D)^2}$ and $\{\mathbf{u}^i\}^{1\leq i\leq (D)^2}$ be the eigenvalues and eigenvectors respectively of $\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\ \mathbf{C}^s\ \mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H$ with the eigenvalues arranged in decreasing order, i.e.:

$$\Big(\mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\ \mathbf{C}^s\ \mathbf{A}_\diamond^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H\Big)\mathbf{u}^i = \lambda_i^s\mathbf{u}^i \quad \forall\ 1\leq i\leq (D)^2 \quad \text{with} \quad \lambda_1^s\geq\cdots\geq\lambda_{(D)^2}^s\,. \tag{3.3.18}$$

Since $\mathbf{C}^x$ is a Hermitian positive definite matrix of size $(D)^2\times(D)^2$ with rank S (3.3.16), it has S real-valued eigenvalues greater than zero and $(D)^2-S$ eigenvalues equal to zero. Let $\{\lambda_i^x\}_{1\leq i\leq (D)^2}$ and $\{\mathbf{u}^i\}^{1\leq i\leq (D)^2}$ be the eigenvalues and length-$(D)^2$ eigenvectors of $\mathbf{C}^x$ respectively with the eigenvalues arranged in decreasing order. Then it is clear that:

$$\lambda_i^x = \begin{cases}\lambda_i^s, & i=1,\ldots,S\,;\\ 0, & i=S+1,\ldots,(D)^2\,.\end{cases} \tag{3.3.19}$$

As in Section 3.2.1.3, the eigenvalues are divided into two groups. The non-zero eigenvalues $\{\lambda_i^x\}_{1\leq i\leq S}$ in the first group are called *signal eigenvalues*, whereas the zero eigenvalues $\{\lambda_i^x\}_{S+1\leq i\leq (D)^2}$ in the second group are called *noise eigenvalues*. Since there is no noise term contributing to $\mathbf{C}^x$, *zero eigenvalues* would be a more appropriate name for the latter category (at least for ideal statistics). Likewise, as in Section 3.2.1.3 the corresponding eigenvectors are divided into two groups. The eigenvectors $\{\mathbf{u}^i\}^{1\leq i\leq S}$ corresponding to the signal eigenvalues are called *signal eigenvectors* and span an S-dimensional subspace $\mathcal{L}\left(\{\mathbf{u}^i\}^{1\leq i\leq S}\right)$ called *signal subspace*. The eigenvectors $\{\mathbf{u}^i\}^{S+1\leq i\leq (D)^2}$ corresponding to the zero eigenvalues are called *noise eigenvectors* or *zero eigenvectors* and span a $\big((D)^2-S\big)$-dimensional subspace $\mathcal{L}\left(\{\mathbf{u}^i\}^{S+1\leq i\leq (D)^2}\right)$ called *noise subspace* or *zero subspace*. Summarizing, the EVD of $\mathbf{C}^x$ can be written as follows:

$$\mathbf{C}^x = \sum_{i=1}^{(D)^2}\lambda_i^x\mathbf{u}^i(\mathbf{u}^i)^H = \sum_{i=1}^{S}\lambda_i^s\mathbf{u}^i(\mathbf{u}^i)^H + \sum_{i=S+1}^{(D)^2}0\cdot\mathbf{u}^i(\mathbf{u}^i)^H = \sum_{i=1}^{S}\lambda_i^s\mathbf{u}^i(\mathbf{u}^i)^H. \tag{3.3.20}$$

In matrix notation, this decomposition can be written as:

$$\boxed{\mathbf{C}^x = \mathbf{U}^x\boldsymbol{\Lambda}^x(\mathbf{U}^x)^H = \mathbf{U}^s\boldsymbol{\Lambda}^s(\mathbf{U}^s)^H + \mathbf{U}^\nu\boldsymbol{\Lambda}^\nu(\mathbf{U}^\nu)^H = \mathbf{U}^s\boldsymbol{\Lambda}^s(\mathbf{U}^s)^H\,,} \tag{3.3.21}$$

where:

$$\mathbf{U}^x \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^{(D)^2}\end{bmatrix} \in \mathbb{C}^{(D)^2}_{(D)^2}, \quad \mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^S\end{bmatrix} \in \mathbb{C}^{S}_{(D)^2}, \quad \text{and}$$
$$\mathbf{U}^\nu \triangleq \begin{bmatrix}\mathbf{u}^{S+1} & \cdots & \mathbf{u}^{(D)^2}\end{bmatrix} \in \mathbb{C}^{(D)^2-S}_{(D)^2}$$

are matrices containing the sensor, signal and noise eigenvectors respectively, and:

$$\begin{aligned}&\boldsymbol{\Lambda}^x \triangleq \operatorname{diag}(\lambda_1^x, \ldots, \lambda_{(D)^2}^x) \in \mathbb{R}_{(D)^2}^{(D)^2}, \quad \boldsymbol{\Lambda}^s \triangleq \operatorname{diag}(\lambda_1^s, \ldots, \lambda_S^s) \in \mathbb{R}_S^S, \quad \text{and} \\ &\boldsymbol{\Lambda}^\nu \triangleq \operatorname{diag}(0, \ldots, 0) \in \mathbb{R}_{(D)^2-S}^{(D)^2-S}\end{aligned} \tag{3.3.22}$$

are diagonal matrices containing the sensor, signal and noise/zero eigenvalues respectively. Compare (3.3.20), (3.3.21), and (3.3.22) with (3.2.9), (3.2.10), and (3.2.11) respectively. As in Section 3.2.1.3, notice that all matrices $\mathbf{U}^x$, $\boldsymbol{\Lambda}^x$, $\mathbf{U}^s$, $\boldsymbol{\Lambda}^s$, $\mathbf{U}^\nu$ and $\boldsymbol{\Lambda}^\nu$ depend on the source DOA's in $\tilde{\boldsymbol{\theta}}$ because they are determined by the sensor cumulant matrix $\mathbf{C}^x$, which is a function of $\tilde{\boldsymbol{\theta}}$; see (3.3.13) and (3.3.15), for example. The matrices resulting from the EVD satisfy several properties. Firstly, note that:

$$\mathbf{U}^x = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix} \quad \text{and} \quad \boldsymbol{\Lambda}^x = \begin{bmatrix}\boldsymbol{\Lambda}^s & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}^\nu\end{bmatrix} = \begin{bmatrix}\boldsymbol{\Lambda}^s & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix},$$

and that the signal and zero subspaces are orthogonal complements of each other. In addition, employing (3.3.21) together with fundamental subspace properties *FSP1–FSP6* on page 450 and the fact that $\mathbf{C}^x$ is Hermitian yields the following enlightening relations regarding the fundamental subspaces of the subspace matrix $\mathbf{C}^x$:

$$\begin{aligned}&\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x) = \mathcal{R}_{\mathrm{c}}(\mathbf{U}^s) \doteqdot \mathcal{N}_{\mathrm{l}}(\mathbf{U}^\nu); \\ &\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x) = \mathcal{R}_{\mathrm{c}}(\mathbf{U}^\nu) \doteqdot \mathcal{N}_{\mathrm{l}}(\mathbf{U}^s); \\ &\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x) = \left(\mathcal{R}_{\mathrm{c}}\left((\mathbf{C}^x)^H\right)\right)^T = \left(\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x)\right)^T = \left(\mathcal{R}_{\mathrm{c}}(\mathbf{U}^s)\right)^T = \mathcal{N}_{\mathrm{l}}(\mathbf{U}^\nu); \\ &\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x) = \left(\mathcal{N}_{\mathrm{r}}\left((\mathbf{C}^x)^H\right)\right)^T = \left(\mathcal{R}_{\mathrm{c}}(\mathbf{U}^\nu)\right)^T = \mathcal{N}_{\mathrm{l}}(\mathbf{U}^s),\end{aligned}$$

and thus:

$$\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x) = \left(\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x)\right)^T = \mathcal{R}_{\mathrm{c}}(\mathbf{U}^s) \doteqdot \mathcal{N}_{\mathrm{l}}(\mathbf{U}^\nu); \tag{3.3.23}$$

$$\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x) = \left(\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x)\right)^T = \mathcal{R}_{\mathrm{c}}(\mathbf{U}^\nu) \doteqdot \mathcal{N}_{\mathrm{l}}(\mathbf{U}^s). \tag{3.3.24}$$

Note that the latter two equations also state that the row and column spaces, as well as the right and left null spaces of $\mathbf{C}^x$, are isomorphic. Clearly, this is due to the fact that $\mathbf{C}^x$ is Hermitian. Finally, we remark that the definitions in (3.2.17) and property (3.2.18) remain unchanged; only the sizes of the matrices need to be adjusted accordingly.

3.3.4 Relation between generalized array response vectors and subspaces; obtaining a system of equations satisfied by source DOA's

Now we are in a position to draw conclusions w.r.t. the relation between the generalized array response vectors on the one hand, and the signal and noise subspaces on the other, which lead us to a system of equations in the unknown DOA θ. We start by proving the following theorem which states that the linear span of the signal eigenvectors, i.e. the signal subspace, equals the linear span of the generalized array response vectors if $\mathbf{C}^s$ has full rank (which is the case in our problem definition: see *AS2* on page 114).

Theorem 3.3.2. Linear span of signal eigenvectors equals linear span of generalized array response vectors.
If the source auto-cumulant matrix $\mathbf{C}^s$ has full rank, the linear span of the signal eigenvectors equals the linear span of the generalized array response vectors:

$$\boxed{\mathcal{L}\left(\{\mathbf{u}^i\}^{1\leq i\leq S}\right) = \mathcal{L}\left(\left\{\mathbf{z}(\theta^j)\otimes\left(\mathbf{z}(\theta^j)\right)^*\right\}^{1\leq j\leq S}\right)\,,} \tag{3.3.25}$$

or equivalently in matrix language (compare these equations to (3.2.19)*):*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)\,.} \tag{3.3.26}$$

Proof. This assertion can be proven along the same lines as Theorem 3.2.1 presented in Section 3.2.1.4. From (3.3.18) it follows that each column of $\mathbf{U}^s$ belongs to the column space of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$, i.e.:

$$\mathbf{u}^i \in \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big) \qquad \forall\, 1\leq i\leq S\,,$$

and that each column of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ belongs to the column space of $\mathbf{U}^s$, i.e.:

$$\mathbf{z}(\theta^j)\otimes\left(\mathbf{z}(\theta^j)\right)^* \;\in\; \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) \qquad \forall\, 1\leq j\leq S\,.$$

Hence, each column of $\mathbf{U}^s$ can be expressed as a linear combination of the columns of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$ and vice versa. From this the assertion follows immediately. □

Having established this theorem, the following generalizations of Corollaries 3.2.2 and 3.2.3 are evident.

Corollary 3.3.3. Noise subspace is isomorphic to left null space of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$.
Under the assumption(s) of Theorem 3.3.2 the noise subspace is isomorphic to the left null space of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$:

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{\nu}\right) \doteqdot \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big) \qquad \equiv \qquad \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{\nu}\right) = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)\Big)^T\,.} \tag{3.3.27}$$

Proof. Applying property *FSP1* on page 450 to both sides of (3.3.26) yields:

$$\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^s\big)\Big)^{\perp} = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)\Big)^{\perp} \quad\Longrightarrow\quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{U}^s\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)\big)\,.$$

Combining this result with (3.2.14) directly implies the corollary. □

Corollary 3.3.4. Noise subspace equals (right) null space of Hermitian of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)$.
Under the assumption(s) of Theorem 3.2.1 the noise subspace equals the (right) null space of $\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H$:

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{\nu}\right) = \mathcal{N}_{\mathrm{r}}\Big(\mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big)^H\Big)\,.} \tag{3.3.28}$$

Proof. Applying property *FSP4* on page 450 to (3.3.27) directly yields the corollary. □

Using Theorem 3.3.2 and Corollaries 3.3.3 and 3.3.4, it follows immediately that (compare this with (3.2.22)):

$$\boxed{\left(\mathbf{U}^{\nu}\right)^H \mathbf{A}_{\diamond}^{\circ *}\big(\tilde{\boldsymbol{\theta}}\big) = \mathbf{0}^S_{(D)^2-S}\,,} \tag{3.3.29}$$

which is equivalent to:

$$\boxed{\left(\mathbf{u}^i\right)^H \left[\mathbf{z}(\theta^j) \otimes \left(\mathbf{z}(\theta^j)\right)^*\right] = 0 \qquad \forall\, S+1 \leq i \leq (D)^2, \quad \forall\, 1 \leq j \leq S\ .} \tag{3.3.30}$$

Hence, considering the generalized array response vector $\mathbf{z}(\theta) \otimes \left(\mathbf{z}(\theta)\right)^*$ as a function-valued vector, we have the following (compare this to (3.2.23)):

$$\boxed{\left\langle \mathbf{u}, \mathbf{z}(\theta) \otimes \left(\mathbf{z}(\theta)\right)^* \right\rangle \Big|_{\theta=\theta^j} = 0 \qquad \forall\, 1 \leq j \leq S, \quad \forall\, \mathbf{u} \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right)\ ,} \tag{3.3.31}$$

where the inner product denoted by $\langle \cdot, \cdot \rangle$ is now a function from $\mathbb{C}_{(D)^2} \times \mathbb{C}_{(D)^2}$ to $\mathbb{C}$ defined by $\langle \mathbf{v}, \mathbf{w} \rangle \triangleq \mathbf{v}^H \mathbf{w} = \sum_{i=1}^{(D)^2} (v_i)^* \, w_i \quad \forall\ \mathbf{v}, \mathbf{w} \in \mathbb{C}_{(D)^2}$ (see Appendix A). Equation (3.3.31) shows that *the desired source DOA's can be obtained by solving θ from a system of equations of the form* $\left\langle \mathbf{u}, \mathbf{z}(\theta) \otimes \left(\mathbf{z}(\theta)\right)^* \right\rangle = 0$ *(or* $\mathbf{u}^H \left[\mathbf{z}(\theta) \otimes \left(\mathbf{z}(\theta)\right)^*\right] = 0$*, which is the same) for a 'sufficient' number of vectors* $\mathbf{u}$ *belonging to the zero subspace.* Obviously, in practice such a system can only be solved approximately because only an approximate measured version of the sensor cumulant matrix is available (see next section).

Summarizing, from (3.3.31) it follows that the source DOA's can be found by solving the following system of nonlinear equations for θ (compare this with (3.2.24)):

$$\boxed{\left\langle \mathbf{u}, \mathbf{z}(\theta) \otimes \left(\mathbf{z}(\theta)\right)^* \right\rangle = 0 \qquad \forall\, \mathbf{u} \in \mathcal{U}^\nu\ ,} \tag{3.3.32}$$

where $\mathcal{U}^\nu$ is an appropriate set of null/zero subspace vectors. As for Second Order Statistics based subspace methods, the particular choice of such a set is one of the characteristics that distinguishes one category of subspace methods from another. Again, based on (3.3.32) algorithms similar to MUSIC, ROOT-MUSIC, and MIN-NORM can be devised. See the comments after (3.2.24) in Section 3.2.1.4 for issues and possibilities concerning the choice of the set $\mathcal{U}^\nu$.

3.3.5 Dealing with estimated statistics

As we have remarked in Section 3.2.1.5, in practice we have to work with estimated statistics. In this case, an estimate $\hat{\mathbf{C}}^x$ of the sensor cumulant matrix, defined in (3.3.10) for example, has to be employed. We will not concern ourselves here with the estimation procedure and associated problems, such as the possibly large number of samples that may be required for reliable estimation (see Chapter 1, Sections B.1.3.6 and B.2.6.9, and [122, 130, 154] for information on this topic), but rather assume that a statistically consistent estimator $\hat{\mathbf{C}}^x$ of $\mathbf{C}^x$ is available and focus on principles. The equivalent of (3.2.26) is given by:

$$\hat{\mathbf{C}}^x = \hat{\mathbf{U}}^x \hat{\mathbf{\Lambda}}^x (\hat{\mathbf{U}}^x)^H = \hat{\mathbf{U}}^s \hat{\mathbf{\Lambda}}^s (\hat{\mathbf{U}}^s)^H + \hat{\mathbf{U}}^\nu \hat{\mathbf{\Lambda}}^\nu (\hat{\mathbf{U}}^\nu)^H \approx \hat{\mathbf{U}}^s \hat{\mathbf{\Lambda}}^s (\hat{\mathbf{U}}^s)^H, \tag{3.3.33}$$

where $\hat{\mathbf{\Lambda}}^{\nu}$ should be approximately zero. Similarly to (3.2.27), we define a cost function whose minima are located exactly at the source DOA's for exactly known statistics, and approximately for estimated (or approximately known) statistics:

$$\begin{aligned} J_{\text{ss}}(\theta) &\triangleq \sum_{\mathbf{u}\in\mathcal{U}^{\nu}} \left| \left\langle \mathbf{u},\ \mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right\rangle \right|^{2} \\ &= \sum_{\mathbf{u}\in\mathcal{U}^{\nu}} \left| \mathbf{u}^{H} \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right] \right|^{2} \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right]^{H} \left(\sum_{\mathbf{u}\in\mathcal{U}^{\nu}} \mathbf{u}\mathbf{u}^{H} \right) \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right] \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right]^{H} \mathbf{F}(\mathcal{U}^{\nu}) \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right], \end{aligned} \tag{3.3.34}$$

where $\mathbf{F}(\mathcal{U}^{\nu}) \triangleq \sum_{\mathbf{u}\in\mathcal{U}^{\nu}} \mathbf{u}\mathbf{u}^{H}$ is a (sub-)frame operator [123]. If the vectors $\mathbf{u} \in \mathcal{U}^{\nu}$ are mutually orthonormal, the frame operator reduces to the standard orthogonal projection operator $\mathbf{P}^{\nu}$ onto the noise subspace (see (3.2.17), and the next section). As in Section 3.2.1.5, an expression like $\mathbf{z}(\theta)^{H}\,\mathbf{F}(\mathcal{U}^{\nu})\,\mathbf{z}(\theta)$ or $\mathbf{z}(\theta)^{H}\,\mathbf{P}^{\nu}\,\mathbf{z}(\theta)$ in (3.3.34) will be called *null-spectrum*. The null-spectrum employed by Porat and Friedlander in [129] is defined as:

$$d(\theta) \triangleq \left\| \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^{*} \right]^{H} \hat{\mathbf{U}}^{\nu} \right\|^{2}, \tag{3.3.35}$$

which can be obtained from (3.3.34) by substituting the (ordered) set $\left\{\mathbf{u}^{S+1}, \ldots, \mathbf{u}^{(D)^2}\right\}$ for $\mathcal{U}^{\nu}$. The source DOA's are estimated by computing $d(\theta)$ over a suitably chosen grid and searching for the local minima (including interpolation at each local minimum). As in Section 3.2.2 here we are mainly interested in the way the null-spectrum in (3.3.34) is obtained. Therefore, as for the SOS case we will define a pseudo-spectrum based on the null-spectrum in Section 3.3.6.

If the estimate $\hat{\mathbf{C}}^{x}$ of $\mathbf{C}^{x}$ is known sufficiently accurately, i.e. if $\hat{\mathbf{C}}^{x} \approx \mathbf{C}^{x}$ (which usually implies that $\hat{\mathbf{U}}^{\nu} \approx \mathbf{U}^{\nu}$), the cost function in (3.3.34) is approximately zero for the true source DOA's (compare this with (3.2.28)):

$$J_{\text{ss}}(\theta)\big|_{\theta=\theta^{j}} \to 0 \qquad \forall\, 1 \le j \le S \quad \text{if} \quad \hat{\mathbf{C}}^{x} \to \mathbf{C}^{x}\,. \tag{3.3.36}$$

Hence, the source DOA's can be estimated by locating the minima or approximate zeros of (3.3.34). Obviously, a DOA estimator based on this principle is statistically consistent because $\hat{\mathbf{C}}^{x} \to \mathbf{C}^{x}$ when the number of samples used for its estimation goes to infinity. In the next section, we will demonstrate the theory by presenting a SPECTRAL-MUSIC-like algorithm. The (straightforward) extension of the HOS based subspace theory presented above to other algorithms such as ROOT-MUSIC and MIN-NORM is left to the reader.

3.3.6 SPECTRAL-MUSIC with HOS

Similarly to the (SOS-)SPECTRAL-MUSIC algorithm presented in Section 3.2.2, we can define a HOS-SPECTRAL-MUSIC pseudo-spectrum that is proportional to the reciprocal of (3.3.34) or (3.3.35). When exploiting the full zero subspace as in (3.3.35), the denominator of the pseudo-spectrum becomes (compare this with (3.2.29)):

$$\begin{aligned} J_{\text{ss}}(\theta) &\triangleq \sum_{i=S+1}^{(D)^2} \left| \left\langle \mathbf{u}^i,\ \mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^* \right\rangle \right|^2 \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H \left(\sum_{i=S+1}^{(D)^2} \mathbf{u}^i \big(\mathbf{u}^i\big)^H \right) \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right] \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H \hat{\mathbf{U}}^\nu (\hat{\mathbf{U}}^\nu)^H \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right] \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H \hat{\mathbf{P}}^\nu \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right] \\ &= \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H (\mathbf{I} - \hat{\mathbf{P}}^s) \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right] \overset{(3.3.35)}{=} d(\theta)\,, \end{aligned} \tag{3.3.37}$$

and the HOS-SPECTRAL-MUSIC pseudo-spectrum is defined by (compare this with (3.2.30)):

$$\begin{aligned} P_{\text{HSM}}(\theta) &\triangleq \frac{\left\| \mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^* \right\|^2}{J_{\text{ss}}(\theta)} \\ &= \frac{\left\| \mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^* \right\|^2}{\sum_{i=S+1}^{(D)^2} \left| \left\langle \mathbf{u}^i,\ \mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^* \right\rangle \right|^2} \\ &= \frac{(D)^2}{\left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H \hat{\mathbf{P}}^\nu \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]} \\ &= \frac{(D)^2}{\left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]^H (\mathbf{I} - \hat{\mathbf{P}}^s) \left[\mathbf{z}(\theta) \otimes \big(\mathbf{z}(\theta)\big)^*\right]} \qquad \forall\, \theta \in \Theta\,. \end{aligned} \tag{3.3.38}$$

Hence, instead of searching for the minima of $d(\theta)$, the maxima of the pseudo-spectrum defined in (3.3.38) are sought. As in Section 3.2.2, equations (3.3.34) and (3.3.36) imply that the denominator of $P_{\text{HSM}}(\theta)$ is approximately zero at the true source DOA's if $\hat{\mathbf{C}}^x \approx \mathbf{C}^x$. Hence, in this case the pseudo-spectrum exhibits sharp peaks in the vicinity of the source DOA's $\{\theta^j\}^{1 \le j \le S}$. Consequently, the source DOA's can be estimated by locating the S highest and/or sharpest peaks of the pseudo-spectrum. This also follows immediately by observing that locating the minima of $J_{\text{ss}}(\theta)$ is equivalent to locating the maxima of $1/J_{\text{ss}}(\theta)$. The different steps of the HOS-SPECTRAL-MUSIC algorithm are summarized in Alg. 3.5 on the next page. As before, it is formulated in terms of ideal statistics but is completely the same for estimated statistics. The reader is suggested to compare this algorithm with Alg. 3.1 on page 98.

Algorithm 3.5 HOS-SPECTRAL-MUSIC.

1: Compute/estimate sensor cumulant matrix $\mathbf{C}^x$*;*

2: Compute EVD of $\mathbf{C}^x$ *and split the result into signal and noise subspace parts:* $\mathbf{C}^x = \mathbf{U}^x\mathbf{\Lambda}^x(\mathbf{U}^x)^H = \mathbf{U}^s\mathbf{\Lambda}^s(\mathbf{U}^s)^H + \mathbf{U}^\nu\mathbf{\Lambda}^\nu(\mathbf{U}^\nu)^H$*;*

3: Compute matrix for projection onto noise subspace: $\mathbf{P}^\nu \triangleq \mathbf{U}^\nu(\mathbf{U}^\nu)^H$*;*

4: Evaluate HOS-SPECTRAL-MUSIC pseudo-spectrum on fine grid in θ*-domain:*

$$P_{\text{HSM}}(\theta) = \frac{(D)^2}{\Big[\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^*\Big]^H \mathbf{P}^\nu \Big[\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^*\Big]} \qquad \forall\,\theta\in\Theta\,;$$

5: Determine source DOA's by locating the S *sharpest peaks of* $P_{\text{HSM}}(\theta)$*, e.g. by grid search method.*

Example and discussion

Now, we will give an example of HOS-SPECTRAL-MUSIC for a ULA with sensor spacing $d = \frac{\lambda_c}{2}$ (see Section 3.1.2). From (3.2.31), it follows that the generalized array response vector $\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^*$ is given by:

$$\mathbf{z}(\theta)\otimes\big(\mathbf{z}(\theta)\big)^* = \begin{bmatrix} 1 \\ \exp\big(\jmath\,\pi\sin(\theta)\big) \\ \vdots \\ \exp\big(\jmath\,(D-1)\pi\sin(\theta)\big)\end{bmatrix} \otimes \begin{bmatrix} 1 \\ \exp\big(-\jmath\,\pi\sin(\theta)\big) \\ \vdots \\ \exp\big(-\jmath\,(D-1)\pi\sin(\theta)\big)\end{bmatrix} \quad \forall\,\theta\in\Theta\,, \tag{3.3.39}$$

which similarly to $\mathbf{z}(\theta)$ itself only contains phase factors. As we have alluded to in Section 3.3.1, the null-spectrum $d(\theta)$ employed by Porat and Friedlander (3.3.35), and consequently also the pseudo-spectrum $P_{\text{HSM}}(\theta)$ in (3.3.38), can also 'discriminate' the DOA's of statistically independent sources if there are more sources than sensors, i.e. $S > D$, up to a certain maximum. Using a technique similar to the one developed in Section 7.2.3, it can be proven that this maximum is given by $S_{\max} = 2(D-1)$; in fact (7.2.16) applies. Since Porat and Friedlander assume that $D > S$ in [129], they have not noticed this. In the following example, we will demonstrate the ability of Alg. 3.5 of estimating more source DOA's than sensors by applying it to a scenario with $S = 4$ sources and $D = 3$ sensors. The DOA's of the four sources are given by $\tilde{\boldsymbol{\theta}} = [-38, 7, 19, 81\,]$ degrees. The signals and system are generated in such a way that the assumptions listed in Section 3.3.1 are satisfied; in the next paragraph, some specifics will be highlighted. As in Section 3.2.2, two pseudo-spectra are computed for the considered scenario, viz. one using the ideal sensor cumulant matrix $\mathbf{C}^x$ computed using e.g. (3.3.13) with ideally known source DOA's and auto-cumulants, and the other using its estimate $\hat{\mathbf{C}}^x$ obtained from $T = 40000$ samples of $\mathbf{x}$.

Since we want to exploit HOS, we cannot use complex circular Gaussian source signals as in Section 3.2.2 because such signals have zero higher order cumulants. Instead, we use signals which we will call *complex circular uniform* and satisfy the 'complex circularity requirements' $E\{\mathbf{s}[m]\mathbf{s}[n]^H\} = \delta_{mn}(\sigma^s)^2\,\mathbf{I}$ and $E\{\mathbf{s}[m]\mathbf{s}[n]^T\} = \mathbf{0}$, where δ_{mn} denotes a Kronecker delta function and $(\sigma^s)^2$ the variance of the signals. For example, a complex circular uniform source signal $s[n] = s_{\text{r}}[n] + \jmath s_{\text{i}}[n]$ can be constructed from two real-valued

statistically independent stationary white uniformly distributed sequences $s_{\rm r}[n]$ and $s_{\rm i}[n]$, which represent the real and imaginary parts of $s[n]$ respectively, with the same variance. The signals $s_{\rm r}[n]$ and $s_{\rm i}[n]$ are drawn from the same probability density function (pdf), say $p_v(v)$ with $v = s_{\rm r}[n]$ and $v = s_{\rm i}[n]$ respectively, which is given by:

$$p_v(v) \triangleq \begin{cases} \frac{1}{2b}, & -b \leq v \leq b; \\ 0, & \text{elsewhere}, \end{cases} \tag{3.3.40}$$

where b is a real-valued positive number. From (B.1.12) it follows that the second and fourth order moments of v are given by $E\{(v)^2\} = \frac{1}{3}(b)^2$ and $E\{(v)^4\} = \frac{1}{5}(b)^4$ respectively. Using (B.1.26d), it then follows that the variance and fourth order cumulant of v are given by $(\sigma^v)^2 = \kappa^{v,2} = E\{(v)^2\} = \frac{1}{3}(b)^2$ and $\kappa^{v,4} = E\{(v)^4\} - 3\left(E\{(v)^2\}\right)^2 = -\frac{2}{15}(b)^4$ respectively. Hence, the fourth order cumulant of a real-valued uniform random variable with parameter b is always negative. In general, real-valued random variables having negative kurtosis are called *platykurtic* or *Sub-Gaussian*; see Sections B.1.3.3 and B.1.3.4. In order to compute the cumulant of a complex signal, say $w = w_{\rm r} + \jmath w_{\rm i}$, we need to combine the statistics of its real and imaginary parts (see Appendix B). These statistics are described completely by the jpdf $p_{w_{\rm r},w_{\rm i}}(w_{\rm r}, w_{\rm i})$ of $w_{\rm r}$ and $w_{\rm i}$, which is the product of the individual pdf's $p_{w_{\rm r}}(w_{\rm r})$ and $p_{w_{\rm i}}(w_{\rm i})$ because $w_{\rm r}$ and $w_{\rm i}$ are now assumed to be statistically independent:

$$p_{w_{\rm r},w_{\rm i}}(w_{\rm r}, w_{\rm i}) = p_{w_{\rm r}}(w_{\rm r})\, p_{w_{\rm i}}(w_{\rm i})\,. \tag{3.3.41}$$

Substituting the pdf in (3.3.40) for $p_{w_{\rm r}}(w_{\rm r})$ and $p_{w_{\rm i}}(w_{\rm i})$ into (3.3.41) yields the following:

$$p_{w_{\rm r},w_{\rm i}}(w_{\rm r}, w_{\rm i}) = \begin{cases} \frac{1}{4(b)^2}, & -b \leq w_{\rm r}, w_{\rm i} \leq b; \\ 0, & \text{elsewhere}. \end{cases} \tag{3.3.42}$$

This jpdf can be used to compute the statistics of the complex random variable w. The variance of w equals $(\sigma^w)^2 \triangleq E\{w\,(w)^*\} = E\{(w_{\rm r}+\jmath w_{\rm i})(w_{\rm r}-\jmath w_{\rm i})\} = (\sigma^{w_{\rm r}})^2 + (\sigma^{w_{\rm i}})^2 = \frac{2}{3}(b)^2$. Note that $E\{(w)^2\} = 0$. Using (B.2.44d) with $v_{i_1} = w$, $v_{i_2} = (w)^*$, $v_{i_3} = (w)^*$, $v_{i_4} = w$, the fourth order cumulant of w with conjugation tuple $\mathbf{c}_4 = (\circ, *, *, \circ)$ is computed as follows:

$$\begin{aligned} \kappa_{\mathbf{i}_4}^{w,\mathbf{c}_4} &\triangleq \operatorname{cum}\left(w, (w)^*, (w)^*, w\right) \\ &= E\{w\,(w)^*(w)^* w\} - 2E\{w\,(w)^*\}E\{(w)^* w\} - E\{(w)^2\}E\{\left((w)^*\right)^2\} \\ &= E\{|w|^4\} - 2\big(E\{|w|^2\}\big)^2 = E\{(w_{\rm r})^4 + 2(w_{\rm r})^2(w_{\rm i})^2 + (w_{\rm i})^4\} - 2\big((\sigma^{w_{\rm r}})^2 + (\sigma^{w_{\rm i}})^2\big)^2 \\ &= \tfrac{1}{5}(b)^4 + 2\tfrac{1}{3}(b)^2\tfrac{1}{3}(b)^2 + \tfrac{1}{5}(b)^4 - 2\big(\tfrac{2}{3}(b)^2\big)^2 = -\tfrac{4}{15}(b)^4\,. \end{aligned} \tag{3.3.43}$$

Hence, similarly to the real-valued case, the fourth order cumulant of a complex-valued circular uniformly distributed random variable with parameter b is always real-valued and negative. Instead of fixing the variance of each source signal to one, as we did in the examples presented in Section 3.2.2, in the current example we fix the kurtosis of each source signal to -1. Hence, we consider source signals with 'unit absolute kurtosis'. This implies that $b = \sqrt[4]{15/4} \approx 1.392$ and $(\sigma^s)^2 = \sqrt{15/9} \approx 1.291$, and thus $\sigma^s = \sqrt[4]{15/9} \approx 1.136$.

Now that we have discussed the source signals and statistics, we briefly turn to the noise signals and statistics. According to assumption *AS5* on page 114, the noise signals are assumed to be realizations of complex circular Gaussian stochastic processes satisfying the requirements $E\{\boldsymbol{\nu}[m]\boldsymbol{\nu}[n]^H\} = \delta_{mn}(\sigma^\nu)^2\,\mathbf{I}$ and $E\{\boldsymbol{\nu}[m]\boldsymbol{\nu}[n]^T\} = \mathbf{0}$. In Section 3.2.2 we

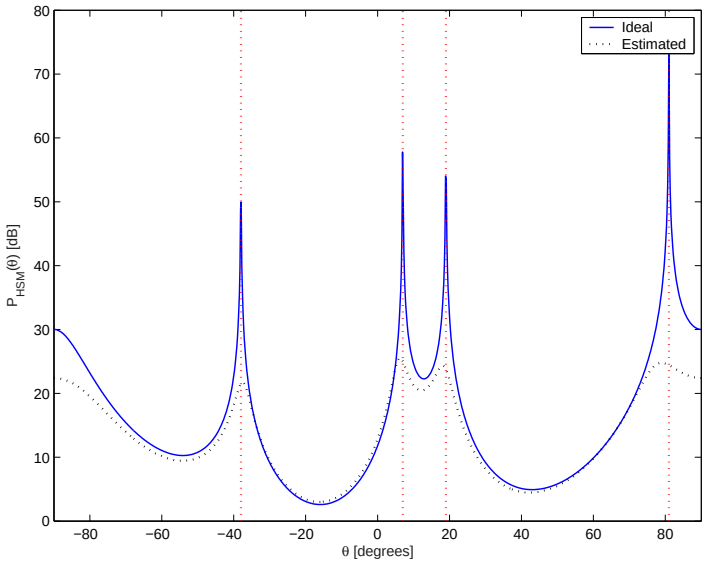

Figure 3.10: *Ideal and estimated HOS-SPECTRAL-MUSIC spectra* $P_{\text{HSM}}(\theta)$ *for* $D = 3$ *and* $S = 4$.

have discussed how such signals can be generated. Here, we only note that we set the noise variance equal to the source variance, i.e. $(\sigma^\nu)^2 = (\sigma^s)^2 = \sqrt{15/9}$ in order to have the same input SNR as in the examples of Section 3.2.2, viz. 0 dB.

The pseudo-spectra for this example are computed using Alg. 3.5. Firstly, $\mathbf{C}^x$ ($\hat{\mathbf{C}}^x$) is computed/estimated. The estimated sensor cumulants are obtained by replacing the mathematical expectation operators in (3.3.6) by their counterparts obtained by time-averaging over a certain window length. After the computation of the EVD of $\mathbf{C}^x$, the $(D)^2 - S$ eigenvectors corresponding to the $(D)^2 - S$ smallest eigenvalues are used to construct $\mathbf{U}^\nu$ and the noise projection matrix $\mathbf{P}^\nu$ according to (3.2.17). Finally, the pseudo-spectra are obtained by evaluating (3.3.38) on a fine grid in the θ-domain for $\theta \in \Theta$, where $\Theta \triangleq [-\pi/2, \pi/2]$ in radians. The resulting spectra in dB are plotted as a function of the DOA θ in degrees in Figure 3.10. As expected, the ideal spectrum exhibits very sharp peaks at the source DOA's, whereas the estimated spectrum exhibits peaks that are less sharp and deviate slightly from their ideal positions. Both spectra do not contain spurious peaks. Hence, the (estimated) source DOA's can be determined uniquely from the pseudo-spectra. See also the desired properties of a pseudo-spectrum listed on page 99 in Section 3.2.2. Note that since the estimated pseudo-spectrum is statistically consistent, it converges to the ideal one when the number of samples used for the estimation of the cumulants goes to infinity. Finally, comparing this example to the examples in Section 3.2, we note that a disadvantage of using HOS is that reliable estimation of higher order moments and cumulants requires many more samples than for SOS. According to Brillinger [22], the sample size needed for the estimation of the l-th order statistics of a stochastic process, under the constraint of prescribed estimation bias and variance, increases almost exponentially with l. Another disadvantage of HOS is that they can be very sensitive to outliers in the data. Since our main purpose here was to provide insight that can be used for generalizing subspace techniques to HOS and in several other directions, we circumvented these problems by using Sub-Gaussian data.

3.4 Retrospective analysis

In this section, we abstract and emphasize the key assumptions, conditions and principles of the subspace techniques explained in the previous sections, and generalize them in several directions; see Alg. 3.6 on page 129. Subspace methods for exploiting any type of statistical variability in the data, such as arbitrary order lag-zero or temporal statistical diversity, can be developed based on a properly defined matrix, say $\mathbf{C}^x$, containing the considered sensor statistics arranged in a specific manner. In the sequel, we will refer to this matrix of sensor statistics as *'subspace matrix'*. Basically, all subspace methods rely on the same set of assumptions that we present in the following list for a single statistical order. Although most of these assumptions and the algorithms that are based on them can be generalized to a mixture of several statistical orders, for the sake of clarity we formulate the assumptions for a fixed but arbitrary order l. Furthermore, we will express ourselves in terms of *generalized* array response vectors, matrices, manifolds, and so on. We start by making an inventory of the fundamental assumptions underlying any subspace method (explanatory comments are provided after the list):

MSA1: *The S source signals $s_1[n], \ldots, s_S[n]$ are realizations of zero mean random processes possessing l-th order statistics/cumulants with sufficient diversity;*

MSA2: *All l-th order source cross-cumulants are zero, whereas all l-th order source auto-cumulants are non-zero;*

MSA3: *The sets $\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S$ of source position parameters are different;*

MSA4: *The source and sensor noise signals are mutually statistically independent;*

MSA5: *The sensor noise signals have a simpler structure than the source signals (in the sense made precise for instance in Section 1.2.5);*

MSA6: *The parameterization of the generalized array manifold $\boldsymbol{\mathcal{A}}_{\mathrm{g}}(\boldsymbol{\Upsilon})$, i.e. the functional dependence of the generalized array response vector $\mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon})$ on the set of source position parameters $\boldsymbol{\upsilon}$ for the parameter space of interest $\boldsymbol{\Upsilon}$ is one-to-one and known a priori;*

MSA7: *The number of sources S is smaller than a certain maximum $S_{\max}$, which is smaller than the number of rows of the generalized array response matrix $\mathbf{A}_{\mathrm{g}}\big(\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S\big)$ of $\mathbf{A}\big(\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S\big)$; $S_{\max}$ primarily depends on the diversity of the source signals, array geometry, structure of the subspace matrix, number of sensors, conjugation tuple, and order of the considered statistics;*

MSA8: *The generalized array response matrix $\mathbf{A}_{\mathrm{g}}\big(\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S\big)$ is unambiguous over the parameter space of interest, meaning that the source position parameters $\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S$ can be determined uniquely from the intersections between the column range $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{g}}\big(\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S\big)\big)$ and the generalized array manifold $\boldsymbol{\mathcal{A}}_{\mathrm{g}}(\boldsymbol{\Upsilon})$.*

The abbreviation '*MSA*' stands for '<u>M</u>ain <u>S</u>ubspace <u>A</u>ssumption'. We have used the upsilon symbol $\boldsymbol{\upsilon}$ for denoting the set $\boldsymbol{\upsilon} \triangleq \{\upsilon_1, \upsilon_2, \ldots\}$ of parameters that parameterize a certain source position. For example, in the three dimensional near field source localization scenario, often a spherical coordinate system with parameters θ, ζ and ρ is used, and thus $\boldsymbol{\upsilon} \triangleq \{\theta, \zeta, \rho\}$ for this case (see also Section 1.1.3). As can be verified by the reader, the need for most assumptions listed above is clear from the examples of SOS and HOS subspace

methods presented in the previous sections. Therefore, we restrict ourselves to emphasizing some important implications and relationships later in this section when we summarize the rationale behind a general subspace method that exploits these assumptions. Firstly, in Alg. 3.6 on the facing page we present a general *'algorithm for producing subspace based algorithms'* that can exploit any kind of sensor statistics. For simplicity we only consider subspace matrices constructed from noise-free sensor statistics. Although we only show the 'MUSIC-version' of the algorithm, the reader should realize that it can be generalized or adapted straightforwardly to algorithms like ROOT-MUSIC and Min-Norm. Step 1 of the algorithm is very important. Here, information about the signal scenario and array geometry is used to design a certain suitable arrangement of sensor statistics in a subspace matrix $\mathbf{C}^x$ that satisfies the listed properties. Obviously, the condition that $\mathbf{C}^x$ can be expressed in a 'structured way' in the S generalized source array response vectors $\mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}^1), \ldots, \mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}^S)$ is necessary in order to have an exploitable relation between the different subspaces[‡] of $\mathbf{C}^x$ and the generalized source array response vectors (for example, see Sections 3.2.1.4 and 3.3.4), and thus a link between the observable subspaces and the sets $\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S$ of source position parameters that have to be estimated from the sensor data. Such a 'structured manner' is made possible due to assumptions *MSA1-MSA5*, and the involved fundamental 'link' between the observable subspaces and the sets of source position parameters is ensured to be unique due to assumptions *MSA6-MSA8*.

For examples of the kind of expressions alluded to in the first 'sub-condition' of Step 1, see (3.2.1) and (3.3.11). The second 'sub-condition' of Step 1 is needed because the subspace matrix must be able to represent all information about the S different generalized source array response vectors associated with the sets $\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S$. Hence, in general (also in the noisy case) the 'source part of $\mathbf{C}^x$' should have rank S. If this were not the case, essential information about the generalized source array response vectors would not be present in (the subspaces of) $\mathbf{C}^x$. Finally, as is evident from Sections 3.2.1.4 and 3.3.4, and in fact from the whole chapter, one of the conditions in the third 'sub-condition' of Step 1 forms the core on which Alg. 3.6 is based. By virtue of this condition, a system of equations in the unknown parameters can be obtained. Note that the left null/noise subspace $\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x)$ employed in (3.4.1) is isomorphic to the orthogonal complement $\big(\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x)\big)^{\perp}$ of the column space $\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x)$, which can be formulated as $\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x) \doteqdot \big(\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x)\big)^{\perp}$ or $\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x) = \big(\mathcal{R}_{\mathrm{c}}(\mathbf{C}^x)\big)^{\perp}\big)^{T}$[‡]. Likewise, the right null/noise subspace $\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x)$ employed in (3.4.2) is isomorphic to the orthogonal complement $\big(\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x)\big)^{\perp}$ of the row space $\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x)$, which can be formulated as $\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x) \doteqdot \big(\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x)\big)^{\perp}$ or $\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x) = \big(\big(\mathcal{R}_{\mathrm{r}}(\mathbf{C}^x)\big)^{\perp}\big)^{T}$[‡]. Obviously, in order to exploit (3.4.1) or (3.4.2), the involved null space must be non-trivial. This implies that if the left null space of $\mathbf{C}^x$ is employed, the number of rows of $\mathbf{C}^x$ must be larger than S. Likewise, if the right null space is employed, the number of columns of $\mathbf{C}^x$ must be larger than S. Note that both the left and right null spaces of $\mathbf{C}^x$ can be computed by means of the Singular Value Decomposition (SVD) (see Appendix D). See steps 3 and 4 of Alg. 3.6. Note that in the examples presented earlier in this chapter, the left and right null spaces were equal because the considered matrices were Hermitian. However, this is not necessary in general.

[‡] See properties *FSP1–FSP6* on page 450 of the fundamental subspaces considered in linear algebra.

Algorithm 3.6 Key ingredients of a general subspace algorithm.

1: Construct a subspace matrix $\mathbf{C}^x$ of size N_{r}^x by N_{c}^x, containing any kind of noise-free sensor statistics arranged in such a way that:

- *It can be expressed in S generalized source array response vectors $\mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}^1), \ldots, \mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}^S)$ in a 'structured manner';*
- *It has rank S if employed sensor statistics are noise-free;*
- *The generalized source array response vectors are orthogonal to (transpose of) left null space $\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x)$ of $\mathbf{C}^x$:*

$$\left\langle \mathbf{u}, \mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}) \right\rangle\big|_{\boldsymbol{\upsilon}=\boldsymbol{\upsilon}^j} = 0 \qquad \forall\, 1 \leq j \leq S, \quad \forall\, \mathbf{u} \in \left(\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x)\right)^T. \tag{3.4.1}$$

Alternatively, they are orthogonal to the right null space $\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x)$ of $\mathbf{C}^x$:

$$\left\langle \mathbf{v}, \mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon}) \right\rangle\big|_{\boldsymbol{\upsilon}=\boldsymbol{\upsilon}^j} = 0 \qquad \forall\, 1 \leq j \leq S, \quad \forall\, \mathbf{v} \in \mathcal{N}_{\mathrm{r}}(\mathbf{C}^x)\,; \tag{3.4.2}$$

2: Compute/estimate the subspace matrix $\mathbf{C}^x$;

3: Compute the Singular Value Decomposition of $\mathbf{C}^x$ and split the result into signal and null/noise subspace parts as follows:

$$\mathbf{C}^x = \mathbf{U}^x \boldsymbol{\Lambda}^x (\mathbf{V}^x)^* = \mathbf{U}^s \boldsymbol{\Lambda}^s (\mathbf{V}^s)^* + \mathbf{U}^\nu \boldsymbol{\Lambda}^\nu (\mathbf{V}^\nu)^* = \mathbf{U}^s \boldsymbol{\Lambda}^s (\mathbf{V}^s)^*; \tag{3.4.3}$$

4: If $\mathbf{C}^x$ is constructed such that (3.4.1) *holds, compute matrix for projection onto transpose $\left(\mathcal{N}_{\mathrm{l}}(\mathbf{C}^x)\right)^T = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right)$ of left null/noise space of $\mathbf{C}^x$:*

$$\mathbf{P}^\nu \triangleq \mathbf{U}^\nu (\mathbf{U}^\nu)^H; \tag{3.4.4}$$

Alternatively, if $\mathbf{C}^x$ is constructed such that (3.4.2) *holds, compute matrix for projection onto right null/noise space $\mathcal{N}_{\mathrm{r}}(\mathbf{C}^x) = \mathcal{R}_{\mathrm{c}}\left((\mathbf{V}^\nu)^T\right)$ of $\mathbf{C}^x$:*

$$\mathbf{P}^\nu \triangleq (\mathbf{V}^\nu)^T (\mathbf{V}^\nu)^*\,; \tag{3.4.5}$$

5: Evaluate the following SPECTRAL-MUSIC pseudo-spectrum on fine grid in the source position parameter domain, i.e. the $\boldsymbol{\upsilon}$-domain (F is a normalization factor):

$$P(\boldsymbol{\upsilon}) = \frac{F}{\mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon})^H \mathbf{P}^\nu \mathbf{z}_{\mathrm{g}}(\boldsymbol{\upsilon})} \qquad \forall\, \boldsymbol{\upsilon} \in \boldsymbol{\Upsilon}\,; \tag{3.4.6}$$

6: Determine source position parameters $\boldsymbol{\upsilon}^1, \ldots, \boldsymbol{\upsilon}^S$ by locating the S sharpest peaks of $P(\boldsymbol{\upsilon})$, e.g. by grid search method.

3.5 Conclusions and discussion

In this chapter we have explained the rationale behind subspace methods for the semi-blind Direction Of Arrival (DOA) estimation problem in such a way that a unifying overview of the field is furnished, thereby providing enhanced insight and allowing natural generalizations in several directions. Our main reasons for doing this are that this rationale is of paramount importance for the blind identification method that we will develop in the chapters ahead, and that a coherent and rather complete overview of this research area is not available in the literature in a single place. The subspace approach is commonly adopted for solving semi-blind array signal processing problems, such as the DOA estimation problem that was used as a framework in this chapter, but we will show in the following chapters that it is much more general and can equally well be applied to fully blind identification problems. We have abstracted and emphasized the key ideas employed by subspace methods in a retrospective analysis. Among other things, these ideas encompass the application of subspace decomposition principles to a properly defined subspace matrix under the conditions listed as *MSA1-MSA8* on page 127. Several requirements on the properties of the subspace matrix were made explicit. Furthermore, we have argued that the main differences between different subspace approaches and/or algorithms concern the manner in which the subspace matrix is formed and exploited. Since the specific choice for arranging the sensor statistics in this matrix has a large influence on the suitability for a certain signal scenario, the maximum number of DOA's that can be identified, the computational 'convenience' and complexity of a certain algorithm, and so on, the choice of a particular subspace matrix is of crucial importance. Note that in the literature usually a well-known fixed arrangement is considered and no attention is paid to this issue. The key ideas referred to above have also been summarized in the form of an algorithm that highlights the main steps of any subspace based algorithm and in fact is a general algorithm for designing subspace algorithms (see Alg. 3.6 on the previous page). One of our contributions to the subspace research area is that we have formulated this algorithm in such a way that it is unifying, generally applicable, and provides a lot of insight. The techniques and generalizations that we have presented in this chapter, as well as the notation used to describe them, will be employed intensively in the rest of the thesis. In the next chapters, we will show how subspace techniques exploiting temporal structure can be developed for solving fully blind MIBI problems.

CHAPTER 4

MIBI based on Second Order Temporal Structure: Real-valued 2×2 Mixing Case

This chapter introduces the rationale behind using subspace techniques for exploiting the Second Order Temporal Structure (SOTS) in the data for performing Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI). We focuss on a specific kind of temporal structure, viz. second-order non-whiteness. In later chapters we also consider non-stationarity, and the simultaneous exploitation of non-whiteness and non-stationarity. Our method is based on subspace techniques and the construction of a 'subspace matrix' that contains certain sensor correlation values for different lags in a specific arrangement. We have stressed already several times the importance of the specific arrangement of sensor statistics in a subspace matrix. In this chapter we will see an example of this. Based on certain assumptions about the second order temporal structure of the source and noise signals, we show that if the SOTS is exploited in a specific way the MIBI problem can be 'projected onto' two complementary or dual mathematical problems by applying subspace techniques to the subspace matrix. In the first problem, a well-structured system of homogeneous polynomial equations of degree two has to be solved, whereas in the second one a multi-matrix generalized eigenvalue problem has to be solved. We describe several possible methods for the solution of these problems. Furthermore, we highlight the geometric and algebraic structure of both problem formulations. In this chapter we consider the relatively simple real-valued mixing case with two sources and two sensors (see Fig. 4.1) because this allows us to clearly point out and emphasize the key ideas of our method along with the explanations. Furthermore, we use the same notation as in the previous chapters. At first sight, our notation and problem formulation might seem much more convoluted than necessary for the problem considered in this chapter. However, the reason for using them is that they make the generalizations presented in later chapters much easier, and in a sense even trivial. It should be kept in mind that the notation adopted in this work has been set up in such a way that it allows a uniform description of the theory for the general MIBI problem.

The outline of this chapter is as follows. Section 4.1 describes the structure of the 2×2 MIBI model, whereas Section 4.2 explains the model assumptions. Next, in Section 4.3 a system of homogeneous polynomial equations of degree two satisfied by the columns of the mixing matrix is derived, thereby highlighting the algebraic structure of our problem formulation. After that, in Section 4.4 the geometric structure associated with this system of equations is investigated and examples are given. Subsequently, in Section 4.5 methods for solving the obtained system are presented. Then, in Section 4.6 the MIBI problem is written as a multi-matrix generalized eigenvalue problem, and a solution method is given. Finally, conclusions and discussion are presented in Section 4.7. In later chapters, we will show that all the key assumptions, ideas and principles of the presented method can be generalized in several directions such as w.r.t. the type of temporal statistical variability in the data, the

order of the exploited statistics, other sizes of the mixing matrix, etc. In order to keep a clear overview of the main theme of this chapter, we summarize the key ingredients of the resulting two by two MIBI algorithm in the following high-level algorithm:

Algorithm 4.1 Pre-overview of the 2×2 MIBI method developed in this chapter.

1: *Compute/estimate the sensor correlation functions for several 'noise-free' lags;*

2: *Arrange these values in a subspace matrix* $\mathbf{C}^x$*;*

3: *Compute the Singular Value Decomposition (SVD) of* $\mathbf{C}^x$ *and split the result into signal and null/noise subspace parts;*

4: *Choose one of the following two MIBI problem 'projections':*

- *Construct a system of homogeneous polynomial equations of degree two, whose solutions are the columns of the mixing matrix;*
- *Construct a multi-matrix generalized eigenvalue problem, whose generalized eigenvectors are directly related to the columns of the mixing matrix;*

5: *Solve the problem defined in the previous step.*

4.1 MIBI model structure

In this chapter, we consider an instantaneous mixing model with two source and two sensor signals as depicted in Fig. 4.1. According to (1.1.1), the i-th sensor signal $x_i[n]$ is given by:

$$\begin{aligned} x_i[n] &= \sum_{j=1}^{2} a_i^j s_j[n] + \nu_i[n] \\ &= a_i^1 s_1[n] + a_i^2 s_2[n] + \nu_i[n] \qquad \forall\, n \in \mathbb{Z}, \quad \forall\, 1 \leq i \leq 2\,, \end{aligned} \tag{4.1.1}$$

where a_i^j is the instantaneous transfer coefficient from the j-th source to the i-th sensor, $s_j[n]$ is the j-th source signal at discrete time n, and $\nu_i[n]$ is the i-th noise signal at discrete time n. The system and signals are assumed to be real-valued. Moreover, it is assumed that all involved signals are zero-mean and stationary. The zero-mean assumption is made solely for

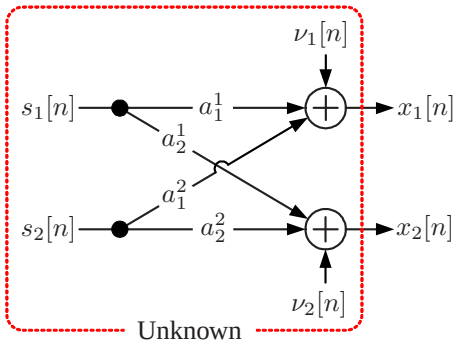

Figure 4.1: *2×2 MIBI problem setup.*

mathematical convenience and does not imply any loss of generality. In case non-zero-mean signals are involved, correlations simply have to be replaced by covariances. Because all signals are (assumed to be) stationary, all involved correlation functions depend on a discrete lag k only. In matrix-vector notation, (4.1.1) can be written as (see also (1.1.1)):

$$\mathbf{x}[n] = \sum_{j=1}^{2} \mathbf{a}^j \, s_j[n] + \boldsymbol{\nu}[n] = \mathbf{A}\mathbf{s}[n] + \boldsymbol{\nu}[n] \qquad \forall \, n \in \mathbb{Z}\,, \tag{4.1.2}$$

where:

$$\mathbf{x}[n] \triangleq \begin{bmatrix} x_1[n] \\ x_2[n] \end{bmatrix}, \quad \mathbf{s}[n] \triangleq \begin{bmatrix} s_1[n] \\ s_2[n] \end{bmatrix}, \quad \boldsymbol{\nu}[n] \triangleq \begin{bmatrix} \nu_1[n] \\ \nu_2[n] \end{bmatrix}, \quad \text{and} \quad \mathbf{a}^j \triangleq \begin{bmatrix} a_1^j \\ a_2^j \end{bmatrix}$$

are column vectors of sensor signals, source signals, noise signals and mixing elements respectively that are all elements of $\mathbb{R}_2$. The mixing matrix $\mathbf{A}$ is an element of $\mathbb{R}_2^2$ (see Appendix A), and can be written in terms of its columns as $\mathbf{A} = \begin{bmatrix} \mathbf{a}^1 & \mathbf{a}^2 \end{bmatrix}$ and in terms of its elements as $\mathbf{A} = \begin{bmatrix} a_1^1 & a_1^2 \\ a_2^1 & a_2^2 \end{bmatrix}$. In this work, following the terminology in previous chapters the columns of the mixing matrix are sometimes also called 'array response vectors', and the mixing matrix 'array response matrix'. For completeness and clarity of the following developments, (4.1.2) is written out entirely in terms of the individual quantities:

$$\begin{aligned} \begin{bmatrix} x_1[n] \\ x_2[n] \end{bmatrix} &= \begin{bmatrix} a_1^1 \\ a_2^1 \end{bmatrix} s_1[n] + \begin{bmatrix} a_1^2 \\ a_2^2 \end{bmatrix} s_2[n] + \begin{bmatrix} \nu_1[n] \\ \nu_2[n] \end{bmatrix} \\ &= \begin{bmatrix} a_1^1 & a_1^2 \\ a_2^1 & a_2^2 \end{bmatrix} \begin{bmatrix} s_1[n] \\ s_2[n] \end{bmatrix} + \begin{bmatrix} \nu_1[n] \\ \nu_2[n] \end{bmatrix} \qquad \forall \, n \in \mathbb{Z}\,. \end{aligned}$$

Indeterminacies in instantaneous blind identification

As we have shown in Section 2.4, two indeterminacies or ambiguities are involved in instantaneous blind identification that cannot be resolved without any prior knowledge. For the simple two by two model these are particularly clear. Firstly, from (4.1.2) it is clear that permuting the columns of $\mathbf{A}$ together with an equal permutation of the source signals in $\mathbf{s}[n]$ still yields the same vector of sensor signals $\mathbf{x}[n]$:

$$\mathbf{x}[n] = \big(\mathbf{a}^1 \, s_1[n] + \mathbf{a}^2 \, s_2[n]\big) + \boldsymbol{\nu}[n] = \big(\mathbf{a}^2 \, s_2[n] + \mathbf{a}^1 \, s_1[n]\big) + \boldsymbol{\nu}[n]\,. \tag{4.1.3}$$

This means that the order of the columns, respectively sources, cannot be determined. Secondly, as is also clear from (4.1.2), scaling the columns of $\mathbf{A}$ by arbitrary real-valued scalars, say α^1 and α^2, together with a corresponding inverse scaling of the source signals in $\mathbf{s}[n]$, again yields the same vector of sensor signals $\mathbf{x}[n]$:

$$\mathbf{x}[n] = \big(\mathbf{a}^1 \, s_1[n] + \mathbf{a}^2 \, s_2[n]\big) + \boldsymbol{\nu}[n] = \left(\frac{\mathbf{a}^1}{\alpha^1}\big(\alpha^1 s_1[n]\big) + \frac{\mathbf{a}^2}{\alpha^2}\big(\alpha^2 s_2[n]\big)\right) + \boldsymbol{\nu}[n]\,. \tag{4.1.4}$$

This means that the columns and source signals can only be recovered up to a scale factor. Taking into account these two indeterminacies, the goal of MIBI is to recover the columns of the mixing system in arbitrary order and with arbitrary non-zero (Euclidian) norms. Usually, these two indeterminacies do not cause serious problems because for many applications the most relevant information is in the 'directions' of the columns rather than in their order or magnitudes.

4.2 MIBI model assumptions

In order to be able to exploit the temporal structure in the data, several assumptions need to be made. For the current example, for demonstration purposes we will make specific assumptions on the source, noise, and joint source-noise Second Order Temporal Structure (SOTS). Later on, these assumptions will be generalized in several ways. The SOTS is represented mathematically by correlation functions. Hence, the second order 'temporal cross-structure' of two signals is represented by their cross-correlation function, whereas the second order 'temporal auto-structure' of an individual signal is represented by its auto-correlation function. As is usual in the development of theoretical concepts involving stochastic signals, correlations are (at least initially) expressed in terms of mathematical expectations, i.e. ensemble averages, as opposed to time-averaged correlations that have to be used in practical applications. For the sake of simplicity it is assumed here that the involved signals possess the appropriate form of ergodicity, thereby ensuring that time-averaged quantities converge to their corresponding ensemble-averaged counterparts if the available amount of data increases. We start by listing the definitions of the four correlation functions that are needed for the formulation of the required assumptions and derivation of the theory for the current scenario, viz. the source, noise, source-noise, and noise-source correlation functions. After that we define the various Regions Of Support (ROS) on which these functions are defined. Finally we discuss and summarize the essence of the various assumptions.

The source correlation functions are defined as follows:

$$r^{s}_{j_1 j_2}[k] \triangleq E\left\{s_{j_1}[n] s_{j_2}[n-k]\right\} = r^{s}_{j_2 j_1}[-k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq j_1, j_2 \leq 2\,, \tag{4.2.1}$$

the noise correlation functions as:

$$r^{\nu}_{i_1 i_2}[k] \triangleq E\left\{\nu_{i_1}[n] \nu_{i_2}[n-k]\right\} = r^{\nu}_{i_2 i_1}[-k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq i_1, i_2 \leq 2\,, \tag{4.2.2}$$

the source-noise correlation functions by:

$$r^{s\nu}_{ji}[k] \triangleq E\left\{s_{j}[n] \nu_{i}[n-k]\right\} = r^{\nu s}_{ij}[-k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq i, j \leq 2\,, \tag{4.2.3}$$

and finally the noise-source correlation functions are defined by:

$$r^{\nu s}_{ij}[k] \triangleq E\left\{\nu_{i}[n] s_{j}[n-k]\right\} = r^{s\nu}_{ji}[-k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq i, j \leq 2\,. \tag{4.2.4}$$

To clearly and unambiguously formulate the assumptions, we need to make some definitions regarding the Regions Of Support of the various correlation functions, see Fig. 4.2 on the facing page. Among other things, these definitions are necessary for specifying explicitly what conditions are required on the temporal structure/diversity of the source and noise signals, and in what sense the noise signals have a 'simpler' temporal structure than the source signals. To start with, let $\Omega^{s}_{k} \subseteq \mathbb{Z}$ be some ROS in the domain of lags k on which the source correlation functions $r^{s}_{11}[k]$, $r^{s}_{12}[k]$, $r^{s}_{21}[k]$ and $r^{s}_{22}[k]$ generally are non-zero. This *'Source Region Of Support'* is indicated schematically by the set of all pentagons, i.e. black and white, in Fig. 4.2. The source correlation functions may also be non-zero outside this ROS, but this is not necessary. Likewise, let $\Omega^{\nu}_{k} \subseteq \mathbb{Z}$ be some ROS on which the noise correlation functions $r^{\nu}_{11}[k]$, $r^{\nu}_{12}[k]$, $r^{\nu}_{21}[k]$ and $r^{\nu}_{22}[k]$ generally are non-zero. This *'Noise Region Of Support'* is indicated by the white pentagon in the figure. It is assumed that the noise correlation functions are zero outside Ω^{ν}_{k}. Furthermore, let $\Omega^{s\nu}_{k} \subseteq \mathbb{Z}$ be the *'Source-Noise Region Of Support'* on which the source-noise correlation functions $r^{s\nu}_{11}[k], r^{s\nu}_{12}[k], r^{s\nu}_{21}[k]$ and $r^{s\nu}_{22}[k]$

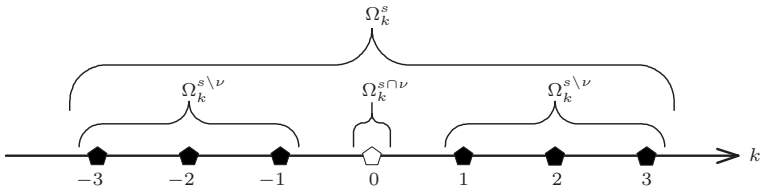

Figure 4.2: *Regions of support in the domain of lags k.*

generally are non-zero. Similarly, let $\Omega_k^{\nu s} \subseteq \mathbb{Z}$ be the *'Noise-Source Region Of Support'* on which the noise-source correlation functions $r_{11}^{\nu s}[k], r_{12}^{\nu s}[k], r_{21}^{\nu s}[k]$ and $r_{22}^{\nu s}[k]$ generally are non-zero. These latter two Regions Of Support are also indicated by the white pentagon. It is assumed that both the source-noise and noise-source correlation functions are zero outside their respective regions of support. The union $\Omega_k^{\nu} \cup \Omega_k^{s\nu} \cup \Omega_k^{\nu s}$ defines the ROS where unwanted noise is present. The part of this union that extends into the Signal ROS is given by the intersection $\Omega_k^{s\cap\nu} \triangleq \Omega_k^s \cap \left(\Omega_k^{\nu} \cup \Omega_k^{s\nu} \cup \Omega_k^{\nu s}\right)$ and is also indicated by the white pentagon. Finally, let the *'Noise-Free Region Of Support'* $\Omega_k^{s\setminus\nu}$ be defined by:

$$\boxed{\Omega_k^{s\setminus\nu} \triangleq \Omega_k^s \setminus \Omega_k^{s\cap\nu} = \Omega_k^s \setminus \left(\Omega_k^s \cap \left(\Omega_k^{\nu} \cup \Omega_k^{s\nu} \cup \Omega_k^{\nu s}\right)\right).}$$

This ROS is indicated by the set of black pentagons. In order to apply the theory that will be developed in the sequel, the various regions of support have to be known in advance. More precisely, in fact knowledge of $\Omega_k^{s\setminus\nu}$ is sufficient for the formulation of the assumptions in the remainder of this section. The other regions of support are merely required for defining or finding a suitable Noise-Free ROS $\Omega_k^{s\setminus\nu}$. Hence, in principle the various regions of support are part of the required a priori knowledge for a particular application, but it suffices to know only $\Omega_k^{s\setminus\nu}$. Guidelines for choosing the different regions of support in a logical and sensible manner easily follow from examples that will be presented during our exposition.

Using the definitions above for the various regions of support, the assumptions about the MIBI model on which the results in this chapter are based will now be discussed in turn. For reference purposes, they are also formulated as *AS1-AS4* in the list on page 138. Firstly, an assumption is made on the temporal cross-structure of the source signals, viz. that the cross-correlation function $r_{12}^s[k]$ of the two source signals $s_1[n]$ and $s_2[n]$ is zero for all lags $k \in \mathbb{Z}$. Mathematically, this can be formulated as follows (*AS1*):

$$r_{12}^s[k] = r_{21}^s[-k] = 0 \qquad \forall\, k \in \mathbb{Z}\,. \tag{4.2.5}$$

Secondly, an assumption is made on the temporal auto-structure of the source signals, viz. that the source signals are realizations of unit variance stationary autoregressive (AR) stochastic processes of order one. Consequently, each of them can be thought of as being generated according to the following AR(1) model:

$$s_j[n] = \rho_j s_j[n-1] + w_j[n]\,, \qquad j = 1, 2\,, \tag{4.2.6}$$

where ρ_j is an AR(1) regression coefficient and $w_j[n]$ is a sequence of independent identically distributed (i.i.d.) random variables with some probability density function (which is irrelevant for the current discussion). For demonstration purposes, we assume that $\rho_1 = 0.8$ and $\rho_2 = 0.6$. These values can be chosen more or less arbitrary as long as they are different. From (4.2.6), it is clear that both source signals possess an 'AR(1) temporal structure'. From this, and the assumption that each source signal has variance one, it follows that the

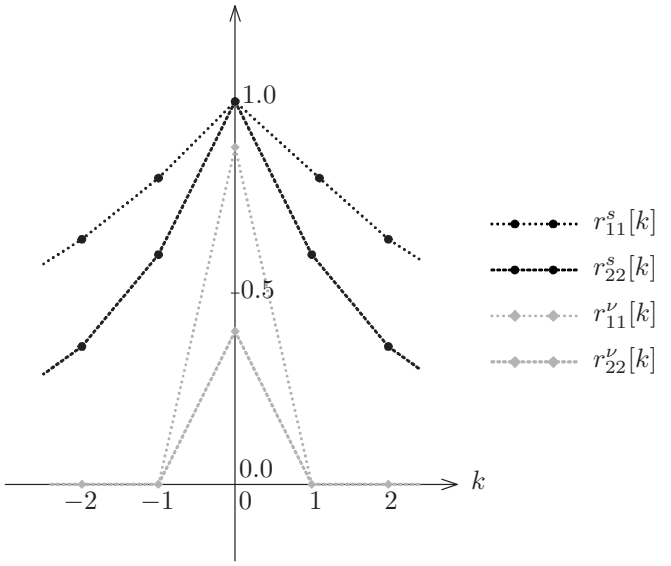

Figure 4.3: *Source and noise auto-correlation functions on* Ω^s_k.

auto-correlation function of the j-th source signal is given by (see also Fig. 4.3):

$$r^s_{jj}[k] = (\rho_j)^{|k|} \qquad \forall\, k \in \mathbb{Z}, \quad j = 1, 2\,. \tag{4.2.7}$$

From the fact that the source signals have different AR(1) coefficients, it follows that their corresponding auto-correlation functions are linearly independent in the following sense:

$$\sum_{j=1}^{2} \xi^j\, r^s_{jj}[k] = \xi^1\, r^s_{11}[k] + \xi^2\, r^s_{22}[k] = 0 \quad \forall\, k \in \mathbb{Z} \quad \Longrightarrow \quad \xi^1 = \xi^2 = 0\,, \tag{4.2.8}$$

where ξ^1 and ξ^2 are real- or complex-valued scalars. Because there are only two sources, (4.2.8) simply means that the source auto-correlation functions are no scaled versions of each other, i.e. there exists no scalar α such that $r^s_{11}[k] \neq \alpha\, r^s_{22}[k] \;\; \forall\, k \in \mathbb{Z}$. As can be seen in Fig. 4.3, this is clearly the case. The linear independence property (*AS2*) will prove to be very important for the method that will be presented in the sequel, both for the specific example considered in this chapter and for much more general scenarios. Assumptions (4.2.5) and (4.2.8) lead us to the following definition of the Source ROS:

$$\Omega^s_k \triangleq \mathbb{Z} = \{...,-2,-1,0,1,2,...\}\,. \tag{4.2.9}$$

Thirdly, the noise signals are assumed to be white and uncorrelated with each other for all lags. Hence, the auto-correlation function $r^\nu_{ii}[k]$ of the i-th noise signal can be written as:

$$r^\nu_{ii}[k] = (\sigma^\nu_i)^2\, \delta[k] \qquad \forall\, k \in \mathbb{Z}, \quad i = 1, 2\,, \tag{4.2.10}$$

where $(\sigma^\nu_i)^2$ denotes the variance of the i-th noise signal, and $\delta[k]$ is a Kronecker delta function ($\delta[k]$ is 1 iff $k = 0$ and 0 elsewhere[‡]). See Fig. 4.3, where we have taken $(\sigma^\nu_1)^2 = 0.88$ and $(\sigma^\nu_2)^2 = 0.4$. The cross-correlation function $r^\nu_{12}[k]$ of the two noise signals $\nu_1[n]$ and $\nu_2[n]$ is given by:

$$r^\nu_{12}[k] = r^\nu_{21}[-k] = 0 \qquad \forall\, k \in \mathbb{Z}\,. \tag{4.2.11}$$

[‡]See Section A.4 for the definition of various Kronecker delta functions.

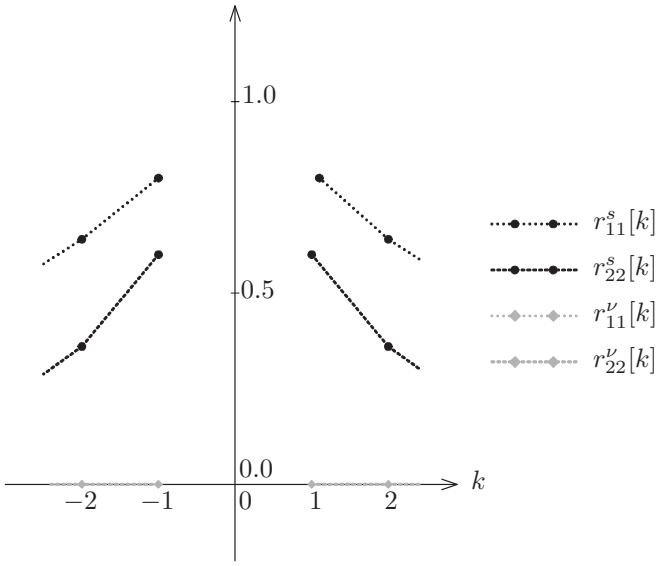

Figure 4.4: *Source and noise auto-correlation functions on* $\Omega_k^{s\setminus\nu}$.

Assumptions (4.2.10) and (4.2.11) can be written together in the following manner:

$$r^\nu_{i_1 i_2}[k] = \left(\sigma^\nu_{i_1}\right)^2 \delta_{i_1 i_2}\, \delta[k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \le i_1, i_2 \le 2\,. \tag{4.2.12}$$

where $\delta_{i_1 i_2}$ is also a Kronecker delta function ($\delta_{i_1 i_2}$ is 1 iff $i_1 = i_2$ and 0 elsewhere). This results in the following definition for the Noise ROS:

$$\Omega^\nu_k \triangleq \{0\}\,. \tag{4.2.13}$$

The fourth and final assumption concerns the cross-correlation between the source and noise signals. As is common and reasonable in many signal processing theories and applications, it is assumed that the source and noise signals are uncorrelated for all lags. Hence, the cross-correlation function $r^{\nu s}_{ij}[k]$ of the i-th noise signal and the j-th source signal can be written as follows:

$$r^{\nu s}_{ij}[k] = r^{s\nu}_{ji}[-k] = 0 \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \le i, j \le 2\,. \tag{4.2.14}$$

This assumption leads us to the following definition for the Source-Noise and Noise-Source Regions Of Support:

$$\Omega^{s\nu}_k = \Omega^{\nu s}_k \triangleq \emptyset\,. \tag{4.2.15}$$

The Noise-Free Region Of Support can now be deduced from (4.2.9), (4.2.13) and (4.2.15) yielding:

$$\begin{aligned}\Omega_k^{s\setminus\nu} &\triangleq \Omega^s_k \setminus \Omega^{s\cap\nu}_k = \Omega^s_k \setminus \left(\Omega^s_k \cap \left(\Omega^\nu_k \cup \Omega^{s\nu}_k \cup \Omega^{\nu s}_k\right)\right) \\ &= \{k \in \mathbb{Z} \mid k \neq 0\} = \{...,-2,-1,1,2,...\}\,.\end{aligned} \tag{4.2.16}$$

For convenience of notation, let $\mathbb{R}[\Omega_k^{s\setminus\nu}]$ denote the set/space of all real-valued functions that depend on a lag k and are defined on the Noise-Free ROS $\Omega_k^{s\setminus\nu}$. Most functions used in the sequel of this chapter are assumed to be elements of $\mathbb{R}[\Omega_k^{s\setminus\nu}]$. If functions are defined on another ROS, this will be mentioned explicitly. As an example, in Fig. 4.4 the source and

noise auto-correlation functions occurring in the assumptions above are plotted on the Noise-Free ROS $\Omega_k^{s\backslash\nu}$ given by (4.2.16). Compare this figure with Fig. 4.3. Finally, we remark that throughout the thesis, we will use the dedicated symbol N to denote the cardinality of the Noise-Free Region Of Support:

$$N \triangleq \left|\Omega_k^{s\backslash\nu}\right|. \tag{4.2.17}$$

For our purposes, it is sufficient that the assumptions formulated in (4.2.5), (4.2.8), (4.2.12), and (4.2.14) are satisfied only for $k \in \Omega_k^{s\backslash\nu}$ instead of $k \in \mathbb{Z}$. For the sake of overview and convenient referencing, we summarize these assumptions both in words and in mathematical form in the following list:

AS1: *The source signals are realizations of zero-mean real-valued random processes with zero cross-correlation functions on the Noise-Free ROS:*

$$r_{12}^s[k] = r_{21}^s[-k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu}\,;$$

AS2: *The source auto-correlation functions are linearly independent on the Noise-Free ROS:*

$$\sum_{j=1}^{2} \xi^j\, r_{jj}^s[k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu} \quad \Longrightarrow \quad \xi^1 = \xi^2 = 0\,;$$

AS3: *The noise signals are realizations of zero-mean real-valued random processes with zero auto- and cross-correlation functions on the Noise-Free ROS:*

$$r_{i_1 i_2}^{\nu}[k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad 1 \le i_1, i_2 \le 2\,;$$

AS4: *The cross-correlation functions between the source and noise signals are zero on the Noise-Free ROS:*

$$r_{ij}^{\nu s}[k] = r_{ji}^{s\nu}[-k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad 1 \le i, j \le 2\,.$$

The fact that these assumptions are formulated in terms of properties of correlation functions defined on a certain Region Of Support (ROS) that consists of several lag values is reminiscent of the fact that the temporal structure of the signals is involved and will be exploited. As we will see later on, for general scenarios the minimum required number of lags depends on the specific problem, viz. the number of sources and sensors, the temporal structure in the data, etc. For example, generally the more source signals are present, the more lags need to be considered and the more temporal structure is required, and vice versa. Based on assumptions *AS1-AS4*, we will show in the next sections that the MIBI problem has an elegant algebraic and geometric structure when formulated in the proper way.

4.3 Formulating MIBI as the problem of solving a system of homogeneous polynomial equations

Using the assumptions made in the previous section, in this section we show that the array response vectors, i.e. the columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of $\mathbf{A}$, satisfy the well-structured system of bivariate homogeneous polynomial equations of degree two alluded to in the introduction of this chapter (see the first 'subitem' of step 4 in Alg. 4.1 on page 132). This way, the MIBI problem is 'projected onto' a mathematical problem, the solution of which yields estimates of the array response vectors. The flow of argument in the derivations presented in the next

sections closely follows that in our derivation of subspace methods discussed in the previous chapter, thereby emphasizing that the discussed principles and concepts are very similar. In the course of our derivation we highlight the algebraic structure of the problem formulation.

In Section 4.3.1, we start by examining the kind and structure of the sensor statistics that are available for analysis. This primarily amounts to expressing the sensor correlation functions in terms of the mixing matrix elements and the source, noise, and joint source-noise correlation functions, and to studying their properties. As we have already indicated earlier, our goal is to derive the system of polynomial equations referred to above from the sensor correlation functions in such a way that the sensor noise is annihilated. No matter what the derivation procedure will be, this goal can always be achieved by using only those values of the sensor correlation functions that are noise-free, i.e. by considering all correlation functions on the Noise-Free ROS $\Omega_k^{s\backslash\nu}$ only. As we have remarked before (e.g. see Section 1.2.5) and will demonstrate in the sequel, for additive white sensor noise this means that only sensor correlation values for lags unequal to zero should be used. These values form the input for the rest of our argument, in which we consider sensor correlation function values on a chosen ROS in a specific configuration. In Section 4.3.2, we explain the notation used for denoting sets of correlation functions and index pairs. Then, in Section 4.3.3 we determine a set containing all unique sensor correlation functions, which plays a major role in the derivations presented in the sequel of this chapter.

The derivation of the system of equations can be formulated either in terms of correlation functions, in terms of the associated row vector notation, or in terms of the associated matrix-vector notation. The corresponding derivations are discussed in Sections 4.3.4, 4.3.5 and 4.3.6 respectively. We have chosen to firstly develop the theory in Section 4.3.4 in functional notation, i.e. terms of functions, because this is the most general formulation from which the other two follow straightforwardly, and because it can easily be generalized to more complex scenarios, as is done in later chapters. In Section 4.3.5 we briefly express the main results obtained in Section 4.3.4 in terms of the associated row vectors. Finally, we demonstrate in Section 4.3.6 that both the functional and row vector formulations naturally lead to the proper definition of a subspace matrix $\mathbf{C}^x$ that contains the employed sensor correlation values in a specific arrangement. Based on this matrix, again the main results of the theory are re-derived and expressed in matrix-vector notation, thereby demonstrating the power of our approach and the close link to the conventional subspace approach. For each formulation, our main argument can be divided into three stages. Firstly, the sensor correlation values are arranged in a specific configuration that allows to exploit the relation between the various involved linear subspaces on the one hand, and the elements of the mixing matrix on the other hand. Secondly, we derive the form of the system of polynomial equations satisfied by the columns of $\mathbf{A}$ by studying the properties of the linear (sub)spaces spanned by the (abstract) vectors occurring in the previous stage. Finally, we show how subspace decomposition using the Singular Value Decomposition (SVD) can be used to obtain proper coefficients for the polynomial functions in the system. This stage will complete the derivation of the system. Possible solutions of the obtained system are discussed in Section 4.5.

4.3.1 Structure of sensor correlation functions

We start our derivation by expressing the sensor correlation functions in terms of the mixing matrix elements and the source and noise correlation functions. The structure revealed in this way will be used in the following sections for developing our subspace based blind identifi-

cation method. For stationary signals, the correlation function $r^x_{i_1 i_2}[k]$ of the i_1-th and i_2-th sensor signals ($1 \leq i_1, i_2 \leq 2$) for lag k can be expressed as follows:

$$\begin{aligned} r^x_{i_1 i_2}[k] &\triangleq E\left\{x_{i_1}[n] x_{i_2}[n-k]\right\} = r^x_{i_2 i_1}[-k] \\ &\overset{(4.1.1)}{=} E\left\{ \left(\sum_{j_1=1}^{2} a_{i_1}^{j_1} s_{j_1}[n] + \nu_{i_1}[n] \right) \left(\sum_{j_2=1}^{2} a_{i_2}^{j_2} s_{j_2}[n-k] + \nu_{i_2}[n-k] \right) \right\} \\ &= \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} a_{i_1}^{j_1} a_{i_2}^{j_2} E\big\{s_{j_1}[n] s_{j_2}[n-k]\big\} + \sum_{j_1=1}^{2} a_{i_1}^{j_1} E\big\{s_{j_1}[n] \nu_{i_2}[n-k]\big\} \\ &\quad + \sum_{j_2=1}^{2} a_{i_2}^{j_2} E\big\{\nu_{i_1}[n] s_{j_2}[n-k]\big\} + E\big\{\nu_{i_1}[n] \nu_{i_2}[n-k]\big\} \\ &= \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} a_{i_1}^{j_1} a_{i_2}^{j_2} r^s_{j_1 j_2}[k] + \sum_{j_1=1}^{2} a_{i_1}^{j_1} r^{s\nu}_{j_1 i_2}[k] + \sum_{j_2=1}^{2} a_{i_2}^{j_2} r^{\nu s}_{i_1 j_2}[k] + r^{\nu}_{i_1 i_2}[k] \,. \end{aligned} \tag{4.3.1}$$

Using *AS1* and *AS4* from the list on page 138, this equation can be simplified to:

$$r^x_{i_1 i_2}[k] = \sum_{j=1}^{2} a_{i_1}^{j} a_{i_2}^{j} \, r^s_{jj}[k] + r^{\nu}_{i_1 i_2}[k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq i_1, i_2 \leq 2 \,. \tag{4.3.2}$$

By *AS3*, the sensor correlation functions are noise-free for all $k \in \Omega_k^{s\backslash\nu}$, i.e. for all lags unequal to zero. This follows from (4.2.12), which implies that the noise auto-correlation functions contribute to the sensor correlation functions for $k = 0$ only. Hence, we obtain the following result from (4.3.2), which will be used intensively in the remainder of the chapter:

$$\boxed{r^x_{i_1 i_2}[k] = \sum_{j=1}^{2} a_{i_1}^{j} a_{i_2}^{j} \, r^s_{jj}[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad \forall\, 1 \leq i_1, i_2 \leq 2 \,.} \tag{4.3.3}$$

4.3.2 Notation for sets of correlation functions and index pairs

For the sake of convenience of the following developments, the smooth generalization of various results, and uniformity and compactness of notation, we now introduce the notation that will be used extensively in the sequel for denoting sets of correlation functions and index pairs, as well as their associated cardinalities. The general form of the notation described in this section is discussed at length in Appendix A, in particular in Section A.2. As can be seen in the derivations above, if a pair of indices is given explicitly in the sub- or superscript position of a symbol, the parentheses and commas are omitted. For example, the index pair (i_1, i_2) in the subscript position of the sensor correlation function symbol $r^x_{i_1 i_2}[k]$ in (4.3.3) is written as $i_1 i_2$. Because sets containing index pairs occur on many occasions in this work we have developed a consistent and compact notation for them, whose general form is explained in detail in the second half of Appendix A. Now we introduce the notation for sets of index pairs pertaining to the current scenario. After that, we show how these sets can be employed to define sets of correlation functions.

The set containing all pairs of indices (i_1, i_2) obtained by varying both i_1 and i_2 from 1 till 2 is denoted by:

$$\mathcal{I}^2_{\mathrm{t},2} = \{(i_1, i_2) \mid 1 \leq i_1, i_2 \leq 2\} = \big\{(1,1), (1,2), (2,1), (2,2)\big\} \,. \tag{4.3.4}$$

This is a special case of (A.2.14); see also Fig. A.1 on page 408. The number 2 in the superscript position of $\mathcal{I}_{\mathrm{t},2}^{2}$ indicates that we are considering two different indices, viz. i_1 and i_2, whereas the number 2 in the subscript position indicates that both indices i_1 and i_2 range from 1 till 2. Finally, the first subscript index letter 't' stands for 'total' because the considered set contains all pairs obtained by varying the indices from 1 till 2 without any further constraints. The cardinality of the set $\mathcal{I}_{\mathrm{t},2}^{2}$ is denoted $M_{\mathrm{t},2}^{2}$ and given by:

$$M_{\mathrm{t},2}^{2} \triangleq \left|\mathcal{I}_{\mathrm{t},2}^{2}\right| = 4\,. \tag{4.3.5}$$

The set containing only those pairs of $\mathcal{I}_{\mathrm{t},2}^{2}$ that are different if the pairs are considered as sets (which are unordered, see Section A.3), is the same as that obtained by varying i_1 and i_2 from 1 till 2 in ascending order, and is denoted by:

$$\mathcal{I}_{\mathrm{a},2}^{2} = \{(i_1, i_2) \mid 1 \leq i_1 \leq i_2 \leq 2\} = \big\{(1,1),(1,2),(2,2)\big\}\,. \tag{4.3.6}$$

This is a special case of (A.2.15). The numbers 2 in the sub- and superscript positions have the same meaning as for $\mathcal{I}_{\mathrm{t},2}^{2}$, whereas the letter 'a' indicates that this set is of the 'ascending type' because the indices in each pair are ordered in ascending order. The cardinality of $\mathcal{I}_{\mathrm{a},2}^{2}$ is denoted $M_{\mathrm{a},2}^{2}$ and given by:

$$M_{\mathrm{a},2}^{2} \triangleq \left|\mathcal{I}_{\mathrm{a},2}^{2}\right| = 3\,. \tag{4.3.7}$$

Because a pair (i_1, i_2) can be used to index a function, e.g. $(1,2)$ indexes $r_{12}^{x}[k]$, sets like $\mathcal{I}_{\mathrm{t},2}^{2}$ and $\mathcal{I}_{\mathrm{a},2}^{2}$ containing index pairs can be employed to define sets of correlation functions. The set containing all sensor correlation functions is denoted by:

$$\begin{aligned}\mathcal{K}_{\mathrm{t},2}^{x,2} &\triangleq \big\{r_{i_1 i_2}^{x}[k] \mid 1 \leq i_1, i_2 \leq 2\big\} = \big\{r_{i_1 i_2}^{x}[k]\big\}_{1\leq i_1, i_2 \leq 2} \\ &= \big\{r_{i_1 i_2}^{x}[k]\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^{2}} = \big\{r_{11}^{x}[k], r_{12}^{x}[k], r_{21}^{x}[k], r_{22}^{x}[k]\big\}\,,\end{aligned} \tag{4.3.8}$$

where it is understood implicitly that the functions in the set are defined on the Noise-Free ROS $\Omega_k^{s\setminus\nu}$. The number 2 in the superscript position now represents the order of the employed statistics, whereas the number 2 in the subscript position has the same meaning as for $\mathcal{I}_{\mathrm{t},2}^{2}$. The cardinality of $\mathcal{K}_{\mathrm{t},2}^{x,2}$ is denoted by $M_{\mathrm{t},2}^{x,2}$. Since the elements of $\mathcal{K}_{\mathrm{t},2}^{x,2}$ are indexed by the elements of $\mathcal{I}_{\mathrm{t},2}^{2}$, this cardinality is given by:

$$M_{\mathrm{t},2}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{t},2}^{x,2}\right| = \left|\mathcal{I}_{\mathrm{t},2}^{2}\right| \overset{(4.3.5)}{=} M_{\mathrm{t},2}^{2} = 4\,. \tag{4.3.9}$$

Similarly, we define the set of sensor correlation functions indexed by $\mathcal{I}_{\mathrm{a},2}^{2}$ as follows:

$$\begin{aligned}\mathcal{K}_{\mathrm{a},2}^{x,2} &\triangleq \big\{r_{i_1 i_2}^{x}[k] \mid 1 \leq i_1 \leq i_2 \leq 2\big\} = \big\{r_{i_1 i_2}^{x}[k]\big\}_{1\leq i_1 \leq i_2 \leq 2} \\ &= \big\{r_{i_1 i_2}^{x}[k]\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^{2}} = \big\{r_{11}^{x}[k], r_{12}^{x}[k], r_{22}^{x}[k]\big\}\,.\end{aligned} \tag{4.3.10}$$

Since the elements of $\mathcal{K}_{\mathrm{a},2}^{x,2}$ are indexed by the elements of $\mathcal{I}_{\mathrm{a},2}^{2}$, the cardinality $M_{\mathrm{a},2}^{x,2}$ of $\mathcal{K}_{\mathrm{a},2}^{x,2}$ is given by:

$$M_{\mathrm{a},2}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{a},2}^{x,2}\right| = \left|\mathcal{I}_{\mathrm{a},2}^{2}\right| \overset{(4.3.7)}{=} M_{\mathrm{a},2}^{2} = 3\,. \tag{4.3.11}$$

Using single indices instead of index pairs, we denote and define the set containing the source auto-correlation functions as follows:

$$\mathcal{K}_{2}^{s,2} \triangleq \big\{r_{jj}^{s}[k] \mid 1 \leq j \leq 2\big\} = \big\{r_{jj}^{s}[k]\big\}_{1\leq j\leq 2} = \big\{r_{11}^{s}[k], r_{22}^{s}[k]\big\}\,. \tag{4.3.12}$$

Evidently, this set has cardinality $\left|\mathcal{K}_{2}^{s,2}\right| = S = 2$. The number 2 in the superscript position has the same meaning as for $\mathcal{K}_{\mathrm{t},2}^{x,2}$ and $\mathcal{K}_{\mathrm{a},2}^{x,2}$, i.e. it indicates the order of the employed statistics, whereas the number 2 in the subscript position indicates that the source auto-correlation function index j ranges from 1 till 2.

4.3.3 Determining set of unique sensor correlation functions

In this section we determine a set containing all unique sensor correlation functions; this set will play a major role in the derivations in the sequel of this chapter. In order to do so, we return to our study of the structure of the sensor correlation functions and examine possible symmetries in the elements of the set $\mathcal{K}_{\mathrm{t},2}^{x,2}$ defined in (4.3.8). Applying the commutativity property of the algebraic product operator to the product $a_{i_1}^j a_{i_2}^j$ occurring in (4.3.3) yields the following symmetry property of the sensor correlation functions:

$$\boxed{r_{i_1 i_2}^x[k] = r_{i_2 i_1}^x[k] \qquad \forall\,(i_1, i_2) \in \mathcal{I}_{\mathrm{t},2}^2\,.} \tag{4.3.13}$$

Recall that unless otherwise stated it is understood implicitly that all functions are defined on $\Omega_k^{s\backslash\nu}$. For index pairs $(i_1, i_2) = (1, 1)$ and $(i_1, i_2) = (2, 2)$ this property is trivial from the correlation function notation already because then it states that $r_{ii}^x[k] = r_{ii}^x[k]$ with $i = i_1 = i_2$. However, for $(i_1, i_2) = (1, 2)$ it means that $r_{12}^x[k] = r_{21}^x[k]$, which is not trivial because in general the cross-correlation functions $r_{i_1 i_2}^x[k]$ and $r_{i_2 i_1}^x[k]$ of two arbitrary signals $x_{i_1}[n]$ and $x_{i_2}[n]$ satisfy only the following property:

$$r_{i_1 i_2}^x[k] \triangleq E\left\{x_{i_1}[n] x_{i_2}[n-k]\right\} = E\left\{x_{i_2}[n] x_{i_1}[n+k]\right\} \triangleq r_{i_2 i_1}^x[-k]\,. \tag{4.3.14}$$

By combining properties (4.3.13) and (4.3.14) as follows:

$$r_{i_1 i_2}^x[k] = r_{i_2 i_1}^x[k] = r_{i_2 i_1}^x[-k] \qquad \forall\,(i_1, i_2) \in \mathcal{I}_{\mathrm{t},2}^2\,,$$

we can deduce the following 'lag-argument symmetry':

$$r_{i_1 i_2}^x[k] = r_{i_1 i_2}^x[-k] \qquad \forall\,(i_1, i_2) \in \mathcal{I}_{\mathrm{t},2}^2\,. \tag{4.3.15}$$

Hence, not only the sensor auto- but also the cross-correlation functions are symmetric in their lag argument, which is not true/trivial in general for cross-correlation functions.

Due to symmetry property (4.3.13) there exist only three 'essentially different' or 'unique' sensor correlation functions for the real-valued two by two mixing case, namely those in the set $\mathcal{K}_{\mathrm{u},2}^{x,2}$ defined as follows:

$$\begin{aligned}\mathcal{K}_{\mathrm{u},2}^{x,2} &\triangleq \left\{r_{i_1 i_2}^x[k] \mid 1 \le i_1 \le i_2 \le 2\right\} = \left\{r_{i_1 i_2}^x[k]\right\}_{1 \le i_1 \le i_2 \le 2} \\ &\triangleq \left\{r_{i_1 i_2}^x[k]\right\}_{(i_1, i_2) \in \mathcal{I}_{\mathrm{u},2}^2} = \left\{r_{11}^x[k], r_{12}^x[k], r_{22}^x[k]\right\},\end{aligned} \tag{4.3.16}$$

which for the current scenario equals $\mathcal{K}_{\mathrm{a},2}^{x,2}$ defined in (4.3.10). The set $\mathcal{K}_{\mathrm{u},2}^{x,2}$ contains the *'unique'* sensor correlation functions and has cardinality three. It is obtained from $\mathcal{K}_{\mathrm{t},2}^{x,2}$ by selecting only the unique functions, i.e. by incorporating only one member of each equivalence class with the equivalence relation defined by (4.3.13). It is in this sense that we sometimes call $\mathcal{K}_{\mathrm{u},2}^{x,2}$ the *'uniquiefied'* version of $\mathcal{K}_{\mathrm{t},2}^{x,2}$ or that $\mathcal{K}_{\mathrm{u},2}^{x,2}$ is obtained from $\mathcal{K}_{\mathrm{t},2}^{x,2}$ by *uniquiefication* of the latter set; hence the subscript 'u'. Note that for the current scenario the index set $\mathcal{I}_{\mathrm{u},2}^2$ corresponding to $\mathcal{K}_{\mathrm{u},2}^{x,2}$ is given by $\mathcal{I}_{\mathrm{u},2}^2 = \mathcal{I}_{\mathrm{a},2}^2$.

A remark concerning our symbolic notation is in place here. Note that although the set $\mathcal{K}_{\mathrm{t},2}^{x,2}$ in principle contains only three 'essentially different' or 'unique' functions, symbolically it contains all four different/possible sensor correlation functions. Therefore, its cardinality equals four and not three. In order to avoid confusion, we will say that the set $\mathcal{K}_{\mathrm{t},2}^{x,2}$ contains the *'(symbolically) different'* sensor correlation functions, whereas the set $\mathcal{K}_{\mathrm{a},2}^{x,2}$ contains the *'unique'* sensor correlation functions.

4.3.4 Derivation of system of bivariate homogeneous polynomial equations of degree two; functional notation

Now we are in a position to derive a system of bivariate homogeneous polynomial equations of degree two satisfied by the columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of the mixing matrix $\mathbf{A}$. As we have explained at the beginning of Section 4.3, in this section we first develop the theory in functional notation, i.e. in terms of functions.

4.3.4.1 Dimensions of signal and source subspaces

We start our derivation by considering the dimension of the linear space spanned by the sensor correlation functions (see Section 1.4), which is denoted by $d_2^{x,2}$:

$$\begin{aligned} d_2^{x,2} &\triangleq \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{t},2}^{x,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{r_{i_1 i_2}^x[k]\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^2}\Big)\Big) \\ &\equiv \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{r_{i_1 i_2}^x[k]\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2}\Big)\Big)\,. \end{aligned} \tag{4.3.17}$$

The meaning of the sub- and superscript quantities is exactly the same as for $\mathcal{K}_{\mathrm{t},2}^{x,2}$ and $\mathcal{K}_{\mathrm{a},2}^{x,2}$; see Section 4.3.2. In analogy with subspace method jargon the linear space $\mathcal{L}\big(\mathcal{K}_{\mathrm{t},2}^{x,2}\big) = \mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)$ will be called the *signal subspace*; see also Chapter 3. For obvious reasons we will also refer to this subspace as *sensor subspace*. Let $d_2^{s,2}$ denote the dimension of the linear space spanned by the source auto-correlation functions:

$$d_2^{s,2} \triangleq \dim\Big(\mathcal{L}\big(\mathcal{K}_2^{s,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{r_{jj}^s[k]\big\}_{1\le j\le 2}\Big)\Big)\,. \tag{4.3.18}$$

We will refer to the linear space $\mathcal{L}\big(\mathcal{K}_2^{s,2}\big)$ spanned by the source auto-correlation functions as *source subspace*. From *AS2* on page 138, i.e. from the fact that the source auto-correlation functions are linearly independent, it follows directly that its dimension is given by $d_2^{s,2} = S = 2$. Equation (4.3.3) states that each sensor correlation function $r_{i_1 i_2}^x[k]$ can be expressed as a linear combination of the two source auto-correlation functions $r_{11}^s[k]$ and $r_{22}^s[k]$ with coefficients $a_{i_1}^1 a_{i_2}^1$ and $a_{i_1}^2 a_{i_2}^2$ respectively. Hence, using linear algebra [115, 150] the following relation between $d_2^{x,2}$ and $d_2^{s,2}$ can be deduced:

$$d_2^{x,2} = \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\Big) \le \dim\Big(\mathcal{L}\big(\mathcal{K}_2^{s,2}\big)\Big) = d_2^{s,2} = S = 2\,. \tag{4.3.19}$$

This is a logical and intuitive consequence of the fact that all sensor correlation functions are linear combinations of the same two source auto-correlation functions. In principle, depending on the values of the mixing matrix elements, $d_2^{x,2}$ can be smaller than $d_2^{s,2}$. However, in general $d_2^{x,2}$ will be equal to $d_2^{s,2}$ for a randomly generated $\mathbf{A}$. In Section 4.3.6.3, we will show in detail how $d_2^{x,2}$ depends on $\mathbf{A}$ and $d_2^{s,2}$.

4.3.4.2 Linear dependence of sensor correlation functions

Now we will present an essential part of our derivation that is based on the observation that the number of unique sensor correlation functions in $\mathcal{K}_{\mathrm{a},2}^{x,2}$, i.e. the cardinality $\big|\mathcal{K}_{\mathrm{a},2}^{x,2}\big| = |\mathcal{I}_{\mathrm{a},2}^2| = M_{\mathrm{a},2}^2 = 3$ (see (4.3.7) and (4.3.11)), is larger than the dimension $d_2^{x,2}$ of the signal subspace spanned by the functions in $\mathcal{K}_{\mathrm{a},2}^{x,2}$, which is equal to or smaller than 2 due to (4.3.19):

$$M_{\mathrm{a},2}^{x,2} = \big|\mathcal{K}_{\mathrm{a},2}^{x,2}\big| = 3 \quad > \quad d_2^{x,2} \le d_2^{s,2} = S = 2\,. \tag{4.3.20}$$

Stated differently, the number $\left|\mathcal{K}_{\mathrm{a},2}^{x,2}\right|$ of vectors in the linear vector space $\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)$ is larger than the dimension $\dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\right)$ of the space spanned by these vectors:

$$\left|\mathcal{K}_{\mathrm{a},2}^{x,2}\right| > \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\right) . \tag{4.3.21}$$

Hence, we must conclude that *the sensor correlation functions in* $\mathcal{K}_{\mathrm{a},2}^{x,2}$ *are linearly dependent*. The same important conclusion can be drawn if we replace $\mathcal{K}_{\mathrm{a},2}^{x,2}$ by $\mathcal{K}_{\mathrm{t},2}^{x,2}$, but we have not done so because the latter set contains twice the same function $\big(r_{12}^{x}[k] = r_{21}^{x}[k]\big)$, thereby making the linear dependence property of the set trivial. By the definition of linear dependence, there exist non-zero and non-unique sets of coefficients $\big\{\varphi_1^{11}, \varphi_1^{12}, \varphi_1^{22}\big\}, \ldots,$ $\big\{\varphi_Q^{11}, \varphi_Q^{12}, \varphi_Q^{22}\big\}$, indexed by an arbitrarily integer-valued index q, such that:

$$\varphi_q^{11}\, r_{11}^{x}[k] + \varphi_q^{12}\, r_{12}^{x}[k] + \varphi_q^{22}\, r_{22}^{x}[k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad \forall\, q \in \mathcal{Q}\,, \tag{4.3.22}$$

where the set $\mathcal{Q}$ is defined as:

$$\mathcal{Q} \triangleq \{1, \ldots, Q\} \tag{4.3.23}$$

with $Q \triangleq |\mathcal{Q}|$ the maximum number of linearly independent equations. This number equals the dimension of the orthogonal complement of the linear space spanned by the sensor correlation functions in $\mathcal{K}_{\mathrm{a},2}^{x,2}$, i.e. the signal subspace spanned by $r_{11}^{x}[k]$, $r_{12}^{x}[k]$, and $r_{22}^{x}[k]$. From the results in Appendix C (in particular, see the box at the bottom of page 449) it follows that this dimension is given by:

$$\boxed{Q = \dim\left(\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\right)^{\perp}\right) = \left|\mathcal{K}_{\mathrm{a},2}^{x,2}\right| - \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{a},2}^{x,2}\big)\right) = M_{\mathrm{a},2}^{x,2} - d_2^{x,2}\,.} \tag{4.3.24}$$

Hence, the number Q of linearly independent equations in (4.3.22) is given by the difference between the left and right hand sides of (4.3.21). In Section 4.3.8 we will show that a valid set of coefficients of the equations in (4.3.22) can be deduced from the Singular Value Decomposition (SVD) of a function-valued vector associated with the set $\mathcal{K}_{\mathrm{a},2}^{x,2}$, or equivalently from the SVD of a properly defined subspace matrix. Thus, we can consider the coefficients of the equations as (approximately) known quantities. In concise form, the linear dependence property of the sensor correlation functions given in (4.3.22) can be written as follows:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^{2}} \varphi_q^{i_1 i_2} r_{i_1 i_2}^{x}[k] \;=\; \sum_{1\le i_1\le i_2\le 2} \varphi_q^{i_1 i_2} r_{i_1 i_2}^{x}[k] = 0 \quad \forall\, k \in \Omega_k^{s\backslash\nu}, \;\; \forall\, q \in \mathcal{Q}\,. \tag{4.3.25}$$

The set containing the different sets of coefficients for each value of $q \in \mathcal{Q}$ in (4.3.22) and (4.3.25) is denoted by Φ, i.e.:

$$\Phi \triangleq \left\{ \left\{\varphi_q^{i_1 i_2}\right\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^{2}} \right\}_{q\in\mathcal{Q}} = \Big\{ \big\{\varphi_1^{11}, \varphi_1^{12}, \varphi_1^{22}\big\}, \ldots, \big\{\varphi_Q^{11}, \varphi_Q^{12}, \varphi_Q^{22}\big\} \Big\}\,. \tag{4.3.26}$$

Note that all linear combinations of the sets in Φ also satisfy equations like (4.3.22) and (4.3.25).

It should be noted again that completely similar and equivalent equations and results are obtained if we replace $\mathcal{K}_{\mathrm{a},2}^{x,2}$ by $\mathcal{K}_{\mathrm{t},2}^{x,2}$. In this case, we could define non-zero and non-unique sets of coefficients $\big\{\acute{\varphi}_1^{11}, \acute{\varphi}_1^{12}, \acute{\varphi}_1^{21}, \acute{\varphi}_1^{22}\big\}, \ldots, \big\{\acute{\varphi}_Q^{11}, \acute{\varphi}_Q^{12}, \acute{\varphi}_Q^{21}, \acute{\varphi}_Q^{22}\big\}$, indexed by an arbitrarily integer-valued index q, such that the equivalent of (4.3.25) would become:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^{2}} \acute{\varphi}_q^{i_1 i_2} r_{i_1 i_2}^{x}[k] = 0 \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad \forall\, q \in \mathcal{Q} \tag{4.3.27}$$

and the set containing the different sets of coefficients would be given by:

$$'\Phi \triangleq \left\{ \left\{ \varphi_q^{i_1 i_2} \right\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^2} \right\}_{q\in\mathcal{Q}} . \tag{4.3.28}$$

In fact, we can easily set up a mapping between the sets Φ and $'\Phi$. In the sequel, in order to eliminate trivial redundancy we will mainly work with the set $\mathcal{K}_{\mathrm{a},2}^{x,2}$, which will also reduce the computational complexity of implementations. However, sometimes we will equivalently choose to work with $\mathcal{K}_{\mathrm{t},2}^{x,2}$ for providing additional insight and for convenience of notation. The reader can easily convert the results from one approach into the other.

4.3.4.3 Deriving the system of equations by exploiting linear independence of source auto-correlation functions

The next main stage of our derivation amounts to combining (4.3.3) and (4.3.25), and exploiting assumption *AS2*; this directly leads us to the desired system of equations. Firstly, (4.3.25) is expressed in the source auto-correlation functions by substituting (4.3.3):

$$\begin{aligned}
\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} r_{i_1 i_2}^x[k] &= \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} \left(\sum_{j=1}^{2} a_{i_1}^j a_{i_2}^j \, r_{jj}^s[k] \right) \\
&= \sum_{j=1}^{2} \left(\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^j a_{i_2}^j \right) r_{jj}^s[k] \\
&= \left(\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^1 a_{i_2}^1 \right) r_{11}^s[k] + \left(\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^2 a_{i_2}^2 \right) r_{22}^s[k] \\
&= 0[k] \qquad \forall\, k \in \Omega_k^{s\setminus\nu}, \quad \forall\, q \in \mathcal{Q}\,,
\end{aligned} \tag{4.3.29}$$

where $0[k]$ is written to stress the fact that the right hand side of the equation (and thus also the left hand side) represents a function that in theory is identically zero on $\Omega_k^{s\setminus\nu}$ (see Section 1.4). Now, we have arrived at the point where the essence of assumption *AS2*, viz. the linear independence of the source auto-correlation functions, comes into play. Applying *AS2* to the last line in (4.3.29) by identifying:

$$\xi_q^1 = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^1 a_{i_2}^1 \quad \text{and} \quad \xi_q^2 = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^2 a_{i_2}^2 \,,$$

it follows that:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} a_{i_1}^j a_{i_2}^j = 0 \qquad \forall\, q \in \mathcal{Q}, \qquad j = 1,2\,. \tag{4.3.30}$$

This system of equations describes the relation between the *unknown* coefficients of the mixing matrix $\mathbf{A}$ and the *known* set of sets of coefficients Φ. From (4.3.30), it follows that for each set of coefficients $\left\{\varphi_q^{11}, \varphi_q^{12}, \varphi_q^{22}\right\}$ in Φ, i.e. for each value of q, the elements of the mixing matrix $\mathbf{A}$ satisfy the following system of two equations:

$$\begin{cases}
\varphi_q^{11} a_1^1 a_1^1 + \varphi_q^{12} a_1^1 a_2^1 + \varphi_q^{22} a_2^1 a_2^1 &= 0\,; \\
\varphi_q^{11} a_1^2 a_1^2 + \varphi_q^{12} a_1^2 a_2^2 + \varphi_q^{22} a_2^2 a_2^2 &= 0\,.
\end{cases}$$

The important point to note here is that the first equation only contains the unknowns a_1^1 and a_2^1 from the first column $\mathbf{a}^1$ of $\mathbf{A}$, whereas the second only contains the unknowns a_1^2 and a_2^2 from the second column $\mathbf{a}^2$. Hence, both columns $\mathbf{a}^1$ and $\mathbf{a}^2$ satisfy the same system of equations indexed by $q \in \mathcal{Q}$. More specifically, let $\mathbf{z} = \left[\begin{smallmatrix} z_1 \\ z_2 \end{smallmatrix}\right] \in \mathbb{C}_2$ be a column vector of variables having the same size as a column of $\mathbf{A}$, and define the functions $\{f_q(\mathbf{z})\}_{q\in\mathcal{Q}}$ by:

$$\boxed{\begin{aligned} f_q(\mathbf{z}) &\triangleq \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} z_{i_1} z_{i_2} \\ &= \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \qquad \forall\, \mathbf{z} \in \mathbb{C}_2\,, \quad \forall\, q \in \mathcal{Q}\,. \end{aligned}} \tag{4.3.31}$$

Then, (4.3.30) states that both columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of $\mathbf{A}$ satisfy the system of equations $\{f_q(\mathbf{z}) = 0\}_{q\in\mathcal{Q}}$, i.e.:

$$f_q(\mathbf{a}^j) = 0 \qquad \forall\, q \in \mathcal{Q}\,, \quad j = 1, 2\,. \tag{4.3.32}$$

Hence, at this point *the MIBI problem has been 'projected onto' the problem of solving the following system of equations for the columns of the mixing matrix* $\mathbf{A}$:

$$\boxed{\{f_q(\mathbf{z}) = 0\}_{q\in\mathcal{Q}}\,.} \tag{4.3.33}$$

By 'projected' we mean that the system of equations follows from our MIBI problem definition, but not necessarily vice versa. In other words, in a sense we have projected the MIBI problem onto (4.3.33), but (4.3.33) does not necessarily 'imply' the MIBI problem. Later on, we will see that solving a system like (4.3.33) solves our MIBI problem.

4.3.4.4 Properties and structure of functions and system

All functions in system (4.3.33) have the same specific form. Only the coefficients $\varphi_q^{11}, \varphi_q^{12}$ and φ_q^{22} are different for each function. From (4.3.31), it is clear that each $f_q(\mathbf{z}) = f_q(z_1, z_2)$ is a bivariate polynomial of degree two. As is also evident from this equation, *all* terms in each function $f_q(\mathbf{z})$ have the same degree two. This implies that each function $f_q(\mathbf{z})$ in (4.3.33) is *homogeneous of degree two*, also called 2*-homogeneous*, meaning that:

$$\boxed{f_q(\eta\mathbf{z}) = (\eta)^2 f_q(\mathbf{z}) \qquad \forall\, \eta \in \mathbb{C}, \qquad \forall\, \mathbf{z} \in \mathbb{C}_2\,.} \tag{4.3.34}$$

Recall that parentheses are used to indicate the computation of a power, e.g. $(a)^b$ means a to the power b; see Appendix A. This property can easily be proven from the definition in (4.3.31) as follows:

$$f_q(\eta z_1, \eta z_2) = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} \big(\eta z_{i_1}\big)\big(\eta z_{i_2}\big) = (\eta)^2 \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} z_{i_1} z_{i_2} = (\eta)^2 f_q(z_1, z_2)\,.$$

For each function $f_q(\mathbf{z})$ in the system, the following nice property of homogeneous functions follows immediately from (4.3.34):

$$\boxed{f_q(\mathbf{v}) = 0 \qquad \Longrightarrow \qquad f_q(\eta\mathbf{v}) = 0 \qquad \forall\, \eta \in \mathbb{C}\,.} \tag{4.3.35}$$

Hence, if $\mathbf{v}$ is a solution of the system $\{f_q(\mathbf{z}) = 0\}_{q\in\mathcal{Q}}$, then also $\eta\mathbf{v}$ is a solution for all $\eta \in \mathbb{C}$. This is a logical result of the scaling indeterminacy inherent to MIBI (see Section 2.4). In addition, the homogeneity property of the functions in the system implies that:

$$f_q(\mathbf{v}) = 0 \quad \forall\, q \in \mathcal{Q} \quad \Longrightarrow \quad \sum_{q\in\mathcal{Q}} \sum_{p\in\mathbb{N}} \alpha_p^q f_q(\eta^p \mathbf{v}) = 0 \quad \forall\, \alpha_p^q \in \mathbb{C}\,, \quad \forall\, \eta^p \in \mathbb{C}\,, \tag{4.3.36}$$

and thus:

$$f_q(\mathbf{v}) = 0 \quad \forall\, q \in \mathcal{Q} \quad \Longrightarrow \quad \sum_{q\in\mathcal{Q}} \alpha_p^q\, f_q(\mathbf{v}) = 0 \quad \forall\, \alpha_p^q \in \mathbb{C}\,, \quad \forall\, p \in \mathbb{N}\,. \tag{4.3.37}$$

Hence, applying a full rank linear transformation to system (4.3.33) and solving the resulting system is equivalent to solving (4.3.33) itself, i.e. instead of solving (4.3.33) we can also solve the following system:

$$\left\{ \sum_{q\in\mathcal{Q}} \alpha_p^q f_q(\mathbf{z}) = 0 \right\}_{p\in\mathcal{Q}}\,, \tag{4.3.38}$$

where the matrix $\boldsymbol{\Upsilon} \triangleq \left[\alpha_p^q\right] \in \mathbb{C}_Q^Q$ is any full rank matrix. This is reminiscent of the remark made just after (4.3.26), viz. that all linear combinations of the sets in Φ also give 'proper' sets of coefficients because the polynomials in (4.3.38) have exactly the same form as those in (4.3.31), but with coefficients $\left\{ \left\{ \varrho_p^{i_1 i_2} \right\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \right\}_{p\in\mathcal{Q}}$ that are linear combinations of those in (4.3.31):

$$\begin{aligned}
\sum_{q\in\mathcal{Q}} \alpha_p^q f_q(\mathbf{z}) \;&=\; \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \left(\sum_{q\in\mathcal{Q}} \alpha_p^q \varphi_q^{i_1 i_2} \right) z_{i_1} z_{i_2} \;\triangleq\; \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varrho_p^{i_1 i_2} z_{i_1} z_{i_2} \\
&= \left(\sum_{q\in\mathcal{Q}} \alpha_p^q \varphi_q^{11} \right) z_1 z_1 + \left(\sum_{q\in\mathcal{Q}} \alpha_p^q \varphi_q^{12} \right) z_1 z_2 + \left(\sum_{q\in\mathcal{Q}} \alpha_p^q \varphi_q^{22} \right) z_2 z_2 \\
&\triangleq\; \varrho_p^{11} z_1 z_1 + \varrho_p^{12} z_1 z_2 + \varrho_p^{22} z_2 z_2 \qquad \forall\, \mathbf{z} \in \mathbb{C}_2\,, \quad \forall\, \alpha_p^q \in \mathbb{C}\,.
\end{aligned} \tag{4.3.39}$$

Hence, the coefficients of the polynomials may be replaced by arbitrary (linearly independent) linear combinations:

$$\varrho_p^{i_1 i_2} \triangleq \sum_{q\in\mathcal{Q}} \alpha_p^q \varphi_q^{i_1 i_2} \qquad \forall\, p \in \mathcal{Q}\,, \quad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2\,. \tag{4.3.40}$$

Clearly, also if rank-deficient linear transformations are considered the obtained system of equations is still valid, but in this case it is not equivalent to the original system any more because information is lost. More properties of the system, and the functions and equations it contains, will be investigated in Section 4.4, where we study several aspects of the algebraic and geometric structure of the problem induced by the system of equations.

4.3.5 Derivation of system of bivariate homogeneous polynomial equations of degree two; row vector notation

In this section, we briefly repeat the derivation in the previous section in terms of row vectors. This will serve as a natural and intuitive intermediate step towards the derivation in terms of a subspace matrix, which is discussed in Section 4.3.6. In fact, we will see that we only need to set up a proper one-to-one mapping between the correlation functions considered in the previous section on the one hand, and corresponding correlation row vectors on the other; see Fig. 1.11 on page 26. We do this by associating a row vector with each correlation function defined on $\Omega_k^{s\backslash\nu}$. This correspondence will also be used later on for the construction of the subspace matrix.

4.3.5.1 Bijective mapping between correlation functions and correlation row vectors

In Section 1.4, we have explained how a bijective mapping can be established between a function defined on some interval on the one hand, and a row vector on the other. Here, we will apply this to correlation functions. For each pair $(i_1, i_2) \in \mathcal{I}_{\mathrm{t},2}^2$, a row vector $\tilde{\mathbf{r}}_{i_1 i_2}^x$ is associated with the sensor correlation function $r_{i_1 i_2}^x[k]$ defined on $\Omega_k^{s\setminus\nu}$ given in (4.2.16) by defining:

$$\tilde{\mathbf{r}}_{i_1 i_2}^x \triangleq \begin{bmatrix} \cdots & r_{i_1 i_2}^x[-2] & r_{i_1 i_2}^x[-1] & r_{i_1 i_2}^x[1] & r_{i_1 i_2}^x[2] & \cdots \end{bmatrix} . \tag{4.3.41}$$

In the same manner, we can construct a bijective mapping for a general ROS, e.g. if:

$$\Omega_k^{s\setminus\nu} \triangleq \{k^1, k^2, \ldots, k^N\} , \tag{4.3.42}$$

the row vector associated with $r_{i_1 i_2}^x[k]$ defined on $\Omega_k^{s\setminus\nu}$ is given by:

$$\tilde{\mathbf{r}}_{i_1 i_2}^x \triangleq \begin{bmatrix} r_{i_1 i_2}^x[k^1] & r_{i_1 i_2}^x[k^2] & \cdots & r_{i_1 i_2}^x[k^N] \end{bmatrix} . \tag{4.3.43}$$

The length of $\tilde{\mathbf{r}}_{i_1 i_2}^x$ equals the cardinality N of the Noise-Free Region Of Support $\Omega_k^{s\setminus\nu}$ defined in (4.2.17). Following the same procedure as described above, a row vector $\tilde{\mathbf{r}}_{jj}^s$ is associated bijectively with the j-th source auto-correlation function $r_{jj}^s[k]$. For instance, for the ROS in (4.3.42) we define:

$$\tilde{\mathbf{r}}_{jj}^s \triangleq \begin{bmatrix} r_{jj}^s[k^1] & r_{jj}^s[k^2] & \cdots & r_{jj}^s[k^N] \end{bmatrix} . \tag{4.3.44}$$

Using these notational conventions, the row vector equivalent of (4.3.3) becomes:

$$\boxed{\tilde{\mathbf{r}}_{i_1 i_2}^x = \sum_{j=1}^{2} a_{i_1}^j a_{i_2}^j \, \tilde{\mathbf{r}}_{jj}^s \qquad \forall\, 1 \le i_1, i_2 \le 2 .} \tag{4.3.45}$$

Since the mapping between functions and their associated row vectors is bijective, we will denote the various sets of correlation row vectors by symbols that are similar to those denoting the corresponding sets of correlation functions. The set containing all sensor correlation row vectors, i.e. the row vector equivalent of (4.3.8), is denoted and given by:

$$\begin{aligned} \tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2} &\triangleq \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \mid 1 \le i_1, i_2 \le 2 \right\} = \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \right\}_{1 \le i_1, i_2 \le 2} \\ &= \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \right\}_{(i_1, i_2) \in \mathcal{I}_{\mathrm{t},2}^2} = \left\{ \tilde{\mathbf{r}}_{11}^x, \tilde{\mathbf{r}}_{12}^x, \tilde{\mathbf{r}}_{21}^x, \tilde{\mathbf{r}}_{22}^x \right\} . \end{aligned} \tag{4.3.46}$$

Likewise, the set containing the unique sensor correlation row vectors corresponding to (4.3.10) is denoted and given by:

$$\begin{aligned} \tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2} &\triangleq \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \mid 1 \le i_1 \le i_2 \le 2 \right\} = \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \right\}_{1 \le i_1 \le i_2 \le 2} \\ &= \left\{ \tilde{\mathbf{r}}_{i_1 i_2}^x \right\}_{(i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2} = \left\{ \tilde{\mathbf{r}}_{11}^x, \tilde{\mathbf{r}}_{12}^x, \tilde{\mathbf{r}}_{22}^x \right\} , \end{aligned} \tag{4.3.47}$$

and finally the set containing the source auto-correlation row vectors corresponding to (4.3.12) is denoted and given by:

$$\tilde{\mathcal{K}}_2^{s,2} \triangleq \left\{ \tilde{\mathbf{r}}_{jj}^s \mid 1 \le j \le 2 \right\} = \left\{ \tilde{\mathbf{r}}_{jj}^s \right\}_{1 \le j \le 2} = \left\{ \tilde{\mathbf{r}}_{11}^s, \tilde{\mathbf{r}}_{22}^s \right\} . \tag{4.3.48}$$

Evidently, the cardinalities of $\tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2}$, $\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}$ and $\tilde{\mathcal{K}}_2^{s,2}$ are the same as those of $\mathcal{K}_{\mathrm{t},2}^{x,2}$, $\mathcal{K}_{\mathrm{a},2}^{x,2}$ and $\mathcal{K}_2^{s,2}$ respectively, and are given in Section 4.3.2.

4.3.5.2 Dimensions of signal and source subspaces

Now we are in a position to derive the same system of equations as in the previous section in terms of row vectors. By analogy, we start the derivation by considering the dimension of the signal subspace spanned by the sensor correlation row vectors. First note that due to the bijective mapping between functions and row vectors, the dimensions of the corresponding linear vector spaces spanned by them are the same. Therefore, we can use the same symbols as in the previous section. Hence, the dimension of the signal subspace spanned by the sensor correlation row vectors, which is expressed in (4.3.17) in functional notation, can now be expressed in terms of the associated row vectors as follows:

$$\begin{aligned} d_2^{x,2} &\triangleq \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{\tilde{\mathbf{r}}_{i_1 i_2}^{x}\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^2}\Big)\Big) \\ &\equiv \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{\tilde{\mathbf{r}}_{i_1 i_2}^{x}\big\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2}\Big)\Big)\,. \end{aligned} \tag{4.3.49}$$

Likewise, the dimension of the linear space spanned by the source auto-correlation functions, which is expressed in (4.3.18) in functional notation, can now be expressed in terms of the associated row vectors as follows:

$$d_2^{s,2} \triangleq \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_2^{s,2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{\tilde{\mathbf{r}}_{jj}^{s}\big\}_{1\leq j\leq 2}\Big)\Big)\,. \tag{4.3.50}$$

The row vector equivalent of assumption *AS2* on page 138, which states that the source auto-correlation functions are linearly independent, is given by:

$$\sum_{j=1}^{2}\xi^j\,\tilde{\mathbf{r}}_{jj}^{s} = \xi^1\,\tilde{\mathbf{r}}_{11}^{s} + \xi^2\,\tilde{\mathbf{r}}_{22}^{s} = \tilde{\mathbf{0}}^N \qquad \Longrightarrow \qquad \xi^1 = \xi^2 = 0\,. \tag{4.3.51}$$

From this property it follows again that $d_2^{s,2} = 2$. Equation (4.3.45) states that each sensor correlation row vector $\tilde{\mathbf{r}}_{i_1 i_2}^{x}$ can be expressed as a linear combination of the two source auto-correlation row vectors $\tilde{\mathbf{r}}_{11}^{s}$ and $\tilde{\mathbf{r}}_{22}^{s}$ with coefficients $a_{i_1}^1 a_{i_2}^1$ and $a_{i_1}^2 a_{i_2}^2$ respectively. This again implies relation (4.3.19) between $d_2^{x,2}$ and $d_2^{s,2}$.

4.3.5.3 Linear dependence of sensor correlation row vectors

Since the number $M_{\mathrm{a},2}^{x,2} = |\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}|$ of unique sensor correlation row vectors in $\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}$ is larger than the dimension $d_2^{x,2}$ of the signal subspace, we obtain the following equivalent of (4.3.21):

$$\big|\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}\big| > \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}\big)\Big)\,. \tag{4.3.52}$$

Hence, now we must conclude that the sensor correlation row vectors in $\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}$ are linearly dependent. Thus, equivalently to (4.3.22), by the definition of linear dependence there exist non-zero and non-unique sets of coefficients $\big\{\varphi_1^{11},\varphi_1^{12},\varphi_1^{22}\big\},\ldots,\big\{\varphi_Q^{11},\varphi_Q^{12},\varphi_Q^{22}\big\}$, indexed by an index q, such that:

$$\varphi_q^{11}\,\tilde{\mathbf{r}}_{11}^{x} + \varphi_q^{12}\,\tilde{\mathbf{r}}_{12}^{x} + \varphi_q^{22}\,\tilde{\mathbf{r}}_{22}^{x} = \tilde{\mathbf{0}}^N \qquad \forall\, q\in\mathcal{Q}\,, \tag{4.3.53}$$

where the set $\mathcal{Q}$ is defined in (4.3.23). Note that the number $Q \triangleq |\mathcal{Q}|$ of linearly independent equations is given by (4.3.24) and that it also equals the dimension of the orthogonal complement of the linear space spanned by the sensor correlation row vectors in $\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}$, i.e. the signal

subspace spanned by $\tilde{\mathbf{r}}^x_{11}$, $\tilde{\mathbf{r}}^x_{12}$, and $\tilde{\mathbf{r}}^x_{22}$:

$$\boxed{Q = \dim\left(\left(\mathcal{L}\big(\tilde{\mathcal{K}}^{x,2}_{\mathrm{a},2}\big)\right)^{\perp}\right) = \big|\tilde{\mathcal{K}}^{x,2}_{\mathrm{a},2}\big| - \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}^{x,2}_{\mathrm{a},2}\big)\right) = M^{x,2}_{\mathrm{a},2} - d^{x,2}_{2}\,.} \tag{4.3.54}$$

Hence, in the same way as Q is given by the difference between the left and right hand sides of (4.3.21), it is also given by the difference between the left and right hand sides of (4.3.52). Similarly to (4.3.25), in concise form this linear dependence property of the sensor correlation row vectors can be written as follows (leaving out the length N of the zero row vector):

$$\sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2}\tilde{\mathbf{r}}^x_{i_1 i_2} \;=\; \sum_{1\le i_1\le i_2\le 2} \varphi_q^{i_1 i_2}\tilde{\mathbf{r}}^x_{i_1 i_2} = \tilde{\mathbf{0}} \qquad \forall\, q \in \mathcal{Q}\,. \tag{4.3.55}$$

Note that the orthogonal complement of the signal subspace is the noise subspace.

4.3.5.4 Deriving the system of equations by exploiting linear independence of source auto-correlation row vectors

The main stage of our derivation leading to the desired system of equations amounts to combining (4.3.45) and (4.3.55), and exploiting assumption *AS2* in the form of (4.3.51). Hence, expressing (4.3.55) in terms of the source auto-correlation row vectors by substituting (4.3.45) yields:

$$\sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2}\tilde{\mathbf{r}}^x_{i_1 i_2} = \sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2}\left(\sum_{j=1}^{2} a^j_{i_1} a^j_{i_2}\,\tilde{\mathbf{r}}^s_{jj}\right) = \sum_{j=1}^{2}\left(\sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2} a^j_{i_1} a^j_{i_2}\right)\tilde{\mathbf{r}}^s_{jj}$$

$$= \left(\sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2} a^1_{i_1} a^1_{i_2}\right)\tilde{\mathbf{r}}^s_{11} + \left(\sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2} a^2_{i_1} a^2_{i_2}\right)\tilde{\mathbf{r}}^s_{22} = \tilde{\mathbf{0}} \quad \forall\, q \in \mathcal{Q}\,. \tag{4.3.56}$$

Now exploiting the linear independence property of the source auto-correlation row vectors, i.e. applying (4.3.51) to the last line in (4.3.56) in the same manner as we did in the previous section again gives system (4.3.30). Hence, from this point on the derivation of the system of polynomial equations is exactly the same as that in the previous section from (4.3.30) on.

4.3.6 Derivation of system of bivariate homogeneous polynomial equations of degree two; matrix-vector notation and subspace matrix

In this section we finally present the derivation of the system of homogeneous polynomial equations in terms of matrix-vector notation and a subspace matrix $\mathbf{C}^x$. In the previous two sections we have shown that the derivation of this system is based on the properties of the signal and noise subspaces. In order to be able to naturally define a proper subspace matrix for our problem formulation and to derive the system of equations from this, in addition to the one-to-one mapping between correlation functions and their corresponding correlation row vectors that we used in the previous section, now we also need to set up a proper one-to-one mapping between function-valued vectors on the one hand and matrices on the other; see Fig. 1.11 on page 26.

4.3.6.1 Bijective mapping between function-valued vectors and matrices

In Section 1.4 we have explained in detail how a bijective mapping can be established between a function-valued vector defined on some domain/ROS and a matrix containing the function values on this domain. In fact, because a function-valued vector contains functions as its elements we can associate a matrix with any set of functions. Here, we will apply this to the current scenario by associating a matrix with each function-valued vector defined on the Noise-Free ROS $\Omega_k^{s\backslash\nu}$, or equivalently with the set containing the elements of the function-valued vector. We will proceed in two steps. Firstly, with each set of functions defined on $\Omega_k^{s\backslash\nu}$ (4.3.42) we associate a function-valued column vector that is also defined on $\Omega_k^{s\backslash\nu}$ by simply stacking the 'function elements' of the considered set in a column vector. Then, we associate a matrix with the function-valued column vector, and thus also with the associated set of functions. For example, the function-valued column vectors $\mathbf{r}^x[k]$, $\mathbf{r}_{\mathrm{u}}^x[k]$, and $\mathbf{r}^s[k]$ associated with the sets $\mathcal{K}_{\mathrm{t},2}^{x,2}$ (4.3.8), $\mathcal{K}_{\mathrm{u},2}^{x,2} = \mathcal{K}_{\mathrm{a},2}^{x,2}$(4.3.10), and $\mathcal{K}_2^{s,2}$ (4.3.12), are defined by:

$$\mathbf{r}^x[k] \triangleq \begin{bmatrix} r_{11}^x[k] \\ r_{12}^x[k] \\ r_{21}^x[k] \\ r_{22}^x[k] \end{bmatrix}, \quad \mathbf{r}_{\mathrm{u}}^x[k] \triangleq \begin{bmatrix} r_{11}^x[k] \\ r_{12}^x[k] \\ r_{22}^x[k] \end{bmatrix}, \quad \mathbf{r}^s[k] \triangleq \begin{bmatrix} r_{11}^s[k] \\ r_{22}^s[k] \end{bmatrix} \qquad \forall\, k \in \Omega_k^{s\backslash\nu}. \quad (4.3.57)$$

Note that for the current scenario $\mathbf{r}_{\mathrm{u}}^x[k]$ represents the set $\mathcal{K}_{\mathrm{a},2}^{x,2}$ because $\mathcal{K}_{\mathrm{a},2}^{x,2}$ is a specific instance of the uniquiefied set $\mathcal{K}_{\mathrm{u},2}^{x,2}$. We will return to this issue later on. For convenience of notation, let $\mathbb{R}_P[\Omega_k^{s\backslash\nu}]$ denote the set or space of all function-valued column vectors of length P whose elements are members of $\mathbb{R}[\Omega_k^{s\backslash\nu}]$. Then, it is clear that:

$$\mathbf{r}^x[k] \in \mathbb{R}_{M_{\mathrm{t},2}^{x,2}}[\Omega_k^{s\backslash\nu}], \quad \mathbf{r}_{\mathrm{u}}^x[k] \in \mathbb{R}_{M_{\mathrm{u},2}^{x,2}}[\Omega_k^{s\backslash\nu}], \quad \text{and} \quad \mathbf{r}^s[k] \in \mathbb{R}_S[\Omega_k^{s\backslash\nu}] \qquad (4.3.58)$$

with $M_{\mathrm{t},2}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{t},2}^{x,2}\right|$, $M_{\mathrm{u},2}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{u},2}^{x,2}\right| = M_{\mathrm{a},2}^{x,2}$, and $S = 2$.

Due to the bijective mapping between functions and row vectors that we have introduced earlier, a function-valued column vector associated with a certain set of functions can be associated in the same manner with the set of row vectors that are associated with the functions in the function-valued column vector. For example, the mappings between $\mathbf{r}^x[k]$, $\mathbf{r}_{\mathrm{u}}^x[k]$, and $\mathbf{r}^s[k]$ on the one hand, and the sets $\mathcal{K}_{\mathrm{t},2}^{x,2}$, $\mathcal{K}_{\mathrm{u},2}^{x,2}$ and $\mathcal{K}_2^{s,2}$ on the other hand, at the same time define equivalent mappings between $\mathbf{r}^x[k]$, $\mathbf{r}_{\mathrm{u}}^x[k]$, and $\mathbf{r}^s[k]$ on the one hand and the sets $\tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2}$, $\tilde{\mathcal{K}}_{\mathrm{u},2}^{x,2}$, and $\tilde{\mathcal{K}}_2^{s,2}$ of row vectors defined in (4.3.46)-(4.3.48) on the other. This way, a bijective mapping is established between sets of row vectors on the one hand and matrices on the other hand in a truly natural manner. In fact, this directly leads us to the second step referred to above. Thus, at the same time we can associate a *subspace matrix* $\mathbf{C}^x$ with the three related quantities $\mathbf{r}^x[k]$, $\mathcal{K}_{\mathrm{t},2}^{x,2}$ and $\tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2}$. This matrix is obtained by stacking the sensor correlation column vectors in the set $\{\mathbf{r}^x[k]\}_{k\in\Omega_k^{s\backslash\nu}}$ with $\Omega_k^{s\backslash\nu}$ along each other, or equivalently by stacking the sensor correlation row vectors $\tilde{\mathbf{r}}_{11}^x$, $\tilde{\mathbf{r}}_{12}^x$, $\tilde{\mathbf{r}}_{21}^x$, and $\tilde{\mathbf{r}}_{22}^x$ on top of each other as follows:

$$\mathbf{C}^x \triangleq \begin{bmatrix} \mathbf{r}^x[k^1] & \cdots & \mathbf{r}^x[k^N] \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{r}}_{11}^x \\ \tilde{\mathbf{r}}_{12}^x \\ \tilde{\mathbf{r}}_{21}^x \\ \tilde{\mathbf{r}}_{22}^x \end{bmatrix} = \begin{bmatrix} r_{11}^x[k^1] & \cdots & r_{11}^x[k^N] \\ r_{12}^x[k^1] & \cdots & r_{12}^x[k^N] \\ r_{21}^x[k^1] & \cdots & r_{21}^x[k^N] \\ r_{22}^x[k^1] & \cdots & r_{22}^x[k^N] \end{bmatrix} \in \mathbb{R}_{M_{\mathrm{t},2}^{x,2}}^N, \quad (4.3.59)$$

where N is defined in (4.2.17). Hence, we have established a bijective mapping between four associated quantities, viz. the set $\mathcal{K}_{\mathrm{t},2}^{x,2}$ of sensor correlation functions, the set $\tilde{\mathcal{K}}_{\mathrm{t},2}^{x,2}$ of

associated sensor correlation row vectors, the associated sensor function-valued vector $\mathbf{r}^x[k]$, and finally the matrix $\mathbf{C}^x$ containing the sensor correlation values on the considered ROS arranged in a logical and specific manner; see also Fig. 1.11. In the same way, we establish a bijective mapping between $\mathcal{K}_{\mathrm{u},2}^{x,2}$, $\tilde{\mathcal{K}}_{\mathrm{u},2}^{x,2}$, $\mathbf{r}_{\mathrm{u}}^x[k]$, and the uniquiefied version $\mathbf{C}_{\mathrm{u}}^x$ of $\mathbf{C}^x$:

$$\mathbf{C}_{\mathrm{u}}^x \triangleq \begin{bmatrix}\mathbf{r}_{\mathrm{u}}^x[k^1] & \cdots & \mathbf{r}_{\mathrm{u}}^x[k^N]\end{bmatrix} = \begin{bmatrix}\tilde{\mathbf{r}}_{11}^x \\ \tilde{\mathbf{r}}_{12}^x \\ \tilde{\mathbf{r}}_{22}^x\end{bmatrix} = \begin{bmatrix}r_{11}^x[k^1] & \cdots & r_{11}^x[k^N] \\ r_{12}^x[k^1] & \cdots & r_{12}^x[k^N] \\ r_{22}^x[k^1] & \cdots & r_{22}^x[k^N]\end{bmatrix} \in \mathbb{R}_{M_{\mathrm{u},2}^{x,2}}^N . \quad (4.3.60)$$

Finally, we establish a bijective mapping between the set $\mathcal{K}_2^{s,2}$ of source auto-correlation functions, the set $\tilde{\mathcal{K}}_2^{s,2}$ of associated source auto-correlation row vectors, the associated source function-valued vector $\mathbf{r}^s[k]$, and the matrix $\mathbf{C}^s$ containing the source auto-correlation values on the considered ROS:

$$\mathbf{C}^s \triangleq \begin{bmatrix}\mathbf{r}^s[k^1] & \cdots & \mathbf{r}^s[k^N]\end{bmatrix} = \begin{bmatrix}\tilde{\mathbf{r}}_{11}^s \\ \tilde{\mathbf{r}}_{22}^s\end{bmatrix} = \begin{bmatrix}r_{11}^s[k^1] & \cdots & r_{11}^s[k^N] \\ r_{22}^s[k^1] & \cdots & r_{22}^s[k^N]\end{bmatrix} \in \mathbb{R}_S^N . \quad (4.3.61)$$

Note that in definitions (4.3.59), (4.3.60), and (4.3.61) the lag index indexes the columns of the involved matrix. Therefore, it is in the superscript position.

Clearly, the concepts associated with the various bijectively mapped quantities are related bijectively as well. Suppose that the functions in some set $\mathcal{F} \triangleq \{f_i[k]\}_{1\le i\le M}$ are defined on some ROS $\mathcal{T}$ with $|\mathcal{T}| \triangleq N$. As above, with these functions we associate the row vectors in the set $\tilde{\mathcal{F}} \triangleq \{\tilde{\mathbf{f}}_i\}_{1\le i\le M}$. The function-valued column vector $\mathbf{f}[k]$ is defined by stacking the functions on top of each other in a vector; see Fig. 1.11. Likewise, the matrix $\mathbf{F}$ associated with $\mathbf{f}[k]$ is defined by stacking the row vectors on top of each other in a matrix. Now we consider the relations between the fundamental subspaces of $\mathbf{f}[k]$ and $\mathbf{F}$.

Firstly, the linear span $\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[k]\big)$ of the functions in the function-valued column vector argument $\mathbf{f}[k]$ of $\mathcal{L}_{\mathrm{r}}\big(\,\cdot\,\big)$, i.e. the 'linear span in the row direction', can be mapped bijectively to the row range of $\mathbf{F}$ as follows:

$$\boxed{\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[k]\big) \triangleq \mathcal{L}\big(\{f_i[k]\}_{1\le i\le M}\big) = \mathcal{L}\big(\mathcal{F}\big) \doteqdot \mathcal{L}\big(\tilde{\mathcal{F}}\big) \triangleq \mathcal{L}\big(\{\tilde{\mathbf{f}}_i\}_{1\le i\le M}\big) \equiv \mathcal{R}_{\mathrm{r}}\left(\mathbf{F}\right).} \quad (4.3.62)$$

Secondly, the linear span $\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[k]\big)$ of the column vectors obtained by evaluating the argument $\mathbf{f}[k]$ of $\mathcal{L}_{\mathrm{c}}\big(\,\cdot\,\big)$ at the N lags in $\mathcal{T}$, i.e. the 'linear span in the column direction' equals the column range of $\mathbf{F}$:

$$\boxed{\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[k]\big) \triangleq \mathcal{L}\big(\{\mathbf{f}[k]\}_{k\in\mathcal{T}}\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{F}\right).} \quad (4.3.63)$$

Thirdly, the right null space of $\mathbf{f}[k]$ can be mapped bijectively to that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[k]\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{F}\big).} \quad (4.3.64)$$

Finally, the left null space of $\mathbf{f}[k]$ equals that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{F}\big).} \quad (4.3.65)$$

Because of these relations the dimensions of the spaces in the bijections are equal, e.g.:

$$\boxed{\begin{aligned}&\dim\Big(\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[k]\big)\Big) = \dim\Big(\mathcal{L}\big(\{f_i[n]\}_{1\le i\le M}\big)\Big) = \dim\Big(\mathcal{L}\big(\mathcal{F}\big)\Big) = \dim\Big(\mathcal{L}\big(\tilde{\mathcal{F}}\big)\Big) \\ &= \dim\Big(\mathcal{L}\big(\{\tilde{\mathbf{f}}_i\}_{1\le i\le M}\big)\Big) = \dim\Big(\mathcal{R}_{\mathrm{r}}\left(\mathbf{F}\right)\Big) = \dim\Big(\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[k]\big)\Big) \\ &= \dim\Big(\mathcal{L}\big(\{\mathbf{f}[k]\}_{k\in\mathcal{T}}\big)\Big) = \dim\Big(\mathcal{R}_{\mathrm{c}}\left(\mathbf{F}\right)\Big) = \operatorname{rank}(\mathbf{F}) = \operatorname{rank}\left(\mathbf{f}[k]\right).\end{aligned}} \quad (4.3.66)$$

4.3.6.2 Expressing sensor correlation function-valued vector and matrix in mixing matrix elements and source auto-correlation function-valued vector and matrix

Now, before moving on to the actual derivation of the system of equations, we will first express the function-valued sensor correlation vector $\mathbf{r}^x[k]$ in terms of the elements of the mixing matrix $\mathbf{A}$ and the function-valued source auto-correlation vector $\mathbf{r}^s[k]$. Then, we do something similar for the associated sensor correlation matrix. To start with, we use (4.3.3) to express $\mathbf{r}^x[k]$ in terms of $\mathbf{A}$ and $\mathbf{r}^s[k]$ as follows:

$$\begin{bmatrix} r^x_{11}[k] \\ r^x_{12}[k] \\ r^x_{21}[k] \\ r^x_{22}[k] \end{bmatrix} = \begin{bmatrix} a^1_1 a^1_1 \\ a^1_1 a^1_2 \\ a^1_2 a^1_1 \\ a^1_2 a^1_2 \end{bmatrix} r^s_{11}[k] + \begin{bmatrix} a^2_1 a^2_1 \\ a^2_1 a^2_2 \\ a^2_2 a^2_1 \\ a^2_2 a^2_2 \end{bmatrix} r^s_{22}[k] = \begin{bmatrix} a^1_1 a^1_1 & a^2_1 a^2_1 \\ a^1_1 a^1_2 & a^2_1 a^2_2 \\ a^1_2 a^1_1 & a^2_2 a^2_1 \\ a^1_2 a^1_2 & a^2_2 a^2_2 \end{bmatrix} \begin{bmatrix} r^s_{11}[k] \\ r^s_{22}[k] \end{bmatrix} \quad \forall\, k \in \Omega^{s\backslash\nu}_k. \tag{4.3.67}$$

We denote and define the second order Kronecker product with conjugation tuple $\mathbf{c}_2 = (c_1, c_2) = (\circ, \circ)$ of the j-th column $\mathbf{a}^j$ of $\mathbf{A}$ by:

$$\check{\mathbf{a}}^{j,\mathbf{c}_2} = \check{\mathbf{a}}^{j,\circ\circ} \triangleq \check{\mathbf{a}}^{j,2} \triangleq \mathbf{a}^j \otimes \mathbf{a}^j = \begin{bmatrix} a^j_1 \\ a^j_2 \end{bmatrix} \otimes \begin{bmatrix} a^j_1 \\ a^j_2 \end{bmatrix} = \begin{bmatrix} a^j_1 a^j_1 \\ a^j_1 a^j_2 \\ a^j_2 a^j_1 \\ a^j_2 a^j_2 \end{bmatrix}, \quad j = 1, 2\,. \tag{4.3.68}$$

Applying (A.9.4) in Section A.9 to the current scenario, we can express the second order real-valued Khatri-Rao product with conjugation tuple $\mathbf{c}_2 = (c_1, c_2) = (\circ, \circ)$ of $\mathbf{A}$, which is denoted by $\mathbf{A}^2_\diamond$, as follows:

$$\mathbf{A}^{\mathbf{c}_2}_\diamond = \mathbf{A}^{\circ\circ}_\diamond \triangleq \mathbf{A}^2_\diamond \triangleq \begin{bmatrix} \check{\mathbf{a}}^{1,2} & \check{\mathbf{a}}^{2,2} \end{bmatrix} = \begin{bmatrix} \mathbf{a}^1 \otimes \mathbf{a}^1 & \mathbf{a}^2 \otimes \mathbf{a}^2 \end{bmatrix} = \begin{bmatrix} a^1_1 a^1_1 & a^2_1 a^2_1 \\ a^1_1 a^1_2 & a^2_1 a^2_2 \\ a^1_2 a^1_1 & a^2_2 a^2_1 \\ a^1_2 a^1_2 & a^2_2 a^2_2 \end{bmatrix}. \tag{4.3.69}$$

Note that in analogy with Chapter 3 we may refer to $\check{\mathbf{a}}^{j,2}$ and $\mathbf{A}^2_\diamond$ as 'array response vector' and 'array response matrix' respectively. As is clear from this definition, the Khatri-Rao product is defined as a column-wise Kronecker product; for more details see Section A.9. Using these notational conventions, equation (4.3.67), i.e. the function-valued column vector equivalent of (4.3.3) and (4.3.45), becomes:

$$\boxed{\mathbf{r}^x[k] = \sum_{j=1}^{2} \check{\mathbf{a}}^{j,2}\, r^s_{jj}[k] = \mathbf{A}^2_\diamond\, \mathbf{r}^s[k] \qquad \forall\, k \in \Omega^{s\backslash\nu}_k.} \tag{4.3.70}$$

Note that in the same spirit as the derivation of (3.3.15) in Section 3.3.2, this result can be derived in an intuitive manner as follows:

$$\begin{aligned} \mathbf{r}^x[k] &\triangleq \mathrm{mom}\big(\mathbf{x}[n], \mathbf{x}[n-k]\big) \triangleq E\big\{\mathbf{x}[n] \otimes \mathbf{x}[n-k]\big\} \\ &= E\left\{ \left(\sum_{j_1=1}^{2} \mathbf{a}^{j_1} s_{j_1}[n] + \boldsymbol{\nu}[n] \right) \otimes \left(\sum_{j_1=1}^{2} \mathbf{a}^{j_2} s_{j_2}[n-k] + \boldsymbol{\nu}[n-k] \right) \right\} \end{aligned}$$

$$= \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} \left[\mathbf{a}^{j_1} \otimes \mathbf{a}^{j_2}\right] E\left\{s_{j_1}[n] s_{j_2}[n-k]\right\} = \sum_{j=1}^{2} \left[\mathbf{a}^{j} \otimes \mathbf{a}^{j}\right] E\left\{s_{j}[n] s_{j}[n-k]\right\}$$

$$= \sum_{j=1}^{2} \check{\mathbf{a}}^{j,2}\, r_{jj}^{s}[k] = \mathbf{A}_{\diamond}^{2}\, \mathbf{r}^{s}[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}. \tag{4.3.71}$$

Using the bijective mapping between $\mathbf{r}^{x}[k]$ defined on $\Omega_k^{s\backslash\nu}$ on the one hand and $\mathbf{C}^{x}$ on the other hand, as well as the bijective mapping between $\mathbf{r}^{s}[k]$ defined on $\Omega_k^{s\backslash\nu}$ on the one hand and $\mathbf{C}^{s}$ on the other, these expressions can be written concisely as follows:

$$\boxed{\mathbf{C}^{x} = \mathbf{A}_{\diamond}^{2}\, \mathbf{C}^{s}\,.} \tag{4.3.72}$$

Equations (4.3.70)-(4.3.72) are natural and compact ways of representing the structure of the set of sensor correlation functions. For each lag k, the vector $\mathbf{r}^{x}[k]$ is a linear combination of $\check{\mathbf{a}}^{1,2} = \mathbf{a}^{1} \otimes \mathbf{a}^{1}$ and $\check{\mathbf{a}}^{2,2} = \mathbf{a}^{2} \otimes \mathbf{a}^{2}$. The matrix $\mathbf{A}_{\diamond}^{2}$ containing these vectors defines a linear transformation from $\mathbb{R}_S[\Omega_k^{s\backslash\nu}]$ to $\mathbb{R}_{M_{\mathrm{t,2}}^{x,2}}[\Omega_k^{s\backslash\nu}]$, in particular from the function-valued source auto-correlation vector for a certain lag to a function-valued sensor correlation vector for the same lag value. It also defines a linear transformation from $\mathbb{R}_S$ to $\mathbb{R}_{M_{\mathrm{t,2}}^{x,2}}$.

Clearly, all results in the previous paragraph can equally well be expressed in terms of their uniquiefied versions. Since in the remainder of this thesis we will mainly work with uniquiefied quantities, we briefly re-express the results above in terms of these quantities. To start with, the uniquiefied version of (4.3.67) is obtained by eliminating the third row yielding:

$$\begin{bmatrix} r_{11}^{x}[k] \\ r_{12}^{x}[k] \\ r_{22}^{x}[k] \end{bmatrix} = \begin{bmatrix} a_1^1 a_1^1 \\ a_1^1 a_2^1 \\ a_2^1 a_2^1 \end{bmatrix} r_{11}^{s}[k] + \begin{bmatrix} a_1^2 a_1^2 \\ a_1^2 a_2^2 \\ a_2^2 a_2^2 \end{bmatrix} r_{22}^{s}[k] = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 \\ a_1^1 a_2^1 & a_1^2 a_2^2 \\ a_2^1 a_2^1 & a_2^2 a_2^2 \end{bmatrix} \begin{bmatrix} r_{11}^{s}[k] \\ r_{22}^{s}[k] \end{bmatrix} \quad \forall\, k \in \Omega_k^{s\backslash\nu}. \tag{4.3.73}$$

Denoting the uniquiefied version of $\check{\mathbf{a}}^{j,2}$ in (4.3.68) by $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$, i.e.:

$$\check{\mathbf{a}}_{\mathrm{u}}^{j,\mathbf{c}_2} = \check{\mathbf{a}}_{\mathrm{u}}^{j,\circ\circ} \triangleq \check{\mathbf{a}}_{\mathrm{u}}^{j,2} \triangleq \begin{bmatrix} a_1^j a_1^j \\ a_1^j a_2^j \\ a_2^j a_2^j \end{bmatrix}, \quad j = 1, 2\,, \tag{4.3.74}$$

we can express the uniquiefied version $\mathbf{A}_{\mathrm{u},\diamond}^{2}$ of $\mathbf{A}_{\diamond}^{2}$ defined in (4.3.69) as follows:

$$\mathbf{A}_{\mathrm{u},\diamond}^{\mathbf{c}_2} = \mathbf{A}_{\mathrm{u},\diamond}^{\circ\circ} \triangleq \mathbf{A}_{\mathrm{u},\diamond}^{2} \triangleq \left[\check{\mathbf{a}}_{\mathrm{u}}^{1,2} \;\; \check{\mathbf{a}}_{\mathrm{u}}^{2,2}\right] = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 \\ a_1^1 a_2^1 & a_1^2 a_2^2 \\ a_2^1 a_2^1 & a_2^2 a_2^2 \end{bmatrix}. \tag{4.3.75}$$

Note that $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$ is a subvector of the Kronecker product $\check{\mathbf{a}}^{j,2} = \mathbf{a}^{j} \otimes \mathbf{a}^{j}$ with conjugation tuple $(\circ, \circ)$ that contains only those elements of the latter vector that are selected by the index pairs in $\mathcal{I}_{\mathrm{a},2}^{2}$, i.e. the unique elements. Likewise, $\mathbf{A}_{\mathrm{u},\diamond}^{2}$ is a submatrix of the second order Khatri-Rao product $\mathbf{A}_{\diamond}^{2}$ with conjugation tuple $(\circ, \circ)$ that contains only those rows of the latter matrix that are selected by the index pairs in $\mathcal{I}_{\mathrm{a},2}^{2}$, i.e. the unique rows. Using these notational conventions, the uniquiefied versions of (4.3.70) and (4.3.72) become:

$$\boxed{\mathbf{r}_{\mathrm{u}}^{x}[k] = \sum_{j=1}^{2} \check{\mathbf{a}}_{\mathrm{u}}^{j,2}\, r_{jj}^{s}[k] = \mathbf{A}_{\mathrm{u},\diamond}^{2}\, \mathbf{r}^{s}[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}} \tag{4.3.76}$$

and:

$$\boxed{\mathbf{C}_{\mathrm{u}}^{x} = \mathbf{A}_{\mathrm{u},\diamond}^{2}\, \mathbf{C}^{s}} \tag{4.3.77}$$

respectively.

With these concepts in place we can once again derive the system of homogeneous polynomial equations in a way that enhances our insight and clearly shows the connections with conventional subspace methods. In fact, we have already done an important part of the job by defining a matrix that can serve naturally as the subspace matrix containing all relevant information in the proper way, viz. the matrix $\mathbf{C}^{x}$ or $\mathbf{C}_{\mathrm{u}}^{x}$. In the sequel we will mainly work with $\mathbf{C}_{\mathrm{u}}^{x}$ for reasons that we have explained earlier, but in virtually all equations and results $\mathbf{C}_{\mathrm{u}}^{x}$ can by replaced by $\mathbf{C}^{x}$. We leave it to the reader to make this kind of translation if desired.

4.3.6.3 Dimensions of signal and source subspaces

Again we start our derivation by considering the dimension of the signal subspace. First note from (4.3.62) that the row range of the subspace matrix, which equals the linear span of the corresponding sensor correlation row vectors, i.e. the signal or sensor subspace, is mapped bijectively to the linear span of the sensor correlation functions:

$$\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right) = \mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},2}^{x,2}\big) \doteqdot \mathcal{L}\big(\mathcal{K}_{\mathrm{u},2}^{x,2}\big) = \mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big)\,, \tag{4.3.78}$$

where $\mathcal{K}_{\mathrm{u},2}^{x,2}$ is given by $\mathcal{K}_{\mathrm{a},2}^{x,2}$ and $\tilde{\mathcal{K}}_{\mathrm{u},2}^{x,2}$ by $\tilde{\mathcal{K}}_{\mathrm{a},2}^{x,2}$. Likewise, according to (4.3.63) the column range of the subspace matrix is mapped bijectively to the linear span of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right) = \mathcal{L}\big(\{\mathbf{r}_{\mathrm{u}}^{x}[k]\}_{k\in\Omega_{k}^{s\backslash\nu}}\big) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big)\,. \tag{4.3.79}$$

From linear algebra and (4.3.66) it follows that the dimensions of all these spaces are equal to each other and are given by the rank of the subspace matrix. Hence, the dimension of the signal subspace can be expressed in the following equivalent ways:

$$\begin{aligned} d_{2}^{x,2} &\triangleq \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^{x}) = \dim\left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right)\right) = \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},2}^{x,2}\big)\Big) = \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},2}^{x,2}\big)\Big) \\ &= \mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) = \dim\left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right)\right) = \dim\big(\mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big)\big) = \operatorname{rank}\left(\mathbf{r}_{\mathrm{u}}^{x}[k]\right) \\ &= \dim\left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^{x}\right)\right) = \dim\left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^{x}\right)\right) = \operatorname{rank}(\mathbf{C}^{x}) = \operatorname{rank}\left(\mathbf{r}^{x}[k]\right); \end{aligned} \tag{4.3.80}$$

see also (4.3.17) and (4.3.49). Hence, naturally *the dimension of the signal subspace equals the rank of the subspace matrix*.

Similar to the treatment of the ranges and rank of the subspace matrix in the previous paragraph, we now consider the ranges and rank of the source auto-correlation matrix $\mathbf{C}^{s}$ defined in (4.3.61). To start with, the row range of $\mathbf{C}^{s}$, which equals the linear span of the corresponding source auto-correlation row vectors, i.e. the source subspace, is mapped bijectively to the linear span of the source auto-correlation functions:

$$\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^{s}\right) = \mathcal{L}\big(\tilde{\mathcal{K}}_{2}^{s,2}\big) \doteqdot \mathcal{L}\big(\mathcal{K}_{2}^{s,2}\big) = \mathcal{L}_{\mathrm{r}}\big(\mathbf{r}^{s}[k]\big)\,. \tag{4.3.81}$$

Likewise, the column range of $\mathbf{C}^{s}$ is mapped bijectively to the linear span of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^{s}\right) = \mathcal{L}\big(\{\mathbf{r}^{s}[k]\}_{k\in\Omega_{k}^{s\backslash\nu}}\big) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}^{s}[k]\big)\,. \tag{4.3.82}$$

Again, the dimensions of all these spaces are equal to each other and are given by the rank of the source auto-correlation matrix. Hence, the dimension of the 'source subspace' can be expressed in the following equivalent ways:

$$\begin{aligned} d_2^{s,2} \triangleq \operatorname{rank}(\mathbf{C}^s) &= \dim\left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^s\right)\right) = \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_2^{s,2}\big)\right) = \dim\left(\mathcal{L}\big(\mathcal{K}_2^{s,2}\big)\right) \\ &= \dim\left(\mathcal{L}_{\mathrm{r}}\big(\mathbf{r}^s[k]\big)\right) = \dim\left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^s\right)\right) = \dim\left(\mathcal{L}_{\mathrm{c}}\big(\mathbf{r}^s[k]\big)\right) = \operatorname{rank}\left(\mathbf{r}^s[k]\right); \end{aligned} \quad (4.3.83)$$

see also (4.3.18) and (4.3.50). Hence, naturally *the dimension of the source subspace equals the rank of the source auto-correlation matrix*. The matrix equivalent of assumption *AS2* on page 138, which states that the source auto-correlation functions and row vectors respectively are linearly independent, is given by:

$$\tilde{\boldsymbol{\xi}}\mathbf{C}^s = \tilde{\mathbf{0}}^N \qquad \Longrightarrow \qquad \tilde{\boldsymbol{\xi}} = \begin{bmatrix} \xi^1 & \xi^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}. \quad (4.3.84)$$

Hence, from this property it follows again that $d_2^{s,2} = \operatorname{rank}(\mathbf{C}^s) = S = 2$, i.e. $\mathbf{C}^s \in \mathbb{R}_2^N$ has full rank.

As we will see in the sequel, essentially our *subspace approach to MIBI is based on the fact that we can compute or estimate the (properties of the) various subspaces of the unknown matrix $\mathbf{A}_{\mathrm{u},\diamond}^2$ from the known matrix $\mathbf{C}_{\mathrm{u}}^x$*. As we have also seen in Chapter 3, this is a major theme in all subspace-based methods. We start our exposition by showing how the signal subspace dimension $d_2^{s,2}$ depends on the mixing matrix and the source signal properties. From (4.3.77) and linear algebra [115, 150] it follows that:

$$\operatorname{rank}(\mathbf{C}_{\mathrm{u}}^x) \leq \min\big(\operatorname{rank}(\mathbf{A}_{\mathrm{u},\diamond}^2), \operatorname{rank}(\mathbf{C}^s)\big), \quad (4.3.85)$$

which again implies relation (4.3.19) between $d_2^{x,2}$ and $d_2^{s,2}$:

$$d_2^{x,2} = \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^x) \leq \operatorname{rank}(\mathbf{C}^s) = d_2^{s,2} = S = 2. \quad (4.3.86)$$

Similar results are obtained when using (4.3.72) instead of (4.3.77). Equations (4.3.77) and (4.3.85) show that the dimension $d_2^{x,2}$ of the signal subspace is determined on the one hand by the matrix $\mathbf{A}_{\mathrm{u},\diamond}^2$, which depends on the mixing matrix $\mathbf{A}$ only, and the dimension $d_2^{s,2}$ of the source subspace on the other. For the current simple scenario with $\mathbf{A} \in \mathbb{R}_2^2$, it can easily be proven that $\operatorname{rank}(\mathbf{A}_{\mathrm{u},\diamond}^2) = \operatorname{rank}(\mathbf{A}_{\diamond}^2) = \operatorname{rank}(\mathbf{A})$. Hence:

$$\operatorname{rank}(\mathbf{A}_{\mathrm{u},\diamond}^2) = \dim\left(\mathcal{L}\left(\left\{\check{\mathbf{a}}_{\mathrm{u}}^{1,2}, \check{\mathbf{a}}_{\mathrm{u}}^{2,2}\right\}\right)\right) = \operatorname{rank}(\mathbf{A}) \leq 2 \quad \text{for} \quad \mathbf{A} \in \mathbb{R}_2^2. \quad (4.3.87)$$

In general, for more complicated scenarios the rank of the involved Khatri-Rao product is related to $\mathbf{A}$ in a much more complicated manner [61, 95, 141, 141, 151, 181]. Since the rank of such a product is of paramount importance for the work in this thesis we will study this topic in more detail later on. Now, from linear algebra it is known that the rank of a matrix is unchanged upon either left or right multiplication by a nonsingular matrix [115]. Hence, since $\mathbf{C}^s$ has full rank it follows immediately from (4.3.77) that (4.3.86) can be specialized further to:

$$d_2^{x,2} = \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^x) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\,\mathbf{C}^s\big) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big), \quad (4.3.88)$$

i.e. $d_2^{x,2}$ equals the rank of the linear transformation defined by $\mathbf{A}_{\mathrm{u},\diamond}^2$ in (4.3.75), which in turn equals the dimension of the linear space spanned by the vectors $\check{\mathbf{a}}_{\mathrm{u}}^{1,2}$ and $\check{\mathbf{a}}_{\mathrm{u}}^{2,2}$ defined in (4.3.74). Summarizing, we have specialized (4.3.85) to the following result:

$$\boxed{d_2^{x,2} = \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^x) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big) \quad \equiv \quad d_2^{x,2} = \operatorname{rank}(\mathbf{C}^x) = \operatorname{rank}\big(\mathbf{A}_{\diamond}^2\big).} \quad (4.3.89)$$

As will be illustrated in later chapters, a similar relation also holds for more complicated scenarios. This explains the importance of studying the rank of a Khatri-Rao product.

4.3.6.4 Linear dependence of subspace matrix rows

Once again we have arrived at quite an essential part of our derivation. Since the number $M_{\mathrm{u},2}^{x,2}$ of rows of the subspace matrix $\mathbf{C}_{\mathrm{u}}^{x}$ in (4.3.60), which is given by $M_{\mathrm{a},2}^{x,2}$, is larger than the rank of $\mathbf{C}_{\mathrm{u}}^{x}$, i.e. the dimension of the signal subspace, we obtain the following equivalent of (4.3.21) and (4.3.52):

$$\#_{\text{rows}}\{\mathbf{C}_{\mathrm{u}}^{x}\} > \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^{x})\,, \tag{4.3.90}$$

where $\#_{\text{rows}}\{\cdot\}$ denotes the 'number of rows' operator. Hence, now we must conclude that *the rows (and columns) of the subspace matrix* $\mathbf{C}_{\mathrm{u}}^{x}$ *are linearly dependent*, which implies that $\mathbf{C}_{\mathrm{u}}^{x}$ must have a non-zero left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$. In other words, there exists a matrix $\mathbf{\Phi} \in \mathbb{R}_{Q}^{M_{\mathrm{a},2}^{x,2}}$ with Q linearly independent rows such that (see Definition C.2.4):

$$\mathbf{\Phi}\mathbf{C}_{\mathrm{u}}^{x} = \mathbf{0}_{Q}^{N} \qquad \equiv \qquad \mathbf{\Phi}\mathbf{r}_{\mathrm{u}}^{x}[k] = \mathbf{0}_{Q} \quad \forall\, k \in \Omega_{k}^{s\backslash\nu}\,. \tag{4.3.91}$$

The dimension Q of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ is given by the difference between the number of rows of $\mathbf{C}_{\mathrm{u}}^{x}$ and its rank (see the box at the bottom of page 449), which is precisely equal to the number Q derived in (4.3.24) and (4.3.54):

$$\boxed{Q = \dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)\big) = \#_{\text{rows}}\{\mathbf{C}_{\mathrm{u}}^{x}\} - \operatorname{rank}(\mathbf{C}_{\mathrm{u}}^{x}) = M_{\mathrm{a},2}^{x,2} - d_{2}^{x,2}\,.} \tag{4.3.92}$$

Hence, in the same way as Q is given by the difference between the left and right hand sides of (4.3.21) and similarly for (4.3.52), it is also given by the difference between the left and right hand sides of (4.3.90). Equation (4.3.91) is the matrix-vector equivalent of (4.3.22), (4.3.25), (4.3.53), and (4.3.55). In fact, defining the row vectors:

$$\tilde{\boldsymbol{\varphi}}_{q} \triangleq \begin{bmatrix}\varphi_{q}^{11} & \varphi_{q}^{12} & \varphi_{q}^{22}\end{bmatrix} \in \mathbb{R}^{M_{\mathrm{a},2}^{x,2}} \qquad \forall\, q \in \mathcal{Q}\,, \tag{4.3.93}$$

and stacking them in the matrix $\mathbf{\Phi}$ as follows:

$$\mathbf{\Phi} \triangleq \begin{bmatrix}\tilde{\boldsymbol{\varphi}}_{1}\\ \vdots \\ \tilde{\boldsymbol{\varphi}}_{Q}\end{bmatrix} = \begin{bmatrix}\varphi_{1}^{11} & \varphi_{1}^{12} & \varphi_{1}^{22}\\ \vdots & \vdots & \vdots \\ \varphi_{Q}^{11} & \varphi_{Q}^{12} & \varphi_{Q}^{22}\end{bmatrix} \in \mathbb{R}_{Q}^{M_{\mathrm{a},2}^{x,2}}\,, \tag{4.3.94}$$

this equivalence is clear immediately. For example, the q-th row of $\mathbf{\Phi}\mathbf{C}_{\mathrm{u}}^{x}$ in (4.3.91) is given by $\tilde{\boldsymbol{\varphi}}_{q}\,\mathbf{C}_{\mathrm{u}}^{x}$, which equals the left hand side of (4.3.53) and (4.3.55). Clearly, all rows $\{\tilde{\boldsymbol{\varphi}}_{q}\}_{q\in\mathcal{Q}}$ belong to the left null space of $\mathbf{C}_{\mathrm{u}}^{x}$, i.e.:

$$\tilde{\boldsymbol{\varphi}}_{q} \in \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) \qquad \forall\, q \in \mathcal{Q}\,. \tag{4.3.95}$$

In order to obtain a system of equations with as many linearly independent equations as possible, the vectors in the set $\{\tilde{\boldsymbol{\varphi}}_{q}\}_{q\in\mathcal{Q}}$ should be chosen such that they span $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ completely. For a matrix $\mathbf{\Phi}$ that contains such as set of rows, we write symbolically (see page xxi):

$$\boxed{\mathbf{\Phi} \twoheadrightarrow \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) \doteqdot \big(\mathcal{R}_{\mathrm{c}}\,(\mathbf{C}_{\mathrm{u}}^{x})\big)^{\perp}\,.} \tag{4.3.96}$$

The indicated isomorphism follows from property *FSP1* on page 450, which states that the left null space of a matrix is isomorphic to the orthogonal complement of its column space.

4.3.6.5 Deriving the system of equations by exploiting nonsingularity of source auto-correlation matrix

Similarly to the previous sections, the next main part of our argument leading to the desired system of equations amounts to combining (4.3.77) and (4.3.91), and exploiting assumption *AS2* which implies that the source auto-correlation matrix $\mathbf{C}^s$ has full rank. Expressing (4.3.91) in terms of $\mathbf{C}^s$ and the uniquiefied Khatri-Rao product matrix $\mathbf{A}^2_{\mathrm{u},\diamond}$ by substituting (4.3.77) yields:

$$\mathbf{\Phi}\mathbf{C}^x_{\mathrm{u}} = \mathbf{\Phi}\mathbf{A}^2_{\mathrm{u},\diamond}\,\mathbf{C}^s = \mathbf{0}^N_Q\,. \tag{4.3.97}$$

Because $\mathbf{C}^s$ has full rank due to assumption *AS2*, its pseudo-inverse $(\mathbf{C}^s)^\dagger = (\mathbf{C}^s)^T(\mathbf{C}^s(\mathbf{C}^s)^T)^{-1}$ exists. Hence, we can multiply both sides (4.3.97) from the right by $(\mathbf{C}^s)^\dagger$ in order to obtain the following system of equations:

$$\mathbf{\Phi}\mathbf{A}^2_{\mathrm{u},\diamond} = \mathbf{0}^2_Q \qquad \equiv \qquad \mathbf{\Phi}\,\check{\mathbf{a}}^{j,2}_{\mathrm{u}} = \mathbf{0}, \quad j = 1,2\,, \tag{4.3.98}$$

which equals the system in (4.3.30). Hence, again from this point on the derivation of the system of homogeneous polynomial equations is exactly the same as that in Section 4.3.4 from (4.3.30) on.

4.3.6.6 Concise enlightening derivation of system from subspace matrix

An alternative and enlightening derivation in the same spirit as the derivations given in Sections 3.2.1.4 and 3.3.4 of Chapter 3 follows immediately from (4.3.77). Firstly, we prove that $\mathbf{C}^x_{\mathrm{u}}$ and $\mathbf{A}^2_{\mathrm{u},\diamond}$ have the same column spaces and thus also the same left null spaces. Then we use these properties to obtain the system of homogeneous polynomial equations in a very concise manner. We start by showing that the column range of the subspace matrix $\mathbf{C}^x_{\mathrm{u}}$, i.e. the (uniquiefied) signal subspace, equals the column range of the (uniquiefied) second order Khatri-Rao product $\mathbf{A}^2_{\mathrm{u},\diamond}$ of the mixing matrix $\mathbf{A}$, i.e. the linear span of the (uniquiefied) array response vectors, whenever the source auto-correlation matrix $\mathbf{C}^s$ has full rank (which is the case in our problem definition). For reference purposes, we summarize this important result explicitly in the following theorem.

Theorem 4.3.1. Column range of $\mathbf{C}^x_{\mathrm{u}}$ equals column range of $\mathbf{A}^2_{\mathrm{u},\diamond}$.
If the source auto-correlation matrix $\mathbf{C}^s$ has full rank, then the column range of the subspace matrix $\mathbf{C}^x_{\mathrm{u}}$ equals the column range of the second order Khatri-Rao product $\mathbf{A}^2_{\mathrm{u},\diamond}$ of the mixing matrix $\mathbf{A}$:

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^x_{\mathrm{u}}\right) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{A}^2_{\mathrm{u},\diamond}\right)\,.} \tag{4.3.99}$$

Proof. We prove this assertion using (4.3.77). From this equation it follows directly that

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^x_{\mathrm{u}}\right) \subseteq \mathcal{R}_{\mathrm{c}}\left(\mathbf{A}^2_{\mathrm{u},\diamond}\right)\,.$$

In addition, multiplying both sides of (4.3.77) from the right by $(\mathbf{C}^s)^\dagger$ yields the equality $\mathbf{A}^2_{\mathrm{u},\diamond} = \mathbf{C}^x_{\mathrm{u}}(\mathbf{C}^s)^\dagger$, which implies that:

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{A}^2_{\mathrm{u},\diamond}\right) \subseteq \mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^x_{\mathrm{u}}\right)\,.$$

These two equations immediately imply (4.3.99). □

Corollary 4.3.2. Left null space of $\mathbf{C}_{\mathrm{u}}^{x}$ equals left null space of $\mathbf{A}_{\mathrm{u},\diamond}^{2}$.
Under the assumption(s) of Theorem 4.3.1 the left null space of $\mathbf{C}_{\mathrm{u}}^{x}$ equals the left null space of $\mathbf{A}_{\mathrm{u},\diamond}^{2}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big)\,.} \tag{4.3.100}$$

Proof. Applying property *FSP1* on page 450 to both sides of (4.3.99) immediately proves the assertion. □

As we alluded to already earlier, our subspace approach to MIBI is based on the fact that we can compute/estimate the various subspaces of the unknown generalized array response matrix $\mathbf{A}_{\mathrm{u},\diamond}^{2}$ containing the generalized array response vectors from the known subspace matrix $\mathbf{C}_{\mathrm{u}}^{x}$. We will now demonstrate that we can directly derive the system of homogeneous polynomial equations by using Corollary 4.3.2 together with the trivial property that the left null space of any matrix is orthogonal to the column range of the same matrix. Firstly, Corollary 4.3.2 implies that the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big)$ of the unknown matrix $\mathbf{A}_{\mathrm{u},\diamond}^{2}$ equals the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ of the known matrix $\mathbf{C}_{\mathrm{u}}^{x}$, and thus can be computed from the subspace matrix. Secondly, applying the triviality to $\mathbf{A}_{\mathrm{u},\diamond}^{2}$ yields:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big) \quad \overset{(4.3.100)}{\Longrightarrow} \quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big)\,.} \tag{4.3.101}$$

Now realizing that the columns of $\mathbf{A}_{\mathrm{u},\diamond}^{2}$, i.e. the (uniquiefied) generalized array response vectors, form a basis for $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big)$, and defining the matrix $\boldsymbol{\Phi}$ such that its rows form a basis for $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$, it finally follows that:

$$\boldsymbol{\Phi}\mathbf{A}_{\mathrm{u},\diamond}^{2} = \mathbf{0}_{Q}^{2}\,, \tag{4.3.102}$$

which equals the systems in (4.3.30) and (4.3.98). Hence, once more from this point on the derivation of the system of homogeneous polynomial equations is exactly the same as that in Section 4.3.4 from (4.3.30) on.

The few steps outlined in the previous paragraph clearly reflect and constitute the essence of our subspace approach to MIBI. It is important to note that this concise derivation of the system of equations is possible by virtue of the fact that we have imposed a fixed order on the set of sensor correlation row vectors by stacking them in the chosen order on top of each other in the subspace matrix. We could have given similar concise derivations for the functional and row vector formulations in Sections 4.3.4 and 4.3.5 by also imposing a fixed order on the set of sensor correlation functions and sensor correlation row vectors respectively, and defining and employing appropriate concepts of left null spaces. We leave these insightful derivations to the reader. Finally we remark that the order in which the sensor correlation row vectors are stacked on top of each other in the subspace matrix is not important because for a different order a completely equivalent system can be derived and completely equivalent conclusions can be drawn. The same remark applies to the order in which the elements of the function-valued vectors $\mathbf{r}^{x}[k]$ and $\mathbf{r}^{s}[k]$ in (4.3.57) are stacked. Obviously, if we would choose a different order, the transformation matrix from $\mathbf{C}^{s}$ to $\mathbf{C}_{\mathrm{u}}^{x}$, or equivalently from $\mathbf{r}^{s}[k]$ to $\mathbf{r}_{\mathrm{u}}^{x}[k]$, would be different from the one in (4.3.75) in that its columns and/or rows would be ordered differently. Finally we note that *in the derivation above we could have used $\mathbf{r}_{\mathrm{u}}^{x}[k]$ instead of $\mathbf{C}_{\mathrm{u}}^{x}$ because of the bijective mapping defined between them.*

In the same spirit as explained at the end of Section 4.3.4, we still obtain a valid system of equations if the matrix $\mathbf{\Phi}$ is replaced by a matrix containing arbitrary (linearly independent) linear combinations of its rows. More specifically, let $\mathbf{\Upsilon} \triangleq \left[\alpha_p^q\right] \in \mathbb{C}_Q^Q$ be any full rank matrix. Then, the rows in (4.3.93) may be replaced by the following linear combinations:

$$\tilde{\boldsymbol{\varrho}}_q \triangleq \sum_{q \in \mathcal{Q}} \alpha_p^q \tilde{\boldsymbol{\varphi}}_q \qquad \forall\, p \in \mathcal{Q}\,, \tag{4.3.103}$$

or, equivalently, the matrix $\mathbf{\Phi}$ may be replaced by $\mathbf{\Phi}$ left-multiplied by $\mathbf{\Upsilon}$:

$$\mathbf{\Psi} \triangleq \mathbf{\Upsilon}\mathbf{\Phi}\,.$$

Clearly, also if rank-deficient linear transformations are considered the obtained system of equations is still valid, but in this case it is not equivalent to the original system any more because information is lost. If it is required that the transformed system has the same solutions as the original one, then $\mathbf{\Psi}$ must satisfy (see (4.3.96)):

$$\mathbf{\Psi} \circledcirc \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^x[k]\big) \doteqdot \big(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^x\right)\big)^{\perp}\,,$$

i.e. its rows must span the left null space of $\mathbf{C}_{\mathrm{u}}^x$, which equals that of $\mathbf{r}_{\mathrm{u}}^x[k]$.

4.3.6.7 Writing the system of equations in matrix-vector notation

We conclude this section by writing the system of equations $\{f_q(\mathbf{z}) = 0\}_{q \in \mathcal{Q}}$ that we have derived in various ways in the last three sections in matrix-vector notation. First note that using (4.3.93) each function $f_q(\mathbf{z})$ in (4.3.31) can we written as follows:

$$\boxed{f_q(\mathbf{z}) \triangleq \sum_{(i_1,i_2) \in \mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} z_{i_1} z_{i_2} = \tilde{\boldsymbol{\varphi}}_q\, \check{\mathbf{z}}_{\mathrm{u}}^2 = \tilde{\boldsymbol{\varphi}}_q\, \mathbf{w}_{\mathrm{u}}^2(\mathbf{z}) \quad \forall\, \mathbf{z} \in \mathbb{C}_2, \quad \forall\, q \in \mathcal{Q}\,,} \tag{4.3.104}$$

where:

$$\check{\mathbf{z}}_{\mathrm{u}}^2 \equiv \mathbf{w}_{\mathrm{u}}^2(\mathbf{z}) \triangleq \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_2 z_2 \end{bmatrix}\,. \tag{4.3.105}$$

Compare this definition with (4.3.74). Defining the column vector $\mathbf{f}(\mathbf{z})$ containing the functions $\{f_q(\mathbf{z})\}_{q \in \mathcal{Q}}$ as:

$$\mathbf{f}(\mathbf{z}) \triangleq \begin{bmatrix} f_1(\mathbf{z}) \\ \vdots \\ f_Q(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} f_1(z_1, z_2) \\ \vdots \\ f_Q(z_1, z_2) \end{bmatrix}\,, \tag{4.3.106}$$

system (4.3.33) can finally be written in the following manner:

$$\boxed{\mathbf{f}(\mathbf{z}) = \mathbf{\Phi}\, \check{\mathbf{z}}_{\mathrm{u}}^2 = \mathbf{\Phi}\, \mathbf{w}_{\mathrm{u}}^2(\mathbf{z}) = \mathbf{0}_Q\,,} \tag{4.3.107}$$

From this equation and (4.3.96) it follows that *the problem of solving the system* $\mathbf{f}(\mathbf{z}) = \mathbf{0}$ *is equivalent to finding all solutions for* $\mathbf{z}$ *such that the structured vector* $\check{\mathbf{z}}_{\mathrm{u}}^2$*, which is a function* $\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})$ *of* $\mathbf{z}$ *and has the same structure as the columns of* $\mathbf{A}_{\mathrm{u},\diamond}^2$*, belongs to the orthogonal complement of the left null space* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big)$ *of* $\mathbf{A}_{\mathrm{u},\diamond}^2$ *that can be computed from the known matrix* $\mathbf{C}_{\mathrm{u}}^{x,2}$ *because* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big)$ *equals* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x,2}\big)$*. Equivalently,* $\check{\mathbf{z}}_{\mathrm{u}}^2$ *belongs to the column range* $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big)$ *of* $\mathbf{A}_{\mathrm{u},\diamond}^2$.

In order to keep a clear understanding of the results obtained so far in this chapter, we summarize them in high-level algorithmic form in Alg. 4.2.

Algorithm 4.2 Overview of results obtained so far.

1: Compute/estimate the sensor correlation functions for several 'noise-free' lags;

2: Arrange these values in a subspace matrix $\mathbf{C}^x$ or its uniquiefied version $\mathbf{C}^x_{\mathrm{u}}$;

3: Compute a matrix $\boldsymbol{\Phi}$ whose rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}}$ have the form $\tilde{\boldsymbol{\varphi}}_q \triangleq \begin{bmatrix}\varphi_q^{11} & \varphi_q^{12} & \varphi_q^{22}\end{bmatrix}$ and span the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^x_{\mathrm{u}}\big)$ of $\mathbf{C}^x_{\mathrm{u}}$, which equals that of $\mathbf{r}^x_{\mathrm{u}}[k]$:

$$\boldsymbol{\Phi} \mathrel{\circ\!\!-} \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^x_{\mathrm{u}}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) \doteq \big(\mathcal{R}_{\mathrm{c}}\,(\mathbf{C}^x_{\mathrm{u}})\,\big)^{\perp};$$

4: With each row associate a homogeneous polynomial function of degree two:

$$f_q(z_1,z_2) \triangleq \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}\,;$$

5: The following system remains to be solved for the columns of the mixing matrix:

$$\{f_q(z_1,z_2)=0\}_{q\in\mathcal{Q}} \qquad \equiv \qquad \mathbf{f}(\mathbf{z}) = \boldsymbol{\Phi}\,\check{\mathbf{z}}^2_{\mathrm{u}} = \mathbf{0}_Q\,. \tag{4.3.108}$$

4.3.7 Indexing schemes and ordering

In the previous section, we have encountered several vectors, vector-like quantities, and matrices, whose elements are indexed by index pairs, and have their elements arranged in some arbitrarily chosen fixed order. For example, the order in which the elements of vectors like $\mathbf{r}^x[k]$ and $\check{\mathbf{a}}^{j,2}$ defined in (4.3.57) and (4.3.68) respectively, and the rows of matrices like $\mathbf{C}^x$ and $\mathbf{A}^2_{\diamond}$ defined in (4.3.59) and (4.3.69) respectively are arranged is prescribed by the order in which the elements of the index set $\mathcal{I}^2_{\mathrm{t},2}$ defined in (4.3.4) are listed, viz. $(1,1),(1,2),(2,1)$, $(2,2)$; in other words, we consider $\mathcal{I}^2_{\mathrm{t},2}$ as being an ordered set. Likewise, as to the uniquiefied versions of these quantities, the order in which the elements of vectors like $\mathbf{r}^x_{\mathrm{u}}[k]$ and $\check{\mathbf{a}}^{j,2}_{\mathrm{u}}$ defined in (4.3.57) and (4.3.74) respectively, and the rows of matrices like $\mathbf{C}^x_{\mathrm{u}}$ and $\mathbf{A}^2_{\mathrm{u},\diamond}$ defined in (4.3.60) and (4.3.75) respectively are arranged is prescribed in the same manner by the 'ordered index set' $\mathcal{I}^2_{\mathrm{a},2}$ defined in (4.3.6), viz. $(1,1),(1,2),(2,2)$. As we have remarked in Section 4.3.6.6, the concise derivation of the system of equations given there was possible by virtue of the fact that we had imposed a fixed order on the rows of the subspace matrix, and we could have done the same for the functional and row vector formulations of Sections 4.3.4 and 4.3.5 respectively. For this and other reasons, it is important to explain clearly how the indexed elements of the type of quantity referred to above can be accessed. Note that the indexing by pairs as exhibited in the examples above is not restricted to the 'row direction', but may also be used in the 'column direction'. Examples of quantities with such column indexing by pairs are row vectors like $\tilde{\boldsymbol{\varphi}}_q$ defined in (4.3.93) and matrices like $\boldsymbol{\Phi}$ defined in (4.3.94).

Accessing the elements of quantities whose elements are indexed by index pairs, and have their elements arranged in a fixed order, essentially can be done in two different logical and consistent manners. Firstly, in accordance with our main notational conventions outlined in Appendix A we can use a single sub- or superscript index to denote an element of a column-

or row-like vector respectively. In order to highlight and denote the fact that a certain quantity has elements that are indexed by index pairs and are arranged in some fixed order, we write the sub- or superscript index outside square brackets. For example, the third elements of $\mathbf{r}^x[k]$ and $\mathbf{r}_{\mathrm{u}}^x[k]$ defined in (4.3.57) are accessed and denoted by $[\mathbf{r}^x[k]]_3 = r_{21}^x[k]$ and $[\mathbf{r}_{\mathrm{u}}^x[k]]_3 = r_{22}^x[k]$ respectively, the third elements of $\check{\mathbf{a}}^{j,2}$ and $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$ defined in (4.3.68) and (4.3.74) respectively are denoted by $[\check{\mathbf{a}}^{j,2}]_3 = a_2^j a_1^j$ and $[\check{\mathbf{a}}_{\mathrm{u}}^{j,2}]_3 = a_2^j a_2^j$ respectively, the second and third elements of $\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})$ defined in (4.3.105) are denoted by $[\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_2 = z_1 z_2$ and $[\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_3 = z_2 z_2$ respectively, and so on.

Secondly, we can directly use the index pairs. For example, with this convention the third elements of $\mathbf{r}^x[k]$ and $\mathbf{r}_{\mathrm{u}}^x[k]$ are denoted by $[\mathbf{r}^x[k]]_{21} = r_{21}^x[k]$ and $[\mathbf{r}_{\mathrm{u}}^x[k]]_{22} = r_{22}^x[k]$ respectively, the third elements of $\check{\mathbf{a}}^{j,2}$ and $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$ by $[\check{\mathbf{a}}^{j,2}]_{21} = a_2^j a_1^j$ and $[\check{\mathbf{a}}_{\mathrm{u}}^{j,2}]_{22} = a_2^j a_2^j$ respectively, the second and third elements of $\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})$ by $[\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{12} = z_1 z_2$ and $[\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{22} = z_2 z_2$ respectively, and so on. This notation is logical and consistent in view of the way the elements of the involved quantities are indexed and/or constructed, but not in view of the 'indexing one dimension by one index' convention discussed in Section A.1.

In the sequel, we will encounter situations in which it is convenient and desirable to use both accessing methods. It will be clear from the context which method is most suitable for a particular situation. However, since both methods are logically consistent it is always possible to use both methods. We can make the logical consistence more evident by explicitly defining a one-to-one mapping between the p-th element of a quantity and the index pair $(i_1, i_2)_p$ of the corresponding element. For example, if the index set $\mathcal{I}_{\mathrm{t},2}^2$ is considered to be ordered and used for indexing quantities in its prescribed order $(1,1), (1,2), (2,1), (2,2)$, this mapping is given by:

$$\begin{aligned} 1 &\longleftrightarrow (i_1, i_2)_1 = (1,1)\,; \\ 2 &\longleftrightarrow (i_1, i_2)_2 = (1,2)\,; \\ 3 &\longleftrightarrow (i_1, i_2)_3 = (2,1)\,; \\ 4 &\longleftrightarrow (i_1, i_2)_4 = (2,2)\,. \end{aligned} \tag{4.3.109}$$

Similarly, if index set $\mathcal{I}_{\mathrm{a},2}^2$ is considered to be ordered and used for indexing quantities in its prescribed order $(1,1), (1,2), (2,2)$, the mapping is given by:

$$\begin{aligned} 1 &\longleftrightarrow (i_1, i_2)_1 = (1,1)\,; \\ 2 &\longleftrightarrow (i_1, i_2)_2 = (1,2)\,; \\ 3 &\longleftrightarrow (i_1, i_2)_3 = (2,2)\,. \end{aligned} \tag{4.3.110}$$

Employing these mappings, the examples given above become:

$$\begin{aligned} [\mathbf{r}^x[k]]_3 &= [\mathbf{r}^x[k]]_{(i_1,i_2)_3} = [\mathbf{r}^x[k]]_{21} = r_{21}^x[k]\,; \\ [\mathbf{r}_{\mathrm{u}}^x[k]]_3 &= [\mathbf{r}_{\mathrm{u}}^x[k]]_{(i_1,i_2)_3} = [\mathbf{r}_{\mathrm{u}}^x[k]]_{22} = r_{22}^x[k]\,; \\ [\check{\mathbf{a}}^{j,2}]_3 &= [\check{\mathbf{a}}^{j,2}]_{(i_1,i_2)_3} = [\check{\mathbf{a}}^{j,2}]_{21} = a_2^j a_1^j\,; \\ [\check{\mathbf{a}}_{\mathrm{u}}^{j,2}]_3 &= [\check{\mathbf{a}}_{\mathrm{u}}^{j,2}]_{(i_1,i_2)_3} = [\check{\mathbf{a}}_{\mathrm{u}}^{j,2}]_{21} = a_2^j a_2^j\,; \\ [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_2 &= [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{(i_1,i_2)_2} = [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{12} = z_1 z_2\,; \\ [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_3 &= [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{(i_1,i_2)_3} = [\mathbf{w}_{\mathrm{u}}^2(\mathbf{z})]_{22} = z_2 z_2\,. \end{aligned} \tag{4.3.111}$$

The notation $i_{p,q}$ will be used for accessing the p-th index of a numbered index pair $(i_1, i_2)_q$. Hence, $(i_1, i_2)_q$ can also be written as $(i_{1,q}, i_{2,q})$. For example, in the first example of (4.3.111), $(i_1, i_2)_3 = (i_{1,3}, i_{2,3}) = (2,1)$, i.e. $i_{1,3} = 2$ and $i_{2,3} = 1$.

Finally, we discuss an important convention regarding the indexing of elements of certain types of vectors. In the sequel we will also encounter situations in which it is convenient and desirable to index the elements of a 'plain vector', i.e. a vector whose elements would be indexed by a single index according to our standard notational convention, by index pairs according to some chosen fixed order as explained in the previous paragraphs. For example, consider the p-th column vector $\mathbf{u}^p \in \mathbb{R}_3$ in the Singular Value Decomposition of the function-valued sensor correlation vector $\mathbf{r}^x_{\mathrm{u}}[k]$ discussed in Theorem 4.3.3 of the next section:

$$\mathbf{r}^x_{\mathrm{u}}[k] \triangleq \begin{bmatrix} r^x_{11}[k] \\ r^x_{12}[k] \\ r^x_{22}[k] \end{bmatrix} \overset{SVD}{=} \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \, v_p[k] = \sum_{p=1}^{P} \sigma_p \begin{bmatrix} u^p_1 \\ u^p_2 \\ u^p_3 \end{bmatrix} v_p[k] \qquad \forall \; k \in \Omega_k^{s \backslash \nu} \,. \quad (4.3.112)$$

As is clear intuitively and will be demonstrated later on, on several occasions it is desirable to write a vector like $\mathbf{u}^p$ as $\mathbf{u}^p \triangleq [u^p_{11} \; u^p_{12} \; u^p_{22}]^T$ by identifying u^p_1 with u^p_{11}, u^p_2 with u^p_{12}, and u^p_3 with u^p_{22} respectively, where we have used one-to-one mapping (4.3.110) between the numbers 1, 2 and 3 on the one hand, and the index pairs $(i_1, i_2)_1$, $(i_1, i_2)_2$ and $(i_1, i_2)_3$ on the other hand:

$$\mathbf{u}^p = \begin{bmatrix} u^p_1 \\ u^p_2 \\ u^p_3 \end{bmatrix} \equiv \begin{bmatrix} u^p_{(i_1,i_2)_1} \\ u^p_{(i_1,i_2)_2} \\ u^p_{(i_1,i_2)_3} \end{bmatrix} = \begin{bmatrix} u^p_{(i_{1,1},i_{2,1})} \\ u^p_{(i_{1,2},i_{2,2})} \\ u^p_{(i_{1,3},i_{2,3})} \end{bmatrix} = \begin{bmatrix} u^p_{11} \\ u^p_{12} \\ u^p_{22} \end{bmatrix} . \quad (4.3.113)$$

Now we can already demonstrate why the kind of notation discussed in this paragraph may be convenient and desirable. Consider the different possible expressions for the q-th element of the Singular Value Decomposition of $\mathbf{r}^x_{\mathrm{u}}[k]$ in (4.3.112):

$$\begin{aligned} \big[\mathbf{r}^x_{\mathrm{u}}[k]\big]_q &= r^x_{(i_1,i_2)_q}[k] = r^x_{i_{1,q}i_{2,q}}[k] = \sum_{p=1}^{P} \sigma_p \, u^p_q \, v_p[k] \\ &= \sum_{p=1}^{P} \sigma_p \, u^p_{(i_1,i_2)_q} v_p[k] = \sum_{p=1}^{P} \sigma_p \, u^p_{i_{1,q}i_{2,q}} v_p[k] \qquad \forall \; k \in \Omega_k^{s \backslash \nu} \,. \end{aligned}$$

It is clear that a very natural and insightful expression for the (i_1, i_2)-th sensor correlation function is given by:

$$r^x_{i_1 i_2}[k] = \sum_{p=1}^{P} \sigma_p \, u^p_{i_1 i_2} \, v_p[k] \qquad \forall \; k \in \Omega_k^{s \backslash \nu} \,. \quad (4.3.114)$$

Note that $u^p_{i_1 i_2}$ can be seen as an element of a holor and that choosing a particular 'distribution' of sub- and superscripts can be seen as a particular manner for slicing the holor. Since both indices i_1 and i_2 of $u^p_{i_1 i_2}$ are in the subscript position, it is clear that $u^p_{i_1 i_2}$ can also be seen as an element of a column vector. Another advantage of our notation becomes clear when computing inner products. For example, consider the following inner product between two vectors $\mathbf{u}^m, \mathbf{u}^n \in \mathbb{R}_3$ of the type given in (4.3.113):

$$\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} \triangleq \sum_{q=1}^{3} u^m_q \, u^n_q = \sum_{q=1}^{3} u^m_{(i_1,i_2)_q} \, u^n_{(i_1,i_2)_q} = \sum_{(i_1,i_2) \in \mathcal{I}^2_{\mathrm{a},2}} u^m_{i_1 i_2} \, u^n_{i_1 i_2} \,, \quad (4.3.115)$$

where the employed inner product function $\langle \cdot, \cdot \rangle_{\mathrm{v}}$ is defined in Eq. (D.2.3) of Section D.2. Given the structure of the considered vectors, the latter expression is the most natural and

insightful one. As a final example, the q-th equation in the system $\mathbf{\Phi}\mathbf{C}^x = \mathbf{0}_Q^N$ given in (4.3.91) can be written as follows:

$$\tilde{\boldsymbol{\varphi}}_q \mathbf{C}_{\mathrm{u}}^x = \sum_{p=1}^{3} [\tilde{\boldsymbol{\varphi}}_q]^p \, [\mathbf{C}_{\mathrm{u}}^x]_p = \sum_{p=1}^{3} \varphi_q^{(i_1,i_2)_p} \, \tilde{\mathbf{r}}^x_{(i_1,i_2)_p} = \sum_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{a},2}} \varphi_q^{i_1 i_2} \, \tilde{\mathbf{r}}^x_{i_1 i_2} = \tilde{\mathbf{0}} \, .$$

Again, the latter expression is the most natural and insightful one.

We conclude this section by remarking that the desirability and use of the different accessing and indexing schemes discussed above will always be clear from the context. Given a certain scenario, we will often use the most natural and intuitive notation.

4.3.8 Using the SVD to find proper coefficients of the polynomials

In the previous sections we have projected the MIBI problem onto the problem of solving a system of 2-homogeneous polynomial equations for the columns of the mixing matrix $\mathbf{A}$. We have also remarked there that the coefficients of the polynomial equations can be considered as known because they can be deduced from the sensor correlation functions by means of the Singular Value Decomposition (SVD). The procedure for doing this is explained in this section and amounts to performing a subspace decomposition that is similar to that performed by conventional subspace methods; see also Chapter 3. Although it is not common in the literature, along with the standard matrix formulation of the SVD theorem [72, 115], here we also provide a formulation in functional notation. We do so because in our opinion the functional formulation provides more insight and more clearly illustrates the true functional nature of the problem. As is shown in Appendix D, the matrix and functional SVD's are related bijectively. In fact, the matrix formulation can be seen as a kind of implementation of the functional notation. In the sequel of this section, both the *'functional SVD'* of $\mathbf{r}_{\mathrm{u}}^x[k]$ and the equivalent *'standard matrix SVD'* of $\mathbf{C}_{\mathrm{u}}^x$ are used to obtain a proper set Φ (4.3.26) containing a set of coefficients for each equation in the system, or equivalently the matrix $\mathbf{\Phi}$ (4.3.94) containing a row for each equation. In Section 4.3.6, in particular Sections 4.3.6.4 and 4.3.6.6, we have seen that the rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}}$ of $\mathbf{\Phi}$ belong to the left null space $\mathcal{N}_{\mathrm{l}}(\mathbf{r}_{\mathrm{u}}^x[k]) = \mathcal{N}_{\mathrm{l}}(\mathbf{C}_{\mathrm{u}}^x)$ of $\mathbf{r}_{\mathrm{u}}^x[k]$ and $\mathbf{C}_{\mathrm{u}}^x$, which can be determined directly from the SVD of $\mathbf{r}_{\mathrm{u}}^x[k]$ or $\mathbf{C}_{\mathrm{u}}^x$ by choosing and transposing the left singular vectors that correspond to zero singular values [115]. Hence, in principle it is sufficient to determine this left null space. However, for the sake of insight and because we will also use another fundamental subspace of $\mathbf{r}_{\mathrm{u}}^x[k]$ and $\mathbf{C}_{\mathrm{u}}^x$ later on in this chapter (in Section 4.6, where the MIBI problem is projected onto a multi-matrix generalized eigenvalue problem that is complementary to the current projection onto the system of homogeneous polynomial equations), we will discuss the fully fledged Singular Value Decomposition here. We start by formulating the functional SVD and then present the standard matrix SVD.

4.3.8.1 Functional SVD of function-valued sensor correlation vector $\mathbf{r}_{\mathrm{u}}^x[k]$

The following theorem presents the functional singular value decomposition[‡] of the function-valued vector $\mathbf{r}_{\mathrm{u}}^x[k]$ understood to be defined on $\Omega_k^{s\setminus\nu}$:

[‡] See Section D.2 for the definition of the functional singular value decomposition, the rank of a function-valued vector, the relevant inner product functions, etc.

Theorem 4.3.3. SVD of function-valued sensor correlation vector.
Let $\mathbf{r}_{\mathrm{u}}^{x}[k] \in \mathbb{R}_{M_{\mathrm{u},2}^{x,2}}[\Omega_k^{s\backslash\nu}]$ *be a (uniquefied) real- and function-valued sensor correlation vector with rank* $d_2^{x,2} \triangleq \operatorname{rank}(\mathbf{r}_{\mathrm{u}}^{x}[k])$*, number of elements* $M_{\mathrm{u},2}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{u},2}^{x,2}\right|$ *and ROS* $\Omega_k^{s\backslash\nu}$ *with cardinality* $N \triangleq \left|\Omega_k^{s\backslash\nu}\right|$*. Then, there exist real-valued positive constants* $\sigma_p \in \mathbb{R}$ *for* $1 \leq p \leq P$*, column vectors* $\mathbf{u}^p \in \mathbb{R}_{M_{\mathrm{u},2}^{x,2}}$ *for* $1 \leq p \leq M_{\mathrm{u},2}^{x,2}$ *and functions* $v_p[k] \in \mathbb{R}[\Omega_k^{s\backslash\nu}]$ *for* $1 \leq p \leq N$*, such that* $\mathbf{r}_{\mathrm{u}}^{x}[k]$ *can be decomposed as follows:*

$$\boxed{\mathbf{r}_{\mathrm{u}}^{x}[k] = \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \, v_p[k] \qquad \forall \; k \in \Omega_k^{s\backslash\nu} \,,}$$

where:

- $P = \min(M_{\mathrm{u},2}^{x,2}, N)\,;$
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall \; 1 \leq m, n \leq M_{\mathrm{u},2}^{x,2}\,;$
- $\langle v_m[k], v_n[k] \rangle_{\mathrm{f}} = \delta_{mn} \quad \forall \; 1 \leq m, n \leq N\,;$
- $\sigma_1 \geq \cdots \geq \sigma_{d_2^{x,2}} > 0,\ \sigma_{d_2^{x,2}+1}, \ldots, \sigma_P = 0\,.$

This theorem is a special case of Theorem D.2.1 in Section D.2. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}$, functions $v_1[k], \ldots, v_N[k]$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors*, *right singular functions*, and *singular values* respectively of $\mathbf{r}_{\mathrm{u}}^{x}[k]$. The definitions of the employed inner product functions $\langle \cdot, \cdot \rangle_{\mathrm{v}}$ and $\langle \cdot, \cdot \rangle_{\mathrm{f}}$ are given by (D.2.3) and (D.2.4) respectively of Section D.2. For the current scenario these definitions can be given as follows. The first inner product is a function from $\mathbb{R}_{M_{\mathrm{u},2}^{x,2}} \times \mathbb{R}_{M_{\mathrm{u},2}^{x,2}}$ to $\mathbb{R}$ and is defined by (see also (4.3.115)):

$$\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} \triangleq \sum_{p=1}^{M_{\mathrm{u},2}^{x,2}} u_p^m u_p^n = \sum_{(i_1,i_2) \in \mathcal{I}_{\mathrm{u},2}^2} u_{i_1 i_2}^m \, u_{i_1 i_2}^n \,, \tag{4.3.116}$$

where the 'v' in the subscript position indicates that this inner product is of the '<u>v</u>ector type'. The second inner product is a function from $\mathbb{R}[\Omega_k^{s\backslash\nu}] \times \mathbb{R}[\Omega_k^{s\backslash\nu}]$ to $\mathbb{R}$ and is defined by:

$$\langle v_m[k], v_n[k] \rangle_{\mathrm{f}} \triangleq \sum_{k \in \Omega_k^{s\backslash\nu}} v_m[k] \, v_n[k] \,, \tag{4.3.117}$$

where the 'f' in the subscript position indicates that this inner product is of the '<u>f</u>unctional type'. The theorem can be proven by using the standard SVD theorem [72, 115] described in Section D.1 and the various bijective mappings that we have discussed earlier, i.e. those between functions and row vectors, and function-valued vectors and matrices.

For convenience and insight the set of vectors and the set of functions in the SVD are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. Since this partitioning is particularly relevant for the subspace techniques used in this chapter and the rest of the thesis, we explain the associated notation in detail. Firstly, we define the set $\mathcal{U}$ containing the left singular vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}$ of the decomposition as follows:

$$\mathcal{U} \triangleq \left\{ \mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}} \right\}.$$

This set is split into the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$, where:

$$\mathcal{U}^s \triangleq \left\{\mathbf{u}^1, \ldots, \mathbf{u}^{d_2^{x,2}}\right\}$$

contains the left singular vectors corresponding to the non-zero singular values, and:

$$\mathcal{U}^\nu \triangleq \left\{\mathbf{u}^{d_2^{x,2}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}\right\}$$

contains the left singular vectors corresponding to the zero singular values. In subspace decomposition terms the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, we define the set $\mathcal{V}$ containing the right singular functions $v_1[k], \ldots, v_N[k]$ of the decomposition as follows:

$$\mathcal{V} \triangleq \left\{v_1[k], \ldots, v_N[k]\right\}.$$

This set is split into the sets $\mathcal{V}_s$ and $\mathcal{V}_\nu$, where:

$$\mathcal{V}_s \triangleq \left\{v_1[k], \ldots, v_{d_2^{x,2}}[k]\right\}$$

contains the right singular functions corresponding to the non-zero singular values, and:

$$\mathcal{V}_\nu \triangleq \left\{v_{d_2^{x,2}+1}[k], \ldots, v_N[k]\right\}$$

contains the right singular functions corresponding to the zero singular values. Now splitting the SVD expression in Theorem 4.3.3 accordingly yields:

$$\mathbf{r}_{\mathrm{u}}^x[k] = \sum_{p=1}^{P} \sigma_p \mathbf{u}^p v_p[k] = \sum_{p=1}^{d_2^{x,2}} \sigma_p \mathbf{u}^p v_p[k] \; + \sum_{p=d_2^{x,2}+1}^{P} \sigma_p \mathbf{u}^p v_p[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}\,, \tag{4.3.118}$$

where the first summation contains only vectors and functions from $\mathcal{U}^s$ and $\mathcal{V}_s$ respectively, and the second contains only vectors and functions from $\mathcal{U}^\nu$ and $\mathcal{V}_\nu$ respectively.

Corollary 4.3.4. Reduced SVD of function-valued sensor correlation vector.
Under the assumptions formulated in Theorem 4.3.3, $\mathbf{r}_{\mathrm{u}}^x[k]$ *can be written in the following reduced form:*

$$\boxed{\mathbf{r}_{\mathrm{u}}^x[k] = \sum_{p=1}^{d_2^{x,2}} \sigma_p\, \mathbf{u}^p\, v_p[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}\,.} \tag{4.3.119}$$

Proof. Substituting $\sigma_{d_2^{x,2}+1}, \ldots, \sigma_P = 0$ into (4.3.118) directly yields the Corollary. □

Using the notation introduced in Section 4.3.7 we can now write each sensor correlation function $r_{i_1 i_2}^x[k]$ in the following natural 'functional SVD-form':

$$\boxed{r_{i_1 i_2}^x[k] = \sum_{p=1}^{d_2^{x,2}} \sigma_p\, u_{i_1 i_2}^p\, v_p[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2\,.} \tag{4.3.120}$$

Hence, the function $r_{i_1 i_2}^x[k]$ is a linear combination of the right singular functions $v_1[k], \ldots, v_{d_2^{x,2}}[k]$ with coefficients $\sigma_1 u_{i_1 i_2}^1, \ldots, \sigma_{d_2^{x,2}} u_{i_1 i_2}^{d_2^{x,2}}$. Compare this expression with (4.3.3), which is repeated here for convenience:

$$r_{i_1 i_2}^x[k] = \sum_{j=1}^{2} a_{i_1}^j a_{i_2}^j\, r_{jj}^s[k] \qquad \forall\, k \in \Omega_k^{s\backslash\nu}, \quad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2\,. \tag{4.3.121}$$

This equation says that each sensor correlation function $r^x_{i_1 i_2}[k]$ is also a linear combination of the source auto-correlation functions $r^s_{11}[k]$ and $r^s_{22}[k]$ with coefficients $a^1_{i_1} a^1_{i_2}$ and $a^2_{i_1} a^2_{i_2}$ respectively. The precise relations between the various (fundamental) subspaces can be formulated as follows (see also (D.2.6)-(D.2.9)):

$$\mathcal{L}_{\mathrm{c}}\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{C}^x_{\mathrm{u}}\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{A}^2_{\mathrm{u},\diamond}\right) = \mathcal{L}\left(\mathcal{U}^s\right); \tag{4.3.122}$$

$$\mathcal{L}_r\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) = \mathcal{L}\big(\mathcal{V}_s\big) \doteqdot \mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^x_{\mathrm{u}}\right) \subseteq \mathcal{L}_r\big(\mathbf{r}^s[k]\big) \doteqdot \mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^s\right); \tag{4.3.123}$$

$$\mathcal{N}_{\mathrm{r}}\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) = \mathcal{L}\big(\mathcal{V}_\nu\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}^x_{\mathrm{u}}\big) \supseteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}^s[k]\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}^s\big); \tag{4.3.124}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}^x_{\mathrm{u}}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^x_{\mathrm{u}}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}^2_{\mathrm{u},\diamond}\big) = \big(\mathcal{L}\left(\mathcal{U}^\nu\right)\big)^T, \tag{4.3.125}$$

See Section 4.3.8.3 for explicit expressions of a proper set Φ containing sets of coefficients for the polynomial equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}}$, or a matrix $\boldsymbol{\Phi}$ representing these coefficients, in terms of the left null space of $\mathbf{r}^x_{\mathrm{u}}[k]$ found by the SVD.

4.3.8.2 Matrix SVD of subspace matrix $\mathbf{C}^x_{\mathrm{u}}$

In this section, we present the matrix equivalent of Theorem 4.3.3, which is given by the standard Singular Value Decomposition of the subspace matrix $\mathbf{C}^x_{\mathrm{u}}$. Because the function-valued vector $\mathbf{r}^x_{\mathrm{u}}[k]$ and the subspace matrix $\mathbf{C}^x_{\mathrm{u}}$ are related bijectively, their SVD's are related bijectively as well. For details about this relation, we refer the reader to Appendix D. The following theorem presents the SVD of the subspace matrix $\mathbf{C}^x_{\mathrm{u}}$:

Theorem 4.3.5. SVD of subspace matrix.
Let $\mathbf{C}^x_{\mathrm{u}} \in \mathbb{R}^N_{M^{x,2}_{\mathrm{u},2}}$ be a (uniquefied sensor correlation) real-valued subspace matrix with rank $d^{x,2}_2 \triangleq \mathrm{rank}(\mathbf{C}^x_{\mathrm{u}})$, and with $M^{x,2}_{\mathrm{u},2}$ and N defined as in Theorem 4.3.3. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, column vectors $\mathbf{u}^p \in \mathbb{R}_{M^{x,2}_{\mathrm{u},2}}$ for $1 \leq p \leq M^{x,2}_{\mathrm{u},2}$ and row vectors $\tilde{\mathbf{v}}_p \in \mathbb{R}^N$ for $1 \leq p \leq N$, such that $\mathbf{C}^x_{\mathrm{u}}$ can be decomposed as follows:

$$\mathbf{C}^x_{\mathrm{u}} = \sum_{p=1}^{P} \sigma_p\, \mathbf{u}^p\, \tilde{\mathbf{v}}_p \equiv \mathbf{U}\boldsymbol{\Sigma}\mathbf{V},$$

where:

- $P = \min(M^{x,2}_{\mathrm{u},2}, N)$;
- $\mathbf{U} \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^{M^{x,2}_{\mathrm{u},2}}\end{bmatrix} \in \mathbb{R}^{M^{x,2}_{\mathrm{u},2}}_{M^{x,2}_{\mathrm{u},2}}$;
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall\, 1 \leq m,n \leq M^{x,2}_{\mathrm{u},2} \quad \equiv \quad \mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$;
- $\mathbf{V} \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_N\end{bmatrix} \in \mathbb{R}^N_N$;
- $\langle \tilde{\mathbf{v}}_m, \tilde{\mathbf{v}}_n \rangle_{\mathrm{v}} = \delta_{mn} \quad \forall\, 1 \leq m,n \leq N \quad \equiv \quad \mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}$;
- $\boldsymbol{\Sigma} \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_P\} \in \mathbb{R}^N_{M^{x,2}_{\mathrm{u},2}}$;
- $\sigma_1 \geq \cdots \geq \sigma_{d^{x,2}_2} > 0,\ \sigma_{d^{x,2}_2+1}, \ldots, \sigma_P = 0$.

This theorem is a special case of Theorem D.1.1 in Section D.1. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}$, row vectors $\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_N$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors*, *right singular vectors*, and *singular values* respectively of $\mathbf{C}_{\mathrm{u}}^x$. Note that now only vector type inner product functions are used. For a proof we refer the reader to standard textbooks on this topic, e.g. see [72]. The matrices $\mathbf{U}$ and $\mathbf{V}$ are square and orthonormal, i.e. $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$ and $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}$. Although the matrix $\mathbf{\Sigma}$ containing the singular values is not a square matrix, notationally it is written as $\mathrm{diag}\{\sigma_1, \ldots, \sigma_P\}$, and its off-diagonal elements are zero. The singular values $\sigma_1 \geq \cdots \geq \sigma_P$ are written along the main diagonal in descending order, and rows or columns of zeros are appended as necessary to obtain the proper dimensions. Note that only the first $d_2^{x,2}$ singular values are non-zero because $\mathrm{rank}(\mathbf{C}_{\mathrm{u}}^x) = d_2^{x,2}$. See Appendix D for details and examples.

Similarly to the functional SVD in Section 4.3.8.1, for convenience and insight the various components of the matrix SVD of $\mathbf{C}_{\mathrm{u}}^x$ are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. To start with, let $\mathbf{\Sigma}_s^s$ represent the square diagonal block of $\mathbf{\Sigma}$ that contains these non-zero singular values, i.e. $\mathbf{\Sigma}_s^s$ is the upper left block of $\mathbf{\Sigma}$ of size $d_2^{x,2} \times d_2^{x,2}$. Furthermore, let $\mathbf{\Sigma}_\nu^\nu$ represent the (not necessarily square) block of zeros of $\mathbf{\Sigma}$ to the right and below $\mathbf{\Sigma}_s^s$, i.e. $\mathbf{\Sigma}_\nu^\nu$ is the lower right block of $\mathbf{\Sigma}$ of size $(M_{\mathrm{u},2}^{x,2} - d_2^{x,2}) \times (N - d_2^{x,2})$. Thus, we have:

$$\mathbf{\Sigma}_s^s \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_{d_2^{x,2}}\} \in \mathbb{R}^{d_2^{x,2}}_{d_2^{x,2}} \qquad \text{and} \qquad \mathbf{\Sigma}_\nu^\nu \triangleq \mathbf{0}^{N-d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}}\,, \tag{4.3.126}$$

and $\mathbf{\Sigma}$ can be written as follows:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-r}_{d_2^{x,2}} \\ \mathbf{0}^{d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}} & \mathbf{\Sigma}_\nu^\nu \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-d_2^{x,2}}_{d_2^{x,2}} \\ \mathbf{0}^{d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}} & \mathbf{0}^{N-d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}} \end{bmatrix}. \tag{4.3.127}$$

The matrix $\mathbf{U}$ is split into two parts as $\mathbf{U} = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix}$. The first part $\mathbf{U}^s$ consists of columns that are multiplied by the non-zero singular values in the SVD-expression of $\mathbf{C}_{\mathrm{u}}^x$, whereas the second part $\mathbf{U}^\nu$ consists of columns that are multiplied by the zero singular values. Hence:

$$\mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 \cdots \mathbf{u}^{d_2^{x,2}}\end{bmatrix} \in \mathbb{R}^{d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}} \quad \text{and} \quad \mathbf{U}^\nu \triangleq \begin{bmatrix}\mathbf{u}^{d_2^{x,2}+1} \cdots \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}\end{bmatrix} \in \mathbb{R}^{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}}, \tag{4.3.128}$$

where $\mathbf{u}^p$ is the p-th column of $\mathbf{U}$. In subspace decomposition terms the matrices $\mathbf{U}^s$ and $\mathbf{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, the matrix $\mathbf{V}$ is split as $\mathbf{V} = \begin{bmatrix}\mathbf{V}_s \\ \mathbf{V}_\nu\end{bmatrix}$ with:

$$\mathbf{V}_s \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_{d_2^{x,2}}\end{bmatrix} \in \mathbb{R}^N_{d_2^{x,2}} \quad \text{and} \quad \mathbf{V}_\nu \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_{d_2^{x,2}+1} \\ \vdots \\ \tilde{\mathbf{v}}_N\end{bmatrix} \in \mathbb{R}^N_{N-d_2^{x,2}}\,, \tag{4.3.129}$$

where $\tilde{\mathbf{v}}_p$ is the p-th row of $\mathbf{V}$. Now splitting the SVD expression in Theorem 4.3.5 accordingly yields:

$$\mathbf{C}_{\mathrm{u}}^x = \mathbf{U}\mathbf{\Sigma}\mathbf{V} = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-d_2^{x,2}}_{d_2^{x,2}} \\ \mathbf{0}^{d_2^{x,2}}_{M_{\mathrm{u},2}^{x,2}-d_2^{x,2}} & \mathbf{\Sigma}_\nu^\nu \end{bmatrix} \begin{bmatrix}\mathbf{V}_s \\ \mathbf{V}_\nu\end{bmatrix} = \mathbf{U}^s\mathbf{\Sigma}_s^s\mathbf{V}_s + \mathbf{U}^\nu\mathbf{\Sigma}_\nu^\nu\mathbf{V}_\nu$$

$$= \sum_{p=1}^{d_2^{x,2}} \sigma_p \mathbf{u}^p \, \tilde{\mathbf{v}}_p \; + \sum_{p=d_2^{x,2}+1}^{P} \sigma_p \mathbf{u}^p \, \tilde{\mathbf{v}} \,, \tag{4.3.130}$$

where the first summation in each of the last two expressions contains only vectors from $\mathbf{U}^s$ and $\mathbf{V}_s$ respectively, and the second only from $\mathbf{U}^\nu$ and $\mathbf{V}_\nu$ respectively. Note the resemblance of the partitioning described above to (4.3.118), and the partitioning into signal and noise subspaces that is reminiscent of the subspace methods explained in Chapter 3.

Corollary 4.3.6. Reduced SVD of subspace matrix.
Under the assumptions formulated in Theorem 4.3.5, $\mathbf{C}_{\mathrm{u}}^x$ can be written in the following reduced form:

$$\boxed{\mathbf{C}_{\mathrm{u}}^x = \mathbf{U}^s \mathbf{\Sigma}_s^s \mathbf{V}_s = \sum_{p=1}^{d_2^{x,2}} \sigma_p \, \mathbf{u}^p \, \tilde{\mathbf{v}}_p \,.} \tag{4.3.131}$$

Proof. Substituting $\sigma_{d_2^{x,2}+1}, \ldots, \sigma_P = 0$ into (4.3.130) directly yields the Corollary, which clearly is the equivalent of Corollary (4.3.4). □

Using the notation introduced in Section 4.3.7 we can now write each sensor correlation row vector $\tilde{\mathbf{r}}_{i_1 i_2}^x$ in the following natural SVD-form:

$$\boxed{\tilde{\mathbf{r}}_{i_1 i_2}^x = \sum_{p=1}^{d_2^{x,2}} \sigma_p \, u_{i_1 i_2}^p \, \tilde{\mathbf{v}}_p \qquad \forall \, (i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2 \,,} \tag{4.3.132}$$

which is the row vector equivalent of (4.3.120). Hence, the row vector $\tilde{\mathbf{r}}_{i_1 i_2}^x$ is a linear combination of the row vectors $\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_{d_2^{x,2}}$ with coefficients $\sigma_1 u_{i_1 i_2}^1, \ldots, \sigma_{d_2^{x,2}} u_{i_1 i_2}^{d_2^{x,2}}$, which is consistent with the results obtained in Section 4.3.8.1. Compare (4.3.132) with (4.3.45), which is repeated here for convenience:

$$\tilde{\mathbf{r}}_{i_1 i_2}^x = \sum_{j=1}^{2} a_{i_1}^j a_{i_2}^j \, \tilde{\mathbf{r}}_{jj}^s \qquad \forall \, (i_1, i_2) \in \mathcal{I}_{\mathrm{a},2}^2 \,. \tag{4.3.133}$$

This equation says that $\tilde{\mathbf{r}}_{i_1 i_2}^x$ is also a linear combination of the source auto-correlation row vectors $\tilde{\mathbf{r}}_{11}^s$ and $\tilde{\mathbf{r}}_{22}^s$ with coefficients $a_{i_1}^1 a_{i_2}^1$ and $a_{i_1}^2 a_{i_2}^2$ respectively. Analogously to (4.3.122)–(4.3.125), the precise relationships between the various (fundamental) subspaces can be formulated as follows (see also (D.1.8)–(D.1.11)):

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^x\right) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u}}^x[k]\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{A}_{\mathrm{u},\diamond}^2\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) \,; \tag{4.3.134}$$

$$\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_{\mathrm{u}}^x\right) = \mathcal{R}_{\mathrm{r}}\left(\mathbf{V}_s\right) \doteqdot \mathcal{L}_r\big(\mathbf{r}_{\mathrm{u}}^x[k]\big) \subseteq \mathcal{R}_{\mathrm{r}}\left(\mathbf{C}^s\right) \doteqdot \mathcal{L}_r\big(\mathbf{r}^s[k]\big) \,; \tag{4.3.135}$$

$$\mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u}}^x\big) = \mathcal{R}_{\mathrm{c}}\left(\left(\mathbf{V}_\nu\right)^T\right) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u}}^x[k]\big) \supseteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}^s\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}^s[k]\big) \,; \tag{4.3.136}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^x[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big) = \big(\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^\nu\right)\big)^T \,. \tag{4.3.137}$$

This concludes our discussion on the matrix SVD of $\mathbf{C}_{\mathrm{u}}^x$. See the next section for explicit expressions of a proper set Φ containing sets of coefficients for the polynomial equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q \in \mathcal{Q}}$, or a matrix $\mathbf{\Phi}$ representing these coefficients, in terms of the left null space of $\mathbf{C}_{\mathrm{u}}^x$ found by the SVD.

4.3.8.3 Coefficients of polynomials in terms of SVD results

Finally, in this section we show how the coefficients of the equations in the system of homogeneous polynomial equations derived earlier can be expressed in terms of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ of $\mathbf{r}_{\mathrm{u}}^{x}[k]$ and $\mathbf{C}_{\mathrm{u}}^{x}$. From (4.3.95), (4.3.96), (4.3.125) and (4.3.137) it follows directly how a proper set of sets of coefficients Φ, or equivalently rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}}$ of the matrix $\boldsymbol{\Phi}$, can be expressed explicitly in terms of this left null space because these equations state that $\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ is spanned by the transposed versions of the left singular vectors $\mathbf{u}^{d_2^{x,2}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}$ that are contained in the set $\mathcal{U}^{\nu}$ and the matrix $\mathbf{U}^{\nu}$. Hence, proper nontrivial vectors $\tilde{\boldsymbol{\varphi}}_q$ satisfying (4.3.95) are given by:

$$\tilde{\boldsymbol{\varphi}}_q = \big(\mathbf{u}^{d_2^{x,2}+q}\big)^T \qquad \forall\, 1 \leq q \leq Q\,, \tag{4.3.138}$$

or by arbitrary linear combinations of these vectors:

$$\tilde{\boldsymbol{\varphi}}_q = \sum_{j=d_2^{x,2}+1}^{M_{\mathrm{u},2}^{x,2}} \alpha_q^j\,(\mathbf{u}^j)^T \qquad \forall\, \alpha_q^j \in \mathbb{C}\,; \tag{4.3.139}$$

see also the remarks on page 147. Equivalently, a nontrivial matrix $\boldsymbol{\Phi}$ satisfying (4.3.96) is given by:

$$\boldsymbol{\Phi} = (\mathbf{U}^{\nu})^T = \begin{bmatrix} \tilde{\boldsymbol{\varphi}}_1 \\ \vdots \\ \tilde{\boldsymbol{\varphi}}_Q \end{bmatrix} = \begin{bmatrix} \big(\mathbf{u}^{d_2^{x,2}+1}\big)^T \\ \vdots \\ \big(\mathbf{u}^{M_{\mathrm{u},2}^{x,2}}\big)^T \end{bmatrix}\,, \tag{4.3.140}$$

or any matrix containing arbitrary linear combinations of the rows of $\boldsymbol{\Phi}$. As we have remarked on page 157, in order to obtain a system of equations with as many linearly independent equations as possible, the row vectors in the set $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}}$ or the matrix $\boldsymbol{\Phi}$ should be chosen such that they span $\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big)$ completely; see (4.3.96). Examples of the computation of $\boldsymbol{\Phi}$ from the sensor correlation values will be given in Section 4.4. By realizing that we can set up a one-to-one mapping between the set $\big\{\varphi_q^{i_1 i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2}$ of coefficients corresponding to the q-th equation in the system and the row $\tilde{\boldsymbol{\varphi}}_q$, we can write the equivalent of (4.3.140) in terms of the set Φ defined in (4.3.26) by means of two equivalent formulations. Firstly, from (4.3.138) it follows that for $1 \leq p \leq 3$:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{(i_1,i_2)_p} = \big[\big(\mathbf{u}^{d_2^{x,2}+q}\big)^T\big]^p = \Big[u_1^{d_2^{x,2}+q} \;\; u_2^{d_2^{x,2}+q} \;\; u_3^{d_2^{x,2}+q}\Big]^p = u_p^{d_2^{x,2}+q}\,,$$

which yields:

$$\begin{aligned} \Phi &\triangleq \Big\{\big\{\varphi_q^{i_1 i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2}\Big\}_{q\in\mathcal{Q}} = \Big\{\big\{u_p^{d_2^{x,2}+q}\big\}_{1\leq p\leq 3}\Big\}_{q\in\mathcal{Q}} \\ &= \Big\{\big\{u_1^{d_2^{x,2}+1}, u_2^{d_2^{x,2}+1}, u_3^{d_2^{x,2}+1}\big\}, \ldots, \big\{u_1^{d_2^{x,2}+Q}, u_2^{d_2^{x,2}+Q}, u_3^{d_2^{x,2}+Q}\big\}\Big\}\,. \end{aligned} \tag{4.3.141}$$

Secondly, using the mapping defined in (4.3.110) from (4.3.138) it also follows that for $1 \leq p \leq 3$:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{(i_1,i_2)_p} = \big[\big(\mathbf{u}^{d_2^{x,2}+q}\big)^T\big]^p = \Big[u_{(i_1,i_2)_1}^{d_2^{x,2}+q} \;\; u_{(i_1,i_2)_2}^{d_2^{x,2}+q} \;\; u_{(i_1,i_2)_3}^{d_2^{x,2}+q}\Big]^p = u_{(i_1,i_2)_p}^{d_2^{x,2}+q}\,,$$

which yields:

$$\begin{aligned}\Phi &\triangleq \left\{ \left\{ \varphi_q^{i_1 i_2} \right\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \right\}_{q\in\mathcal{Q}} = \left\{ \left\{ u_{(i_1,i_2)_p}^{d_2^{x,2}+q} \right\}_{1\le p\le 3} \right\}_{q\in\mathcal{Q}} \\ &= \left\{ \left\{ u_{(i_1,i_2)_1}^{d_2^{x,2}+1}, u_{(i_1,i_2)_2}^{d_2^{x,2}+1}, u_{(i_1,i_2)_3}^{d_2^{x,2}+1} \right\}, \ldots, \left\{ u_{(i_1,i_2)_1}^{d_2^{x,2}+Q}, u_{(i_1,i_2)_2}^{d_2^{x,2}+Q}, u_{(i_1,i_2)_3}^{d_2^{x,2}+Q} \right\} \right\}. \end{aligned} \tag{4.3.142}$$

This concludes our derivation of the coefficients of the polynomials in the system.

Note that the Singular Value Decomposition confirms once more that the number of linearly independent equations in our system is given by (4.3.24), (4.3.54), and (4.3.92). This follows directly by considering the dimension of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big)$ of $\mathbf{C}_{\mathrm{u}}^x$. See also (4.3.139) for example. Since the singular vectors $\mathbf{u}^{d_2^{x,2}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},2}^{x,2}}$ correspond to the zero singular values, in subspace terminology we call them noise subspace vectors or 'zero subspace vectors'. Also note from the results above the notational consistency that the transpose operation effectively turns a superscript index into a subscript index, and vice versa. Finally, note that in practice only an approximate estimated version $\hat{\mathbf{C}}_{\mathrm{u}}^x$ of $\mathbf{C}_{\mathrm{u}}^x$ is available. In this case, the singular values are divided into two sets, one with the $d_2^{x,2}$ largest ones, and the other with the remaining $M_{\mathrm{u},2}^{x,2} - d_2^{x,2}$ smallest ones. This implies that in the statements made above $\sigma_1 \ge \cdots \ge \sigma_{d_2^{x,2}} > 0,\ \sigma_{d_2^{x,2}+1}, \ldots, \sigma_P = 0$ has to be replaced with $\sigma_1 \ge \cdots \ge \sigma_{d_2^{x,2}} > 0,\ \sigma_{d_2^{x,2}+1}, \ldots, \sigma_P \approx 0$. This stage completes the derivation of the system of equations. The algorithmic part of the method is summarized in Alg. 4.3; see also Alg. 4.2. The last step of the algorithm will be discussed in later sections.

Algorithm 4.3 High-level algorithm for subspace based real-valued 2×2 MIBI.

1: Compute/estimate sensor correlation functions $r_{11}^x[k], r_{12}^x[k]$ and $r_{22}^x[k]$ for lags k in Noise-Free ROS $\Omega_k^{s\setminus\nu}$;

2: Arrange these values in uniquiefied subspace matrix $\mathbf{C}_{\mathrm{u}}^x$;

3: Compute Singular Value Decomposition (SVD) of $\mathbf{C}_{\mathrm{u}}^x$ and split the result into signal and null/noise subspace parts as follows:

$$\mathbf{C}_{\mathrm{u}}^x = \mathbf{U}^x \mathbf{\Lambda}^x \mathbf{V}^x = \mathbf{U}^s \mathbf{\Lambda}^s \mathbf{V}^s + \mathbf{U}^\nu \mathbf{\Lambda}^\nu \mathbf{V}^\nu\,;$$

4: Compute matrix $\mathbf{\Phi}$ whose rows span the left null space of $\mathbf{C}_{\mathrm{u}}^x$:

$$\mathbf{\Phi} = \begin{bmatrix} \varphi_1^{11} & \varphi_1^{12} & \varphi_1^{22} \\ \vdots & \vdots & \vdots \\ \varphi_Q^{11} & \varphi_Q^{12} & \varphi_Q^{22} \end{bmatrix} = (\mathbf{U}^\nu)^T\,;$$

5: With each row $\tilde{\boldsymbol{\varphi}}_q$ of $\mathbf{\Phi}$ with $q \in \mathcal{Q}$ associate a bivariate homogeneous polynomial function of degree two:

$$f_q(z_1, z_2) \triangleq \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}\,;$$

6: The following system remains to be solved for the columns of the mixing matrix:

$$\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}} \qquad \equiv \qquad \mathbf{f}(\mathbf{z}) = \mathbf{\Phi}\, \check{\mathbf{z}}_{\mathrm{u}}^2 = \mathbf{0}_Q\,.$$

4.4 Algebraic and geometric structure

The purpose of this section is to provide insight into the algebraic and geometric structure of the problem induced by the system $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ in (4.3.33). As we have shown in the previous sections, the type of function we are considering is defined in (4.3.31), which is repeated here for convenience:

$$f_q(z_1, z_2) \triangleq \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}\,. \tag{4.4.1}$$

Due to the real-valued nature of the currently considered MIBI problem, the domain of main interest is the real-valued $z_1 - z_2$ plane. Because of the form specified in (4.4.1), the graph of each function $f_q(z_1, z_2)$ above the (real-valued) $z_1 - z_2$ plane is a so-called *quadric surface*. By definition, the set of values of the pair of variables (z_1, z_2) for which $f_q(z_1, z_2) = 0$ is the zero contour level of $f_q(z_1, z_2)$. Hence:

> Geometrically, solving the system $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ is equivalent to finding the intersections between the zero contour levels of the functions $f_1(z_1, z_2), \ldots, f_Q(z_1, z_2)$.

From (4.3.35), i.e. the homogeneity property, it is clear that the zero contour level of each function $f_q(z_1, z_2)$ in the system defines a *cone* in the 2-dimensional Euclidian space $\mathbb{C}_2$, where a cone is defined as follows [108]:

Definition 4.4.1. Cone. A set $\mathcal{C}$ in a linear vector space is said to be a *cone with vertex at the origin* if $\mathbf{z} \in \mathcal{C}$ implies that $\eta\mathbf{z} \in \mathcal{C}$ for all $\eta \in \mathbb{C}$.

For the current scenario we are only interested in the intersection of such a cone with the two-dimensional real plane. From the definition it follows that in the 2-dimensional Euclidian space $\mathbb{R}_2$, a cone is a set of lines through the origin. From the boxed statement above it is now clear that geometrically, solving the system $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ for the current scenario is equivalent to finding the intersections between the Q cones in the 2-dimensional Euclidian space $\mathbb{R}_2$ that correspond with the zero contour levels of the functions $f_1(z_1, z_2), \ldots$, $f_Q(z_1, z_2)$. The fact that the zero contour lines of the functions in $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ are lines through the origin implies certain restrictions on the possible values of the coefficients of each equation in the system. As we will see in the sequel, for the current scenario only combinations of φ_q^{11}, φ_q^{12}, and φ_q^{22} in each function $f_q(z_1, z_2)$ are allowed in which not all coefficients have the same sign. The saddle-shaped quadric surfaces corresponding to such combinations are so-called *hyperbolic paraboloids*; see Fig. 4.5 on page 177 for example. The trace of such a hyperbolic paraboloid surface in the $z_1 - z_2$ plane is a pair of lines intersecting at the origin, whereas the traces in planes parallel to the $z_1 - z_2$ plane are hyperbolas. For example, see Figures 4.5, 4.6, 4.8 and 4.9.

In the Sections 4.4.2 and 4.4.3 we will demonstrate the issues and properties discussed in the previous paragraph by means of two examples that are representative for the two possible essentially different cases for real-valued two-by-two MIBI, viz. the case with a nonsingular and the case singular mixing matrix. Before discussing these examples, in Section 4.4.1 we first examine the structure of a function $f_q(z_1, z_2)$ of the form specified in (4.4.1) in some detail because this provides additional insight that is also elementary for understanding the examples. In particular the factorization of a bivariate homogeneous polynomial of degree two into two linear terms is considered. Although this factorization immediately yields the solution for solving an equation of the form $f_q(z_1, z_2) = 0$, where $f_q(z_1, z_2)$ has the specified properties, our main purpose here is not to solve the system but to provide insight into the

structure of the considered equations. Possible ways for solving the system will be examined in Section 4.5, where we also briefly show how the source signals can be recovered after the columns of the mixing matrix have been estimated.

For the examples presented in Sections 4.4.2 and 4.4.3 we will assume that temporally white noise is present and that the two source signals are generated according to the model in (4.2.6) with AR(1) regression coefficients $\rho_1 = 0.8$ and $\rho_2 = 0.6$. The source auto-correlation functions are given by (4.2.7) and depicted in Fig. 4.3. The considered Noise-Free Region Of Support is chosen as $\Omega_k^{s\setminus\nu} = \{1, 2\}$, which is just large enough to ensure that the two source auto-correlation functions $r_{11}^s[k]$ and $r_{22}^s[k]$ are linearly independent over $\Omega_k^{s\setminus\nu}$. This can easily be seen by considering for example the ideal source auto-correlation matrix, which is given by:

$$\mathbf{C}^s \triangleq \begin{bmatrix} \tilde{\mathbf{r}}_{11}^s \\ \tilde{\mathbf{r}}_{22}^s \end{bmatrix} = \begin{bmatrix} r_{11}^s[1] & r_{11}^s[2] \\ r_{22}^s[1] & r_{22}^s[2] \end{bmatrix} = \begin{bmatrix} (\rho_1)^1 & (\rho_1)^2 \\ (\rho_2)^1 & (\rho_2)^2 \end{bmatrix} = \begin{bmatrix} 0.8 & 0.64 \\ 0.6 & 0.36 \end{bmatrix}.$$

This matrix clearly has full rank; hence $r_{11}^s[k]$ and $r_{22}^s[k]$ are linearly independent over $\Omega_k^{s\setminus\nu}$. It can easily be shown that also other valid choices for $\Omega_k^{s\setminus\nu}$ are possible. For each example we will derive the system of equations using both ideal sensor correlation functions, as well as sensor correlation functions estimated from simulated data. Contour plots associated with the equations in the system will be plotted. For reference and clarity, in these plots the columns of the ideal mixing matrix are indicated by black arrows; see Figures 4.8 and 4.13 for examples.

4.4.1 Factorization of bivariate homogeneous polynomial of degree two into two linear terms

In order to provide additional insight into the structure of functions of the type in (4.4.1), we show how such a function can be factorized into two linear terms, thereby also yielding the solutions. Consider the following general bivariate homogeneous polynomial equation of degree two:

$$f(z_1, z_2) = \varphi^{11} z_1 z_1 + \varphi^{12} z_1 z_2 + \varphi^{22} z_2 z_2 \,. \tag{4.4.2}$$

We want to demonstrate that this function can be factorized explicitly using simple algebra, thereby revealing its structure. In fact, it is the 2-homogeneity of $f(z_1, z_2)$ which implies that it can be written as the product of two linear terms:

$$f(z_1, z_2) = \varphi^{11} z_1 z_1 + \varphi^{12} z_1 z_2 + \varphi^{22} z_2 z_2 = \varphi^{11}(z_1 - \alpha_1 z_2)(z_1 - \alpha_2 z_2)\,, \tag{4.4.3}$$

where for the time being it is assumed that $\varphi^{11} \neq 0$. Given the scalars α_1 and α_2 in this factorization, a valid set of solutions of $f(z_1, z_2) = 0$ is given by:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} \alpha_1 \\ 1 \end{bmatrix}, \eta_2 \begin{bmatrix} \alpha_2 \\ 1 \end{bmatrix} \right\} \quad \forall\, \eta_1, \eta_2 \in \mathbb{C}\,, \tag{4.4.4}$$

which can easily be verified by substitution. Hence, in order to find the zero contour level of $f(z_1, z_2)$ we need to compute a set of valid values for α_1 and α_2. Setting the coefficients of the monomials in (4.4.3) equal to each other for both sides of the equation yields the following system of two equations in α_1 and α_2:

$$\begin{cases} \varphi^{12} &= -\varphi^{11}(\alpha_1 + \alpha_2)\,, \\ \varphi^{22} &= \varphi^{11}\alpha_1\alpha_2\,, \end{cases} \tag{4.4.5}$$

which can be solved for either α_1 or α_2. Because the equations in this system are symmetric in both variables it is sufficient to solve the following equation in the single variable α representing both α_1 and α_2:

$$\varphi^{11}(\alpha)^2 + \varphi^{12}\alpha + \varphi^{22} = 0\,, \qquad \alpha = \alpha_1,\, \alpha_2\,,$$

which yields:

$$\alpha = \frac{1}{2\varphi^{11}}\Big(-\varphi^{12} \pm \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}}\Big)\,.$$

Using (4.4.5) it can easily be verified that choosing:

$$\alpha_1 = \frac{1}{2\varphi^{11}}\Big(-\varphi^{12} - \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}}\Big), \quad \alpha_2 = \frac{1}{2\varphi^{11}}\Big(-\varphi^{12} + \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}}\Big)$$

yields the correct solution. Now, from (4.4.4) it follows that for arbitrary $\eta_1, \eta_2 \in \mathbb{C}$ the solutions of $f(z_1, z_2) = 0$ are given by:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} -\varphi^{12} - \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}} \\ 2\varphi^{11} \end{bmatrix}, \eta_2 \begin{bmatrix} -\varphi^{12} + \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}} \\ 2\varphi^{11} \end{bmatrix} \right\}. \tag{4.4.6}$$

Note that if $\varphi^{11} = 0$, then (4.4.2) is trivially factorized as follows:

$$f(z_1, z_2) = \varphi^{12} z_1 z_2 + \varphi^{22} z_2 z_2 = \left(\varphi^{12} z_1 + \varphi^{22} z_2\right) z_2\,,$$

and its roots are given by:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \eta_2 \begin{bmatrix} \varphi^{22} \\ -\varphi^{12} \end{bmatrix} \right\} \quad \forall\, \eta_1, \eta_2 \in \mathbb{C}\,. \tag{4.4.7}$$

Likewise, if $\varphi^{22} = 0$ (4.4.2) is trivially factorized as follows:

$$f(z_1, z_2) = \varphi^{11} z_1 z_1 + \varphi^{12} z_1 z_2 = z_1(\varphi^{11} z_1 + \varphi^{12} z_2)\,,$$

and its roots are given by:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \eta_2 \begin{bmatrix} \varphi^{12} \\ -\varphi^{11} \end{bmatrix} \right\} \quad \forall\, \eta_1, \eta_2 \in \mathbb{C}\,.$$

Note that this agrees with (4.4.6). Finally, if $\varphi^{12} = 0$ (4.4.3) reduces to:

$$f(z_1, z_2) = \varphi^{11} z_1 z_1 + \varphi^{22} z_2 z_2 = \varphi^{11}(z_1)^2 + \varphi^{22}(z_2)^2\,.$$

Here we see the reason why not all coefficients can have the same sign. If this would be the case, $z_1 = z_2 = 0$ would be the only solution over the real $z_1 - z_2$ plane, which contradicts the fact the our solutions consist of lines through the origin. The roots of the last equation above are easily obtained as:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} \sqrt{|\varphi^{22}|} \\ \sqrt{|\varphi^{11}|} \end{bmatrix}, \eta_2 \begin{bmatrix} -\sqrt{|\varphi^{22}|} \\ \sqrt{|\varphi^{11}|} \end{bmatrix} \right\} \quad \forall\, \eta_1, \eta_2 \in \mathbb{C}\,.$$

Note that this also agrees with (4.4.6). Hence, the roots of $f(z_1, z_2)$ for all combinations of coefficients φ^{11}, φ^{12} and φ^{22} are given by (4.4.6), except for $\varphi^{11} = 0$, in which case they are given by (4.4.7).

Intuition tells us that there exists another elegant factorization that covers all cases at once. We briefly formulate this factorization because it is elegant and provides additional insight. We write $f(z_1, z_2)$ as the product of two linear terms multiplied by a constant:

$$f(z_1, z_2) = \gamma(\alpha z_1 - z_2)(z_1 - \beta z_2) = \gamma(\alpha z_1 z_1 - (1 + \alpha\beta) z_1 z_2 + \beta z_2 z_2)\,. \tag{4.4.8}$$

It follows immediately that γ must be non-zero because otherwise $f(z_1, z_2)$ would be the zero function. Following the same reasoning as above yields a valid set of solutions of $f(z_1, z_2) = 0$ for arbitrary $\eta_1, \eta_2 \in \mathbb{C}$:

$$\mathcal{F} = \left\{ \eta_1 \begin{bmatrix} -\varphi^{12} - \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}} \\ 2\varphi^{11} \end{bmatrix}, \eta_2 \begin{bmatrix} 2\varphi^{22} \\ -\varphi^{12} - \sqrt{\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22}} \end{bmatrix} \right\}, \tag{4.4.9}$$

which holds for all values of the coefficients φ^{11}, φ^{12} and φ^{22}. Both from this equation as well as from equation (4.4.6) we see that in order to have real-valued solutions the condition $\left(\varphi^{12}\right)^2 - 4\varphi^{11}\varphi^{22} \leq 0$ should be satisfied.

Now we are in a position to discuss several examples that will elucidate the theory that has been presented so far in this chapter. The examples focus on highlighting algebraic and geometric structure. The actual estimation of the mixing matrix columns and, if applicable, the separated signals will be presented in Section 4.5.

4.4.2 Example 1: Full rank mixing matrix and stationary AR(1) signals

In this section we perform the various steps of Alg. 4.3 on page 171 for the framework signal scenario discussed in Section 4.2, in particular see Equations (4.2.5)-(4.2.16), and with a full rank mixing matrix. In Section 4.4.3, we do the same for a rank-deficient mixing matrix. In the next section we start by analyzing the scenario for an ideally known subspace matrix and in Section 4.4.2.2 we give a similar analysis for a subspace matrix estimated from simulated sensor data.

4.4.2.1 Employing ideal subspace matrix

Let a full rank mixing matrix be given by:

$$\mathbf{A} = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.6 \end{bmatrix}. \tag{4.4.10}$$

The uniquiefied Khatri-Rao product of $\mathbf{A}$ is given by (4.3.75):

$$\mathbf{A}^2_{\mathrm{u},\diamond} = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 \\ a_1^1 a_2^1 & a_1^2 a_2^2 \\ a_2^1 a_2^1 & a_2^2 a_2^2 \end{bmatrix} = \begin{bmatrix} 0.9 \cdot 0.9 & 0.1 \cdot 0.1 \\ 0.9 \cdot 0.2 & 0.1 \cdot 0.6 \\ 0.2 \cdot 0.2 & 0.6 \cdot 0.6 \end{bmatrix} = \begin{bmatrix} 0.81 & 0.01 \\ 0.18 & 0.06 \\ 0.04 & 0.36 \end{bmatrix}.$$

It can easily be verified that $\mathrm{rank}\left(\mathbf{A}^2_{\mathrm{u},\diamond}\right) = \mathrm{rank}(\mathbf{A}) = 2$; see also the remark just above (4.3.87). Hence, $d_2^{x,2} = 2$ according to (4.3.89). Using this result in (4.3.24), (4.3.54), or (4.3.92) implies that the number of linearly independent equations in the system $\{f_q(z_1, z_2) = 0\}_{q \in \mathcal{Q}}$ equals $Q = M_{\mathrm{a},2}^{x,2} - d_2^{x,2} = 3 - 2 = 1$, i.e. $\mathcal{Q} = \{1\}$. Consequently, there is only one equation in the system that is given by $f(z_1, z_2) = 0$ with $f(z_1, z_2)$ defined in (4.3.31) (we have omitted the subscript which is unnecessary now). As has been described in Section 4.3.8 and summarized in Alg. 4.3 on page 171, the coefficients $\varphi^{11}, \varphi^{12}$ and φ^{22} can be computed from the Singular Value Decomposition (SVD) of the subspace matrix $\mathbf{C}_{\mathrm{u}}^x$ that contains the sensor correlations for the lags specified in $\Omega_k^{s\backslash\nu} = \{1, 2\}$ (recall that the sensor noise is assumed to be temporally white) stacked according to (4.3.60). The

ideal value of $\mathbf{C}_{\mathrm{u}}^{x}$ can be computed by means of (4.3.77):

$$\begin{aligned}\mathbf{C}_{\mathrm{u}}^{x} &= \begin{bmatrix}\tilde{\mathbf{r}}_{11}^{x}\\ \tilde{\mathbf{r}}_{12}^{x}\\ \tilde{\mathbf{r}}_{22}^{x}\end{bmatrix} = \begin{bmatrix}r_{11}^{x}[1] & r_{11}^{x}[2]\\ r_{12}^{x}[1] & r_{12}^{x}[2]\\ r_{22}^{x}[1] & r_{22}^{x}[2]\end{bmatrix} = \begin{bmatrix}a_1^1 a_1^1 & a_1^2 a_1^2\\ a_1^1 a_2^1 & a_1^2 a_2^2\\ a_2^1 a_2^1 & a_2^2 a_2^2\end{bmatrix} \begin{bmatrix}r_{11}^{s}[1] & r_{11}^{s}[2]\\ r_{22}^{s}[1] & r_{22}^{s}[2]\end{bmatrix}\\ &= \begin{bmatrix}0.81 & 0.01\\ 0.18 & 0.06\\ 0.04 & 0.36\end{bmatrix} \begin{bmatrix}0.8 & 0.64\\ 0.6 & 0.36\end{bmatrix} = \begin{bmatrix}0.6540 & 0.5220\\ 0.1800 & 0.1368\\ 0.2480 & 0.1552\end{bmatrix}. \end{aligned} \tag{4.4.11}$$

From Theorem 4.3.5 on page 167 it follows that we can obtain the following Singular Value Decomposition of the subspace matrix:

$$\mathbf{C}_{\mathrm{u}}^{x} = \underbrace{\begin{bmatrix}-0.9152 & 0.3459 & 0.2070\\ -0.2473 & -0.0762 & -0.9659\\ -0.3183 & -0.9352 & 0.1552\end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix}0.9143 & 0\\ 0 & 0.0317\\ 0 & 0\end{bmatrix}}_{\mathbf{\Sigma}} \underbrace{\begin{bmatrix}-0.7897 & -0.6135\\ -0.6135 & 0.7897\end{bmatrix}}_{\mathbf{V}}.$$

Equation (4.3.138) shows that a valid vector $\tilde{\boldsymbol{\varphi}}$ containing the coefficients of the single equation $f(z_1, z_2) = 0$ in the system is given by:

$$\tilde{\boldsymbol{\varphi}} = \left(\mathbf{u}^{d_2^{x,2}+1}\right)^T = (\mathbf{u}^3)^T = \begin{bmatrix}0.2070 & -0.9659 & 0.1552\end{bmatrix}. \tag{4.4.12}$$

Hence the function $f(z_1, z_2)$ is given by:

$$f(z_1, z_2) = 0.2070\, z_1 z_1 - 0.9659\, z_1 z_2 + 0.1552\, z_2 z_2\,. \tag{4.4.13}$$

The graph and contour plot of $f(z_1, z_2)$ are shown in Fig. 4.5 on the next page. As we have explained at the beginning of Section 4.4, the graph is a saddle-shaped quadric surface called *hyperbolic paraboloid*. For clarity, Fig. 4.6 displays the contour plot of $f(z_1, z_2)$ only. As can be seen in both figures, the trace in the $z_1 - z_2$ plane is a pair of lines intersecting at the origin, and the traces in planes parallel to the $z_1 - z_2$ plane are hyperbolas. The black arrows in Fig. 4.6 represent the columns of the mixing matrix. The two lines through the origin of each figure, i.e. the trace in the $z_1 - z_2$ plane, define the zero contour level of $f(z_1, z_2)$. According to Definition 4.4.1, they form a cone in the two-dimensional Euclidian space. The contour levels for function values different from zero clearly do not form straight lines through the origin. Because there is only one equation in the current example, no intersections between cones need to be computed. It is clear from Fig. 4.6 that the arrows representing the columns of the mixing matrix are exactly in the directions of the two lines describing the zero contour level of $f(z_1, z_2)$, thereby revealing that both columns $\mathbf{a}^1$ and $\mathbf{a}^2$ are determined correctly and uniquely by the zero contour level. In Section 4.5, we will show how the array response vectors can be extracted from $f(z_1, z_2)$, i.e. how the equation $f(z_1, z_2) = 0$ can be solved.

Finally, for the sake of providing insight we show how function (4.4.13) is related to the (possibly scaled) columns of $\mathbf{A}$. We want to construct a bivariate homogeneous polynomial function $f_{\mathbf{A}}(z_1, z_2)$ of degree two in such a way that $f_{\mathbf{A}}(z_1, z_2) = 0$ for each possibly scaled column $\mathbf{z} \in \{\eta_1 \mathbf{a}^1, \eta_2 \mathbf{a}^2\}$ with arbitrary $\eta_1, \eta_2 \in \mathbb{C}$. Now, if $\mathbf{z} = \eta_j \mathbf{a}^j$ for some $\eta_j \in \mathbb{C}$ and at least one $j \in \{1, 2\}$, i.e. if $\mathbf{z}$ equals a (scaled) column of $\mathbf{A}$, then the following condition on η holds:

$$\eta_j = \frac{z_1}{a_1^j} = \frac{z_2}{a_2^j}\,, \tag{4.4.14}$$

which implies that:

$$a_2^j z_1 - a_1^j z_2 = 0\,. \tag{4.4.15}$$

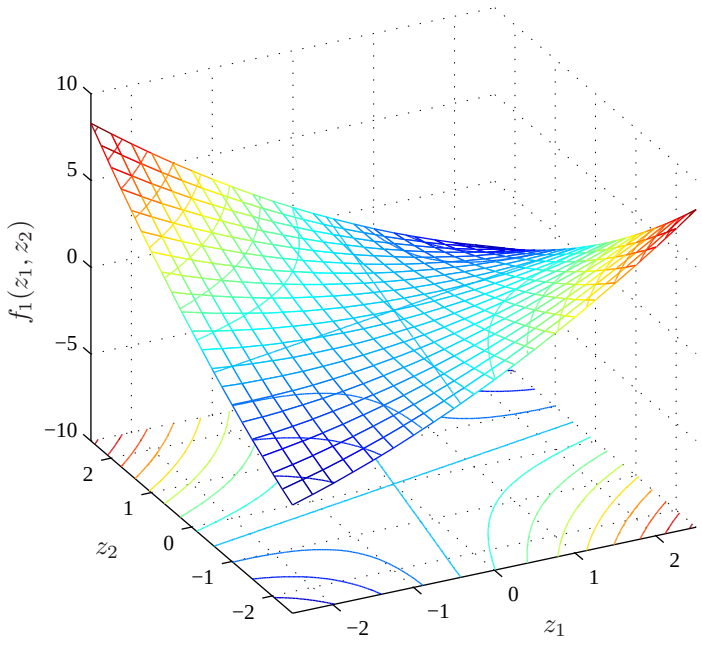

Figure 4.5: *Graph of $f(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.1.*

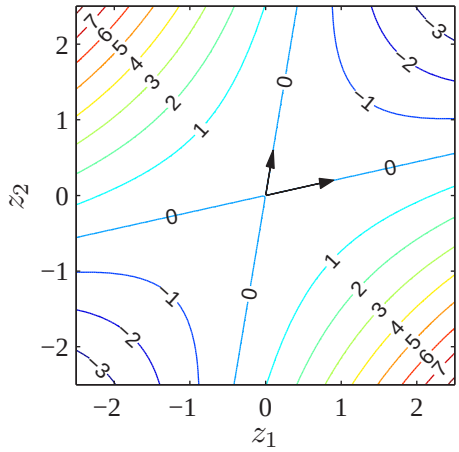

Figure 4.6: *Contour plot of $f(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.1.*

Now it is trivial to construct a function $f_{\mathbf{A}}(\mathbf{z})$ that is zero for both $\mathbf{z} = \eta_1 \mathbf{a}^1$ and $\mathbf{z} = \eta_2 \mathbf{a}^2$ with arbitrary $\eta_1, \eta_2 \in \mathbb{C}$ and possesses the properties mentioned above by taking the product of the linear terms $a_2^1 z_1 - a_1^1 z_2$ and $a_2^2 z_1 - a_1^2 z_2$ as follows:

$$f_{\mathbf{A}}(z_1, z_2) \triangleq (a_2^1 z_1 - a_1^1 z_2)(a_2^2 z_1 - a_1^2 z_2) \; . \tag{4.4.16}$$

This function is identically zero for all $\mathbf{z} \in \{\eta_1 \mathbf{a}^1, \eta_2 \mathbf{a}^2\}$ with arbitrary $\eta_1, \eta_2 \in \mathbb{C}$ because for such $\mathbf{z}$ at least one of the linear terms is identically zero. Its form is inspired by the discussion in Section 4.4.1 on the factorization of a bivariate homogeneous polynomial of degree two into two linear terms. Expanding (4.4.16) yields:

$$f_{\mathbf{A}}(z_1, z_2) \triangleq a_2^1 a_2^2 z_1 z_1 - (a_1^1 a_2^2 + a_2^1 a_1^2) z_1 z_2 + a_1^1 a_1^2 z_2 z_2 \triangleq \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \check{\mathbf{z}}_{\mathrm{u}}^2 = \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \mathbf{w}_{\mathrm{u}}^2(\mathbf{z}), \tag{4.4.17}$$

where:

$$\tilde{\boldsymbol{\varphi}}_{\mathbf{A}} \triangleq \begin{bmatrix} a_2^1 a_2^2 & -(a_1^1 a_2^2 + a_2^1 a_1^2) & a_1^1 a_1^2 \end{bmatrix} .$$

Comparing function (4.4.17) to (4.3.31) implies that the row vectors $\tilde{\boldsymbol{\varphi}}_1 = \begin{bmatrix} \varphi^{11} & \varphi^{12} & \varphi^{22} \end{bmatrix}$ and $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}} = \begin{bmatrix} a_2^1 a_2^2 & -(a_1^1 a_2^2 + a_2^1 a_1^2) & a_1^1 a_1^2 \end{bmatrix}$ should span the same space $\mathcal{N}_1\big(\mathbf{C}_{\mathrm{u}}^x\big) = \mathcal{N}_1\big(\mathbf{r}_{\mathrm{u}}^x[k]\big)$. This can easily be verified by substituting the coefficients in (4.4.10) into $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}$ and normalizing it to unit norm. The resulting coefficients turn out to be equal to those in (4.4.13). As another insightful method, multiplying $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}$ by $\mathbf{A}_{\mathrm{u},\diamond}^2$ it is immediately seen that (4.3.101) is satisfied:

$$\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \mathbf{A}_{\mathrm{u},\diamond}^2 = \begin{bmatrix} a_2^1 a_2^2 & -(a_1^1 a_2^2 + a_2^1 a_1^2) & a_1^1 a_1^2 \end{bmatrix} \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 \\ a_1^1 a_2^1 & a_1^2 a_2^2 \\ a_2^1 a_2^1 & a_2^2 a_2^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} .$$

In fact, we have constructed $f_{\mathbf{A}}(z_1, z_2)$ in such way that the latter condition is fulfilled. These results reconfirm that we have derived correct and useful results.

4.4.2.2 Employing subspace matrix estimated from simulated sensor data

Now we will compute and plot the same contour levels as above for a subspace matrix estimated from computer generated signals. We assume that the two AR(1) source signals $s_1[n]$ and $s_2[n]$ are unit variance Gaussian signals consisting of $N_s = 4000$ time samples. To this end the sequences $w_1[n]$ and $w_2[n]$ in model (4.2.6) are generated independently of each other as i.i.d. Gaussian random variables with variances chosen in such a way that $s_1[n]$ and $s_2[n]$ both have unit variance. The sensor signals $x_1[n]$ and $x_2[n]$ are computed according to the model in (4.1.2), where $\mathbf{A}$ is given by (4.4.10) and the noise signals $\nu_1[n]$ and $\nu_2[n]$ are mutually statistically independent white Gaussian noise sequences with standard deviation 0.5. The resulting standard deviations of the noise-free parts of the sensor signals $x_1[n]$ and $x_2[n]$ are given by 0.91 and 0.63 respectively. Hence, the SNR at the first and second sensors equal 5.2 and 2.0 dB respectively. For a specific realization, Fig. 4.7 on the facing page shows 200 samples of the source signals $s_1[n]$ and $s_2[n]$ at the left side, and the noisy mixtures $x_1[n]$ and $x_2[n]$ at the right side. From the sensor signals, we estimate the sensor correlation functions $r_{11}^x[k], r_{12}^x[k]$ and $r_{22}^x[k]$ for all lags $k \in \Omega_k^{s\backslash\nu} = \{1, 2\}$ by averaging products of the form $x_{i_1}[n]\, x_{i_2}[n-k]$ over the available time samples:

$$\boxed{\hat{r}_{i_1 i_2}^x[k] = \frac{1}{N_s} \sum_{n=1}^{N_s} x_{i_1}[n]\, x_{i_2}[n-k] \,.} \tag{4.4.18}$$

Using this formula for estimating the subspace matrix from the sensor signals yields:

$$\hat{\mathbf{C}}_{\mathrm{u}}^x = \begin{bmatrix} \hat{r}_{11}^x[1] & \hat{r}_{11}^x[2] \\ \hat{r}_{12}^x[1] & \hat{r}_{12}^x[2] \\ \hat{r}_{22}^x[1] & \hat{r}_{22}^x[2] \end{bmatrix} = \begin{bmatrix} 0.6804 & 0.5342 \\ 0.1710 & 0.1308 \\ 0.2700 & 0.1640 \end{bmatrix} .$$

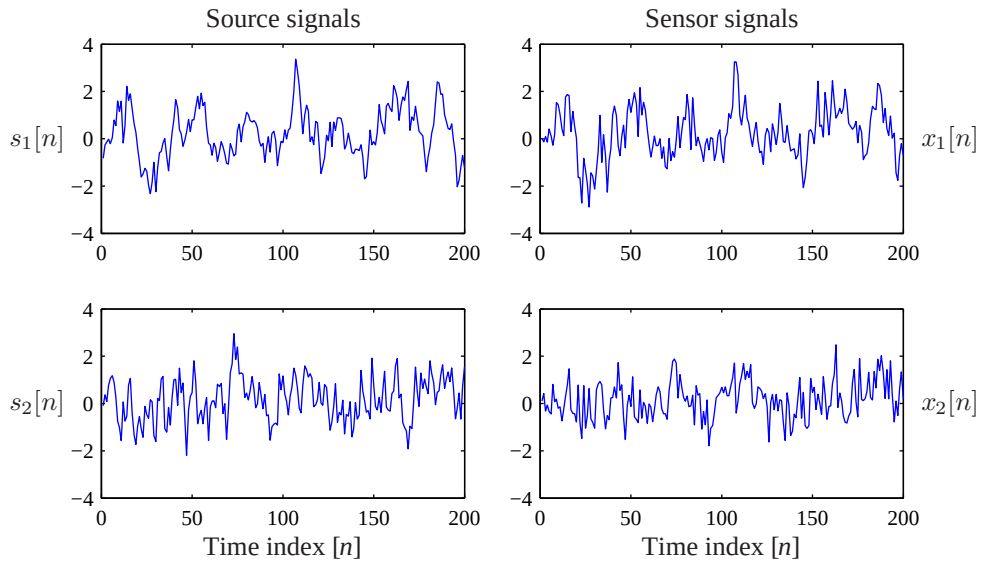

Figure 4.7: *Stationary Gaussian $AR(1)$ source signals (left) and their noisy mixtures (right) for example with full rank mixing matrix in Section 4.4.2.2.*

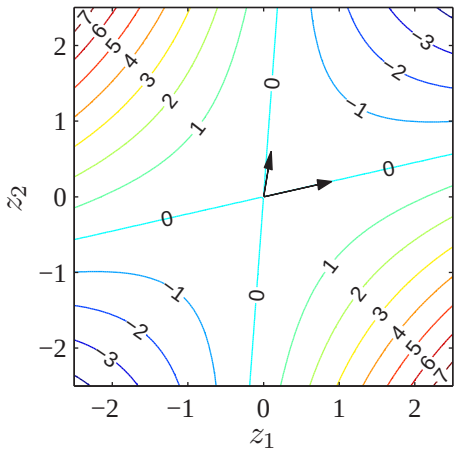

Figure 4.8: *Contour plot of $\hat{f}(z_1, z_2)$ for example with full rank mixing matrix in Section 4.4.2.2.*

Obviously, if the number N_s of data samples is increased $\hat{\mathbf{C}}_{\mathrm{u}}^x$ converges to $\mathbf{C}_{\mathrm{u}}^x$ given in (4.4.11). Similarly to the previous section, using Theorem 4.3.5 it follows from the SVD of $\hat{\mathbf{C}}_{\mathrm{u}}^x$ that the estimate $\hat{\tilde{\boldsymbol{\varphi}}}$ of $\tilde{\boldsymbol{\varphi}}$ is given by:

$$\hat{\tilde{\boldsymbol{\varphi}}} = \left(\mathbf{u}^{d_2^{x,2}+1}\right)^T = (\mathbf{u}^3)^T = \begin{bmatrix} 0.2168 & -0.9737 & 0.0702 \end{bmatrix} . \tag{4.4.19}$$

Hence, the estimate $\hat{f}(z_1, z_2)$ of the function $f(z_1, z_2)$ is given by:

$$\hat{f}(z_1, z_2) = 0.2168\, z_1 z_1 - 0.9737\, z_1 z_2 + 0.0702\, z_2 z_2\,. \tag{4.4.20}$$

The contour plot of $\hat{f}(z_1, z_2)$ is shown in Fig. 4.8 on the previous page, which should be compared to the one in Fig. 4.6 on page 177. As in Fig. 4.6, the black arrows represent the columns of the true mixing matrix. The two lines through the origin define the zero contour level of $\hat{f}(z_1, z_2)$. If the columns of the mixing matrix would be estimated by solving $\hat{f}(z_1, z_2) = 0$, the solution vectors would point in the direction of these lines. Since the black arrows approximately point in the same directions, the estimated columns $\hat{\mathbf{a}}^1$ and $\hat{\mathbf{a}}^2$ of the mixing matrix approximate the true ones rather well; see also Section 4.5. It is also evident from the figure that they are determined uniquely by the zero contour level of $\hat{f}(z_1, z_2)$. Note that the norms/scalings of the true columns cannot be estimated due to the permutation indeterminacy, and thus may be chosen freely.

4.4.3 Example 2: Rank-deficient mixing matrix and stationary AR(1) signals

In this section, similarly to Section 4.4.2 we perform the various steps of Alg. 4.3 on page 171 for the framework signal scenario discussed in Section 4.2, but now with a rank-deficient mixing matrix. Again, we start by analyzing the scenario for an ideally known subspace matrix, which is then followed in Section 4.4.3.2 by a similar analysis for a subspace matrix estimated from simulated sensor data.

4.4.3.1 Employing ideal subspace matrix

Let a rank-deficient mixing matrix be given by:

$$\mathbf{A} = \begin{bmatrix} 0.9 & 1.8 \\ 0.2 & 0.4 \end{bmatrix}\,. \tag{4.4.21}$$

Note that the rank deficiency follows directly from the fact that $\mathbf{a}^2 = 2\mathbf{a}^1$. The uniquiefied Khatri-Rao product of $\mathbf{A}$ is given by (4.3.75):

$$\mathbf{A}^2_{\mathrm{u},\diamond} = \begin{bmatrix} a^1_1 a^1_1 & a^2_1 a^2_1 \\ a^1_1 a^1_2 & a^2_1 a^2_2 \\ a^1_2 a^1_2 & a^2_2 a^2_2 \end{bmatrix} = \begin{bmatrix} 0.9 \cdot 0.9 & 1.8 \cdot 1.8 \\ 0.9 \cdot 0.2 & 1.8 \cdot 0.4 \\ 0.2 \cdot 0.2 & 0.4 \cdot 0.4 \end{bmatrix} = \begin{bmatrix} 0.81 & 3.24 \\ 0.18 & 0.72 \\ 0.04 & 0.16 \end{bmatrix}\,.$$

It can easily be verified that now $\mathrm{rank}\big(\mathbf{A}^2_{\mathrm{u},\diamond}\big) = \mathrm{rank}(\mathbf{A}) = 1$. Hence, $d_2^{x,2} = 1$ according to (4.3.89). It should be noted that in general if the elements of $\mathbf{A}$ were drawn independently from a continuous probability distribution, $\mathrm{rank}\big(\mathbf{A}^2_{\mathrm{u},\diamond}\big)$ would be two with probability one. Therefore, the current example is a bit pathological because it will almost never occur in practice, but it serves to demonstrate the rationale behind the method. Using (4.3.24), (4.3.54), or (4.3.92), it follows that the number of linearly independent equations in the system $\{f_q(z_1, z_2) = 0\}_{q \in \mathcal{Q}}$ equals $Q = M^{x,2}_{\mathrm{a},2} - d_2^{x,2} = 3 - 1 = 2$, i.e. $\mathcal{Q} = \{1, 2\}$. Consequently, there are two equations in the system that are given by $f_1(z_1, z_2) = 0$ and $f_2(z_1, z_2) = 0$. The functions $f_1(z_1, z_2)$ and $f_2(z_1, z_2)$ are defined in (4.3.31) as:

$$\begin{cases} f_1(z_1, z_2) \triangleq \varphi_1^{11} z_1 z_1 + \varphi_1^{12} z_1 z_2 + \varphi_1^{22} z_2 z_2\,; \\ f_2(z_1, z_2) \triangleq \varphi_2^{11} z_1 z_1 + \varphi_2^{12} z_1 z_2 + \varphi_2^{22} z_2 z_2\,. \end{cases}$$

Using (4.3.107), the system $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ is written in matrix-vector notation as follows:

$$\mathbf{f}(\mathbf{z}) \triangleq \mathbf{\Phi}\,\check{\mathbf{z}}_{\mathrm{u}}^2 = \mathbf{\Phi}\,\mathbf{w}_{\mathrm{u}}^2(\mathbf{z}) = \mathbf{0} \quad \text{or} \quad \begin{bmatrix} f_1(z_1,z_2) \\ f_2(z_1,z_2) \end{bmatrix} \triangleq \begin{bmatrix} \varphi_1^{11} & \varphi_1^{12} & \varphi_1^{22} \\ \varphi_2^{11} & \varphi_2^{12} & \varphi_2^{22} \end{bmatrix} \begin{bmatrix} z_1z_1 \\ z_1z_2 \\ z_2z_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Again, a valid set of values for the coefficients of the equations in the system, i.e. the coefficients of the matrix $\mathbf{\Phi}$, can be computed from the SVD of the subspace matrix $\mathbf{C}_{\mathrm{u}}^x$ that contains the sensor correlations for the lags specified in $\Omega_k^{s\backslash\nu} = \{1, 2\}$ stacked according to (4.3.60). The ideal value of $\mathbf{C}_{\mathrm{u}}^x$ can be computed by means of (4.3.77) as follows

$$\begin{aligned} \mathbf{C}_{\mathrm{u}}^x &= \begin{bmatrix} \tilde{\mathbf{r}}_{11}^x \\ \tilde{\mathbf{r}}_{12}^x \\ \tilde{\mathbf{r}}_{22}^x \end{bmatrix} = \begin{bmatrix} r_{11}^x[1] & r_{11}^x[2] \\ r_{12}^x[1] & r_{12}^x[2] \\ r_{22}^x[1] & r_{22}^x[2] \end{bmatrix} = \begin{bmatrix} a_1^1a_1^1 & a_1^2a_1^2 \\ a_1^1a_2^1 & a_1^2a_2^2 \\ a_2^1a_2^1 & a_2^2a_2^2 \end{bmatrix} \begin{bmatrix} r_{11}^s[1] & r_{11}^s[2] \\ r_{22}^s[1] & r_{22}^s[2] \end{bmatrix} \\ &= \begin{bmatrix} 0.81 & 3.24 \\ 0.18 & 0.72 \\ 0.04 & 0.16 \end{bmatrix} \begin{bmatrix} 0.8 & 0.64 \\ 0.6 & 0.36 \end{bmatrix} = \begin{bmatrix} 2.5920 & 1.6848 \\ 0.5760 & 0.3744 \\ 0.1280 & 0.0832 \end{bmatrix}. \end{aligned} \tag{4.4.22}$$

Once more, a Singular Value Decomposition of the subspace matrix follows from Theorem 4.3.5 on page 167 and is given by:

$$\mathbf{C}_{\mathrm{u}}^x = \underbrace{\begin{bmatrix} -0.9751 & -0.2167 & -0.0482 \\ -0.2167 & 0.9762 & -0.0053 \\ -0.0482 & -0.0053 & 0.9988 \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} 3.1705 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{\mathbf{\Sigma}} \underbrace{\begin{bmatrix} -0.8384 & -0.5450 \\ -0.5450 & 0.8384 \end{bmatrix}}_{\mathbf{V}}.$$

Note from (4.3.89) that $\mathrm{rank}(\mathbf{C}_{\mathrm{u}}^x) = \mathrm{rank}\big(\mathbf{A}_{\mathrm{u},\diamond}^2\big) = d_2^{x,2} = 1$. Equation (4.3.140) shows that a valid matrix $\mathbf{\Phi}$ containing the coefficients of the two equations in the system is given:

$$\mathbf{\Phi} = \begin{bmatrix} \tilde{\boldsymbol{\varphi}}_1 \\ \tilde{\boldsymbol{\varphi}}_2 \end{bmatrix} = \begin{bmatrix} \big(\mathbf{u}^2\big)^T \\ \big(\mathbf{u}^3\big)^T \end{bmatrix}^T = \begin{bmatrix} -0.2167 & 0.9762 & -0.0053 \\ -0.0482 & -0.0053 & 0.9988 \end{bmatrix}. \tag{4.4.23}$$

Hence, the functions $f_1(z_1, z_2)$ and $f_2(z_1, z_2)$ are given by:

$$\begin{cases} f_1(z_1,z_2) = -0.2167\,z_1z_1 + 0.9762\,z_1z_2 - 0.0053\,z_2z_2\,; \\ f_2(z_1,z_2) = -0.0482\,z_1z_1 - 0.0053\,z_1z_2 + 0.9988\,z_2z_2\,. \end{cases} \tag{4.4.24}$$

The corresponding contour plots are shown in Fig. 4.9 on the next page. As we have explained at the beginning of Section 4.4 on page 172, the graphs of the functions are saddle-shaped quadric surfaces called *hyperbolic paraboloids*. The two lines through the origin of the left figure define the zero contour level of $f_1(z_1, z_2)$, which again form a cone in the two-dimensional Euclidian space. Similarly, the two lines through the origin of the right figure define the zero contour level of $f_2(z_1, z_2)$ and form a cone too. In each figure, the black arrows representing the columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of the mixing matrix point in the direction of one of the two lines describing the zero contour level. Because $\mathbf{a}^2 = 2\mathbf{a}^1$, only one column of $\mathbf{A}$, say $\mathbf{a}^1$, has to be found since the scaling ambiguity cannot be resolved. This is also clear by rewriting the 2×2 Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI) problem as a 2×1 MIBI problem, i.e. with two sensor signals and only one source signal:

$$\begin{aligned} \mathbf{x}[n] &= \big(\mathbf{a}^1\,s_1[n] + \mathbf{a}^2\,s_2[n]\big) + \boldsymbol{\nu}[n] = \big(\mathbf{a}^1\,s_1[n] + 2\mathbf{a}^1\,s_2[n]\big) + \boldsymbol{\nu}[n] \\ &= \mathbf{a}^1\;(s_1[n] + 2s_2[n]) + \boldsymbol{\nu}[n]\,. \end{aligned}$$

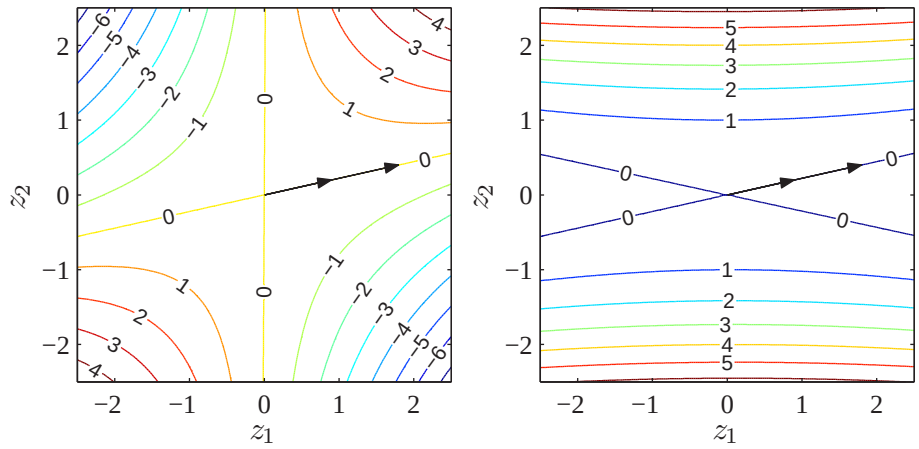

Figure 4.9: *Contour plots of $f_1(z_1, z_2)$ (left) and $f_2(z_1, z_2)$ (right) for example with rank-deficient mixing matrix in Section 4.4.3.1.*

Hence, the mixing 'matrix' in this model is given by $\mathbf{a}^1$, and the source signal by $s_1[n] + 2s_2[n]$. Since there are two lines in each of the subfigures of Fig. 4.9, the column $\mathbf{a}^1$ cannot be determined uniquely from one of the figures separately. However, it is determined uniquely by the intersection of the two sets of lines, i.e. the cones, which is clearly illustrated by the fact that both figures have exactly one line in common that 'points' in the direction of $\mathbf{a}^1$ and $\mathbf{a}^2$. Because there are two equations in the system, the intersection between the two corresponding cones has to be computed in order to solve the system. In Section 4.5, we will show how $\mathbf{a}^1$ can be found by solving the system $\{f_1(z_1, z_2) = 0, f_2(z_1, z_2) = 0\}$ consisting of two polynomial equations.

As in Section 4.4.2.1, for the sake of providing insight we derive a function related to the functions in (4.4.24) that can be computed from the columns of $\mathbf{A}$. In Section 4.4.2.1 we have seen that if $\mathbf{z}$ equals a possibly scaled column $\mathbf{a}^j$ of $\mathbf{A}$, then condition (4.4.15) holds. Again, it is trivial to construct a bivariate homogeneous polynomial function $f_{\mathbf{A}}(\mathbf{z})$ of degree two that is zero for both $\mathbf{z} = \eta_1\mathbf{a}^1$ and $\mathbf{z} = \eta_2\mathbf{a}^2 = 2\eta_2\mathbf{a}^1$ with arbitrary $\eta_1, \eta_2 \in \mathbb{C}$ by taking the product of the linear terms $a_2^1 z_1 - a_1^1 z_2$ and $a_2^2 z_1 - a_1^2 z_2$. Since $\mathbf{a}^2 = 2\mathbf{a}^1$ this is equivalent to choosing the following function:

$$f_{\mathbf{A}}(z_1, z_2) \triangleq (a_2^1 z_1 - a_1^1 z_2)^2 \,, \tag{4.4.25}$$

which is identically zero for all $\mathbf{z} = \eta\mathbf{a}^1$ with arbitrary $\eta \in \mathbb{C}$; see also Section 4.4.1 for more information on the factorization of a bivariate homogeneous polynomial of degree two into two linear terms. Expanding (4.4.25) yields:

$$\begin{aligned} f_{\mathbf{A}}(z_1, z_2) &\triangleq a_2^1 a_2^1 z_1 z_1 - 2a_1^1 a_2^1 z_1 z_2 + a_1^1 a_1^1 z_2 z_2 \\ &\triangleq \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \check{\mathbf{z}}_{\mathrm{u}}^2 = \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \mathbf{w}_{\mathrm{u}}^2(\mathbf{z}) \end{aligned} \tag{4.4.26}$$

with:

$$\tilde{\boldsymbol{\varphi}}_{\mathbf{A}} \triangleq \begin{bmatrix} a_2^1 a_2^1 & -2a_1^1 a_2^1 & a_1^1 a_1^1 \end{bmatrix} = \begin{bmatrix} 0.04 & -0.36 & 0.81 \end{bmatrix} .$$

From (4.3.95) it follows that $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}$ must be an element of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big)$ of $\mathbf{C}_{\mathrm{u}}^x$, i.e. it should lie in the row space $\mathcal{L}\left(\left\{(\mathbf{u}^2)^T, (\mathbf{u}^3)^T\right\}\right)$ of $\boldsymbol{\Phi}$ in (4.4.23). This can easily

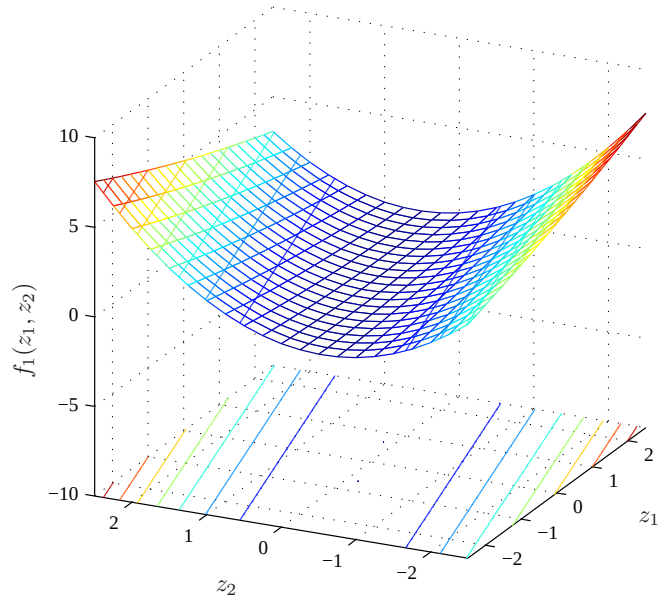

Figure 4.10: *Graph of $f_{\mathbf{A}}(z_1, z_2)$ for example in Section 4.4.3.1 with rank-deficient mixing matrix.*

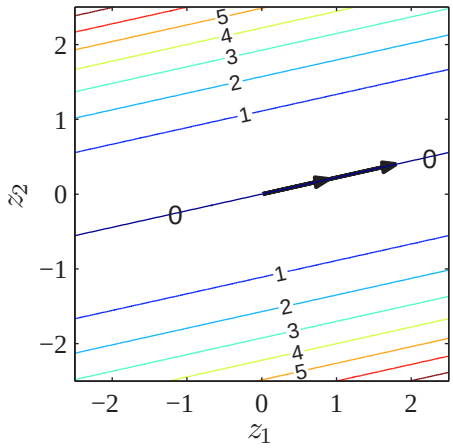

Figure 4.11: *Contour plot of $f_{\mathbf{A}}(z_1, z_2)$ for example in Section 4.4.3.1 with rank-deficient mixing matrix.*

be verified both numerically and symbolically. For example, the symbolical proof directly follows from (4.3.95) and (4.3.100), which state that $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}} \in \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^{x}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},\diamond}^{2}\big)$. This is easily verified by computing the inner product with either column of $\mathbf{A}_{\mathrm{u},\diamond}^{2}$, for example the

first:

$$\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}\, \check{\mathbf{a}}^1 = \begin{bmatrix} a_2^1 a_2^1 & -2a_1^1 a_2^1 & a_1^1 a_1^1 \end{bmatrix} \begin{bmatrix} a_1^1 a_1^1 \\ a_1^1 a_2^1 \\ a_2^1 a_2^1 \end{bmatrix} = 0\,.$$

These latter derivations and results confirm that we have derived correct and useful results. The graph and contour plot of $f_{\mathbf{A}}(z_1, z_2)$ are shown in Figures 4.10 and 4.11 on the previous page respectively. Again, the black arrows in the latter figure represent the columns of the mixing matrix. By comparing these figures to Figures 4.5 and 4.6 respectively, it can be seen that the graph of $f_{\mathbf{A}}(z_1, z_2)$ is a degenerated version of the graph of a general homogeneous bivariate polynomial of degree two. Instead of a saddle-shaped hyperbolic paraboloid, the quadric surface in Fig.4.10 is a parabolic valley. The traces of this parabolic valley in planes parallel to the $z_1 - z_2$ plane, i.e. the contour levels, are straight lines (in the direction of the columns of $\mathbf{A}$) instead of hyperbolas. The trace in the $z_1 - z_2$ plane, which in the general case comprises a pair of lines intersecting at the origin, now consists of a single line through the origin. In other words, the cone representing the zero contour level of $f_{\mathbf{A}}(z_1, z_2)$ has degenerated into a single line through the origin in the direction of the columns of $\mathbf{A}$. This is also clear from the fact that $f_{\mathbf{A}}(z_1, z_2) = (a_2^1 z_1 - a_1^1 z_2)^2 = 0$ has the same solutions as the equation $a_2^1 z_1 - a_1^1 z_2 = 0$. As we have remarked above, the row vector $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}$ is an element of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big)$ of $\mathbf{C}_{\mathrm{u}}^x$. This means that the function $f_{\mathbf{A}}(z_1, z_2)$ in (4.4.26) is a linear combination of the functions $f_1(z_1, z_2)$ and $f_2(z_1, z_2)$ in (4.4.24). In particular, its coefficient vector $\tilde{\boldsymbol{\varphi}}_{\mathbf{A}}$ can be written and computed as follows:

$$\tilde{\boldsymbol{\varphi}}_{\mathbf{A}} = \big\langle \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}, (\mathbf{u}^1)^T \big\rangle\, (\mathbf{u}^1)^T + \big\langle \tilde{\boldsymbol{\varphi}}_{\mathbf{A}}, (\mathbf{u}^2)^T \big\rangle\, (\mathbf{u}^2)^T = \tilde{\boldsymbol{\varphi}}_{\mathbf{A}} \boldsymbol{\Phi}^T \boldsymbol{\Phi}\,.$$

4.4.3.2 Employing subspace matrix estimated from simulated sensor data

Now we will compute and plot the same contour levels as in the previous section for a subspace matrix estimated from computer generated signals. The same source and noise signals as for the full rank example in Section 4.4.2.2 are used. Again, the sensor signals $x_1[n]$ and $x_2[n]$ are computed according to the model in (4.1.2), where the rank-deficient mixing matrix $\mathbf{A}$ is given by (4.4.21). The resulting standard deviations of the noise-free parts of the sensor signals $x_1[n]$ and $x_2[n]$ are given by 2.01 and 0.45 respectively. Hence, the SNR at the first and second sensors equal 12.1 and -1.0 dB respectively. Fig. 4.12 on the facing page shows 200 samples of $s_1[n]$ and $s_2[n]$ at the left side, and the noisy mixtures $x_1[n]$ and $x_2[n]$ at the right side. The subspace matrix $\mathbf{C}_{\mathrm{u}}^x$ is estimated from the sensor signals in the same manner as in Section 4.4.2.2 resulting in:

$$\hat{\mathbf{C}}_{\mathrm{u}}^x = \begin{bmatrix} \hat{r}_{11}^x[1] & \hat{r}_{11}^x[2] \\ \hat{r}_{12}^x[1] & \hat{r}_{12}^x[2] \\ \hat{r}_{22}^x[1] & \hat{r}_{22}^x[2] \end{bmatrix} = \begin{bmatrix} 2.5628 & 1.5961 \\ 0.5900 & 0.3748 \\ 0.1454 & 0.0918 \end{bmatrix}. \tag{4.4.27}$$

From the SVD of $\hat{\mathbf{C}}_{\mathrm{u}}^x$ it now follows that an estimate $\hat{\boldsymbol{\Phi}}$ of a nontrivial matrix $\boldsymbol{\Phi}$ is given by:

$$\hat{\boldsymbol{\Phi}} = \begin{bmatrix} \hat{\tilde{\boldsymbol{\varphi}}}_1 \\ \hat{\tilde{\boldsymbol{\varphi}}}_2 \end{bmatrix} = \big[\hat{\mathbf{u}}^2\ \hat{\mathbf{u}}^3\big]^T = \begin{bmatrix} 0.2314 & -0.9586 & -0.1657 \\ -0.0158 & -0.1740 & 0.9846 \end{bmatrix}. \tag{4.4.28}$$

Note that the rows of this matrix are not necessarily estimates of the rows of $\boldsymbol{\Phi}$ in (4.4.23). The only requirement is that the subspace spanned by the rows of $\hat{\boldsymbol{\Phi}}$ approximate the subspace spanned by the rows of $\boldsymbol{\Phi}$. In other words, the left null space $\mathcal{N}_{\mathrm{l}}\big(\hat{\mathbf{C}}_{\mathrm{u}}^x\big)$ of $\hat{\mathbf{C}}_{\mathrm{u}}^x$ should approximate the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u}}^x\big)$ of $\mathbf{C}_{\mathrm{u}}^x$ (see Section 4.3.6). This can easily shown

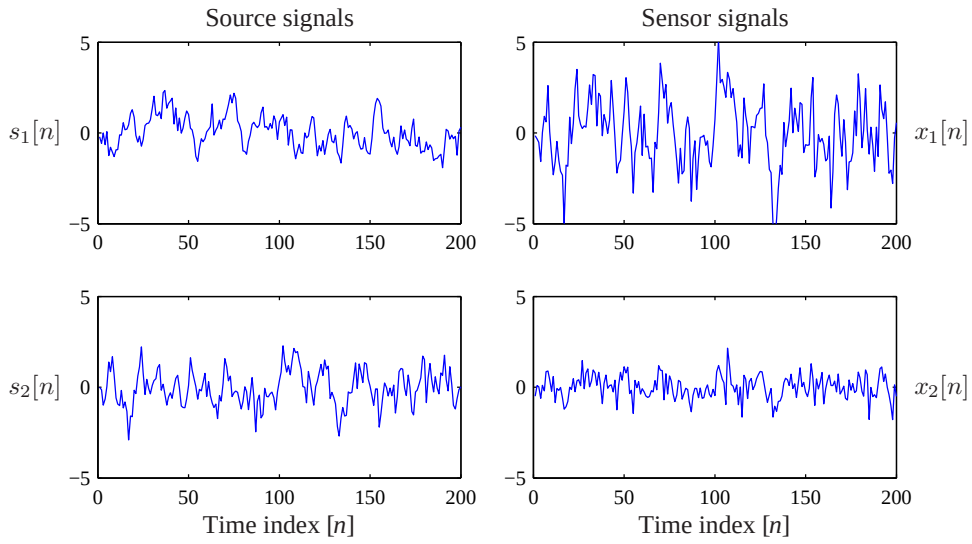

Figure 4.12: *Stationary Gaussian $AR(1)$ source signals (left) and their noisy mixtures (right) for example in Section 4.4.3.2 with rank-deficient mixing matrix.*

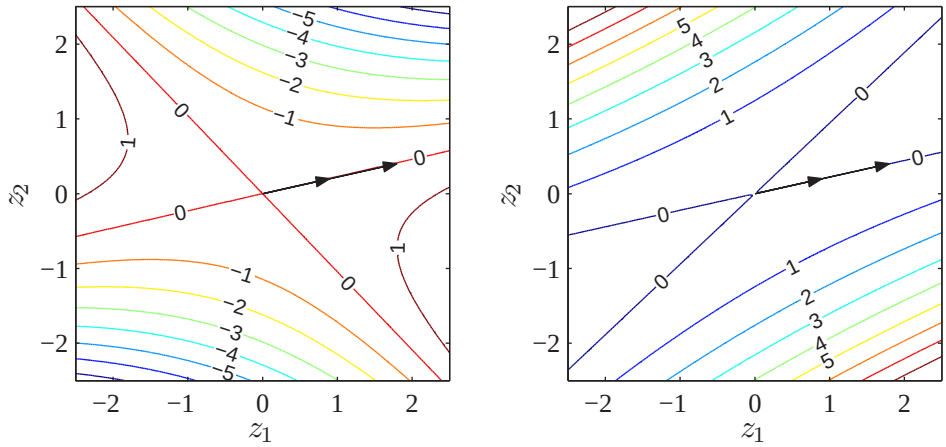

Figure 4.13: *Contour plots of $\hat{f}_1(z_1, z_2)$ (left) and $\hat{f}_2(z_1, z_2)$ (right) for example in Section 4.4.3 with rank-deficient mixing matrix.*

to be the case by computing and comparing the reduced row echelon forms of (4.4.23) and (4.4.28). The functions $\hat{f}_1(z_1, z_2)$ and $\hat{f}_2(z_1, z_2)$ corresponding to the rows of $\hat{\mathbf{\Phi}}$ are:

$$\begin{cases} \hat{f}_1(z_1, z_2) = \quad 0.2314\, z_1 z_1 - 0.9586\, z_1 z_2 - 0.1657\, z_2 z_2\,; \\ \hat{f}_2(z_1, z_2) = -0.0158\, z_1 z_1 - 0.1740\, z_1 z_2 + 0.9846\, z_2 z_2\,. \end{cases} \tag{4.4.29}$$

The corresponding contour plots are shown in Fig. 4.13 and should be compared to those in Fig. 4.9 on page 182. As in the previous contour plots, the black arrows represent the

columns of the true mixing matrix. The two lines through the origin of each figure define the zero contour level of the associated function. Since the black arrows approximately point in the same direction as the zero contour lines of the estimated functions, the estimated columns $\hat{\mathbf{a}}^1$ and $\hat{\mathbf{a}}^2$ of the mixing matrix approximate the true ones rather well (recall that we only need to estimate one column of $\mathbf{A}$ because the two columns are scaled versions of each other). Clearly, $\hat{\mathbf{a}}^1$ and $\hat{\mathbf{a}}^2$ are determined uniquely by the intersection of the zero contour levels of $\hat{f}_1(z_1, z_2)$ and $\hat{f}_2(z_1, z_2)$. Hence, they can be found by solving the system $\{\hat{f}_1(z_1, z_2) = 0, \hat{f}_2(z_1, z_2) = 0\}$, which is the topic of Section 4.5.

4.4.4 Example 3: Speech signals mixed by full rank mixing matrix

In this section, similarly to Sections 4.4.3 and 4.4.2 we analyze and perform the various steps of Alg. 4.3 on page 171, but now very briefly and for a different signal scenario, viz. the example with mixed speech signals and zero additive sensor noise that was introduced in Section 1.1.2 of Chapter 1. Although we do not exploit the non-stationary character of the speech signals (see Section 5.4.2 for this) but only the non-whiteness, i.e. we pretend that the correlation functions are obtained from stationary data and use (4.4.18) to estimate them, this shows that our approach also works for other types of signals.

Fig. 1.5 on page 8 shows 8000 samples (one second) of the original speech signals at the left side, and the two mixtures obtained from (1.1.1) with the full rank mixing matrix given by (1.1.4), and with zero additive noise, at the right side. Note that $\mathbf{A}$ is well-conditioned because its columns are orthonormal. As in Section 4.4.2.1, because $\mathbf{A}$ is full rank there is only one equation $f(z_1, z_2) = 0$ in the system with $f(z_1, z_2)$ defined in (4.3.31). It can easily be shown that this function is given by:

$$f(z_1, z_2) = -\frac{1}{2}\sqrt{2}\, z_1 z_1 + \frac{1}{2}\sqrt{2}\, z_2 z_2 \,. \tag{4.4.30}$$

The corresponding contour plot is shown at the left side of Fig. 4.14. Again, the black arrows represent the columns of the mixing matrix and the two lines through the origin define the zero

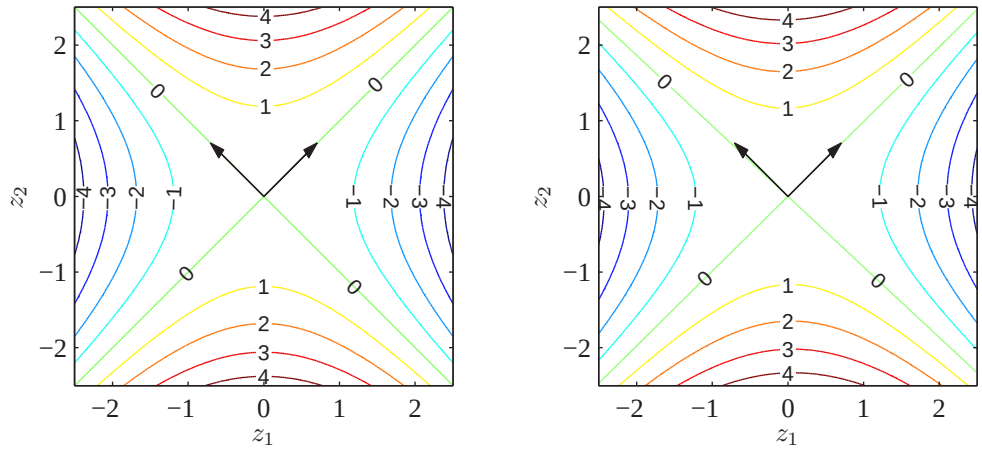

Figure 4.14: *Contour plots of $f(z_1, z_2)$ (left) and $\hat{f}(z_1, z_2)$ (right) for example with speech signals in Section 4.4.4.*

contour level of $f(z_1, z_2)$. Clearly, the arrows representing the columns of the mixing matrix are exactly in the directions of the two lines describing the zero contour level of $f(z_1, z_2)$, thereby revealing that both columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of $\mathbf{A}$ are determined correctly and uniquely by the zero contour level. Using (4.4.18) with $N_s = 8000$ we estimate the sensor correlation functions $r^x_{11}[k], r^x_{12}[k]$ and $r^x_{22}[k]$ for all lags $k \in \Omega_k^{s\backslash\nu} = \{1, 2\}$ from the sensor signals. Applying Alg. 4.3 to the resulting estimate $\hat{\mathbf{C}}^x_{\mathrm{u}}$ of the subspace matrix yields the following estimate estimate $\hat{f}(z_1, z_2)$ of $f(z_1, z_2)$:

$$\hat{f}(z_1, z_2) = -0.6766\, z_1 z_1 - 0.0409\, z_1 z_2 + 0.7352\, z_2 z_2\,. \tag{4.4.31}$$

The contour plot of this function is shown at the right side of Fig. 4.14. Clearly, the ideal and estimated contour plots resemble each other very much. In the estimated contour plot we see a slight misalignment between the two lines through the origin defining the zero contour level of $\hat{f}(z_1, z_2)$ on the one hand, and the black arrows representing the columns of the true mixing matrix on the other hand. If the columns of the mixing matrix would be estimated by solving $\hat{f}(z_1, z_2) = 0$, the solution vectors would point in the direction of the contour lines. Since the black arrows approximately point in the same directions, the estimated columns $\hat{\mathbf{a}}^1$ and $\hat{\mathbf{a}}^2$ of the mixing matrix approximate the true ones rather well. As will be explained in Section 4.5, these columns, and thus the mixing matrix, can be estimated in several manners. For one method Fig. 4.18 shows the two speech source signals at the left side and the separated speech signals obtained by using the inverse of the estimated mixing matrix at the right side.

4.4.5 Example 4: Images mixed by full rank mixing matrix

In this section we analyze and perform the various steps of Alg. 4.3, but now for the example with mixed images and zero additive sensor noise that was introduced in Section 1.1.2 of Chapter 1. Similarly to the previous section, although we do not exploit the non-stationary character of the image signals but only the non-whiteness, this shows that our approach also works for other types of signals.

Fig. 1.6 on page 9 shows the two source images at the left side and the two mixtures at the right side. The mixed images were obtained from (1.1.5) with (full rank) mixing matrix:

$$\mathbf{A} = \frac{1}{5}\begin{bmatrix} 4 & 3 \\ 3 & 4 \end{bmatrix}\,, \tag{4.4.32}$$

and zero additive noise. Note that the columns of this matrix are rather close, see also Fig. 4.15. Recall that u and v in (1.1.5) represent the pixel indices in the horizontal and vertical directions respectively. The size of the images is $N \times N$ with $N = 512$. As in Section 4.4.2.1, because $\mathbf{A}$ is full rank there is only one equation $f(z_1, z_2) = 0$ in the system with $f(z_1, z_2)$ defined in (4.3.31). From Alg. 4.3 it follows that this function is given by:

$$f(z_1, z_2) = 0.3971\, z_1 z_1 - 0.8274\, z_1 z_2 + 0.3971\, z_2 z_2\,. \tag{4.4.33}$$

The corresponding contour plot is shown at the left side of Fig. 4.15. Again, the black arrows represent the columns of the mixing matrix and the two lines through the origin define the zero contour level of $f(z_1, z_2)$. Clearly, the arrows representing the columns of the mixing matrix are exactly in the directions of the two lines describing the zero contour level of $f(z_1, z_2)$, thereby revealing that both columns $\mathbf{a}^1$ and $\mathbf{a}^2$ of $\mathbf{A}$ are determined correctly and uniquely by the zero contour level.

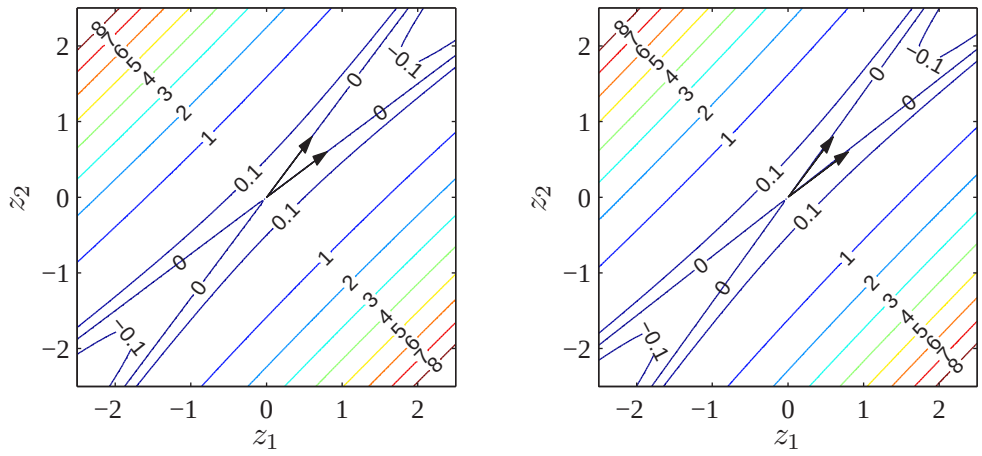

Figure 4.15: *Contour plots of $f(z_1, z_2)$ (left) and $\hat{f}(z_1, z_2)$ (right) for example with images in Section 4.4.5.*

From the 'sensor signals', i.e. the mixed images, we compute the spatially averaged covariance functions $c^x_{11}[m,n], c^x_{12}[m,n]$ and $c^x_{22}[m,n]$ for $(m,n) \in \Omega^{s\backslash\nu}_{m,n} \triangleq \{(0,0),(-1,1)\}$ by averaging products of the form $x_{i_1}[u,v]\, x_{i_2}[u-m, v-n]$ over the available pixels:

$$c^x_{i_1 i_2}[m,n] \triangleq \frac{1}{(N-m)(N-n)} \sum_{u=m+1}^{N} \sum_{v=n+1}^{N} \left(x_{i_1}[u,v] - \mu^x_{i_1}\right) \left(x_{i_2}[u-m, v-n] - \mu^x_{i_2}\right), \tag{4.4.34}$$

where the spatially averaged mean μ^x_i of $x_i[u,v]$ is defined as follows:

$$\mu^x_i \triangleq \frac{1}{(N)^2} \sum_{u=1}^{N} \sum_{v=1}^{N} x_i[u,v]\,, \qquad i = 1,2\,. \tag{4.4.35}$$

Using this formula for estimating the subspace matrix from the mixed image signals yields:

$$\hat{\mathbf{C}}^x_{\mathrm{u}} = \begin{bmatrix} c^x_{11}[0,0] & \hat{r}^x_{11}[-1,1] \\ c^x_{12}[0,0] & \hat{r}^x_{12}[-1,1] \\ c^x_{22}[0,0] & \hat{r}^x_{22}[-1,1] \end{bmatrix} = \begin{bmatrix} 0.0429 & 0.0411 \\ 0.0367 & 0.0350 \\ 0.0331 & 0.0314 \end{bmatrix}. \tag{4.4.36}$$

Similarly to the previous examples, using Alg. 4.3 the estimate $\hat{f}(z_1, z_2)$ of $f(z_1, z_2)$ can be computed from $\hat{\mathbf{C}}^x_{\mathrm{u}}$ and is given by:

$$\hat{f}(z_1, z_2) = 0.4082\, z_1 z_1 - 0.8265\, z_1 z_2 + 0.3876\, z_2 z_2\,. \tag{4.4.37}$$

The contour plot of this function is shown at the right side of Fig. 4.15. Clearly, the ideal and estimated contour plots resemble each other very much. Again, in the estimated contour plot we see a slight misalignment between the two lines through the origin defining the zero contour level of $\hat{f}(z_1, z_2)$ on the one hand, and the black arrows representing the columns of the true mixing matrix on the other hand. As will be explained in the next section, the mixing matrix can be estimated in several manners. For one method Fig. 4.19 shows the two source images at the left side and the separated images obtained by using the inverse of the estimated mixing matrix at the right side.

4.5 Solving the system of polynomial equations

The derivations in Section 4.3 and the illustrations in Section 4.4 showed that the MIBI problem can be projected onto the problem of solving a system of bivariate homogeneous polynomial equations of degree two for the columns of the mixing matrix $\mathbf{A}$. In this section, we discuss the final step of Alg. 4.3 on page 171, viz. that of solving the system $\{f_q(z_1, z_2) = 0\}_{q\in\mathcal{Q}}$ in (4.3.33) for the columns of $\mathbf{A}$. Using (4.3.31), the target system can be written in expanded form as follows:

$$\left\{\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 = 0\right\}_{q\in\mathcal{Q}} . \tag{4.5.1}$$

or, even more elaborately:

$$\begin{cases} f_1(z_1, z_2) &= \varphi_1^{11} z_1 z_1 + \varphi_1^{12} z_1 z_2 + \varphi_1^{22} z_2 z_2 &= 0\,; \\ \vdots & \vdots & \vdots & \vdots \ \vdots \\ f_Q(z_1, z_2) &= \varphi_Q^{11} z_1 z_1 + \varphi_Q^{12} z_1 z_2 + \varphi_Q^{22} z_2 z_2 &= 0\,. \end{cases} \tag{4.5.2}$$

In the next sections we will present several approaches for solving this system, but first we discuss some general topics. In Sections 4.3.4.4 and 4.4 we have provided insight into the structure of an individual equation $f_q(z_1, z_2) = 0$. We have seen that the zero contour level of each such function consists of a set of two lines through the origin, and thus forms a cone in the two-dimensional Euclidian space. As an example, see the zero contour level of $f(z_1, z_2)$ in (4.4.13) that is depicted in Fig. 4.6 on page 177. Note that in this figure several contour levels are depicted. For convenience and reference, in Fig. 4.16 we have extracted the zero contour level of $f(z_1, z_2)$ defined in (4.4.13) only. Due to the scaling indeterminacy discussed in Section 2.4, the norm of the solution vectors of each individual equation in system (4.5.2), as well as the norm of the solution vectors of the whole system itself, cannot be determined and will be assumed to be non-zero. If no a priori knowledge is available, we typically assume that the solution vectors have unit Euclidian norm, i.e. $\|\mathbf{z}\| = 1$. Let the unit-norm constraint function $c(z_1, z_2)$ be defined by:

$$c(z_1, z_2) \triangleq (z_1)^2 + (z_2)^2 - 1\,, \tag{4.5.3}$$

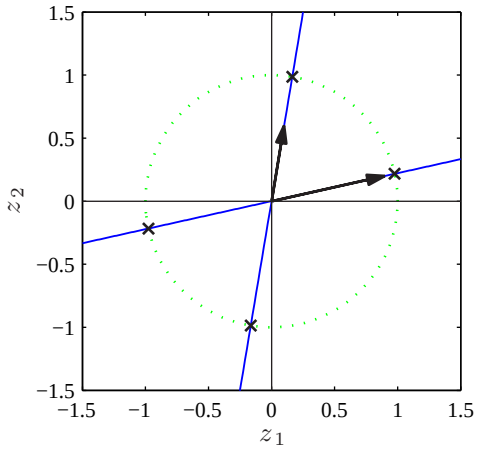

Figure 4.16: *Zero contour plot of function $f(z_1, z_2)$ occurring in full rank example of Section 4.4.2 together with unit-norm constraint.*

then the unit-norm constraint can be written as the equation $c(z_1, z_2) = 0$. This constraint is indicated by the dotted circle in Fig. 4.16.

As is clear from Fig. 4.16, in general there are four intersection points between the zero contour level of an individual function and the unit-norm constraint; these point are indicated by the black crosses. Essentially, only two of them are different because points that are opposite w.r.t. the origin are equivalent from the solution point of view; see also (4.4.9) on page 175. We make this statement precise as follows. The homogeneity property formulated in (4.3.35) implies that only solution vectors can be distinguished that are different according to the following equivalence relation $\sim$ defined on the non-zero points of $\mathbb{R}^2$:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \sim \begin{bmatrix} z_1' \\ z_2' \end{bmatrix} \quad \Longleftrightarrow \quad \exists\, \eta \in \mathbb{R}, \eta \neq 0 : \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \eta z_1' \\ \eta z_2' \end{bmatrix} . \tag{4.5.4}$$

Hence, vectors $\mathbf{z} \triangleq \left[\begin{smallmatrix} z_1 \\ z_2 \end{smallmatrix}\right]$ and $\mathbf{z}' \triangleq \left[\begin{smallmatrix} z_1' \\ z_2' \end{smallmatrix}\right]$ with arbitrary lengths and the same or opposite directions are considered to be equivalent according to this relation, and all other vectors are different. In the coming sections our purpose is to find all unit-norm solutions of system (4.5.2) that are different according to relation (4.5.4). For a single homogeneous polynomial equation this means that for each of the two lines comprising the zero contour level one of the two intersection points with the circle has to be found, see Fig. 4.16.

In the previous sections we have seen that for the scenario studied in this chapter there usually only is a single equation in the system to be solved, but sometimes there are two. In addition, when dealing with estimated data and equations, in the latter case there may not exist exact solutions. Therefore, in order to be able to solve the general case with multiple equations with possibly inexact coefficients, in Sections 4.5.1, 4.5.2, and 4.5.4 we will define several appropriate cost functions from which the solution vectors can be found by minimization. In Section 4.5.3, we will show how the solutions of a single bivariate homogeneous polynomial equation of arbitrary degree can be found. This technique will be used in Section 4.5.4 and in later chapters.

4.5.1 Bivariate homogeneous polynomial cost function

Consider the following cost function that takes into account all possibly inexact equations of system (4.5.2):

$$J(z_1, z_2) \triangleq \sum_{q \in \mathcal{Q}} \big(f_q(z_1, z_2)\big)^2 = \sum_{q \in \mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)^2 . \tag{4.5.5}$$

Note that this function is a bivariate 4-homogeneous polynomial. The values of (z_1, z_2) for which $J(z_1, z_2)$ attains its minima can be found by determining where the partial derivatives w.r.t. z_1 and z_2 equal zero:

$$\begin{cases} \dfrac{\partial J(z_1, z_2)}{\partial z_1} = 2 \displaystyle\sum_{q \in \mathcal{Q}} f_q(z_1, z_2) \, \frac{\partial f_q(z_1, z_2)}{\partial z_1} = 0 \,; \\ \dfrac{\partial J(z_1, z_2)}{\partial z_2} = 2 \displaystyle\sum_{q \in \mathcal{Q}} f_q(z_1, z_2) \, \frac{\partial f_q(z_1, z_2)}{\partial z_2} = 0 \,. \end{cases} \tag{4.5.6}$$

This yields a system of two bivariate homogeneous polynomial equations of degree three:

$$\begin{cases} \sum\limits_{q\in\mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)\big(2\varphi_q^{11} z_1 + \varphi_q^{12} z_2\big) = 0\,; \\ \sum\limits_{q\in\mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)\big(\varphi_q^{12} z_1 + 2\varphi_q^{22} z_2\big) = 0\,. \end{cases} \tag{4.5.7}$$

In principle, this system can be solved by solving each equation individually (see Section 4.5.3), and then selecting the common solution vectors. There also exist of-the-shelf algorithms in packages like Matlab and Mathematica for solving relatively simple systems as in (4.5.7). Rather than simply using these functions, in Section 4.5.6 we will explain in detail the so-called *homotopy* [2, 144] method, which can also be used for solving the more complex systems that we will encounter in later chapters.

4.5.2 Bivariate polynomial cost function with unit-norm constraint

We can also define a bivariate cost function that takes into account the unit-norm constraint (4.5.3) directly as follows:

$$\begin{aligned} J(z_1, z_2) &\triangleq \big(c(z_1, z_2)\big)^2 + \sum_{q\in\mathcal{Q}} \big(f_q(z_1, z_2)\big)^2 \\ &= \big((z_1)^2 + (z_2)^2 - 1\big)^2 + \sum_{q\in\mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)^2\,. \end{aligned} \tag{4.5.8}$$

Note that contrary to (4.5.5) this polynomial is non-homogeneous due to the additional unit-norm term. Again, the values of (z_1, z_2) for which $J(z_1, z_2)$ attains its minima can be found by determining where the partial derivatives w.r.t. z_1 and z_2 equal zero:

$$\begin{cases} \dfrac{\partial J(z_1, z_2)}{\partial z_1} = 2\, c(z_1, z_2)\, \dfrac{\partial c(z_1, z_2)}{\partial z_1} + 2 \sum\limits_{q\in\mathcal{Q}} f_q(z_1, z_2)\, \dfrac{\partial f_q(z_1, z_2)}{\partial z_1} = 0\,; \\ \dfrac{\partial J(z_1, z_2)}{\partial z_2} = 2\, c(z_1, z_2)\, \dfrac{\partial c(z_1, z_2)}{\partial z_2} + 2 \sum\limits_{q\in\mathcal{Q}} f_q(z_1, z_2)\, \dfrac{\partial f_q(z_1, z_2)}{\partial z_2} = 0\,. \end{cases} \tag{4.5.9}$$

This yields a system consisting of two bivariate polynomial equations of degree three:

$$\begin{cases} 2\, z_1 \big((z_1)^2 + (z_2)^2 - 1\big) + \sum\limits_{q\in\mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)\big(2\varphi_q^{11} z_1 + \varphi_q^{12} z_2\big) = 0\,; \\ 2\, z_2 \big((z_1)^2 + (z_2)^2 - 1\big) + \sum\limits_{q\in\mathcal{Q}} \big(\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2\big)\big(\varphi_q^{12} z_1 + 2\varphi_q^{22} z_2\big) = 0\,. \end{cases} \tag{4.5.10}$$

Similarly to system (4.5.7), in principle this system can be solved by solving each equation individually and then selecting the common solution vectors. See also the other remarks made below (4.5.7).

4.5.3 Finding all different solutions of a bivariate homogeneous polynomial equation of arbitrary degree

In this section we show how all essentially different solution vectors of a single bivariate m-homogeneous polynomial equation $p(z_1, z_2) = 0$ equation of arbitrary degree m can be found. Because $p(z_1, z_2)$ is m-homogeneous all its terms have the same degree m; hence, it can be written as follows (see also Chapter 6, in particular Eq. (6.2.64) and Section 6.2.4.4):

$$p(z_1, z_2) \triangleq \sum_{(i_1,\ldots,i_m)\in\mathcal{I}^m_{\mathrm{a},D}} \alpha^{i_1\cdots i_m}\, z_{i_1}\cdots z_{i_m} = \sum_{\mathbf{i}_m\in\mathcal{I}^m_{\mathrm{a},D}} \alpha^{\mathbf{i}_m}\, \check{z}_{\mathbf{i}_m} \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \tag{4.5.11}$$

where $\{\alpha^{\mathbf{i}_m}\}^{\mathbf{i}_m\in\mathcal{I}^m_{\mathrm{a},D}}$ is a set of real- or complex-valued scalars, and the index set $\mathcal{I}^m_{\mathrm{a},D}$ is defined in (A.2.15) on page 407 as $\mathcal{I}^m_{\mathrm{a},D} \triangleq \{\mathbf{i}_m \mid 1 \le i_1 \le \cdots \le i_m \le D\}$. In mathematical terms, the m-homogeneity of $p(z_1, z_2)$ can be formulated as follows:

$$p(\eta z_1, \eta z_2) = (\eta)^m p(z_1, z_2) \qquad \forall\, \eta, z_1, z_2 \in \mathbb{C}\,. \tag{4.5.12}$$

See also (4.3.34) and (6.2.67). Similarly to (4.3.35) and (6.2.68), because of its homogeneity the function $p(v_1, v_2)$ has the following nice property:

$$p(v_1, v_2) = 0 \qquad \Longrightarrow \qquad p(\eta v_1, \eta v_2) = 0 \qquad \forall\, \eta \in \mathbb{C}\,. \tag{4.5.13}$$

Hence, if (v_1, v_2) is a solution of the equation $p(v_1, v_2) = 0$, then so is $(\eta v_1, \eta v_2)$ for all $\eta \in \mathbb{C}$. Thus, there is a scaling indeterminacy in the solution vectors.

Now we show how the solutions of $p(z_1, z_2) = 0$ that are different according to equivalence relation (4.5.4) can be found by deriving a related univariate polynomial equation. To start with, note that according to (4.5.4), the solution vectors $\mathbf{z} \triangleq \left[\begin{smallmatrix} z_1 \\ z_2 \end{smallmatrix}\right]$ and $\mathbf{z}' \triangleq \left[\begin{smallmatrix} 1 \\ \frac{z_2}{z_1} \end{smallmatrix}\right]$ are equivalent because $\mathbf{z} \triangleq z_1 \left[\begin{smallmatrix} 1 \\ \frac{z_2}{z_1} \end{smallmatrix}\right] = z_1\mathbf{z}'$, where for convenience it has been assumed that $z_1 \neq 0$. Hence, it is sufficient to determine the ratio $r_1^2 \triangleq \frac{z_2}{z_1}$ of each solution vector $\mathbf{z}$. The symbol r_i^j should be read as 'the ratio of z_j to z_i'. Clearly, if $z_1 = 0$ for some solution, the ratio $\frac{z_2}{z_1}$ is not defined. However, in this case the ratio $\frac{z_1}{z_2}$ is well defined and can be considered instead of $\frac{z_2}{z_1}$ because from $\|\mathbf{z}\|^2 = (z_1)^2 + (z_2)^2 \neq 0$ it follows that $z_2 \neq 0$. For this scenario we would write $\mathbf{z} \triangleq z_2 \left[\begin{smallmatrix} \frac{z_1}{z_2} \\ 1 \end{smallmatrix}\right] = z_2\mathbf{z}'$ for $\mathbf{z}' \triangleq \left[\begin{smallmatrix} \frac{z_1}{z_2} \\ 1 \end{smallmatrix}\right]$. Without loss of generality, in the sequel we will only consider the case $z_1 \neq 0$. Since $\mathbf{z}$ and $\mathbf{z}'$ are equivalent, solving $p(\mathbf{z}) = 0$ is equivalent to solving $p(\mathbf{z}') = 0$. This can easily be seen from (4.5.12): $p(\mathbf{z}) = p(z_1\mathbf{z}') = \big(z_1\big)^m p(\mathbf{z}')$. Hence, if $z_1 \neq 0$ solving $p(\mathbf{z}') = 0$ for $\mathbf{z}'$ is equivalent to solving $p(\mathbf{z}) = 0$ for $\mathbf{z}$. Now, substituting $\mathbf{z}' = \left[\begin{smallmatrix} 1 \\ \frac{z_2}{z_1} \end{smallmatrix}\right] = \left[\begin{smallmatrix} 1 \\ r_1^2 \end{smallmatrix}\right]$ into the bivariate polynomial $p\left(\mathbf{z}'\right) = p\left(z_1', z_2'\right)$ defined in (4.5.11) yields a univariate polynomial of degree m in the ratio r_1^2:

$$g(r_1^2) \triangleq p(1, r_1^2)\,. \tag{4.5.14}$$

For example, suppose that $p(\mathbf{z}) = f_q(\mathbf{z})$ with $f_q(\mathbf{z})$ given by (4.3.31). Then, (4.5.14) becomes:

$$g(r_1^2) \triangleq p(1, r_1^2) = \varphi_q^{11} + \varphi_q^{12} r_1^2 + \varphi_q^{22}\big(r_1^2\big)^2, \tag{4.5.15}$$

which is a second-degree univariate polynomial in the ratio r_1^2. The general univariate equation $g(r_1^2) = 0$ with $g(r_1^2)$ defined in (4.5.14) is a standard univariate polynomial equation

of degree m and thus can easily be solved. Once the solutions for r_1^2 have been found, the corresponding solution vectors $\mathbf{z}' = \begin{bmatrix} 1 \\ \frac{z_2}{z_1} \end{bmatrix} = \begin{bmatrix} 1 \\ r_1^2 \end{bmatrix}$ can easily be computed. If desired, $\mathbf{z}'$ can be normalized to unit Euclidian norm:

$$\hat{\mathbf{a}} \triangleq \frac{\mathbf{z}'}{\|\mathbf{z}'\|} = \frac{1}{\sqrt{1+(r_1^2)^2}} \begin{bmatrix} 1 \\ r_1^2 \end{bmatrix} .$$

In principle, with the technique developed above we can solve systems (4.5.7) and (4.5.10) by finding the solutions of each individual equation first, and then selecting the common solution(s). However, as we will show in the next section we can also reduce the whole problem to finding the roots of a single univariate polynomial equation of degree three.

4.5.4 Univariate polynomial cost function

For convenience, let us (momentarily) denote the ratio $r_1^2 = z_2/z_1$ defined in the previous section by r. Then, with abuse of notation we can define the following univariate cost function $J(r)$ by substituting $\begin{bmatrix} z_1' \\ z_2' \end{bmatrix} = \begin{bmatrix} 1 \\ r \end{bmatrix}$ into (4.5.5):

$$J(r) \triangleq J(1,r) \triangleq \sum_{q\in\mathcal{Q}} \big(f_q(1,r)\big)^2 = \sum_{q\in\mathcal{Q}} \big(g_q(r)\big)^2 = \sum_{q\in\mathcal{Q}} \left(\varphi_q^{11} + \varphi_q^{12}\, r + \varphi_q^{22}\,(r)^2\right)^2 , \tag{4.5.16}$$

where:

$$g_q(r) \triangleq f_q(1,r) = \varphi_q^{11} + \varphi_q^{12}\, r + \varphi_q^{22}\,(r)^2 . \tag{4.5.17}$$

The values of r for which $J(r)$ attains its minima can be found by determining where the derivative $J'(r)$ equals zero:

$$J'(r) \triangleq \frac{dJ(r)}{dr} = 2\sum_{q\in\mathcal{Q}} g_q(r)\, g_q'(r) = 2\sum_{q\in\mathcal{Q}} \left(\varphi_q^{11} + \varphi_q^{12}\, r + \varphi_q^{22}\,(r)^2\right)\left(\varphi_q^{12} + 2\varphi_q^{22}\, r\right) = 0 .$$

Hence, this yields the following univariate polynomial equation of degree three:

$$\sum_{q\in\mathcal{Q}} \left[\varphi_q^{11}\varphi_q^{12} + \left((\varphi_q^{12})^2 + 2\varphi_q^{11}\varphi_q^{22}\right) r + 3\varphi_q^{12}\varphi_q^{22}\,(r)^2 + 2(\varphi_q^{22})^2\,(r)^3\right] = 0 , \tag{4.5.18}$$

whose roots can easily be found (see also the previous section). Since $J'(r)$ is a univariate polynomial of the third degree it has exactly three roots. Furthermore, since its coefficients are real-valued it has either three real roots or one real root and two complex conjugate roots. In the first case we can easily detect the minima by checking for which of the roots the second derivative $J''(r)$ of $J(r)$ is real-valued and larger than zero, and in the second case it is immediately clear that the single real root provides the desired solution. Finally, for each value of r where $J(r)$ attains a (local) minimum, the corresponding unit-norm column of the mixing matrix can be computed by:

$$\hat{\mathbf{z}} = \frac{1}{\sqrt{1+(r)^2}} \begin{bmatrix} 1 \\ r \end{bmatrix} . \tag{4.5.19}$$

4.5.4.1 Application to Example 1 in Section 4.4.2

In this example there is only one equation to be solved and thus we can directly apply the technique developed in Section 4.5.3. We will consider the examples with ideal and estimated subspace matrices in turn. For the ideal case, from (4.4.13) and (4.5.15) it follows that the equation to be solved becomes:

$$g(r_1^2) \triangleq f(1, r_1^2) = 0.2070 - 0.9659\, r_1^2 + 0.1552 \left(r_1^2\right)^2 = 0.$$

Using for example the `roots` function of Matlab or the `Solve` function of Mathematica, the two roots of this equation are found to be $r_1^2 = 0.2222$ and $r_1^2 = 6.000$. Hence, $\mathbf{z}'_1 = [\,{}^{1}_{0.2222}\,]$ and $\mathbf{z}'_2 = [\,{}^{1}_{6.000}\,]$. The ratio 0.2222 equals the ratio $\frac{a_2^1}{a_1^1} = \frac{0.2}{0.9}$ between the second and first elements of the first column $\mathbf{a}^1$ of $\mathbf{A} = [\begin{smallmatrix} 0.9 & 0.1 \\ 0.2 & 0.6 \end{smallmatrix}]$. Similarly, the ratio 6.000 equals the ratio $\frac{a_2^2}{a_1^2} = \frac{0.6}{0.1}$ between the second and first elements of the second column $\mathbf{a}^2$ of $\mathbf{A}$. This means that after normalization to unit norm, each of the solutions for $\mathbf{z}'$ equals a normalized column of $\mathbf{A}$. In other words, $\frac{\mathbf{z}^{j\prime}}{\|\mathbf{z}^{j\prime}\|} = \frac{\mathbf{a}^j}{\|\mathbf{a}^j\|}$ for $j = 1, 2$:

$$\frac{\mathbf{z}^{1\prime}}{\|\mathbf{z}^{1\prime}\|} = \frac{\mathbf{a}^1}{\|\mathbf{a}^1\|} = \begin{bmatrix} 0.9762 \\ 0.2169 \end{bmatrix}, \qquad \frac{\mathbf{z}^{2\prime}}{\|\mathbf{z}^{2\prime}\|} = \frac{\mathbf{a}^2}{\|\mathbf{a}^2\|} = \begin{bmatrix} 0.1644 \\ 0.9864 \end{bmatrix}. \tag{4.5.20}$$

Hence the columns are recovered correctly.

We now perform the same procedure as described above for the estimated case. From (4.4.20) and (4.5.15) it follows that now the equation to be solved becomes:

$$\hat{g}(r_1^2) \triangleq \hat{f}(1, r_1^2) = 0.2168 - 0.9737\, r_1^2 + 0.0702 \left(r_1^2\right)^2 = 0\,.$$

Solving this equation gives the ratios $\hat{r}_1^2 = 0.2264$ and $\hat{r}_1^2 = 13.64$ corresponding to estimates of the first and second columns of $\mathbf{A}$ respectively. The normalized solutions for $\hat{\mathbf{z}}'$ now equal:

$$\frac{\hat{\mathbf{z}}^{1\prime}}{\|\hat{\mathbf{z}}^{1\prime}\|} = \begin{bmatrix} 0.9753 \\ 0.2208 \end{bmatrix}, \qquad \frac{\hat{\mathbf{z}}^{2\prime}}{\|\hat{\mathbf{z}}^{2\prime}\|} = \begin{bmatrix} 0.0731 \\ 0.9973 \end{bmatrix}. \tag{4.5.21}$$

The error in first column is quite small, but the error in the second is larger (this can also be seen in Fig. 4.8 on page 179 and Fig. 4.22 on page 209. We are mainly interested in the correctness of the directions of the columns of the mixing matrix. For the current scenario, these directions can be represented by angles. Hence, a sensible manner for comparing the results is to compare angles. The vector corresponding to $r_1^2 = 6$ associated with the second column of $\mathbf{A}$ is located at 80.5 degrees w.r.t. the positive z_1 axis, whereas that corresponding to $\hat{r}_1^2 = 13.64$ associated with the estimate of the second column of $\mathbf{A}$ is located at 85.8 degrees w.r.t. the positive z_1 axis. Hence, the angular error in the estimate of the second column of $\mathbf{A}$ equals 5.3 degrees.

We will now use the estimates in (4.5.21) to perform separation (see Section 1.1.2). Firstly, we define the estimated mixing matrix $\hat{\mathbf{A}}$ as follows:

$$\hat{\mathbf{A}} \triangleq \left[\frac{\hat{\mathbf{z}}^{1\prime}}{\|\hat{\mathbf{z}}^{1\prime}\|} \quad \frac{\hat{\mathbf{z}}^{2\prime}}{\|\hat{\mathbf{z}}^{2\prime}\|} \right] = \begin{bmatrix} 0.9753 & 0.0731 \\ 0.2208 & 0.9973 \end{bmatrix}. \tag{4.5.22}$$

Because our purpose is to perform IBSS and not noise reduction, for demonstration purposes we will apply the inverse of $\hat{\mathbf{A}}$ to the noise-free sensor signals to recover the source signals.

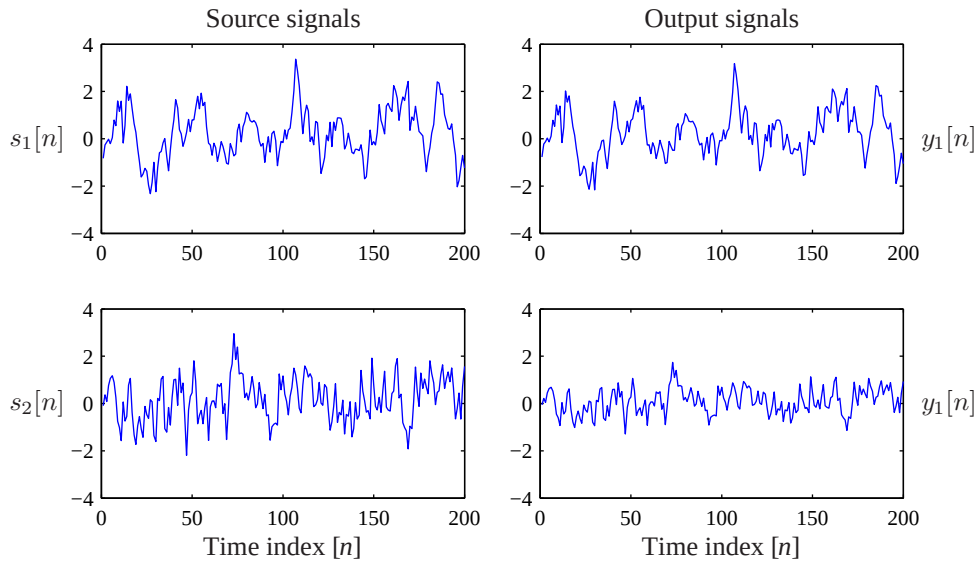

Figure 4.17: *Gaussian $AR(1)$ source signals (left) and noise-free estimated source signals (right) for 'full rank example' in Section 4.4.2.2.*

This can be done in our simulation because we know the noise signals (recall that $\hat{\mathbf{A}}$ has been estimated from the noisy sensor data; see Section4.4.2.2):

$$\mathbf{y}[n] = \hat{\mathbf{A}}^{-1}\left(\mathbf{x}[n] - \boldsymbol{\nu}[n]\right). \tag{4.5.23}$$

Fig. 4.17 on the current page shows 200 samples of the source signals $s_1[n]$ and $s_2[n]$ at the left side (see also the left side of Fig. 4.7 on page 179), and the noise-free estimated source signals $y_1[n]$ and $y_2[n]$ at the right side. As a measure of performance, we compute the total transfer matrix from the source signals to the output signals:

$$\mathbf{T} \triangleq \hat{\mathbf{A}}^{-1}\mathbf{A} = \begin{bmatrix} 0.9230 & 0.0584 \\ -0.0038 & 0.5887 \end{bmatrix}.$$

Hence, the contributions of $s_1[n]$ and $s_2[n]$ to $y_1[n]$ equal 94.0% and 6.0% respectively, whereas the contributions of $s_1[n]$ and $s_2[n]$ to $y_2[n]$ equal 0.6% and 99.4% respectively.

4.5.4.2 Application to Example 2 in Section 4.4.3

In this example, the system to be solved contains two equations. Therefore, we will employ the technique developed in Section 4.5.4. Again, we will consider the examples with ideal and estimated subspace matrices in turn. For the ideal case, from (4.4.24) and (4.5.15) it follows that the functions $g_1(r)$ and $g_2(r)$ used in the definition of the cost function $J(r)$ in (4.5.16) are given by:

$$\begin{cases} g_1(r) \triangleq f_1(1,r) = -0.2167 + 0.9762\,r - 0.0053\,(r)^2 = 0\,; \\ g_2(r) \triangleq f_2(1,r) = -0.0482 - 0.0053\,r + 0.9988\,(r)^2 = 0\,. \end{cases}$$

The equation obtained by setting the derivative $J'(r)$ of $J(r)$ to zero can now be computed from (4.5.18) and is given by:

$$-0.2113 + 0.8592r - 0.0313(r)^2 + 1.9954(r)^3 = 0\,.$$

The roots of this equation are found to be $-0.1033 + 0.6825\jmath$, $-0.1033 - 0.6825\jmath$, and 0.2222. It can easily be verified that 0.2222 is the only correct root; see also the comments below (4.5.18) on page 193. This root correctly identifies the columns of $\mathbf{A} = \left[\begin{smallmatrix} 0.9 & 1.8 \\ 0.2 & 0.4 \end{smallmatrix}\right]$ because $\mathbf{a}^1 = 0.9\left[\begin{smallmatrix} 1 \\ 0.2222 \end{smallmatrix}\right]$ and $\mathbf{a}^2 = 1.8\left[\begin{smallmatrix} 1 \\ 0.2222 \end{smallmatrix}\right]$.

We now perform the same procedure as described above for the estimated case. From (4.4.29) and (4.5.18) it follows that now the equation to be solved is:

$$-0.2149 + 0.8473r - 0.0353(r)^2 + 1.9943(r)^3 = 0\,,$$

whose roots are are found to be $-0.1051 + 0.6795\jmath$, $-0.1051 - 0.6795\jmath$, and 0.2279. Now, 0.2279 is the only correct root. The true and estimated normalized vectors of interest are:

$$\frac{\mathbf{a}^{1\prime}}{\|\mathbf{a}^{1\prime}\|} = \frac{\mathbf{a}^{2\prime}}{\|\mathbf{a}^{2\prime}\|} = \begin{bmatrix} 0.9762 \\ 0.2169 \end{bmatrix}, \qquad \frac{\hat{\mathbf{z}}'}{\|\hat{\mathbf{z}}'\|} = \begin{bmatrix} 0.9750 \\ 0.2222 \end{bmatrix}. \tag{4.5.24}$$

See also Fig. 4.13 on page 185 and Fig. 4.23 on page 209.

4.5.4.3 Application to Example 3 in Section 4.4.4

In this section we briefly present the estimation results for the scenario with two mixed speech signals that was introduced in Section 1.1.2 of Chapter 1 and discussed further in Section 4.4.4. The estimation procedure is exactly the same as in Section 4.5.4.1 because again there is only one equation to be solved. The estimate of the mixing matrix with normalized columns found by means of the technique developed in Section 4.5.3 is given by:

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.7115 & 0.7316 \\ 0.7027 & -0.6818 \end{bmatrix}.$$

Note that the ideal mixing matrix rounded to four decimals follows from (1.1.4) as:

$$\mathbf{A} = \begin{bmatrix} -0.7071 & 0.7071 \\ 0.7071 & 0.7071 \end{bmatrix}.$$

The source signals can now be estimated by computing $\mathbf{y}[n] = \hat{\mathbf{A}}^{-1}\,\mathbf{x}[n]$. Fig. 4.18 on the facing page shows 8000 samples of the original speech signals signals $s_1[n]$ and $s_2[n]$ at the left side (see also the left side of Fig. 1.5 on page 8), and the estimated speech signals $y_1[n]$ and $y_2[n]$ at the right side. As a measure of performance, we compute the total transfer matrix from the source signals to the output signals:

$$\mathbf{T} \triangleq \hat{\mathbf{A}}^{-1}\mathbf{A} = \begin{bmatrix} -1.0008 & -0.0063 \\ 0.0352 & 1.0002 \end{bmatrix}.$$

Hence, the contributions of $s_1[n]$ and $s_2[n]$ to $y_1[n]$ equal 99.4% and 0.6% respectively, whereas the contributions of $s_1[n]$ and $s_2[n]$ to $y_2[n]$ equal 3.4% and 96.6% respectively.

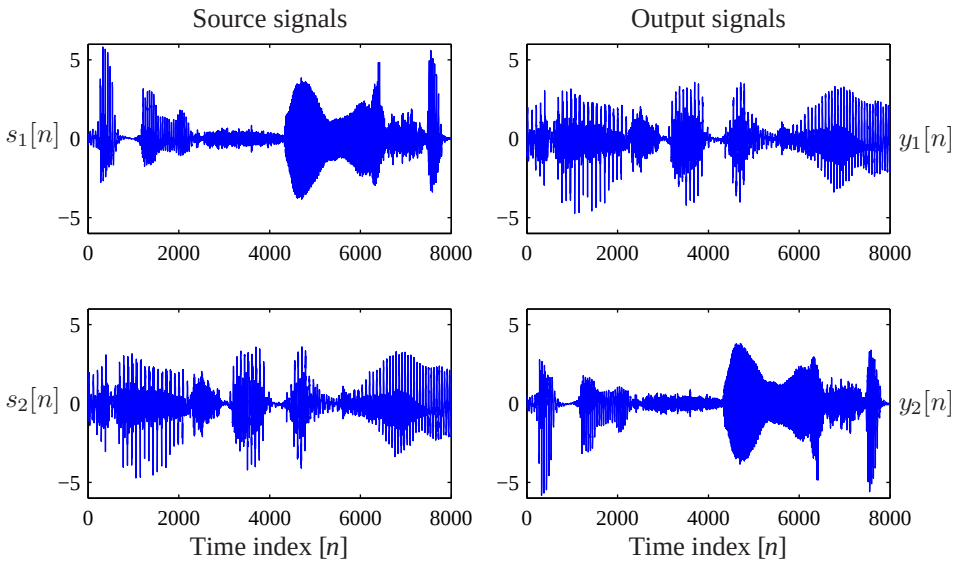

Figure 4.18: *Original speech source signals (left) and estimated speech signals (right) for example with speech signals in Section 4.4.4.*

4.5.4.4 Application to Example 4 signals in Section 4.4.5

In this section we briefly present the estimation results for the scenario with two mixed image signals that was introduced in Section 1.1.2 of Chapter 1 and discussed further in Section 4.4.5. Again, the estimation procedure is exactly the same as in Section 4.5.4.1 because again there is only one equation to be solved. The estimate of the mixing matrix with normalized columns found by means of the technique developed in Section 4.5.3 is given by:

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.5936 & 0.7896 \\ 0.8047 & 0.6136 \end{bmatrix} .$$

Note that the ideal mixing matrix is given by (4.4.32) as $\mathbf{A} = \left[\begin{smallmatrix} 0.9 & 0.6 \\ 0.6 & 0.8 \end{smallmatrix}\right]$. The source signals can now be estimated by computing $\mathbf{y}[u,v] = \hat{\mathbf{A}}^{-1}\,\mathbf{x}[u,v]$. Fig. 4.19 on the following page shows the two source images $s_1[u,v]$ and $s_2[u,v]$ at the left side (see also the left side of Fig. 1.6 on page 9) and the estimated images $y_1[u,v]$ and $y_2[u,v]$ at the right side. Note that $y_1[u,v]$ provides an estimate of $s_2[u,v]$, whereas $y_2[u,v]$ provides an estimate of $s_1[u,v]$. As a measure of performance, we compute the total transfer matrix from the source signals to the output signals:

$$\mathbf{T} \triangleq \hat{\mathbf{A}}^{-1}\mathbf{A} = \begin{bmatrix} 0.0629 & 0.9718 \\ 1.0604 & 0.0293 \end{bmatrix} .$$

Hence, the contributions of $s_1[u,v]$ and $s_2[u,v]$ to $y_1[u,v]$ equal 6.1% and 93.9% respectively, whereas the contributions of $s_1[u,v]$ and $s_2[u,v]$ to $y_2[u,v]$ equal 97.3% and 2.7% respectively.

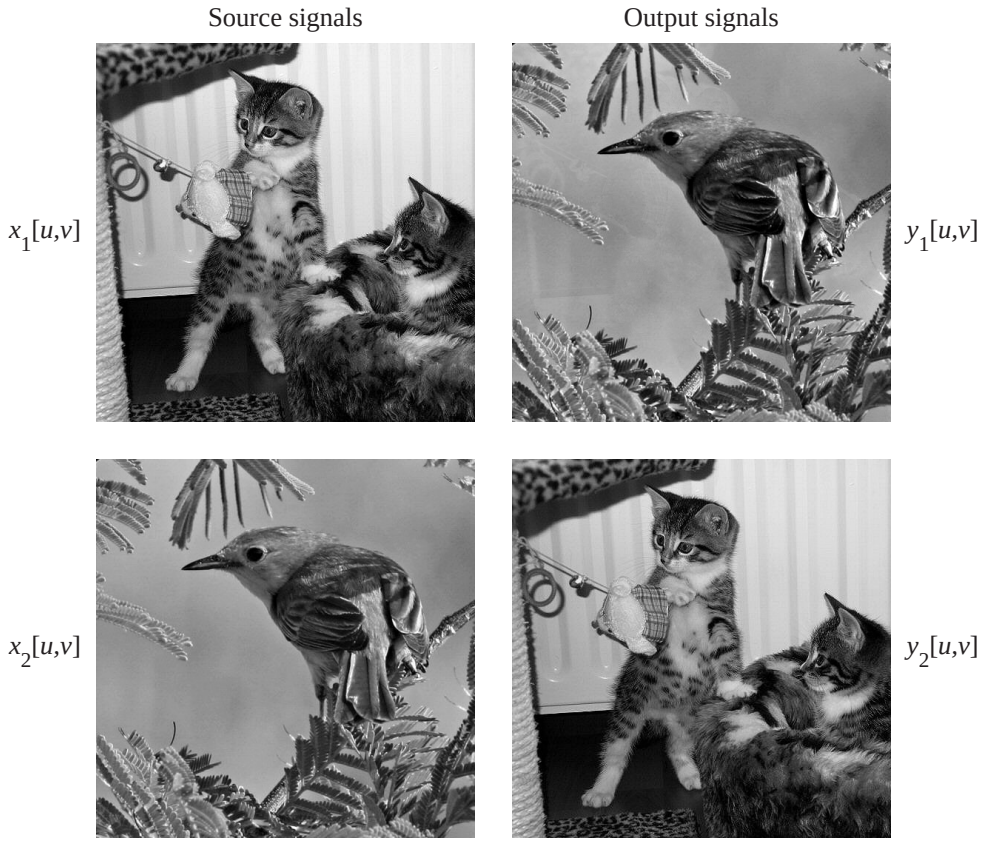

Figure 4.19: *Original source images (left) and separated images (right) for example with image signals in Section 4.4.5.*

4.5.5 Solving a bivariate polynomial equation by factorization

For 'systems' with one equation, which we encounter in case of a full rank mixing matrix, we can also use another simple and direct method for obtaining the solutions. In Section 4.4.1, we have shown how functions of the type in (4.4.1) can be factorized into two linear terms. From this factorization, it followed that the solutions are directly given by (4.4.9). If desired, these solutions can easily be normalized to unit norm. Applying this approach to the examples in Sections 4.4.2, 4.4.4, and 4.4.5 yields exactly the same results as those obtained in Sections 4.5.4.1, 4.5.4.3, and 4.5.4.4 respectively both for the ideal and estimated cases. Note that this is due to fact that all these discussed approaches implicitly use the same algebra.

Whereas the approaches for solving the system discussed so far were mainly algebraic, in the following section we introduce a very powerful method that is truly numerical. As opposed to the algebraic approaches, this method is much more easy to generalize to systems of equations of 'any' complexity. Since the systems that we will encounter in the following chapters are very hard or impossible to solve algebraically, we do need such a method.

4.5.6 Solving a system of bivariate polynomial equations: homotopy

In this section, we introduce the so-called *homotopy* method for solving the system of bivariate homogeneous polynomial equations. Although the currently considered system can also be solved algebraically (as we have seen in the previous sections), we present this homotopy method here because it is our method of choice for solving the more complex systems arising in the following chapters, which are hard or impossible to solve algebraically, and to provide insight. Moreover, it can also deal with non-exact equations and solutions (to a certain extent, of course). Homotopy methods provide a deterministic means for solving a system of nonlinear equations by smoothly deforming the known solutions of a simple start system into the desired solutions of the target system [2, 144]. They are based on the so-called *path following* or *continuation* techniques. In Appendix E, an overview is given of the basic principles and essentials of homotopy methods, along with some elucidating examples. In addition, an extension that we have developed, and which turns out to work well for the type of problems we consider in this work, is described. Excellent discussions about homotopy methods can be found in several articles and books [2, 105, 116, 144]. Since we use the method here for the first time in this thesis, we recapitulate some of its terminology and rationale along with our explanations and derivations. For more details, we refer to Appendix E and in the coming chapters we restrict ourselves to applying the resulting algorithm.

4.5.6.1 Rationale behind homotopy methods

In the context of homotopy methods the system of equations to be solved is commonly referred to as the *target system*. In this thesis we denote a general target system by $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ and assume that $\mathbf{p} : \mathbb{C}_P \rightarrow \mathbb{C}_P$ and its solution set is denoted by $\mathcal{P}$, i.e. $\mathcal{P} \triangleq \{\mathbf{z}_p \in \mathbb{C}_P \mid \mathbf{p}(\mathbf{z}_p) = \mathbf{0}\}$. For this chapter our purpose is to find the unit-norm solutions of systems as in (4.3.33) consisting of bivariate homogeneous polynomial equations of degree two of the form specified in (4.3.31). Hence, the target system we consider here consists of bivariate homogeneous polynomial equations together with the unit-norm constraint. We now first describe homotopy methods for more general systems and in a more general setting. Then, later in this section we will apply the theory to the system to be solved for this chapter. The rationale behind homotopy methods is to deform the known solutions of a simple *start system* into the a priori unknown solutions of the *target system*. The start system should be compatible with the target system as much as possible in the sense that it possesses the same or a similar structure and the same number of solutions. For our current problem, this primarily means that it should also consist of bivariate homogeneous polynomial equations of degree two, preferably with the same number of solutions. In this thesis we denote a general start system by $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, where $\mathbf{g} : \mathbb{C}_P \rightarrow \mathbb{C}_P$, and its solution set by $\mathcal{G}$, i.e. $\mathcal{G} \triangleq \{\mathbf{z}_g \in \mathbb{C}_P \mid \mathbf{g}(\mathbf{z}_g) = \mathbf{0}\}$. The start and target systems are embedded in a *family of systems, the homotopy*, and then all members in the family are solved in a sequential and iterative manner. Hence, a homotopy continuation procedure starts with a start system of equations the solutions of which are known. Then this system is deformed slightly in the 'direction' of the target system. Since this new system is close to the previous system, under some mild conditions its solutions deviate only slightly from those of the previous system. Hence, each solution of the new system can be found by a local iterative optimization method that uses the corresponding solution of the previous system as its initial solution. For each solution of the start system, these steps are repeated until the target system is reached. In theory, under some mild uniqueness and smoothness assumptions each path converges to a geometrically isolated solution [2].

We first embed the start and target systems in a suitable homotopy. Instead of the conventional homotopy (E.1.1) in Appendix E that is commonly described and used in the literature, we have defined our own more natural and intuitive homotopy as follows in (E.1.17):

$$\mathbf{h}(\mathbf{z}, \lambda) = \gamma_g(\lambda - \lambda_e)\, \mathbf{g}(\mathbf{z}) + \gamma_p(\lambda - \lambda_0)\, \mathbf{p}(\mathbf{z}) \qquad \forall\, \lambda \in \mathcal{C}\,, \tag{4.5.25}$$

where $\lambda \in \mathbb{C}$ is called the so-called *continuation parameter* and $\mathcal{C}$ is a certain predefined curve in the complex plane that is to be traversed by λ. The curve $\mathcal{C}$ has a certain starting point $\lambda_0 \in \mathbb{C}$ and a certain end point $\lambda_e \in \mathbb{C}$ such that $\lambda_0 \neq \lambda_e$. Some examples of possible paths that may be traversed in both directions are given in Fig. E.1. The constants $\gamma_g \in \mathbb{C}$ and $\gamma_p \in \mathbb{C}$ are randomly chosen fixed constants whose purpose will be explained soon. For each solution $\mathbf{z}_g$ of the start system $\mathbf{g}(\mathbf{z})$, numerical path following or continuation methods attempt to trace the path/curve $(\mathbf{z}, \lambda) \in \mathbf{h}^{-1}(\mathbf{0})$ defined implicitly by $\mathbf{h}\big(\mathbf{z}, \lambda\big) = \mathbf{0}$ from the starting point $(\mathbf{z}_g, \lambda_0)$, which is a solution of the start system, to a solution point $(\mathbf{z}_p, \lambda_e)$ of the target system where $\mathbf{p}(\mathbf{z}_p) = \mathbf{0}$. Note that $\mathbf{h}(\mathbf{z}, \lambda_0) = \gamma_g(\lambda_0 - \lambda_e)\, \mathbf{g}(\mathbf{z})$ and $\mathbf{h}(\mathbf{z}, \lambda_e) = \gamma_p(\lambda_e - \lambda_0)\, \mathbf{p}(\mathbf{z})$ with $\lambda_0 - \lambda_e \neq 0$. Hence, solving $\mathbf{h}(\mathbf{z}, \lambda_0) = \mathbf{0}$ is equivalent to solving $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, whereas solving $\mathbf{h}(\mathbf{z}, \lambda_e) = \mathbf{0}$ is equivalent to solving $\mathbf{p}(\mathbf{z}) = \mathbf{0}$. We now see that the continuation parameter parameterizes each solution path $(\mathbf{z}(\lambda), \lambda)$ of (4.5.25). In the course of tracing such a path by a homotopy continuation method, λ follows the user-defined curve $\mathcal{C}$ from its starting point λ_0 to its end point λ_e. As is explained in Appendix E and [116, 144], the constants $\gamma_g \in \mathbb{C}$ and $\gamma_p \in \mathbb{C}$ serve to avoid singularities and crossings along the different paths. Hence, the main problem to be solved is to find the curve $(\mathbf{z}(\lambda), \lambda)$ that solves the following system as λ moves from λ_0 to λ_e:

$$\mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big) = \mathbf{0} \qquad \forall\, \lambda \in \mathcal{C}\,. \tag{4.5.26}$$

As we explain in detail in Appendix E, this can be done by deforming the system and finding the solutions of the new system by means of the so-called *predictor* and *corrector* steps that can be derived from (4.5.25) and its derivative w.r.t. λ. This derivative is given by:

$$\frac{d}{d\lambda}\mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big) = \nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda), \lambda)\frac{d\mathbf{z}(\lambda)}{d\lambda} + \partial_\lambda \mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big) = \mathbf{0} \qquad \forall\, \lambda \in \mathcal{C}\,, \tag{4.5.27}$$

where:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \begin{bmatrix} \frac{\partial h_1(\mathbf{z},\lambda)}{\partial z_1} & \cdots & \frac{\partial h_1(\mathbf{z},\lambda)}{\partial z_P} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_P(\mathbf{z},\lambda)}{\partial z_1} & \cdots & \frac{\partial h_P(\mathbf{z},\lambda)}{\partial z_P} \end{bmatrix} \quad \text{and} \quad \partial_\lambda \mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big) \triangleq \frac{\partial \mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big)}{\partial \lambda}$$

are the Jacobian and partial derivative of $\mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big)$ w.r.t. $\mathbf{z}$ and λ respectively. All solutions of the target system can be found by solving (4.5.27) for each of the initial conditions $\mathbf{z}(0) = \mathbf{z}_g \in \mathcal{G}$. Note that (4.5.27) is a system of ordinary differential equations. Hence, given $\mathbf{z}(0)$ solving this system for $\mathbf{z}(\lambda)$ is a standard ODE initial value problem that it equivalent to solving (4.5.26) and in principle can be solved by using any standard software package for numerically solving systems of differential equations with initial values. This would come down to only performing the predictor step referred to above. Although this is possible, in general it is not a good and efficient approach because it can lead to the same accumulation of errors occurring in numerical differential equation solvers and ignores the strong contractive properties which the curve has relative to the corrector steps in view of the fact that it satisfies (4.5.26). It is this very property that leads to the predictor-corrector approach employed by homotopy continuation methods. Given a certain point on the path, consider a single iteration of the method. Suppose that we have a solution $\mathbf{z}(\lambda)$ of $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}$ for a

certain (known) value of λ. Then, this solution is also an approximate solution of the slightly deformed system $\mathbf{h}(\mathbf{z}(\lambda + \Delta\lambda), \lambda + \Delta\lambda) = \mathbf{0}$, where $\Delta\lambda$ is a small increment of λ along the path $\mathcal{C}$. Then, the solution of the deformed system is found by means of the predictor and corrector steps. Firstly, in the *predictor* step an approximation of the solution $\mathbf{z}(\lambda + \Delta\lambda)$ of the deformed system is predicted, e.g. by using an Euler step (see further). Secondly, the predicted approximate solution is corrected to lie on the solution curve by applying a *corrector* step, which can be any local zero finding method such as Newton's method [2]. Performing these two steps for each solution of the start system and each iteration from $\lambda = \lambda_0$ to $\lambda = \lambda_e$, we finally end up at the solutions of the target system. Both the predictor as well as the corrector step can be derived by using (4.5.26) and (4.5.27). In the sequel we will use the Euler predictor and a Newton corrector that we have derived in Appendix E. In concise notation, they can be written as:

$$\mathbf{z} := \mathbf{z} - \Delta\lambda \left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1} \frac{\partial \mathbf{h}}{\partial \lambda} \tag{4.5.28}$$

and:

$$\mathbf{z} := \mathbf{z} - \left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1} \mathbf{h} \tag{4.5.29}$$

respectively, where ':=' denotes the assignment operation. Note from (4.5.25) that:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}, \lambda) = \gamma_g(\lambda - \lambda_e)\, \nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z}) + \gamma_p(\lambda - \lambda_0)\, \nabla_{\mathbf{z}}\mathbf{p}(\mathbf{z}) \,, \tag{4.5.30}$$

and:

$$\partial_\lambda \mathbf{h}(\mathbf{z}, \lambda) = \gamma_g\, \mathbf{g}(\mathbf{z}) + \gamma_p\, \mathbf{p}(\mathbf{z}) \,. \tag{4.5.31}$$

The key ingredients of the resulting homotopy algorithm are summarized in Alg. 4.4. Note that at each continuation iteration of the homotopy algorithm the Euler predictor is evaluated at $(\hat{\mathbf{z}}(\lambda), \lambda)$ (see (E.1.8)), whereas the Newton corrector is evaluated at $(\hat{\mathbf{z}}(\lambda + \Delta\lambda), \lambda + \Delta\lambda)$ (see (E.1.13)). Also notice that at each fixed value of λ we stop applying the corrector steps either if the Newton correction becomes smaller than a certain tolerance $NwtTol$, or when a certain maximum number $MaxNnwtIt$ of Newton corrections has been applied. See Appendix E for details. Useful and common choices for γ_g and γ_p are given by $\gamma_g = \exp(\jmath\theta_g)$ and $\gamma_p = \exp(\jmath\theta_p)$ with θ_g and θ_p picked at random from a uniform distribution on the interval $[0, 2\pi]$ [144].

Now we are in a position to apply the theory described above and in Appendix E to the systems arising in this chapter. Firstly, the target system and its ideal solutions are considered. As we have seen in the previous sections, the target system consists of a system of bivariate homogeneous polynomial equations of degree two together with a unit-norm constraint. Hence, our target system can be written as follows:

$$\begin{cases} \{f_q(z_1, z_2) = 0\}_{q \in \mathcal{Q}} \,; \\ \quad c(z_1, z_2) = 0 \,, \end{cases} \tag{4.5.32}$$

which in expanded form equals:

$$\begin{cases} \{\varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 = 0\}_{q \in \mathcal{Q}} \,; \\ \qquad (z_1)^2 + (z_2)^2 - 1 = 0 \,, \end{cases} \tag{4.5.33}$$

Algorithm 4.4 Homotopy continuation method employed in thesis.

1: Initialize γ_g, γ_p, $MaxNnwtIt$, $NwtTol$ and $NwtNoConvergenceTol$;

2: Construct a start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ that has the same structure and number of solutions as the target system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ (or more), and whose solutions are known;

3: Embed $\mathbf{g}(\mathbf{z})$ and $\mathbf{p}(\mathbf{z})$ in the following convex homotopy system:

$$\mathbf{h}(\mathbf{z}, \lambda) = \gamma_g(\lambda - \lambda_e)\,\mathbf{g}(\mathbf{z}) + \gamma_p(\lambda - \lambda_0)\,\mathbf{p}(\mathbf{z}) \qquad \forall\, \lambda \in \mathcal{C}\,,$$

where the continuation parameter $\lambda \in \mathbb{R}$ ($\mathbb{C}$) follows the predefined curve $\mathcal{C}$ in the complex plane from λ_0 to λ_e with $\lambda_0 \neq \lambda_e$;

4: Compute the solutions of the start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, which are the solutions of the homotopy system for $\lambda = \lambda_0$, and store them in a set $\mathcal{G}$;

5: Choose an ordered sequence $\mathcal{C}_o$ of values for λ lying on $\mathcal{C}$ such that λ progresses sequentially from λ_0 to λ_e in 'sufficiently small' steps;

6: For each solution $\mathbf{z}_g \in \mathcal{G}$ of $\mathbf{h}(\mathbf{z}, \lambda_0) = \mathbf{0} \equiv \mathbf{g}(\mathbf{z}) = \mathbf{0}$, trace the curve $(\mathbf{z}(\lambda), \lambda) \in \mathbf{h}^{-1}(\mathbf{0})$ defined implicitly by $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}$ from the starting point $(\mathbf{z}_g, \lambda_0)$ to a solution point $(\mathbf{z}_p, \lambda_e)$ of $\mathbf{h}(\mathbf{z}, \lambda_e) = \mathbf{0} \equiv \mathbf{p}(\mathbf{z}) = \mathbf{0}$ by the following continuation method:

for all** $\mathbf{z}_g \in \mathcal{G}$ **do
  $\mathbf{z} := \mathbf{z}_g$;
  for** $\lambda = \lambda_0 \rightarrow \lambda_e$ along $\mathcal{C}_o$ **do
    % Euler predictor step:
    $\frac{d\mathbf{z}}{d\lambda} = -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}\partial_\lambda \mathbf{h}$;
    $\mathbf{z} := \mathbf{z} + \Delta\lambda\,\frac{d\mathbf{z}}{d\lambda}$;
    % Newton corrector steps:
    $\lambda :=$ *next* λ *from* $\mathcal{C}_o$;
    for** $m = 1 \rightarrow MaxNnwtIt$ **do
      $NwtCorr := -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}\mathbf{h}$;
      $\mathbf{z} := \mathbf{z} + NwtCorr$;
      if** $norm(NwtCorr) < NwtTol$ **then
        Leave corrector loop;
      end if
    end for
    if** $norm(NwtCorr) > NwtNoConvergenceTol$ **then
      Print warning or break because Newton corrector did not converge;
    end if
  end for
  Store solution $\mathbf{z}_p = \mathbf{z}$ in set $\mathcal{P}$;
end for
Return $\mathcal{P}$;

or, more elaborately:

$$\begin{cases} p_1(z_1,z_2) &=& f_1(z_1,z_2) &=& \varphi_1^{11}z_1z_1+\varphi_1^{12}z_1z_2+\varphi_1^{22}z_2z_2 &=& 0\,; \\ \vdots & & \vdots & & \vdots & & \vdots \\ p_Q(z_1,z_2) &=& f_Q(z_1,z_2) &=& \varphi_Q^{11}z_1z_1+\varphi_Q^{12}z_1z_2+\varphi_Q^{22}z_2z_2 &=& 0\,; \\ p_{Q+1}(z_1,z_2) &=& c(z_1,z_2) &=& (z_1)^2+(z_2)^2-1 &=& 0\,, \end{cases} \tag{4.5.34}$$

where $c(z_1,z_2) \triangleq (z_1)^2+(z_2)^2-1$ represents the function that equals zero if the unit-norm constraint is satisfied. In matrix-vector notation, this system can be written as $\mathbf{p}(\mathbf{z})=\mathbf{0}$ with $\mathbf{p}:\mathbb{C}_P \rightarrow \mathbb{C}_P$ given by:

$$\mathbf{p}(\mathbf{z}) \triangleq \begin{bmatrix} \mathbf{f}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix}, \tag{4.5.35}$$

where $P=Q+1$. For simplicity, we will first apply homotopy Alg. 4.4 to a target system with one bivariate polynomial equation together with the unit-norm constraint; see Section E.3 for an elaborate example. After that, we will extend the results to systems with two bivariate polynomial equations together with the unit-norm constraint.

4.5.6.2 Application to solving a single bivariate homogeneous polynomial equation subject to unit-norm constraint

As we have shown in earlier sections, following the approach developed in this chapter we obtain a system with one bivariate polynomial equation when the mixing matrix is full rank; see Section 4.4.2, for example. In this case, target system (4.5.34) becomes:

$$\mathbf{p}(\mathbf{z}) = \begin{bmatrix} p_1(z_1,z_2) \\ p_2(z_1,z_2) \end{bmatrix} \triangleq \begin{bmatrix} f_q(z_1,z_2) \\ c(z_1,z_2) \end{bmatrix} = \begin{bmatrix} \varphi_q^{11}z_1z_1+\varphi_q^{12}z_1z_2+\varphi_q^{22}z_2z_2 \\ (z_1)^2+(z_2)^2-1 \end{bmatrix}. \tag{4.5.36}$$

Although there is only one homogeneous equation in this system, we retain the index q for easy extension to more general scenarios later on. As we have explained before, solving the system comes down to finding the intersections between the two lines representing the zero contour level of $f_q(z_1,z_2)$ and the unit circle representing the zero contour level of $c(z_1,z_2)$, see Fig. 4.16 on page 189. There are four intersection points (indicated by the black stars), two of which are essentially different because points that are opposite w.r.t. the origin are equivalent according to equivalence relation (4.5.4). As we have seen in Sections 4.4.1 and 4.5.5, the ideal solutions of first equation in (4.5.36) are given by (4.4.9). Two solutions of (4.5.36) that are different according to (4.5.4) are obtained by choosing η_1 and η_2 in (4.4.9) such that the solutions have unit Euclidian norm.

We now compute the quantities needed for applying Alg. 4.4. The Jacobian of $\mathbf{p}(\mathbf{z})$ in (4.5.36) is given by:

$$\nabla_{\mathbf{z}}\mathbf{p}(\mathbf{z}) = \begin{bmatrix} \frac{\partial p_1(z_1,z_2)}{\partial z_1} & \frac{\partial p_1(z_1,z_2)}{\partial z_2} \\ \frac{\partial p_2(z_1,z_2)}{\partial z_1} & \frac{\partial p_2(z_1,z_2)}{\partial z_2} \end{bmatrix} = \begin{bmatrix} 2\varphi_q^{11}z_1+\varphi_q^{12}z_2 & \varphi_q^{12}z_1+2\varphi_q^{22}z_2 \\ 2z_1 & 2z_2 \end{bmatrix}. \tag{4.5.37}$$

Now that we have discussed the target system, we define and consider an appropriate start system $\mathbf{g}(\mathbf{z})=\mathbf{0}$ with $\mathbf{g}:\mathbb{C}_2 \rightarrow \mathbb{C}_2$. As we have mentioned above and explained in Appendix E, the start and target system should be compatible as much as possible in the sense that they possess the same or a similar structure and the same number of solutions. For our current problem, this means that the start system should consist of equations that have the same structure as those in (4.5.36), and possess the same number of essentially different

solutions. From (4.5.36) and (4.4.8) it follows that a natural and intuitive choice for the start system function vector is given by:

$$\mathbf{g}(\mathbf{z}) = \begin{bmatrix} g_1(z_1, z_2) \\ g_2(z_1, z_2) \end{bmatrix} \triangleq \begin{bmatrix} (\alpha z_1 - z_2)(z_1 - \beta z_2) \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} = \begin{bmatrix} \alpha z_1 z_1 - (1 + \alpha\beta) z_1 z_2 + \beta z_2 z_2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix}, \tag{4.5.38}$$

where α and β are randomly chosen fixed complex constants that we choose in the same way as we choose the 'γ-constants', i.e. $\alpha = \exp(\jmath\theta_\alpha)$ and $\beta = \exp(\jmath\theta_\beta)$ with θ_α and θ_β picked at random from the interval $[0, 2\pi]$. Using simple algebra (see also Section 4.4.1), it can be shown that the two essentially different solutions of the start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ can be represented by the elements of the following set:

$$\mathcal{G} = \left\{ \frac{1}{\sqrt{1 + |\alpha|^2}} \begin{bmatrix} 1 \\ \alpha \end{bmatrix}, \frac{1}{\sqrt{1 + |\beta|^2}} \begin{bmatrix} \beta \\ 1 \end{bmatrix} \right\}. \tag{4.5.39}$$

The Jacobian of $\mathbf{g}(\mathbf{z})$ is given by:

$$\nabla_{\mathbf{z}} \mathbf{g}(\mathbf{z}) = \begin{bmatrix} \frac{\partial g_1(z_1,z_2)}{\partial z_1} & \frac{\partial g_1(z_1,z_2)}{\partial z_2} \\ \frac{\partial g_2(z_1,z_2)}{\partial z_1} & \frac{\partial g_2(z_1,z_2)}{\partial z_2} \end{bmatrix} = \begin{bmatrix} 2\alpha z_1 - (1 + \alpha\beta) z_2 & -(1 + \alpha\beta) z_1 + 2\beta z_2 \\ 2z_1 & 2z_2 \end{bmatrix}. \tag{4.5.40}$$

Embedding the start and target function vectors in homotopy (4.5.25) yields:

$$\begin{aligned} \mathbf{h}(\mathbf{z}, \lambda) = {} & \gamma_g (\lambda - \lambda_e) \begin{bmatrix} (\alpha z_1 - z_2)(z_1 - \beta z_2) \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} \\ & + \gamma_p (\lambda - \lambda_0) \begin{bmatrix} \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} \qquad \forall\, \lambda \in \mathcal{C}\,. \end{aligned} \tag{4.5.41}$$

To compute the Euler predictor and Newton corrector steps in Alg. 4.4, we need to compute the quantities $\nabla_{\mathbf{z}} \mathbf{h}(\mathbf{z}, \lambda)$ and $\partial_\lambda \mathbf{h}(\mathbf{z}, \lambda)$. According to (4.5.30), (4.5.36), and (4.5.38), the first is given by:

$$\begin{aligned} \nabla_{\mathbf{z}} \mathbf{h}(\mathbf{z}, \lambda) = {} & \gamma_g (\lambda - \lambda_e) \begin{bmatrix} 2\alpha z_1 - (1 + \alpha\beta) z_2 & -(1 + \alpha\beta) z_1 + 2\beta z_2 \\ 2z_1 & 2z_2 \end{bmatrix} \\ & + \gamma_p (\lambda - \lambda_0) \begin{bmatrix} 2\varphi_q^{11} z_1 + \varphi_q^{12} z_2 & \varphi_q^{12} z_1 + 2\varphi_q^{22} z_2 \\ 2z_1 & 2z_2 \end{bmatrix} \qquad \forall\, \lambda \in \mathcal{C}\,, \end{aligned} \tag{4.5.42}$$

and according to (4.5.31), (4.5.40), and (4.5.36) the second is given by:

$$\partial_\lambda \mathbf{h}(\mathbf{z}, \lambda) = \gamma_g \begin{bmatrix} (\alpha z_1 - z_2)(z_1 - \beta z_2) \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} + \gamma_p \begin{bmatrix} \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{22} z_2 z_2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} \quad \forall\, \lambda \in \mathcal{C}\,. \tag{4.5.43}$$

We now have all ingredients required by Alg. 4.4. The parameter values are chosen as $MaxNnwtIt = 4$, $NwtTol = 10^{-8}$ and $NwtNoConvergenceTol = 10^{-2}$. Furthermore, picking γ_g, γ_p and β_1, β_2 as described in the previous sections yielded $\gamma_g = 0.9513 - 0.3082\jmath$, $\gamma_p = 0.1182 + 0.9930\jmath$, and $\alpha = -0.7830 - 0.6220\jmath$, $\beta = -0.9961 + 0.0880\jmath$. As the discretized curve $\mathcal{C}_o$ in the complex plane to be traversed by the continuation parameter λ, we choose the straight line between the start point $\lambda_0 = 0$ and the end point $\lambda_e = \frac{1}{4}(1 + \jmath)$ with constant step size $\Delta\lambda = \frac{1}{400}(1 + \jmath)$ (consisting of 101 points), see the dashed curve in Fig. E.1. Now we apply the algorithm to Example 1 in Section 4.4.2. Firstly, we compute the unit-norm solutions of $f(z_1, z_2)$ in (4.4.13), which

was derived from the ideal subspace matrix (see Section 4.4.2.1). After that, we compute the unit-norm solutions of $\hat{f}(z_1, z_2)$ in (4.4.20), which was derived from the estimated subspace matrix (see Section 4.4.2.2). The results are visualized in Figures 4.20 and 4.21 as follows. For each of the two solution paths of the homotopy system in (4.5.25), the real and imaginary parts of $z_1(|\lambda|)$ and $z_2(|\lambda|)$ are depicted as a function of the absolute value of the continuation parameter λ in the interval $[|\lambda_0|, |\lambda_e|] = [0, \frac{1}{4}\sqrt{2}]$. The z_1-components of the solution paths are represented by the solid lines, whereas the z_2-components are represented by the dotted lines. A pair of lines consisting of a solid and a dotted line of the same color or gray shade constitutes one solution path. The z_1-components of the beginnings of the paths, i.e. the z_1-components of the solutions of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, are denoted by the crosses '$\times$', whereas the corresponding z_2-components are denoted by the diamonds '$\diamond$'. Likewise, the z_1-components of the endpoints of the paths, i.e. the z_1-components of the solutions of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$, are denoted by the stars '$*$', whereas the corresponding z_2-components are denoted by the dots '$\cdot$'.For reference, we have drawn the unit circle in the $z_1 - z_2$ plane for $|\lambda| = |\lambda_0| = 0$, and the z_1- and z_2-axes in the $z_1 - z_2$ plane for $|\lambda| = |\lambda_e| = \frac{1}{4}\sqrt{2}$.

Example 1 in Section 4.4.2 with ideal subspace matrix
Running Alg. 4.4 with the values of $\varphi^{11}, \varphi^{12}$ and φ^{22} given by (4.4.12) substituted into the first equation $p_1(z_1, z_2)$ of (4.5.36) gives the following solution set of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ with four decimals accuracy:

$$\mathcal{P} = \left\{ \begin{bmatrix} 0.9762 \\ 0.2169 \end{bmatrix}, \begin{bmatrix} 0.1644 \\ 0.9864 \end{bmatrix} \right\}.$$

As can be verified by comparing this set to the normalized columns of $\mathbf{A}$, (see (4.5.20), for example), all roots have been found correctly. The solution paths are depicted in Fig. 4.20 on the next page, which shows that the solution paths run smoothly from those of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ to those of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ without going through the same point.

Example 1 in Section 4.4.2 with estimated subspace matrix
Running Alg. 4.4 with the estimated values $\hat{\varphi}^{11}, \hat{\varphi}^{12}$ and $\hat{\varphi}^{22}$ given by (4.4.19) substituted into the first equation $p_1(z_1, z_2)$ of (4.5.36) gives the following solution set of $\hat{\mathbf{p}}(\mathbf{z}) = \mathbf{0}$ with four decimals accuracy:

$$\hat{\mathcal{P}} = \left\{ \begin{bmatrix} 0.9753 \\ 0.2208 \end{bmatrix}, \begin{bmatrix} 0.0731 \\ 0.9973 \end{bmatrix} \right\}.$$

These solutions are exactly the same as those obtained in Section 4.5.4.1. The solution paths are depicted in Fig. 4.21 on the following page, which shows that the solution paths run smoothly from those of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ to those of $\hat{\mathbf{p}}(\mathbf{z}) = \mathbf{0}$ without going through the same point.

Applying the same or a similar approach to the examples in Sections 4.4.3, 4.4.4, and 4.4.5 yields the same results as those obtained in Sections 4.5.4.2, 4.5.4.3, and 4.5.4.4 respectively both for the ideal and estimated cases.

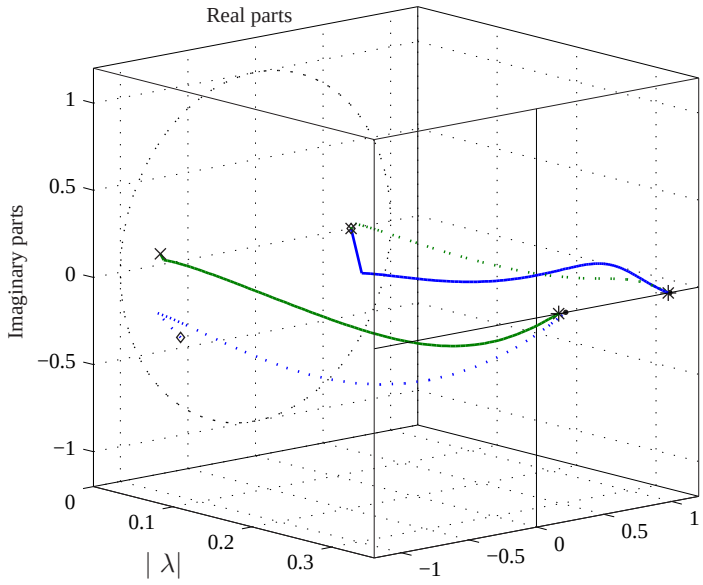

Figure 4.20: *Real and imaginary parts of $z_1(|\lambda|)$-components (solid lines) and $z_2(|\lambda|)$-components (dotted lines) of homotopy solution paths of $\mathbf{h}(\mathbf{z}, \lambda) = \mathbf{0}$ for Example 1 in Section 4.4.2 with ideal subspace matrix.*

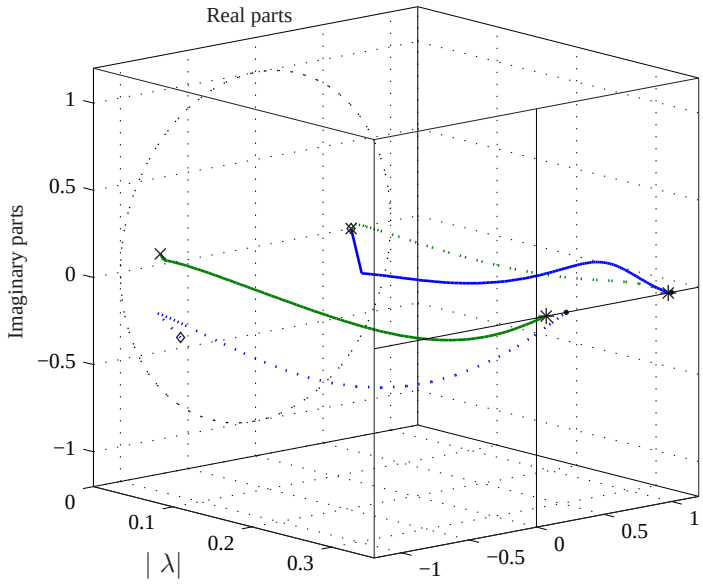

Figure 4.21: *Real and imaginary parts of $z_1(|\lambda|)$-components (solid lines) and $z_2(|\lambda|)$-components (dotted lines) of homotopy solution paths of $\mathbf{h}(\mathbf{z}, \lambda) = \mathbf{0}$ for Example 1 in Section 4.4.2 with estimated subspace matrix.*

4.5.7 TIME-MUSIC Pseudo-Spectrum

Finally, in this section we develop an idea that is similar to that employed by the conventional MUSIC algorithm explained in Chapter 3. The j-th column $\mathbf{a}^j$ of $\mathbf{A}$ can be written as:

$$\mathbf{a}^j = \left\|\mathbf{a}^j\right\| \begin{bmatrix}\cos(\theta^j)\\ \sin(\theta^j)\end{bmatrix}, \quad \text{where} \quad \theta^j = \arctan\left(\frac{a_2^j}{a_1^j}\right), \quad j = 1, 2\,.$$

Since the norm of the columns is irrelevant, the columns can be parameterized by a single parameter θ, which is the angle measured counterclockwise w.r.t. the positive 'first-coordinate' axis (e.g. z_1 in Fig. 4.6). Likewise, the vector variable $\mathbf{z}$ in system (4.3.33) is parameterized by the single parameter θ as follows (assuming that $\|\mathbf{z}\| = 1$):

$$\mathbf{z}(\theta) = \begin{bmatrix}z_1(\theta)\\ z_2(\theta)\end{bmatrix} = \begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix} \qquad \forall\, \theta \in \Theta\,. \tag{4.5.44}$$

We can consider the angle θ as the analogue of the Direction Of Arrival (DOA) in narrowband array signal processing (see Section 1.1.3 and Chapter 3) for real-valued mixing systems. Due to (4.5.4), the vectors $\mathbf{z}(\theta)$ and $\mathbf{z}(\theta+\pi) = -\mathbf{z}(\theta)$ are equivalent. Therefore, the range of θ is restricted to an interval of length π, e.g. $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$ as in Chapter 3, and the computation of the angle θ^j of $\mathbf{a}^j$ is done modulo π, i.e. $\theta^j = \arctan\left(\frac{a_2^j}{a_1^j}\right) \mod \pi$. With abuse of notation, the functions $\{f_q(\mathbf{z})\}_{q\in\mathcal{Q}}$ in system (4.3.33) can now be written as functions of θ, i.e. $f_q(\theta) \triangleq f_q(\mathbf{z}(\theta))$. For all $q \in \mathcal{Q}$, the function $f_q(\theta)$ is given by:

$$\begin{aligned} f_q(\theta) \triangleq f_q(\mathbf{z}(\theta)) &= \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} \varphi_q^{i_1 i_2} z_{i_1}(\theta) z_{i_2}(\theta) \\ &= \varphi_q^{11} z_1(\theta) z_1(\theta) + \varphi_q^{12} z_1(\theta) z_2(\theta) + \varphi_q^{22} z_2(\theta) z_2(\theta) \\ &= \varphi_q^{11} (\cos(\theta))^2 + \varphi_q^{12} \cos(\theta)\sin(\theta) + \varphi_q^{22} (\sin(\theta))^2 \qquad \forall\, \theta \in \Theta\,. \end{aligned} \tag{4.5.45}$$

From (4.3.32) it follows that (note the similarity to MUSIC):

$$f_q(\theta^j) = 0 \qquad \forall\, q \in \mathcal{Q}, \quad j = 1, 2\,. \tag{4.5.46}$$

Hence, the DOA's can be estimated by locating the zeros of the following system:

$$\{f_q(\theta) = 0\}_{q\in\mathcal{Q}}\,. \tag{4.5.47}$$

This is a one-dimensional search problem that can be solved using existing mathematical tools. However, this issue is not pursued here, but similarly to the conventional MUSIC algorithm [137], a pseudo-spectrum, called the Second Order TIME-MUSIC Pseudo-Spectrum, is defined by $P(\theta) \triangleq \left(\sum_{q\in\mathcal{Q}} |f_q(\theta)|^2\right)^{-1}$ or, in expanded form:

$$P(\theta) \triangleq \frac{1}{\sum\limits_{q\in\mathcal{Q}} \left|\varphi_q^{11}(\cos(\theta))^2 + \varphi_q^{12}\cos(\theta)\sin(\theta) + \varphi_q^{22}(\sin(\theta))^2\right|^2} \quad \forall\, \theta \in \Theta\,. \tag{4.5.48}$$

This spectrum is periodic with period π and thus again it is sufficient to consider it only for an interval of length π in the angular domain. Similarly to the conventional MUSIC spectrum, $P(\theta)$ exhibits sharp peaks at $\theta = \theta^1$ and $\theta = \theta^2$, where θ^1 and θ^2 are the true DOA's corresponding to the first and second columns of $\mathbf{A}$ respectively. As in conventional MUSIC, the DOA's can be estimated by locating the peaks of the pseudo-spectrum. The

above procedure is called TIME-MUSIC because it exploits the temporal (time) structure in the data and due to its resemblance with the conventional MUSIC algorithm.

By using the definitions in (4.3.106) and (4.3.105) with abuse of notation as follows:

$$\mathbf{f}(\theta) \triangleq \begin{bmatrix} f_1(\theta) \\ \vdots \\ f_q(\theta) \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{z}(\theta)) \\ \vdots \\ f_q(\mathbf{z}(\theta)) \end{bmatrix}, \ \mathbf{w}_{\mathrm{u}}^2(\theta) \triangleq \begin{bmatrix} z_1(\theta)z_1(\theta) \\ z_1(\theta)z_2(\theta) \\ z_2(\theta)z_2(\theta) \end{bmatrix} = \begin{bmatrix} (\cos(\theta))^2 \\ \cos(\theta)\sin(\theta) \\ (\sin(\theta))^2 \end{bmatrix} \quad \forall\, \theta \in \Theta, \tag{4.5.49}$$

system (4.5.47) can be written in matrix-vector notation as follows:

$$\mathbf{f}(\theta) = \mathbf{\Phi}\mathbf{w}_{\mathrm{u}}^2(\theta) = \mathbf{0}\,. \tag{4.5.50}$$

Note that (4.5.46) states that $\mathbf{f}(\theta^j) = \mathbf{\Phi}\,\mathbf{w}_{\mathrm{u}}^2(\theta^j) = \mathbf{0}$ for $j = 1, 2$, and the pseudo-spectrum can be written as follows:

$$\boxed{P(\theta) \triangleq \frac{1}{\mathbf{f}^T(\theta)\mathbf{f}(\theta)} = \frac{1}{\left(\mathbf{w}_{\mathrm{u}}^2(\theta)\right)^T \mathbf{\Phi}^T \mathbf{\Phi}\, \mathbf{w}_{\mathrm{u}}^2(\theta)} \qquad \forall\, \theta \in \Theta\,.} \tag{4.5.51}$$

Now, we will compute and plot the pseudo-spectra for the scenarios used in the examples of Sections 4.4.2 and 4.4.3 respectively. The TIME-MUSIC pseudo-spectra are obtained for both the ideal and estimated subspace matrices by evaluating (4.5.51) with $\mathbf{\Phi}$ and $\hat{\mathbf{\Phi}}$ respectively on a fine grid in the parameter space of interest, which is chosen as $\Theta \triangleq [-\frac{\pi}{2}, \frac{\pi}{2}]$. The resulting spectra expressed in decibels [dB] are plotted as a function of the DOA θ expressed in degrees with the ideal and estimated spectra represented by solid and dotted lines respectively. The true source DOA's are indicated by dashed vertical lines.

4.5.7.1 Application to Example 1 in Section 4.4.2

The true DOA's corresponding to the first and second column of $\mathbf{A}$ defined in (4.4.10) are given by $\theta^1 = 12.5$ and $\theta^2 = 80.5$ degrees respectively. The TIME-MUSIC pseudo-spectra obtained by using $\mathbf{\Phi} = \tilde{\boldsymbol{\varphi}}$ in (4.4.12) and $\hat{\mathbf{\Phi}} = \hat{\tilde{\boldsymbol{\varphi}}}$ in (4.4.19) are depicted in Fig. 4.22. The figure shows that both the ideal and estimated spectra exhibit sharp peaks at respectively near the true source DOA's, and that no spurious peaks are present. For a detailed judgement of results like those, see the discussions in Chapter 3 concerning pseudo-spectra.

4.5.7.2 Application to Example 2 in Section 4.4.3

The true DOA, which now is the same for both columns of $\mathbf{A}$ defined in (4.4.21) equals $\theta^1 = \theta^2 = 12.5$ degrees. The TIME-MUSIC pseudo-spectra obtained by using $\mathbf{\Phi}$ in (4.4.23) and $\hat{\mathbf{\Phi}}$ in (4.4.28) are depicted in Fig. 4.23. The figure shows that both the ideal and estimated spectra exhibit a sharp peak at respectively near the true source DOA, and that no spurious peaks are present. Again, for a detailed judgement of results like those, see the discussions in Chapter 3 concerning pseudo-spectra.

Evidently, for the examples in Sections 4.4.4 and 4.4.5 similar results are obtained.

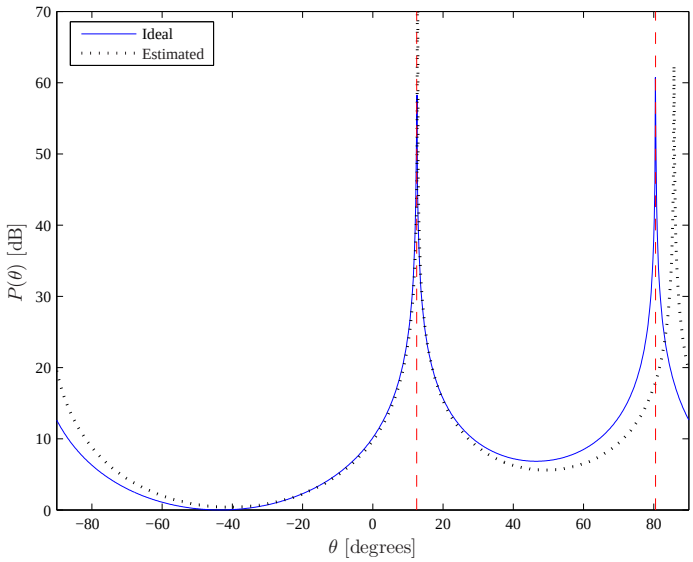

Figure 4.22: *Pseudo-spectrum $P(\theta)$ for Example 1 in Section 4.4.2.*

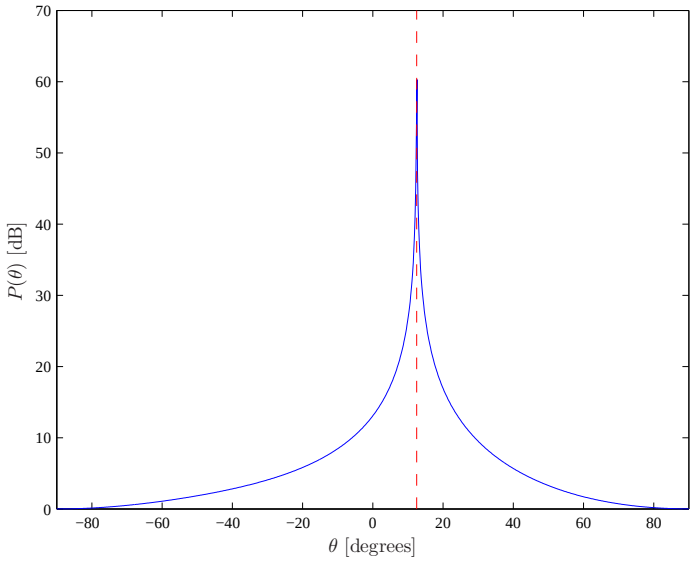

Figure 4.23: *Pseudo-spectrum $P(\theta)$ for Example 2 in Section 4.4.3.*

4.6 Writing MIBI as a GEVD problem

In this section, the MIBI problem will be 'projected onto' a *Generalized Eigenvalue Decomposition (GEVD)* problem. We will show that this formulation is complementary to the previous problem description in terms of a system of homogeneous polynomial equations. In fact, this can already be seen right from the beginning because the derivation of system (4.3.33) was primarily based on (4.3.100), whereas the derivation we are about to present is primarily based on the complementary relation (4.3.99). More precisely, under the assumption that the source auto-correlation matrix has full rank, system (4.3.33) was derived directly from the fact that the left null space of the (uniquiefied) subspace matrix equals the left null space of the (uniquiefied) second order Khatri-Rao product of the mixing matrix, which was proven in Section 4.3.6.6. Likewise, in the sequel of this section we will derive the complementary GEVD problem from the fact that the column range of the (uniquiefied) subspace matrix equals the column range of the (uniquiefied) second order Khatri-Rao product of the mixing matrix, which was also proven in Section 4.3.6.6.

To start with, from (4.3.99) it follows directly that:

$$\check{\mathbf{a}}_{\mathrm{u}}^{j,2} = \mathbf{w}_{\mathrm{u}}^{2}(\mathbf{a}^{j}) \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u}}^{x}[k]\big) \overset{(4.3.134)}{=} \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{s}\right) , \tag{4.6.1}$$

where $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$ and $\mathbf{w}_{\mathrm{u}}^{2}(\cdot)$ are defined in (4.3.74) and (4.3.105) respectively. Hence, each $\check{\mathbf{a}}_{\mathrm{u}}^{j,2}$ can be written as a linear combination of the columns of $\mathbf{U}^{s}$:

$$\check{\mathbf{a}}_{\mathrm{u}}^{j,2} = \mathbf{w}_{\mathrm{u}}^{2}(\mathbf{a}^{j}) = \sum_{p=1}^{d_2^{x,2}} \mu_p^j \, \mathbf{u}^p = \mathbf{U}^s \, \boldsymbol{\mu}^j, \qquad j = 1, 2\,, \tag{4.6.2}$$

where $\boldsymbol{\mu}^j \triangleq \begin{bmatrix} \mu_1^j & \cdots & \mu_{d_2^{x,2}}^j \end{bmatrix}^T$ with $\mu_p^j \in \mathbb{R}$ for $1 \leq p \leq d_2^{x,2}$ and $j = 1, 2$. Thus, the MIBI problem can be projected onto the problem of finding the solutions for the unknown vector $\mathbf{z}$ that satisfy the following equation:

$$\boxed{\check{\mathbf{z}}_{\mathrm{u}}^{2} = \mathbf{w}_{\mathrm{u}}^{2}(\mathbf{z}) = \sum_{p=1}^{d_2^{x,2}} \mu_p \, \mathbf{u}^p = \mathbf{U}^s \, \boldsymbol{\mu}\,,} \tag{4.6.3}$$

where to each solution $\mathbf{z}$ there corresponds a different vector $\boldsymbol{\mu}$ of coefficients. The latter two equations say that the problem of finding the array response vectors is equivalent to finding vectors $\mathbf{z}$ such that the structured vectors $\mathbf{w}_{\mathrm{u}}^{2}(\mathbf{z}) = \check{\mathbf{z}}_{\mathrm{u}}^{2}$ belong to the column range $\mathcal{R}_{\mathrm{c}}\left(\mathbf{A}_{\mathrm{u},\diamond}^{2}\right)$ of $\mathbf{A}_{\mathrm{u},\diamond}^{2}$, which can be computed from the known matrix $\mathbf{C}_{\mathrm{u}}^{x}$ because $\mathcal{R}_{\mathrm{c}}\left(\mathbf{A}_{\mathrm{u},\diamond}^{2}\right)$ equals $\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u}}^{x}\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^{s}\right)$. In order to find the solutions for $\mathbf{z}$, or equivalently the solutions for $\boldsymbol{\mu}$, we exploit the specific structure of $\check{\mathbf{z}}_{\mathrm{u}}^{2} = \mathbf{w}_{\mathrm{u}}^{2}(\mathbf{z})$ given by (4.3.105). Firstly, (4.6.3) is written in terms of the elements of $\mathbf{z}$ as follows:

$$\begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_2 z_2 \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_1^p \\ u_2^p \\ u_3^p \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_{11}^p \\ u_{12}^p \\ u_{22}^p \end{bmatrix} , \tag{4.6.4}$$

where the mapping defined in (4.3.110) has been used. Note that each product of the form $z_{i_1} z_{i_2}$ can be expressed as:

$$z_{i_1} z_{i_2} = \sum_{p=1}^{d_2^{x,2}} \mu_p \, u_{i_1 i_2}^p \, .$$

From (4.6.4) the following systems of equations follow directly:

$$z_1 \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_1^p \\ u_2^p \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_{11}^p \\ u_{12}^p \end{bmatrix}, \quad z_2 \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_2^p \\ u_3^p \end{bmatrix} = \sum_{p=1}^{d_2^{x,2}} \mu_p \begin{bmatrix} u_{12}^p \\ u_{22}^p \end{bmatrix}. \tag{4.6.5}$$

Defining the submatrices $\mathbf{G}_1$ and $\mathbf{G}_2$ of $\mathbf{U}^s$, and the vector $\boldsymbol{\mu}$, as follows:

$$\mathbf{G}_1 \triangleq \begin{bmatrix} u_{11}^1 \cdots u_{11}^{d_2^{x,2}} \\ u_{12}^1 \cdots u_{12}^{d_2^{x,2}} \end{bmatrix}, \quad \mathbf{G}_2 \triangleq \begin{bmatrix} u_{12}^1 \cdots u_{12}^{d_2^{x,2}} \\ u_{22}^1 \cdots u_{22}^{d_2^{x,2}} \end{bmatrix}, \text{ and } \quad \boldsymbol{\mu} \triangleq \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{d_2^{x,2}} \end{bmatrix}, \tag{4.6.6}$$

the systems in (4.6.5) can be written as follows:

$$z_1 \, \mathbf{z} = \mathbf{G}_1 \, \boldsymbol{\mu}, \qquad z_2 \, \mathbf{z} = \mathbf{G}_2 \, \boldsymbol{\mu} \, . \tag{4.6.7}$$

Since both $\mathbf{G}_1 \, \boldsymbol{\mu}$ and $\mathbf{G}_2 \, \boldsymbol{\mu}$ should yield a vector in the same direction, viz. in the direction of $\mathbf{z}$, Eq. (4.6.7) describes a generalized eigenproblem. This can be seen more clearly as follows. Suppose for the moment that $z_1 \neq 0, z_2 \neq 0$ (without loss of generality) and write (4.6.7) as:

$$\mathbf{z} = \frac{1}{z_1} \mathbf{G}_1 \boldsymbol{\mu} = \frac{1}{z_2} \mathbf{G}_2 \boldsymbol{\mu} \, . \tag{4.6.8}$$

The vector $\boldsymbol{\mu}$ is a generalized eigenvector of the matrices $\mathbf{G}_1$ and $\mathbf{G}_2$. In more familiar terms, $\boldsymbol{\mu}$ is a generalized eigenvector of the *matrix pencil* $\mathbf{G}_2 - \lambda \mathbf{G}_1$ with generalized eigenvalue $\lambda = \frac{z_2}{z_1}$ because it satisfies $\mathbf{G}_2 \boldsymbol{\mu} = \frac{z_2}{z_1} \mathbf{G}_1 \boldsymbol{\mu}$. We will denote the Generalized Eigenvalue Decomposition of $\mathbf{G}_1$ and $\mathbf{G}_2$ by $\mathrm{gevd}(\mathbf{G}_1, \mathbf{G}_2)$. It can be computed using standard tools from linear algebra, e.g. in Matlab the function `eig` can be used. Assume that the generalized eigenvectors and eigenvalues resulting from the GEVD are stored in the matrices $\mathbf{M}_G$ and $\boldsymbol{\Lambda}_G$ respectively:

$$[\mathbf{M}_G, \boldsymbol{\Lambda}_G] = \mathrm{gevd} \left(\mathbf{G}_1, \mathbf{G}_2 \right). \tag{4.6.9}$$

This system has $d_2^{x,2}$ generalized eigenvectors and eigenvalues. Hence, $\mathbf{M}_G = \left[\boldsymbol{\mu}^1 \cdots \boldsymbol{\mu}^{d_2^{x,2}} \right]$ and $\boldsymbol{\Lambda}_G = \mathrm{diag}(\lambda_1, \ldots, \lambda_{d_2^{x,2}})$. After having computed this decomposition, the columns of $\mathbf{A}$ can be recovered either from the eigenvectors or the eigenvalues. Firstly, Equations (4.6.7) and (4.6.8) show that for the solution eigenvector $\boldsymbol{\mu}^j$ the vectors $\mathbf{G}_1 \boldsymbol{\mu}^j$ and $\mathbf{G}_2 \boldsymbol{\mu}^j$ lie in the direction of a solution $\mathbf{z}^j$ of (4.6.3). Hence, given $\boldsymbol{\mu}^j$ the corresponding solution for $\mathbf{z}^j$ can be computed as an arbitrary linear combination of these vectors, i.e.

$$\mathbf{z}^j = \frac{(\alpha_1 \mathbf{G}_1 + \alpha_2 \mathbf{G}_2) \boldsymbol{\mu}^j}{\| (\alpha_1 \mathbf{G}_1 + \alpha_2 \mathbf{G}_2) \boldsymbol{\mu}^j \|}, \quad j = 1, 2 \, , \tag{4.6.10}$$

where α_1 and α_2 are arbitrary non-zero real-valued scalars. Alternatively, since the eigenvalue λ_j represents the ratio $\frac{z^{j,2}}{z^{j,1}}$, the columns can be recovered from this ratio using the same approach as in Section 4.5, i.e.:

$$\mathbf{z}^j = \frac{1}{\sqrt{1 + (\lambda_j)^2}} \begin{bmatrix} 1 \\ \lambda_j \end{bmatrix}, \quad j = 1, 2 \, . \tag{4.6.11}$$

Now, we will apply the approach described in this section to the examples in Sections 4.4.2 and 4.4.3 respectively.

4.6.1 Application to Example 1 in Section 4.4.2

From the Singular Value Decomposition of $\mathbf{C}_{\mathrm{u}}^{x}$ in Section 4.4.2, it follows that for the ideal case $\mathbf{U}^{s}$ is given by ($d_2^{x,2} = 2$):

$$\mathbf{U}^{s} = \begin{bmatrix} \mathbf{u}^1 & \mathbf{u}^2 \end{bmatrix} = \begin{bmatrix} -0.9152 & 0.3459 \\ -0.2473 & -0.0762 \\ -0.3183 & -0.9352 \end{bmatrix}.$$

From (4.6.6) it then follows that $\mathbf{G}_1$ and $\mathbf{G}_2$ are given by:

$$\mathbf{G}_1 \triangleq \begin{bmatrix} -0.9152 & 0.3459 \\ -0.2473 & -0.0762 \end{bmatrix}, \quad \mathbf{G}_2 \triangleq \begin{bmatrix} -0.2473 & -0.0762 \\ -0.3183 & -0.9352 \end{bmatrix}. \tag{4.6.12}$$

Using the `eig` function of Matlab for computing (4.6.9), the following result is obtained:

$$\mathbf{M} = \begin{bmatrix} 1.0000 & -0.4103 \\ -0.2868 & -1.0000 \end{bmatrix}, \qquad \mathbf{\Sigma}_{\mathrm{g}} = \mathrm{diag}(0.2222, 6.0000). \tag{4.6.13}$$

Using (4.6.10) with $\alpha_1 = \alpha_2 = 1$ gives:

$$\mathbf{z}^1 = \begin{bmatrix} -0.9762 \\ -0.2169 \end{bmatrix}, \quad \mathbf{z}^2 = \begin{bmatrix} 0.1644 \\ 0.9864 \end{bmatrix},$$

which lie in the direction of the first and second column of $\mathbf{A}$ respectively. Note that the eigenvalues 0.2222 and 6.0000 are equal to the ratios obtained in Section 4.5.4.1, and also correspond to the correct solutions. Similarly, for the estimated case also the same results as in Section 4.5.4.1 are obtained.

4.6.2 Application to Example 2 in Section 4.4.3

From the Singular Value Decomposition of $\mathbf{C}_{\mathrm{u}}^{x}$ in Section 4.4.3, it follows that for ideal case $\mathbf{U}^{s}$ is given by ($d_2^{x,2} = 1$):

$$\mathbf{U}^{s} = \begin{bmatrix} \mathbf{u}^1 & \mathbf{u}^2 \end{bmatrix} = \begin{bmatrix} -0.9751 \\ -0.2167 \\ -0.0482 \end{bmatrix}.$$

From (4.6.6) it then follows that $\mathbf{G}_1$ and $\mathbf{G}_2$ are given by:

$$\mathbf{G}_1 \triangleq \begin{bmatrix} -0.9751 \\ -0.2167 \end{bmatrix}, \quad \mathbf{G}_2 \triangleq \begin{bmatrix} -0.2167 \\ -0.0482 \end{bmatrix}. \tag{4.6.14}$$

In this simple case, the matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ have degenerated to vectors and the eigenvectors to scalars. Consequently, as is evident from (4.6.7) or (4.6.8), the solution vector $\mathbf{z}^1 = \mathbf{z}^1$ lies in the same direction as specified by $\mathbf{G}_1$ and $\mathbf{G}_2$, i.e.:

$$\mathbf{z}^1 = \frac{\mathbf{G}_1}{\|\mathbf{G}_1\|} = \frac{\mathbf{G}_2}{\|\mathbf{G}_2\|} = \begin{bmatrix} -0.9762 \\ -0.2169 \end{bmatrix} = -\frac{\mathbf{a}^1}{\|\mathbf{a}^1\|} = -\frac{\mathbf{a}^2}{\|\mathbf{a}^2\|}.$$

Hence, the columns have been identified correctly. Similarly, for the estimated case also the same results as in Section 4.5.4.2 are obtained.

Likewise, it can also be shown that for the examples in Sections 4.4.4 and 4.4.5 exactly the same results as those obtained in Sections 4.5.4.3 and 4.5.4.4 are obtained.

4.7 Conclusions and discussion

In this chapter we have introduced the rationale behind using subspace techniques for exploiting the Second Order Temporal Structure (SOTS) in the data for performing Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI). We have focussed on exploiting a specific kind of temporal structure, viz. second-order non-whiteness. In the next chapters we will also consider the exploitation of non-stationarity, as well as the simultaneous exploitation of non-whiteness and non-stationarity. Our method is based on the subspace decomposition of a subspace matrix that contains certain sensor correlation values for different lags in a specific arrangement. Based on certain assumptions about the second order temporal structure of the source and noise signals, we have shown that the MIBI problem can be 'projected onto' two dual mathematical problems. In the first problem, a well-structured system of homogeneous polynomial equations of degree two that is satisfied by the columns of the mixing matrix was derived. We have provided several possible methods for the solution of this system and studied the structure of its equations. In the second problem, a generalized eigenvalue problem has been formulated and solved. The assumptions underlying the two problem formulations have been defined on a certain Region Of Support consisting of a set of lag indices and mainly serve to ensure that the source signals are mutually unrelated, that sufficient temporal structure is present in the source signals, that the source and noise signals are mutually unrelated, and that the noise signals have a 'simpler' temporal structure than the source signals. This latter fact is employed in a systematic manner for eliminating the effect of additive sensor noise on the mixing matrix estimation. In order to keep a clear overview of the main themes presented in this chapter, we have summarized the key ingredients of the resulting two by two MIBI algorithm in Alg. 4.5 on the next page.

Algorithm 4.5 Overview of our 2×2 MIBI method developed in this chapter.

1: *Compute/estimate the sensor correlation functions* $r_{11}^x[k] \triangleq E\{x_1[n]x_1[n-k]\}$, $r_{12}^x[k] = r_{21}^x[k] \triangleq E\{x_1[n]x_2[n-k]\}$ *and* $r_{22}^x[k] \triangleq E\{x_2[n]x_2[n-k]\}$ *for several lags* k *in Noise-Free ROS* $\Omega_k^{s\setminus\nu}$*;*

2: *Arrange these values in a (uniquefied) subspace matrix* $\mathbf{C}_{\mathrm{u}}^x$*, e.g. if white sensor noise is present we could consider the noise-free lags* $k = \{\ldots,-2,-1,1,2,\ldots\}$*, and would construct* $\mathbf{C}_{\mathrm{u}}^x$ *as follows:*

$$\mathbf{C}_{\mathrm{u}}^x \triangleq \begin{bmatrix} \cdots & r_{11}^x[-2] & r_{11}^x[-1] & r_{11}^x[1] & r_{11}^x[2]\cdots \\ \cdots & r_{12}^x[-2] & r_{12}^x[-1] & r_{12}^x[1] & r_{12}^x[2]\cdots \\ \cdots & r_{22}^x[-2] & r_{22}^x[-1] & r_{22}^x[1] & r_{22}^x[2]\cdots \end{bmatrix};$$

3: *Compute the Singular Value Decomposition (SVD) of* $\mathbf{C}_{\mathrm{u}}^x$ *and split the result into signal and null/noise subspace parts as follows:*

$$\mathbf{C}_{\mathrm{u}}^x = \mathbf{U}^x\mathbf{\Lambda}^x\mathbf{V}^x = \mathbf{U}^s\mathbf{\Lambda}^s\mathbf{V}^s + \mathbf{U}^\nu\mathbf{\Lambda}^\nu\mathbf{V}^\nu;$$

4: *Choose one of the following two MIBI problem reformulations:*

- *Using only the coefficients in* $\mathbf{U}^\nu$*, a system of homogeneous polynomial equations of degree two is constructed, the solutions of which are the columns of the mixing matrix;*
- *Using only the coefficients in* $\mathbf{U}^s$*, a multi-matrix generalized eigenvalue problem is formulated, whose generalized eigenvectors are directly related to the columns of the mixing matrix;*

The duality/complementarity of these two formulations is illustrated by the fact that the coefficients they use come from $\mathbf{U}^\nu$ *and* $\mathbf{U}^s$ *respectively, which both belong to* $\mathbf{U}^x = [\mathbf{U}^s\ \mathbf{U}^\nu]$*;*

5: *Solve the problem defined in the previous step, e.g. using homotopy.*

CHAPTER 5

MIBI based on second order temporal structure: complex-valued $D \times S$ mixing case

In this chapter, the results from the previous chapter are generalized in several directions. Firstly, instead of the MIBI scenario with two sources and two sensors depicted in Fig. 4.1, here we consider the scenario with S sources and D sensors depicted in Fig. 5.1. Secondly, now we consider the more general complex-valued case with arbitrary conjugation pattern. Finally, instead of employing only one type of second order temporal statistical diversity, viz. non-whiteness, here we also employ second order non-stationarity or quasi-stationarity under the assumption that each source signal has a 'non-stationarity pattern' that is sufficiently different from the others; see also Chapter 1. We will demonstrate that these different types of statistical variability can be exploited in the same unified manner. It is possible to either exploit the non-whiteness or non-stationarity alone, or both simultaneously. We explain the reasoning behind the basic assumptions upon which our method is based, and justify their equity in practice. As will become clear, because the notation adopted in this work has been set up in such a way that it allows a uniform description of the theory for the general MIBI problem, most generalizations presented in this chapter follow from the corresponding ones in the previous chapter in a more or less trivial way. As before, the available statistics are arranged in a subspace matrix, which may now contain complex-valued sensor correlation values for different lags and times. Based on certain assumptions about the second order non-whiteness and non-stationarity of the source and noise signals, the MIBI problem again is 'projected onto' two complementary or dual mathematical problems by applying subspace techniques to the subspace matrix. In the first problem, a well-structured system of homogeneous polynomial-like equations of degree two has to be solved. For reasons that will become clear later on we will call the polynomial-like functions in the system *polyconjugals*; see Section 5.3.4.4, for example. In the second problem, a multi-matrix generalized eigenvalue decomposition has to be performed. Possible methods for the solution of these problems are presented, and their geometric and algebraic structure are highlighted. Along our disquisition we introduce several new symbols, e.g. for pairs of time indices, that make the transition to the most general scenario discussed in the next chapter as smooth as possible.

The outline of this chapter is similar to that of Chapter 4. Section 5.1 describes the structure of the $D \times S$ MIBI model, whereas Section 5.2 explains the model assumptions. Next, in Section 5.3 a system of homogeneous polyconjugal equations of degree two satisfied by the columns of the mixing matrix is derived, thereby highlighting the algebraic structure of our problem formulation. After that, in Section 5.4 the geometric structure associated with this system of equations is investigated and examples are given. Subsequently, in Section 5.5 a homotopy method for solving the obtained system is presented. Then, in Section 5.6 the MIBI problem is written as a multi-matrix generalized eigenvalue problem, and a solution method is given. Finally, conclusions and discussion are presented in Section 5.7. In order to

keep a clear overview of the main theme of this chapter, we summarize the key ingredients of the resulting $D \times S$ MIBI algorithm in the following high-level algorithm:

Algorithm 5.1 Pre-overview of the $D \times S$ MIBI method developed in this chapter.

1: *Compute/estimate all sensor correlation functions on a Noise-Free ROS consisting of several time or time-lag pairs;*

2: *Arrange these values in a uniquified subspace matrix* $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$*;*

3: *Compute the Singular Value Decomposition of* $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ *and split the result into signal and null/noise subspace parts;*

4: *Choose one of the following two MIBI problem projections:*

- *Construct a system of D-variate homogeneous polyconjugal equations with conjugation pair (c_1, c_2), whose solutions are the columns of the mixing matrix;*
- *Construct a multi-matrix generalized eigenvalue problem, whose generalized eigenvectors are directly related to the columns of the mixing matrix;*

5: *Solve the problem defined in the previous step.*

5.1 MIBI model structure

In this chapter, we consider an instantaneous mixing model with S source and D sensor signals as depicted in Fig. 5.1; see also Section 1.1.1 and Fig. 1.1. According to (1.1.1), the i-th sensor signal $x_i[n]$ is given by:

$$\begin{aligned} x_i[n] &= \sum_{j=1}^{S} a_i^j s_j[n] + \nu_i[n] \\ &= a_i^1 s_1[n] + \cdots + a_i^S s_S[n] + \nu_i[n] \qquad \forall\, n \in \mathbb{Z}, \quad \forall\, 1 \leq i \leq D\,, \end{aligned} \tag{5.1.1}$$

where a_i^j is the instantaneous transfer coefficient from the j-th source to the i-th sensor, $s_j[n]$ is the j-th source signal at discrete time n, and $\nu_i[n]$ is the i-th noise signal at discrete time n. The system and signals may be real- or complex-valued. Moreover, it is assumed that all involved signals are zero-mean and are allowed to be non- or quasi-stationary. The

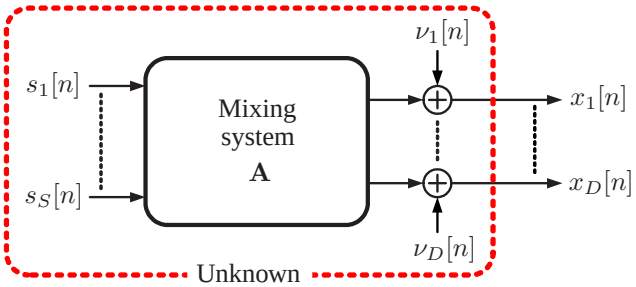

Figure 5.1: *MIBI problem setup.*

zero-mean assumption is made solely for mathematical convenience and does not imply any loss of generality. In case non-zero-mean signals are involved, correlations simply have to be replaced by covariances. In the case of non- or quasi-stationary signals, the involved correlation functions and other occurring quantities may be considered as depending on two time indices, say n_1 and n_2, or on a time index and a lag index, say n and k. In matrix-vector notation, (5.1.1) can be written as follows (see also (1.1.1)):

$$\mathbf{x}[n] = \sum_{j=1}^{S} \mathbf{a}^j \, s_j[n] + \boldsymbol{\nu}[n] = \mathbf{A}\mathbf{s}[n] + \boldsymbol{\nu}[n] \qquad \forall\, n \in \mathbb{Z}\,, \tag{5.1.2}$$

where:

$$\mathbf{x}[n] \triangleq \begin{bmatrix} x_1[n] \\ \vdots \\ x_D[n] \end{bmatrix}, \quad \mathbf{s}[n] \triangleq \begin{bmatrix} s_1[n] \\ \vdots \\ s_S[n] \end{bmatrix}, \quad \boldsymbol{\nu}[n] \triangleq \begin{bmatrix} \nu_1[n] \\ \vdots \\ \nu_D[n] \end{bmatrix}, \quad \text{and} \quad \mathbf{a}^j \triangleq \begin{bmatrix} a_1^j \\ \vdots \\ a_D^j \end{bmatrix}$$

are column vectors of sensor signals, source signals, additive noise signals and mixing elements respectively. The vectors $\mathbf{x}[n]$, $\boldsymbol{\nu}[n]$, and $\mathbf{a}^1, \ldots, \mathbf{a}^S$ are elements of $\mathbb{R}_D$ or $\mathbb{C}_D$, whereas the vector $\mathbf{s}[n]$ is an element of $\mathbb{R}_S$ or $\mathbb{C}_S$. The unknown mixing matrix $\mathbf{A}$ is an element of $\mathbb{R}_D^S$ or $\mathbb{C}_D^S$, and can be written in terms of its columns as $\mathbf{A} = \begin{bmatrix} \mathbf{a}^1 \cdots \mathbf{a}^S \end{bmatrix}$ and in terms of its elements as:

$$\mathbf{A} = \begin{bmatrix} a_1^1 & \cdots & a_1^S \\ \vdots & \ddots & \vdots \\ a_D^1 & \cdots & a_D^S \end{bmatrix}.$$

As before, we sometimes refer to the mixing matrix as the *array response matrix*, and to the columns of the mixing matrix as the *array response vectors*. For completeness and clarity of the following developments, (5.1.2) is written out entirely in terms of the individual quantities:

$$\begin{aligned} \mathbf{x}[n] &= \begin{bmatrix} a_1^1 \\ \vdots \\ a_D^1 \end{bmatrix} s_1[n] + \cdots + \begin{bmatrix} a_1^S \\ \vdots \\ a_D^S \end{bmatrix} s_S[n] + \begin{bmatrix} \nu_1[n] \\ \vdots \\ \nu_D[n] \end{bmatrix} \\ &= \begin{bmatrix} a_1^1 & \cdots & a_1^S \\ \vdots & \ddots & \vdots \\ a_D^1 & \cdots & a_D^S \end{bmatrix} \begin{bmatrix} s_1[n] \\ \vdots \\ s_S[n] \end{bmatrix} + \begin{bmatrix} \nu_1[n] \\ \vdots \\ \nu_D[n] \end{bmatrix} \qquad \forall\, n \in \mathbb{Z}\,. \end{aligned}$$

As we have explained in Section 2.4, two indeterminacies or ambiguities are inherent to this model, viz. the norms and the order of the columns of the mixing matrix cannot be resolved; see also Section 4.1. This means that the columns and source signals can only be recovered up to a scale factor. Taking into account these two indeterminacies, the goal of MIBI is to recover the columns of the mixing system in some arbitrary order and with some arbitrary non-zero norms. Usually, these two indeterminacies do not cause serious problems because for many applications the most relevant information is in the 'directions' of the columns rather than in their order or magnitudes.

5.2 MIBI model assumptions

In the previous chapter it was shown that in order to be able to exploit the Second Order Temporal Structure (SOTS) in the data, several assumptions need to be made. The essence of these assumptions was summarized in *AS1-AS4* on page 138. For simplicity, it was assumed that all involved signals were stationary. This allowed the involved Region Of Support (ROS) to be defined in the relatively simple one-dimensional 'lag-domain'. In this chapter, we will see that the stationarity assumption can and should be relaxed in order to allow the exploitation of second order non-stationarity. The involved correlation functions may then be considered as depending on two time indices, say n_1 and n_2, or on a time index and a lag index, say n and k. Consequently, the ROS should be defined in the corresponding two-dimensional domain. In fact, as we will see in the next chapter, the dimensionality, shape, and size of a ROS are not essential for the rationale behind our approach. The true essence lies in the relations between the various functions that are defined on the ROS under consideration. The assumptions about these relations mainly serve two purposes. Firstly, they ensure that sufficient temporal structure is present in the source signals, and secondly that the noise signals have a 'simpler' temporal structure than the source signals. These properties are conveyed in general form by *MAS1-MAS4* on page 23. In the sequel of this section, we formulate them for the scenario considered in this chapter. In Section 5.2.1 we start by recapitulating and generalizing the definitions of correlation functions presented earlier to make them suitable for the current scenario, and then we discuss and summarize the various assumptions. In Section 5.2.2 we illustrate the different concepts by means of an example.

5.2.1 Definitions of correlation functions and regions of support

As we have explained in Sections 1.2.7, A.6, and B.3, a two-dimensional correlation or moment function can be defined in different ways corresponding to the pattern in which its arguments are conjugated. Such a *conjugation pattern* is represented by a *conjugation tuple*, which in the two-argument case is also called *conjugation pair*, and is written in the upper-right corner of the considered symbol or function. As we also have remarked in Sections 1.2.7 and A.6, and justified in Section B.3, the most suitable conjugation pair for a particular application depends on the type of signals involved. In the derivations in this chapter it will often be assumed without loss of generality that a specific conjugation pair (c_1, c_2) has been chosen and fixed subsequently, where c_1 and c_2 can either be '$*$', which means conjugation, or '$\circ$', which means no conjugation (see Section A.5). In other words, the theory will be developed for an arbitrary fixed conjugation pair (c_1, c_2), but it is applicable to any $(c_1, c_2) \in \mathcal{C}_2$, where the set $\mathcal{C}_2$ of all four possible conjugation pairs is defined in (A.5.2).

As in the previous chapter (see Section 4.2), for the current scenario four different correlation functions are needed for the formulation of the required assumptions and derivation of the theory, viz. the source, noise, source-noise, and noise-source auto- and cross-correlation functions. For convenience we list their definitions here. The source correlation functions are defined as follows (see (A.6.6)):

$$r_{j_1 j_2}^{s,c_1 c_2}[n_1, n_2] \triangleq \operatorname{mom}\Big(\big(s_{j_1}[n_1]\big)^{c_1}, \big(s_{j_2}[n_2]\big)^{c_2} \Big) = E\Big\{ \big(s_{j_1}[n_1]\big)^{c_1} \big(s_{j_2}[n_2]\big)^{c_2} \Big\}$$
$$\forall\, (n_1, n_2) \in \mathbb{Z}_2, \quad \forall\, 1 \leq j_1, j_2 \leq S, \quad \forall\, (c_1, c_2) \in \mathcal{C}_2\,, \tag{5.2.1}$$

the noise correlation functions as:

$$r_{i_1 i_2}^{\nu,c_1c_2}[n_1,n_2] \triangleq \mathrm{mom}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1},\big(\nu_{i_2}[n_2]\big)^{c_2}\Big) = E\Big\{\big(\nu_{i_1}[n_1]\big)^{c_1}\big(\nu_{i_2}[n_2]\big)^{c_2}\Big\}$$
$$\forall\,(n_1,n_2)\in\mathbb{Z}_2,\quad \forall\,1\le i_1,i_2\le D,\quad \forall\,(c_1,c_2)\in\mathcal{C}_2\,, \tag{5.2.2}$$

the source-noise correlation functions by:

$$r_{ji}^{s\nu,c_1c_2}[n_1,n_2] \triangleq \mathrm{mom}\Big(\big(s_j[n_1]\big)^{c_1},\big(\nu_i[n_2]\big)^{c_2}\Big) = E\Big\{\big(s_j[n_1]\big)^{c_1}\big(\nu_i[n_2]\big)^{c_2}\Big\}$$
$$\forall\,(n_1,n_2)\in\mathbb{Z}_2,\quad \forall\,1\le j\le S,\quad \forall\,1\le i\le D,\quad \forall\,(c_1,c_2)\in\mathcal{C}_2\,, \tag{5.2.3}$$

and finally the noise-source correlation functions are defined by:

$$r_{ij}^{\nu s,c_1c_2}[n_1,n_2] \triangleq \mathrm{mom}\Big(\big(\nu_i[n_1]\big)^{c_1},\big(s_j[n_2]\big)^{c_2}\Big) = E\Big\{\big(\nu_i[n_1]\big)^{c_1}\big(s_j[n_2]\big)^{c_2}\Big\}$$
$$\forall\,(n_1,n_2)\in\mathbb{Z}_2,\quad \forall\,1\le i\le D,\quad \forall\,1\le j\le S,\quad \forall\,(c_1,c_2)\in\mathcal{C}_2\,. \tag{5.2.4}$$

To clearly and unambiguously formulate the assumptions, we need to make some definitions regarding the Regions Of Support of the various correlation functions; see Fig. 5.2. Among other things, these definitions are necessary for specifying explicitly what conditions are required on the temporal structure/diversity of the source and noise signals, and in what sense the noise signals have a 'simpler' temporal structure than the source signals. To start with, let $\Omega_{n_1n_2}^{s,c_1c_2}\subseteq\mathbb{Z}_2$ be some ROS in the domain of time index pairs (n_1,n_2) on which all source correlation functions $r_{j_1j_2}^{s,c_1c_2}[n_1,n_2]$ with $1\le j_1,j_2\le S$ generally are non-zero. This *'Source Region Of Support'* is indicated schematically by the largest ellipse in Fig. 5.2. The source correlation functions may also be non-zero outside this ROS, but this is not necessary. Note that the ROS now also depends on the conjugation pair (c_1,c_2) because the arguments of the involved moment functions are conjugated according to this specific conjugation pair. The same holds for the other regions of support that will be defined now. Let $\Omega_{n_1n_2}^{\nu,c_1c_2}\subseteq\mathbb{Z}_2$ be some ROS on which all noise correlation functions $r_{i_1i_2}^{\nu,c_1c_2}[n_1,n_2]$ with $1\le i_1,i_2\le D$ generally are non-zero. This *'Noise Region Of Support'* is indicated by the small tilted ellipse with the one to lightest gray shade. It is assumed that the noise correlation functions are zero outside $\Omega_{n_1n_2}^{\nu,c_1c_2}$. Furthermore, let $\Omega_{n_1n_2}^{s\nu,c_1c_2}\subseteq\mathbb{Z}_2$ be the *'Source-Noise Region Of Support'* on which the source-noise correlation functions $r_{ji}^{s\nu,c_1c_2}[n_1,n_2]$ with $1\le j\le S$, $1\le i\le D$ generally are non-zero. See Fig. 5.2 for its schematic indication. Similarly, let $\Omega_{n_1n_2}^{\nu s,c_1c_2}\subseteq\mathbb{Z}_2$ be the *'Noise-Source Region Of Support'* on which the noise-source correlation functions $r_{ij}^{\nu s,c_1c_2}[n_1,n_2]$ with $1\le i\le D$, $1\le j\le S$ generally are non-zero. Again see Fig. 5.2 for a schematic indication. It is assumed that both the source-noise and noise-source correlation functions are zero outside their respective regions of support. The union $\Omega_{n_1n_2}^{\nu,c_1c_2}\cup\Omega_{n_1n_2}^{s\nu,c_1c_2}\cup\Omega_{n_1n_2}^{\nu s,c_1c_2}$ defines the ROS where unwanted noise is present. The part of this union that extends into the Source ROS is given by the intersection $\Omega_{n_1n_2}^{s\cap\nu,c_1c_2}\triangleq\Omega_{n_1n_2}^{s,c_1c_2}\cap\big(\Omega_{n_1n_2}^{\nu,c_1c_2}\cup\Omega_{n_1n_2}^{s\nu,c_1c_2}\cup\Omega_{n_1n_2}^{\nu s,c_1c_2}\big)$ and is indicated by the dashed line in the figure. Finally, let the *'Noise-Free Region Of Support'* $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ be defined by:

$$\boxed{\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}\triangleq\Omega_{n_1n_2}^{s,c_1c_2}\setminus\Omega_{n_1n_2}^{s\cap\nu,c_1c_2}=\Omega_{n_1n_2}^{s,c_1c_2}\setminus\big(\Omega_{n_1n_2}^{s,c_1c_2}\cap\big(\Omega_{n_1n_2}^{\nu,c_1c_2}\cup\Omega_{n_1n_2}^{s\nu,c_1c_2}\cup\Omega_{n_1n_2}^{\nu s,c_1c_2}\big)\big)\,.} \tag{5.2.5}$$

This ROS is indicated by the area with the lightest gray shade, i.e. the large ellipse minus the part surrounded by the dashed line. In order to apply the theory that will be developed in the sequel, the various regions of support have to be known in advance. More precisely, in fact knowledge of $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ is sufficient for the formulation of the assumptions in the remainder

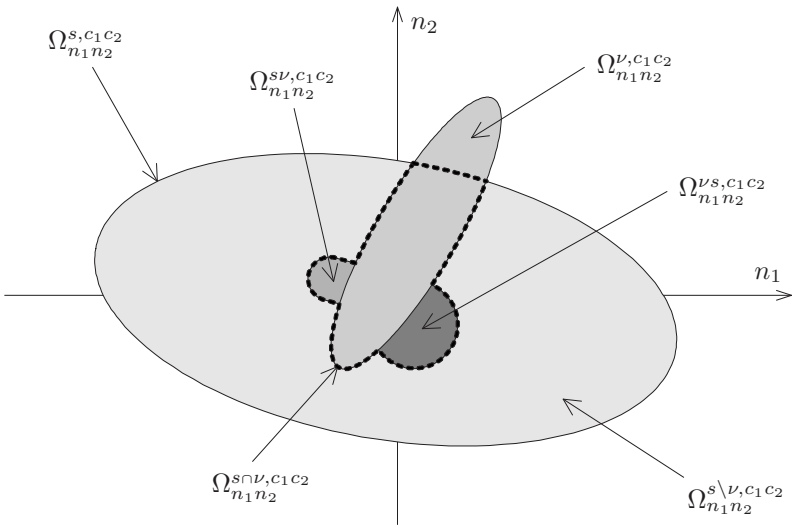

Figure 5.2: *Regions of support in the domain of time index pairs* (n_1, n_2).

of this section. The other regions of support are merely required for defining or finding a suitable Noise-Free ROS $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$. Hence, in principle the various regions of support are part of the required a priori knowledge for a particular application, but it suffices to know only $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$. Guidelines for choosing the different regions of support in a logical and sensible manner easily follow from examples that will be presented during our exposition; see also Chapter 4. In case the conjugation pair $(c_1, c_2) = (\circ, \circ)$ is chosen, i.e. none of the arguments is conjugated, it is omitted from the symbolic notation. For example, $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$ would simply be written as $\Omega_{n_1 n_2}^{s\setminus\nu}$, $r_{ij}^{\nu s,c_1c_2}[n_1, n_2]$ as $r_{ij}^{\nu s}[n_1, n_2]$, etcetera. Note that in a non-stationary signal scenario with abuse of notation we may write 'n, k' or '$n; k$' instead of 'n_1, n_2', and 'nk' or '$n; k$' instead of '$n_1 n_2$' in all occurring quantities.

For convenience of notation, let $\mathbb{C}[\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}]$ denote the set/space of all complex-valued functions that depend on a time index pair (n_1, n_2) and are defined on the Noise-Free ROS $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$. Most functions used in the sequel of this chapter are assumed to be elements of $\mathbb{C}[\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}]$. If functions are defined on another ROS, this will be mentioned explicitly. As we have said on page 138 in Section 4.2, throughout the thesis we use the dedicated symbol N to denote the cardinality of the Noise-Free Region Of Support. Hence, similarly to (4.2.17) for the current scenario we define:

$$N \triangleq \left|\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}\right|. \tag{5.2.6}$$

Using the definitions above for the various regions of support, the assumptions about the MIBI model on which the results in this chapter are based will now be discussed along the lines of Section 4.2. Now, the random processes are allowed to be complex-valued and the possibly complex correlation functions are defined on $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$ instead of $\Omega_k^{s\setminus\nu}$. The assumptions formulated in this section are summarized as *AS1-AS4* in the list on page 222. Similarly to Section 4.2, we discuss each of the assumptions in turn; in addition to the kind of exposition given there, we also justify the practical equity of the assumptions by explaining their physical plausibility.

To start with, in *AS1* on the following page an assumption is made on the temporal cross-structure of the source signals, viz. that all possible source cross-correlation functions $\{r_{j_1 j_2}^{s,c_1c_2}[n_1,n_2]\}_{1\leq j_1\neq j_2\leq S}$ are zero on the Noise-Free ROS $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$. Mathematically, this can be stated as follows:

$$r_{j_1 j_2}^{s,c_1c_2}[n_1,n_2]=0 \qquad \forall\,(n_1,n_2)\in\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2},\quad \forall\,1\leq j_1\neq j_2\leq S\,. \tag{5.2.7}$$

This assumption is reminiscent of the conventional assumption adopted in several sensor array models that the information carrying signals are spatially and temporally statistically independent from each other, see also Chapter 1 and Section 2.3.2. It is physically plausible in many practical situations because signals often are generated independently of each other by different sources. For example, generally speech signals uttered by different persons speaking different words at the same or different times are un(cor)related. In this respect, the uncorrelatedness assumption in (5.2.7) or *AS1* is weak w.r.t. the more general statistical independence assumption because the latter one requires the cross-cumulant functions of all orders to be zero, whereas (5.2.7) only requires the second order cross-moment functions to be zero.

Secondly, in *AS2* an assumption is made on the temporal auto-structure of the source signals, viz. that the source auto-correlation functions $\left\{r_{jj}^{s,c_1c_2}[n_1,n_2]\right\}_{1\leq j\leq S}$ are linearly independent on $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$. Mathematically, this can be stated as follows:

$$\sum_{j=1}^{S}\xi^j\,r_{jj}^{s,c_1c_2}[n_1,n_2]=0\quad\forall\,(n_1,n_2)\in\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}\quad\Longrightarrow\quad\xi^j=0\quad\forall\,1\leq j\leq S\,, \tag{5.2.8}$$

where $\xi^1,\ldots,\xi^S$ are real- or complex-valued scalars. This assumption is also physically plausible in many practical situations because different source signals usually have quite different temporal structures. For example, considering the auto-correlation functions of speech signals (computed over a certain time window) uttered by different persons speaking different words at the same or different times, it is very likely that these functions are so different from each other that they indeed are linearly independent.

Thirdly, in *AS3* an assumption is made on the temporal structure of the noise signals, viz. that all noise auto- and cross-correlation functions $\{r_{i_1 i_2}^{\nu,c_1c_2}[n_1,n_2]\}_{1\leq i_1,i_2\leq D}$ are identically zero on $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$. Mathematically, this can be stated as follows:

$$r_{i_1 i_2}^{\nu,c_1c_2}[n_1,n_2]=0 \qquad \forall\,(n_1,n_2)\in\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2},\quad \forall\,1\leq i_1,i_2\leq D\,. \tag{5.2.9}$$

The existence of such a Noise-Free ROS is physically plausible in many practical situations since often the statistics of noise signals are quite different from, and simpler than, those of information-carrying source signals. In fact, this is an essential assumption in many signal processing applications, where typical signal scenarios include colored information-carrying signals observed in white noise, and (possibly colored) non-Gaussian information-carrying signals observed in (possibly colored) Gaussian noise; see also Section 1.2.5.

Finally, in *AS4* an assumption is made on the temporal cross-structure of the source and noise signals, viz. that all cross-correlation functions $\{r_{ji}^{s\nu,c_1c_2}[n_1,n_2]\}_{1\leq j\leq S,1\leq i\leq D}$ and $\{r_{ij}^{\nu s,c_1c_2}[n_1,n_2]\}_{1\leq i\leq D,1\leq j\leq S}$ between the source and noise signals are identically zero on $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$. This can be stated mathematically as:

$$r_{ji}^{s\nu,c_1c_2}[n_1,n_2]=0 \qquad \forall\,(n_1,n_2)\in\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2},\quad \forall\,1\leq j\leq S,\quad \forall\,1\leq i\leq D \tag{5.2.10}$$

and:

$$r_{ij}^{\nu s,c_1c_2}[n_1,n_2] = 0 \qquad \forall\,(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}, \quad \forall\,1 \le i \le D, \quad \forall\,1 \le j \le S \quad (5.2.11)$$

respectively. This assumption is reminiscent of the assumption commonly adopted in several sensor array models that the information carrying signals are spatially and temporally statistically independent from sensor noise signals. Many signal processing theories and applications adopt this assumption. It is physically plausible in many in practical situations because the source signals often are generated independently of the sensor noise signals.

The last two assumptions imply that the considered ROS $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$, which is the only ROS required for the formulation of assumptions *AS1-AS4*, is noise-free, thereby allowing us to gather noise-free sensor statistics on it. In other words, as we have explained above, the Noise-Free ROS $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2} \triangleq \Omega_{n_1n_2}^{s,c_1c_2} \setminus \Omega_{n_1n_2}^{s\cap\nu,c_1c_2}$ is a 'Source-Only' ROS that equals the Source ROS $\Omega_{n_1n_2}^{s,c_1c_2}$ minus the part $\Omega_{n_1n_2}^{s\cap\nu,c_1c_2} \triangleq \Omega_{n_1n_2}^{s,c_1c_2} \cap \left(\Omega_{n_1n_2}^{\nu,c_1c_2} \cup \Omega_{n_1n_2}^{s\nu,c_1c_2} \cup \Omega_{n_1n_2}^{\nu s,c_1c_2}\right)$ that $\Omega_{n_1n_2}^{s,c_1c_2}$ has in common with the Noise-Contaminated ROS $\Omega_{n_1n_2}^{\nu,c_1c_2} \cup \Omega_{n_1n_2}^{s\nu,c_1c_2} \cup \Omega_{n_1n_2}^{\nu s,c_1c_2}$. Since $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ is non-empty (otherwise *AS1-AS4* could never be satisfied at all), this implies that within the Source Region Of Support the noise signals have a 'simpler temporal structure' than the source signals in the sense that $\Omega_{n_1n_2}^{s\cap\nu,c_1c_2}$ is smaller than $\Omega_{n_1n_2}^{s,c_1c_2}$. Note that outside $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ the noise correlation functions are allowed to be non-zero. As has been said earlier in this section and is also clear from intuition, from the various regions of support the a priori knowledge of $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ only is sufficient for, and essential to, our approach. In essence, the role of *AS3* and *AS4* is to specify this knowledge precisely. Given a suitable Source ROS $\Omega_{n_1n_2}^{s,c_1c_2}$, these two assumptions provide general guidelines for choosing $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$, viz. $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ should be chosen such that the noise, source-noise, and noise-source correlation functions have no support on it.

For the sake of overview and convenient referencing, we summarize the assumptions made above both in words and in mathematical form in the following list:

AS1: The source signals are realizations of zero-mean real- or complex-valued random processes with zero cross-correlation functions on the Noise-Free ROS:
$r_{j_1j_2}^{s,c_1c_2}[n_1,n_2] = 0 \quad \forall\,(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}, \quad \forall\,1 \le j_1 \ne j_2 \le S\,;$

AS2: The source auto-correlation functions are linearly independent on the Noise-Free ROS:
$\sum_{j=1}^{S} \xi^j\, r_{jj}^{s,c_1c_2}[n_1,n_2] = 0 \quad \forall\,(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2} \quad \Longrightarrow \quad \xi^j = 0\;\forall\,1 \le j \le S\,;$

AS3: The noise signals are realizations of zero-mean real- or complex-valued random processes with zero auto- and cross-correlation functions on the Noise-Free ROS:
$r_{i_1i_2}^{\nu,c_1c_2}[n_1,n_2] = 0 \quad \forall\,(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}, \quad \forall\,1 \le i_1,i_2 \le D\,;$

AS4: The cross-correlation functions between the source and noise signals are zero on the Noise-Free ROS:
$r_{ij}^{\nu s,c_1c_2}[n_1,n_2] = r_{ji}^{s\nu,c_1c_2}[n_1,n_2] = 0 \quad \forall\,(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}\,,$
$\forall\,1 \le i \le D, \quad \forall\,1 \le j \le S\,.$

5.2.2 Example for scenario with real-valued system, stationary AR(1) source signals, and additive white noise

In this section we apply the framework for describing the model assumptions to an example with AR(1) source signals. For the sake of visualization and insight, we assume a real-valued system and real-valued stationary signals, and adopt assumptions on the source and noise signals that are similar to those in Section 4.2. In particular, we generalize assumptions *AS1-AS4* listed on page 138 to *AS1-AS4* listed on the facing page respectively, i.e. we formulate the generalizations of (4.2.5)-(4.2.17) to (5.2.6)-(5.2.11) respectively as they pertain to the current scenario. Since the system and signals are real-valued the conjugation pair (c_1, c_2) is irrelevant and will be omitted from the notation or is replaced by the number 2 representing the employed statistical order. In addition, because all signals are assumed to be stationary all involved correlation functions depend on a lag k only. Therefore, as in Section 4.2 the various regions of support can also be chosen in the k-domain instead of in the domain of time index pairs (n_1, n_2), and we will write Ω_k^s, Ω_k^ν and $\Omega_k^{s\backslash\nu}$ instead of $\Omega_{n_1n_2}^s$, $\Omega_{n_1n_2}^\nu$ and $\Omega_{n_1n_2}^{s\backslash\nu}$, respectively. See Section 5.4.2 for an example in which both the non-stationarity and non-whiteness of speech source signals are exploited by using the sensor correlation values for different times and different lags.

5.2.2.1 Signal scenario

Firstly, (4.2.5) concerning the temporal cross-structure of the source signals generalizes to:

$$r_{j_1j_2}^s[k] = r_{j_2j_1}^s[-k] = 0 \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq j_1 \neq j_2 \leq S\,. \tag{5.2.12}$$

Secondly, (4.2.6) concerning the temporal auto-structure of the source signals becomes:

$$s_j[n] = \rho_j s_j[n-1] + w_j[n] \qquad \forall\, 1 \leq j \leq S\,, \tag{5.2.13}$$

where $\rho_1, \ldots, \rho_S$ are AR(1) regression coefficients and $w_j[n]$ is a sequence of independent identically distributed (i.i.d.) random variables with some probability density function. As is evident from (4.2.7), the source auto-correlation functions are given by:

$$r_j^s[k] = (\rho_j)^{|k|} \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \leq j \leq S\,. \tag{5.2.14}$$

In the sequel of this example, we will often use $S = 4$ for demonstrating the various concepts. For $S = 4$ the source auto-correlation functions $r_1^s[k]$, $r_2^s[k]$, $r_3^s[k]$, and $r_4^s[k]$ with $\rho_1 = 0.9$, $\rho_2 = 0.8$, $\rho_3 = 0.6$, and $\rho_4 = 0.3$, respectively, are depicted (among other things) in Fig. 5.3 for lags $-4, \ldots, 4$. The function values are indicated by the black circles and the black lines with different kinds of dashing for each function. Note that due to the symmetry of the auto-correlation functions, the values for negative lags do not provide extra information w.r.t. the values for the positive lags, and vice versa. From the fact that the source signals $s_1[n], \ldots, s_4[n]$ are generated using different AR(1) coefficients, it follows that the corresponding auto-correlation functions are linearly independent in the following sense:

$$\sum_{j=1}^{4} \xi^j\, r_j^s[k] = 0 \qquad \forall\, k \in \mathbb{Z} \quad \Longrightarrow \quad \xi^1 = \xi^2 = \xi^3 = \xi^4 = 0\,, \tag{5.2.15}$$

where $\xi^1, \ldots, \xi^4$ are scalars. In fact, as we will see this holds on any ROS in the k-domain

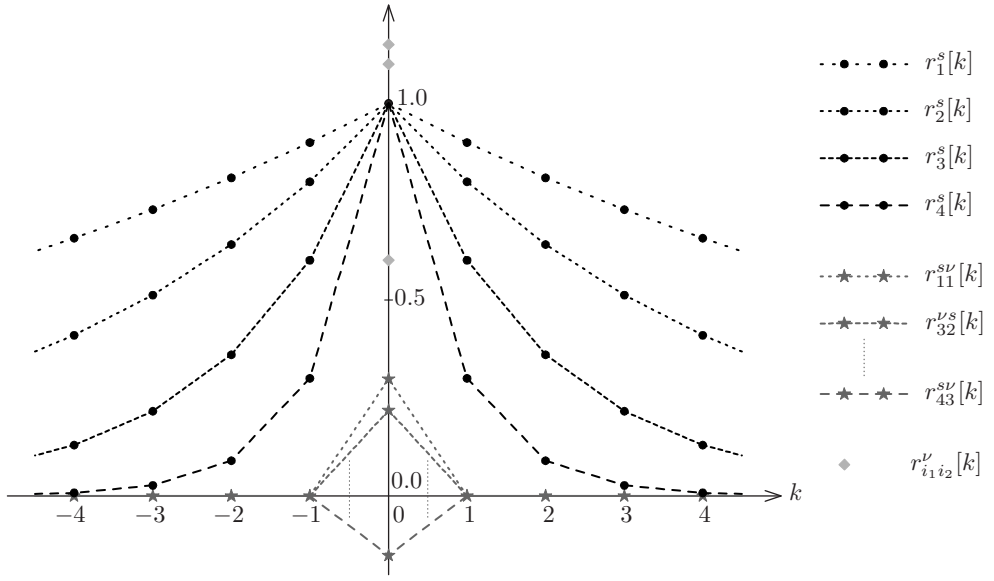

Figure 5.3: *Correlation functions and regions of support for scenario with four AR(1) source signals and additive white noise.*

that contains at least four different consecutive lags, where 'different' means that the corresponding correlation values are not automatically the same, e.g. lags -1 and 1 are considered to be the same. It suffices to consider the first four non-zero different lags $1, \ldots, 4$. The corresponding values for each auto-correlation function $r_j^s[k]$ are stacked in a source auto-correlation row vector $\tilde{\mathbf{r}}_j^s$ as follows: $\tilde{\mathbf{r}}_j^s \triangleq \begin{bmatrix} r_{jj}^s[1] & r_{jj}^s[2] & r_{jj}^s[3] & r_{jj}^s[4] \end{bmatrix}$. Hence, as we have explained before each source correlation function is represented by a source correlation row vector. These vectors are stacked on top of each other in the source correlation matrix $\mathbf{C}_S^{s,\circ\circ} = \mathbf{C}_4^{s,2}$ as follows:

$$\mathbf{C}_4^{s,2} \triangleq \begin{bmatrix} \tilde{\mathbf{r}}_1^s \\ \tilde{\mathbf{r}}_2^s \\ \tilde{\mathbf{r}}_3^s \\ \tilde{\mathbf{r}}_4^s \end{bmatrix} = \begin{bmatrix} r_1^s[1] & r_1^s[2] & r_1^s[3] & r_1^s[4] \\ r_2^s[1] & r_2^s[2] & r_2^s[3] & r_2^s[4] \\ r_3^s[1] & r_3^s[2] & r_3^s[3] & r_3^s[4] \\ r_4^s[1] & r_4^s[2] & r_4^s[3] & r_4^s[4] \end{bmatrix} = \begin{bmatrix} \rho_1 & (\rho_1)^2 & (\rho_1)^3 & (\rho_1)^4 \\ \rho_2 & (\rho_2)^2 & (\rho_2)^3 & (\rho_2)^4 \\ \rho_3 & (\rho_3)^2 & (\rho_3)^3 & (\rho_3)^4 \\ \rho_4 & (\rho_4)^2 & (\rho_4)^3 & (\rho_4)^4 \end{bmatrix}. \qquad (5.2.16)$$

The source auto-correlation functions are linearly independent if and only if $\operatorname{rank}\left(\mathbf{C}_4^{s,2}\right) = 4$, i.e. when $\mathbf{C}_4^{s,2}$ has full rank[‡]. In case all coefficients $\rho_1, \ldots, \rho_4$ are non-zero and different, we can prove this assertion by showing that the only solution to the equation $\mathbf{C}_4^{s,2}\boldsymbol{\beta} = \mathbf{0}$, where $\boldsymbol{\beta}$ is the column vector $\boldsymbol{\beta} = \begin{bmatrix} \beta_0 & \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^T$, is given by $\boldsymbol{\beta} = \mathbf{0}$. In order to do so, note that the system $\mathbf{C}_4^{s,2}\boldsymbol{\beta} = \mathbf{0}$ is equivalent to:

$$\beta_0 + \beta_1\rho + \beta_2(\rho)^2 + \beta_3(\rho)^3 = 0 \qquad \text{for} \quad \rho = \rho_j \quad \forall\, 1 \leq j \leq 4\,.$$

Since the left hand side of this equation is a polynomial in ρ of degree 3, it cannot have four different zeroes $\rho_1, \ldots, \rho_4$ for $\boldsymbol{\beta} \neq \mathbf{0}$. Hence, it can be concluded that $\mathbf{C}_4^{s,2}$ has full rank.

[‡]In fact, it is the *Vandermonde* structure of $\mathbf{C}_S^{s,2}$ that implies its full rankness [147].

Consequently, *AS2* on page 222 is satisfied whenever the Noise-Free ROS $\Omega_k^{s\backslash\nu}$ contains at least four or more different consecutive lags. For example, in the noise-free case $\Omega_k^{s\backslash\nu}$ could be chosen as $\{0,1,2,3\}$, $\{-3,-2,-1,0\}$, $\{-3,-2,-1,0,1,2,3\}$, $\{-4,-3,-2,-1,0,1,2,3,4\}$, $\{-3,-1,0,2\}$, etc., and in the white noise case as $\{1,2,3,4\}$, $\{-1,-2,-3,-4\}$, $\{2,3,4,5\}$, $\{-3,-2,0,1,4\}$, $\{-4,-3,-2,-1,1,2,3,4\}$, and so on. Similarly, for $S=3$ three or more different consecutive lags are sufficient to guarantee that $\mathbf{C}_4^{s,2}$ has full rank. Recall that due to the symmetry of the source auto-correlation functions, the values for negative lags do not provide extra information w.r.t. the values for the positive lags, and vice versa.

Assumption (4.2.10) concerning the temporal structure of the noise signals becomes:

$$r_{i_1 i_2}^{\nu}[k] = r_{i_1 i_2}^{\nu}[-k] = r_{i_1 i_2}^{\nu}[0]\,\delta[k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \le i_1, i_2 \le D\,, \tag{5.2.17}$$

i.e. the D noise signals are assumed to be temporally white and are allowed to be spatially cross-correlated only for lag zero. Note that the variance of the i-th noise signal is given by $r_{ii}^{\nu}[0] \triangleq (\sigma_\nu^i)^2$. A few of the noise auto- and cross-correlation functions $\{r_{i_1 i_2}^{\nu}[k]\}_{1\le i_1,i_2\le D}$ are indicated in Fig. 5.3 by the light grey diamonds ◆. Note that for lags -4, -3, -2, -1, 1, 2, 3, 4, the diamonds coincide with the dark grey stars ★ (indicating the source-noise cross-correlation functions, see further) and therefore are invisible in the figure. Clearly, the assumption formulated in (5.2.17) leads to the following intuitive definition $\Omega_k^{\nu} \triangleq \{0\}$ for the Noise ROS, which implies that lag 0 is excluded from the Noise-Free ROS $\Omega_k^{s\backslash\nu}$. Note that the noise cross-correlation values $\{r_{i_1 i_2}^{\nu}[0]\}_{1\le i_1,i_2\le D}$ for lag zero that are depicted on the vertical axis may have arbitrary values. A spatially correlated noise vector process with the properties described above, can be thought of as being generated by the following simple model: $\boldsymbol{\nu}[n] = \boldsymbol{\Xi}\,\boldsymbol{\xi}[n]$. Here, $\boldsymbol{\Xi}$ is an arbitrary real-valued matrix of size D by K, where K is arbitrary and $\boldsymbol{\xi}[n]$ is a sequence of uncorrelated length-K column vectors with uncorrelated elements.

Finally, assumption (4.2.14) concerning the cross-correlation between the source and noise signals becomes:

$$r_{ij}^{\nu s}[k] = r_{ji}^{s\nu}[-k] = r_{ij}^{\nu s}[k]\,\delta[k] \qquad \forall\, k \in \mathbb{Z}, \quad \forall\, 1 \le i \le D,\ 1 \le j \le S\,, \tag{5.2.18}$$

i.e. it is assumed that all cross-correlation functions of the source and the noise signals are non-zero only for lag zero. The fact that the cross-correlations of the source and the noise signals are allowed to be non-zero for lag zero may seem artificial, but it is intended to demonstrate the rationale for eliminating noise by defining a Noise-Free ROS $\Omega_k^{s\backslash\nu}$ that does not include lag zero. In any case, the situation where the cross-correlation functions of the noise and source signals are assumed to be zero for all lags is a stronger assumption and consequently can be considered as a subcase of the general assumption. A few of the source-noise cross-correlation functions $\{r_{ij}^{\nu s}[k]\}_{1\le i\le D,1\le j\le S}$ are indicated in Fig. 5.3 by the dark grey stars ★. In case the cross-correlation functions of the noise and source signals are assumed to be zero, the constants $\{r_{ij}^{\nu s}[0]\}_{1\le i\le D,1\le j\le S}$ in (5.2.18) are zero. An example of this is when the vector process $\boldsymbol{\nu}[n]$ is generated independently from $\mathbf{w}[n]$ consisting of the components $w_j[n]$ in (5.2.13).

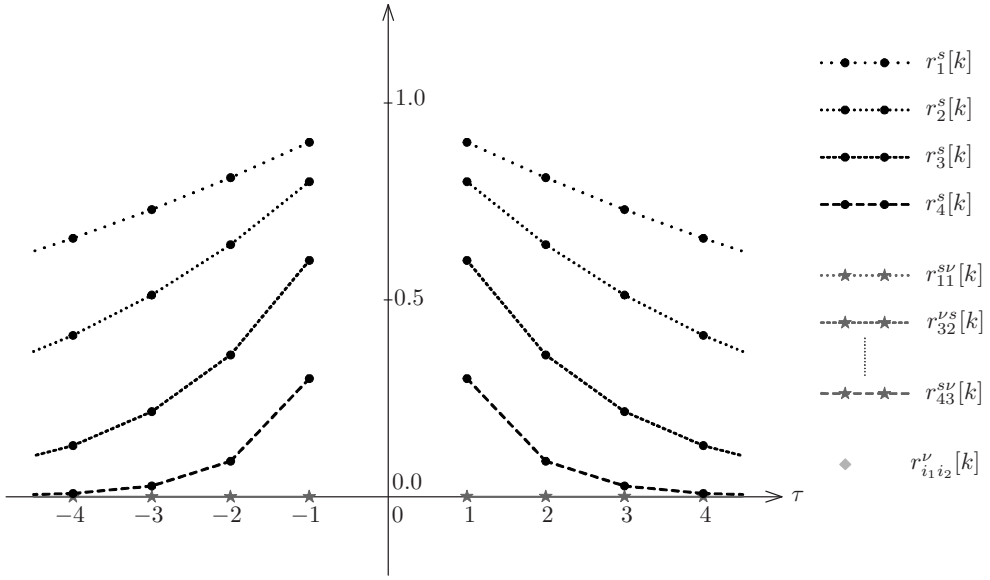

Figure 5.4: *Correlation functions on Noise-Free ROS for scenario with four AR(1) source signals and additive white noise.*

5.2.2.2 Regions Of Support

The assumptions worked out in the previous section lead to natural choices for the various Regions of Support defined in Section 5.2. Firstly, due to (5.2.14) and (5.2.15) the Source ROS is given by:

$$\Omega_k^s \triangleq \mathbb{Z}\,. \tag{5.2.19}$$

Secondly, due to (5.2.17) the Noise ROS equals:

$$\Omega_k^\nu \triangleq \{0\}\,. \tag{5.2.20}$$

Likewise, due to (5.2.18) natural choices for the Source-Noise and Noise-Source Regions Of Support are given by:

$$\Omega_k^{s\nu} = \Omega_k^{\nu s} \triangleq \{0\}\,. \tag{5.2.21}$$

Finally, using (4.2.16) or (5.2.5) the Noise-Free Region Of Support can be deduced from (5.2.19)-(5.2.21) as follows:

$$\Omega_k^{s\setminus\nu} \triangleq \Omega_k^s \setminus \Omega_k^{s\cap\nu} = \Omega_k^s \setminus \left(\Omega_k^s \cap \left(\Omega_k^\nu \cup \Omega_k^{s\nu} \cup \Omega_k^{\nu s}\right)\right) = \{k \in \mathbb{Z} \mid k \neq 0\}\,. \tag{5.2.22}$$

The 'clean' source, noise, and source-noise correlation functions defined on this Noise-Free ROS are depicted in Fig. 5.4. Note that in theory for a scenario of the kind discussed in this section it suffices to choose:

$$\Omega_k^{s\setminus\nu} = \{1, \ldots, S\}\,. \tag{5.2.23}$$

5.3 Formulating MIBI as the problem of solving a system of homogeneous polyconjugal equations

Using the assumptions made in the previous section, in this section we will show along the same lines as in Section 4.3 that the array response vectors, i.e. the columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of $\mathbf{A}$, satisfy a well-structured system of D-variate homogeneous *polyconjugal* equations of degree two, thereby 'projecting' the MIBI problem onto a mathematical problem, the solution of which yields estimates of the array response vectors. Again, in the course of our derivation we highlight the algebraic structure of the problem formulation.

In Section 5.3.1, we start by examining the kind and structure of the sensor statistics that are available for analysis. This primarily amounts to expressing the sensor correlation functions in terms of the mixing matrix elements and the source, noise, and joint source-noise correlation functions, and to studying their properties. Our goal is to derive the system of polyconjugal equations referred to above from the sensor correlation functions in such a way that the sensor noise is annihilated. This can always be achieved by using only those values of the sensor correlation functions that are noise-free, i.e. by considering all correlation functions on the Noise-Free ROS $\Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}$ only. The noise-free sensor correlation values form the input for the rest of our argument, in which we arrange and consider them in a specific configuration. In Section 5.3.2, we generalize the notation for denoting sets of correlation functions and index pairs explained in Section 4.3.2 to the current scenario. Likewise, in Section 5.3.3 we generalize the results obtained in Section 4.3.3, i.e. we determine a set containing all unique sensor correlation functions. As in the previous chapter, the derivation of the system of equations is formulated in terms of correlation functions, in terms of the associated row vector notation, and in terms of the associated matrix-vector notation in Sections 5.3.4, 5.3.5 and 5.3.6 respectively. For each formulation, our main argument can be divided into three stages. Firstly, the sensor correlation values are arranged in a specific configuration that allows to exploit the relation between the various involved linear subspaces on the one hand, and the elements of the mixing matrix on the other. Secondly, we derive the form of the system of polyconjugal equations satisfied by the columns of $\mathbf{A}$ by studying the properties of the linear (sub)spaces spanned by the (abstract) vectors occurring in the previous stage. Finally, we show how subspace decomposition using the Singular Value Decomposition (SVD) can be used to obtain proper coefficients for the polyconjugal functions in the system. This stage will complete the derivation of the system of equations. In Section 5.5 a homotopy method for solving the system is discussed.

5.3.1 Structure of sensor correlation functions

Similarly to Section 4.3.1, we start our derivation by expressing the sensor correlation functions in terms of the mixing matrix elements and the source auto-correlation functions. The structure revealed this way will be used in the following sections for developing our subspace based identification method along the lines of Section 4.3.1. We start by giving the sensor correlation function definition that is consistent with definitions (5.2.1)-(5.2.4):

$$r_{i_1 i_2}^{x, c_1 c_2}[n_1, n_2] \triangleq \operatorname{mom}\Big(\big(x_{i_1}[n_1]\big)^{c_1}, \big(x_{i_2}[n_2]\big)^{c_2} \Big) = E\Big\{ \big(x_{i_1}[n_1]\big)^{c_1} \big(x_{i_2}[n_2]\big)^{c_2} \Big\}$$
$$\forall\, (n_1, n_2) \in \mathbb{Z}_2, \quad \forall\, 1 \leq i_1, i_2 \leq D, \quad \forall\, (c_1, c_2) \in \mathcal{C}_2\,. \tag{5.3.1}$$

This definition can be used for stationary as well as for non- or quasi-stationary signals. We consider the general case of nonstationary signals. If stationary signals are considered, the time index pair (n_1, n_2) can simply be replaced by $(n, n-k)$ resulting in functions that are dependent on the single lag-variable. From now on, without any loss of generality we assume that a specific conjugation pair (c_1, c_2) has been chosen that is fixed subsequently. Hence, the theory that will now be developed holds for any arbitrary conjugation tuple $(c_1, c_2) \in \mathcal{C}_2$.

For all indices $1 \leq i_1, i_2 \leq D$, time index pairs $(n_1, n_2) \in \mathbb{Z}_2$, and for all conjugation pairs $(c_1, c_2) \in \mathcal{C}_2$, the sensor correlation function $r_{i_1 i_2}^{x,c_1 c_2}[n_1, n_2]$ of the i_1-th and i_2-th sensor signals can be written as follows:

$$
\begin{aligned}
r_{i_1 i_2}^{x,c_1 c_2}[n_1, n_2] &\overset{(5.3.1)}{=} E\Big\{ \big(x_{i_1}[n_1]\big)^{c_1} \big(x_{i_2}[n_2]\big)^{c_2} \Big\} \\
&\overset{(5.1.1)}{=} E\Bigg\{ \Bigg(\sum_{j_1=1}^{S} a_{i_1}^{j_1} s_{j_1}[n_1] + \nu_{i_1}[n_1] \Bigg)^{c_1} \Bigg(\sum_{j_2=1}^{S} a_{i_2}^{j_2} s_{j_2}[n_2] + \nu_{i_2}[n_2] \Bigg)^{c_2} \Bigg\} \\
&= \sum_{j_1=1}^{S} \sum_{j_2=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} \big(a_{i_2}^{j_2}\big)^{c_2} E\Big\{ \big(s_{j_1}[n_1]\big)^{c_1} \big(s_{j_2}[n_2]\big)^{c_2} \Big\} \\
&\quad + \sum_{j_1=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} E\Big\{ \big(s_{j_1}[n_1]\big)^{c_1} \big(\nu_{i_2}[n_2]\big)^{c_2} \Big\} \\
&\quad + \sum_{j_2=1}^{S} \big(a_{i_2}^{j_2}\big)^{c_2} E\Big\{ \big(\nu_{i_1}[n_1]\big)^{c_1} \big(s_{j_2}[n_2]\big)^{c_2} \Big\} \; + \; E\Big\{ \big(\nu_{i_1}[n_1]\big)^{c_1} \big(\nu_{i_2}[n_2]\big)^{c_2} \Big\} \\
&= \sum_{j_1=1}^{S} \sum_{j_2=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} \big(a_{i_2}^{j_2}\big)^{c_2} r_{j_1 j_2}^{s,c_1 c_2}[n_1, n_2] + \sum_{j_1=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} r_{j_1 i_2}^{s\nu,c_1 c_2}[n_1, n_2] \\
&\quad + \sum_{j_2=1}^{S} \big(a_{i_2}^{j_2}\big)^{c_2} r_{i_1 j_2}^{\nu s,c_1 c_2}[n_1, n_2] + r_{i_1 i_2}^{\nu,c_1 c_2}[n_1, n_2] \,. \qquad (5.3.2)
\end{aligned}
$$

Note the similarity of this derivation to (4.3.1). Using *AS1* and *AS4* the result in (5.3.2) can be simplified to:

$$
\begin{aligned}
& r_{i_1 i_2}^{x,c_1 c_2}[n_1, n_2] = \sum_{j=1}^{S} \big(a_{i_1}^{j}\big)^{c_1} \big(a_{i_2}^{j}\big)^{c_2} r_{jj}^{s,c_1 c_2}[n_1, n_2] + r_{i_1 i_2}^{\nu,c_1 c_2}[n_1, n_2] \\
& \forall\, (n_1, n_2) \in \mathbb{Z}_2, \quad \forall\, 1 \leq i_1, i_2 \leq D \,. \qquad (5.3.3)
\end{aligned}
$$

Again, note the similarity of this equation to (4.3.2). From *AS3* it is clear that similarly to (4.3.3) on the Noise-Free ROS $\Omega_{n_1 n_2}^{s \setminus \nu}$ the sensor correlation functions can be expressed in terms of the mixing matrix elements and source auto-correlation functions as follows:

$$
\boxed{
\begin{aligned}
& r_{i_1 i_2}^{x,c_1 c_2}[n_1, n_2] = \sum_{j=1}^{S} \big(a_{i_1}^{j}\big)^{c_1} \big(a_{i_2}^{j}\big)^{c_2} r_{jj}^{s,c_1 c_2}[n_1, n_2] \\
& \forall\, (n_1, n_2) \in \Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}, \quad \forall\, 1 \leq i_1, i_2 \leq D \,.
\end{aligned}
} \qquad (5.3.4)
$$

5.3.2 Notation for (sets of) index pairs, correlation functions, etcetera

Similarly to Section 4.3.2 of the previous chapter, for the sake of convenience of the following developments, the smooth generalization of various results, and uniformity and compactness of notation, we now first discuss several symbols that will be used extensively in the sequel for denoting various quantities in a compact and intuitive form, for denoting sets of correlation functions and index pairs as well as their associated cardinalities, and so on. We have already encountered some specific instances of these symbols earlier, for example, see Sections 2.7.3, 2.7.4, and 3.3. Some issues that we will discuss in this section are generalizations of those discussed in Section 4.3.2, whereas some other notational issues pertaining to the current more general scenario are also discussed. The most generic form of the notation described in this section is discussed at length in Appendix A. We start by discussing a symbolic notation required for compactly and uniformly formulating our results.

5.3.2.1 Notation for compact formulation of results

According to (A.2.1), a pair of indices like (i_1, i_2) is denoted by $\mathbf{i}_2$, i.e. $\mathbf{i}_2 \triangleq (i_1, i_2)$. If the indices in a pair are equal we use a shorthand notation such as $(i)_2 \triangleq (i, i)$ or $(j)_2 \triangleq (j, j)$. Likewise, a pair (n_1, n_2) of discrete time indices is denoted by $\mathbf{n}_2$, i.e. $\mathbf{n}_2 \triangleq (n_1, n_2)$, and a conjugation pair by $\mathbf{c}_2 \triangleq (c_1, c_2)$. Summarizing, we use the following shorthand notations:

$$\mathbf{i}_2 \triangleq (i_1, i_2), \quad (i)_2 \triangleq (i, i), \quad \mathbf{n}_2 \triangleq (n_1, n_2), \quad \mathbf{c}_2 \triangleq (c_1, c_2), \quad \text{etc.} \tag{5.3.5}$$

As we have remarked already in Section 4.3.2, if a pair of indices is given explicitly in the sub- or superscript position of a symbol, the parentheses and commas are omitted. Similarly, if a time index pair is given explicitly in the argument position of a correlation function, the parentheses are omitted. As an example illustrating the two points that we have discussed so far in this section, consider both the compact and elaborate symbolic notations for a sensor correlation function with conjugation pair $\mathbf{c}_2 = (c_1, c_2)$ that is indexed by index pair $\mathbf{i}_2 = (i_1, i_2)$ and has time index pair $\mathbf{n}_2 = (n_1, n_2)$ as its argument:

$$r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] = r_{(i_1,i_2)}^{x,(c_1,c_2)}[(n_1, n_2)] = r_{i_1 i_2}^{x,c_1 c_2}[n_1, n_2]\,. \tag{5.3.6}$$

Likewise, a source auto-correlation function can be written as follows:

$$r_{(j)_2}^{s,\mathbf{c}_2}[\mathbf{n}_2] = r_{jj}^{s,\mathbf{c}_2}[\mathbf{n}_2] = r_{(j,j)}^{s,(c_1,c_2)}[(n_1, n_2)] = r_{jj}^{s,c_1 c_2}[n_1, n_2]\,. \tag{5.3.7}$$

Note also that Regions Of Support can now also be written more compactly, for example $\Omega_{\mathbf{n}_2}^{s,\mathbf{c}_2} \equiv \Omega_{n_1 n_2}^{s,c_1 c_2}$, $\Omega_{\mathbf{n}_2}^{\nu,\mathbf{c}_2} \equiv \Omega_{n_1 n_2}^{\nu,c_1 c_2}$, $\Omega_{\mathbf{n}_2}^{s\setminus\nu,\mathbf{c}_2} \equiv \Omega_{n_1 n_2}^{s\setminus\nu,c_1 c_2}$, and so on. Finally, we also define the following equivalence for compactly writing auto-correlation functions:

$$r_{(j)_2}^{v,\mathbf{c}_2}[\mathbf{n}_2] \;\equiv\; r_j^{v,\mathbf{c}_2}[\mathbf{n}_2] \qquad \text{or} \qquad r_{jj}^{v,c_1 c_2}[n_1, n_2] \;\equiv\; r_j^{v,c_1 c_2}[n_1, n_2] \tag{5.3.8}$$

for an arbitrary signal $v[n]$. On many occasions in this thesis we will encounter products of the form $\left(a_{i_1}^{j_1}\right)^{c_1}\left(a_{i_2}^{j_2}\right)^{c_2}$ with $(c_1, c_2) \in \mathcal{C}_2$, as in (5.3.2) for example. As is explained in Section A.2, such a product is denoted as follows:

$$\check{a}_{\mathbf{i}_2}^{\mathbf{j}_2,\mathbf{c}_2} = \check{a}_{i_1 i_2}^{j_1 j_2, c_1 c_2} \triangleq \left(a_{i_1}^{j_1}\right)^{c_1}\left(a_{i_2}^{j_2}\right)^{c_2}. \tag{5.3.9}$$

If the indices in $\mathbf{j}_2$ are equal, as in (5.3.4) for example, this becomes:

$$\check{a}_{\mathbf{i}_2}^{(j)_2,\mathbf{c}_2} = \check{a}_{i_1 i_2}^{(j)_2, c_1 c_2} = \check{a}_{i_1 i_2}^{jj, c_1 c_2} \triangleq \left(a_{i_1}^{j}\right)^{c_1}\left(a_{i_2}^{j}\right)^{c_2} \triangleq \check{a}_{i_1 i_2}^{j, c_1 c_2} = \check{a}_{\mathbf{i}_2}^{j,\mathbf{c}_2}\,. \tag{5.3.10}$$

Note that here we have defined the following equivalence (compare this to (5.3.8)):

$$\check{a}_{\mathbf{i}_2}^{(j)_2,\mathbf{c}_2} \equiv \check{a}_{\mathbf{i}_2}^{j,\mathbf{c}_2} \qquad \text{or} \qquad \check{a}_{i_1 i_2}^{(j)_2, c_1 c_2} = \check{a}_{i_1 i_2}^{jj, c_1 c_2} \;\equiv\; \check{a}_{i_1 i_2}^{j, c_1 c_2}\,.$$

5.3.2.2 Notation for sets of index pairs and cardinalities

Now we generalize the notation for sets of index pairs and their associated cardinalities discussed in Section 4.3.2 to the current scenario; see also Appendix A. The set containing all pairs of indices $\mathbf{i}_2 \triangleq (i_1, i_2)$ obtained by varying both i_1 and i_2 from 1 till D is denoted by:

$$\mathcal{I}^2_{\mathrm{t},D} = \{(i_1, i_2) \mid 1 \leq i_1, i_2 \leq D\}\,. \tag{5.3.11}$$

This is a special case of (A.2.14); see also Fig. A.1 on page 408. Again, the number 2 in the superscript position of $\mathcal{I}^2_{\mathrm{t},D}$ indicates that we are considering two different indices, viz. i_1 and i_2, whereas the number D in the subscript position indicates that both indices i_1 and i_2 can range from 1 till D. Finally, the first subscript index letter 't' stands for 'total' because the considered set contains all pairs obtained by varying the indices from 1 till D without any further constraints. The cardinality of $\mathcal{I}^2_{\mathrm{t},D}$ is denoted and given by:

$$M^2_{\mathrm{t},D} \triangleq \left|\mathcal{I}^2_{\mathrm{t},D}\right| = (D)^2\,. \tag{5.3.12}$$

As an example of this notation, consider the set $\mathcal{J}^2_{\mathrm{t},3}$ with index symbol j and $D = 3$:

$$\begin{aligned}\mathcal{J}^2_{\mathrm{t},3} &\triangleq \{(j_1, j_2) \mid 1 \leq j_1, j_2 \leq 3\} \\ &= \{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(3,1),(3,2),(3,3)\}\,,\end{aligned}$$

the cardinality $\left|\mathcal{J}^2_{\mathrm{t},3}\right|$ of which is given correctly by (5.3.12) as $M^2_{\mathrm{t},3} = (3)^2 = 9$. For an example with $D = 2$ see (4.3.4) and (4.3.5) in Section 4.3.2.

The set containing only those pairs of $\mathcal{I}^2_{\mathrm{t},D}$ that are different if the pairs are considered as sets (which are unordered, see Section A.3), is the same as that obtained by varying i_1 and i_2 from 1 till D in ascending order, and is denoted by:

$$\mathcal{I}^2_{\mathrm{a},D} = \{(i_1, i_2) \mid 1 \leq i_1 \leq i_2 \leq D\}\,. \tag{5.3.13}$$

This is a special case of (A.2.15). The numbers D and 2 in the sub- and superscript positions respectively have the same meaning as for $\mathcal{I}^2_{\mathrm{t},D}$, whereas the letter 'a' indicates that this set is of the 'ascending type' because the indices in each pair are ordered in ascending order. The cardinality of $\mathcal{I}^2_{\mathrm{a},D}$ is denoted and given by:

$$M^2_{\mathrm{a},D} \triangleq \left|\mathcal{I}^2_{\mathrm{a},D}\right| \overset{\text{(A.2.15)}}{=} \tfrac{1}{2}D(D+1)\,. \tag{5.3.14}$$

As an example of this notation, consider the set $\mathcal{J}^2_{\mathrm{a},3}$ with index symbol j and $D = 3$:

$$\mathcal{J}^2_{\mathrm{a},3} \triangleq \{(j_1, j_2) \mid 1 \leq j_1 \leq j_2 \leq 3\} = \{(1,1),(1,2),(1,3),(2,2),(2,3),(3,3)\}\,,$$

the cardinality $\left|\mathcal{J}^2_{\mathrm{a},3}\right|$ of which is given correctly by (5.3.14) as $M^2_{\mathrm{a},3} = 6$. For an example with $D = 2$ see (4.3.6) and (4.3.7) in Section 4.3.2. In general, as we have also remarked in Section 4.3.2, each part of a notation like $\mathcal{I}^p_{\mathrm{t},G}$, $\mathcal{I}^p_{\mathrm{a},G}$, or $\mathcal{I}^p_{\mathrm{d},G}$ has a specific unambiguous meaning that is explained in Appendix A and illustrated in Fig. A.1 on page 408.

5.3.2.3 Notation for sets of correlation functions and cardinalities

Now we will generalize the notation for denoting sets of correlation functions and their associated cardinalities discussed in Section 4.3.2 to the current scenario. The set containing all sensor correlation functions with conjugation pair (c_1, c_2) is denoted by:

$$\mathcal{K}^{x,c_1c_2}_{\mathrm{t},D} \triangleq \left\{r^{x,c_1c_2}_{i_1i_2}[n_1,n_2] \mid 1 \leq i_1, i_2 \leq D\right\} = \left\{r^{x,c_1c_2}_{i_1i_2}[n_1,n_2]\right\}_{(i_1,i_2)\in\mathcal{I}^2_{\mathrm{t},D}}, \tag{5.3.15}$$

which can be written compactly as:

$$\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_2} \triangleq \left\{ r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] \mid \mathbf{i}_2 \in \mathcal{I}_{\mathrm{t},D}^2 \right\} = \left\{ r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] \right\}_{\mathbf{i}_2 \in \mathcal{I}_{\mathrm{t},D}^2} . \tag{5.3.16}$$

It is understood implicitly that the functions in this set are defined on the appropriate Noise-Free ROS, which is $\Omega_{n_1 n_2}^{s\setminus\nu,c_1c_2}$ for the current scenario. As an example, with $D = 2$, $\mathbf{c}_2 = (\circ,\circ)$ and Noise-Free ROS $\Omega_k^{s\setminus\nu}$ the set $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_2}$ equals $\mathcal{K}_{\mathrm{t},2}^{x,2}$ defined in (4.3.8):

$$\mathcal{K}_{\mathrm{t},2}^{x,\circ\circ} = \left\{ r_{i_1 i_2}^{x}[k] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},2}^2} = \mathcal{K}_{\mathrm{t},2}^{x,2} .$$

Here we also see an example of a general convention adopted in this work, viz. that if all elements of a conjugation pair or tuple denote 'no conjugation', the conjugation pair may also be replaced by its length. The cardinality of $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_2}$ is denoted by $M_{\mathrm{t},D}^{x,\mathbf{c}_2}$ and given by:

$$M_{\mathrm{t},D}^{x,\mathbf{c}_2} \triangleq \left|\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_2}\right| = \left|\mathcal{I}_{\mathrm{t},D}^2\right| \overset{(5.3.12)}{=} M_{\mathrm{t},D}^2 = (D)^2 . \tag{5.3.17}$$

For reasons that will become clear soon, for the moment we do not consider sets of the 'ascending form' $\mathcal{K}_{\mathrm{a},D}^{x,c_1c_2}$ with arbitrary conjugation pair (c_1,c_2), but only with $(c_1,c_2) = (\circ,\circ)$, i.e. we consider the set of sensor correlation functions indexed by $\mathcal{I}_{\mathrm{a},D}^2$ (5.3.13):

$$\mathcal{K}_{\mathrm{a},D}^{x,\circ\circ} \equiv \mathcal{K}_{\mathrm{a},D}^{x,2} \triangleq \left\{ r_{i_1 i_2}^{x}[n_1,n_2] \mid 1 \le i_1 \le i_2 \le D \right\} = \left\{ r_{i_1 i_2}^{x}[n_1,n_2] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},D}^2} , \tag{5.3.18}$$

which can be written compactly as:

$$\mathcal{K}_{\mathrm{a},D}^{x,2} \triangleq \left\{ r_{\mathbf{i}_2}^{x}[\mathbf{n}_2] \mid \mathbf{i}_2 \in \mathcal{I}_{\mathrm{a},D}^2 \right\} = \left\{ r_{\mathbf{i}_2}^{x}[\mathbf{n}_2] \right\}_{\mathbf{i}_2\in\mathcal{I}_{\mathrm{a},D}^2} . \tag{5.3.19}$$

For example, with $D = 2$ and Noise-Free ROS $\Omega_k^{s\setminus\nu}$ the set $\mathcal{K}_{\mathrm{a},D}^{x,\circ\circ}$ equals $\mathcal{K}_{\mathrm{a},2}^{x,2}$ defined in (4.3.10):

$$\mathcal{K}_{\mathrm{a},D}^{x,\circ\circ} = \left\{ r_{i_1 i_2}^{x}[k] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{a},2}^2} = \mathcal{K}_{\mathrm{a},2}^{x,2} .$$

The cardinality of $\mathcal{K}_{\mathrm{a},D}^{x,2}$ is denoted by $M_{\mathrm{a},D}^{x,2}$ and given by:

$$M_{\mathrm{a},D}^{x,2} \triangleq \left|\mathcal{K}_{\mathrm{a},D}^{x,2}\right| = \left|\mathcal{I}_{\mathrm{a},D}^2\right| \overset{(5.3.14)}{=} M_{\mathrm{a},D}^2 = \tfrac{1}{2}D(D+1) . \tag{5.3.20}$$

The set containing all unique sensor correlation functions, i.e. the *uniquiefied* version of $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ defined in (5.3.15), is denoted by:

$$\boxed{\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2} \triangleq \left\{ r_{i_1 i_2}^{x,c_1c_2}[n_1,n_2] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} ,} \tag{5.3.21}$$

which can be written compactly as:

$$\boxed{\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2} \triangleq \left\{ r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] \mid \mathbf{i}_2 \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2} \right\} = \left\{ r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] \right\}_{\mathbf{i}_2\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2}} .} \tag{5.3.22}$$

Hence, by definition $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ is indexed by $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2} = \mathcal{I}_{\mathrm{u},D}^{c_1c_2}$. Similarly to the role of $\mathcal{K}_{\mathrm{u},2}^{x,2}$ (4.3.16) in Chapter 4, $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ plays a role of paramount importance in this chapter. For this reason, we explain its derivation in detail in the next section. The cardinality of $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2}$, i.e. the number of unique sensor correlation functions for conjugation pair $\mathbf{c}_2$, is denoted by:

$$\boxed{M_{\mathrm{u},D}^{x,\mathbf{c}_2} = M_{\mathrm{u},D}^{x,c_1c_2} \triangleq \left|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\right| .} \tag{5.3.23}$$

The set containing all source auto-correlation functions is denoted by:

$$\mathcal{K}_S^{s,c_1c_2} \triangleq \left\{ r_{jj}^{s,c_1c_2}[n_1,n_2] \mid 1 \le j \le S \right\} = \left\{ r_{jj}^{s,c_1c_2}[n_1,n_2] \right\}_{1\le j\le S}, \tag{5.3.24}$$

which can be written compactly as:

$$\mathcal{K}_S^{s,\mathbf{c}_2} \triangleq \left\{ r_{jj}^{s,\mathbf{c}_2}[\mathbf{n}_2] \mid 1 \le j \le S \right\} = \left\{ r_{jj}^{s,\mathbf{c}_2}[\mathbf{n}_2] \right\}_{1\le j\le S}. \tag{5.3.25}$$

As an example, for $D=2$, $\mathbf{c}_2=(\circ,\circ)$ and Noise-Free ROS $\Omega_k^{s\backslash\nu}$ the set $\mathcal{K}_S^{s,\mathbf{c}_2}$ equals $\mathcal{K}_2^{s,2}$ defined in (4.3.12):

$$\mathcal{K}_S^{s,\circ\circ} = \mathcal{K}_2^{s,\circ\circ} = \left\{ r_{jj}^{s}[k] \mid 1 \le j \le 2 \right\} = \mathcal{K}_2^{s,2}.$$

With these definitions in place, our main result in (5.3.4) can be formulated in compact notation as follows:

$$\boxed{r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2] = \sum_{j=1}^{S} \breve{a}_{\mathbf{i}_2}^{j,\mathbf{c}_2}\, r_j^{s,\mathbf{c}_2}[\mathbf{n}_2] \qquad \forall\, \mathbf{n}_2 \in \Omega_{\mathbf{n}_2}^{s\backslash\nu,\mathbf{c}_2}, \quad \forall\, \mathbf{i}_2 \in \mathcal{I}_{\mathrm{t},D}^2\,.} \tag{5.3.26}$$

Note the similarity to (3.3.8) in Chapter 3 which formulates a similar relation for fourth order cumulants without using temporal information.

5.3.3 Determining set of unique sensor correlation functions

In this section we determine the sets $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2} = \mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ and $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2} = \mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ defined by (5.3.21) and (5.3.22). Similarly to Section 4.3.3, in order to do so we need to study the possible symmetries in the set $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ (5.3.15) containing all sensor correlation functions. More precisely, given a specific conjugation pair $(c_1,c_2) \in \mathcal{C}_2$, we have to determine for which index pairs $(i_1,i_2) \in \mathcal{I}_{\mathrm{t},D}^2$ the following generalization of (4.3.13) holds:

$$r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] = r_{i_2i_1}^{x,c_1c_2}[n_1,n_2]\,. \tag{5.3.27}$$

As can be deduced from (5.3.4), this implies that we have to determine which products in the set $\{\breve{a}_{i_1i_2}^{j,c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$ associated with $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ are always the same. In other words, we have to find out for which index pairs $(i_1,i_2) \in \mathcal{I}_{\mathrm{t},D}^2$ the following type of symmetry is present in the products of mixing matrix elements:

$$\left(a_{i_1}^j\right)^{c_1}\left(a_{i_2}^j\right)^{c_2} = \left(a_{i_2}^j\right)^{c_1}\left(a_{i_1}^j\right)^{c_2}. \tag{5.3.28}$$

Since this problem is more complicated than the corresponding one in the previous chapter, and because we need to generalize the results in the next chapter, we will now examine in detail how a unique set of products associated with $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ and indexed by the corresponding index set $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ can be obtained in a systematic way.

To start with, for each fixed index $1 \le j \le S$ and conjugation pair $(c_1,c_2) \in \mathcal{C}_2$, define the Kronecker product vector $\breve{\mathbf{a}}_D^{j,c_1c_2}$ with conjugation pair (c_1,c_2) constructed from a single length-D vector $\mathbf{a}^j$ and containing all product terms $\{\breve{a}_{i_1i_2}^{j,c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$ of the form specified

in (5.3.10) as follows (compare this definition with (4.3.68)):

$$\check{\mathbf{a}}_D^{j,c_1c_2} \triangleq \left(\mathbf{a}^j\right)^{c_1} \otimes \left(\mathbf{a}^j\right)^{c_2} = \begin{bmatrix} \left(a_1^j\right)^{c_1} \\ \vdots \\ \left(a_D^j\right)^{c_1} \end{bmatrix} \otimes \begin{bmatrix} \left(a_1^j\right)^{c_2} \\ \vdots \\ \left(a_D^j\right)^{c_2} \end{bmatrix} = \begin{bmatrix} \check{a}_{11}^{j,c_1c_2} \\ \vdots \\ \check{a}_{1D}^{j,c_1c_2} \\ \vdots \\ \check{a}_{D1}^{j,c_1c_2} \\ \vdots \\ \check{a}_{DD}^{j,c_1c_2} \end{bmatrix} = \begin{bmatrix} \left(a_1^j\right)^{c_1}\left(a_1^j\right)^{c_2} \\ \vdots \\ \left(a_1^j\right)^{c_1}\left(a_D^j\right)^{c_2} \\ \vdots \\ \left(a_D^j\right)^{c_1}\left(a_1^j\right)^{c_2} \\ \vdots \\ \left(a_D^j\right)^{c_1}\left(a_D^j\right)^{c_2} \end{bmatrix}. \tag{5.3.29}$$

As we have explained above, in order to determine which product term definitions in the set $\{\check{a}_{i_1i_2}^{j,c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$ are unique, and thus which elements of $\check{\mathbf{a}}_D^{j,c_1c_2}$ in (5.3.29) are unique, we need to study the symmetries in the product terms. Since for a fixed conjugation tuple the set $\{\check{a}_{i_1i_2}^{j,c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$ has the same structure for each $1 \leq j \leq S$, it is sufficient to consider a single set with the same structure, e.g. a set $\{\check{z}_{i_1i_2}^{c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2} = \{(z_{i_1})^{c_1}(z_{i_2})^{c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$ containing all products of the form $\check{z}_{i_1i_2}^{c_1c_2} \triangleq (z_{i_1})^{c_1}(z_{i_2})^{c_2}$. Equivalently, since the Kronecker vectors $\{\check{\mathbf{a}}_D^{j,c_1c_2}\}^{1\leq j\leq S}$ constructed from the columns of $\mathbf{A}$ all have the same form, it is sufficient to consider a single vector $\check{\mathbf{z}}_D^{c_1c_2} \equiv \mathbf{w}_D^{c_1c_2}(\mathbf{z}) = (\mathbf{z})^{c_1} \otimes (\mathbf{z})^{c_2}$ with this form that contains all products in $\{\check{z}_{i_1i_2}^{c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$. Without any loss of generality, here we consider vectors of this type for studying the symmetries of the considered product terms. Hence, let $\mathbf{z}$ (or $\mathbf{z}_D$) be a column vector of length D, then the vector $\check{\mathbf{z}}_D^{c_1c_2} \triangleq \mathbf{w}_D^{c_1c_2}(\mathbf{z})$ is defined similarly to (5.3.29) and (4.3.105):

$$\check{\mathbf{z}}_D^{c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{z}) \triangleq (\mathbf{z})^{c_1} \otimes (\mathbf{z})^{c_2} = \begin{bmatrix} \left(z_1\right)^{c_1} \\ \vdots \\ \left(z_D\right)^{c_1} \end{bmatrix} \otimes \begin{bmatrix} \left(z_1\right)^{c_2} \\ \vdots \\ \left(z_D\right)^{c_2} \end{bmatrix} = \begin{bmatrix} \check{z}_{11}^{c_1c_2} \\ \vdots \\ \check{z}_{1D}^{c_1c_2} \\ \vdots \\ \check{z}_{D1}^{c_1c_2} \\ \vdots \\ \check{z}_{DD}^{c_1c_2} \end{bmatrix} = \begin{bmatrix} \left(z_1\right)^{c_1}\left(z_1\right)^{c_2} \\ \vdots \\ \left(z_1\right)^{c_1}\left(z_D\right)^{c_2} \\ \vdots \\ \left(z_D\right)^{c_1}\left(z_1\right)^{c_2} \\ \vdots \\ \left(z_D\right)^{c_1}\left(z_D\right)^{c_2} \end{bmatrix}. \tag{5.3.30}$$

Our task now is to construct and count the unique elements of $\check{\mathbf{z}}_D^{c_1c_2}$. The uniquiefied version of $\check{\mathbf{z}}_D^{c_1c_2} \triangleq \mathbf{w}_D^{c_1c_2}(\mathbf{z})$ will be denoted by the vector $\check{\mathbf{z}}_{\mathrm{u},D}^{c_1c_2} \triangleq \mathbf{w}_{\mathrm{u},D}^{c_1c_2}(\mathbf{z})$. Clearly, the results depend on the length D of $\mathbf{z}$, as well as on the conjugation pair (c_1, c_2). As will we see in the sequel, in as far as the information provided by the conjugation pair is concerned, the number $M_{\mathrm{u},D}^{x,c_1c_2}$ of unique products is determined only by the number of conjugations, or, equivalently, by the number of 'no conjugations'. Hence, $M_{\mathrm{u},D}^{x,c_1c_2}$ does not depend on which particular elements are conjugated or not.

In the next two sections, we will examine the symmetries for the scenarios without and with conjugations respectively.

5.3.3.1 Symmetries for scenario without conjugations

For the scenario without conjugations the conjugation pair is given by $\mathbf{c}_2 = (\circ, \circ) = (\circ)_2$; see Chapter 4, for example. In this case, the Kronecker product vector $\check{\mathbf{z}}_D^{c_1 c_2} = \check{\mathbf{z}}_D^{\circ\circ} = \check{\mathbf{z}}_D^{(\circ)_2} \triangleq \check{\mathbf{z}}_D^2$ in (5.3.30) reduces to the conventional second order Kronecker product $\mathbf{z} \otimes \mathbf{z}$ of the column $\mathbf{z}$ with itself. Because of the following symmetry:

$$\check{z}_{i_1 i_2}^{\circ\circ} = \check{z}_{i_2 i_1}^{\circ\circ} \qquad \equiv \qquad z_{i_1} z_{i_2} = z_{i_2} z_{i_1} \qquad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{t},D}^2\,, \tag{5.3.31}$$

the auto-product terms $(z_1)^2, \ldots, (z_D)^2$ are unique, whereas each cross-product term $z_{i_1} z_{i_2}$ with $i_1 \neq i_2$ occurs twice. Since there are $(D)^2$ combinations of $1 \leq i_1, i_2 \leq D$, from which D pairs correspond to auto-product terms, there are $(D)^2 - D = D(D-1)$ cross-product terms. Since the latter ones occur twice, there are $\frac{1}{2}D(D-1)$ unique cross-product terms and thus $D + \frac{1}{2}D(D-1) = \frac{1}{2}D(D+1)$ unique product terms in total. These unique terms can be indexed by the index set $\mathcal{I}_{\mathrm{u},D}^{\circ\circ} = \mathcal{I}_{\mathrm{u},D}^2$ equalling $\mathcal{I}_{\mathrm{a},D}^2$ defined in (5.3.13). The cardinality of (5.3.13) is given by (A.2.15) and (5.3.14). Hence, the number of unique product terms in the vector $\check{\mathbf{z}}_D^{\circ\circ} = \check{\mathbf{z}}_D^{(\circ)_2}$, which equals the cardinality of the set $\mathcal{K}_{\mathrm{u},D}^{x,\circ\circ}$, is given by:

$$M_{\mathrm{u},D}^{x,\circ\circ} \triangleq M_{\mathrm{u},D}^{x,2} = \left|\mathcal{K}_{\mathrm{u},D}^{x,\circ\circ}\right| = M_{\mathrm{a},D}^{x,2} = \binom{2+D-1}{2} = \frac{1}{2}D(D+1)\,. \tag{5.3.32}$$

As an example, for $D = 2$ the vector $\check{\mathbf{z}}_D^{c_1 c_2}$ in (5.3.30) becomes:

$$\check{\mathbf{z}}_2^{\circ\circ} = \check{\mathbf{w}}_2^{\circ\circ}(\mathbf{z}) \triangleq \mathbf{z} \otimes \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \otimes \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_2 z_1 \\ z_2 z_2 \end{bmatrix},$$

which has the following uniquiefied version:

$$\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ} = \mathbf{w}_{\mathrm{u},2}^{\circ\circ}(\mathbf{z}) = \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_2 z_2 \end{bmatrix}. \tag{5.3.33}$$

Recall that the 'u' in the subscript position stands for 'unique'. Note that the number of elements of $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ}$ is given correctly as 3 by (5.3.32). As another example, for $D = 3$ the vector $\check{\mathbf{z}}_D^{c_1 c_2}$ in (5.3.30) and its uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},D}^{\circ\circ}$ become:

$$\check{\mathbf{z}}_3^{\circ\circ} = \mathbf{w}_3^{\circ\circ}(\mathbf{z}) \triangleq \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \otimes \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_1 z_3 \\ z_2 z_1 \\ z_2 z_2 \\ z_2 z_3 \\ z_3 z_1 \\ z_3 z_2 \\ z_3 z_3 \end{bmatrix} \quad \text{and} \quad \check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ} = \mathbf{w}_{\mathrm{u},3}^{\circ\circ}(\mathbf{z}) = \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_1 z_3 \\ z_2 z_2 \\ z_2 z_3 \\ z_3 z_3 \end{bmatrix} \tag{5.3.34}$$

respectively. Again, the number of elements of $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ}$ is given correctly as 6 by (5.3.32). Note that the products also satisfy the following relation:

$$\left(z_{i_1} z_{i_2}\right)^* = (z_{i_1})^* (z_{i_2})^* \qquad \equiv \qquad \left(\check{z}_{i_1 i_2}^{\circ\circ}\right)^* = \check{z}_{i_1 i_2}^{**} \qquad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{t},D}^2\,, \tag{5.3.35}$$

which shows that $\check{\mathbf{z}}_D^{**} = (\check{\mathbf{z}}_D^{\circ\circ})^*$ has the same number of unique products as $\check{\mathbf{z}}_D^{\circ\circ}$, i.e. $M_{\mathrm{u},D}^{x,**} = M_{\mathrm{u},D}^{x,\circ\circ} = \frac{1}{2}D(D+1)$ given by (5.3.32).

Development of combinatorics for scenario without conjugations

As will have become clear by now, constructing and counting unique product terms in vectors like $\check{\mathbf{z}}_3^{\circ\circ}$ resulting in vectors like $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ}$ in essence is a combinatorial problem. In order to be able to perform this construction and counting task in a structured manner that can also be employed for general scenarios in which conjugations are involved, in the sequel we explain the combinatorial setting that we have developed for solving this problem; see also Section A.3. In Section 5.3.3.2, this setting will be generalized to the scenario with conjugations. To start with, let $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ be a set containing D independent complex variables with cardinality $D \triangleq |\mathcal{Z}|$, and consider the following definition (see also Definition A.3.8):

Definition 5.3.1. *2-**combination with repetition of the set** $\mathcal{Z} = \{z_1, \ldots, z_D\}$ **containing** $D = |\mathcal{Z}|$ **independent complex variables.*** A 2-combination with repetition of the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ containing $D = |\mathcal{Z}|$ independent complex variables is an arbitrary unordered selection of 2 elements from this set, where the elements may be selected repeatedly. In other words, a 2-combination with repetition of the set $\mathcal{Z}$ is an arbitrary 2-multiset of $\mathcal{Z}$ [‡].

The problem of constructing and counting the unique elements of Kronecker product vectors like $\check{\mathbf{z}}_D^{\circ\circ} = \mathbf{z} \otimes \mathbf{z}$ *amounts to constructing and counting all possible unique* 2*-combinations with repetition of* $\mathcal{Z}$ *because each* 2*-combination can be associated bijectively with a product term*. For example, let $\mathcal{Z} = \{z_1, z_2, z_3\}$ be a set containing $D = 3$ complex variables and let us construct and count the unique elements of the vector $\check{\mathbf{z}}_3^{\circ\circ}$ defined in (5.3.34). All 2-combinations with repetition of $\mathcal{Z}$ are given by the set of 2-multisets $\mathcal{M}_{\mathrm{u},3}^{\circ\circ}\langle\mathcal{Z}\rangle \triangleq \{\langle z_1, z_1\rangle, \langle z_1, z_2\rangle, \langle z_1, z_3\rangle, \langle z_2, z_2\rangle, \langle z_2, z_3\rangle, \langle z_3, z_3\rangle\}$ with cardinality given by (A.3.5) as $M_{\mathrm{u},3}^2 = \frac{(2+3-1)!}{2!(3-1)!} = 6$ (we will derive this soon). Since order in multisets and multiplicity in sets[‡]are irrelevant, this set is the same as the 'full set' $\{\langle z_1, z_1\rangle, \langle z_1, z_2\rangle, \langle z_1, z_3\rangle, \langle z_2, z_1\rangle, \langle z_2, z_2\rangle, \langle z_2, z_3\rangle, \langle z_3, z_1\rangle, \langle z_3, z_2\rangle, \langle z_3, z_3\rangle\}$ for example, which also has cardinality 6 because it has only 6 unique elements according to the definition of multisets (for example, elements like $\langle z_1, z_2\rangle$ and $\langle z_2, z_1\rangle$ are considered to be the same because they belong to the same multiset-equivalence class).

Let $\mathcal{M}_{\mathrm{u},D}^{\circ\circ}\langle\mathcal{Z}\rangle$ be the set containing all 2-multisets of some set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ with cardinality D, i.e. it is the set of all 2-combinations with repetition of $\mathcal{Z}$; see Definition 5.3.1. An unambiguous representation $\mathcal{M}_{\mathrm{u},D}^{\circ\circ}(\mathcal{Z})$ of $\mathcal{M}_{\mathrm{u},D}^{\circ\circ}\langle\mathcal{Z}\rangle$ that contains the different 2-multisets of $\mathcal{M}_{\mathrm{u},D}^{\circ\circ}\langle\mathcal{Z}\rangle$ exactly once can be defined for example by the set containing exactly one element from each multiset-equivalence class, which is represented by a 2-tuple (note that the tuple-representation is crucial because order is relevant now). Although it is not necessary, for convenience the elements of $\mathcal{M}_{\mathrm{u},D}^{\circ\circ}(\mathcal{Z})$ may be arranged according to some fixed ordering scheme of the element indices, e.g. ascending order. For the example above this results in the following representation for the set containing all unique 2-combinations with repetition of the set $\mathcal{Z} = \{z_1, z_2, z_3\}$: $\mathcal{M}_{\mathrm{u},3}^{\circ\circ}(\mathcal{Z}) \triangleq \{(z_1, z_1), (z_1, z_2), (z_1, z_3), (z_2, z_2), (z_2, z_3), (z_3, z_3)\}$. Clearly, this set can be indexed by an index set of the type in (5.3.13). Calling the index i for example, the index set associated with $\mathcal{M}_{\mathrm{u},3}^{\circ\circ}(\mathcal{Z})$ is given by: $\mathcal{I}_{\mathrm{u},3}^2 = \{(1,1), (1,2), (1,3), (2,2), (2,3), (3,3)\}$. The product terms associated with the elements of $\mathcal{M}_{\mathrm{u},3}^{\circ\circ}(\mathcal{Z})$ are $z_1z_1, z_1z_2, z_1z_3, z_2z_2, z_2z_3$ and z_3z_3. Clearly, there is a one-to-one correspondence between $\mathcal{M}_{\mathrm{u},3}^{\circ\circ}(\mathcal{Z})$ and the uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ}$ of $\check{\mathbf{z}}_3^{\circ\circ}$ given in (5.3.34). This also holds for the more general scenario with D sensors, i.e.:

[‡]See Section A.3 for the definitions of 2-combination, multiset, and other related combinatorial concepts.

There is a one-to-one correspondence between the set $\mathcal{M}^{\circ\circ}_{\mathrm{u},D}(\mathcal{Z})$ containing all unique 2-multisets of $\mathcal{M}^{\circ\circ}_{\mathrm{u},D}\langle\mathcal{Z}\rangle$ exactly once and the vector $\mathbf{z}^{\circ\circ}_{\mathrm{u},D}$ of unique products.

In general, with Definition 5.3.1 in place we can *formulate the problem of constructing and counting all unique product terms of $\dot{\mathbf{z}}^{\circ\circ}_{D}$ as that of constructing and counting the set $\mathcal{M}^{\circ\circ}_{\mathrm{u},D}(\mathcal{Z})$ containing all unique 2-combinations with repetition of the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$*.

We will now investigate this problem in detail; see also Section A.3. Consider a set $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ containing $D \triangleq |\mathcal{Z}|$ independent variables, from which we are to construct and count all possible 2-multisets, i.e. all 2-combinations with repetition. As we have indicated above, the set containing these 2-combinations is denoted by $\mathcal{M}^{\circ\circ}_{\mathrm{u},D}\langle\mathcal{Z}\rangle$. To start with, consider a specific 2-multiset containing m_1 times element z_1, m_2 times element z_2, and so on, till m_D times element z_D, i.e. m_i is the multiplicity of element z_i. Note that the sum $m_1 + m_2 + \cdots + m_D$ of the multiplicities of $z_1, \ldots, z_D$ respectively equals 2 because each multiset contains exactly 2 elements. Hence, $m_i \in \{0, 1, 2\}$ for all $1 \leq i \leq D$. This 2-multiset can be represented uniquely by the following set:

$$\underbrace{\big\{\{\underbrace{z_1, \ldots, z_1}_{m_1}\}, \{\underbrace{z_2, \ldots, z_2}_{m_2}\}, \ldots, \{\underbrace{z_D, \ldots, z_D}_{m_D}\}\big\}}_{m_1+\cdots+m_D=2 \text{ elements from } \mathcal{Z}} . \tag{5.3.36}$$

In turn this set can be represented uniquely by an ordered sequence of 2 arbitrary symbols denoting the elements of the set, e.g. bullets, and $D-1$ other arbitrary symbols denoting the 'transition to the next element', e.g. vertical bars, as follows:

$$\underbrace{\big(\underbrace{\bullet\cdots\bullet}_{m_1}\overbrace{|\underbrace{\bullet\cdots\bullet}_{m_2}|\cdots|}^{D-1 \text{ bars}}\underbrace{\bullet\cdots\bullet}_{m_D}\big)}_{\substack{m_1+\cdots+m_D=2 \text{ bullets} \\ D+1 \text{ symbols in total}}} . \tag{5.3.37}$$

The number m_1 of bullets to the left of the first bar represents the number of repetitions (i.e. multiplicity) of z_1, the number of bullets m_2 between the first and second bar represents the number of repetitions of z_2, and so on. Finally, the number of bullets m_D to the right of the last bar represents the number of repetitions of z_D. The total number of bullets equals the length of the multiset, i.e. $m_1 + \cdots + m_D = 2$. Since there are D different elements in $\mathcal{Z}$ the number of separating bars, each of which indicates a transition to the next element of the set, equals $D-1$. Hence, the total number of symbols in (5.3.37) including both bullets and bars equals $(m_1 + \cdots + m_D) + (D-1) = 2 + (D-1) = D+1$. Thus, we can conclude that the number of 2-multisets of $\mathcal{Z}$ equals the number of ways to arrange the $D-1$ bars among the $D+1$ symbols in (5.3.37), which equals the number of subsets of size $D-1$ in a set of size $D+1$, which is precisely the binomial coefficient $\binom{D+1}{D-1} = \binom{D+1}{2} = \frac{1}{2}D(D+1)$ that we already found above in (5.3.32). In conclusion, *the number of 2-multisets of a set $\mathcal{Z}$ with cardinality D is the same as the number of 2-subsets of a set of size $D+1$*, which is expressed as follows (see also (A.3.5)):

$$M^{\circ\circ}_{\mathrm{u},D} \triangleq M^{2}_{\mathrm{u},D} \triangleq \left\langle \begin{matrix} D \\ 2 \end{matrix} \right\rangle \triangleq \frac{(D+1)!}{2!(D-1)!} = \binom{D+1}{2} = \binom{D+1}{D-1} = \frac{1}{2}D(D+1) . \tag{5.3.38}$$

Summarizing the results of this section, for the scenario without conjugations the set of unique sensor correlation functions and the corresponding set of index pairs are given by:

$$\mathcal{K}^{x,\circ\circ}_{\mathrm{u},D} \equiv \mathcal{K}^{x,2}_{\mathrm{u},D} = \big\{r^{x}_{i_1 i_2}[n_1,n_2]\big\}_{(i_1,i_2)\in\mathcal{I}^{2}_{\mathrm{u},D}} \overset{(5.3.18)}{=} \mathcal{K}^{x,\circ\circ}_{\mathrm{a},D} \equiv \mathcal{K}^{x,2}_{\mathrm{a},D} \tag{5.3.39}$$

and:

$$\mathcal{I}^{\circ\circ}_{\mathrm{u},D} \equiv \mathcal{I}^{2}_{\mathrm{u},D} = \big\{(i_1,i_2) \mid 1 \le i_1 \le i_2 \le D\big\} \overset{(5.3.13)}{=} \mathcal{I}^{\circ\circ}_{\mathrm{a},D} = \mathcal{I}^{2}_{\mathrm{a},D}$$

respectively. Both sets have cardinality $M^{x,\circ\circ}_{\mathrm{u},D} \triangleq M^{x,2}_{\mathrm{u},D} = M^{2}_{\mathrm{u},D}$ defined in (5.3.38). The following table lists the values of $M^{2}_{\mathrm{u},D} = M^{x,2}_{\mathrm{u},D} = M^{x,**}_{\mathrm{u},D} = |\mathcal{K}^{x,\circ\circ}_{\mathrm{u},D}|$ for $1 \le D \le 5$.

Table 5.1: *Number $M^{2}_{\mathrm{u},D}$ of unique products and sensor correlation functions.*

D	1	2	3	4	5
$M^{2}_{\mathrm{u},D}$	1	3	6	10	15

5.3.3.2 Symmetries for scenario with conjugations

Now we are in a position to consider the more complex case of scenarios with conjugations. First note that the number of unique product terms in the vector $\check{\mathbf{z}}_D^{c_1c_2}$ does not change if the vector is conjugated. Hence, as we have already remarked in the previous section the vector $\check{\mathbf{z}}_D^{\circ\circ}$ has the same number of (unique) product terms as the vector $(\check{\mathbf{z}}_D^{\circ\circ})^* = \check{\mathbf{z}}_D^{**}$; see also (5.3.35) and Section 5.3.3.3. Likewise, $\check{\mathbf{z}}_D^{\circ *} = (\check{\mathbf{z}}_D^{*\circ})^*$ has the same number of (unique) terms as $\check{\mathbf{z}}_D^{*\circ}$. This can easily be verified in the following examples for $D=2$ and $D=3$ respectively:

$$\check{\mathbf{z}}_2^{\circ *} = \mathbf{w}_2^{\circ *}(\mathbf{z}) = \begin{bmatrix} z_1(z_1)^* \\ z_1(z_2)^* \\ z_2(z_1)^* \\ z_2(z_2)^* \end{bmatrix}, \quad \check{\mathbf{z}}_3^{\circ *} = \mathbf{w}_3^{\circ *}(\mathbf{z}) = \begin{bmatrix} z_1(z_1)^* \\ z_1(z_2)^* \\ z_1(z_3)^* \\ z_2(z_1)^* \\ z_2(z_2)^* \\ z_2(z_3)^* \\ z_3(z_1)^* \\ z_3(z_2)^* \\ z_3(z_3)^* \end{bmatrix}. \tag{5.3.40}$$

All product terms in these vectors are unique, and the same holds for $\check{\mathbf{z}}_D^{\circ *}$, because:

$$z_{i_1}(z_{i_2})^* \neq z_{i_2}(z_{i_1})^* \qquad \forall\, 1 \le i_1 \neq i_2 \le D\,, \tag{5.3.41}$$

where $z_1,\ldots,z_D$ have to be considered as independent complex-valued variables/functions. Hence, the Kronecker vector $\check{\mathbf{z}}_D^{\circ *}$ is the same as its uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},D}^{\circ *}$. Summarizing, for the scenario with conjugation pair $(c_1,c_2) = (\circ,*)$ the set of unique sensor correlation functions and the corresponding set of index pairs are given by:

$$\mathcal{K}^{x,\circ *}_{\mathrm{u},D} \equiv \mathcal{K}^{x,\circ *}_{\mathrm{t},D} \triangleq \big\{r^{x,\circ *}_{i_1 i_2}[n_1,n_2]\big\}_{(i_1,i_2)\in\mathcal{I}^{\circ *}_{\mathrm{u},D}} \overset{(5.3.16)}{=} \mathcal{K}^{x,\mathbf{c}_2}_{\mathrm{t},D} \tag{5.3.42}$$

and:

$$\mathcal{I}^{\circ *}_{\mathrm{u},D} \equiv \mathcal{I}^{\circ *}_{\mathrm{t},D} = \big\{(i_1,i_2) \mid 1 \le i_1, i_2 \le D\big\} \overset{(5.3.11)}{=} \mathcal{I}^{2}_{\mathrm{t},D}$$

respectively. Both sets have cardinality $M^{x,\circ *}_{\mathrm{u},D} = M^{x,*\circ}_{\mathrm{u},D} \overset{(5.3.12)}{=} M^{2}_{\mathrm{t},D} = (D)^2$.

In fact, now we have analyzed the scenarios for all possible values of the conjugation pair. However, in order to be able to perform the construction and counting task in a structured manner that can be used for the most general case with an arbitrary conjugation tuple (see the next chapter), we now explain the combinatorial setting that we have developed for solving this problem; see also Sections 5.3.3.1 and A.3. That is, we examine the problem of finding and counting the unique terms in $\check{\mathbf{z}}_D^{c_1c_2}$ for an arbitrary conjugation pair (c_1, c_2) in a manner similar to the one presented at the end of the previous section, thereby constructing its uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},D}^{c_1c_2}$, the set $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ (5.3.21) of unique sensor correlation functions, the associated set $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ of index pairs, and the cardinality $M_{\mathrm{u},D}^{x,c_1c_2}$ of these sets.

Development of combinatorics for scenario with conjugations

To start with, let $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ be a set containing D independent complex variables with cardinality $D \triangleq |\mathcal{Z}|$ and let the set containing the complex conjugates of the elements of $\mathcal{Z}$ be denoted by $\mathcal{Z}^*$, i.e. $\mathcal{Z}^* = \{(z_1)^*, \ldots, (z_D)^*\}$. In addition, let the compound set $\mathcal{Z}^{\circ *}$ be defined as the set containing the elements of both $\mathcal{Z}$ and $\mathcal{Z}^*$:

$$\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\} = \{z_1, \ldots, z_D, (z_1)^*, \ldots, (z_D)^*\}\,. \tag{5.3.43}$$

Denote the number of conjugations in the considered conjugation pair (c_1, c_2) by:

$$n_{c_1c_2} \triangleq \#\{\text{conjugations in } (c_1, c_2)\} = \#_*\{(c_1, c_2)\} \quad \in \quad \{0, 1, 2\}\,, \tag{5.3.44}$$

where $\#\{\cdot\}$ and $\#_*\{\cdot\}$ denote the 'number of' and 'number of conjugations' operators respectively. Since both c_1 and c_2 can either be '$*$' or '$\circ$', there are $(2)^2 = 4$ different conjugation pairs with the following number of conjugations:

$$\#_*\{(\circ, \circ)\} = 0, \quad \#_*\{(\circ, *)\} = 1, \quad \#_*\{(*, \circ)\} = 1, \quad \#_*\{(*, *)\} = 2\,, \tag{5.3.45}$$

i.e. $n_{\circ\circ} = 0$, $n_{\circ *} = n_{*\circ} = 1$ and $n_{**} = 2$. Now, the generalization of Definition 5.3.1 to the current scenario can be given as follows:

Definition 5.3.2. *2-combination with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains $D = |\mathcal{Z}|$ independent complex variables.* A 2-combination with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains D independent complex variables, is an arbitrary unordered selection of 2 elements from this set, where the elements may be selected repeatedly but $2 - n_{c_1c_2}$ elements have to be chosen from $\mathcal{Z}$ and $n_{c_1c_2}$ elements from $\mathcal{Z}^*$. In other words, a 2-combination with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *}$ is an arbitrary 2-multiset of $\mathcal{Z}^{\circ *}$ containing $2 - n_{c_1c_2}$ elements from $\mathcal{Z}$ and $n_{c_1c_2}$ elements from $\mathcal{Z}^*$.

With this definition in place, *the problem of finding and counting the unique terms in the vector $\check{\mathbf{z}}_D^{c_1c_2}$ and/or set $\{\check{z}_{i_1i_2}^{c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{i},D}^2}$ for an arbitrary conjugation pair (c_1, c_2) can be formulated as that of constructing and counting all possible 2-combinations with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains D independent complex variables, $2 - n_{c_1c_2}$ elements have to be chosen from $\mathcal{Z}$, and $n_{c_1c_2}$ elements have to be chosen from $\mathcal{Z}^*$. This can be done because each such 2-combination with conjugation pair (c_1, c_2) can be associated bijectively with a product term.* For example, let $\mathcal{Z} = \{z_1, z_2, z_3\}$ be a set containing $D = 3$ complex variables and let the conjugation pair be given by $(c_1, c_2) \triangleq (\circ, *)$. This scenario corresponds to

the vector $\check{\mathbf{z}}_3^{\circ *}$ defined in (5.3.40). All 2-combinations with conjugation pair $(\circ, *)$ with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\} = \{z_1, z_2, z_3, (z_1)^*, (z_2)^*, (z_3)^*\}$ are given by the set of 2-multisets $\mathcal{M}_{\mathrm{u},3}^{\circ *}\langle\mathcal{Z}\rangle = \{\langle z_1, (z_1)^*\rangle, \langle z_1, (z_2)^*\rangle, \langle z_1, (z_3)^*\rangle, \langle z_2, (z_1)^*\rangle, \langle z_2, (z_2)^*\rangle, \langle z_2, (z_3)^*\rangle, \langle z_3, (z_1)^*\rangle, \langle z_3, (z_2)^*\rangle, \langle z_3, (z_3)^*\rangle\}$ $(= \mathcal{M}_{\mathrm{t},3}^{\circ *}\langle\mathcal{Z}\rangle)$ with cardinality given by (5.3.50) as $(3)^2 = 9$ (we will derive this soon). Since order is irrelevant in multisets, this set is the same as the set $\{\langle (z_1)^*, z_1\rangle, \langle (z_2)^*, z_1\rangle, \langle (z_3)^*, z_1\rangle, \langle (z_1)^*, z_2\rangle, \langle (z_2)^*, z_2\rangle, \langle (z_3)^*, z_2\rangle, \langle (z_1)^*, z_3\rangle, \langle (z_2)^*, z_3\rangle, \langle (z_3)^*, z_3\rangle\}$, for example. The product terms associated with the elements of $\mathcal{M}_{\mathrm{u},3}^{\circ *}\langle\mathcal{Z}\rangle$ are $z_1(z_1)^*$, $z_1(z_2)^*$, $z_1(z_3)^*$, $z_2(z_1)^*$, $z_2(z_2)^*$, $z_2(z_3)^*$, $z_3(z_1)^*$, $z_3(z_2)^*$ and $z_3(z_3)^*$. Note that all different elements of $\mathcal{M}_{\mathrm{u},3}^{\circ *}\langle\mathcal{Z}\rangle$ are unique.

Let $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}\langle\mathcal{Z}\rangle$ be the set containing all 2-multisets of $\mathcal{Z}^{\circ *}$ containing $2 - n_{c_1c_2}$ elements from $\mathcal{Z}$ and $n_{c_1c_2}$ elements from $\mathcal{Z}^*$, i.e. it is the set of all 2-combinations with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains $D = |\mathcal{Z}|$ independent complex variables; see Definition 5.3.2. An unambiguous representation $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}(\mathcal{Z})$ of $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}\langle\mathcal{Z}\rangle$ that contains the different 2-multisets of $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}\langle\mathcal{Z}\rangle$ exactly once can be defined for example by the set containing exactly one element from each multiset-equivalence class, which is represented by a 2-tuple. For the example above this results in the following representation for $\mathcal{M}_{\mathrm{u},3}^{\circ *}\langle\mathcal{Z}\rangle$: $\mathcal{M}_{\mathrm{u},3}^{\circ *}(\mathcal{Z}) = \{(z_1, (z_1)^*), (z_1, (z_2)^*), (z_1, (z_3)^*), (z_2, (z_1)^*), (z_2, (z_2)^*), (z_2, (z_3)^*), (z_3, (z_1)^*), (z_3, (z_2)^*), (z_3, (z_3)^*)\}$. The unique product terms associated with the elements of $\mathcal{M}_{\mathrm{u},3}^{\circ *}(\mathcal{Z})$ are $z_1(z_1)^*$, $z_1(z_2)^*$, $z_1(z_3)^*$, $z_2(z_1)^*$, $z_2(z_2)^*$, $z_2(z_3)^*$, $z_3(z_1)^*$, $z_3(z_2)^*$ and $z_3(z_3)^*$. Clearly, there is a one-to-one correspondence between $\mathcal{M}_{\mathrm{u},3}^{\circ *}(\mathcal{Z})$ and the uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ *}$ of $\check{\mathbf{z}}_3^{\circ *}$ given in (5.3.40). This also holds for the general situation, i.e.:

> There is a one-to-one correspondence between the set $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}(\mathcal{Z})$ containing all unique 2-multisets of $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}\langle\mathcal{Z}\rangle$ exactly once and the vector $\mathbf{z}_{\mathrm{u},D}^{c_1c_2}$ of unique products.

In general, with Definition 5.3.2 in place we can *formulate the problem of constructing and counting all unique terms of $\check{\mathbf{z}}_D^{c_1c_2}$ for an arbitrary conjugation pair (c_1, c_2) as that of constructing and counting the set $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}(\mathcal{Z})$ containing all unique 2-combinations with conjugation pair (c_1, c_2) with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains D independent complex variables, $2 - n_{c_1c_2}$ elements have to be chosen from $\mathcal{Z}$, and $n_{c_1c_2}$ elements have to be chosen from $\mathcal{Z}^*$*.

We will now examine this problem in such a way that the results can be easily generalized to the most general scenario with arbitrary conjugation tuple later on; see Chapter 6. Consider again the set $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ containing $D \triangleq |\mathcal{Z}|$ complex variables, its 'conjugate set' $\mathcal{Z}^* \triangleq \{(z_1)^*, \ldots, (z_D)^*\}$, and the compound set $\mathcal{Z}^{\circ *} = \{z_1, \ldots, z_D, (z_1)^*, \ldots, (z_D)^*\}$. As we have explained above, our task is to construct and count the number of all unique 2-multisets of $\mathcal{Z}^{\circ *}$ containing $2 - n_{c_1c_2}$ elements from $\mathcal{Z}$ and $n_{c_1c_2}$ elements from $\mathcal{Z}^*$. To start with, consider a specific 2-multiset containing m_1 times element z_1, and so on, till m_D times element z_D, m_1^* times element $(z_1)^*$, and so on, till m_D^* times element $(z_D)^*$, i.e. the multiplicities of the i-th element z_i of $\mathcal{Z}$ and the i-th element z_i^* of $\mathcal{Z}^*$ are denoted by m_i and m_i^* respectively. Since each multiset contains $2 - n_{c_1c_2}$ elements from $\mathcal{Z}$, it follows that $m_1 + \cdots + m_D = 2 - n_{c_1c_2}$ with $m_i \in \{0, \ldots, 2 - n_{c_1c_2}\}$ for all $1 \leq i \leq D$. Likewise, since each multiset contains $n_{c_1c_2}$ elements from $\mathcal{Z}^*$, it follows that $m_1^* + \cdots + m_D^* = n_{c_1c_2}$ with $m_i^* \in \{0, \ldots, n_{c_1c_2}\}$ for all $1 \leq i \leq D$. Note that the sum of all multiplicities equals $(m_1 + \cdots + m_D) + (m_1^* + \cdots + m_D^*) = 2$. The 2-multiset can be represented uniquely by

the following set:

$$\underbrace{\Big\{\underbrace{\{\underbrace{z_1,\ldots,z_1}_{m_1}\},\ldots,\{\underbrace{z_D,\ldots,z_D}_{m_D}\}}_{(2-n_{c_1c_2})\text{ elements from }\mathcal{Z}},\ \underbrace{\{\underbrace{(z_1)^*,\ldots,(z_1)^*}_{m_1^*}\},\ldots,\{\underbrace{(z_D)^*,\ldots,(z_D)^*}_{m_D^*}\}}_{n_{c_1c_2}\text{ elements from }\mathcal{Z}^*}\Big\}}_{2\text{ elements from }\mathcal{Z}^{\circ *}=\{\mathcal{Z},\mathcal{Z}^*\}}. \tag{5.3.46}$$

Our problem can be decomposed into two independent problems each of which is similar to the one discussed in the previous section for the situation without conjugation(s). In one of the problems all possible $(2-n_{c_1c_2})$-combinations with repetition of $\mathcal{Z} = \{z_1, \ldots, z_D\}$ have to be constructed and counted, whereas in the other the same has to be done for all possible $n_{c_1c_2}$-combinations with repetition of $\mathcal{Z}^* \triangleq \{(z_1)^*, \ldots, (z_D)^*\}$. Hence, constructing and counting all unique elements of the form (5.3.46) amounts to doing the same for each of the two sets:

$$\underbrace{\Big\{\underbrace{\{\underbrace{z_1,\ldots,z_1}_{m_1},\ldots,\{\underbrace{z_D,\ldots,z_D}_{m_D}\}}_{(2-n_{c_1c_2})\text{ elements from }\mathcal{Z}},\ \underbrace{\{\underbrace{(z_1)^*,\ldots,(z_1)^*}_{m_1^*},\ldots,\{\underbrace{(z_D)^*,\ldots,(z_D)^*}_{m_D^*}\}}_{n_{c_1c_2}\text{ elements from }\mathcal{Z}^*}\Big\}}$$

separately, and then combining the results using the Cartesian product rule. Following the same reasoning as the one below (5.3.36) leading to result (5.3.38), these two sets are represented as:

$$\underbrace{\Big(\underbrace{\bullet\cdots\bullet}_{m_1}\overbrace{|}^{D-1\text{ bars}}\underbrace{\bullet\cdots\bullet}_{m_2}|\cdots|\underbrace{\bullet\cdots\bullet}_{m_D}\Big)}_{\substack{m_1+\cdots+m_D=(2-n_{c_1c_2})\text{ bullets}\\ (D-n_{c_1c_2}+1)\text{ symbols in total}}} \quad\text{and}\quad \underbrace{\Big(\underbrace{\bullet\cdots\bullet}_{m_1^*}\overbrace{|}^{D-1\text{ bars}}\underbrace{\bullet\cdots\bullet}_{m_2^*}|\cdots|\underbrace{\bullet\cdots\bullet}_{m_D^*}\Big)}_{\substack{m_1^*+\cdots+m_D^*=n_{c_1c_2}\text{ bullets}\\ (D+n_{c_1c_2}-1)\text{ symbols in total}}} \tag{5.3.47}$$

respectively; see also (5.3.37). Since there are D different elements in both $\mathcal{Z}$ and $\mathcal{Z}^*$, the number of separating bars equals $D-1$ for each set. Hence, the total number of symbols including bullets and bars equals $(m_1 + \cdots + m_D) + (D-1) = (2 - n_{c_1c_2}) + (D-1) = D - n_{c_1c_2} + 1$ for the first set, and $(m_1^* + \cdots + m_D^*) + (D-1) = n_{c_1c_2} + (D-1) = D + n_{c_1c_2} - 1$ for the second. Thus, the number of $(2 - n_{c_1c_2})$–multisets of $\mathcal{Z}$ equals the number of ways to arrange $D-1$ bars among $D - n_{c_1c_2} + 1$ symbols, which equals the number of subsets of size $D-1$ or $2 - n_{c_1c_2}$ in a set of size $D - n_{c_1c_2} + 1$ and is given by the binomial coefficient as follows (see also (5.3.38) and (A.3.5)):

$$M_{\mathrm{u},D}^{2-n_{c_1c_2}} \triangleq \left\langle \begin{matrix} D \\ 2-n_{c_1c_2} \end{matrix} \right\rangle = \binom{D-n_{c_1c_2}+1}{2-n_{c_1c_2}} = \binom{D-n_{c_1c_2}+1}{D-1}. \tag{5.3.48}$$

Likewise, the number of $n_{c_1c_2}$–multisets of $\mathcal{Z}^*$ equals the number of ways to arrange $D-1$ bars among $D + n_{c_1c_2} - 1$ symbols, which equals the number of subsets of size $D-1$ or $n_{c_1c_2}$ in a set of size $D + n_{c_1c_2} - 1$ and is also given by the binomial coefficient as follows:

$$M_{\mathrm{u},D}^{n_{c_1c_2}} \triangleq \left\langle \begin{matrix} D \\ n_{c_1c_2} \end{matrix} \right\rangle = \binom{D+n_{c_1c_2}-1}{n_{c_1c_2}} = \binom{D+n_{c_1c_2}-1}{D-1}. \tag{5.3.49}$$

Combining (5.3.48) and (5.3.49) using the standard product rule from combinatorics gives the number of 2-multisets of a set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$ containing $2 - n_{c_1c_2}$ elements from $\mathcal{Z}$

(having cardinality D) and $n_{c_1c_2}$ from $\mathcal{Z}^*$ (also having cardinality D) as the product of the number of $(2-n_{c_1c_2})$-subsets of a set of size $D-n_{c_1c_2}+1$ on the one hand, and the number of $n_{c_1c_2}$-subsets of a set of size $D+n_{c_1c_2}-1$ on the other hand:

$$M_{\mathrm{u},D}^{c_1c_2} = \left\langle \begin{matrix} D \\ 2-n_{c_1c_2} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{c_1c_2} \end{matrix} \right\rangle = \binom{D-n_{c_1c_2}+1}{2-n_{c_1c_2}} \binom{D+n_{c_1c_2}-1}{n_{c_1c_2}} . \tag{5.3.50}$$

Hence, the number of unique elements in the vector $\check{\mathbf{z}}_D^{c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{z})$ and/or set $\{\check{z}_{i_1i_2}^{c_1c_2}\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}$, and thus the number of unique sensor correlation functions, can be expressed as follows in various ways:

$$\begin{aligned} M_{\mathrm{u},D}^{c_1c_2} &= M_{\mathrm{u},D}^{2-n_{c_1c_2}} \cdot M_{\mathrm{u},D}^{n_{c_1c_2}} = \left\langle \begin{matrix} D \\ 2-n_{c_1c_2} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{c_1c_2} \end{matrix} \right\rangle \\ &= \binom{D-n_{c_1c_2}+1}{2-n_{c_1c_2}} \binom{D+n_{c_1c_2}-1}{n_{c_1c_2}} = \binom{D-n_{c_1c_2}+1}{D-1} \binom{D+n_{c_1c_2}-1}{D-1} \\ &= \frac{(D-n_{c_1c_2}+1)!(D+n_{c_1c_2}-1)!}{(D-1)!(2-n_{c_1c_2})!(D-1)!(n_{c_1c_2})!} , \end{aligned} \tag{5.3.51}$$

or compactly:

$$\boxed{M_{\mathrm{u},D}^{\mathbf{c}_2} = \left\langle \begin{matrix} D \\ 2-n_{\mathbf{c}_2} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\mathbf{c}_2} \end{matrix} \right\rangle = \binom{D-n_{\mathbf{c}_2}+1}{2-n_{\mathbf{c}_2}} \binom{D+n_{\mathbf{c}_2}-1}{n_{\mathbf{c}_2}} .} \tag{5.3.52}$$

Summarizing the results of this section, using the procedure for constructing $\mathcal{M}_{\mathrm{u},D}^{c_1c_2}(\mathcal{Z})$ explained above and the various one-to-one correspondences, for the scenario with a general conjugation pair (c_1,c_2) we can construct the vector $\check{\mathbf{z}}_{\mathrm{u},D}^{c_1c_2}$ of unique products and the set $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ of unique sensor correlation functions defined in (5.3.21):

$$\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2} = \left\{ r_{i_1i_2}^{x}[n_1,n_2] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^2} \triangleq \left\{ r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] \right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} , \tag{5.3.53}$$

both with associated set $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ of index pairs. The cardinality $M_{\mathrm{u},D}^{c_1c_2} = M_{\mathrm{u},D}^{x,c_1c_2} \triangleq \left|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\right|$ of these sets is given by (5.3.50), (5.3.51) and/or (5.3.52), and its values are listed in the following table for $1 \le D \le 5$.

Table 5.2: *Number $M_{\mathrm{u},D}^{c_1c_2}$ of unique products and sensor correlation functions.*

Conjugation pair parameters		$M_{\mathrm{u},D}^{c_1c_2} = M_{\mathrm{u},D}^{x,c_1c_2}$	Number of sensors D				
(c_1,c_2)	$n_{c_1c_2}$	$\left\langle \begin{matrix} D \\ n_{c_1c_2} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ 2-n_{c_1c_2} \end{matrix} \right\rangle$	1	2	3	4	5
$(\circ,\circ),(*,*)$	0, 2	$\left\langle \begin{matrix} D \\ 0 \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ 2 \end{matrix} \right\rangle$	1	3	6	10	15
$(\circ,*),(*,\circ)$	1, 1	$\left\langle \begin{matrix} D \\ 1 \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ 1 \end{matrix} \right\rangle$	1	4	9	16	25

Examples and properties

We will now provide some examples illustrating the use and correctness of (5.3.50), (5.3.51) and/or (5.3.52). As a first example, consider vectors containing products without conjugations as considered in the previous section, i.e. with $\mathbf{c}_2 = (\circ, \circ)$. Substituting $n_{\circ\circ} = 0$ into one of the equations above yields:

$$M_{\mathrm{u},D}^{\circ\circ} = M_{\mathrm{u},D}^{2} = M_{\mathrm{u},D}^{2-0} \cdot M_{\mathrm{u},D}^{0} = \binom{D-0+1}{2-0}\binom{D+0-1}{0} = \binom{D+1}{2},$$

which equals (5.3.32) and (5.3.38), as expected. For example, the length of $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ}$ (5.3.33) with $D = 2$ equals 3, and the length of $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ}$ (5.3.34) with $D = 3$ equals 6.

Now, consider vectors containing products with conjugation pair $\mathbf{c}_2 = (\circ, *)$. Substituting $n_{\circ *} = 1$ into one of the equations above gives:

$$M_{\mathrm{u},D}^{\circ *} = M_{\mathrm{u},D}^{2-1} \cdot M_{\mathrm{u},D}^{1} = \binom{D-1+1}{2-1}\binom{D+1-1}{1} = \binom{D}{1}\binom{D}{1} = (D)^2 .$$

Hence, as we have determined earlier all $(D)^2$ product terms of $\check{\mathbf{z}}_D^{\circ *}$ are unique due to (5.3.41) and thus $\check{\mathbf{z}}_{\mathrm{u},D}^{\circ *} = \check{\mathbf{z}}_D^{\circ *}$. Using (5.3.40), this result can easily be verified for $\check{\mathbf{z}}_2^{\circ *}$ and $\check{\mathbf{z}}_3^{\circ *}$ with $D = 2$ and $D = 3$ respectively. In conclusion, as we have seen already essentially there are only two different possibilities, viz. $\check{\mathbf{z}}_D^{\circ\circ} \equiv \mathbf{w}_D^{\circ\circ}(\mathbf{z}) = \mathbf{z} \otimes \mathbf{z}$ and its conjugate $\check{\mathbf{z}}_D^{**} \equiv \mathbf{w}_D^{**}(\mathbf{z}) = (\mathbf{z})^* \otimes (\mathbf{z})^*$ both contain $M_{\mathrm{u},D}^{\circ\circ} = M_{\mathrm{u},D}^{**} = \left\langle {D \atop 2} \right\rangle = \frac{1}{2}D(D+1)$ unique product terms, whereas all product terms of $\check{\mathbf{z}}_D^{\circ *} \equiv \mathbf{w}_D^{\circ *}(\mathbf{z}) = \mathbf{z} \otimes (\mathbf{z})^*$ and its conjugate $\check{\mathbf{z}}_D^{*\circ} \equiv \mathbf{w}_D^{*\circ}(\mathbf{z}) = (\mathbf{z})^* \otimes \mathbf{z}$ are unique, i.e. they both contain $M_{\mathrm{u},D}^{\circ *} = M_{\mathrm{u},D}^{*\circ} = \left(\left\langle {D \atop 1} \right\rangle\right)^2 = (D)^2$ terms.

5.3.3.3 Other symmetries

In this section, we briefly note some symmetries that are not relevant for determining the number of unique sensor correlation functions, but nevertheless are useful on other occasions. The first one we have encountered earlier and means that a vector and its conjugate have the same number of unique elements. This can be formulated in a general manner as follows. Denoting the complement of a conjugation pair $\mathbf{c}_2 = (c_1, c_2)$ by $\bar{\mathbf{c}}_2 = (\bar{c}_1, \bar{c}_2)$, which means replacing conjugations by 'no conjugations' and vice versa (see Section A.5 and Eq. A.5.3), we can say that the vector $\check{\mathbf{z}}_D^{\mathbf{c}_2} = \check{\mathbf{z}}_D^{c_1 c_2}$ has the same number of unique product terms as $\check{\mathbf{z}}_D^{\bar{\mathbf{c}}_2} = \check{\mathbf{z}}_D^{\bar{c}_1 \bar{c}_2}$, i.e. $M_{\mathrm{u},D}^{c_1 c_2} = M_{\mathrm{u},D}^{\bar{c}_1 \bar{c}_2}$, and that the terms of $\check{\mathbf{z}}_D^{\bar{\mathbf{c}}_2} = \check{\mathbf{z}}_D^{\bar{c}_1 \bar{c}_2}$ are the conjugates of the corresponding ones in $\check{\mathbf{z}}_D^{\mathbf{c}_2} = \check{\mathbf{z}}_D^{c_1 c_2}$, i.e. $\check{\mathbf{z}}_D^{\bar{c}_1 \bar{c}_2} = (\check{\mathbf{z}}_D^{c_1 c_2})^*$. In general, product terms satisfy the following symmetry:

$$\left((z_{i_1})^{c_1}(z_{i_2})^{c_2}\right)^* = (z_{i_1})^{\bar{c}_1}(z_{i_2})^{\bar{c}_2} \qquad \equiv \qquad \left(\check{z}_{i_1 i_2}^{c_1 c_2}\right)^* = \check{z}_{i_1 i_2}^{\bar{c}_1 \bar{c}_2} \qquad \forall\, (i_1, i_2) \in \mathcal{I}_{\mathrm{t},D}^2 ,$$

which is a generalization of (5.3.35). Note that the cross-products under consideration only commute if $c_1 = c_2$, see (5.3.31) for an example.

The next trivial symmetry property pertains to the source auto-correlation functions:

$$r_{jj}^{s,c_1 c_2}[n_1, n_2] = E\left\{\left(s_j[n_1]\right)^{c_1}\left(s_j[n_2]\right)^{c_2}\right\} = E\left\{\left(s_j[n_2]\right)^{c_2}\left(s_j[n_1]\right)^{c_1}\right\}$$
$$= r_{jj}^{s,c_2 c_1}[n_2, n_1] \qquad \forall\, (n_1, n_2) \in \Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}, \quad \forall\, 1 \le j \le S .$$

Finally, another symmetry property of the source auto-correlation functions is:

$$\left(r_{jj}^{s,c_1 c_2}[n_1, n_2]\right)^* = \left(E\left\{\left(s_j[n_1]\right)^{c_1}\left(s_j[n_2]\right)^{c_2}\right\}\right)^* = E\left\{\left(s_j[n_1]\right)^{\bar{c}_1}\left(s_j[n_2]\right)^{\bar{c}_2}\right\}$$
$$= r_{jj}^{s,\bar{c}_1 \bar{c}_2}[n_1, n_2] \qquad \forall\, (n_1, n_2) \in \Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}, \quad \forall\, 1 \le j \le S .$$

5.3.3.4 Maximum rank of second order Khatri-Rao product with conjugation pair

As has become clear in the previous chapters, the (maximum) rank of the involved Khatri-Rao product of the mixing matrix $\mathbf{A}$ is of paramount importance for determining the number of linearly independent equations in the system of homogeneous polyconjugal equations, and thus also for determining the maximum number $S^{\mathbf{c}_2}_{\max,D}$ of columns and/or sources that can be identified. For example, see Equations (4.3.89) and (4.3.92), and Section 6.6. As we will see in the sequel of this chapter, this also holds for the current scenario. Therefore, we examine this rank here. The second order Khatri-Rao product $\mathbf{A}^{\mathbf{c}_2}_{D,\diamond} = \mathbf{A}^{c_1c_2}_{D,\diamond}$ of $\mathbf{A}$ with conjugation tuple $\mathbf{c}_2 = (c_1, c_2)$ is defined as the matrix formed by stacking the column vectors $\check{\mathbf{a}}^{1,c_1c_2}_D, \ldots, \check{\mathbf{a}}^{S,c_1c_2}_D$ defined in (5.3.29) alongside each other as follows (see also Section A.9, in particular Eq. (A.9.4)):

$$\mathbf{A}^{c_1c_2}_{D,\diamond} \triangleq \left(\mathbf{A}\right)^{c_1} \diamond \left(\mathbf{A}\right)^{c_2} = \left[\left(\mathbf{a}^1\right)^{c_1} \otimes \left(\mathbf{a}^1\right)^{c_2} \;\; \cdots \;\; \left(\mathbf{a}^S\right)^{c_1} \otimes \left(\mathbf{a}^S\right)^{c_2}\right], \tag{5.3.54}$$

which can be written in expanded form as follows:

$$\mathbf{A}^{c_1c_2}_{D,\diamond} = \begin{bmatrix} \check{a}^{11,c_1c_2}_{11} & & \check{a}^{SS,c_1c_2}_{11} \\ \vdots & \cdots & \vdots \\ \check{a}^{11,c_1c_2}_{1D} & & \check{a}^{SS,c_1c_2}_{1D} \\ \vdots & \cdots & \vdots \\ \check{a}^{11,c_1c_2}_{D1} & & \check{a}^{SS,c_1c_2}_{D1} \\ \vdots & \cdots & \vdots \\ \check{a}^{11,c_1c_2}_{DD} & & \check{a}^{SS,c_1c_2}_{DD} \end{bmatrix} = \begin{bmatrix} \left(a^1_1\right)^{c_1}\left(a^1_1\right)^{c_2} & \cdots & \left(a^S_1\right)^{c_1}\left(a^S_1\right)^{c_2} \\ \vdots & \ddots & \vdots \\ \left(a^1_1\right)^{c_1}\left(a^1_D\right)^{c_2} & \cdots & \left(a^S_1\right)^{c_1}\left(a^S_D\right)^{c_2} \\ \vdots & \ddots & \vdots \\ \left(a^1_D\right)^{c_1}\left(a^1_1\right)^{c_2} & \cdots & \left(a^S_D\right)^{c_1}\left(a^S_1\right)^{c_2} \\ \vdots & \ddots & \vdots \\ \left(a^1_D\right)^{c_1}\left(a^1_D\right)^{c_2} & \cdots & \left(a^S_D\right)^{c_1}\left(a^S_D\right)^{c_2} \end{bmatrix}. \tag{5.3.55}$$

From linear algebra [115, 150] it is well known that the rank of a matrix is smaller than or equal to both the number of rows and the number of columns. Hence, $\mathrm{rank}(\mathbf{A}^{c_1c_2}_{D,\diamond})$ must be smaller than or equal to the number of columns S and the number of linearly independent rows of $\mathbf{A}^{c_1c_2}_{D,\diamond}$. In general, if a matrix is 'sufficiently random' (e.g. when it is generated at random) it attains its maximum possible rank as limited by its dimensions. From the derivations in Section 5.3.3.2 it follows that the maximum possible rank of $\mathbf{A}^{c_1c_2}_{D,\diamond}$ as imposed by its row dimension equals $M^{c_1c_2}_{\mathrm{u},D}$ defined in (5.3.50)-(5.3.52). Thus, we have:

$$\mathrm{rank}(\mathbf{A}^{c_1c_2}_{D,\diamond}) \leq \min\left(M^{c_1c_2}_{\mathrm{u},D},\; S\right) = \min\left(\binom{D - n_{c_1c_2} + 1}{2 - n_{c_1c_2}}\binom{D + n_{c_1c_2} - 1}{n_{c_1c_2}},\; S\right). \tag{5.3.56}$$

As we have seen in the previous chapter and will see again in this chapter, in order to be able to identify the columns of $\mathbf{A}$ the number of linearly independent rows of the Khatri-Rao product of $\mathbf{A}$ has to be larger than the number of sources. Therefore, we assume that $M^{c_1c_2}_{\mathrm{u},D} > S$. Hence, for fixed D the maximum possible rank of (a matrix with the same structure as) $\mathbf{A}^{c_1c_2}_{D,\diamond}$ equals the maximum row rank of $\mathbf{A}^{c_1c_2}_{D,\diamond}$, i.e.:

$$\boxed{\max\left(\mathrm{rank}_{\mathrm{row}}\left(\mathbf{A}^{c_1c_2}_{D,\diamond}\right)\right) = M^{c_1c_2}_{\mathrm{u},D} = \binom{D - n_{c_1c_2} + 1}{2 - n_{c_1c_2}}\binom{D + n_{c_1c_2} - 1}{n_{c_1c_2}}.} \tag{5.3.57}$$

In general, if the matrix $\mathbf{A}$ is 'sufficiently random' then the second order Khatri-Rao product $\mathbf{A}^{c_1c_2}_{D,\diamond}$ achieves its maximum possible rank $M^{c_1c_2}_{\mathrm{u},D}$. See Table 5.2 on page 241 for the maximum achievable row rank of $\mathbf{A}^{c_1c_2}_{D,\diamond}$ for several combinations of D and (c_1, c_2).

5.3.4 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; functional notation

Now we are in a position to derive a system of D-variate homogeneous equations of degree two satisfied by the columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of the mixing matrix $\mathbf{A} \in \mathbb{C}_D^S$ along the same lines as in Sections 4.3.4, 4.3.5 and 4.3.6 of Chapter 4. We will see that in case the conjugation pair contains a conjugation the equations are not polynomial anymore but become what we call *'polyconjugal'*. In this section, similarly to Section 4.3.4 we first develop the theory in functional notation, i.e. in terms of functions.

5.3.4.1 Dimensions of signal and source subspaces

Again, we start our derivation by considering the dimension d_D^{x,c_1c_2} of the linear space spanned by the sensor correlation functions:

$$\begin{aligned} d_D^{x,c_1c_2} &\triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^2}\right)\right) \\ &\equiv \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}\right)\right), \end{aligned} \tag{5.3.58}$$

which can be written compactly as:

$$d_D^{x,\mathbf{c}_2} \triangleq \dim\left(\mathcal{L}\left(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2}\right)\right) = \dim\left(\mathcal{L}\left(\left\{r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2]\right\}_{\mathbf{i}_2\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2}}\right)\right). \tag{5.3.59}$$

As in Chapter 4, in analogy with subspace method jargon the linear space $\mathcal{L}\big(\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}\big) = \mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)$ is called *signal subspace*; see also Chapter 3. We also refer to this subspace as *sensor subspace*. Let d_S^{s,c_1c_2} denote the dimension of the linear space spanned by the source auto-correlation functions:

$$d_S^{s,c_1c_2} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,c_1c_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{r_j^{s,c_1c_2}[n_1,n_2]\right\}_{1\le j\le S}\right)\right), \tag{5.3.60}$$

which can be written compactly as:

$$d_S^{s,\mathbf{c}_2} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,\mathbf{c}_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{r_j^{s,\mathbf{c}_2}[\mathbf{n}_2]\right\}_{1\le j\le S}\right)\right). \tag{5.3.61}$$

We will refer to the linear space $\mathcal{L}\big(\mathcal{K}_S^{s,c_1c_2}\big)$ spanned by the source auto-correlation functions as *source subspace*. From *AS2* on page 222, i.e. from the fact that the source auto-correlation functions are linearly independent, it follows directly that the dimension of the source subspace equals:

$$d_S^{s,c_1c_2} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,c_1c_2}\big)\right) = S. \tag{5.3.62}$$

Equation (5.3.4) states that each sensor correlation function $r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]$ is a linear combination of the S source auto-correlation functions $r_1^{s,c_1c_2}[n_1,n_2], \ldots, r_S^{s,c_1c_2}[n_1,n_2]$ with coefficients $\big(a_{i_1}^1\big)^{c_1}\big(a_{i_2}^1\big)^{c_2}, \ldots, \big(a_{i_1}^S\big)^{c_1}\big(a_{i_2}^S\big)^{c_2}$ respectively. Hence, using linear algebra [115, 150] the following 'dimensional relation' between d_D^{x,c_1c_2} and d_S^{s,c_1c_2} can be deduced:

$$d_D^{x,c_1c_2} = \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\right) \le \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,c_1c_2}\big)\right) = d_S^{s,c_1c_2} = S. \tag{5.3.63}$$

This is a logical and intuitive consequence of the fact that all sensor correlation functions are linear combinations of the same S source auto-correlation functions. In principle, depending on the values of the mixing matrix elements, d_D^{x,c_1c_2} can be smaller than d_S^{s,c_1c_2}. However, in general d_D^{x,c_1c_2} will be equal to d_S^{s,c_1c_2} for a randomly generated mixing matrix $\mathbf{A}$. In the sequel, we will show in more detail how d_D^{x,c_1c_2} depends on $\mathbf{A}$ and d_S^{s,c_1c_2}.

5.3.4.2 Linear dependence of sensor correlation functions

Now we will present an essential part of our derivation that is based on a trivial linear algebra result, viz. that in an arbitrary linear vector space of a certain dimension d any set containing more than d vectors is linearly dependent. From this property and (5.3.63) it follows that:

$$\dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) \leq \min\Big(\big|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big|,\ S\Big) = \min\Big(M_{\mathrm{u},D}^{x,c_1c_2},\ S\Big)\ .$$

As we will see in Section 5.3.6, this inequality also follows from (5.3.56) because $\dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big)$ is smaller than or equal to $\mathrm{rank}(\mathbf{A}_{D,\diamond}^{c_1c_2})$. Furthermore, if the number $M_{\mathrm{u},D}^{x,c_1c_2} = |\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}|$ of unique sensor correlation functions given by (5.3.50)-(5.3.52) is larger than the dimension d_D^{x,c_1c_2} of the signal subspace spanned by the functions in $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$, i.e. if:

$$\big|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big| > \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,c_1c_2} > d_D^{x,c_1c_2}\ , \tag{5.3.64}$$

then *the sensor correlation functions in $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ are linearly dependent*. The condition in (5.3.64) can be fulfilled by choosing the right number D of sensors in relation to the number S of sources, where in general 'right' depends on the employed order of the statistics, which equals two in this case, and on the chosen conjugation pattern.

Equivalently to (4.3.22), if the sensor correlation functions in $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ are linearly dependent, i.e. if (5.3.64) is satisfied, then by the definition of linear dependence there exist non-zero and non-unique sets of coefficients $\big\{\varphi_1^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}, \ldots, \big\{\varphi_{Q_{D,S}^{c_1c_2}}^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}$, indexed by an arbitrarily integer-valued index q, such that:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}\ r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] = 0[n_1,n_2] \qquad \forall\ (n_1,n_2)\in\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}, \quad \forall\ q\in\mathcal{Q}_{D,S}^{c_1c_2}\ , \tag{5.3.65}$$

where similarly to (4.3.23) the set $\mathcal{Q}_{D,S}^{c_1c_2}$ is defined as follows:

$$\mathcal{Q}_{D,S}^{c_1c_2} \triangleq \big\{1,\ldots,Q_{D,S}^{c_1c_2}\big\} \tag{5.3.66}$$

with $Q_{D,S}^{c_1c_2} \triangleq |\mathcal{Q}_{D,S}^{c_1c_2}|$ the maximum number of linearly independent equations. This number equals the dimension of the orthogonal complement of the signal subspace:

$$\boxed{Q_{D,S}^{c_1c_2} = \dim\Big(\big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\big)^{\perp}\Big) = \big|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big| - \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) = M_{\mathrm{u},D}^{x,c_1c_2} - d_D^{x,c_1c_2}\ .} \tag{5.3.67}$$

Hence, $Q_{D,S}^{c_1c_2}$ equals the difference between the left and right hand sides of each of the equations in (5.3.64). Note the similarity to (4.3.24), (4.3.54), and (4.3.92). In Section 5.3.8 we will show that a valid set of coefficients of the equations in (5.3.65) can be deduced from the Singular Value Decomposition (SVD) of a function-valued vector associated with the set $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$, or equivalently from the SVD of a properly defined subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. Thus, we can consider the coefficients of the equations as (approximately) known quantities. Similarly to (4.3.26), the set containing the different sets of coefficients $\big\{\varphi_q^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}$ for each value of $q\in\mathcal{Q}_{D,S}^{c_1c_2}$ is denoted by $\Phi_{D,S}^{c_1c_2}$, i.e.:

$$\Phi_{D,S}^{c_1c_2} \triangleq \Big\{\big\{\varphi_q^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}\Big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}\ . \tag{5.3.68}$$

Note that all linear combinations of the sets in $\Phi_{D,S}^{c_1c_2}$ also satisfy (5.3.65). As in Section 4.3.4.2, it should be noted again that completely similar and equivalent equations and results are obtained if we replace $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$ by $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ in the discussion above. Since it is trivial to work out this case along the lines of the remarks made at the end of Section 4.3.4.2, we will not discuss this approach here.

5.3.4.3 Deriving the system of equations by exploiting linear independence of source auto-correlation functions

The next main stage of our derivation amounts to combining (5.3.4) and (5.3.65), and exploiting assumption *AS2* on page 222 once again; this directly leads us to the desired system of equations. Firstly, (5.3.65) is expressed in the source auto-correlation functions by substituting (5.3.4):

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}\, r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(\sum_{j=1}^{S} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} r_j^{s,c_1c_2}[n_1,n_2] \right)$$

$$= \sum_{j=1}^{S} \left(\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} \right) r_j^{s,c_1c_2}[n_1,n_2] = 0[n_1,n_2]$$

$$\forall\, (n_1,n_2) \,\in\, \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}, \quad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,, \tag{5.3.69}$$

Now the essence of assumption *AS2*, viz. the linear independence of the source auto-correlation functions, comes into play. Applying *AS2* to the middle line in (5.3.69) by identifying:

$$\xi_q^j = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} \qquad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,, \quad \forall\, 1 \le j \le S\,,$$

the following system of equations is obtained:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} = 0 \qquad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,, \quad \forall\, 1 \le j \le S\,. \tag{5.3.70}$$

This system describes the relation between the *unknown* coefficients of the mixing matrix $\mathbf{A}$ and the *known* set of coefficient sets $\Phi_{D,S}^{c_1c_2}$ (5.3.68). Let $\mathbf{z} \in \mathbb{C}_D$ be a vector of variables having the same size as a column of $\mathbf{A}$, and define the functions $\{f_{D,q}^{c_1c_2}(\mathbf{z})\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ by:

$$\boxed{f_{D,q}^{c_1c_2}(\mathbf{z}) \triangleq \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} (z_{i_1})^{c_1} (z_{i_2})^{c_2} \qquad \forall\, \mathbf{z} \in \mathbb{C}_D\,, \quad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,.} \tag{5.3.71}$$

Then, (5.3.70) states that all columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of $\mathbf{A}$ satisfy the system of equations $\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$, i.e.:

$$f_{D,q}^{c_1c_2}(\mathbf{a}^j) = 0 \qquad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,, \quad 1 \le j \le S\,. \tag{5.3.72}$$

Hence, at this point *the MIBI problem has been 'projected onto' the problem of solving the following system of equations for the columns of the mixing matrix* $\mathbf{A}$:

$$\boxed{\left\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\right\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}.} \tag{5.3.73}$$

As in the previous chapter, by 'projected' we mean that the system of equations follows from our MIBI problem definition and formulation, but not necessarily vice versa.

5.3.4.4 Properties and structure of functions and system

All functions in system (5.3.73) have the same specific form. Firstly, from the definition in (5.3.71) it is clear that each function $f_{D,q}^{c_1c_2}(\mathbf{z}) = f_{D,q}^{c_1c_2}(z_1,\ldots,z_D)$ is a D-variate 'polynomial-like' function containing product terms of degree two. Strictly speaking, $f_{D,q}^{c_1c_2}(\mathbf{z})$ is only a polynomial in $z_1,\ldots,z_D$ (of degree two and in D variables) if $(c_1,c_2) = (\circ,\circ)$ because for $(c_1,c_2) \in \{(\circ,*),(*,\circ),(*,*)\}$ the product terms (also) contain conjugates of the variables, albeit that for $(c_1,c_2) = (*,*)$ we can consider $f_{D,q}^{c_1c_2}(\mathbf{z})$ as a polynomial of degree two in $(\mathbf{z})^*$, viz. in the variables $(z_1)^*,\ldots,(z_D)^*$. In order to clearly make a distinction between a polynomial in certain variables on the one hand, and a polynomial-like function in the same variables as well as their conjugates on the other, we call the latter type of function *'poly-conjugal'*. Hence, a polynomial is a special case of a polyconjugal in the sense that it only contains unconjugated variables. Now, from (5.3.71) it follows that each function $f_{D,q}^{c_1c_2}(\mathbf{z})$ in (5.3.73) is a D-variate polyconjugal of degree two, which is *homogeneous of degree two with conjugation pair* (c_1,c_2), also called 2-*homogeneous with conjugation pair* (c_1,c_2), meaning that:

$$\boxed{f_{D,q}^{c_1c_2}(\eta\mathbf{z}) = (\eta)^{c_1}(\eta)^{c_2} f_{D,q}^{c_1c_2}(\mathbf{z}) \qquad \forall\, \eta \in \mathbb{C}, \qquad \forall\, \mathbf{z} \in \mathbb{C}_D\,.} \tag{5.3.74}$$

Using (5.3.71) this property can easily be proven as follows:

$$\begin{aligned} f_{D,q}^{c_1c_2}(\eta\mathbf{z}) = f_{D,q}^{c_1c_2}(\eta z_1,\ldots,\eta z_D) &\triangleq \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}\,(\eta z_{i_1})^{c_1}(\eta z_{i_2})^{c_2} \\ &= \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} (\eta)^{c_1}(\eta)^{c_2}\,\varphi_q^{i_1i_2}\,(z_{i_1})^{c_1}(z_{i_2})^{c_2} = (\eta)^{c_1}(\eta)^{c_2} f_{D,q}^{c_1c_2}(\mathbf{z})\,. \qquad \square \end{aligned}$$

Since each function $f_{D,q}^{c_1c_2}(\mathbf{z})$ in system (5.3.73) possesses the type of homogeneity defined by (5.3.74), it has the following nice property:

$$\boxed{f_{D,q}^{c_1c_2}(\mathbf{v}) = 0 \qquad \Longrightarrow \qquad f_{D,q}^{c_1c_2}(\eta\mathbf{v}) = 0 \qquad \forall\, \eta \in \mathbb{C}\,.} \tag{5.3.75}$$

Hence, if $\mathbf{v}$ is a solution of the system $\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$, then so is $\eta\mathbf{v}$ for all $\eta \in \mathbb{C}$. This is a logical result of the scaling indeterminacy inherent to MIBI (see Section 2.4). Note that combining the homogeneity property of all functions in the system yields:

$$f_{D,q}^{c_1c_2}(\mathbf{v}) = 0 \quad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2} \qquad \Longrightarrow \qquad \sum_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} \alpha_p^q\, f_{D,q}^{c_1c_2}(\mathbf{v}) = 0 \quad \forall\, \alpha_p^q \in \mathbb{C}\,, \quad \forall\, p \in \mathbb{N}\,.$$

Hence, applying a full rank linear transformation to system (5.3.73) and solving the resulting system is equivalent to solving (5.3.73) itself. Moreover, a new valid equation is obtained by taking any linear combination of the equations in (5.3.73). This is reminiscent of the remark made just after (5.3.68), viz. that all linear combinations of the sets in $\Phi_{D,S}^{c_1c_2}$ also give proper sets of coefficients. See the discussion after (4.3.37) on page 147 for more information and a mathematical formulation that can readily be adapted to the current scenario. More properties of the system, and the functions and equations it contains, will be investigated in Section 5.4, where we study several aspects of the algebraic and geometric structure of the problem induced by the system of equations.

5.3.5 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; row vector notation

Similarly to Section 4.3.5, in this section we briefly repeat the derivation in the previous section in terms of row vectors as an intuitive intermediate step towards the derivation in terms of a subspace matrix, which is discussed in Section 5.3.6. Again, we will see that it suffices to set up a proper one-to-one mapping between the correlation functions considered in the previous section on the one hand, and corresponding correlation row vectors on the other; see Fig. 1.11 on page 26. We do this by associating a row vector with each correlation function defined on $\Omega_{\mathbf{n}_2}^{s\backslash\nu,\mathbf{c}_2} = \Omega_{n_1 n_2}^{s\backslash\nu,c_1c_2}$. This correspondence will also be used later on for the construction of the subspace matrix.

5.3.5.1 Bijective mapping between correlation functions and correlation row vectors

Suppose that we have specified a Noise-Free ROS $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$ by a set of N time pairs as follows:

$$\begin{aligned}\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2} &= \left\{(n_1,n_2)^1,(n_1,n_2)^2,\ldots,(n_1,n_2)^N\right\}\\ &= \left\{(n_1^1,n_2^1),(n_1^2,n_2^2),\ldots,(n_1^N,n_2^N)\right\},\end{aligned} \tag{5.3.76}$$

or, more compactly:

$$\Omega_{\mathbf{n}_2}^{s\backslash\nu,\mathbf{c}_2} = \left\{\mathbf{n}_2^1,\mathbf{n}_2^2,\ldots,\mathbf{n}_2^N\right\}, \tag{5.3.77}$$

where $\mathbf{n}_2^q = (n_1,n_2)^q = (n_1^q,n_2^q)$ is the q-th time pair of $\Omega_{\mathbf{n}_2}^{s\backslash\nu,\mathbf{c}_2} = \Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$, and n_1^q and n_2^q are the first and second elements of $\mathbf{n}_2^q$ respectively. Now we are in a position to define a bijective mapping between a function belonging to $\mathbb{C}[\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}]$ on the one hand, and a row vector on the other hand, see Sections 1.4 and 4.3.5.1. For each pair $(i_1,i_2) \in \mathcal{I}_{\mathfrak{t},D}^2$, a row vector $\tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2}$ is associated with the sensor correlation function $r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]$ defined on $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$ by defining:

$$\tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2} \triangleq \left[r_{i_1i_2}^{x,c_1c_2}[n_1^1,n_2^1]\;\; r_{i_1i_2}^{x,c_1c_2}[n_1^2,n_2^2]\;\; \cdots\;\; r_{i_1i_2}^{x,c_1c_2}[n_1^N,n_2^N]\right], \tag{5.3.78}$$

or, in compact notation:

$$\tilde{\mathbf{r}}_{\mathbf{i}_2}^{x,\mathbf{c}_2} \triangleq \left[r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2^1]\;\; r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2^2]\;\; \cdots\;\; r_{\mathbf{i}_2}^{x,\mathbf{c}_2}[\mathbf{n}_2^N]\right]. \tag{5.3.79}$$

The length of $\tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2}$ equals the cardinality N of the set $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$ (see also (5.2.6)) and thus represents the number of time pairs contained in $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$. Finally, following the same procedure as described above, for each $1 \leq j \leq S$ a row vector $\tilde{\mathbf{r}}_j^{s,c_1c_2}$ is associated with the j-th source auto-correlation function $r_j^{s,c_1c_2}[n_1,n_2]$ defined on $\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}$ by defining:

$$\tilde{\mathbf{r}}_j^{s,c_1c_2} \triangleq \left[r_j^{s,c_1c_2}[n_1^1,n_2^1]\;\; r_j^{s,c_1c_2}[n_1^2,n_2^2]\;\; \cdots\;\; r_j^{s,c_1c_2}[n_1^N,n_2^N]\right], \tag{5.3.80}$$

or, in compact notation:

$$\tilde{\mathbf{r}}_j^{s,\mathbf{c}_2} \triangleq \left[r_j^{s,\mathbf{c}_2}[\mathbf{n}_2^1]\;\; r_j^{s,\mathbf{c}_2}[\mathbf{n}_2^2]\;\; \cdots\;\; r_j^{s,\mathbf{c}_2}[\mathbf{n}_2^N]\right]. \tag{5.3.81}$$

Using these notational conventions, the row vector equivalent of expression (5.3.4) becomes:

$$\boxed{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2} = \sum_{j=1}^{S} \left(a_{i_1}^{j}\right)^{c_1} \left(a_{i_2}^{j}\right)^{c_2} \tilde{\mathbf{r}}_{j}^{s,c_1c_2} \qquad \forall\, 1 \leq i_1, i_2 \leq D\,,} \tag{5.3.82}$$

which in compact notation reads (see (5.3.10) and also (5.3.26)):

$$\tilde{\mathbf{r}}_{\mathbf{i}_2}^{x,\mathbf{c}_2} = \sum_{j=1}^{S} \check{a}_{\mathbf{i}_2}^{j,\mathbf{c}_2} \tilde{\mathbf{r}}_{j}^{s,\mathbf{c}_2} \qquad \forall\, \mathbf{i}_2 \in \mathcal{I}_{\mathrm{t},D}^{2}\,. \tag{5.3.83}$$

The various sets of correlation row vectors are denoted by symbols that are similar to those denoting the corresponding correlation functions. The set containing all sensor correlation row vectors associated with (5.3.15) or (5.3.16) is denoted by:

$$\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,c_1c_2} \triangleq \left\{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2} \mid 1 \leq i_1, i_2 \leq D\right\} = \left\{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^{2}}, \tag{5.3.84}$$

which in compact notation reads:

$$\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,\mathbf{c}_2} \triangleq \left\{\tilde{\mathbf{r}}_{\mathbf{i}_2}^{x,\mathbf{c}_2} \mid \mathbf{i}_2 \in \mathcal{I}_{\mathrm{t},D}^{2}\right\} = \left\{\tilde{\mathbf{r}}_{\mathbf{i}_2}^{x,\mathbf{c}_2}\right\}_{\mathbf{i}_2\in\mathcal{I}_{\mathrm{t},D}^{2}}. \tag{5.3.85}$$

Likewise, the set containing all unique sensor correlation row vectors associated with (5.3.21) or (5.3.22) is denoted by:

$$\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2} \triangleq \left\{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}, \tag{5.3.86}$$

which in compact notation reads:

$$\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_2} \triangleq \left\{\tilde{\mathbf{r}}_{\mathbf{i}_2}^{x,\mathbf{c}_2}\right\}_{\mathbf{i}_2\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_2}}. \tag{5.3.87}$$

Finally, the set containing all source auto-correlation row vectors associated with (5.3.24) or (5.3.25) is denoted by:

$$\tilde{\mathcal{K}}_{S}^{s,c_1c_2} \triangleq \left\{\tilde{\mathbf{r}}_{j}^{s,c_1c_2} \mid 1 \leq j \leq S\right\} = \left\{\tilde{\mathbf{r}}_{j}^{s,c_1c_2}\right\}_{1\leq j\leq S}, \tag{5.3.88}$$

which in compact notation reads:

$$\tilde{\mathcal{K}}_{S}^{s,\mathbf{c}_2} \triangleq \left\{\tilde{\mathbf{r}}_{j}^{s,\mathbf{c}_2} \mid 1 \leq j \leq S\right\} = \left\{\tilde{\mathbf{r}}_{j}^{s,\mathbf{c}_2}\right\}_{1\leq j\leq S}. \tag{5.3.89}$$

Clearly, the cardinalities of $\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,\mathbf{c}_2}$, $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_2}$ and $\tilde{\mathcal{K}}_{S}^{s,\mathbf{c}_2}$ are the same as those of $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_2}$, $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2}$ and $\mathcal{K}_{S}^{s,\mathbf{c}_2}$ respectively, which are given by (5.3.17), (5.3.52) and the number of sources S respectively.

5.3.5.2 Dimensions of signal and source subspaces

Now we are in a position to derive the same system of equations as in the previous section in terms of row vectors. Again, we start the derivation by considering the dimension of the signal subspace spanned by the sensor correlation row vectors, which is expressed in (5.3.58) in functional notation. In row vector notation, this expression becomes:

$$\begin{aligned} d_{D}^{x,c_1c_2} &\triangleq \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,c_1c_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{t},D}^{2}}\right)\right) \\ &\equiv \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}\right)\right). \end{aligned} \tag{5.3.90}$$

Likewise, the dimension of the source subspace spanned by the source auto-correlation functions, which is expressed in (5.3.60) in functional notation, can now be expressed in terms of the associated row vectors as follows:

$$d_S^{s,c_1c_2} \triangleq \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_S^{s,c_1c_2}\big)\Big) = \dim\Big(\mathcal{L}\Big(\big\{\tilde{\mathbf{r}}_j^{s,c_1c_2}\big\}_{1\le j\le S}\Big)\Big)\,. \tag{5.3.91}$$

The row vector equivalent of assumption *AS2* on page 222, which states that the source auto-correlation functions are linearly independent, is given by:

$$\sum_{j=1}^{S} \xi^j\, \tilde{\mathbf{r}}_j^{s,\mathbf{c}_2} = \tilde{\mathbf{0}}^N \quad \Longrightarrow \quad \xi^j = 0 \quad \forall\, 1 \le j \le S\,. \tag{5.3.92}$$

From this property it follows again that $d_S^{s,c_1c_2} = S$; see (5.3.62). Equation (5.3.82) states that each sensor correlation row vector $\tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2}$ is a linear combination of the S source auto-correlation row vectors $\tilde{\mathbf{r}}_1^{s,c_1c_2},\ldots,\tilde{\mathbf{r}}_S^{s,c_1c_2}$ with coefficients $\big(a_{i_1}^1\big)^{c_1}\big(a_{i_2}^1\big)^{c_2},\ldots,$ $\big(a_{i_1}^S\big)^{c_1}\big(a_{i_2}^S\big)^{c_2}$ respectively. This again implies 'dimensional relation' (5.3.63) between d_D^{x,c_1c_2} and d_S^{s,c_1c_2}.

5.3.5.3 Linear dependence of sensor correlation row vectors

Following the same reasoning as in Section 5.3.4.2 it follows that:

$$\dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) \le \min\Big(\big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big|,\ S\Big) = \min\Big(M_{\mathrm{u},D}^{x,c_1c_2},\ S\Big)\,.$$

Now, if the number $M_{\mathrm{u},D}^{x,c_1c_2} = |\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}|$ of unique sensor correlation row vectors given by (5.3.50)-(5.3.52) is larger than the dimension d_D^{x,c_1c_2} of the signal subspace spanned by the row vectors in $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}$, i.e. if:

$$\big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big| > \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,c_1c_2} > d_D^{x,c_1c_2}\,, \tag{5.3.93}$$

then *the sensor correlation row vectors in $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}$ are linearly dependent*. Hence, equivalently to (4.3.53), if (5.3.93) is satisfied then by the definition of linear dependence there exist non-zero and non-unique sets of coefficients $\big\{\varphi_1^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}},\ldots,\big\{\varphi_{Q_{D,S}^{c_1c_2}}^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}$, indexed by an index q, such that:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}\, \tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2} = \tilde{\mathbf{0}}^N \qquad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,, \tag{5.3.94}$$

where the set $\mathcal{Q}_{D,S}^{c_1c_2}$ is defined in (5.3.66). Note that the number $Q_{D,S}^{c_1c_2} \triangleq |\mathcal{Q}_{D,S}^{c_1c_2}|$ of linearly independent equations is given by (5.3.67) and that it also equals the dimension of the orthogonal complement of the linear (signal sub)space spanned by the sensor correlation row vectors in $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}$:

$$\boxed{Q_{D,S}^{c_1c_2} = \dim\Big(\big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)^{\perp}\big)\Big) = \big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big| - \dim\Big(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) = M_{\mathrm{u},D}^{x,c_1c_2} - d_D^{x,c_1c_2}\,.} \tag{5.3.95}$$

Hence, in the same way as $Q_{D,S}^{c_1c_2}$ is given by the difference between the left and right hand sides of (5.3.64), it is also given by the difference between the left and right hand sides of (5.3.93). Note that the orthogonal complement of the signal subspace is the noise subspace.

5.3.5.4 Deriving the system of equations by exploiting linear independence of source auto-correlation row vectors

The main part of our argument leading to the desired system of equations amounts to combining (5.3.82) and (5.3.94), and exploiting assumption *AS2* in the form of (5.3.92). Hence, expressing (5.3.94) in terms of the source auto-correlation row vectors by substituting (5.3.82) yields:

$$\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}\,\tilde{\mathbf{r}}_{i_1i_2}^{x,c_1c_2} = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(\sum_{j=1}^{S} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} \tilde{\mathbf{r}}_j^{s,c_1c_2} \right)$$

$$= \sum_{j=1}^{S} \left(\sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} \right) \tilde{\mathbf{r}}_j^{s,c_1c_2} = \tilde{\mathbf{0}}^N \qquad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,. \quad (5.3.96)$$

Now exploiting the linear independence property of the source auto-correlation row vectors, i.e. applying (5.3.92) to the last line in (5.3.96) in the same manner as we did in the previous section again gives system (5.3.70). Hence, from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in the previous section from (5.3.70) on.

5.3.6 Derivation of system of D-variate homogeneous polyconjugal equations of degree two; matrix-vector notation and subspace matrix

In this section we finally present the derivation of the system of homogeneous polyconjugal equations in terms of matrix-vector notation and a 'uniquiefied' subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ along the same lines as Section 4.3.6. In the previous two sections we have shown that the derivation of this system is based on the properties of the signal and noise subspaces. In order to be able to naturally define a proper subspace matrix for our problem formulation and to derive the system of equations from this, in addition to the one-to-one mapping between correlation functions and their corresponding correlation row vectors that we used in the previous section, now we also need to set up a proper one-to-one mapping between function-valued vectors on the one hand and matrices on the other hand; see Fig. 1.11 on page 26.

5.3.6.1 Bijective mapping between function-valued vectors and matrices

Similarly to Section 4.3.6.1, in this section we associate a matrix with each function-valued vector defined on the Noise-Free ROS $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$, or equivalently with the set containing the elements of the function-valued vector, by means of a bijective mapping. Firstly, with each set of functions defined on $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ we associate a function-valued column vector that is also defined on $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ by simply stacking the 'function elements' of the considered set in a column vector in a certain order. For example, the function-valued column vectors $\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]$ and $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ associated with the sets $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ (5.3.15) and $\mathcal{K}_S^{s,c_1c_2}$ (5.3.24) respectively are defined by:

$$\mathbf{r}_D^{x,c_1c_2}[n_1,n_2] \triangleq \begin{bmatrix} r_{11}^{x,c_1c_2}[n_1,n_2] \\ \vdots \\ r_{DD}^{x,c_1c_2}[n_1,n_2] \end{bmatrix}, \quad \mathbf{r}_S^{s,c_1c_2}[n_1,n_2] \triangleq \begin{bmatrix} r_1^{s,c_1c_2}[n_1,n_2] \\ \vdots \\ r_S^{s,c_1c_2}[n_1,n_2] \end{bmatrix}, \quad (5.3.97)$$

where we have arranged the elements of $\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]$ in 'Kronecker order' and the elements of $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ in ascending index order. For convenience of notation, let $\mathbb{C}_P[\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}]$ denote the space of all function-valued column vectors of length P whose elements are members of $\mathbb{C}[\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}]$. Then, it is clear that:

$$\mathbf{r}_D^{x,c_1c_2}[n_1,n_2] \in \mathbb{C}_{M_{\mathrm{t},D}^{x,c_1c_2}}[\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}], \quad \mathbf{r}_S^{s,c_1c_2}[n_1,n_2] \in \mathbb{C}_S[\Omega_{n_1n_2}^{s\backslash\nu,c_1c_2}],$$

where $M_{\mathrm{t},D}^{x,c_1c_2}$ is defined in (5.3.17). Secondly, following the same reasoning and procedure as in Section 4.3.6.1 we associate a matrix with each function-valued column vector, and thus also with the associated set of functions and/or row vectors. For example, at the same time we can associate the three related quantities $\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]$, $\mathcal{K}_{\mathrm{t},D}^{x,c_1c_2}$ and $\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,c_1c_2}$ with the *subspace matrix* $\mathbf{C}_D^{x,c_1c_2}$ defined as follows (see also Section 5.3.5.1 and Fig. 1.11):

$$\begin{aligned}\mathbf{C}_D^{x,c_1c_2} &\triangleq \left[\mathbf{r}_D^{x,c_1c_2}[n_1^1,n_2^1] \;\cdots\; \mathbf{r}_D^{x,c_1c_2}[n_1^N,n_2^N]\right] = \begin{bmatrix}\tilde{\mathbf{r}}_{11}^{x,c_1c_2}\\ \vdots \\ \tilde{\mathbf{r}}_{DD}^{x,c_1c_2}\end{bmatrix} \\ &= \begin{bmatrix} r_{11}^{x,c_1c_2}[n_1^1,n_2^1] & \cdots & r_{11}^{x,c_1c_2}[n_1^N,n_2^N] \\ \vdots & \ddots & \vdots \\ r_{DD}^{x,c_1c_2}[n_1^1,n_2^1] & \cdots & r_{DD}^{x,c_1c_2}[n_1^N,n_2^N]\end{bmatrix} \in \mathbb{C}_{M_{\mathrm{t},D}^{x,c_1c_2}}^N . \end{aligned} \tag{5.3.98}$$

Using the approach explained in Section 5.3.3 we can construct the corresponding uniquiefied quantities $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$, $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$, $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}$, and $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. As we have explained in the previous sections, the index set indexing the functions of $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ and $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}$, as well as the rows of $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}$ and $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, (by definition) is $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$. Suppose that we number the $M_{\mathrm{u},D}^{x,c_1c_2}$ elements of $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ from 1 to $M_{\mathrm{u},D}^{x,c_1c_2}$ in sequential order, i.e. $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$ is written as follows (see Section 4.3.7):

$$\mathcal{I}_{\mathrm{u},D}^{c_1c_2} = \left\{(i_1,i_2)_1,\ldots,(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}\right\} . \tag{5.3.99}$$

Then, the uniquiefied subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ (with which we will mainly work in the sequel) can be expressed as follows:

$$\begin{aligned}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} &\triangleq \left[\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1^1,n_2^1] \;\cdots\; \mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1^N,n_2^N]\right] = \begin{bmatrix}\tilde{\mathbf{r}}_{(i_1,i_2)_1}^{x,c_1c_2}\\ \vdots \\ \tilde{\mathbf{r}}_{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}^{x,c_1c_2}\end{bmatrix} \\ &= \begin{bmatrix} r_{(i_1,i_2)_1}^{x,c_1c_2}[n_1^1,n_2^1] & \cdots & r_{(i_1,i_2)_1}^{x,c_1c_2}[n_1^N,n_2^N] \\ \vdots & \ddots & \vdots \\ r_{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}^{x,c_1c_2}[n_1^1,n_2^1] & \cdots & r_{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}^{x,c_1c_2}[n_1^N,n_2^N]\end{bmatrix} \in \mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}^N . \end{aligned} \tag{5.3.100}$$

Finally, we associate the three related quantities $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$, $\mathcal{K}_S^{s,c_1c_2}$ and $\tilde{\mathcal{K}}_S^{s,c_1c_2}$ with the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ defined as follows:

$$\begin{aligned}\mathbf{C}_S^{s,c_1c_2} &\triangleq \left[\mathbf{r}_S^{s,c_1c_2}[n_1^1,n_2^1] \;\cdots\; \mathbf{r}_S^{s,c_1c_2}[n_1^N,n_2^N]\right] = \begin{bmatrix}\tilde{\mathbf{r}}_1^{s,c_1c_2}\\ \vdots \\ \tilde{\mathbf{r}}_S^{s,c_1c_2}\end{bmatrix} \\ &= \begin{bmatrix} r_1^{s,c_1c_2}[n_1^1,n_2^1] & \cdots & r_1^{s,c_1c_2}[n_1^N,n_2^N] \\ \vdots & \ddots & \vdots \\ r_S^{s,c_1c_2}[n_1^1,n_2^1] & \cdots & r_S^{s,c_1c_2}[n_1^N,n_2^N]\end{bmatrix} \in \mathbb{C}_S^N . \end{aligned} \tag{5.3.101}$$

Note that in definitions (5.3.98), (5.3.100), and (5.3.101) the time pair index indexes the columns of the involved matrix. Therefore, it is in the superscript position.

The concepts associated with the various bijectively mapped quantities are related bijectively as well. Here, we generalize expressions (4.3.62) through (4.3.66) to account for complex quantities. Suppose that the functions in some set $\mathcal{F} \triangleq \{f_{i_1 i_2}^{c_1 c_2}[n_1, n_2]\}_{(i_1,i_2)\in\mathcal{I}_f^{c_1 c_2}}$ are defined on some ROS $\boldsymbol{\mathcal{T}}$ with $|\boldsymbol{\mathcal{T}}| \triangleq N$. As before, with these functions we associate the row vectors in the set $\tilde{\mathcal{F}} \triangleq \{\tilde{\mathbf{f}}_{i_1 i_2}^{c_1 c_2}\}_{(i_1,i_2)\in\mathcal{I}_f^{c_1 c_2}}$. The function-valued column vector $\mathbf{f}[n_1, n_2]$ is defined by stacking the functions in a column vector. Likewise, the matrix $\mathbf{F}$ associated with $\mathbf{f}[n_1, n_2]$ is defined by stacking the row vectors in a matrix; see Fig. 1.11. Now we consider the relations between the fundamental subspaces of $\mathbf{f}[n_1, n_2]$ and $\mathbf{F}$.

Firstly, the linear span $\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[n_1, n_2]\big)$ of the conjugates of the functions in the function-valued column vector argument $\mathbf{f}[n_1, n_2]$ of $\mathcal{L}_{\mathrm{r}}\big(\,\cdot\,\big)$, i.e. the 'linear span in the row direction', can be mapped bijectively to the row range of $\mathbf{F}$:

$$\boxed{\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[n_1, n_2]\big) \triangleq \mathcal{L}\big(\mathcal{F}^*\big) \doteq \mathcal{L}\big(\tilde{\mathcal{F}}^*\big) = \mathcal{R}_{\mathrm{r}}\left(\mathbf{F}\right),} \tag{5.3.102}$$

where $\mathcal{F}^*$ and $\tilde{\mathcal{F}}^*$ denote the sets containing the conjugated elements of the sets $\mathcal{F}$ and $\tilde{\mathcal{F}}$ respectively.
Secondly, the linear span $\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[n_1, n_2]\big)$ of the column vectors obtained by evaluating the argument $\mathbf{f}[n_1, n_2]$ of $\mathcal{L}_{\mathrm{c}}\big(\,\cdot\,\big)$ at the N time pairs in $\boldsymbol{\mathcal{T}}$, i.e. the 'linear span in the column direction', equals the column range of $\mathbf{F}$:

$$\boxed{\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[n_1, n_2]\big) \triangleq \mathcal{L}\big(\{\mathbf{f}[n_1, n_2]\}_{(n_1,n_2)\in\boldsymbol{\mathcal{T}}}\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{F}\right).} \tag{5.3.103}$$

Thirdly, the right null space of $\mathbf{f}[n_1, n_2]$ can be mapped bijectively to that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[n_1, n_2]\big) \doteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{F}\big).} \tag{5.3.104}$$

Finally, the left null space of $\mathbf{f}[n_1, n_2]$ equals that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[n_1, n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{F}\big).} \tag{5.3.105}$$

Because of these relations the dimensions of the spaces in the various bijections are equal to each other, i.e.:

$$\boxed{\begin{aligned}&\dim\big(\mathcal{L}_{\mathrm{r}}(\mathbf{f}[n_1, n_2])\big) = \dim\big(\mathcal{L}(\mathcal{F}^*)\big) = \dim\big(\mathcal{L}(\tilde{\mathcal{F}}^*)\big) = \dim\big(\mathcal{R}_{\mathrm{r}}(\mathbf{F})\big)\\ &= \dim\big(\mathcal{L}_{\mathrm{c}}(\mathbf{f}[n_1, n_2])\big) = \dim\big(\mathcal{L}(\{\mathbf{f}[n_1, n_2]\}_{(n_1,n_2)\in\boldsymbol{\mathcal{T}}})\big)\\ &= \dim\big(\mathcal{R}_{\mathrm{c}}(\mathbf{F})\big) = \mathrm{rank}(\mathbf{F}) = \mathrm{rank}\,\big(\mathbf{f}[n_1, n_2]\big),\end{aligned}} \tag{5.3.106}$$

$$\boxed{\dim\big(\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[n_1, n_2]\big)\big) = \dim\big(\mathcal{N}_{\mathrm{r}}\big(\mathbf{F}\big)\big),} \tag{5.3.107}$$

and:

$$\boxed{\dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[n_1, n_2]\big)\big) = \dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{F}\big)\big).} \tag{5.3.108}$$

5.3.6.2 Expressing sensor correlation function-valued vector and matrix in mixing matrix elements and source auto-correlation function-valued vector and matrix

Similarly to Section 4.3.6.2, in this section we will express the function-valued sensor correlation vector in terms of the mixing matrix elements and the function-valued source auto-correlation vector. Then, we do the same for the associated sensor correlation matrix, i.e. the subspace matrix. Using (5.3.4), (5.3.26) or a derivation that parallels that of (4.3.70) we can express the function-valued sensor correlation vector $\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]$ for all $(n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ in terms of the elements of the mixing matrix $\mathbf{A}$ and the function-valued source auto-correlation vector $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ in a very intuitive manner as follows:

$$\begin{aligned}
\mathbf{r}_D^{x,c_1c_2}[n_1,n_2] &\triangleq \operatorname{mom}\Big(\big(\mathbf{x}[n_1]\big)^{c_1},\big(\mathbf{x}[n_2]\big)^{c_2}\Big) \triangleq E\Big\{\big(\mathbf{x}[n_1]\big)^{c_1}\otimes\big(\mathbf{x}[n_2]\big)^{c_2}\Big\}\\
&= E\left\{\left(\sum_{j_1=1}^{S}\mathbf{a}^{j_1}s_{j_1}[n_1]+\boldsymbol{\nu}[n_1]\right)^{c_1}\otimes\left(\sum_{j_1=1}^{S}\mathbf{a}^{j_2}s_{j_2}[n_2]+\boldsymbol{\nu}[n_2]\right)^{c_2}\right\}\\
&= \sum_{j_1=1}^{S}\sum_{j_2=1}^{S}\Big[\big(\mathbf{a}^{j_1}\big)^{c_1}\otimes\big(\mathbf{a}^{j_2}\big)^{c_2}\Big]\,E\Big\{\big(s_{j_1}[n_1]\big)^{c_1}\big(s_{j_2}[n_2]\big)^{c_2}\Big\}\\
&= \sum_{j=1}^{S}\Big[\big(\mathbf{a}^{j}\big)^{c_1}\otimes\big(\mathbf{a}^{j}\big)^{c_2}\Big]\,E\Big\{\big(s_{j}[n_1]\big)^{c_1}\big(s_{j}[n_2]\big)^{c_2}\Big\}\\
&= \sum_{j=1}^{S}\check{\mathbf{a}}_D^{j,c_1c_2}\,r_j^{s,c_1c_2}[n_1,n_2] = \mathbf{A}_{D,\diamond}^{c_1c_2}\,\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\\
&\forall\,(n_1,n_2)\in\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}.
\end{aligned} \tag{5.3.109}$$

where the second order Kronecker product vector with conjugation pair (c_1,c_2) $\check{\mathbf{a}}_D^{j,c_1c_2}$ is defined in (5.3.29) and the Khatri-Rao product matrix $\mathbf{A}_{D,\diamond}^{c_1c_2}$ in (5.3.54). Note that $\check{\mathbf{a}}_D^{1,c_1c_2},\ldots,$ $\check{\mathbf{a}}_D^{S,c_1c_2}$ serve as *generalized array response vectors*, whereas $\mathbf{A}_{D,\diamond}^{c_1c_2}$ serves as the *generalized array response matrix*; see also Chapter 3. Summarizing, we have derived the following function-valued column vector equivalent of (5.3.4) and (5.3.82):

$$\boxed{\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]=\sum_{j=1}^{S}\check{\mathbf{a}}_{\mathrm{u},D}^{j,c_1c_2}\,r_j^{s,c_1c_2}[n_1,n_2]=\mathbf{A}_{D,\diamond}^{c_1c_2}\,\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\,,} \tag{5.3.110}$$

which is defined on $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$. Now using the bijective mappings established in Section 5.3.6.1, this expression can be written in matrix notation as follows:

$$\boxed{\mathbf{C}_D^{x,c_1c_2}=\mathbf{A}_{D,\diamond}^{c_1c_2}\,\mathbf{C}_S^{s,c_1c_2}\,.} \tag{5.3.111}$$

Equations (5.3.110) and (5.3.111) are natural and compact ways of representing the structure of the set of sensor correlation functions. For each time pair (n_1,n_2) the vector $\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]$ is a linear combination of the (product) vectors $\check{\mathbf{a}}_D^{1,c_1c_2},\ldots,\ \check{\mathbf{a}}_D^{S,c_1c_2}$. The matrix $\mathbf{A}_{D,\diamond}^{c_1c_2}$ containing these vectors defines a linear transformation from $\mathbb{C}_S[\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}]$ to $\mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}[\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}]$, in particular from the function-valued source auto-correlation vector for a certain time pair to a function-valued sensor correlation vector for the same time pair. It also defines a linear transformation from $\mathbb{C}_S$ to $\mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}$.

Similarly to Section 5.3.6.1, expressions (5.3.110) and (5.3.111) can equally well be expressed in terms of their uniquiefied versions as:

$$\boxed{\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2] = \sum_{j=1}^{S} \check{\mathbf{a}}_{\mathrm{u},D}^{j,c_1c_2}\, r_j^{s,c_1c_2}[n_1,n_2] = \mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\, \mathbf{r}_S^{s,c_1c_2}[n_1,n_2]} \tag{5.3.112}$$

and:

$$\boxed{\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\, \mathbf{C}_S^{s,c_1c_2}} \tag{5.3.113}$$

respectively. Note that $\check{\mathbf{a}}_{\mathrm{u},D}^{j,c_1c_2}$ is a subvector of the Kronecker product $\check{\mathbf{a}}_D^{j,c_1c_2}$ defined in (5.3.29) that contains only the unique elements of the latter vector that are selected by the index pairs in $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$. Likewise, $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ is a submatrix of the second order Khatri-Rao product $\mathbf{A}_{D,\diamond}^{c_1c_2}$ defined in (5.3.55) that contains only the unique rows of the latter matrix that are selected by the index pairs in $\mathcal{I}_{\mathrm{u},D}^{c_1c_2}$.

With these concepts in place we can once again derive the system of homogeneous polyconjugal equations in a way that enhances our insight and clearly shows the connections with conventional subspace methods. In fact, we have already done an important part of the job by defining a matrix that can serve naturally as the subspace matrix containing all relevant information in the proper way, viz. the matrix $\mathbf{C}_D^{x,c_1c_2}$ or $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. In the sequel we will mainly work with $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ for reasons that we have explained earlier, but as before in virtually all equations and results $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ can by replaced by $\mathbf{C}_D^{x,c_1c_2}$. We leave it to the reader to make this kind of translation if desired.

5.3.6.3 Dimensions of signal and source subspaces

From (5.3.102) it follows that the row range of the subspace matrix, which equals the linear span of the conjugated corresponding sensor correlation row vectors, is mapped bijectively to the 'linear span in the row direction' of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{L}\big(\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)^*\big) \div \mathcal{L}\big(\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)^*\big) = \mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\,. \tag{5.3.114}$$

Likewise, from (5.3.103) it follows that the column range of the subspace matrix equals the 'linear span in the column direction' of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{L}\Big(\big\{\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big\}_{(n_1,n_2)\in\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}}\Big) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\,. \tag{5.3.115}$$

From (5.3.106) it follows that the dimensions of all these spaces are equal to each other:

$$\begin{aligned} d_D^{x,c_1c_2} &\triangleq \operatorname{rank}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \dim\Big(\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) = \dim\Big(\mathcal{L}\big(\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,c_1c_2}\big)^*\big)\Big) \\ &= \dim\Big(\mathcal{L}\big(\big(\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\big)^*\big)\Big) = \dim\Big(\mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\Big) = \dim\Big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big) \\ &= \dim\Big(\mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\Big) = \operatorname{rank}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\,. \end{aligned} \tag{5.3.116}$$

Hence, naturally *the dimension of the signal subspace equals the rank of the subspace matrix*.

Similarly to (5.3.114), the row range of the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ defined in (5.3.101) is mapped bijectively to the linear span of the associated function-valued column vector $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ in the row direction:

$$\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big) = \mathcal{L}\big(\big(\tilde{\mathcal{K}}_S^{s,c_1c_2}\big)^*\big) \div \mathcal{L}\big(\big(\mathcal{K}_S^{s,c_1c_2}\big)^*\big) = \mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big)\,. \tag{5.3.117}$$

Likewise, similarly to (5.3.115) the column range of $\mathbf{C}_S^{s,c_1c_2}$ equals the linear span of $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ in the column direction:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_S^{s,c_1c_2}\big) = \mathcal{L}\Big(\big\{\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big\}_{(n_1,n_2)\in\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}}\Big) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big)\,. \quad (5.3.118)$$

Similarly to (5.3.116) the dimension of the source subspace can be expressed as follows:

$$\begin{aligned} d_S^{s,c_1c_2} &\triangleq \operatorname{rank}\big(\mathbf{C}_S^{s,c_1c_2}\big) = \dim\Big(\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big)\Big) = \dim\Big(\mathcal{L}\big(\big(\tilde{\mathcal{K}}_S^{s,c_1c_2}\big)^*\big)\Big) \\ &= \dim\Big(\mathcal{L}\big(\big(\mathcal{K}_S^{s,c_1c_2}\big)^*\big)\Big) = \dim\Big(\mathcal{L}_{\mathrm{r}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big)\Big) = \dim\Big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_S^{s,c_1c_2}\big)\Big) \\ &= \dim\Big(\mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big)\Big) = \operatorname{rank}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big)\,; \end{aligned} \quad (5.3.119)$$

see also (5.3.60) and (5.3.91). Hence, naturally *the dimension of the source subspace equals the rank of the source auto-correlation matrix*. The matrix equivalent of assumption *AS2* on page 222, which states that the source auto-correlation functions and row vectors respectively are linearly independent (see also (5.3.92)), is given by:

$$\tilde{\boldsymbol{\xi}}\mathbf{C}_S^{s,c_1c_2} = \tilde{\mathbf{0}}^N \qquad \Longrightarrow \qquad \tilde{\boldsymbol{\xi}} = \begin{bmatrix}\xi^1 & \cdots & \xi^S\end{bmatrix} = \begin{bmatrix}0 & \cdots & 0\end{bmatrix}\,. \quad (5.3.120)$$

Hence, from this property it follows once more that $d_S^{s,c_1c_2} = \operatorname{rank}\big(\mathbf{C}_S^{s,c_1c_2}\big) = S$, i.e. $\mathbf{C}_S^{s,c_1c_2} \in \mathbb{C}_S^N$ has full rank.

Now, we will show how the signal subspace dimension d_D^{x,c_1c_2} depends on the mixing matrix and the source signal properties. To start with, from (5.3.113) and linear algebra [115, 150] it follows that:

$$\operatorname{rank}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) \leq \min\Big(\operatorname{rank}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big), \operatorname{rank}\big(\mathbf{C}_S^{s,c_1c_2}\big)\Big)\,, \quad (5.3.121)$$

which again implies relation (5.3.63) between d_D^{x,c_1c_2} and d_S^{s,c_1c_2}:

$$d_D^{x,c_1c_2} = \operatorname{rank}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) \leq \operatorname{rank}\big(\mathbf{C}_S^{s,c_1c_2}\big) = d_S^{s,c_1c_2} = S\,. \quad (5.3.122)$$

Similar results are obtained when using (5.3.111) instead of (5.3.113). Equations (5.3.113) and (5.3.121) show that the dimension d_D^{x,c_1c_2} of the signal subspace is determined by the Khatri-Rao product matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ on the one hand, and the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ on the other hand. Now, using the fact that the rank of a matrix is unchanged upon either left or right multiplication by a nonsingular matrix [115] and that $\mathbf{C}_S^{s,c_1c_2}$ has full rank, it follows immediately from (5.3.113) that (5.3.122) can be specialized further to:

$$d_D^{x,c_1c_2} = \operatorname{rank}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\,\mathbf{C}_S^{s,c_1c_2}\big) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.$$

Summarizing, d_D^{x,c_1c_2} equals the rank of the linear transformation defined by $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ in (5.3.55), which in turn equals the dimension of the linear space spanned by the vectors $\check{\mathbf{a}}_D^{1,c_1c_2}, \ldots, \check{\mathbf{a}}_D^{S,c_1c_2}$ defined in (5.3.29):

$$\boxed{d_D^{x,c_1c_2} = \operatorname{rank}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.} \quad (5.3.123)$$

Essentially our subspace approach to MIBI is based on the fact that we can compute or estimate the (properties of the) various subspaces of the unknown matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ from the known matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. Now we can understand the importance of studying the rank of the Khatri-Rao product [61, 95, 141, 141, 151, 181] $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ as we did in Section 5.3.3.4. As we have seen in the previous sections, the dimension d_D^{x,c_1c_2} of the signal subspace is of paramount importance for the derivations in this thesis. Because $d_D^{x,c_1c_2} = \operatorname{rank}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)$ as revealed by (5.3.123), the rank of the Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ is of paramount importance.

5.3.6.4 Linear dependence of subspace matrix rows

Following the same reasoning as in Section 5.3.4.2 it follows that:

$$d_D^{x,c_1c_2} = \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\right) = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\right) \leq \min\left(M_{\mathrm{u},D}^{c_1c_2},\ S\right). \tag{5.3.124}$$

Now, if the number $M_{\mathrm{u},D}^{x,c_1c_2} = \#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\}$ of rows of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ is larger than the dimension d_D^{x,c_1c_2} of the signal subspace spanned by the row vectors of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ (this is the matrix equivalent of (5.3.64) and (5.3.93)), i.e. if:

$$\#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\} > \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\right) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,c_1c_2} > d_D^{x,c_1c_2}\,, \tag{5.3.125}$$

then *the subspace matrix rows are linearly dependent*. Hence, equivalently to (4.3.91), if (5.3.125) is satisfied, then, by the definition of linear dependence there exists a non-zero and non-unique matrix $\mathbf{\Phi} \in \mathbb{C}_{Q_{D,S}^{c_1c_2}}^{M_{\mathrm{u},D}^{x,c_1c_2}}$, which represents the complex conjugate $\left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\right)^*$ of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)$ of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ (see Definition C.2.4), such that:

$$\mathbf{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \mathbf{0}_{Q_{D,S}^{c_1c_2}}^{N} \quad \equiv \quad \mathbf{\Phi}\,\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2] = \mathbf{0}_{Q_{D,S}^{c_1c_2}}\ \forall\ (n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}. \tag{5.3.126}$$

The maximum number of linearly independent rows of $\mathbf{\Phi}$ equals the dimension $Q_{D,S}^{c_1c_2}$ of $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)$ and is given by the difference between the number of rows of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ and its rank [115, 150], which equals the number $Q_{D,S}^{c_1c_2}$ derived in (5.3.67) and (5.3.95):

$$\boxed{Q_{D,S}^{c_1c_2} = \dim\left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\right) = \#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\} - \operatorname{rank}(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}) = M_{\mathrm{u},D}^{x,c_1c_2} - d_D^{x,c_1c_2}\,.} \tag{5.3.127}$$

Hence, in the same way as $Q_{D,S}^{c_1c_2}$ is given by (5.3.67) and (5.3.95) as the differences between the left and right hand sides of (5.3.64) and (5.3.93) respectively, it is also given by the difference between the left and right hand sides of (5.3.125).

Equation (5.3.126) is the matrix-vector equivalent of the equations in (5.3.65) and (5.3.94). In fact, using (5.3.99) to define the row vectors:

$$\tilde{\boldsymbol{\varphi}}_q \triangleq \left[\varphi_q^{(i_1,i_2)_1} \ \cdots \ \varphi_q^{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}\right] \in \mathbb{C}^{M_{\mathrm{u},D}^{x,c_1c_2}}\,, \tag{5.3.128}$$

and stacking these vectors in the matrix $\mathbf{\Phi}$ as follows:

$$\mathbf{\Phi} \triangleq \begin{bmatrix} \tilde{\boldsymbol{\varphi}}_1 \\ \vdots \\ \tilde{\boldsymbol{\varphi}}_{Q_{D,S}^{c_1c_2}} \end{bmatrix} = \begin{bmatrix} \varphi_1^{(i_1,i_2)_1} & \cdots & \varphi_1^{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}} \\ \vdots & \vdots & \vdots \\ \varphi_{Q_{D,S}^{c_1c_2}}^{(i_1,i_2)_1} & \cdots & \varphi_{Q_{D,S}^{c_1c_2}}^{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}} \end{bmatrix} \in \mathbb{C}_{Q_{D,S}^{c_1c_2}}^{M_{\mathrm{u},D}^{x,c_1c_2}}\,, \tag{5.3.129}$$

this equivalence is clear immediately. For example, the q-th row of $\mathbf{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ in (5.3.126) is given by $\tilde{\boldsymbol{\varphi}}_q\,\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, which equals the left hand side of (5.3.94). Clearly, all rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ belong to the conjugate left null space of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, i.e.:

$$\tilde{\boldsymbol{\varphi}}_q \in \left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\right)^* = \left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big)\right)^* \qquad \forall\ q \in \mathcal{Q}_{D,S}^{c_1c_2}\,. \tag{5.3.130}$$

In order to obtain a system of equations with as many linearly independent equations as possible, the vectors in the set $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ should be chosen such that they span $\left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\right)^*$ completely. For a matrix $\mathbf{\Phi}$ that contains such as set of rows, we write symbolically:

$$\mathbf{\Phi}^* \twoheadrightarrow \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) \div \big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\big)^{\perp}\,. \tag{5.3.131}$$

The indicated isomorphism follows from property *FSP1* on page 450.

5.3.6.5 Deriving the system of equations by exploiting nonsingularity of source auto-correlation matrix

Similarly to the previous sections, the main part of our argument leading to the desired system of polyconjugal equations amounts to combining (5.3.113) and (5.3.126), and exploiting assumption *AS2* on page 222, which implies that the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ has full rank. Expressing (5.3.126) in terms of $\mathbf{C}_S^{s,c_1c_2}$ and the uniquiefied Khatri-Rao product matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ by substituting (5.3.113) yields:

$$\mathbf{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \mathbf{\Phi}\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\,\mathbf{C}_S^{s,c_1c_2} = \mathbf{0}_{Q_{D,S}^{c_1c_2}}^{N}\,. \tag{5.3.132}$$

Because $\mathbf{C}_S^{s,c_1c_2}$ has full rank due to assumption *AS2*, its pseudo-inverse $\big(\mathbf{C}_S^{s,c_1c_2}\big)^\dagger$ exists. Hence, as in Section 4.3.6.5 we can multiply both sides (5.3.132) from the right by $\big(\mathbf{C}_S^{s,c_1c_2}\big)^\dagger$ in order to obtain the following system of equations:

$$\mathbf{\Phi}\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2} = \mathbf{0}_{Q_{D,S}^{c_1c_2}}^{S} \qquad \equiv \qquad \mathbf{\Phi}\,\check{\mathbf{a}}_{\mathrm{u},D}^{j,c_1c_2} = \mathbf{0}_{Q_{D,S}^{c_1c_2}} \quad \forall\, 1 \le j \le S\,, \tag{5.3.133}$$

which equals the system in (5.3.70). Hence, again from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in Section 5.3.4 from (5.3.70) on.

5.3.6.6 Concise enlightening derivation of system from subspace matrix

In this section, we present an enlightening derivation in the same spirit as the one given in Section 4.3.6.6 of the previous chapter. Firstly, we prove that $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ and $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ have the same column spaces and thus also the same left null spaces. Then we use these properties to obtain the system of homogeneous polyconjugal equations in a very concise manner. We start by showing that the column range of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, i.e. the signal subspace, equals the column range of the (uniquiefied) second order Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$, i.e. the linear span of the (uniquiefied) array response vectors, whenever the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ has full rank (which is the case in our problem definition). For reference purposes, we summarize this important result explicitly in the following theorem.

Theorem 5.3.3. Column range of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ equals column range of $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$.
If the source auto-correlation matrix $\mathbf{C}_S^{s,c_1c_2}$ has full rank, then the column range of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ equals the column range of the second order Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$ with conjugation pair (c_1, c_2) of the mixing matrix $\mathbf{A}$:

$$\boxed{\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.} \tag{5.3.134}$$

Proof. We prove this assertion using (5.3.113). From this equation it follows directly that

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) \subseteq \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.$$

In addition, multiplying both sides of (5.3.113) from the right by $(\mathbf{C}_S^{s,c_1c_2})^\dagger$ yields the equality $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2} = \mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}(\mathbf{C}_S^{s,c_1c_2})^\dagger$, which implies that:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) \subseteq \mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\,.$$

These two equations immediately imply (5.3.134). □

Corollary 5.3.4. Left null space of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ equals left null space of $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$.
Under the assumption(s) of Theorem 5.3.3 the left null space of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ equals the left null space of $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.} \tag{5.3.135}$$

Proof. Applying property *FSP1* on page 450 to both sides of (5.3.134) immediately proves the assertion. □

For the scenario considered in Chapter 4 we have shown in Section 4.3.6.6 that the system of homogeneous polynomial equations can directly be derived from two facts, viz. on the one hand the property that *the left null space of the unknown matrix containing the generalized array response vectors equals the left null space of the known subspace matrix*, and on the other hand the trivial property that the left null space of any matrix is orthogonal to the column range of the same matrix. Here, we do the same for deriving the system of homogeneous polyconjugal equations. The first fact is formulated in Eq. (5.3.135) of Corollary 5.3.4, whereas applying the second to the uniquiefied generalized array response matrix yields:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) \quad \overset{(5.3.135)}{\Longrightarrow} \quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)\,.} \tag{5.3.136}$$

Now realizing that the columns of $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}$, i.e. the generalized array response vectors, form a basis for $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big)$, and defining the matrix $\boldsymbol{\Phi}$ such that its rows form a basis for $\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\Big)^*$, it finally follows that:

$$\boldsymbol{\Phi}\,\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2} = \mathbf{0}_{Q_{D,S}^{c_1c_2}}^{S}\,,$$

which equals the systems in (5.3.70) and (5.3.133). Hence, again from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in Section 5.3.4 from (5.3.70) on.

The few steps outlined in the previous paragraph clearly reflect and constitute the essence of our subspace approach to MIBI. Again, it is important to note that this concise derivation of the system of equations is possible by virtue of the fact that we have imposed a fixed order on the set of sensor correlation row vectors by stacking them in the chosen order on top of each other in the subspace matrix, and that we could have given similar concise derivations for the functional and row vector formulations in Sections 5.3.4 and 5.3.5 by also imposing a fixed order on the set of sensor correlation functions and sensor correlation row vectors respectively, and defining and employing appropriate concepts of left null spaces. We leave these insightful derivations to the reader. See also the remarks at the end of Sections 4.3.4 and 4.3.6.6. Finally we remark that the order itself, i.e. the order in which the correlation functions and row vectors are stacked on top of each other in the source and sensor function-valued vectors and matrices respectively, is not important because for a different order a completely equivalent system can be derived and completely equivalent conclusions can be drawn. Obviously, if we would choose a different order, the transformation matrix from $\mathbf{C}_S^{s,c_1c_2}$ to $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, or equivalently from $\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]$ to $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$, would be different from the one in (5.3.54) in that its columns and/or rows would be ordered differently. Finally we note that *everywhere in the whole derivation above, we could have used $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ instead of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ because of the bijective mapping defined between them.*

5.3.7 Writing the system of equations in matrix-vector notation

We conclude this section by writing the system of equations $\left\{ f_{D,q}^{c_1 c_2}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$ given by (5.3.73) in matrix-vector notation. Firstly, using (5.3.128) and (5.3.30) each function $f_{D,q}^{c_1 c_2}(\mathbf{z})$ defined in (5.3.71) can we written as follows:

$$\boxed{\begin{array}{l} f_{D,q}^{c_1 c_2}(\mathbf{z}) \triangleq \displaystyle\sum_{(i_1,i_2) \in \mathcal{I}_{\mathrm{u},D}^{c_1 c_2}} \varphi_q^{i_1 i_2} (z_{i_1})^{c_1} (z_{i_2})^{c_2} = \tilde{\boldsymbol{\varphi}}_q \, \check{\mathbf{z}}_{\mathrm{u},D}^{c_1 c_2} = \tilde{\boldsymbol{\varphi}}_q \, \mathbf{w}_{\mathrm{u},D}^{c_1 c_2}(\mathbf{z}) \\ \forall \, \mathbf{z} \in \mathbb{C}_D \, , \quad \forall \, q \in \mathcal{Q}_{D,S}^{c_1 c_2} \, . \end{array}} \tag{5.3.137}$$

Defining the column vector $\mathbf{f}_{D,S}^{c_1 c_2}(\mathbf{z})$ containing the functions $\left\{ f_{D,q}^{c_1 c_2}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$ by:

$$\mathbf{f}_{D,S}^{c_1 c_2}(\mathbf{z}) \triangleq \begin{bmatrix} f_{D,1}^{c_1 c_2}(\mathbf{z}) \\ \vdots \\ f_{D,Q_{D,S}^{c_1 c_2}}^{c_1 c_2}(\mathbf{z}) \end{bmatrix} , \tag{5.3.138}$$

system (5.3.73) can be written in matrix-vector notation in the following manner:

$$\boxed{\mathbf{f}_{D,S}^{c_1 c_2}(\mathbf{z}) = \boldsymbol{\Phi} \, \check{\mathbf{z}}_{\mathrm{u},D}^{c_1 c_2} = \boldsymbol{\Phi} \, \mathbf{w}_{\mathrm{u},D}^{c_1 c_2}(\mathbf{z}) = \mathbf{0}_{Q_{D,S}^{c_1 c_2}} \, .} \tag{5.3.139}$$

From this equation and (5.3.131) it follows that *the problem of solving the system* $\mathbf{f}_{D,S}^{c_1 c_2}(\mathbf{z}) = \mathbf{0}$ *is equivalent to finding all solutions for* $\mathbf{z}$ *such that the structured vector* $\check{\mathbf{z}}_{\mathrm{u},D}^{c_1 c_2}$*, which is a function* $\mathbf{w}_{\mathrm{u},D}^{c_1 c_2}(\mathbf{z})$ *of* $\mathbf{z}$ *and has the same structure as the columns of* $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}$*, belongs to the orthogonal complement of the left null space* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}\big)$ *of* $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}$ *that can be computed from the known matrix* $\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}$ *because* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}\big)$ *equals* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}\big)$*. Equivalently,* $\check{\mathbf{z}}_{\mathrm{u},D}^{c_1 c_2}$ *belongs to the column range* $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}\big)$ *of* $\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1 c_2}$*.*

5.3.8 Using the SVD to find proper coefficients of the polyconjugals

In this section, we generalize the results of Section 4.3.8 of Chapter 4. In the previous chapter and sections we have projected the MIBI problem onto the problem of solving a system of D-variate 2-homogeneous polyconjugal equations for the columns of the mixing matrix $\mathbf{A}$. We also have remarked that the coefficients of the equations in the system can be considered as known because they can be deduced from the sensor cumulant functions by means of the Singular Value Decomposition (SVD). In this section, the procedure for doing this is explained along the lines of Section 4.3.8 and amounts to performing a subspace decomposition that is similar to that performed by conventional subspace methods. Again, we provide both a functional and a matrix formulation of the SVD. In the sequel of this section, both the functional SVD of $\mathbf{r}_{\mathrm{u},D}^{x,c_1 c_2}[n_1, n_2]$ and the equivalent standard matrix SVD of $\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}$ are used to obtain a proper set $\Phi_{D,S}^{c_1 c_2}$ (5.3.68) containing a set of coefficients for each equation in the system, or equivalently the matrix $\boldsymbol{\Phi}$ (5.3.129) containing a row for each equation. In Section 5.3.6, in particular Sections 5.3.6.4 and 5.3.6.6, we have seen that the rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$ of $\Phi_{D,S}^{c_1 c_2}$ belong to the conjugate left null space $\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1 c_2}[n_1, n_2]\big)\Big)^* = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}\big)\Big)^*$ of $\mathbf{r}_{\mathrm{u},D}^{x,c_1 c_2}[n_1, n_2]$ and $\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}$, which can be determined directly from the SVD of $\mathbf{r}_{\mathrm{u},D}^{x,c_1 c_2}[n_1, n_2]$ or $\mathbf{C}_{\mathrm{u},D}^{x,c_1 c_2}$ by choosing and conjugate

transposing the left singular vectors that correspond to zero singular values. Hence, in principle it is sufficient to determine this left null space. However, for the sake of insight and because we will also use another fundamental subspace of $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ and $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ later on in this chapter (in Section 5.6, where the MIBI problem is projected onto a multi-matrix generalized eigenvalue problem complementary to the current projection onto the system of homogeneous polyconjugal equations), we will discuss the fully fledged SVD here. We start by formulating the functional SVD and then present the standard matrix SVD.

5.3.8.1 Functional SVD of function-valued sensor correlation vector $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$

The following theorem presents the functional singular value decomposition[‡] of the function-valued vector $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ understood to be defined on $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$:

Theorem 5.3.5. SVD of function-valued sensor correlation vector.
Let $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2] \in \mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}[\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}]$ be a (uniquefied) function-valued sensor correlation vector with rank $d_D^{x,c_1c_2} \triangleq \operatorname{rank}\left(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\right)^{\ddagger}$, number of elements $M_{\mathrm{u},D}^{x,c_1c_2} \triangleq \left|\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2}\right|$ and ROS $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$ with cardinality $N \triangleq \left|\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}\right|$. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, column vectors $\mathbf{u}^p \in \mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}$ for $1 \leq p \leq M_{\mathrm{u},D}^{x,c_1c_2}$ and functions $v_p[n_1,n_2] \in \mathbb{C}[\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}]$ for $1 \leq p \leq N$, such that $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ can be decomposed as follows:

$$\boxed{\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2] = \sum_{p=1}^{P} \sigma_p\, \mathbf{u}^p\, (v_p[n_1,n_2])^* \qquad \forall\, (n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}\,,}$$

where:

- $P = \min(M_{\mathrm{u},D}^{x,c_1c_2}, N)\,;$
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall\, 1 \leq m,n \leq M_{\mathrm{u},D}^{x,c_1c_2}\,;$
- $\langle v_m[n_1,n_2], v_n[n_1,n_2] \rangle_{\mathrm{f}} = \delta_{mn} \quad \forall\, 1 \leq m,n \leq N\,;$
- $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,c_1c_2}} > 0,\ \sigma_{d_D^{x,c_1c_2}+1}, \ldots, \sigma_P = 0\,.$

This theorem is the result of the application of Theorem D.2.1 in Section D.2 to the problem studied in this chapter. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}$, functions $v_1[n_1,n_2], \ldots, v_N[n_1,n_2]$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors, right singular functions*, and *singular values* respectively of $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$. The definitions of the employed inner product functions $\langle\cdot,\cdot\rangle_{\mathrm{v}}$ and $\langle\cdot,\cdot\rangle_{\mathrm{f}}$ are given by (D.2.3) and (D.2.4) respectively. For the current scenario these definitions can be given as follows. The first inner product is a function from $\mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}} \times \mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}$ to $\mathbb{C}$ and is defined by:

$$\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} \triangleq \sum_{p=1}^{M_{\mathrm{u},D}^{x,c_1c_2}} u_p^m\, (u_p^n)^* = \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} u_{i_1i_2}^m\, (u_{i_1i_2}^n)^*\,, \tag{5.3.140}$$

where the 'v' in the subscript position indicates that this inner product is of the '<u>v</u>ector type' and in the latter expression we have used the notation explained in Section 4.3.7. The second

[‡] See Section D.2 for the definitions of the functional singular value decomposition, the rank of a function-valued vector, the relevant inner product functions, etc.

inner product is a function from $\mathbb{C}[\Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}] \times \mathbb{C}[\Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}]$ to $\mathbb{C}$ and is defined by:

$$\langle v_m[n_1, n_2], v_n[n_1, n_2] \rangle_{\mathrm{f}} \triangleq \sum_{(n_1,n_2) \in \Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}} v_m[n_1, n_2] \left(v_n[n_1, n_2] \right)^*, \tag{5.3.141}$$

where the 'f' in the subscript position indicates that this inner product is of the 'functional type'. The theorem can be proven by using the standard SVD theorem [72, 115] described in Section D.1 and the various bijective mappings that we have discussed earlier, i.e. those between functions and row vectors, and function-valued vectors and matrices; see also Fig. 1.11 on page 26.

For convenience and insight the set of vectors and the set of functions in the SVD are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. Since this partitioning is particularly relevant for the subspace techniques used in this chapter and the rest of the thesis, we explain the associated notation in detail. Firstly, we define the set $\mathcal{U}$ containing the left singular vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1 c_2}}$ of the decomposition as follows:

$$\mathcal{U} \triangleq \left\{ \mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1 c_2}} \right\}.$$

This set is split into the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$, where:

$$\mathcal{U}^s \triangleq \left\{ \mathbf{u}^1, \ldots, \mathbf{u}^{d_D^{x,c_1 c_2}} \right\}$$

contains the left singular vectors corresponding to the non-zero singular values, and:

$$\mathcal{U}^\nu \triangleq \left\{ \mathbf{u}^{d_D^{x,c_1 c_2}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1 c_2}} \right\}$$

contains the left singular vectors corresponding to the zero singular values. In subspace decomposition terms the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, we define the set $\mathcal{V}$ containing the right singular functions $v_1[n_1, n_2], \ldots, v_N[n_1, n_2]$ of the decomposition as follows:

$$\mathcal{V} \triangleq \left\{ v_1[n_1, n_2], \ldots, v_N[n_1, n_2] \right\}.$$

This set is split into the sets $\mathcal{V}_s$ and $\mathcal{V}_\nu$, where:

$$\mathcal{V}_s \triangleq \left\{ v_1[n_1, n_2], \ldots, v_{d_D^{x,c_1 c_2}}[n_1, n_2] \right\}$$

contains the right singular functions corresponding to the non-zero singular values, and:

$$\mathcal{V}_\nu \triangleq \left\{ v_{d_D^{x,c_1 c_2}+1}[n_1, n_2], \ldots, v_N[n_1, n_2] \right\}$$

contains the right singular functions corresponding to the zero singular values. Now splitting the SVD expression in Theorem 5.3.5 accordingly yields:

$$\begin{aligned} \mathbf{r}_{\mathrm{u},D}^{x,c_1 c_2}[n_1, n_2] &= \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \left(v_p[n_1, n_2] \right)^* \\ &= \sum_{p=1}^{d_D^{x,c_1 c_2}} \sigma_p \, \mathbf{u}^p \left(v_p[n_1, n_2] \right)^* + \sum_{p=d_D^{x,c_1 c_2}+1}^{P} \sigma_p \, \mathbf{u}^p \left(v_p[n_1, n_2] \right)^* \qquad \forall \, (n_1, n_2) \in \Omega_{n_1 n_2}^{s \setminus \nu, c_1 c_2}, \end{aligned} \tag{5.3.142}$$

where the first summation contains only vectors and functions from $\mathcal{U}^s$ and $\mathcal{V}_s$ respectively, and the second contains only vectors and functions from $\mathcal{U}^\nu$ and $\mathcal{V}_\nu$ respectively.

Corollary 5.3.6. Reduced SVD of function-valued sensor correlation vector.
Under the assumptions formulated in Theorem 5.3.5, $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ *can be written in the following reduced form:*

$$\boxed{\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2] = \sum_{p=1}^{d_D^{x,c_1c_2}} \sigma_p\, \mathbf{u}^p \left(v_p[n_1,n_2]\right)^* \qquad \forall\, (n_1,n_2) \in \Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}\,.} \tag{5.3.143}$$

Proof. Substituting $\sigma_{d_D^{x,c_1c_2}+1},\ldots,\sigma_P = 0$ into (5.3.142) directly yields the Corollary. □

Using the notation introduced in Section 4.3.7 we can now write each sensor correlation function $r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]$ in the following natural 'functional SVD-form':

$$\boxed{r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] = \sum_{p=1}^{d_D^{x,c_1c_2}} \sigma_p\, u_{i_1i_2}^p \left(v_p[n_1,n_2]\right)^* \qquad \forall\, (i_1,i_2) \in \mathcal{I}_{\mathrm{u},D}^{c_1c_2}\,.} \tag{5.3.144}$$

Hence, the function $r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]$ is a linear combination of the right singular functions $(v_1[n_1,n_2])^*,\ldots,(v_{d_D^{x,c_1c_2}}[n_1,n_2])^*$ with coefficients $\sigma_1\, u_{i_1i_2}^1,\ldots,\sigma_{d_D^{x,c_1c_2}}\, u_{i_1i_2}^{d_D^{x,c_1c_2}}$. Compare this expression with (5.3.4), which is repeated here for convenience:

$$r_{i_1i_2}^{x,c_1c_2}[n_1,n_2] = \sum_{j=1}^{S} \left(a_{i_1}^j\right)^{c_1} \left(a_{i_2}^j\right)^{c_2} r_j^{s,c_1c_2}[n_1,n_2] \qquad \forall\, (i_1,i_2) \in \mathcal{I}_{\mathrm{u},D}^{c_1c_2}\,.$$

This equation says that each sensor correlation function $r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]$ is also a linear combination of the S source auto-correlation functions $r_1^{s,c_1c_2}[n_1,n_2],\ldots,r_S^{s,c_1c_2}[n_1,n_2]$ with coefficients $\left(a_{i_1}^1\right)^{c_1}\left(a_{i_2}^1\right)^{c_2},\ldots,\left(a_{i_1}^S\right)^{c_1}\left(a_{i_2}^S\right)^{c_2}$ respectively. The precise relations between the various (fundamental) subspaces can be formulated as follows (see also (D.2.6)-(D.2.9)):

$$\mathcal{L}_c\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) = \mathcal{L}\left(\mathcal{U}^s\right)\,; \tag{5.3.145}$$

$$\begin{aligned}\mathcal{L}_r\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) &= \mathcal{L}\big(\mathcal{V}_s\big) \doteqdot \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\\ &\subseteq \mathcal{L}_r\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big) \doteqdot \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big)\,;\end{aligned} \tag{5.3.146}$$

$$\begin{aligned}\mathcal{N}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) &= \mathcal{L}\big(\mathcal{V}_\nu\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)\\ &\supseteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big)\,;\end{aligned} \tag{5.3.147}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) = \left(\mathcal{L}\left(\mathcal{U}^\nu\right)\right)^T\,, \tag{5.3.148}$$

where $\mathcal{L}_r(\cdot)$ denotes the linear span of the conjugates of the functions in argument of the function-valued vector. This concludes our discussion on the functional SVD. See Section 5.3.8.3 for explicit expressions of a proper set $\Phi_{D,S}^{c_1c_2}$ (5.3.67) containing sets of coefficients for the polyconjugal equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ (5.3.128), or a matrix $\boldsymbol{\Phi}$ (5.3.129) representing these coefficients, in terms of the (conjugate of the) left null space of $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ found by the SVD.

5.3.8.2 Matrix SVD of subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$

In this section, we present the matrix equivalent of Theorem 5.3.5, which is given by the standard Singular Value Decomposition of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. Because the function-valued vector $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ and the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ are related bijectively, their SVD's are related bijectively as well. For details about this relation, we refer the reader to Appendix D. The following theorem presents the SVD of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$:

Theorem 5.3.7. SVD of subspace matrix.
Let $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} \in \mathbb{C}^{N}_{M_{\mathrm{u},D}^{x,c_1c_2}}$ be a (uniquiefied sensor correlation) subspace matrix with rank $d_D^{x,c_1c_2} \triangleq \mathrm{rank}(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2})$, and with $M_{\mathrm{u},D}^{x,c_1c_2}$ and N defined as in Theorem 5.3.5. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, column vectors $\mathbf{u}^p \in \mathbb{C}_{M_{\mathrm{u},D}^{x,c_1c_2}}$ for $1 \leq p \leq M_{\mathrm{u},D}^{x,c_1c_2}$ and row vectors $\tilde{\mathbf{v}}_p \in \mathbb{C}^N$ for $1 \leq p \leq N$, such that $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ can be decomposed as follows:

$$\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \, (\tilde{\mathbf{v}}_p)^* \equiv \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^* \,,$$

where:

- $P = \min(M_{\mathrm{u},D}^{x,c_1c_2}, N)\,;$
- $\mathbf{U} \triangleq \left[\mathbf{u}^1 \;\; \cdots \;\; \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}\right] \in \mathbb{C}^{M_{\mathrm{u},D}^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}}\,;$
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall\, 1 \leq m,n \leq M_{\mathrm{u},D}^{x,c_1c_2} \quad \equiv \quad \mathbf{U}^H\mathbf{U} = \mathbf{U}\mathbf{U}^H = \mathbf{I}\,;$
- $\mathbf{V} \triangleq \begin{bmatrix} \tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_N \end{bmatrix} \in \mathbb{C}^N_N\,;$
- $\langle \tilde{\mathbf{v}}_m, \tilde{\mathbf{v}}_n \rangle_{\mathrm{v}} = \delta_{mn} \quad \forall\, 1 \leq m,n \leq N \quad \equiv \quad \mathbf{V}^H\mathbf{V} = \mathbf{V}\mathbf{V}^H = \mathbf{I}\,;$
- $\boldsymbol{\Sigma} \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_P\} \in \mathbb{R}^N_{M_{\mathrm{u},D}^{x,c_1c_2}}\,;$
- $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,c_1c_2}} > 0,\; \sigma_{d_D^{x,c_1c_2}+1}, \ldots, \sigma_P = 0\,.$

This theorem is the result of the application of Theorem D.1.1 in Section D.1 to the MIBI problem studied in this chapter. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}$, row vectors $\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_N$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors*, *right singular vectors*, and *singular values* respectively of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. Note that now only vector type inner product functions are used. For a proof we refer the reader to standard textbooks on this topic, e.g. see [72]. The matrices $\mathbf{U}$ and $\mathbf{V}$ are square and unitary, i.e. $\mathbf{U}^H\mathbf{U} = \mathbf{U}\mathbf{U}^H = \mathbf{I}$ and $\mathbf{V}^H\mathbf{V} = \mathbf{V}\mathbf{V}^H = \mathbf{I}$. Although the matrix $\boldsymbol{\Sigma}$ containing the singular values is not a square matrix, notationally it is written as $\mathrm{diag}\{\sigma_1, \ldots, \sigma_P\}$, and its off-diagonal elements are zero. The singular values $\sigma_1 \geq \cdots \geq \sigma_P$ are written along the main diagonal in descending order, and rows or columns of zeros are appended as necessary to obtain the proper dimensions. Note that only the first d_D^{x,c_1c_2} singular values are non-zero because $\mathrm{rank}(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}) = d_D^{x,c_1c_2}$. See Appendix D for details and examples.

Similarly to the functional SVD in Section 5.3.8.1, for convenience and insight the various components of the matrix SVD of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. To start with, let $\mathbf{\Sigma}_s^s$ represent the square diagonal block of $\mathbf{\Sigma}$ that contains these non-zero singular values, i.e. $\mathbf{\Sigma}_s^s$ is the upper left block of $\mathbf{\Sigma}$ of size $d_D^{x,c_1c_2} \times d_D^{x,c_1c_2}$. Furthermore, let $\mathbf{\Sigma}_\nu^\nu$ represent the (not necessarily square) block of zeros of $\mathbf{\Sigma}$ to the right and below $\mathbf{\Sigma}_s^s$, i.e. $\mathbf{\Sigma}_\nu^\nu$ is the lower right block of $\mathbf{\Sigma}$ of size $(M_{\mathrm{u},D}^{x,c_1c_2} - d_D^{x,c_1c_2}) \times (N - d_D^{x,c_1c_2})$. Thus, we have:

$$\mathbf{\Sigma}_s^s \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_{d_D^{x,c_1c_2}}\} \in \mathbb{R}^{d_D^{x,c_1c_2}}_{d_D^{x,c_1c_2}} \qquad \text{and} \qquad \mathbf{\Sigma}_\nu^\nu \triangleq \mathbf{0}^{N-d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}}\,, \quad (5.3.149)$$

and $\mathbf{\Sigma}$ can be written as follows:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-d_D^{x,c_1c_2}}_{d_D^{x,c_1c_2}} \\ \mathbf{0}^{d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}} & \mathbf{\Sigma}_\nu^\nu \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-d_D^{x,c_1c_2}}_{d_D^{x,c_1c_2}} \\ \mathbf{0}^{d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}} & \mathbf{0}^{N-d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}} \end{bmatrix}. \quad (5.3.150)$$

The matrix $\mathbf{U}$ is split into two parts as $\mathbf{U} = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix}$. The first part $\mathbf{U}^s$ consists of columns that are multiplied by the non-zero singular values in the SVD-expression of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$, whereas the second part $\mathbf{U}^\nu$ consists of columns that are multiplied by the zero singular values. Hence:

$$\mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^{d_D^{x,c_1c_2}}\end{bmatrix} \in \mathbb{C}^{d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}} \quad (5.3.151)$$

and:

$$\mathbf{U}^\nu \triangleq \begin{bmatrix}\mathbf{u}^{d_D^{x,c_1c_2}+1} & \cdots & \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}\end{bmatrix} \in \mathbb{C}^{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}}\,, \quad (5.3.152)$$

where $\mathbf{u}^p$ is the p-th column of $\mathbf{U}$. In subspace decomposition terms the matrices $\mathbf{U}^s$ and $\mathbf{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, the matrix $\mathbf{V}$ is split as $\mathbf{V} = \begin{bmatrix}\mathbf{V}_s \\ \mathbf{V}_\nu\end{bmatrix}$ with:

$$\mathbf{V}_s \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_{d_D^{x,c_1c_2}}\end{bmatrix} \in \mathbb{C}^N_{d_D^{x,c_1c_2}} \quad \text{and} \quad \mathbf{V}_\nu \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_{d_D^{x,c_1c_2}+1} \\ \vdots \\ \tilde{\mathbf{v}}_N\end{bmatrix} \in \mathbb{C}^N_{N-d_D^{x,c_1c_2}}\,, \quad (5.3.153)$$

where $\tilde{\mathbf{v}}_p$ is the p-th row of $\mathbf{V}$. Now splitting the SVD expression in Theorem 5.3.7 accordingly yields:

$$\begin{aligned}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} &= \mathbf{U}\mathbf{\Sigma}(\mathbf{V})^* = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_s^s & \mathbf{0}^{N-d_D^{x,c_1c_2}}_{d_D^{x,c_1c_2}} \\ \mathbf{0}^{d_D^{x,c_1c_2}}_{M_{\mathrm{u},D}^{x,c_1c_2}-d_D^{x,c_1c_2}} & \mathbf{\Sigma}_\nu^\nu \end{bmatrix} \begin{bmatrix}(\mathbf{V}_s)^* \\ (\mathbf{V}_\nu)^*\end{bmatrix} \\ &= \mathbf{U}^s\mathbf{\Sigma}_s^s(\mathbf{V}_s)^* + \mathbf{U}^\nu\mathbf{\Sigma}_\nu^\nu(\mathbf{V}_\nu)^* = \sum_{p=1}^{d_D^{x,c_1c_2}} \sigma_p\,\mathbf{u}^p\,(\tilde{\mathbf{v}}_p)^* + \sum_{p=d_D^{x,c_1c_2}+1}^{P} \sigma_p\,\mathbf{u}^p\,(\tilde{\mathbf{v}})^*\,,\end{aligned} \quad (5.3.154)$$

where the first summation in each of the last two expressions contains only vectors from $\mathbf{U}^s$ and $\mathbf{V}_s$ respectively, and the second only from $\mathbf{U}^\nu$ and $\mathbf{V}_\nu$ respectively. Note the resemblance of the partitioning described above to (5.3.142), and the partitioning into signal and noise subspaces that is reminiscent of the subspace methods explained in Chapters 3 and 4.

Corollary 5.3.8. Reduced SVD of subspace matrix.
Under the assumptions formulated in Theorem 5.3.7, $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ *can be written in the following reduced form:*

$$\boxed{\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \mathbf{U}^s \mathbf{\Sigma}_s^s (\mathbf{V}_s)^* = \sum_{p=1}^{d_D^{x,c_1c_2}} \sigma_p \, \mathbf{u}^p \, (\tilde{\mathbf{v}}_p)^* \, .} \tag{5.3.155}$$

Proof. Substituting $\sigma_{d_D^{x,c_1c_2}+1}, \ldots, \sigma_P = 0$ into (5.3.154) directly yields the Corollary, which is the equivalent of Corollary (4.3.4). □

Using the notation introduced in Section 4.3.7 we can now write each sensor correlation row vector $\tilde{\mathbf{r}}_{i_1 i_2}^{x}$ in the following natural SVD-form:

$$\boxed{\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2} = \sum_{p=1}^{d_D^{x,c_1c_2}} \sigma_p \, u_{i_1 i_2}^p \, (\tilde{\mathbf{v}}_p)^* \qquad \forall \, (i_1, i_2) \in \mathcal{I}_{\mathrm{u},D}^{c_1c_2} \, ,} \tag{5.3.156}$$

which is the row vector equivalent of (6.2.130). Hence, the row vector $\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}$ is a linear combination of the row vectors $(\tilde{\mathbf{v}}_1)^*, \ldots, (\tilde{\mathbf{v}}_{d_D^{x,c_1c_2}})^*$ with coefficients $\sigma_1 \, u_{i_1 i_2}^1, \ldots, \sigma_{d_D^{x,c_1c_2}} \, u_{i_1 i_2}^{d_D^{x,c_1c_2}}$, which is consistent with the results obtained in the previous section. Compare (5.3.156) with (5.3.82), which is repeated here for convenience:

$$\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2} = \sum_{j=1}^{S} \big(a_{i_1}^j\big)^{c_1} \big(a_{i_2}^j\big)^{c_2} \, \tilde{\mathbf{r}}_j^{s,c_1c_2} \qquad \forall \, (i_1, i_2) \in \mathcal{I}_{\mathrm{u},D}^{c_1c_2} \, . \tag{5.3.157}$$

This equation states that each sensor correlation row vector $\tilde{\mathbf{r}}_{i_1 i_2}^{x,c_1c_2}$ is a linear combination of the S source auto-correlation row vectors $\tilde{\mathbf{r}}_1^{s,c_1c_2}, \ldots, \tilde{\mathbf{r}}_S^{s,c_1c_2}$ with coefficients $\big(a_{i_1}^1\big)^{c_1}\big(a_{i_2}^1\big)^{c_2}$, …, $\big(a_{i_1}^S\big)^{c_1}\big(a_{i_2}^S\big)^{c_2}$ respectively. Analogously to (4.3.122)-(4.3.125), the precise relationships between the various (fundamental) subspaces can be formulated as follows (see also (5.3.145)-(5.3.148) and (D.1.8)-(D.1.11)):

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{L}_c\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^s\big) \, ; \tag{5.3.158}$$

$$\begin{aligned} \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) &= \mathcal{R}_{\mathrm{r}}\big((\mathbf{V}_s)^*\big) \doteqdot \mathcal{L}_r\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) \\ &\subseteq \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big) \doteqdot \mathcal{L}_r\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big) \, ; \end{aligned} \tag{5.3.159}$$

$$\begin{aligned} \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) &= \mathcal{R}_{\mathrm{c}}\big((\mathbf{V}_\nu)^T\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) \\ &\supseteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_S^{s,c_1c_2}\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{r}_S^{s,c_1c_2}[n_1,n_2]\big) \, ; \end{aligned} \tag{5.3.160}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{c_1c_2}\big) = \big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^\nu\big)\big)^T \, . \tag{5.3.161}$$

This concludes our discussion on the matrix SVD of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. See the next section for explicit expressions of a proper set $\Phi_{D,S}^{c_1c_2}$ (5.3.67) containing sets of coefficients for the polyconjugal equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q \in \mathcal{Q}_{D,S}^{c_1c_2}}$ (5.3.128), or a matrix $\mathbf{\Phi}$ (5.3.129) representing these coefficients, in terms of the left null space of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ found by the SVD.

5.3.8.3 Coefficients of polyconjugals in terms of SVD results

Finally, in this section we show how the coefficients of the equations in the system of homogeneous polyconjugal equations derived earlier can be expressed in terms of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)$ of $\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]$ and $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. From (5.3.130), (5.3.131), (5.3.148) and (5.3.161) it follows directly how a proper set $\Phi_{D,S}^{c_1c_2}$ (5.3.68) of sets of coefficients, or equivalently rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ of the matrix $\boldsymbol{\Phi}$, can be expressed explicitly in terms of this left null space because these equations state that $\mathcal{N}_{\mathrm{l}}\big(\mathbf{r}_{\mathrm{u},D}^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)$ is spanned by the transposed left singular vectors $\mathbf{u}^{d_D^{x,c_1c_2}+1},\ldots,\mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}$ that are contained in the set $\mathcal{U}^\nu$ and the matrix $\mathbf{U}^\nu$. Hence, proper nontrivial vectors $\tilde{\boldsymbol{\varphi}}_q$ satisfying (5.3.130) are given by:

$$\tilde{\boldsymbol{\varphi}}_q = \big(\mathbf{u}^{d_D^{x,c_1c_2}+q}\big)^H \qquad \forall\, 1\le q\le Q_{D,S}^{c_1c_2}\,, \tag{5.3.162}$$

or by arbitrary linear combinations of these vectors:

$$\tilde{\boldsymbol{\varphi}}_q = \sum_{j=d_D^{x,c_1c_2}+1}^{M_{\mathrm{u},D}^{x,c_1c_2}} \alpha_q^j\,(\mathbf{u}^j)^H \qquad \forall\,\alpha_q^j\in\mathbb{C}\,. \tag{5.3.163}$$

Equivalently, a nontrivial matrix $\boldsymbol{\Phi}$ satisfying (5.3.131) is given by (see also (5.3.129)):

$$\boldsymbol{\Phi} = (\mathbf{U}^\nu)^H = \begin{bmatrix} \big(\mathbf{u}^{d_D^{x,c_1c_2}+1}\big)^H \\ \vdots \\ \big(\mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}\big)^H \end{bmatrix}, \tag{5.3.164}$$

or any other matrix containing arbitrary linear combinations of the rows of $\boldsymbol{\Phi}$. Note that it can easily be shown that $\boldsymbol{\Phi}$ given by (5.3.164) satisfies (5.3.126):

$$\boldsymbol{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = (\mathbf{U}^\nu)^H\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} \overset{(5.3.155)}{=} (\mathbf{U}^\nu)^H\mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* = \mathbf{0}_{Q_{D,S}^{c_1c_2}}^N\,.$$

Examples of the computation of $\boldsymbol{\Phi}$ from the sensor correlation values will be given in Section 5.4.

By realizing that we can set up a one-to-one mapping between the set $\big\{\varphi_q^{i_1i_2}\big\}^{i_1i_2\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}$ of coefficients corresponding to the q-th equation in the system and the row $\tilde{\boldsymbol{\varphi}}_q$, we can write the equivalent of (5.3.164) in terms of the set $\Phi_{D,S}^{c_1c_2}$ defined in (5.3.68) by means of two equivalent formulations. Firstly, from (5.3.162) it follows that for $1\le p\le M_{\mathrm{u},D}^{x,c_1c_2}$:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{(i_1,i_2)_p} = \Big[\big(\mathbf{u}^{d_D^{x,c_1c_2}+q}\big)^H\Big]^p = \Big(u_p^{d_D^{x,c_1c_2}+q}\Big)^*\,,$$

which yields:

$$\begin{aligned}\Phi_{D,S}^{c_1c_2} &\triangleq \Big\{\big\{\varphi_q^{i_1i_2}\big\}^{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}\Big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} = \Big\{\Big\{\Big(u_p^{d_D^{x,c_1c_2}+q}\Big)^*\Big\}_{1\le p\le M_{\mathrm{u},D}^{x,c_1c_2}}\Big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} \\ &= \Big\{\Big\{\Big(u_1^{d_D^{x,c_1c_2}+1}\Big)^*,\ldots,\Big(u_{M_{\mathrm{u},D}^{x,c_1c_2}}^{d_D^{x,c_1c_2}+1}\Big)^*\Big\},\ldots, \\ &\quad \Big\{\Big(u_1^{d_D^{x,c_1c_2}+Q_{D,S}^{c_1c_2}}\Big)^*,\ldots,\Big(u_{M_{\mathrm{u},D}^{x,c_1c_2}}^{d_D^{x,c_1c_2}+Q_{D,S}^{c_1c_2}}\Big)^*\Big\}\Big\}.\end{aligned}$$

Secondly, using the mapping between the integer p and index tuple $(i_1, i_2)_p$ (5.3.99) it also follows for $1 \leq p \leq M_{\mathrm{u},D}^{x,c_1c_2}$ that:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{(i_1,i_2)_p} = \left[\left(\mathbf{u}^{d_D^{x,c_1c_2}+q}\right)^H\right]^p = \left(u_{(i_1,i_2)_p}^{d_D^{x,c_1c_2}+q}\right)^*,$$

which yields:

$$\begin{aligned}\Phi_{D,S}^{c_1c_2} &\triangleq \left\{\left\{\varphi_q^{i_1i_2}\right\}^{i_1i_2\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}\right\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} = \left\{\left\{\left(u_{(i_1,i_2)_p}^{d_D^{x,c_1c_2}+q}\right)^*\right\}_{1\leq p\leq M_{\mathrm{u},D}^{x,c_1c_2}}\right\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} \\ &= \left\{\left\{\left(u_{(i_1,i_2)_1}^{d_D^{x,c_1c_2}+1}\right)^*,\ldots,\left(u_{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}^{d_D^{x,c_1c_2}+1}\right)^*\right\},\ldots,\right. \\ &\qquad \left.\left\{\left(u_{(i_1,i_2)_1}^{d_D^{x,c_1c_2}+Q_{D,S}^{c_1c_2}}\right)^*,\ldots,\left(u_{(i_1,i_2)_{M_{\mathrm{u},D}^{x,c_1c_2}}}^{d_D^{x,c_1c_2}+Q_{D,S}^{c_1c_2}}\right)^*\right\}\right\}.\end{aligned}$$

This concludes our derivation of the coefficients of the polyconjugals in the system; see also the fourth step of Alg. 5.2 below.

Algorithm 5.2 High-level algorithm for $D \times S$ MIBI exploiting Second Order Temporal Structure with arbitrary conjugation pairs (c_1, c_2).

1: Compute/estimate sensor correlation functions in set $\mathcal{K}_{\mathrm{u},D}^{x,c_1c_2} = \left\{r_{i_1i_2}^{x,c_1c_2}[n_1,n_2]\right\}_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}}$ *for time index tuples* (n_1, n_2) *in Noise-Free ROS* $\Omega_{n_1n_2}^{s\setminus\nu,c_1c_2}$*;*

2: Arrange these values in uniquiefied subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$*;*

3: Compute Singular Value Decomposition (SVD) of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ *and split result into signal and null/noise subspace parts as follows:*

$$\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2} = \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^* = \mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* + \mathbf{U}^\nu\boldsymbol{\Sigma}_\nu^\nu(\mathbf{V}_\nu)^*;$$

4: Compute matrix $\boldsymbol{\Phi}$ *whose rows span complex conjugate left null space* $\left(\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\right)\right)^*$ *of* $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$*:*

$$\boldsymbol{\Phi} \triangleq (\mathbf{U}^\nu)^H;$$

5: With each row $\tilde{\boldsymbol{\varphi}}_q$ *of* $\boldsymbol{\Phi}$ *with* $q \in \mathcal{Q}_{D,S}^{c_1c_2}$ *associate a* D*-variate polyconjugal that is homogeneous of degree two with conjugation pair* (c_1, c_2)*:*

$$f_{D,q}^{c_1c_2}(\mathbf{z}) \triangleq \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}(z_{i_1})^{c_1}(z_{i_2})^{c_2} = \tilde{\boldsymbol{\varphi}}_q\,\check{\mathbf{z}}_{\mathrm{u},D}^{c_1c_2} = \tilde{\boldsymbol{\varphi}}_q\,\mathbf{w}_{\mathrm{u},D}^{c_1c_2}(\mathbf{z});$$

6: The following system remains to be solved for the columns of the mixing matrix:

$$\left\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\right\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}} \quad\equiv\quad \mathbf{f}_{D,S}^{c_1c_2}(\mathbf{z}) = \boldsymbol{\Phi}\,\check{\mathbf{z}}_{\mathrm{u},D}^{c_1c_2} = \boldsymbol{\Phi}\,\mathbf{w}_{\mathrm{u},D}^{c_1c_2}(\mathbf{z}) = \mathbf{0}_{Q_{D,S}^{c_l}}.$$

Note that the Singular Value Decomposition confirms once more that the number of linearly independent equations in our system is given by (5.3.67), (5.3.95), and (5.3.127). This follows directly by considering the dimension of the left null space $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}\big)$ of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$. See also (5.3.163) for example. Since the singular vectors $\mathbf{u}^{d_D^{x,c_1c_2}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,c_1c_2}}$ correspond to the zero singular values, in subspace terminology we call them noise subspace vectors or 'zero subspace vectors'. Also note from the results above the notational consistency that the transpose operation effectively turns a superscript index into a subscript index, and vice versa. Finally, note that in practice only an approximate estimated version $\hat{\mathbf{C}}_{\mathrm{u},D}^{x,c_1c_2}$ of $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$ is available. In this case, the singular values are divided into two sets, one with the d_D^{x,c_1c_2} largest ones, and the other with the remaining $M_{\mathrm{u},D}^{x,c_1c_2} - d_D^{x,c_1c_2}$ smallest ones. This implies that in the statements made above $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,c_1c_2}} > 0,\ \sigma_{d_D^{x,c_1c_2}+1}, \ldots, \sigma_P = 0$ has to be replaced with $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,c_1c_2}} > 0,\ \sigma_{d_D^{x,c_1c_2}+1}, \ldots, \sigma_P \approx 0$. This stage completes the derivation of the system of equations. The algorithmic part of the method is summarized in Alg. 5.2 on the facing page, which is a generalization of Alg. 4.3. The last step of the algorithm will be discussed in later sections.

5.4 Algebraic and geometric structure

The purpose of this section is to provide insight into the algebraic and geometric structure of the problem induced by the system $\big\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ in (5.3.73). As we have shown in Sections 5.3.4.3 and 5.3.7, the type of function we are considering is defined by (5.3.71) and (5.3.137). For convenience, we write the function definition in its most elaborate form as:

$$f_{D,q}^{c_1c_2}(z_1,\ldots,z_D) \triangleq \sum_{(i_1,i_2)\in\mathcal{I}_{\mathrm{u},D}^{c_1c_2}} \varphi_q^{i_1i_2}(z_{i_1})^{c_1}(z_{i_2})^{c_2} \qquad \forall\, z_1,\ldots,z_D \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}_{D,S}^{c_1c_2}\,. \tag{5.4.1}$$

By definition, the set of values of the tuple $(z_1,\ldots,z_D)$ for which $f_{D,q}^{c_1c_2}(z_1,\ldots,z_D) = 0$ is the zero contour level of $f_{D,q}^{c_1c_2}(z_1,\ldots,z_D)$. Hence, geometrically, finding the different solutions of the system $\big\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$ is equivalent to finding the intersections between the zero contour levels of the functions $\big\{f_{D,q}^{c_1c_2}(\mathbf{z})\big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$. In Section 5.3.4.4 we have seen that the function $f_{D,q}^{c_1c_2}(\mathbf{z})$ defined in (5.4.1) is a D-variate homogeneous polyconjugal of degree two with conjugation pair (c_1,c_2), which possesses the important *homogeneity property* formulated in (5.3.74). This latter property implies that if $\mathbf{v} \in \mathbb{C}_D$ is a solution of the system $\big\{f_{D,q}^{c_1c_2}(\mathbf{z}) = 0\big\}_{q\in\mathcal{Q}_{D,S}^{c_1c_2}}$, then so is $\eta\mathbf{v}$ for all $\eta \in \mathbb{C}$; see (5.3.75). Therefore, only solution vectors can be distinguished that are different according to the following equivalence relation $\sim$ defined on all vectors that belong to $\mathbb{C}_D$:

$$\mathbf{z} \sim \mathbf{z}' \quad \Longleftrightarrow \quad \exists\,\eta \in \mathbb{C}, \eta \neq 0 :\ \mathbf{z} = \eta\mathbf{z}'\,. \tag{5.4.2}$$

Hence, vectors that have the same or opposite directions and different lengths are considered to be equivalent according to this relation, and all other vectors are different. This is a logical result of the scaling indeterminacy discussed in Section 2.4. Now, from Definition 4.4.1 it is clear that the zero contour level of each function $f_{D,q}^{c_1c_2}(z_1,\ldots,z_D)$ defines a cone in the D-dimensional Euclidian space $\mathbb{C}_D$. Thus:

Geometrically, solving the system $\left\{ f_{D,q}^{c_1 c_2}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$ is equivalent to finding the one-dimensional intersections between $Q_{D,S}^{c_1 c_2}$ cones in a D-dimensional Euclidian space that represent the zero contour levels of the functions $\left\{ f_{D,q}^{c_1 c_2}(\mathbf{z}) \right\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$.

Ideally, each intersection between the cones is a one-dimensional linear subspace that corresponds to a column of the mixing matrix. Our purpose in solving (5.3.73) is to find all non-zero solutions $\mathbf{z}^1, \ldots, \mathbf{z}^S$ that are different according to relation (5.4.2).

Now we are in a position to discuss several examples that will elucidate the theory that has been presented so far in this chapter. The examples focus on highlighting algebraic and geometric structure. The actual estimation of the mixing matrix columns and, if applicable, the separated signals will be presented in Section 5.5. For convenience, and without loss of generality, all involved array response vectors have unit Euclidian norm. Furthermore, for ease of visualization we only consider real-valued systems and signals. Therefore, the conjugation pair (c_1, c_2) is irrelevant and will be omitted from the notation or replaced by the number 2 representing the employed statistical order. See Section 7.1.4.1 for an example with a complex-valued scenario employing conjugation pair $(c_1, c_2) = (\circ, *)$.

5.4.1 Real-valued scenarios with three sensors, stationary AR(1) source signals, and additive white noise

In this section we examine and perform the various steps of Alg. 5.2 on page 268 for two real-valued MIBI examples with signal scenarios and Regions Of Support as described in Section 5.2.2. Among other things, this means that we assume stationary AR(1) source signals and additive white noise. The first example deals with the 3×3 and the second with the 3×4 mixing case, i.e. both examples employ three sensors but one of them deals with three and the other with four sources. Substituting $D = 3$, $l = 2$, and $c_1 = c_2 = \circ$ into (5.3.73) and (5.4.1) respectively it follows that the system of equations is given by:

$$\left\{ f_{3,q}^{2}(z_1, z_2, z_3) = 0 \right\}_{q \in \mathcal{Q}_{3,S}^{2}} \tag{5.4.3}$$

and that its functions take the following *quadratic form*:

$$\begin{aligned} f_{3,q}^{2}(z_1, z_2, z_3) = \sum_{(i_1, i_2) \in \mathcal{I}_{\mathrm{a},3}^{2}} \varphi_q^{i_1 i_2} z_{i_1} z_{i_2} = \varphi_q^{11} z_1 z_1 + \varphi_q^{12} z_1 z_2 + \varphi_q^{13} z_1 z_3 + \varphi_q^{22} z_2 z_2 \\ + \varphi_q^{23} z_2 z_3 + \varphi_q^{33} z_3 z_3 \qquad \forall\, z_1, z_2, z_3 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}_{3,S}^{2}\,. \end{aligned} \tag{5.4.4}$$

The set $\mathcal{Q}_{3,S}^{2} \triangleq \mathcal{Q}_{3,S}^{\circ\circ}$ is defined by (5.3.66) and contains the integers $1, \ldots, Q_{3,S}^{2}$, where the number $Q_{3,S}^{2} \triangleq Q_{3,S}^{\circ\circ}$ of linearly independent equations in (5.4.3) is given by (5.3.67), (5.3.95), and/or (5.3.127). For real-valued scenarios with D sensors the number $M_{\mathrm{u},D}^{x,2} \triangleq M_{\mathrm{u},D}^{x,\circ\circ} \equiv M_{\mathrm{a},D}^{x,2}$ of unique rows of the subspace matrix $\mathbf{C}_D^{x,2} \triangleq \mathbf{C}_D^{x,\circ\circ}$, which equals the number $M_{\mathrm{u},D}^{2} \triangleq M_{\mathrm{u},D}^{\circ\circ} \equiv M_{\mathrm{a},D}^{2}$ of unique rows of the second order Khatri-Rao product $\mathbf{A}_{D,\diamond}^{2} \triangleq \mathbf{A}_{D,\diamond}^{\circ\circ}$, is given by (5.3.14) and (5.3.38) as $\frac{1}{2}D(D+1)$. The corresponding index set $\mathcal{I}_{\mathrm{u},D}^{2} \equiv \mathcal{I}_{\mathrm{a},D}^{2}$ is given in (5.3.13). See the example just below (5.3.14) for the case with $D = 3$. In the 3×3 and 3×4 MIBI examples that will be presented shortly the employed

mixing matrices have full rank D, i.e. $\mathrm{rank}(\mathbf{A}) = D = 3$, while their corresponding Khatri-Rao products have full rank S, i.e. $\mathrm{rank}(\mathbf{A}^2_{\mathrm{u},3,\diamond}) = S$. Hence, according to (5.3.123) the signal subspace dimension $d_3^{x,2} \triangleq d_3^{x,\circ\circ} = \mathrm{rank}(\mathbf{C}^{x,2}_{\mathrm{u},3}) = \mathrm{rank}(\mathbf{A}^2_{\mathrm{u},3,\diamond})$ also equals S. Substituting those results into (5.3.67), (5.3.95), or (5.3.127), it follows that the number of linearly independent equations in (5.4.3) equals:

$$Q^2_{3,S} = \tfrac{1}{2} \cdot 3 \cdot (3+1) - d_3^{x,2} = 6 - d_3^{x,2} = 6 - \mathrm{rank}(\mathbf{A}^2_{\mathrm{u},3,\diamond}) = 6 - S\,, \tag{5.4.5}$$

i.e. $\mathcal{Q}^2_{3,S} = \{1, \ldots, 6-S\}$. Using (5.3.139) the system in (5.4.3) can be written as follows:

$$\mathbf{f}^2_{3,S}(\mathbf{z}) = \boldsymbol{\Phi}\, \check{\mathbf{z}}^2_{\mathrm{u},3} = \boldsymbol{\Phi}\, \mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) = \mathbf{0}_{Q^2_{3,S}}\,, \tag{5.4.6}$$

where $\check{\mathbf{z}}^2_{\mathrm{u},3} = \check{\mathbf{z}}^{\circ\circ}_{\mathrm{u},3}$ and $\mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) = \mathbf{w}^{\circ\circ}_{\mathrm{u},3}(\mathbf{z})$ are given by (5.3.34), and $\boldsymbol{\Phi}$ is a $Q^2_{3,S} \times 6$ matrix. In the following sections we will demonstrate that (5.4.3) as derived according to Alg. 5.2 can determine up to four mixing matrix columns. In Section 5.4.1.1 we first analyze the example with three sources followed in Section 5.4.1.2 by the example with four sources. We use the values $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$, and $\rho_4 = -0.4$ for the AR(1) regression coefficients.

5.4.1.1 Three sensors, three sources, and additive white noise

In this section we first analyze and consider the (results of the) various steps of Alg. 5.2 for the scenario with $D = S = 3$ for a subspace matrix that is known exactly. Then, we give a similar analysis for a subspace matrix that is estimated from simulated sensor data.

Ideal subspace matrix

Let the full rank mixing matrix be given by:

$$\mathbf{A} = \begin{bmatrix} \frac{1}{\sqrt{3}}\begin{bmatrix}1\\1\\1\end{bmatrix} & \frac{1}{\sqrt{6}}\begin{bmatrix}1\\-2\\-1\end{bmatrix} & \frac{1}{5\sqrt{3}}\begin{bmatrix}7\\1\\-5\end{bmatrix} \end{bmatrix} \approx \begin{bmatrix} 0.5774 & 0.4082 & 0.8083 \\ 0.5774 & -0.8165 & 0.1155 \\ 0.5774 & -0.4082 & -0.5774 \end{bmatrix}, \tag{5.4.7}$$

then the uniquiefied second order Khatri-Rao product of $\mathbf{A}$ can now be computed by substituting this matrix into (5.3.55) while using $D = S = 3$ and $c_1 = c_2 = \circ$:

$$\mathbf{A}^2_{\mathrm{u},3,\diamond} = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 & a_1^3 a_1^3 \\ a_1^1 a_2^1 & a_1^2 a_2^2 & a_1^3 a_2^3 \\ a_1^1 a_3^1 & a_1^2 a_3^2 & a_1^3 a_3^3 \\ a_2^1 a_2^1 & a_2^2 a_2^2 & a_2^3 a_2^3 \\ a_2^1 a_3^1 & a_2^2 a_3^2 & a_2^3 a_3^3 \\ a_3^1 a_3^1 & a_3^2 a_3^2 & a_3^3 a_3^3 \end{bmatrix}. \tag{5.4.8}$$

The ideal value of the uniquiefied subspace matrix $\mathbf{C}^{x,2}_{\mathrm{u},3}$ can be computed by means of (5.3.113) as follows:

$$\mathbf{C}^{x,2}_{\mathrm{u},3} \triangleq \begin{bmatrix} r^x_{11}[1] & r^x_{11}[2] & r^x_{11}[3] \\ r^x_{12}[1] & r^x_{12}[2] & r^x_{12}[3] \\ r^x_{13}[1] & r^x_{13}[2] & r^x_{13}[3] \\ r^x_{22}[1] & r^x_{22}[2] & r^x_{22}[3] \\ r^x_{23}[1] & r^x_{23}[2] & r^x_{23}[3] \\ r^x_{33}[1] & r^x_{33}[2] & r^x_{33}[3] \end{bmatrix} = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 & a_1^3 a_1^3 \\ a_1^1 a_2^1 & a_1^2 a_2^2 & a_1^3 a_2^3 \\ a_1^1 a_3^1 & a_1^2 a_3^2 & a_1^3 a_3^3 \\ a_2^1 a_2^1 & a_2^2 a_2^2 & a_2^3 a_2^3 \\ a_2^1 a_3^1 & a_2^2 a_3^2 & a_2^3 a_3^3 \\ a_3^1 a_3^1 & a_3^2 a_3^2 & a_3^3 a_3^3 \end{bmatrix} \begin{bmatrix} r^s_1[1] & r^s_1[2] & r^s_1[3] \\ r^s_2[1] & r^s_2[2] & r^s_2[3] \\ r^s_3[1] & r^s_3[2] & r^s_3[3] \end{bmatrix}, \tag{5.4.9}$$

where the source auto-correlation matrix $\mathbf{C}^{s,2}_3$ consists of the upper-left block of size 3×3 of $\mathbf{C}^{s,2}_4$ defined in (5.2.16) with $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$. Using this matrix in Alg. 5.2

the system to be solved follows from the last step of the algorithm. Computing the ranks of $\mathbf{A}$ and $\mathbf{A}^2_{\mathrm{u},3,\diamond}$ shows that both ranks equal $D = S = 3$. Hence, according to (5.4.5) the number of linearly independent equations equals 3, i.e. $\mathcal{Q}^2_{3,3} = \{1, 2, 3\}$. Thus, the system contains three equations $f^2_{3,1}(z_1, z_2, z_3) = 0$, $f^2_{3,2}(z_1, z_2, z_3) = 0$, and $f^2_{3,3}(z_1, z_2, z_3) = 0$ with functions of the form specified by (5.4.4). Using (5.4.6) this system can be written as:

$$\mathbf{f}^2_{3,3}(\mathbf{z}) = \mathbf{\Phi}\, \check{\mathbf{z}}^2_{\mathrm{u},3} = \mathbf{\Phi}\, \mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) = \mathbf{0}_3\,, \tag{5.4.10}$$

where now $\mathbf{\Phi}$ is a 3×6 matrix. In elaborate form this reads:

$$\begin{bmatrix} f^2_{3,1}(z_1, z_2, z_3) \\ f^2_{3,2}(z_1, z_2, z_3) \\ f^2_{3,2}(z_1, z_2, z_3) \end{bmatrix} \triangleq \begin{bmatrix} \varphi_1^{11} & \varphi_1^{12} & \varphi_1^{13} & \varphi_1^{22} & \varphi_1^{23} & \varphi_1^{33} \\ \varphi_2^{11} & \varphi_2^{12} & \varphi_2^{13} & \varphi_2^{22} & \varphi_2^{23} & \varphi_2^{33} \\ \varphi_3^{11} & \varphi_3^{12} & \varphi_3^{13} & \varphi_3^{22} & \varphi_3^{23} & \varphi_3^{33} \end{bmatrix} \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_1 z_3 \\ z_2 z_2 \\ z_2 z_3 \\ z_3 z_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \tag{5.4.11}$$

From steps 3 and 4 of Alg. 5.2 it follows that a valid matrix $\mathbf{\Phi}$ is given by:

$$\mathbf{\Phi} = \begin{bmatrix} -0.5070 & 0.8397 & -0.6959 & 0.2931 & 0.2480 & -0.1777 \\ 0.0869 & 0.1324 & 0.2523 & 0.6438 & -1.0296 & -0.0859 \\ 0.4602 & 0.1044 & 0.0106 & 0.0227 & 0.2663 & -0.8641 \end{bmatrix}.$$

The zero contour levels of $f^2_{3,1}(z_1, z_2, z_3)$, $f^2_{3,2}(z_1, z_2, z_3)$ and $f^2_{3,3}(z_1, z_2, z_3)$ are shown in Fig. 5.5 in different colors/grey shades. Each zero contour level is a two-dimensional quadratic cone in three-dimensional space. In general, the zero contour level of any homogeneous polynomial equation of degree two defines a so-called *quadratic* or *quadric hypersurface* [8, 59]. In this particular case, due to the nature of our problem the quadric hypersurface is an *elliptic cone*. All cones in Fig. 5.5 have been plotted in such a way that no points have a distance larger than one from the origin. Hence, the edges at the two openings of each cone are intersections between the involved cone and the unit sphere. Note that each cone, and thus also its intersection with the unit sphere, consists of two parts that are point-symmetric with respect to the origin. In general, if a quadric cone is intersected with a plane, the intersection defines an ellipse. In the current situation, where each cone is intersected with the unit sphere, the intersection defines a pair of curves that are a kind of 'ellipses on the sphere', opposite to each other. In the sequel, we will refer to those 'ellipses on the sphere' as *'spherical ellipses'*. The black arrows in Fig. 5.5 represent the ideal columns of the mixing matrix multiplied by 1.5, whereas the grey arrows are the opposites of the black arrows. The solid dots indicate the intersections of the columns of the mixing matrix with the unit sphere. It can be seen that the arrows representing the (opposites of the) columns of $\mathbf{A}$ are exactly in the directions of the lines describing the one-dimensional intersections between the three cones. Hence, the columns are determined uniquely by the intersections induced by the system of equations. To achieve a better visualization, instead of plotting the complete cones defined by the zero contour levels of the functions in the system, in Fig. 5.6 we have depicted only the spherical ellipses defined by the intersections of the cones with the unit sphere. Each pair of spherical ellipses is plotted in a different color/grey shade, while two opposite spherical ellipses in each pair have the same color/grey shade. It can be seen that the arrows representing the columns of $\mathbf{A}$, which now have unit length, point exactly to the points where three spherical ellipses intersect. Again, this implies that the directions of the lines describing the one-dimensional intersections between the three cones correspond to the mixing matrix columns.

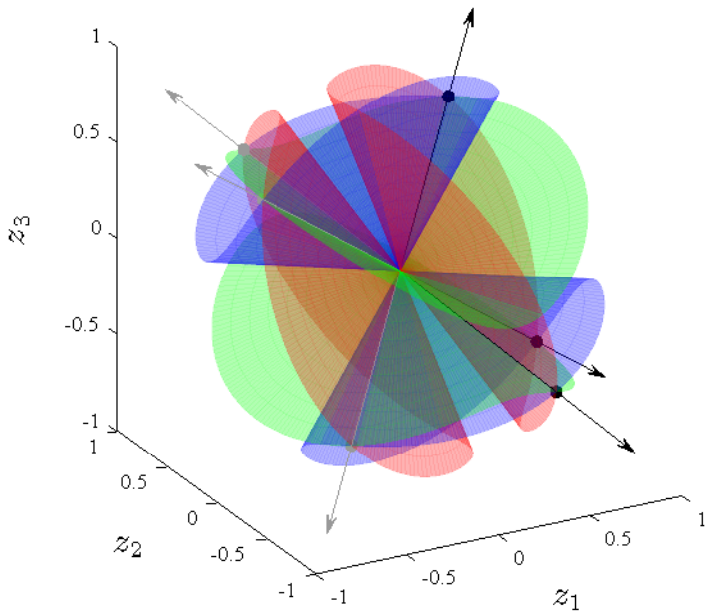

Figure 5.5: *Zero contour level cones of $f^2_{3,1}(z_1, z_2, z_3)$, $f^2_{3,2}(z_1, z_2, z_3)$ and $f^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with ideal subspace matrix.*

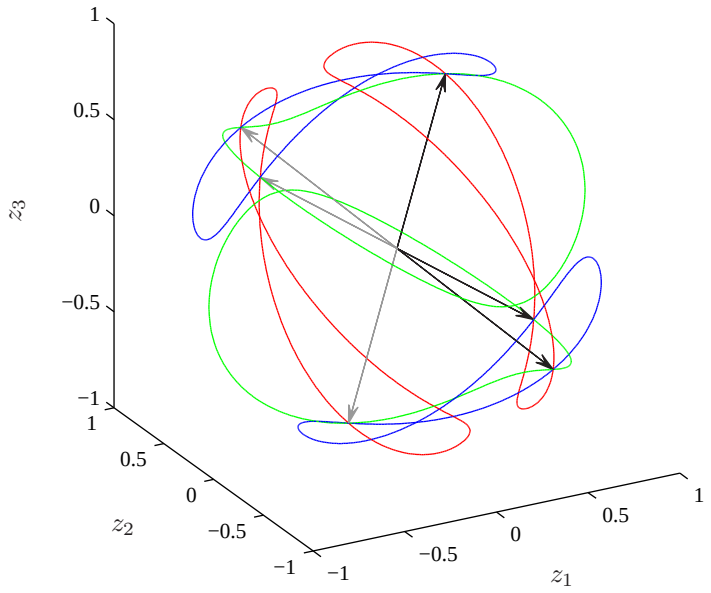

Figure 5.6: *Zero contour spherical ellipses of $f^2_{3,1}(z_1, z_2, z_3)$, $f^2_{3,2}(z_1, z_2, z_3)$ and $f^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with ideal subspace matrix.*

Employing subspace matrix estimated from simulated sensor data

Now, similarly to e.g. Sections 4.4.2.2 and 4.4.3.2 we compute and plot the contour levels for a subspace matrix estimated from computer generated sensor signals. The three source signals $s_1[n]$, $s_2[n]$ and $s_3[n]$ are unit variance Gaussian signals consisting of $N_s = 20000$ time samples. To this end the sequences $w_1[n]$, $w_2[n]$ and $w_3[n]$ in model (5.2.13) are generated independently of each other as i.i.d. Gaussian random variables with variances chosen in such a way that the source signals have unit variance. As above, the AR(1) regression coefficients of the three source signals are given by $\rho_1 = 0.9$, $\rho_2 = -0.8$ and $\rho_3 = 0.6$. The sensor signals $x_1[n]$, $x_2[n]$ and $x_3[n]$ are computed according to (5.1.1)/(5.1.2), where $\mathbf{A}$ is given by (5.4.7) and the noise signals $\nu_1[n]$, $\nu_2[n]$ and $\nu_3[n]$ are mutually statistically independent white Gaussian noise sequences with variance 0.1. For a specific realization, Fig. 5.7 shows 200 samples of the source signals at the left side and their noisy mixtures at the right side. Using formula (4.4.18) with $k = 1, 2, 3$ for estimating the subspace matrix from the sensor signals gives an estimate $\hat{\mathbf{C}}_{\mathrm{u},3}^{x,2}$ of $\mathbf{C}_{\mathrm{u},3}^{x,2}$ that can be used in Alg. 5.2 on page 268 from step 2

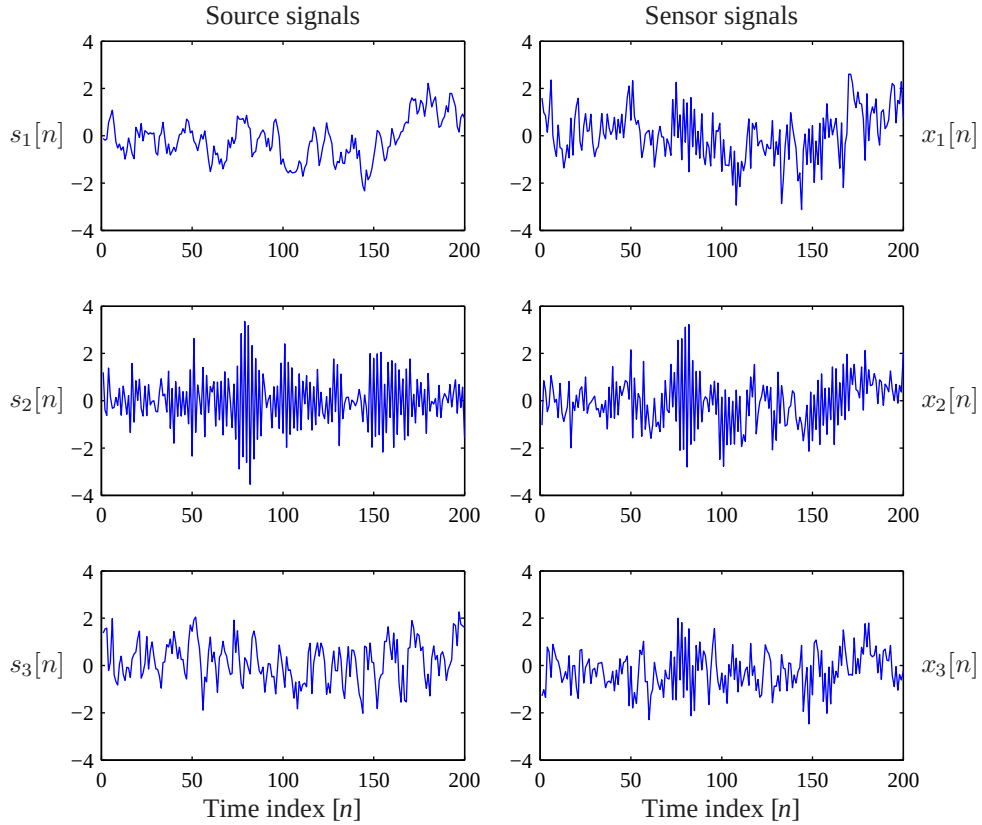

Figure 5.7: *Stationary Gaussian $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 3×3 example.*

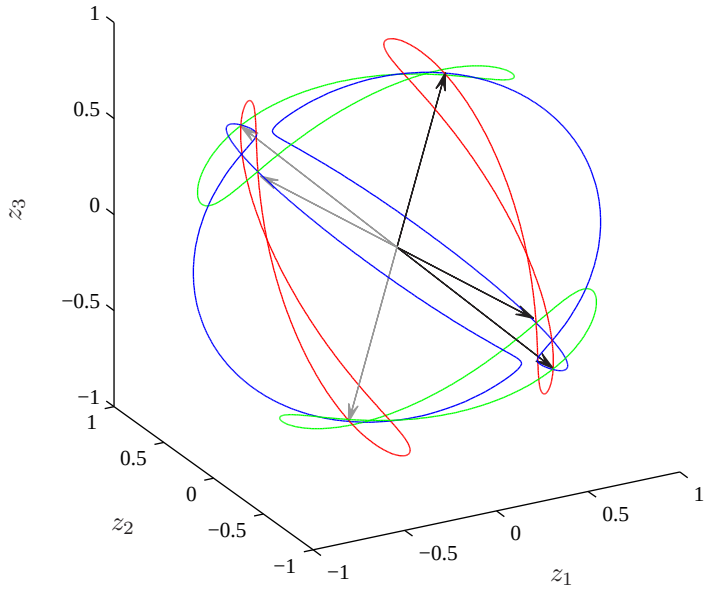

Figure 5.8: *Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,3}(z_1, z_2, z_3)$ for 3×3 AR(1) example with estimated subspace matrix.*

on. Again, steps 3 and 4 of this algorithm yield the following estimate of the matrix $\mathbf{\Phi}$:

$$\hat{\mathbf{\Phi}} = \begin{bmatrix} -0.5726 & 0.8072 & -0.7013 & 0.2897 & 0.1776 & -0.0273 \\ 0.3451 & 0.2153 & -0.1691 & -0.0539 & 0.4950 & -0.8474 \\ -0.1772 & -0.1546 & -0.1898 & -0.6379 & 0.9684 & 0.2506 \end{bmatrix}.$$

Note that the rows of $\hat{\mathbf{\Phi}}$ are not necessarily estimates of the rows of $\mathbf{\Phi}$. The only requirement is that the subspace spanned by the rows of $\hat{\mathbf{\Phi}}$ approximates the subspace spanned by the rows of $\mathbf{\Phi}$ because this implies that the left null space of the estimated subspace matrix approximates the left null space of the ideal subspace matrix. This can easily shown to be the case by computing and comparing the reduced row echelon forms of $\hat{\mathbf{\Phi}}$ and $\mathbf{\Phi}$. Similarly to the ideal case, the spherical ellipses defined by the intersections of the zero contour level cones of the estimated functions $\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,3}(z_1, z_2, z_3)$ on the one hand, and the unit sphere on the other hand, are shown in Fig. 5.8 in different colors/grey shades (compare this figure with Fig. 5.6). Although the three estimated spherical ellipses now do not intersect each other exactly at the true source array response vectors indicated by the black and grey arrows, they do so approximately. Hence, the columns of $\mathbf{A}$ are determined uniquely and approximately by the intersections induced by the system of equations. In Sections 5.5 and 5.6 we will present methods for solving the system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3) = 0$, $\hat{f}^2_{3,2}(z_1, z_2, z_3) = 0$, $\hat{f}^2_{3,3}(z_1, z_2, z_3) = 0\}$.

5.4.1.2 Three sensors, four sources, and additive white noise

Similarly to the previous section, here we first analyze and consider the (results of the) various steps of Alg. 5.2 for the scenario with $D = 3$ and $S = 4$ for a subspace matrix that is known exactly. Then, we give a similar analysis for a subspace matrix that is estimated from simulated sensor data.

Ideal subspace matrix

We use the following full rank mixing matrix that is obtained by appending one column to the matrix in (5.4.7):

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix} \frac{1}{\sqrt{3}}\begin{bmatrix}1\\1\\1\end{bmatrix} & \frac{1}{\sqrt{6}}\begin{bmatrix}1\\-2\\-1\end{bmatrix} & \frac{1}{5\sqrt{3}}\begin{bmatrix}7\\1\\-5\end{bmatrix} & \frac{1}{\sqrt{35}}\begin{bmatrix}-1\\-3\\5\end{bmatrix}\end{bmatrix} \\ &\approx \begin{bmatrix}0.5774 & 0.4082 & 0.8083 & -0.1690\\ 0.5774 & -0.8165 & 0.1155 & -0.5071\\ 0.5774 & -0.4082 & -0.5774 & 0.8452\end{bmatrix}. \end{aligned} \tag{5.4.12}$$

The uniquiefied second order Khatri-Rao product of $\mathbf{A}$ now takes the following form:

$$\mathbf{A}^2_{\mathrm{u},3,\diamond} = \begin{bmatrix} a_1^1 a_1^1 & a_1^2 a_1^2 & a_1^3 a_1^3 & a_1^4 a_1^4 \\ a_1^1 a_2^1 & a_1^2 a_2^2 & a_1^3 a_2^3 & a_1^4 a_2^4 \\ a_1^1 a_3^1 & a_1^2 a_3^2 & a_1^3 a_3^3 & a_1^4 a_3^4 \\ a_2^1 a_2^1 & a_2^2 a_2^2 & a_2^3 a_2^3 & a_2^4 a_2^4 \\ a_2^1 a_3^1 & a_2^2 a_3^2 & a_2^3 a_3^3 & a_2^4 a_3^4 \\ a_3^1 a_3^1 & a_3^2 a_3^2 & a_3^3 a_3^3 & a_3^4 a_3^4 \end{bmatrix}, \tag{5.4.13}$$

i.e. also one column is appended to the Khatri-Rao product (5.4.8). Again, the ideal value of the uniquiefied subspace matrix $\mathbf{C}^{x,2}_{\mathrm{u},3}$ is computed by means of (5.3.113) using $\mathbf{A}^2_{\mathrm{u},3,\diamond}$ from (5.4.13) and $\mathbf{C}^{s,2}_4$ from (5.2.16) with $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$, and $\rho_4 = -0.4$. Using this matrix in Alg. 5.2 the system to be solved follows from the last step of the algorithm. Computing the rank of $\mathbf{A}^2_{\mathrm{u},3,\diamond}$ in (5.4.13) for the values in (5.4.12) shows that $\mathrm{rank}(\mathbf{A}^2_{\mathrm{u},3,\diamond}) = S = 4$. Substituting this into (5.4.5) it follows that the number of linearly independent equations equals 2, i.e. $\mathcal{Q}^2_{3,4} = \{1, 2\}$. Thus, the system now contains the two equations $f^2_{3,1}(z_1, z_2, z_3) = 0$ and $f^2_{3,2}(z_1, z_2, z_3) = 0$, where again the form of the functions is specified by (5.4.4). Using (5.4.6), similarly to (5.4.10) the system can be written as:

$$\mathbf{f}^2_{3,4}(\mathbf{z}) = \mathbf{\Phi}\, \check{\mathbf{z}}^2_{\mathrm{u},3} = \mathbf{\Phi}\, \mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) = \mathbf{0}_2\,, \tag{5.4.14}$$

where now $\mathbf{\Phi}$ is a 2×6 matrix. Similarly to (5.4.11), the elaborate form of (5.4.14) becomes:

$$\begin{bmatrix} f^2_{3,1}(z_1, z_2, z_3) \\ f^2_{3,2}(z_1, z_2, z_3) \end{bmatrix} \triangleq \begin{bmatrix} \varphi_1^{11} & \varphi_1^{12} & \varphi_1^{13} & \varphi_1^{22} & \varphi_1^{23} & \varphi_1^{33} \\ \varphi_2^{11} & \varphi_2^{12} & \varphi_2^{13} & \varphi_2^{22} & \varphi_2^{23} & \varphi_2^{33} \end{bmatrix} \begin{bmatrix} z_1 z_1 \\ z_1 z_2 \\ z_1 z_3 \\ z_2 z_2 \\ z_2 z_3 \\ z_3 z_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{5.4.15}$$

The zero contour level cones of $f^2_{3,1}(z_1, z_2, z_3)$ and $f^2_{3,2}(z_1, z_2, z_3)$, and the corresponding spherical ellipses, are shown in Figures 5.9 and 5.10 respectively in different colors/grey shades. Again, the arrows representing the columns of $\mathbf{A}$ are exactly in the directions of the lines describing the one-dimensional intersections between the two cones. Hence, the columns are determined uniquely by the intersections induced by the system of equations.

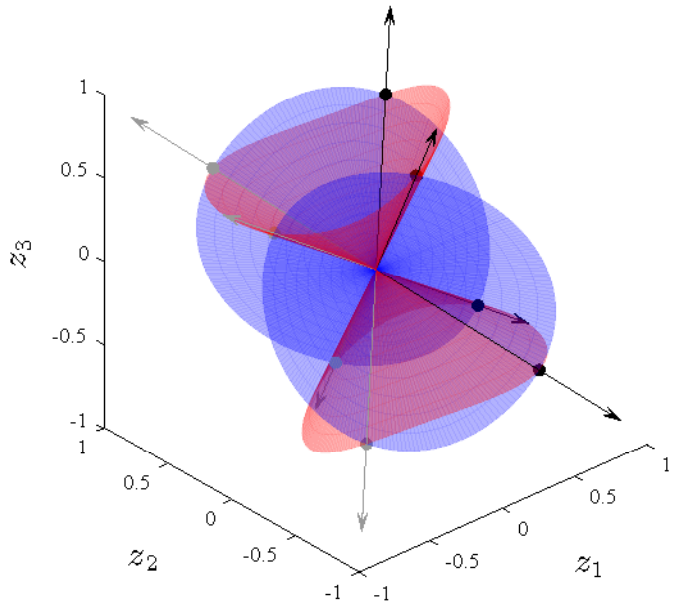

Figure 5.9: *Zero contour level cones of $f^2_{3,1}(z_1, z_2, z_3)$ and $f^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with ideal subspace matrix.*

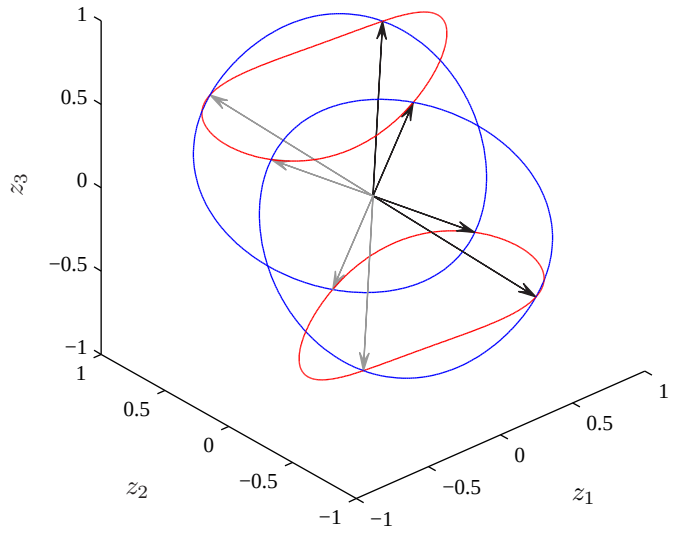

Figure 5.10: *Zero contour spherical ellipses of $f^2_{3,1}(z_1, z_2, z_3)$ and $f^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with ideal subspace matrix.*

Employing subspace matrix estimated from simulated sensor data

As in Section 5.4.1.1 we now compute and plot the contour levels in case the subspace matrix estimated from computer generated sensor signals. The four source signals $s_1[n]$, $s_2[n]$, $s_3[n]$ and $s_4[n]$ are unit variance Gaussian signals consisting of $N_s = 20000$ time samples. The AR(1) regression coefficients of the source signals are given by $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$, and $\rho_4 = -0.4$. The sensor signals $x_1[n]$, $x_2[n]$ and $x_3[n]$ are computed according to (5.1.1)/(5.1.2), where $\mathbf{A}$ is now given by (5.4.12) and again the noise signals $\nu_1[n]$, $\nu_2[n]$ and $\nu_3[n]$ are mutually statistically independent white Gaussian noise sequences with variance 0.1. For a specific realization, Fig. 5.11 shows 200 samples of the source signals at the left

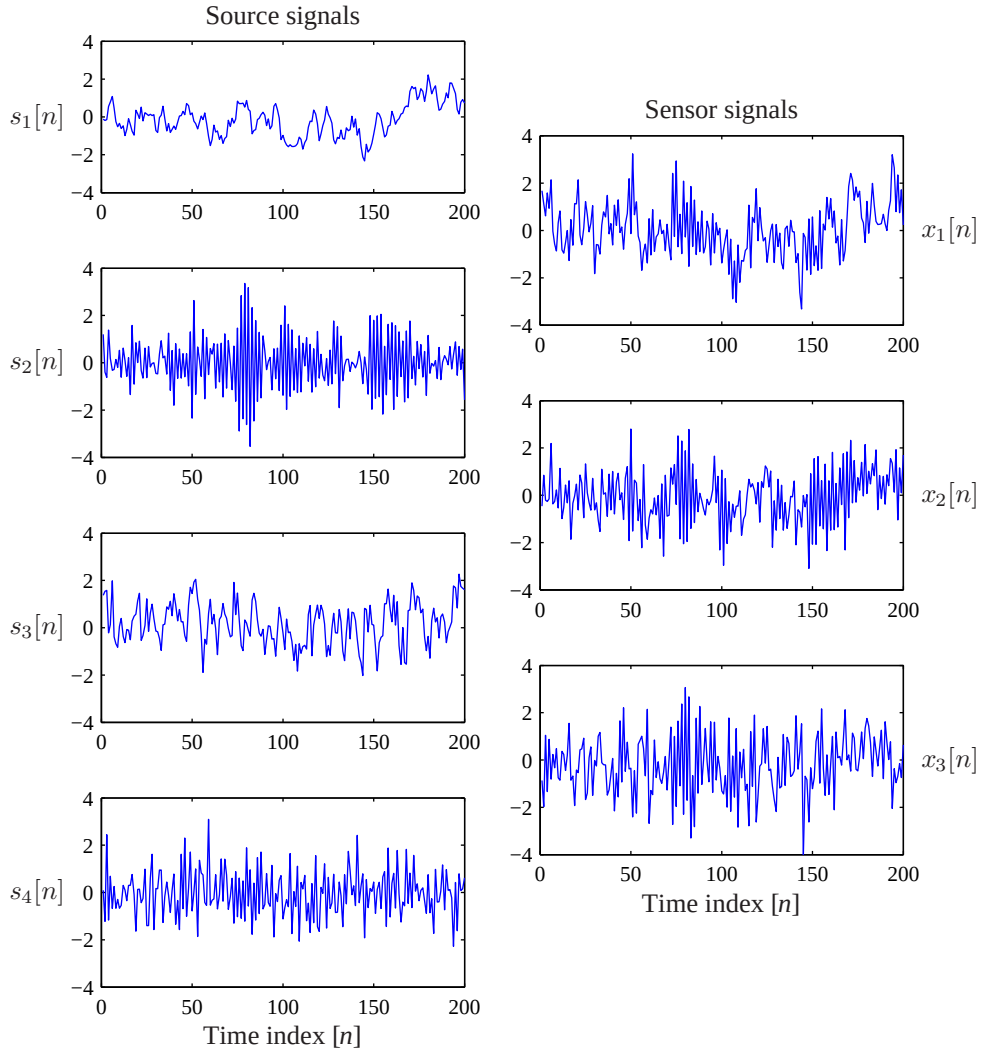

Figure 5.11: *Stationary Gaussian $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 3×4 example.*

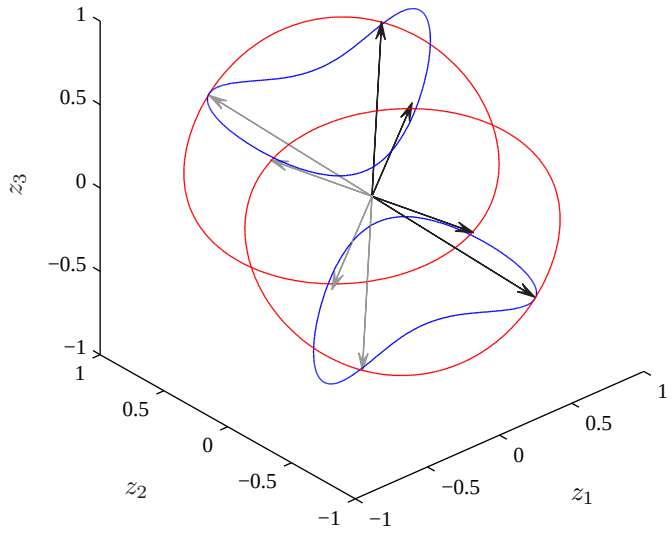

Figure 5.12: *Zero contour spherical ellipses of $\hat{f}^2_{3,1}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ for 3×4 AR(1) example with estimated subspace matrix.*

side and their noisy mixtures at the right side. Using formula (4.4.18) with $k = 1, 2, 3$ for estimating the subspace matrix from the sensor signals gives an estimate $\hat{\mathbf{C}}^{x,2}_{\mathrm{u},3}$ of $\mathbf{C}^{x,2}_{\mathrm{u},3}$ that can be used in Alg. 5.2 on page 268 from step 2 on. Again, steps 3 and 4 of this algorithm yield an estimate of the matrix $\mathbf{\Phi}$. Similarly to the ideal case, the spherical ellipses defined by the intersections of the zero contour level cones of the estimated functions $\hat{f}^2_{3,1}(z_1, z_2, z_3)$ and $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ on the one hand, and the unit sphere on the other hand, are shown in Fig. 5.12 in different colors/grey shades (compare this figure with Fig. 5.10). Although the two estimated spherical ellipses now do not intersect each other exactly at the true source array response vectors indicated by the black and grey arrows, they do so approximately. Hence, the columns of $\mathbf{A}$ are determined uniquely and approximately by the intersections induced by the system of equations. In Sections 5.5 and 5.6 we will present methods for solving the system $\{\hat{f}^2_{3,1}(z_1, z_2, z_3) = 0, \hat{f}^2_{3,2}(z_1, z_2, z_3) = 0\}$.

5.4.1.3 Comparison and number of identifiable columns for three sensors

As we have seen in the examples above, geometrically, solving a system consisting of homogeneous polynomial equations of degree two in three variables is equivalent to finding the intersections between the corresponding two-dimensional quadric hypersurfaces that are embedded in a three-dimensional Euclidian space and take the form of elliptic cones. Ideally, each intersection is a one-dimensional subspace that corresponds to a column of the mixing matrix. In a three-dimensional Euclidian space at least two two-dimensional surfaces are required for uniquely defining one-dimensional intersections. Algebraically, this means that the projectively different solutions of a system of two homogeneous polynomial equations in

three variables in general are one-dimensional. Since for the example with three sources the system consists of three equations, the one-dimensional solution subspaces corresponding to the three columns of the mixing matrix are overdetermined (three instead of two surfaces intersect each other at the one-dimensional linear solution subspaces). In other words, one degree of freedom is left unused, which, as we have seen in Section 5.4.1.2, can be exploited to determine one more column, thereby reducing the number of equations by one. Because at least two cones are required for determining one-dimensional solution subspaces, no more than four columns can be identified with three sensors and second order statistics. If $\mathbf{A}$ would contain five columns, then system (5.4.3) would consist of only one equation defining a two-dimensional solution set. Note that the intersections between 'conventional' elliptic cones are related uniquely to corresponding intersections between the associated ellipses on the sphere. Hence, another way to see that the maximum number of identifiable columns equals four is to realize that the maximum number of different intersections between two (sphere) ellipses equals four. In Section 6.6, we present a systematic approach for determining the maximum number of identifiable columns. A generalization of the algebraic and geometric considerations given in this section is presented in Section 6.3.

5.4.2 Real-valued scenarios with three sensors, speech signals, additive white noise, and estimated subspace matrix

In this section, we consider the same two real-valued examples with three sensors and additive white noise as in Section 5.4.1 but with the AR(1) source signals replaced by speech source signals. Again, the various steps of Alg. 5.2 are examined and performed for the 3×3 and 3×4 MIBI scenarios with the mixing matrix given by (5.4.7) and (5.4.12) respectively. We only show the results for subspace matrices estimated from the sensor data. Contrary to the example with speech signals in Section 4.4.4, where we only exploited the non-whiteness of the speech signals, here we will exploit both the non-stationarity and non-whiteness by using the sensor correlation values for different times and different lags. All speech source signals are sampled at 8 kHz. As in the previous section the noise signals $\nu_1[n]$, $\nu_2[n]$ and $\nu_3[n]$ are mutually statistically independent white Gaussian noise sequences with variance 0.1. The number of time samples in all signals involved is 10000, which equals 1250 milliseconds. To exploit the non-stationarity as well as the non-whiteness by using sensor correlation values for different times and different lags, the signal sequences are partitioned into disjoint blocks consisting of 2000 samples. In order to get an idea of the non-stationarity and non-whiteness of speech signals we have depicted the two-dimensional (lag- and block-dependent) auto-correlation functions of the first and second source signals used in the following examples (see the first two signals at the left of Figures 5.15 and 5.17) as a function of time and lag in Figures 5.13 and 5.14 respectively.

The subspace matrix is constructed from the one-dimensional sensor correlation functions computed for lags 1, 2 and 3 at each block. Lag zero is omitted because the corresponding correlation values are noise-contaminated. Because the number of available samples is 10000, the number of blocks equals 5. Hence, in total for each sensor correlation function 15 values are estimated and employed. Calling the block index b with $1 \leq b \leq 5$, and the lag index k with $1 \leq k \leq 3$, the employed Noise-Free ROS in the domain of block-lag pairs $(b;k)$ can be written as follows:

$$\Omega_{b;k}^{s\setminus\nu} = \big\{(1;1),(1;2),(1;3),\ldots,(5;1),(5;2),(5;3)\big\}\,.$$

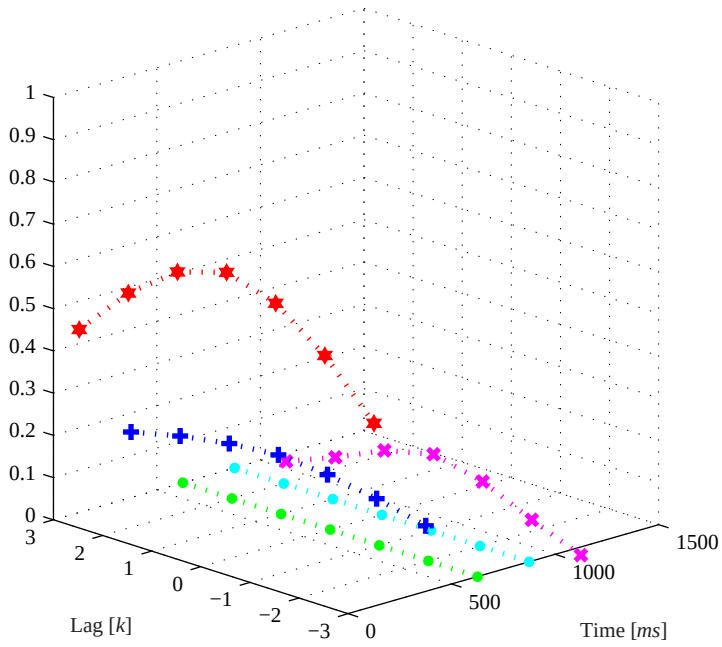

Figure 5.13: *Two-dimensional auto-correlation function of first source signal used in speech examples.*

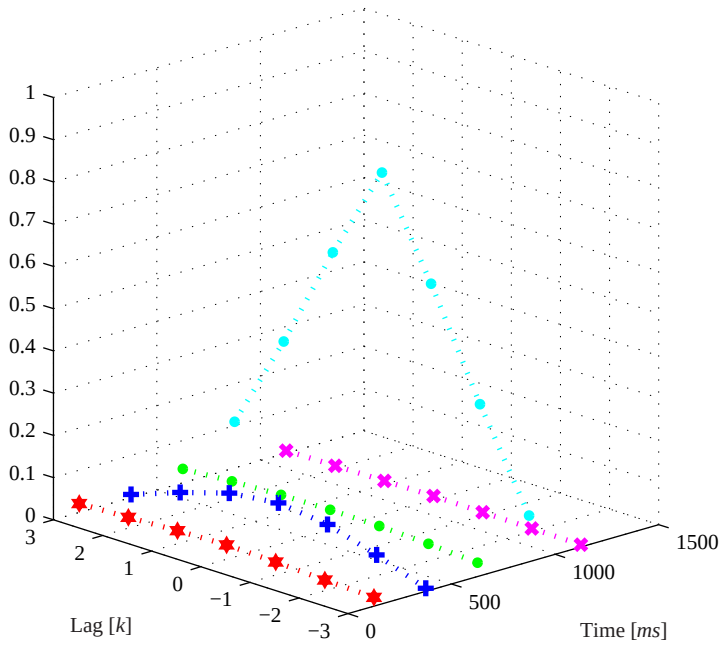

Figure 5.14: *Two-dimensional auto-correlation function of second source signal used in speech examples.*

For each block the sensor correlation values are estimated from the sensor signals by means of (4.4.18) with $k = 1, 2, 3$, and the averaging length N_s equal to the block length $N_s = 2000$. Both for $S = 3$ and $S = 4$ the estimated subspace matrix $\mathbf{C}_{\mathrm{u},3}^{x,2}$ is then constructed as follows:

$$\hat{\mathbf{C}}_{\mathrm{u},3}^{x,2} \triangleq \begin{bmatrix} \hat{r}_{11}^x[1;0] & \hat{r}_{11}^x[1;1] & \hat{r}_{11}^x[1;2] & \cdots\cdots & \hat{r}_{11}^x[5;0] & \hat{r}_{11}^x[5;1] & \hat{r}_{11}^x[5;2] \\ \hat{r}_{12}^x[1;0] & \hat{r}_{12}^x[1;1] & \hat{r}_{12}^x[1;2] & \cdots\cdots & \hat{r}_{12}^x[5;0] & \hat{r}_{12}^x[5;1] & \hat{r}_{12}^x[5;2] \\ \hat{r}_{13}^x[1;0] & \hat{r}_{13}^x[1;1] & \hat{r}_{13}^x[1;2] & \cdots\cdots & \hat{r}_{13}^x[5;0] & \hat{r}_{13}^x[5;1] & \hat{r}_{13}^x[5;2] \\ \hat{r}_{22}^x[1;0] & \hat{r}_{22}^x[1;1] & \hat{r}_{23}^x[1;2] & \cdots\cdots & \hat{r}_{22}^x[5;0] & \hat{r}_{22}^x[5;1] & \hat{r}_{22}^x[5;2] \\ \hat{r}_{23}^x[1;0] & \hat{r}_{23}^x[1;1] & \hat{r}_{22}^x[1;2] & \cdots\cdots & \hat{r}_{23}^x[5;0] & \hat{r}_{23}^x[5;1] & \hat{r}_{22}^x[5;2] \\ \hat{r}_{33}^x[1;0] & \hat{r}_{33}^x[1;1] & \hat{r}_{23}^x[1;2] & \cdots\cdots & \hat{r}_{33}^x[5;0] & \hat{r}_{33}^x[5;1] & \hat{r}_{23}^x[5;2] \end{bmatrix}. \quad (5.4.16)$$

This matrix will be employed in Alg. 5.2 on page 268 for both scenarios. Note that the reasoning regarding the system and form of the equations at the beginning of Section 5.4.1 is still applicable. Hence, we can directly employ results (5.4.3)-(5.4.5) for the coming examples.

5.4.2.1 Three sensors, three sources, and additive white noise

In this section, which except for the source signals is completely similar to Section 5.4.1.1, we analyze and consider the various steps of Alg. 5.2 for the scenario with $D = S = 3$ and the subspace matrix given by (5.4.16). The sensor signals $x_1[n]$, $x_2[n]$ and $x_3[n]$ are computed according to (5.1.1)/(5.1.2), where $\mathbf{A}$ is given by (5.4.7) and, as we have said above, the noise signals $\nu_1[n]$, $\nu_2[n]$ and $\nu_3[n]$ are mutually statistically independent white Gaussian noise sequences with variance 0.1. Fig. 5.15 shows the source signals $s_1[n]$, $s_2[n]$ and $s_3[n]$ at the left side, and the three noise-contaminated mixtures $x_1[n]$, $x_2[n]$ and $x_3[n]$ at the right side. Using the same reasoning as in Section 5.4.1.1, it follows that the system of 3-variate homogeneous polynomial equations of degree two that results from Alg. 5.2 on page 268 consists of three equations and has the matrix-vector form specified by (5.4.11). The estimated zero contour spherical ellipses are depicted in Fig. 5.16 in different colors/grey shades. Although the three estimated spherical ellipses do not intersect each other exactly at the columns of $\mathbf{A}$ represented by the black and grey arrows, they do so approximately. Hence, the array response vectors are determined uniquely and approximately by the intersections induced by the system of equations. In Sections 5.5 and 5.6 we will present methods for solving the system $\{\hat{f}_{3,1}^2(z_1, z_2, z_3) = 0, \hat{f}_{3,2}^2(z_1, z_2, z_3) = 0, \hat{f}_{3,3}^2(z_1, z_2, z_3) = 0\}$.

5.4.2.2 Three sensors, four sources, and additive white noise

In this section, which except for the source signals is completely similar to Section 5.4.1.2, we analyze and consider the various steps of Alg. 5.2 for the scenario with $D = 3$ and $S = 4$, and the subspace matrix given by (5.4.16). Again, the sensor signals $x_1[n]$, $x_2[n]$ and $x_3[n]$ are computed according to (5.1.1)/(5.1.2), where the mixing matrix $\mathbf{A}$ and noise signals $\nu_1[n]$, $\nu_2[n]$ and $\nu_3[n]$ are the same as in Section 5.4.1.2. Fig. 5.15 shows the source signals $s_1[n]$, $s_2[n]$, $s_3[n]$ and $s_4[n]$ at the left side, and the three noise-contaminated mixtures $x_1[n]$, $x_2[n]$ and $x_3[n]$ at the right side. Using the same reasoning as in Section 5.4.1.2, it follows that the system of 3-variate homogeneous polynomial equations of degree two that results from Alg. 5.2 consists of two equations and has the matrix-vector form specified by (5.4.15). The estimated zero contour spherical ellipses are depicted in Fig. 5.18 in different colors/grey shades. It can be seen that the array response vectors are determined uniquely and approximately by the intersections induced by the system of equations. In Sections 5.5 and 5.6 we will present methods for solving the system $\{\hat{f}_{3,1}^2(z_1, z_2, z_3) = 0, \hat{f}_{3,2}^2(z_1, z_2, z_3) = 0\}$.

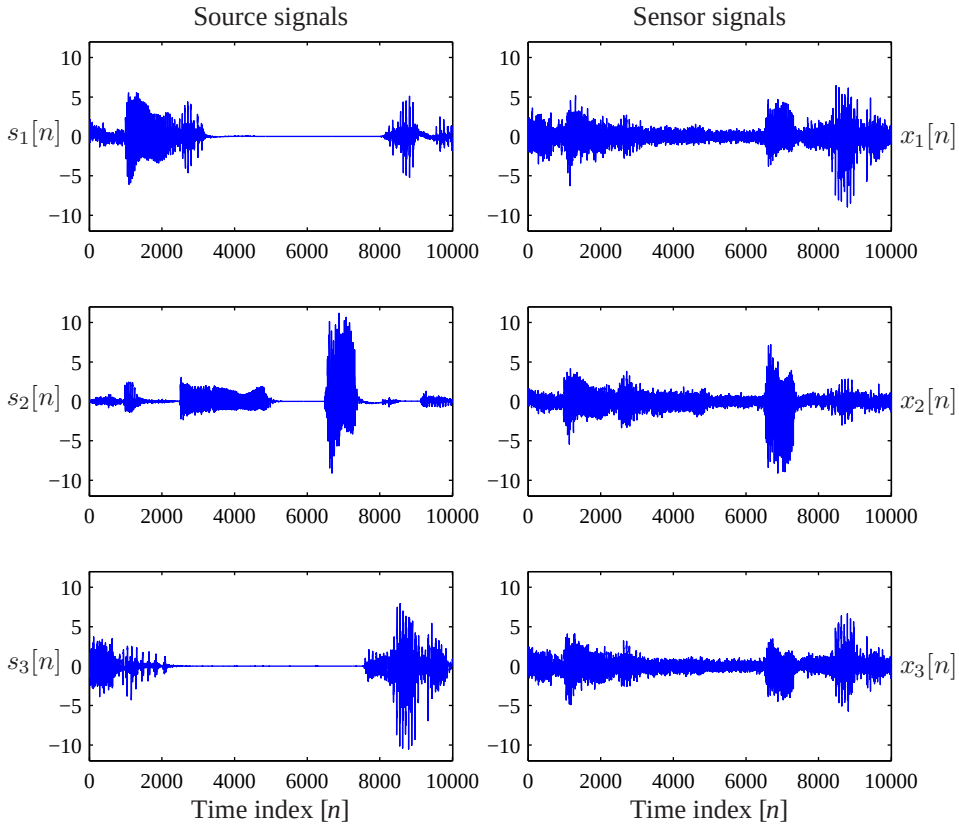

Figure 5.15: *Speech sources (left) and their noisy mixtures (right) for* 3×3 *example.*

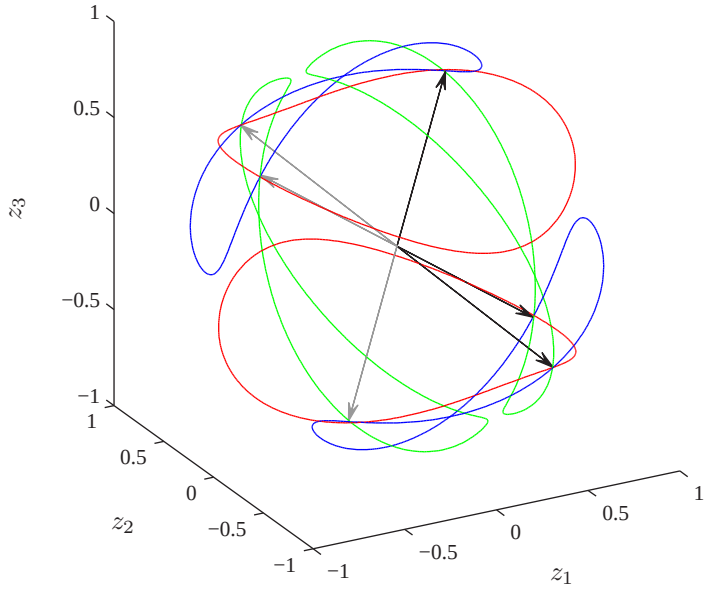

Figure 5.16: *Zero contour spherical ellipses of* $\hat{f}^2_{3,1}(z_1, z_2, z_3)$, $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ *and* $\hat{f}^2_{3,3}(z_1, z_2, z_3)$ *for* 3×3 *speech example.*

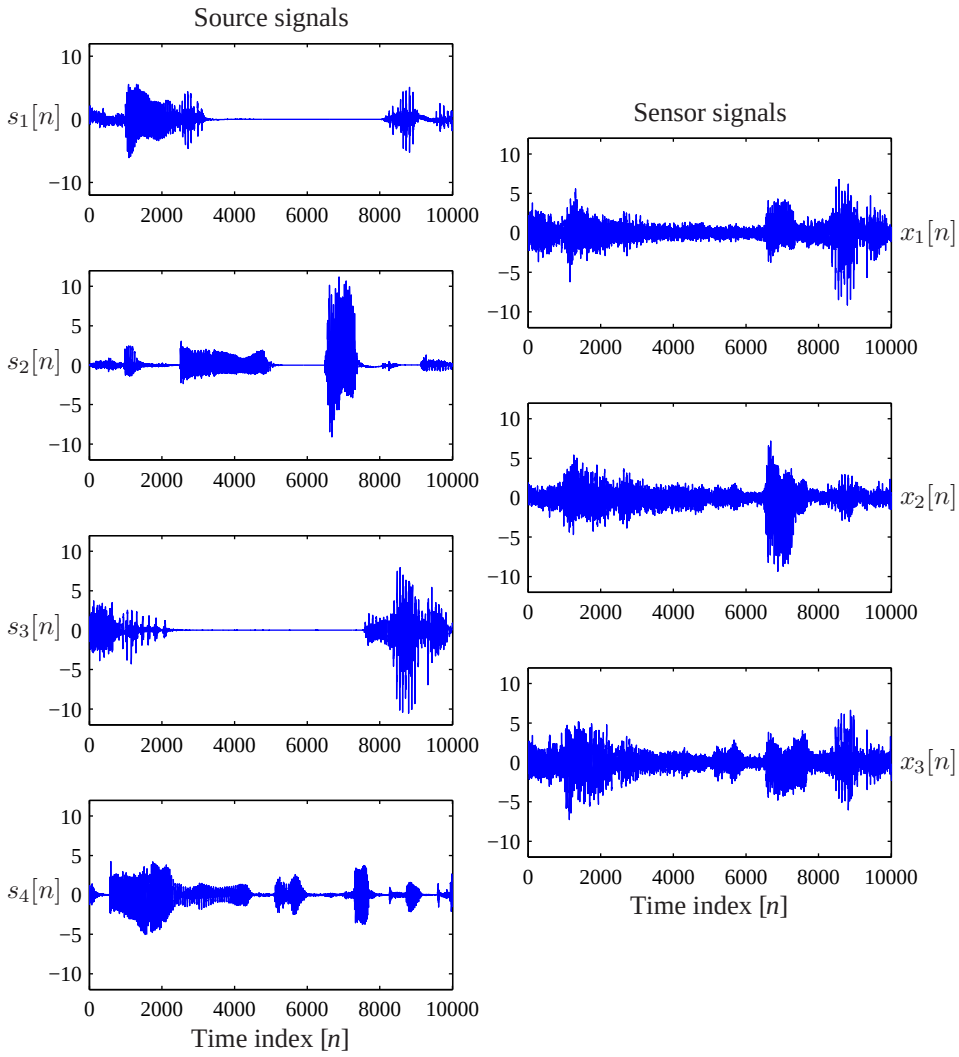

Figure 5.17: *Speech sources (left) and their noise-contaminated mixtures (right) for* 3×4 *example.*

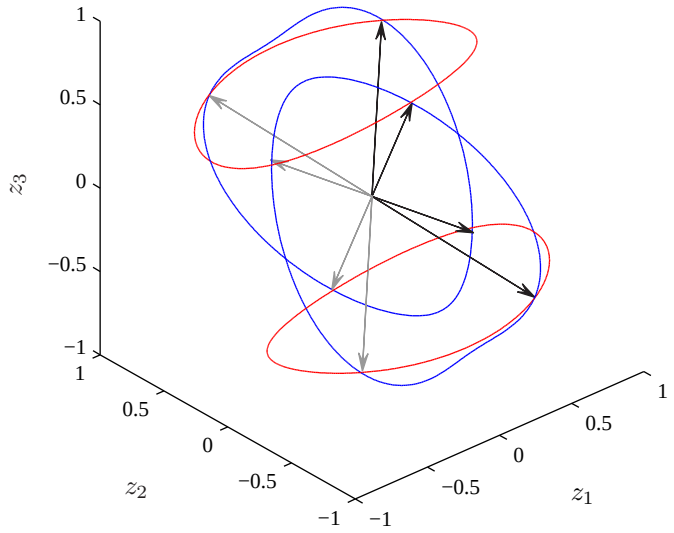

Figure 5.18: *Zero contour spherical ellipses of* $\hat{f}^2_{3,1}(z_1, z_2, z_3)$ *and* $\hat{f}^2_{3,2}(z_1, z_2, z_3)$ *for* 3×4 *speech example.*

5.5 Solving the system of equations by means of homotopy

In Section 4.5.6 we have introduced the homotopy method for solving a system of polynomial equations. Here we will use the same technique in the form of Alg. 4.4 on page 202 for solving the systems that we have derived in Sections 5.4.1 and 5.4.2. Our main task now is to formulate the appropriate start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ and target system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ for the system of equations under consideration. Similarly to (4.5.32) and (4.5.35), the target system comprises the system defined in (5.4.3)/(5.4.6) and a unit-norm constraint equation, i.e.:

$$\begin{cases} \left\{ f^2_{3,q}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}^2_{3,S}} ; \\ c(\mathbf{z}) = 0 \,, \end{cases} \tag{5.5.1}$$

where the unit-norm constraint function $c(\mathbf{z}) = c(z_1, z_2, z_3)$ is defined by:

$$c(\mathbf{z}) \triangleq (z_1)^2 + (z_2)^2 + (z_3)^2 - 1 \,. \tag{5.5.2}$$

We write this system in matrix-vector notation as follows:

$$\mathbf{p}(\mathbf{z}) = \begin{bmatrix} \mathbf{\Phi}_{f,\mathrm{h}}\, \mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} = \mathbf{0} \,, \tag{5.5.3}$$

where $\mathbf{\Phi}_{f,\mathrm{h}}$ is given by the matrix $\mathbf{\Phi}$ in (5.4.6). The start system is constructed in such a way that it has exactly the same structure as the target system (therefore, it also has the same number of solutions as the target system). Similarly to (5.5.3) it can be expressed as:

$$\mathbf{g}(\mathbf{z}) \triangleq \begin{bmatrix} \mathbf{\Phi}_{g,\mathrm{h}}\, \mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} = \mathbf{0} \,. \tag{5.5.4}$$

The coefficients of the homogeneous part of the start system can be obtained by computing the left null space of the uniquiefied second order Khatri-Rao product $\mathbf{B}^2_{\mathrm{u},3,\diamond}$ of a known matrix $\mathbf{B}$ (which might be generated at random), i.e. $\mathbf{\Phi}^* \nobreak\mathrel{\mathop:}= \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}^2_{\mathrm{u},3,\diamond}\big)$ in the formulation of (5.3.131). Now we can express the coefficient matrix $\mathbf{\Phi}_{h,\mathrm{h}}$ of the homogeneous part $\mathbf{h}_{\mathrm{h}}(\mathbf{z},\lambda)$ of the homotopy $\mathbf{h}(\mathbf{z},\lambda)$ that is defined by (4.5.25) and used in the third step of Alg. 4.4 as follows:

$$\mathbf{\Phi}_{h,\mathrm{h}}(\lambda) \triangleq \left[\gamma_g(\lambda-\lambda_e)\mathbf{\Phi}_{g,\mathrm{h}} + \gamma_p(\lambda-\lambda_0)\mathbf{\Phi}_{f,\mathrm{h}}\right] . \tag{5.5.5}$$

Because the non-homogeneous part of the start and target systems, i.e. the constraint function $c(\mathbf{z})$, is not deformed by (4.5.25) it suffices to work with the following homotopy:

$$\mathbf{h}(\mathbf{z},\lambda) \triangleq \begin{bmatrix} \mathbf{\Phi}_{h,\mathrm{h}}(\lambda)\,\mathbf{w}^2_{\mathrm{u},3}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} , \tag{5.5.6}$$

Notice that this homotopy is separated in its $\mathbf{z}$ and λ variables. The Jacobian of $\mathbf{h}(\mathbf{z},\lambda)$ can now be computed as follows:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda) \triangleq \begin{bmatrix} \mathbf{\Phi}_{h,\mathrm{h}}(\lambda)\,\nabla_{\mathbf{z}}\{\mathbf{w}^2_{\mathrm{u},3}(\mathbf{z})\} \\ \nabla_{\mathbf{z}}\{c(\mathbf{z})\} \end{bmatrix} . \tag{5.5.7}$$

The derivative of $\mathbf{h}(\mathbf{z},\lambda)$ w.r.t. λ is given by (4.5.31) with $\mathbf{g}(\mathbf{z})$ and $\mathbf{p}(\mathbf{z})$ defined in (5.5.4) and (5.5.3) respectively. See Section 4.5.6, Appendix E, and the references mentioned there, for more details about homotopy methods.

We now have all ingredients required by Alg. 4.4. For the 3×3 scenarios, the inverse $\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}$ in the sixth step of the algorithm has to be replaced by the pseudo-inverse $\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{\dagger}$, and the third function of the start system is set to the zero function or a linear combination of the first two functions. The parameter values are chosen as $MaxNnwtIt = 2$, $NwtTol = 10^{-4}$ and $NwtNoConvergenceTol = 5 \cdot 10^{-2}$. Recall that the constants $\gamma_g \in \mathbb{C}$ and $\gamma_p \in \mathbb{C}$ in third step of Alg. 4.4 are randomly chosen fixed constants that serve to avoid singularities and crossings along the different paths. For the path $\mathcal{C}$ in the complex plane to be traversed by the continuation parameter λ we choose the arc $\mathcal{C} \triangleq \{\cos(\theta) + \jmath\frac{3}{10}\sin(\theta) \mid 0 \leq \theta \leq \pi\}$ with starting point $\lambda_0 = 1$ and end point $\lambda_e = -1$. The ordered sequence $\mathcal{C}_o$ of values for λ lying on $\mathcal{C}$ in the fifth step of the algorithm is obtained by evaluating the parameterization of $\mathcal{C}$ for 101 linearly spaced values of θ between 0 and π. The algorithm will now be applied to the examples with estimated subspace matrices presented in Sections 5.4.1 and 5.4.2. We will visualize the solutions of the system of equations found by Alg. 4.4 by adding the estimated array response vectors multiplied by 1.5 to Figures 5.8, 5.12, 5.16, and 5.18, which depict the intersecting spherical ellipses for those examples. In addition, similarly to the examples presented in Sections 4.5.4.1-4.5.4.4, for the 3×3 example with speech signals we will compute the total transfer system and depicted the separated signals. To start with, Figures 5.19-5.22 depict the solutions of the estimated systems of equations found by the homotopy method for the 3×3 example with AR(1) signals, 3×4 example with AR(1) signals, 3×3 example with speech signals, and 3×4 example with speech signals, respectively. As before, the black arrows represent the ideal columns of the mixing matrix, whereas the grey arrows are the opposites of the black arrows. The magenta arrows indicate the estimated array response vectors multiplied by 1.5 for ease of visualization. The solid magenta dots indicate the intersections of the estimated array response vectors with the unit sphere. For each case, it can be seen that our homotopy algorithm has found approximate solutions for all array response vectors.

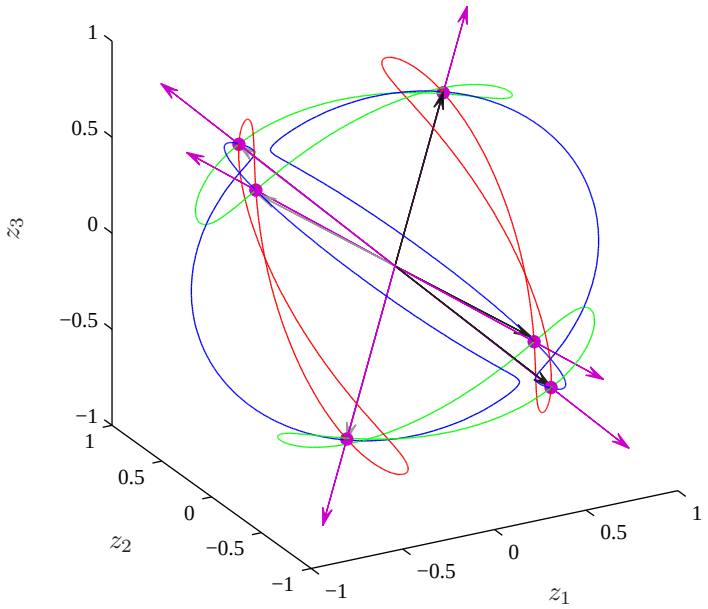

Figure 5.19: *Spherical ellipses and estimated solutions of system* $\left\{\hat{f}^2_{3,1}(z_1,z_2,z_3),\right.$ $\left.\hat{f}^2_{3,2}(z_1,z_2,z_3),\ \hat{f}^2_{3,3}(z_1,z_2,z_3)\right\}$ *for* 3×3 *example with AR(1) signals.*

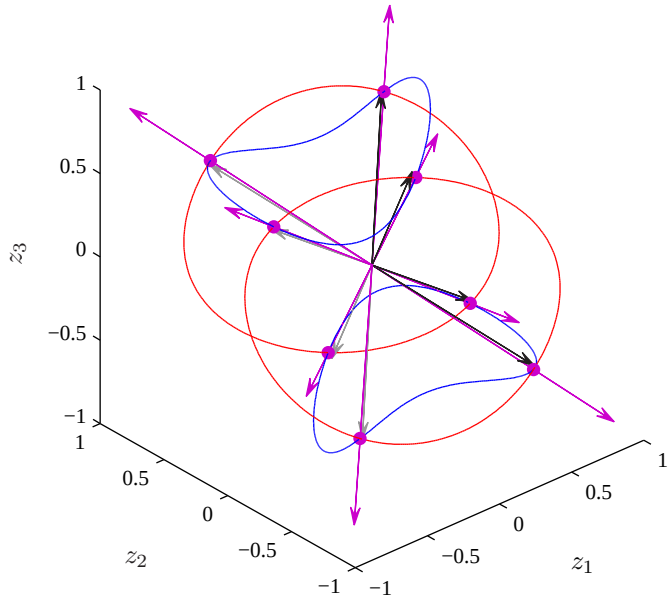

Figure 5.20: *Spherical ellipses and estimated solutions of system* $\left\{\hat{f}^2_{3,1}(z_1,z_2,z_3),\right.$ $\left.\hat{f}^2_{3,2}(z_1,z_2,z_3)\right\}$ *for* 3×4 *example with AR(1) signals.*

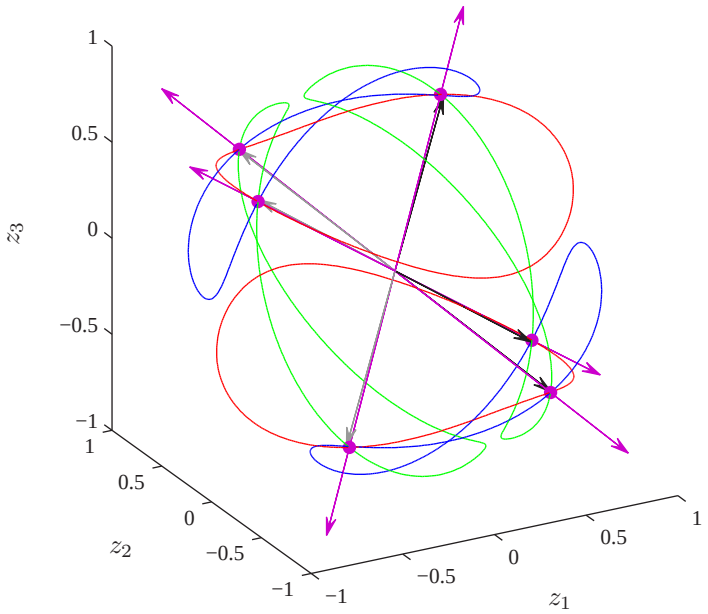

Figure 5.21: *Spherical ellipses and estimated solutions of system* $\left\{\hat{f}^2_{3,1}(z_1, z_2, z_3), \hat{f}^2_{3,2}(z_1, z_2, z_3), \hat{f}^2_{3,3}(z_1, z_2, z_3)\right\}$ *for* 3×3 *example with speech signals.*

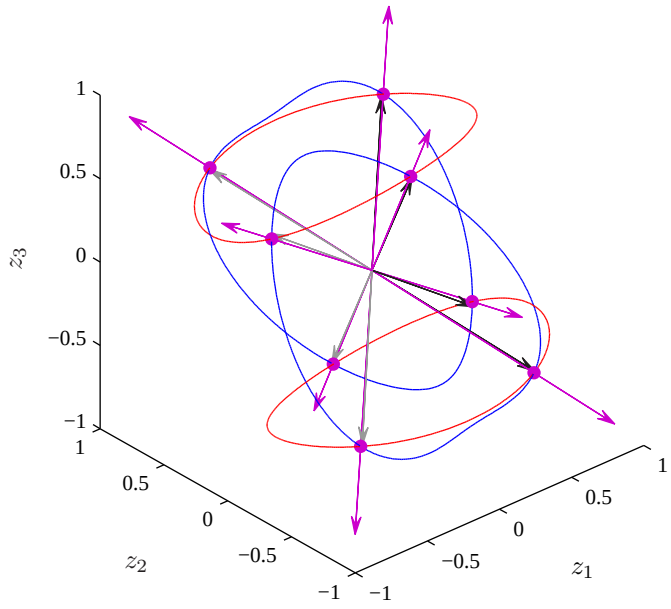

Figure 5.22: *Spherical ellipses and estimated solutions of system* $\left\{\hat{f}^2_{3,1}(z_1, z_2, z_3), \hat{f}^2_{3,2}(z_1, z_2, z_3)\right\}$ *for* 3×4 *example with speech signals.*

As a more detailed account of the results, for the 3×3 example with speech signals we compute the total transfer system as a measure of IBSS performance and depict the separated signals. The estimate of the mixing matrix $\mathbf{A}$ in (5.4.7) found by Alg. 4.4 is given by:

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.5637 & -0.4048 & -0.8123 \\ 0.5805 & 0.8174 & -0.1208 \\ 0.5876 & 0.4099 & 0.5706 \end{bmatrix} .$$

Because our purpose is to demonstrate MIBI and IBSS we use (4.5.23) to estimate the source signals from the noise-free sensor signals (recall that $\hat{\mathbf{A}}$ has been estimated from the noisy sensor data). Fig. 5.23 shows the source signals at the left side and the noise-free estimated source signals at the right side. It is evident that the signals are well-separated. The total transfer matrix from the source to the output signals equals:

$$\mathbf{T} \triangleq \hat{\mathbf{A}}^{-1}\mathbf{A} = \begin{bmatrix} 1.0001 & 0.0044 & -0.0098 \\ -0.0060 & -1.0021 & 0.0002 \\ -0.0138 & -0.0002 & -1.0019 \end{bmatrix} .$$

This matrix approximately equals the identity matrix with two 'flipped' signs, see also Fig. 5.23. Listening to the separated signals confirmed that the separation was successful.

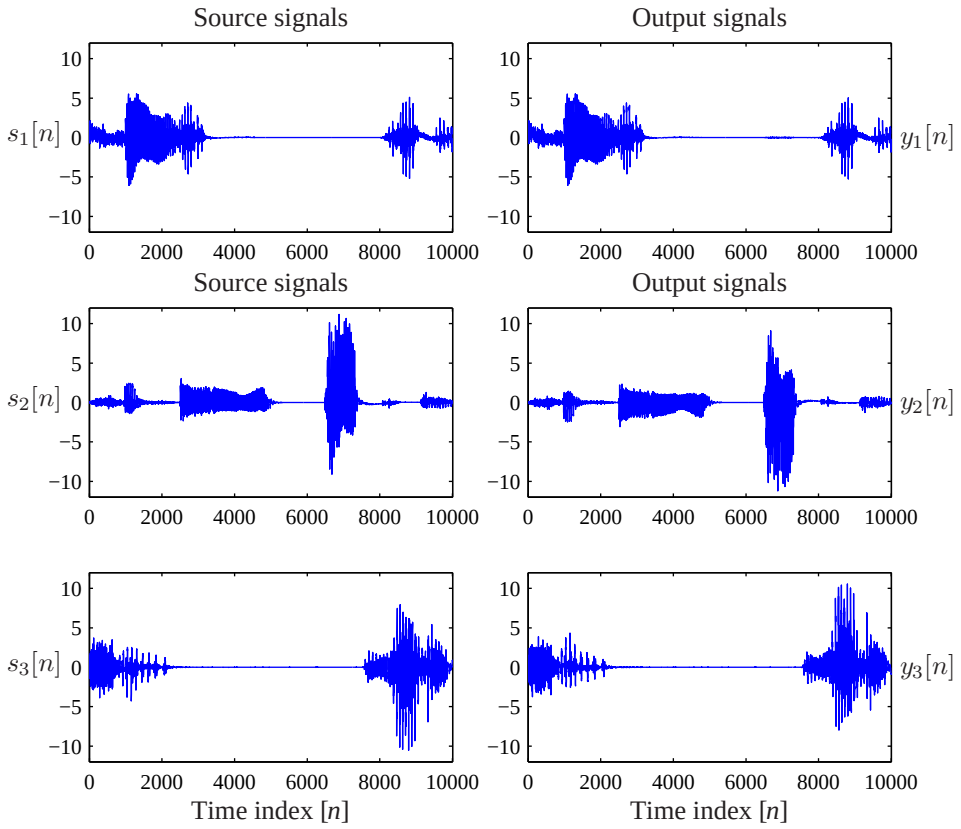

Figure 5.23: *Speech sources (left) and noise-free estimated source signals (right) for* 3×3 *example.*

5.6 Writing MIBI as a Multi-Matrix Generalized Eigenvalue Decomposition problem

In this section, we generalize the results from Section 4.6, i.e. the MIBI problem will be 'projected onto' a generalized eigenproblem. Because the scenario considered in this chapter is more general than the one in the previous chapter, now the eigenproblem becomes what we will call a *Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD)*. Again, this formulation is complementary to the problem description in terms of a system of homogeneous polyconjugal equations because the derivation of system (5.3.73) was primarily based on (5.3.135), whereas the derivation we are about to present is primarily based on the complementary relation (5.3.134). More precisely, under the assumption that the source autocorrelation matrix has full rank system (5.3.73) was derived directly from the fact that the left null space of the (uniquiefied) subspace matrix equals the left null space of the (uniquiefied) second order Khatri-Rao product of the mixing matrix, which was proven in Section 5.3.6.6. Likewise, in the sequel of this section we will derive the complementary MMGEVD problem from the fact that the column range of the (uniquiefied) subspace matrix equals the column range of the (uniquiefied) second order Khatri-Rao product of the mixing matrix, which was also proven in Section 5.3.6.6. *For convenience, without any loss of generality the results in this section are derived from the 'full subspace matrix'* $\mathbf{C}_D^{x,c_1c_2}$ *instead of the uniquiefied matrix* $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$.

To start with, from the 'full versions' of Theorems 5.3.3, 5.3.5, and 5.3.7 it follows directly that:

$$\check{\mathbf{a}}_D^{j,c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{a}^j) \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_D^{x,c_1c_2}\right) = \mathcal{L}_{\mathrm{c}}\big(\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) , \tag{5.6.1}$$

where $\check{\mathbf{a}}_D^{j,c_1c_2}$ and $\mathbf{w}_D^{c_1c_2}(\cdot)$ are defined in (5.3.29) and (5.3.30) respectively. Note that now the matrix $\mathbf{U}^s$ contains the left singular vectors of the 'full subspace matrix' $\mathbf{C}_D^{x,c_1c_2}$ and that the 'full versions' of (5.3.145) and (5.3.158) have been used:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_D^{x,c_1c_2}\big) = \mathcal{L}_c\big(\mathbf{r}_D^{x,c_1c_2}[n_1,n_2]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{c_1c_2}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^s\big) = \mathcal{L}\left(\mathcal{U}^s\right) .$$

Eq. (5.6.1) implies that $\check{\mathbf{a}}_D^{j,c_1c_2}$ can be written as a linear combination of the columns of $\mathbf{U}^s$:

$$\check{\mathbf{a}}_D^{j,c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{a}^j) = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p^j\, \mathbf{u}^p = \mathbf{U}^s\, \boldsymbol{\mu}^j \qquad \forall\, 1 \leq j \leq S\,, \tag{5.6.2}$$

where $\boldsymbol{\mu}^j \triangleq \begin{bmatrix} \mu_1^j & \cdots & \mu_{d_D^{x,c_1c_2}}^j \end{bmatrix}^T$ with $\mu_p^j \in \mathbb{R}$ for $1 \leq p \leq d_D^{x,c_1c_2}$ and $1 \leq j \leq S$. Thus, the MIBI problem can be projected onto the problem of finding the solutions for the unknown vector $\mathbf{z}$ that satisfy the following equation:

$$\boxed{\check{\mathbf{z}}_D^{c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{z}) \triangleq (\mathbf{z})^{c_1} \otimes (\mathbf{z})^{c_2} = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p\, \mathbf{u}^p = \mathbf{U}^S\, \boldsymbol{\mu}\,,} \tag{5.6.3}$$

where to each solution $\mathbf{z}$ there corresponds a different vector $\boldsymbol{\mu}$ of coefficients. The latter two equations say that the problem of finding the array response vectors is equivalent to finding vectors $\mathbf{z}$ such that the structured vectors $\mathbf{w}_D^{c_1c_2}(\mathbf{z}) = \check{\mathbf{z}}_D^{c_1c_2}$ belong to the column range $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{c_1c_2}\big)$ of $\mathbf{A}_{D,\diamond}^{c_1c_2}$, which can be computed from the known matrix $\mathbf{C}_D^{x,c_1c_2}$ because

$\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{c_1c_2}\big)$ equals $\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_D^{x,c_1c_2}\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right)$. In order to find the solutions for $\mathbf{z}$, or equivalently the solutions for $\boldsymbol{\mu}$, we exploit the specific structure of $\check{\mathbf{z}}_D^{c_1c_2} = \mathbf{w}_D^{c_1c_2}(\mathbf{z})$ given by (5.3.30). Firstly, from (5.3.30) it follows directly that (5.6.3) can be written as follows:

$$\begin{bmatrix} (z_1)^{c_1}\,(\mathbf{z})^{c_2} \\ \vdots \\ (z_D)^{c_1}\,(\mathbf{z})^{c_2} \end{bmatrix} = \begin{bmatrix} (z_1)^{c_1}(z_1)^{c_2} \\ \vdots \\ (z_1)^{c_1}(z_D)^{c_2} \\ \vdots \\ (z_D)^{c_1}(z_1)^{c_2} \\ \vdots \\ (z_D)^{c_1}(z_D)^{c_2} \end{bmatrix} = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p \begin{bmatrix} u_1^p \\ \vdots \\ u_D^p \\ \vdots \\ u_{D(D-1)+1}^p \\ \vdots \\ u_{(D)^2}^p \end{bmatrix} = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p \begin{bmatrix} u_{11}^p \\ \vdots \\ u_{1D}^p \\ \vdots \\ u_{D1}^p \\ \vdots \\ u_{DD}^p \end{bmatrix}. \tag{5.6.4}$$

Note that each product of the form $(z_{i_1})^{c_1}(z_{i_2})^{c_2}$ can be expressed as:

$$\boxed{(z_{i_1})^{c_1}(z_{i_2})^{c_2} = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p\, u_{i_1i_2}^p \qquad \forall\, 1 \leq i_1, i_2 \leq D\,.} \tag{5.6.5}$$

From (5.6.5) the following systems of equations follow directly by varying i_2 from 1 till D:

$$(z_i)^{c_1}\,(\mathbf{z})^{c_2} = \begin{bmatrix} (z_i)^{c_1}(z_1)^{c_2} \\ \vdots \\ (z_i)^{c_1}(z_D)^{c_2} \end{bmatrix} = \sum_{p=1}^{d_D^{x,c_1c_2}} \mu_p \begin{bmatrix} u_{i1}^p \\ \vdots \\ u_{iD}^p \end{bmatrix} = \mathbf{G}_i\,\boldsymbol{\mu} \qquad \forall\, 1 \leq i \leq D\,, \tag{5.6.6}$$

where we have made the following definitions:

$$\mathbf{G}_i \triangleq \begin{bmatrix} u_{i1}^1 & \cdots & u_{i1}^{d_D^{x,c_1c_2}} \\ \vdots & \ddots & \vdots \\ u_{iD}^1 & \cdots & u_{iD}^{d_D^{x,c_1c_2}} \end{bmatrix} = \begin{bmatrix} \mathbf{u}_i^1 & \cdots & \mathbf{u}_i^{d_D^{x,c_1c_2}} \end{bmatrix} \in \mathbb{C}_D^{d_D^{x,c_1c_2}} \qquad \forall\, 1 \leq i \leq D\,, \tag{5.6.7}$$

$$\mathbf{u}_i^p \triangleq \begin{bmatrix} u_{i1}^p \\ \vdots \\ u_{iD}^p \end{bmatrix} \qquad \forall\, 1 \leq p \leq d_D^{x,c_1c_2}, \tag{5.6.8}$$

and:

$$\boldsymbol{\mu} \triangleq \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{d_D^{x,c_1c_2}} \end{bmatrix}. \tag{5.6.9}$$

Note that $\mathbf{G}_i$ is the i-th submatrix of size $D \times d_D^{x,c_1c_2}$ of $\mathbf{U}^s$:

$$\mathbf{U}^s = \begin{bmatrix} \mathbf{G}_1 \\ \vdots \\ \mathbf{G}_D \end{bmatrix}. \tag{5.6.10}$$

Eq. (5.6.6) implies that we have projected the MIBI problem onto the problem of solving the following D systems simultaneously:

$$\boxed{(z_i)^{c_1}\,(\mathbf{z})^{c_2} = \mathbf{G}_i\,\boldsymbol{\mu} \qquad \forall\, 1 \leq i \leq D\,.} \tag{5.6.11}$$

In a practice, the only known quantities are the matrices $\mathbf{G}_1$, ..., $\mathbf{G}_D$ that are computed from the SVD of the subspace matrix. Our goal is to find S different vectors $\boldsymbol{\mu}$, say $\boldsymbol{\mu}^1$, ..., $\boldsymbol{\mu}^S$. Each solution vector $\boldsymbol{\mu}^j$ corresponds to an array response vector, say $\mathbf{z}^j$. We will show later in this section how the array response vectors $\mathbf{z}^1$, ..., $\mathbf{z}^S$ corresponding to $\boldsymbol{\mu}^1$, ..., $\boldsymbol{\mu}^S$ respectively can also be determined from (5.6.11) once the latter vectors are known.

Since all matrix-vector products $\mathbf{G}_1\,\boldsymbol{\mu}$, ..., $\mathbf{G}_D\,\boldsymbol{\mu}$ should yield a vector in the same direction, viz. that of $(\mathbf{z})^{c_2}$, system (5.6.11) describes what we will call a *Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD)*. This can be seen more clearly as follows. Supposing for convenience that $z_i \neq 0 \quad \forall\, 1 \leq i \leq D$, (5.6.11) can be written as follows:

$$(\mathbf{z})^{c_2} = \frac{1}{(z_1)^{c_1}}\mathbf{G}_1\,\boldsymbol{\mu} = \cdots = \frac{1}{(z_D)^{c_1}}\mathbf{G}_D\,\boldsymbol{\mu}\,. \tag{5.6.12}$$

Hence, the vector $\boldsymbol{\mu}$ is a generalized eigenvector of the matrices $\mathbf{G}_1$, ..., $\mathbf{G}_D$ and for each pair of indices (i_1, i_2) with $1 \leq i_1, i_2 \leq D$ a standard Generalized Eigenvalue Decomposition (GEVD) problem exists:

$$\frac{1}{(z_{i_1})^{c_1}}\,\mathbf{G}_{i_1}\,\boldsymbol{\mu} = \frac{1}{(z_{i_2})^{c_1}}\,\mathbf{G}_{i_2}\,\boldsymbol{\mu} \qquad \forall\, 1 \leq i_1, i_2 \leq D\,. \tag{5.6.13}$$

That is, for each pair (i_1, i_2) the vector $\boldsymbol{\mu}$ is a generalized eigenvector of the matrix pencil $\mathbf{G}_{i_2} - \lambda\mathbf{G}_{i_1}$ with generalized eigenvalue $\lambda = \left(\frac{z_{i_2}}{z_{i_1}}\right)^{c_1}$ because it satisfies $\mathbf{G}_{i_2}\boldsymbol{\mu} = \left(\frac{z_{i_2}}{z_{i_1}}\right)^{c_1}\mathbf{G}_{i_1}\boldsymbol{\mu}$. As in Section 4.6, we will denote the GEVD of $\mathbf{G}_{i_1}$ and $\mathbf{G}_{i_2}$ by $\mathrm{gevd}(\mathbf{G}_{i_1}, \mathbf{G}_{i_2})$. A much more natural description of a standard or multi-matrix GEVD is obtained by multiplying the involved system by the product of the terms occurring in the denominators pre-multiplying the $\mathbf{G}_i\,\boldsymbol{\mu}$'s in (5.6.12) and (5.6.13).

Firstly, applying this to (5.6.13) yields:

$$\boxed{(z_{i_2})^{c_1}\,\mathbf{G}_{i_1}\,\boldsymbol{\mu} = (z_{i_1})^{c_1}\,\mathbf{G}_{i_2}\,\boldsymbol{\mu} \qquad \forall\, 1 \leq i_1, i_2 \leq D\,.} \tag{5.6.14}$$

In this formulation the problem comes down to finding all generalized eigenvectors and eigenvalue pairs of the matrix pencils $\lambda_1\mathbf{G}_{i_1} + \lambda_2\mathbf{G}_{i_2}$ for $1 \leq i_1, i_2 \leq D$, and then selecting the common eigenvectors. Note that $(\lambda_1, \lambda_2) = \big((z_{i_2})^{c_1}, (z_{i_1})^{c_1}\big)$ forms a solution for the generalized eigenvalue pair. The intersections of the generalized eigenspaces of the GEVD's of all possible matrix pencils defined by (5.6.14) give S possible solutions for the vector $\boldsymbol{\mu}$, each of which corresponds to a column of the mixing matrix.

Secondly, multiplying (5.6.12) by $(z_1 \cdots z_D)^{c_1}$ yields:

$$(z_1 \cdots z_D)^{c_1}(\mathbf{z})^{c_2} = (z_2 \cdots z_D)^{c_1}\,\mathbf{G}_1\,\boldsymbol{\mu} = \cdots\cdots = (z_1 \cdots z_{D-1})^{c_1}\,\mathbf{G}_D\,\boldsymbol{\mu}\,. \tag{5.6.15}$$

These systems show that we do not need to assume that $z_i \neq 0 \quad \forall\, 1 \leq i \leq D$. In this formulation the problem comes down to finding all generalized eigenvectors and *eigenvalue tuples* of the *generalized matrix pencil* $\mathbf{G}\big(\boldsymbol{\lambda}\big)$ that is defined as follows:

$$\boxed{\mathbf{G}\big(\boldsymbol{\lambda}\big) \triangleq \lambda_1\,\mathbf{G}_1 + \cdots + \lambda_D\,\mathbf{G}_D\,,} \tag{5.6.16}$$

where the generalized eigenvalue tuple $\boldsymbol{\lambda}$ is defined by:

$$\boldsymbol{\lambda} \triangleq \big(\lambda_1, \ldots, \lambda_D\big)\,. \tag{5.6.17}$$

Note that an ideal solution for $\boldsymbol{\lambda}$ is given by:

$$\boldsymbol{\lambda} = \left((z_2 \cdots z_D)^{c_1}, \ldots, (z_1 \cdots z_{D-1})^{c_1} \right),$$

and that for this solution:

$$\mathbf{G}\big(\boldsymbol{\lambda}\big)\, \boldsymbol{\mu} = D\, (z_1 \cdots z_D)^{c_1}\, (\mathbf{z})^{c_2}\,.$$

Evidently, the results that we have just obtained can also be derived directly from (5.6.11) by multiplying it by the product of all c_1-conjugated variables $z_1, \ldots, z_D$ except for the i-th:

$$(z_1 \cdots z_D)^{c_1}\, (\mathbf{z})^{c_2} = (z_1 \cdots z_{i-1})^{c_1} (z_{i+1} \cdots z_D)^{c_1} \mathbf{G}_i\, \boldsymbol{\mu} \qquad \forall\, 1 \leq i \leq D\,, \tag{5.6.18}$$

which equals (5.6.15). Summarizing, we have shown that finding all generalized eigenvectors and eigenvalue tuples of the generalized matrix pencil defined in (5.6.16) means solving a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem. This is a mathematical problem that to our knowledge has not yet been formulated in this general form. For future use and convenience we formulate it explicitly as follows. Assume that the generalized eigenvectors and eigenvalues resulting from the MMGEVD are stored in the matrix $\mathbf{M}_G$ and set Λ_G of generalized eigenvalue tuples respectively, then we write:

$$[\mathbf{M}_G, \Lambda_G] = \mathrm{gevd}\left(\mathbf{G}_1, \ldots, \mathbf{G}_D\right). \tag{5.6.19}$$

This system has d_D^{x,c_1c_2} generalized eigenvectors $\boldsymbol{\mu}$ and eigenvalue tuples $\boldsymbol{\lambda}$. Hence, the generalized eigenvector matrix $\mathbf{M}_G$ and generalized eigenvalue tuple set Λ_G are given by $\mathbf{M}_G = \left[\boldsymbol{\mu}^1 \;\; \cdots \;\; \boldsymbol{\mu}^{d_D^{x,c_1c_2}}\right]$ and $\Lambda_G = \left\{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{d_D^{x,c_1c_2}}\right\}$ respectively. After having computed this decomposition, the columns of $\mathbf{A}$ can be recovered straightforwardly from the generalized eigenvectors; see (5.6.23).

Observe that $\boldsymbol{\mu}$ also is a generalized eigenvector of all linear combinations of the matrices $\mathbf{G}_1, \ldots, \mathbf{G}_D$. This can easily be proven as follows. Assume that $\boldsymbol{\mu}$ is a solution of (5.6.11) and for some index $p \in \mathbb{N}^+$ define the scalar coefficients $\xi_p^1, \ldots, \xi_p^D$ as the coefficients of the linear combination $\boldsymbol{\Gamma}_p$ of the matrices $\mathbf{G}_1, \ldots, \mathbf{G}_D$, i.e.:

$$\boldsymbol{\Gamma}_p \triangleq \sum_{i=1}^{D} \xi_p^i\, \mathbf{G}_i\,. \tag{5.6.20}$$

Using (5.6.11) it then follows that:

$$\boldsymbol{\Gamma}_p\, \boldsymbol{\mu} = \sum_{i=1}^{D} \xi_p^i\, \mathbf{G}_i\, \boldsymbol{\mu} = \sum_{i=1}^{D} \xi_p^i\, (z_i)^{c_1}\, (\mathbf{z})^{c_2} = \left(\sum_{i=1}^{D} \xi_p^i\, (z_i)^{c_1} \right) (\mathbf{z})^{c_2}\,, \tag{5.6.21}$$

which shows that $\boldsymbol{\Gamma}_p\, \boldsymbol{\mu}$ is a vector in the direction of $(\mathbf{z})^{c_2}$. Hence, $\boldsymbol{\mu}$ is also a generalized eigenvector of all matrices $\boldsymbol{\Gamma}_p$ consisting of linear combinations of the matrices $\mathbf{G}_1, \ldots, \mathbf{G}_D$.

Depending on the number of sources, number of sensors, and the conjugation pair, the matrices $\mathbf{G}_1, \ldots, \mathbf{G}_D$ of size $D \times d_D^{x,c_1c_2}$ are square or rectangular. If $D \geq d_D^{x,c_1c_2}$, i.e. the matrices are square or have more rows than columns, the MMGEVD problem can easily be solved by traditional techniques in the same manner as presented in Section 4.6 by performing a standard Generalized Eigenvalue Decomposition (GEVD) on two different linear combinations $\boldsymbol{\Gamma}^1$ and $\boldsymbol{\Gamma}^2$ that are randomly generated according to (5.6.20). Assume that the generalized eigenvectors and eigenvalues resulting from the GEVD of $\boldsymbol{\Gamma}_1$ and $\boldsymbol{\Gamma}_2$ are stored in the matrices $\mathbf{M}_G$ and $\boldsymbol{\Lambda}_G$ respectively:

$$[\mathbf{M}_G, \boldsymbol{\Lambda}_G] = \mathrm{gevd}\left(\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2\right). \tag{5.6.22}$$

This system has d_D^{x,c_1c_2} generalized eigenvectors and eigenvalues. Hence, $\mathbf{M}_G = \left[\boldsymbol{\mu}^1 \cdots \boldsymbol{\mu}^{d_D^{x,c_1c_2}}\right]$ and $\boldsymbol{\Lambda}_G = \mathrm{diag}(\lambda_1, \ldots, \lambda_{d_D^{x,c_1c_2}})$. After having computed this decomposition, the columns of $\mathbf{A}$ can be recovered from the generalized eigenvectors. Firstly, (5.6.21) shows that for a generalized solution eigenvector $\boldsymbol{\mu}^j$ the vectors $\boldsymbol{\Gamma}_1\,\boldsymbol{\mu}^j$ and $\boldsymbol{\Gamma}_2\,\boldsymbol{\mu}^j$ lie in the direction of $(\mathbf{z}^j)^{c_2}$, where $\mathbf{z}^j$ is a desired solution of (5.6.3). Hence, given $\boldsymbol{\mu}^j$ the corresponding solution for $\mathbf{z}^j$ can be computed as an arbitrary linear combination of $\boldsymbol{\Gamma}_1\,\boldsymbol{\mu}^j$ and $\boldsymbol{\Gamma}_2\,\boldsymbol{\mu}^j$ that is conjugated according to c_2, i.e.:

$$\mathbf{z}^j = \frac{\left((\alpha_1\,\boldsymbol{\Gamma}_1 + \alpha_2\,\boldsymbol{\Gamma}_1)\,\boldsymbol{\mu}^j\right)^{c_2}}{\|(\alpha_1\,\boldsymbol{\Gamma}_1 + \alpha_2\,\boldsymbol{\Gamma}_1)\,\boldsymbol{\mu}^j\|} \qquad 1 \leq j \leq S\,, \tag{5.6.23}$$

where α_1 and α_2 are arbitrary non-zero real- or complex-valued scalars. See also the next chapter, where the results from this section are generalized. If $D < d_D^{x,c_1c_2}$, i.e. if the matrices $\mathbf{G}_1$, …, $\mathbf{G}_D$ have less rows than columns, the MGEVD problem cannot be solved immediately by traditional techniques. In this case, the solutions are not determined uniquely by the generalized eigenvectors of two (arbitrary linear combinations of) such matrices, but we need all of them.

As an example, we apply the technique outlined above to the 3×3 example with speech signals discussed in Section 5.4.2.1. This yields the following estimate of the mixing matrix $\mathbf{A}$ in (5.4.7):

$$\hat{\mathbf{A}} = \begin{bmatrix} -0.4099 & 0.5910 & -0.8036 \\ 0.8134 & 0.5877 & -0.0829 \\ 0.4128 & 0.5525 & 0.5893 \end{bmatrix},$$

which approximates the true mixing matrix quite well. Compare this result also with the homotopy result in obtained in Section 5.5.

5.7 Conclusions and discussion

In this chapter we have generalized the results from the previous chapter in three senses. Firstly, instead of the MIBI scenario with two sources and two sensors depicted in Fig. 4.1, here we have considered the scenario with S sources and D sensors depicted in Fig. 5.1. Secondly, we have considered the more general complex-valued case with arbitrary conjugation pair that should be chosen in accordance with the characteristics of the involved signals. Finally, instead of employing only one type of second order temporal statistical diversity, viz. non-whiteness, we have also employed second order non-stationarity or quasi-stationarity under the assumption that each source signal has a 'non-stationarity pattern' that is sufficiently different from the others. We have seen that these different types of statistical variability can be exploited in the same unified manner. It is possible to either exploit the non-whiteness or non-stationarity alone, or both simultaneously. We have justified the equity of the assumptions underlying our method. Again, the MIBI problem was projected onto two dual mathematical problems by applying subspace techniques to the subspace matrix. In the first problem, a well-structured system of homogeneous polynomial-like equations of degree two has to be solved. For reasons that we have made clear the polynomial-like functions in the system are called polyconjugals. In the second problem, a multi-matrix generalized eigenvalue decomposition has to be performed. Finally, we have shown that also with second order statistics it is possible to handle underdetermined scenarios with more sources than sensors.

CHAPTER 6

MIBI based on arbitrary order temporal structure: $D \times S$ mixing case

This chapter deals with the most general scenario for Multiple-Input Multiple-Output Instantaneous Blind Identification (MIBI). It provides a unifying framework for exploiting arbitrary order temporal structure in the signals for performing MIBI with real- or complex-valued signals and system. In particular, we consider a complex-valued scenario with S sources and D sensors as depicted in Fig. 5.1 on page 216, and we employ l-th order statistics with an arbitrary fixed conjugation tuple $\mathbf{c}_l \triangleq (c_1, \ldots, c_l)$. Instead of representing the statistics by correlation functions as we did in the previous two chapters we now use cumulant functions, which also in the general case possess convenient mathematical properties such as multilinearity. The MIBI problem is formulated in such a way that any kind of temporal structure in the data, such as arbitrary order non-stationarity and non-whiteness, is exploited in a unified manner. Based on physically plausible assumptions on the temporal structure of the source and noise signals, and applying subspace techniques to a subspace matrix containing cumulant values arranged in a specific manner, it is shown that the MIBI problem can be projected onto two dual mathematical problems. In the first one, MIBI is projected onto the problem of solving a system of multivariate homogeneous polynomial or polyconjugal equations. The number of variables and the degree (of homogeneity) of the functions in the system equal the number of sensors D and the order l of the considered statistics, respectively. The vector-valued solutions of the system are the estimates of the columns of the mixing matrix and are obtained by a homotopy method. In the second problem, MIBI is projected onto the problem of solving a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem that is dual to the first problem. Because of the connection between the system of homogeneous equations on the one hand, and the MMGEVD problem on the other hand, solving one of these mathematical problems also solves the other.

As in the previous chapters, we highlight the geometric and algebraic structure of both problem formulations. Furthermore, we show that the theory allows us to make trade-offs between various related quantities such as the maximum number of identifiable mixing matrix columns/sources, minimum required number of sensors, exploited type(s) of temporal structure, exploited order(s) of temporal structure, employed conjugation pattern(s), number of samples required for reliable estimation of the involved statistics, the arrangement of the statistics in the subspace matrix, etcetera. In this chapter, the full power of the notation developed in the thesis will become clear. It is shown that the notation enables us to give a uniform and consistent description of the theory for the most general MIBI problem, and that all generalizations presented in this chapter follow readily from the corresponding ones in the previous chapters by straightforwardly generalizing the involved notation in an intuitive manner. From this point of view the results and examples presented in Chapters 4 and 5 can be considered as specializations of the 'template theory' presented in this chapter.

The outline of this chapter is largely similar to that of Chapter 5. Since the structure of the $D \times S$ MIBI model is the same as that described in Section 5.1, there is no such section here. Therefore, we start in Section 6.1 by explaining the model assumptions. Next, in Section 6.2 a system of D-variate homogeneous polyconjugal equations of degree l and with conjugation tuple $\mathbf{c}_l$ is derived that is satisfied by the columns of the mixing matrix, thereby highlighting the algebraic structure of our problem formulation. After that, in Section 6.3 the geometric structure associated with this system of equations is investigated and examples are given. Subsequently, in Section 6.4 a homotopy method for solving the homogeneous system is presented. Then, in Section 6.5 the MIBI problem is written as a multi-matrix generalized eigenvalue problem. In Section 6.6, the maximum number of mixing matrix columns that can be identified for a given number of sensors and conjugation pattern is determined. Finally, conclusions and discussion are presented in Section 6.7. In order to keep a clear overview of the main theme of this chapter, we summarize the key ingredients of the resulting $D \times S$ MIBI algorithm based on l-th order statistics with conjugation tuple $\mathbf{c}_l$ in the following high-level algorithm:

Algorithm 6.1 Pre-overview of the $D \times S$ MIBI method based on l-th order statistics and conjugation tuple $\mathbf{c}_l$.

1: *Compute/estimate all l-th order sensor cumulant functions with conjugation tuple $\mathbf{c}_l$ on a Noise-Free ROS consisting of several time, time-lag, or lag tuples;*

2: *Arrange these values in a uniquefied subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$;*

3: *Compute the Singular Value Decomposition of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and split the result into signal and null/noise subspace parts;*

4: *Choose one of the following two MIBI problem projections:*

- *Construct a system of D-variate homogeneous polyconjugal equations with conjugation tuple $\mathbf{c}_l$, whose solutions are the columns of the mixing matrix;*
- *Construct a multi-matrix generalized eigenvalue problem, whose generalized eigenvectors are directly related to the columns of the mixing matrix;*

5: *Solve the problem defined in the previous step.*

6.1 MIBI model assumptions for l-th order statistics

The MIBI model assumptions of the previous two chapters concern correlation functions defined on one- or two-dimensional Regions Of Support and allow to exploit the Second Order Temporal Structure (SOTS) in the data. In this section, assumptions like these are formulated mathematically in their most general form for a fixed arbitrary statistical order l and conjugation tuple $\mathbf{c}_l$. For convenience we assume that the involved Regions Of Support are defined in the space of length-l tuples. Depending on the (higher-order) stationarity properties, it may be possible to define them in a space of dimension smaller than l, but all such scenarios can be derived from the general one that will be presented in the sequel. As we have alluded to already in the previous chapter and will witness in this chapter, *the dimensionality,*

shape, and size of a ROS are not essential for the rationale behind our approach. The true essence lies in the relations between the various functions that are defined on the ROS under consideration. As we have explained earlier, the assumptions about these relations mainly serve to ensure that sufficient temporal structure is present in the source signals and that the noise signals have a simpler temporal structure than the source signals. These properties are conveyed in general form by *MAS1-MAS4* on page 23. The temporal structure of a certain order can be represented mathematically by cumulant functions of the same order. We start by providing definitions of the employed cumulant functions and then discuss and summarize the various assumptions. As we have explained in Sections 1.2.7, A.6, and B.3, a cumulant function can be defined in different ways corresponding to the pattern in which its arguments are conjugated. Such a *conjugation pattern* is represented by a *conjugation tuple* and is written in the upper-right corner of the considered symbol or function. As we also have remarked in Sections 1.2.7 and A.6, and justified in Section B.3, the most suitable conjugation tuple for a particular application depends on the type of the involved signals. In the derivations presented in this chapter, without any loss of generality it will be assumed that a specific conjugation tuple $\mathbf{c}_l \triangleq (c_1, \ldots, c_l)$ has been chosen and is fixed subsequently. In other words, the theory will be developed for an arbitrary fixed conjugation tuple $\mathbf{c}_l$, but it is applicable to any $\mathbf{c}_l \in \mathcal{C}_l$, where the set $\mathcal{C}_l$ of all $(2)^l$ possible conjugation tuples is defined in (A.5.2). See Sections A.5 and A.6 for more information, and Chapter 5 for examples with $l = 2$.

Similarly to the previous two chapters, for the current scenario several different cumulant functions are required for the formulation of the assumptions and derivation of the theory, viz. the source, noise, and source-noise or noise-source auto- and cross-cumulant functions. For convenience we list their definitions here. Note that we can refer to the source-noise cumulant functions by 'noise-source cumulant functions' as well, and vice versa, because the only thing that matters is that the arguments of the involved cumulant functions include both source and noise random variables. The l-th order source cumulant functions with conjugation tuple $\mathbf{c}_l$ are defined as follows (see (A.6.3)):

$$\begin{aligned} &\kappa^{s,\mathbf{c}_l}_{\mathbf{j}_l}[\mathbf{n}_l] \equiv \kappa^{s,c_1\cdots c_l}_{j_1\cdots j_l}[n_1,\ldots,n_l] \triangleq \operatorname{cum}\Big(\big(s_{j_1}[n_1]\big)^{c_1},\ldots,\big(s_{j_l}[n_l]\big)^{c_l}\Big) \\ &\forall\, \mathbf{n}_l \in \mathbb{Z}_l, \quad \forall\, \mathbf{j}_l \in \mathcal{J}^l_{\mathrm{t},S}, \quad \forall\, \mathbf{c}_l \in \mathcal{C}_l\,. \end{aligned} \tag{6.1.1}$$

See Fig. A.1 on page 408 for an explanation of the various elements of an index set notation symbol like $\mathcal{J}^l_{\mathrm{t},S}$. For equal indices, the corresponding source auto-cumulant functions can be written as:

$$\begin{aligned} &\kappa^{s,\mathbf{c}_l}_{j}[\mathbf{n}_l] \equiv \kappa^{s,\mathbf{c}_l}_{(j)_l}[\mathbf{n}_l] = \kappa^{s,c_1\cdots c_l}_{j\cdots j}[n_1,\ldots,n_l] \triangleq \operatorname{cum}\Big(\big(s_{j}[n_1]\big)^{c_1},\ldots,\big(s_{j}[n_l]\big)^{c_l}\Big) \\ &\forall\, \mathbf{n}_l \in \mathbb{Z}_l, \quad 1 \le j \le S, \quad \forall\, \mathbf{c}_l \in \mathcal{C}_l\,. \end{aligned} \tag{6.1.2}$$

Likewise, the l-th order noise cumulant functions with conjugation tuple $\mathbf{c}_l$ are defined as:

$$\begin{aligned} &\kappa^{\nu,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l] \equiv \kappa^{\nu,c_1\cdots c_l}_{i_1\cdots i_l}[n_1,\ldots,n_l] \triangleq \operatorname{cum}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1},\ldots,\big(\nu_{i_l}[n_l]\big)^{c_l}\Big) \\ &\forall\, \mathbf{n}_l \in \mathbb{Z}_l, \quad \forall\, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}, \quad \forall\, \mathbf{c}_l \in \mathcal{C}_l\,, \end{aligned} \tag{6.1.3}$$

and finally the l-th order source-noise/noise-source cumulant functions with conjugation tuple $\mathbf{c}_l$ are defined by those cumulant functions whose argument list contains at least one element from the source signal vector and one element from the noise signal vector, e.g.:

$$\kappa^{s\cdots s\nu,c_1\cdots c_l}_{j_1\cdots j_{l-1}i_l}[n_1,\ldots,n_l] \triangleq \operatorname{cum}\Big(\big(s_{j_1}[n_1]\big)^{c_1},\ldots,\big(s_{j_{l-1}}[n_{l-1}]\big)^{c_{l-1}},\big(\nu_{i_l}[n_l]\big)^{c_l}\Big)\,,$$

$$\kappa^{\nu s \nu s \cdots s, c_1 \cdots c_l}_{i_1 j_2 i_3 j_4 \cdots j_l}[n_1, \ldots, n_l] \triangleq$$
$$\mathrm{cum}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1}, \big(s_{j_2}[n_2]\big)^{c_2}, \big(\nu_{i_3}[n_3]\big)^{c_3}, \big(s_{j_4}[n_4]\big)^{c_4}, \ldots, \big(s_{j_l}[n_l]\big)^{c_l} \Big) . \qquad (6.1.4)$$

See also (6.2.2), in which functions like these arise naturally. In the sequel, for convenience we will call each function of this type 'source-noise cumulant function' and denote a general instance by $\kappa^{s\nu,\mathbf{c}_l}_{\mathbf{m}_l^{'}}[\mathbf{n}_l]$. Furthermore, the set to which all possible tuples $\mathbf{m}_l$ belong is denoted by $\mathcal{M}^{l}_{\mathrm{t},D,S}$.

As in Sections 4.2 and 5.2, in order to clearly and unambiguously formulate the assumptions, we need to make several definitions regarding the Regions Of Support of the various cumulant functions, see Fig. 6.1. Again, these definitions are necessary for specifying explicitly what conditions are required on the temporal structure/diversity of the source and noise signals, and in what sense the noise signals have a simpler temporal structure than the source signals. Since we have explained the rationale behind the ROS definitions in detail in Section 5.2, here we directly provide the generalizations of the definitions presented there. To start with, let $\Omega^{s,\mathbf{c}_l}_{\mathbf{n}_l} \subseteq \mathbb{Z}_l$ be the *'Source Region Of Support'* in the domain of length-l time index tuples $\mathbf{n}_l \triangleq (n_1, \ldots, n_l)$. This ROS is indicated schematically by the largest ellipse in Fig. 6.1. The *'Noise Region Of Support'* is denoted by $\Omega^{\nu,\mathbf{c}_l}_{\mathbf{n}_l} \subseteq \mathbb{Z}_l$ and indicated by the small tilted ellipse with the one to lightest gray shade. Note that the noise cumulant functions are assumed to be zero outside $\Omega^{\nu,\mathbf{c}_l}_{\mathbf{n}_l}$. Furthermore, the *'Source-Noise Region Of Support'* on which the source-noise cumulant functions generally are non-zero is denoted by $\Omega^{s\nu,\mathbf{c}_l}_{\mathbf{n}_l} \subseteq \mathbb{Z}_l$. Again see Fig. 6.1 for a schematic indication. It is assumed that the source-noise cumulant functions are zero outside $\Omega^{s\nu,\mathbf{c}_l}_{\mathbf{n}_l}$. The union $\Omega^{\nu,\mathbf{c}_l}_{\mathbf{n}_l} \cup \Omega^{s\nu,\mathbf{c}_l}_{\mathbf{n}_l}$ defines the ROS where noise is present. The part of this union that extends into the Source ROS is given by the intersection:

$$\Omega^{s\cap\nu,\mathbf{c}_l}_{\mathbf{n}_l} \triangleq \Omega^{s,\mathbf{c}_l}_{\mathbf{n}_l} \cap \left(\Omega^{\nu,\mathbf{c}_l}_{\mathbf{n}_l} \cup \Omega^{s\nu,\mathbf{c}_l}_{\mathbf{n}_l} \right),$$

and is indicated by the dashed line in the figure. Finally, the *'Noise-Free Region Of Support'*

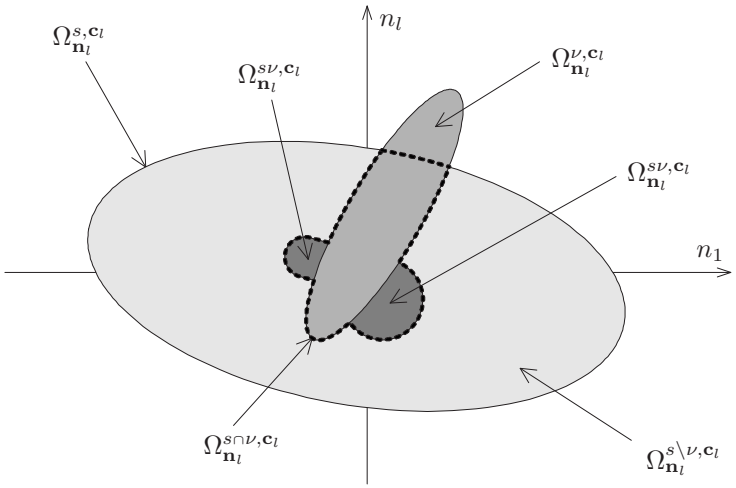

Figure 6.1: *Regions Of Support in the domain of time index tuples* $\mathbf{n}_l$.

is denoted and defined by:

$$\boxed{\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l} \triangleq \Omega_{\mathbf{n}_l}^{s,\mathbf{c}_l} \setminus \Omega_{\mathbf{n}_l}^{s\cap\nu,\mathbf{c}_l} = \Omega_{\mathbf{n}_l}^{s,\mathbf{c}_l} \setminus \left(\Omega_{\mathbf{n}_l}^{\nu,\mathbf{c}_l} \cup \Omega_{\mathbf{n}_l}^{s\nu,\mathbf{c}_l}\right).} \tag{6.1.5}$$

This ROS is indicated by the area with the lightest gray shade, i.e. the large ellipse minus the part surrounded by the dashed line. In order to apply the theory that will be developed in the sequel, the various regions of support have to be known in advance. As we have explained in the previous chapter, knowledge of the Noise-Free ROS is sufficient for the formulation of the assumptions in the remainder of this section. The other regions of support are merely required for defining or finding such a suitable ROS. Hence, in principle the various regions of support are part of the required a priori knowledge for a particular application, but it suffices to know only the Noise-Free ROS $\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}$. In case the conjugation tuple $\mathbf{c}_l = (c_1, \ldots, c_l) = (\circ, \ldots, \circ) = (\circ)_l$ is chosen, i.e. none of the arguments is conjugated, it is often omitted from the symbolic notation or replaced by the order l.

For convenience of notation, let $\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}]$ denote the set/space of all complex-valued functions that depend on a time index tuple $\mathbf{n}_l$ and are defined on the Noise-Free ROS $\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}$. Most functions used in the sequel of this chapter are assumed to be elements of $\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}]$. If functions are defined on another ROS, this will be mentioned explicitly. As we have said on pages 138 and 220 of Sections 4.2 and 5.2 respectively, throughout the thesis we use the dedicated symbol N to denote the cardinality of the Noise-Free Region Of Support. Hence, for the most general scenario considered here we define:

$$N \triangleq \left|\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}\right|. \tag{6.1.6}$$

Using the definitions above for the various regions of support, the assumptions about the MIBI model on which the general results in this chapter are based can be formulated. These assumptions are straightforward generalizations of those discussed in Section 5.2. Since the physical plausibilities of the general assumptions are completely the same as those of their analogues in the previous chapter, which we have already explained in detail in Section 5.2, here we directly formulate the assumptions as *AS1-AS4* in the following list (see Section A.2 for the definitions of the various index sets):

AS1: The source signals are realizations of zero-mean real- or complex-valued random processes with zero cross-cumulant functions on the Noise-Free ROS:

$$\kappa_{\mathbf{j}_l}^{s,\mathbf{c}_l}[\mathbf{n}_l] = 0 \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}, \quad \forall\, \mathbf{j}_l \in \mathcal{J}_{\mathrm{c},S}^{\,l}\,;$$

AS2: The source auto-cumulant functions are linearly independent on the Noise-Free ROS:

$$\sum_{j=1}^{S} \xi^j\, \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] = 0 \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l} \quad \Longrightarrow \quad \xi^j = 0\ \forall\, 1 \le j \le S\,;$$

AS3: The noise signals are realizations of zero-mean real- or complex-valued random processes with zero auto- and cross-cumulant functions on the Noise-Free ROS:

$$\kappa_{\mathbf{i}_l}^{\nu,\mathbf{c}_l}[\mathbf{n}_l] = 0 \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}, \quad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{\,l}\,;$$

AS4: All cross-cumulant functions between the source and noise signals are zero on the Noise-Free ROS:

$$\kappa_{\mathbf{m}_l}^{s\nu,\mathbf{c}_l}[\mathbf{n}_l] = 0 \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}, \quad \forall\, \mathbf{m}_l \in \mathcal{M}_{\mathrm{t},D,S}^{\,l}\,.$$

6.2 Formulating MIBI as the problem of solving a system of D-variate l-homogeneous polyconjugal equations

Using the assumptions made in the previous section, in this section we will show along the same lines as in Section 5.3 that the array response vectors, i.e. the columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of $\mathbf{A}$, satisfy a well-structured *system of D-variate homogeneous polyconjugal equations of degree l and with conjugation tuple $\mathbf{c}_l$*, thereby projecting the MIBI problem onto a mathematical problem, the solution of which yields estimates of the array response vectors. Again, in the course of our derivation we highlight the algebraic structure of the problem formulation.

In Section 6.2.1, we start by examining the kind and structure of the sensor statistics that are available for analysis. This primarily amounts to expressing the sensor cumulant functions in terms of the mixing matrix elements and the source, noise, and joint source-noise cumulant functions, and to studying their properties. Similarly to the previous chapter, our goal is to derive the system of polyconjugal equations referred to above from the sensor cumulant functions in such a way that the sensor noise is annihilated. This can be achieved by considering the employed cumulant functions on the Noise-Free ROS $\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}$. The noise-free sensor cumulant values form the input for the rest of our argument, in which we arrange and consider them in a specific configuration. In Section 6.2.2, we generalize the notation for denoting sets of correlation functions and index pairs explained in Section 5.3.2 to sets of cumulant functions and index tuples, respectively. Likewise, in Section 6.2.3 we generalize the results obtained in Section 5.3.3, i.e. we determine a set containing all unique sensor cumulant functions. Similarly to the previous two chapters, the derivation of the system of equations is formulated in terms of cumulant functions, in terms of the associated row vector notation, and in terms of the associated matrix-vector notation in Sections 6.2.4, 6.2.5 and 6.2.6 respectively. Again, for each formulation the main argument is divided into three stages. Firstly, the noise-free sensor cumulant values are arranged in a specific configuration that allows to exploit the relationship between the various involved linear subspaces on the one hand, and the elements of the mixing matrix on the other hand. Secondly, we derive the form of the system of polyconjugal equations satisfied by the columns of $\mathbf{A}$ by studying the properties of the linear (sub)spaces spanned by the (abstract) vectors occurring in the previous stage. Finally, we show how subspace decomposition using the Singular Value Decomposition (SVD) can be used to obtain proper coefficients for the polyconjugal functions in the system. This stage will complete the derivation of the system. In Section 6.4 a homotopy method for solving the system is discussed.

6.2.1 Structure of sensor cumulant functions

Similarly to Sections 4.3.1 and 5.3.1, we start our derivation by expressing the sensor cumulant functions in terms of the mixing matrix elements and the source auto-cumulant functions. The structure revealed this way will be used in the following sections for developing our subspace based identification method along the lines of Sections 4.3.1 and 5.3.1. We start by giving the definition of l-th order sensor cumulant functions with conjugation tuple $\mathbf{c}_l$ that is consistent with definitions (6.1.1)-(6.1.4):

$$\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] \equiv \kappa_{i_1\cdots i_l}^{x,c_1\cdots c_l}[n_1,\ldots,n_l] \triangleq \mathrm{cum}\Big(\big(x_{i_1}[n_1]\big)^{c_1},\ldots,\big(x_{i_l}[n_l]\big)^{c_l}\Big)$$
$$\forall\ \mathbf{n}_l \in \mathbb{Z}_l, \quad \forall\ \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l, \quad \forall\ \mathbf{c}_l \in \mathcal{C}_l\,. \tag{6.2.1}$$

This definition can be used for stationary as well as for non- or quasi-stationary signals. For all indices $1 \leq i_1, \ldots, i_l \leq D$, time index tuples $\mathbf{n}_l = (n_1, \ldots, n_l) \in \mathbb{Z}_l$, and for all conjugation tuples $\mathbf{c}_l = (c_1, \ldots, c_l) \in \mathcal{C}_l$, the sensor cumulant function $\kappa_{i_1 \cdots i_l}^{x, c_1 \cdots c_l}[n_1, \ldots, n_l]$ of the i_1-th, …, i_l-th sensor signals can be written as follows:

$$
\begin{aligned}
\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] &= \kappa_{i_1 \cdots i_l}^{x, c_1 \cdots c_l}[n_1, \ldots, n_l] \overset{(6.2.1)}{=} \operatorname{cum}\Big(\big(x_{i_1}[n_1]\big)^{c_1}, \ldots, \big(x_{i_l}[n_l]\big)^{c_l} \Big) \\
&\overset{(5.1.1)}{=} \operatorname{cum}\left(\left(\sum_{j_1=1}^{S} a_{i_1}^{j_1}\, s_{j_1}[n_1] + \nu_{i_1}[n_1] \right)^{c_1}, \ldots, \left(\sum_{j_l=1}^{S} a_{i_l}^{j_l}\, s_{j_l}[n_l] + \nu_{i_l}[n_l] \right)^{c_l} \right) \\
&= \sum_{j_1=1}^{S} \cdots \sum_{j_l=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} \cdots \big(a_{i_l}^{j_l}\big)^{c_l}\, \operatorname{cum}\Big(\big(s_{j_1}[n_1]\big)^{c_1}, \ldots, \big(s_{j_l}[n_l]\big)^{c_l} \Big) \\
&\quad + \sum_{j_1=1}^{S} \cdots \sum_{j_{l-1}=1}^{S} \big(a_{i_1}^{j_1}\big)^{c_1} \cdots \big(a_{i_{l-1}}^{j_{l-1}}\big)^{c_{l-1}}\, \operatorname{cum}\Big(\big(s_{j_1}[n_1]\big)^{c_1}, \ldots, \big(s_{j_{l-1}}[n_{l-1}]\big)^{c_{l-1}}, \big(\nu_{i_l}[n_l]\big)^{c_l} \Big) \\
&\quad + \cdots\cdots + \cdots\cdots \\
&\quad + \sum_{j_2=1}^{S} \cdots \sum_{j_l=1}^{S} \big(a_{i_2}^{j_2}\big)^{c_2} \cdots \big(a_{i_l}^{j_l}\big)^{c_l}\, \operatorname{cum}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1}, \big(s_{j_2}[n_2]\big)^{c_2}, \ldots, \big(s_{j_l}[n_l]\big)^{c_l} \Big) \\
&\quad + \cdots\cdots + \cdots\cdots \\
&\quad + \sum_{j_l=1}^{S} \big(a_{i_l}^{j_l}\big)^{c_l}\, \operatorname{cum}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1}, \ldots \big(\nu_{i_{l-1}}[n_{l-1}]\big)^{c_{l-1}}, \big(s_{j_l}[n_l]\big)^{c_l} \Big) \\
&\quad + \operatorname{cum}\Big(\big(\nu_{i_1}[n_1]\big)^{c_1}, \ldots, \big(\nu_{i_l}[n_l]\big)^{c_l} \Big) \qquad \forall\, (n_1, \ldots, n_l) \in \mathbb{Z}_l\,. \qquad (6.2.2)
\end{aligned}
$$

Note the similarity of this derivation to the derivations in (4.3.1) and (5.3.2). Using assumptions *AS1-AS4* on page 299, expression (6.2.2) can be simplified to:

$$
\boxed{
\begin{aligned}
&\kappa_{i_1 \cdots i_l}^{x, c_1 \cdots c_l}[n_1, \ldots, n_l] = \sum_{j=1}^{S} \big(a_{i_1}^{j}\big)^{c_1} \cdots \big(a_{i_l}^{j}\big)^{c_l}\, \operatorname{cum}\Big(\big(s_j[n_1]\big)^{c_1}, \ldots, \big(s_j[n_l]\big)^{c_l} \Big) \\
&\forall\, (n_1, \ldots, n_l) \in \Omega_{n_1 \cdots n_l}^{s \setminus \nu, c_1 \cdots c_l}, \quad \forall\, (i_1, \ldots, i_l) \in \mathcal{I}_{\mathrm{t},D}^{l}\,.
\end{aligned}
} \qquad (6.2.3)
$$

This equation represents the generalization of (4.3.3) and (5.3.4).

6.2.2 Notation for (sets of) index tuples, cumulant functions, etcetera

Similarly to Sections 4.3.2 and 5.3.2 of the previous two chapters, for the sake of convenience of the following developments, the smooth generalization of various results, and uniformity and compactness of notation, we now first discuss several symbols that will be used extensively in the sequel for denoting various quantities in a compact and intuitive form, for denoting sets of cumulant functions and index tuples as well as their associated cardinalities, and so on; see also Appendix A. We start by discussing a symbolic notation required for compactly and uniformly formulating our results.

6.2.2.1 Notation for compact formulation of results

According to (A.2.1) a tuple of indices like $(i_1, \ldots, i_l)$ is denoted by $\mathbf{i}_l$, i.e. $\mathbf{i}_l \triangleq (i_1, \ldots, i_l)$. If all indices in a tuple are equal we use a shorthand notation such as $(i)_l \triangleq (i, \ldots, i)$, see (A.2.2). Likewise, as we have seen already in this chapter a tuple $(n_1, \ldots, n_l)$ of discrete time indices is denoted by $\mathbf{n}_l$, i.e. $\mathbf{n}_l \triangleq (n_1, \ldots, n_l)$, and a conjugation tuple by $\mathbf{c}_l \triangleq (c_1, \ldots, c_l)$. If a tuple of indices is given explicitly in the sub- or superscript position of a symbol, the parentheses and commas are omitted. Similarly, if a time index tuple is given explicitly in the argument position of a cumulant function, the parentheses are omitted. As an example illustrating those points consider both the compact and elaborate symbolic notations for a sensor cumulant function with conjugation tuple $\mathbf{c}_l = (c_1, \ldots, c_l)$ that is indexed by index tuple $\mathbf{i}_l = (i_1, \ldots, i_l)$ and has time index tuple $\mathbf{n}_l = (n_1, \ldots, n_l)$ as its argument:

$$\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \kappa_{(i_1,\ldots,i_l)}^{x,(c_1,\ldots,c_l)}[(n_1, \ldots, n_l)] \equiv \kappa_{i_1\cdots i_l}^{x,c_1\cdots c_l}[n_1, \ldots, n_l]\,. \tag{6.2.4}$$

Likewise, a source auto-cumulant function can be written as follows:

$$\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] = \kappa_{(j)_l}^{s,\mathbf{c}_l}[\mathbf{n}_l] = \kappa_{(j,\ldots,j)}^{s,(c_1,\ldots,c_l)}[(n_1, \ldots, n_l)] = \kappa_{j\cdots j}^{s,c_1\cdots c_l}[n_1, \ldots, n_l]\,. \tag{6.2.5}$$

Note the following equivalence for compactly writing auto-cumulant functions:

$$\kappa_{(j)_l}^{v,\mathbf{c}_l}[\mathbf{n}_l] \;\equiv\; \kappa_j^{v,\mathbf{c}_l}[\mathbf{n}_l] \qquad \text{or} \qquad \kappa_{j\cdots j}^{v,c_1\cdots c_l}[n_1, \ldots, n_l] \;\equiv\; \kappa_j^{v,c_1\cdots c_l}[n_1, \ldots, n_l] \tag{6.2.6}$$

for an arbitrary signal $v[n]$. On many occasions in this chapter we will encounter products of the form $\left(a_{i_1}^{j_1}\right)^{c_1} \cdots \left(a_{i_l}^{j_l}\right)^{c_l}$ with $\mathbf{c}_l \in \mathcal{C}_l$, for example, see (6.2.2) and (6.2.3). As is explained in Section A.2, such a product is denoted by (see (A.2.11)):

$$\check{a}_{\mathbf{i}_l}^{\mathbf{j}_l,\mathbf{c}_l} \equiv \check{a}_{i_1\cdots i_l}^{j_1\cdots j_l,c_1\cdots c_l} \triangleq \left(a_{i_1}^{j_1}\right)^{c_1} \cdots \left(a_{i_l}^{j_l}\right)^{c_l}. \tag{6.2.7}$$

If the indices in $\mathbf{j}_l$ are equal, as in (6.2.3) for example, this becomes (see (A.2.13)):

$$\check{a}_{\mathbf{i}_l}^{(j)_l,\mathbf{c}_l} = \check{a}_{i_1\cdots i_l}^{(j)_l,c_1\cdots c_l} = \check{a}_{i_1\cdots i_l}^{j\cdots j,c_1\cdots c_l} = \left(a_{i_1}^{j}\right)^{c_1} \cdots \left(a_{i_l}^{j}\right)^{c_l} \triangleq \check{a}_{i_1\cdots i_l}^{j,c_1\cdots c_l} = \check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\,. \tag{6.2.8}$$

Here, similarly to (6.2.6) we have defined the following equivalence:

$$\check{a}_{\mathbf{i}_l}^{(j)_l,\mathbf{c}_l} \equiv \check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l} \qquad \text{or} \qquad \check{a}_{i_1\cdots i_l}^{(j)_l,c_1\cdots c_l} = \check{a}_{i_1\cdots i_l}^{j\cdots j,c_1\cdots c_l} \;\equiv\; \check{a}_{i_1\cdots i_l}^{j,c_1\cdots c_l}\,.$$

6.2.2.2 Notation for sets of index tuples and cardinalities

Now we generalize the notation for sets of index pairs and their associated cardinalities discussed in Section 5.3.2.2 to the current scenario; see also Appendix A, in particular Section A.2. The set containing all tuples of indices $\mathbf{i}_l \triangleq (i_1, \ldots, i_l)$ obtained by varying all indices i_1 through i_l from 1 till D is denoted by:

$$\mathcal{I}_{\mathrm{t},D}^{l} = \{(i_1, \ldots, i_l) \mid 1 \le i_1, \ldots, i_l \le D\}\,. \tag{6.2.9}$$

This is a special case of (A.2.14). See Fig. A.1 on page 408 for more information about the meanings of the various parts of an index set notational like $\mathcal{I}_{\mathrm{t},D}^{l}$. The number l in the superscript position of $\mathcal{I}_{\mathrm{t},D}^{l}$ indicates that we are considering l different indices, viz. $i_1, \ldots, i_l$, whereas the number D in the subscript position indicates that all indices range from 1 till D. The first subscript index letter 't' stands for 'total' because the considered set contains all tuples obtained by varying the indices from 1 till D without any further constraints. The cardinality of $\mathcal{I}_{\mathrm{t},D}^{l}$ is denoted and given by:

$$M_{\mathrm{t},D}^{l} \triangleq \left|\mathcal{I}_{\mathrm{t},D}^{l}\right| = (D)^{l}\,. \tag{6.2.10}$$

As an example of this notation, consider the set $\mathcal{J}_{\mathrm{t},2}^{3}$ with index symbol j, $D=2$, and $l=3$:

$$\begin{aligned}\mathcal{J}_{\mathrm{t},2}^{3} &\triangleq \{(j_1,j_2,j_3) \mid 1 \leq j_1,j_2,j_3 \leq 2\} \\ &= \{(1,1,1),(1,1,2),(1,2,1),(1,2,2),(2,1,1),(2,1,2),(2,2,1),(2,2,2)\}\,,\end{aligned}$$

the cardinality $\left|\mathcal{J}_{\mathrm{t},2}^{3}\right|$ of which is given correctly by (6.2.10) as $M_{\mathrm{t},2}^{3}=(2)^3=8$. Other sets can be defined similarly, see Section A.2 and Fig. A.1 for details.

6.2.2.3 Notation for sets of cumulant functions and cardinalities

Now we will generalize the notation for denoting sets of correlation functions and their associated cardinalities discussed in Sections 4.3.2 and 5.3.2.3 to the current scenario. The set containing all sensor cumulant functions with conjugation tuple $\mathbf{c}_l$ is denoted by:

$$\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l} \triangleq \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] \mid \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}\right\} = \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}}. \tag{6.2.11}$$

The cardinality of $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$ is denoted by $M_{\mathrm{t},D}^{x,\mathbf{c}_l}$ and given by:

$$M_{\mathrm{t},D}^{x,\mathbf{c}_l} \triangleq \left|\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}\right| = \left|\mathcal{I}_{\mathrm{t},D}^{l}\right| \overset{(6.2.10)}{=} M_{\mathrm{t},D}^{l} = (D)^l\,. \tag{6.2.12}$$

The set containing all unique sensor cumulant functions, i.e. the uniquiefied version of $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$ defined in (6.2.11), is denoted by:

$$\boxed{\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l} \triangleq \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] \mid \mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}\right\} = \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}},} \tag{6.2.13}$$

where the set $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ of index tuples by definition indexes $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. Similarly to the role of $\mathcal{K}_{\mathrm{u},2}^{x,2}$ (4.3.16) in Chapter 4 and $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_2}$ (5.3.22) in Chapter 5, $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ plays a role of paramount importance in this chapter. For this reason, we present a detailed derivation of this set in the next section. The cardinality of $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, i.e. the number of unique sensor cumulant functions for conjugation tuple $\mathbf{c}_l$, is denoted by:

$$\boxed{M_{\mathrm{u},D}^{x,\mathbf{c}_l} \triangleq \left|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right| = \left|\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}\right|.} \tag{6.2.14}$$

Expressions for this number will also be derived in the next section. The set containing all source auto-cumulant functions is denoted by:

$$\mathcal{K}_{S}^{s,\mathbf{c}_l} \triangleq \left\{\kappa_{j}^{s,\mathbf{c}_l}[\mathbf{n}_l] \mid 1 \leq j \leq S\right\} = \left\{\kappa_{j}^{s,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{1 \leq j \leq S}. \tag{6.2.15}$$

As we have explained in the previous chapters and will be clear by now, we can denote any set of correlation, moment, and/or cumulant functions in this systematic manner.

We conclude this section by using the definitions made above for writing the important expression (6.2.3) compactly as follows:

$$\boxed{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{j=1}^{S} \check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l} \kappa_{j}^{s,\mathbf{c}_l}[\mathbf{n}_l] \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \quad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}\,.} \tag{6.2.16}$$

6.2.3 Determining set of unique sensor cumulant functions

In this section we determine the important 'uniquiefied' sets $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ and $\mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}$ defined by (6.2.13). Similarly to Sections 4.3.3 and 5.3.3, in order to do so we need to study the possible symmetries in the set $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{t},D}$ (6.2.11) containing all sensor cumulant functions that is indexed by $\mathcal{I}^{l}_{\mathrm{t},D}$. Using the concepts explained in Section A.3, in particular Definitions A.3.3 and A.3.9, we define an index tuple permutation function $\sigma(\cdot)$ on the elements of the 'total' index tuple set $\mathcal{I}^{l}_{\mathrm{t},D}$ as follows:

Definition 6.2.1. Index tuple permutation function.
An index tuple permutation function $\sigma(\cdot)$ of order l acting on an index tuple $\mathbf{i}_l \in \mathcal{I}^{l}_{\mathrm{t},D}$ is defined as a permutation without repetition of the elements of $\mathbf{i}_l$. The set of all different index tuple permutation functions for a given index tuple $\mathbf{i}_l$ is denoted by $\Sigma_{\mathbf{i}_l}$. Its cardinality depends on the multiplicities of the different index values in the tuple and is given by the multinomial coefficient (A.3.8) of Definition A.3.9.

Note that if two tuples $\mathbf{i}_l$ and $\mathbf{m}_l$ contain the same indices with the same multiplicities but in a possibly different order, then $\Sigma_{\mathbf{i}_l} = \Sigma_{\mathbf{m}_l}$. A permutation function can be represented by a tuple. As an example consider the index tuple $\mathbf{i}_4 = (1,2,3,2)$. The set of all different permutation functions of order 4 for this tuple is given by $\Sigma_{\mathbf{i}_4} = \{(1,2,2,3),\ (1,2,3,2),\ (1,3,2,2),\ (2,1,2,3),\ (2,1,3,2),\ (2,2,1,3),\ (2,2,3,1),\ (2,3,1,2),\ (2,3,2,1),\ (3,1,2,2),\ (3,2,1,2),(3,2,2,1)\}$. The cardinality of this set is given by (A.3.8) with the set $\mathcal{G}$ in Definition A.3.9 defined as $\mathcal{G} \triangleq \{g_1,g_2,g_3\} = \{1,2,3\}$, and thus $m_1 = 1$, $m_2 = 2$, and $m_3 = 1$. This correctly yields the cardinality of $\Sigma_{\mathbf{i}_4}$ as $P_{1,2,1} = \binom{4}{1,2,1} = \frac{4!}{1!2!1!} = 12$. Note that $\Sigma_{(1,2,2,3)} = \Sigma_{(1,2,3,2)} = \ldots = \Sigma_{(3,2,2,1)}$ and thus an arbitrary element of each set can be chosen to represent the set(s).

Now, we can formulate the study of the symmetries in the set $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{t},D}$ containing all sensor cumulant functions as follows. For each index tuple $\mathbf{i}_l \in \mathcal{I}^{l}_{\mathrm{t},D}$ we have to determine for which different permutation functions $\sigma(\cdot) \in \Sigma_{\mathbf{i}_l}$ the following generalization of (4.3.13) and (5.3.27) holds for any possible mixing matrix $\mathbf{A} \in \mathbb{C}^{S}_{D}$:

$$\boxed{\kappa^{x,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l] = \kappa^{x,\mathbf{c}_l}_{\sigma(\mathbf{i}_l)}[\mathbf{n}_l] \qquad \forall\ \mathbf{n}_l \in \Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}\ ,} \tag{6.2.17}$$

and partition the set $\mathcal{I}^{l}_{\mathrm{t},D}$ accordingly into equivalence classes. In other words, we have to partition all index tuples $\mathbf{i}_l \in \mathcal{I}^{l}_{\mathrm{t},D}$ into equivalence classes according to the equivalence relation $\bowtie$ defined as follows:

$$\mathbf{i}_l \bowtie \mathbf{m}_l \quad \Longleftrightarrow \quad \big(\mathbf{i}_l \in \Sigma_{\mathbf{m}_l}\big) \ \wedge\ \big(\kappa^{x,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l] = \kappa^{x,\mathbf{c}_l}_{\mathbf{m}_l}[\mathbf{n}_l] \quad \forall\ \mathbf{n}_l \in \Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l};\ \forall\,\mathbf{A} \in \mathbb{C}^{S}_{D}\big)\,. \tag{6.2.18}$$

Note that in this definition we require that $\mathbf{i}_l \in \Sigma_{\mathbf{m}_l}$ instead of $\mathbf{i}_l \in \mathcal{I}^{l}_{\mathrm{t},D}$ because for $\mathbf{i}_l \notin \Sigma_{\mathbf{m}_l}$ condition (6.2.17) cannot be satisfied anyway for all possible mixing matrices. The induced partition of the set $\mathcal{I}^{l}_{\mathrm{t},D}$ under equivalence relation (6.2.18) is the set of all equivalence classes. Let $\mathbf{m}_l$ be a representative of an equivalence class, then the equivalence class $\mathcal{C}_{\bowtie}[\mathbf{m}_l]$ of $\mathbf{m}_l$ is the set defined by:

$$\mathcal{C}_{\bowtie}[\mathbf{m}_l] \triangleq \big\{\mathbf{i}_l \in \mathcal{I}^{l}_{\mathrm{t},D} \mid \mathbf{i}_l \bowtie \mathbf{m}_l\big\}\ . \tag{6.2.19}$$

From (6.2.16)-(6.2.19) it now follows that equivalence relation (6.2.18) can be simplified to:

$$\mathbf{i}_l \bowtie \mathbf{m}_l \quad \Longleftrightarrow \quad \big(\mathbf{i}_l \in \Sigma_{\mathbf{m}_l}\big) \quad \wedge \quad \big(\breve{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l} = \breve{a}^{j,\mathbf{c}_l}_{\mathbf{m}_l} \quad \forall\,\mathbf{A} \in \mathbb{C}^{S}_{D}\big)\,.$$

Since for each $1 \leq j \leq S$ the set $\{\check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ has the same structure, it is sufficient to consider a single set with this structure, e.g. a set $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l} = \{(z_{i_1})^{c_1} \cdots (z_{i_l})^{c_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ containing all products of the form $\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l} = \check{z}_{i_1 \cdots i_l}^{c_1 \cdots c_l} \triangleq (z_{i_1})^{c_1} \cdots (z_{i_l})^{c_l}$. Hence, in view of the derivations above we finally can formulate the problem of determining the set of unique sensor cumulant functions as follows. Firstly, partition all index tuples $\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l$ into the equivalence classes defined by (6.2.19) according to the equivalence relation $\bowtie$ that is defined as follows:

$$\mathbf{i}_l \bowtie \mathbf{m}_l \quad \Longleftrightarrow \quad \left(\mathbf{i}_l \in \Sigma_{\mathbf{m}_l}\right) \quad \wedge \quad \left(\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l} = \check{z}_{\mathbf{m}_l}^{\mathbf{c}_l} \quad \forall\, \mathbf{z} \in \mathbb{C}_D\right). \tag{6.2.20}$$

Secondly, define the uniquiefied index tuple set $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ containing exactly one representative from each equivalence class. Finally, use this set to define the set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ containing all unique sensor cumulant functions according to (6.2.13).

In the sequel Kronecker vectors that are related bijectively with sets like $\{\check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ and $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ will arise naturally. Hence, the results obtained for such sets can be translated directly to corresponding results for the associated vectors. Here, we briefly introduce these vectors. To start with, for each fixed index $1 \leq j \leq S$ and conjugation tuple $\mathbf{c}_l \in \mathcal{C}_l$, define the column vector $\check{\mathbf{a}}_D^{j,\mathbf{c}_l} = \check{\mathbf{a}}_D^{j,c_1 \cdots c_l}$ constructed from a single vector $\mathbf{a}^j$ and containing all product terms $\{\check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ of the form specified by (6.2.8) and arranged in 'Kronecker format' as follows (compare this definition with (4.3.68) and (5.3.29)):

$$\check{\mathbf{a}}_D^{j,\mathbf{c}_l} \triangleq \left(\mathbf{a}^j\right)^{c_1} \otimes \cdots \otimes \left(\mathbf{a}^j\right)^{c_l} = \begin{bmatrix} \left(a_1^j\right)^{c_1} \\ \vdots \\ \left(a_D^j\right)^{c_1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} \left(a_1^j\right)^{c_l} \\ \vdots \\ \left(a_D^j\right)^{c_l} \end{bmatrix}. \tag{6.2.21}$$

As we have explained above, in order to determine which product term definitions in the set $\{\check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ are unique, and thus which elements of $\check{\mathbf{a}}_D^{j,\mathbf{c}_l}$ in (6.2.21) are unique, we need to study the symmetries in the product terms. As we have made clear in the previous paragraph, it is sufficient to consider a single set $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ with the same structure as $\{\check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$. Equivalently, since the Kronecker vectors $\{\check{\mathbf{a}}_D^{j,\mathbf{c}_l}\}^{1 \leq j \leq S}$ constructed from the columns of $\mathbf{A}$ all have the same form, it is sufficient to consider a single vector $\check{\mathbf{z}}_D^{\mathbf{c}_l} \equiv \mathbf{w}_D^{\mathbf{c}_l}(\mathbf{z}) = (\mathbf{z})^{c_1} \otimes \cdots \otimes (\mathbf{z})^{c_l}$ with this form that contains all products in $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$. Without any loss of generality, here we consider vectors of this type for studying the symmetries of the considered product terms. Hence, let $\mathbf{z}$ (or $\mathbf{z}_D$) be a column vector of length D, then the vector $\check{\mathbf{z}}_D^{\mathbf{c}_l} \triangleq \mathbf{w}_D^{\mathbf{c}_l}(\mathbf{z})$ is defined similarly to (4.3.106), (5.3.29), (5.3.30), and (6.2.21) as follows:

$$\check{\mathbf{z}}_D^{\mathbf{c}_l} = \check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{z}) \triangleq (\mathbf{z})^{c_1} \otimes \cdots \otimes (\mathbf{z})^{c_l} = \begin{bmatrix} \left(z_1\right)^{c_1} \\ \vdots \\ \left(z_D\right)^{c_1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} \left(z_1\right)^{c_l} \\ \vdots \\ \left(z_D\right)^{c_l} \end{bmatrix}. \tag{6.2.22}$$

Our task now is to construct and count the unique elements of $\check{\mathbf{z}}_D^{\mathbf{c}_l}$. The uniquiefied version of $\check{\mathbf{z}}_D^{\mathbf{c}_l} \triangleq \mathbf{w}_D^{\mathbf{c}_l}(\mathbf{z})$ will be denoted by the vector $\check{\mathbf{z}}_{\mathrm{u},D}^{\mathbf{c}_l} \triangleq \mathbf{w}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathbf{z})$. Clearly, the results depend on the length D of $\mathbf{z}$, as well as on the conjugation tuple $\mathbf{c}_l$. As will we see in the sequel, in as far as the information provided by the conjugation tuple is concerned, the number $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ of unique products is determined only by the number of conjugations, or equivalently, by the number of 'no conjugations'. Hence, $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ does not depend on which particular elements are conjugated or not.

Since the problem of determining which products in sets like $\{\check{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l}\}_{\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}}$ and $\{\check{z}^{\mathbf{c}_l}_{\mathbf{i}_l}\}_{\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}}$ associated with $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{t},D}$ are the same is more complicated than the corresponding one in the previous chapter we will now examine in detail how a unique set of products associated with $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ and indexed by the corresponding index set $\mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}$ can be obtained in a systematic manner. In the next two sections, we will examine the symmetries for the scenarios without and with conjugations respectively.

6.2.3.1 Symmetries for scenario without conjugations

As in Chapter 5, we will first consider the scenario in which no conjugations occur, i.e. with conjugation tuple $\mathbf{c}_l = (\circ)_l = (\circ, \ldots, \circ)$. Hence, in this case the Kronecker product vector $\check{\mathbf{z}}^{\mathbf{c}_l}_D = \check{\mathbf{z}}^{(\circ)_l}_D$ in (6.2.22) reduces to the conventional l-th order Kronecker product $\mathbf{z} \otimes \cdots \otimes \mathbf{z}$ of the column $\mathbf{z}$ with itself:

$$\check{\mathbf{z}}^{(\circ)_l}_D \triangleq \underbrace{\mathbf{z} \otimes \cdots \otimes \mathbf{z}}_{l} . \tag{6.2.23}$$

From (6.2.20) it follows that for each index tuple $\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}$ we have to determine for which unique permutation functions $\sigma(\cdot) \in \Sigma_{\mathbf{i}_l}$ the following condition holds:

$$\check{z}_{\mathbf{i}_l} = \check{z}_{\sigma(\mathbf{i}_l)} \quad \forall\, \mathbf{z} \in \mathbb{C}_D . \tag{6.2.24}$$

Equivalently, for each permutation function $\sigma(\cdot) \in \Sigma_{\mathbf{i}_l}$ we have to determine for which (unique) index tuples $\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}$ condition (6.2.24) holds. Since product terms without conjugations are commutative in each factor, it follows directly that for a given tuple $\mathbf{i}_l$ the condition holds for all other index tuples that are permutations of $\mathbf{i}_l$. Hence, all tuples which contain the same index values with the same multiplicity yield the same product factor. Dividing the tuples into equivalence classes according to this equivalence relation and choosing only one element from each equivalence class yields a set of index tuples that can be used for indexing the unique product terms in the set $\{\check{z}_{\mathbf{i}_l}\}_{\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}} = \{z_{i_1} \cdots z_{i_l}\}_{\mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},D}}$ and the vector $\check{\mathbf{z}}^{(\circ)_l}_D$. Arranging these tuples in ascending order for example (descending order could have been chosen as well because the only requirement is that the tuples are unique) yields the set $\mathcal{I}^l_{\mathrm{a},D}$ defined in (A.2.15) for indexing the unique terms in the current scenario:

$$\mathcal{I}^l_{\mathrm{a},D} \triangleq \{(i_1, \ldots, i_l) \mid 1 \le i_1 \le \cdots \le i_l \le D\} . \tag{6.2.25}$$

The cardinality of (6.2.25) is given by (A.2.15) and (A.3.5) as follows:

$$M^l_{\mathrm{a},D} \triangleq \left|\mathcal{I}^l_{\mathrm{a},D}\right| = \left\langle \begin{matrix} D \\ l \end{matrix} \right\rangle = \frac{(l+D-1)!}{l!(D-1)!} = \binom{l+D-1}{l} = \binom{D-1+l}{D-1} . \tag{6.2.26}$$

As an example, for $D = 2$ and $l = 3$ the vector $\check{\mathbf{z}}^{(\circ)_l}_D$ defined in (6.2.22) and (6.2.23), and its 'uniquiefied' version $\check{\mathbf{z}}^{(\circ)_l}_{\mathrm{u},D}$ become:

$$\check{\mathbf{z}}^{(\circ)_3}_2 = \mathbf{w}^{(\circ)_3}_2(\mathbf{z}) \triangleq \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \otimes \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \otimes \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} z_1 z_1 z_1 \\ z_1 z_1 z_2 \\ z_1 z_2 z_1 \\ z_1 z_2 z_2 \\ z_2 z_1 z_1 \\ z_2 z_1 z_2 \\ z_2 z_2 z_1 \\ z_2 z_2 z_2 \end{bmatrix} \quad \text{and} \quad \check{\mathbf{z}}^{(\circ)_3}_{\mathrm{u},2} = \mathbf{w}^{(\circ)_3}_{\mathrm{u},2}(\mathbf{z}) = \begin{bmatrix} z_1 z_1 z_1 \\ z_1 z_1 z_2 \\ z_1 z_2 z_2 \\ z_2 z_2 z_2 \end{bmatrix} \tag{6.2.27}$$

respectively. Note that the number of elements of $\check{\mathbf{z}}_{\mathrm{u},2}^{(\circ)_3}$ is given correctly as 4 by (6.2.26). The elements of $\check{\mathbf{z}}_{2}^{(\circ)_3}$ and $\check{\mathbf{z}}_{\mathrm{u},2}^{(\circ)_3}$ are indexed by the tuples in the index sets:

$$\mathcal{I}_{\mathrm{t},2}^{3} = \{(1,1,1),(1,1,2),(1,2,1),(1,2,2),(2,1,1),(2,1,2),(2,2,1),(2,2,2)\} \quad (6.2.28)$$

and:

$$\mathcal{I}_{\mathrm{a},2}^{3} = \{(1,1,1),(1,1,2),(1,2,2),(2,2,2)\} \quad (6.2.29)$$

respectively. Note that the set of index tuple equivalence classes for this example is given by:

$$\big\{\{(1,1,1)\},\{(1,1,2),(1,2,1),(2,1,1)\},\{(1,2,2),(2,1,2),(2,2,1)\},\{(2,2,2)\}\big\}$$

and that $\mathcal{I}_{\mathrm{a},2}^{3}$ contains one element from each class. Similarly to (5.3.35), note that the products also satisfy the following relation:

$$\big(\check{z}_{\mathbf{i}_l}^{(\circ)_l}\big)^{*} = \check{z}_{\mathbf{i}_l}^{(*)_l} \qquad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}, \quad \forall\, \mathbf{z} \in \mathbb{C}_D\,, \quad (6.2.30)$$

which shows that $\check{\mathbf{z}}_{D}^{(*)_l} = \big(\check{\mathbf{z}}_{D}^{(\circ)_l}\big)^{*}$ has the same number of unique products as $\check{\mathbf{z}}_{D}^{(\circ)_l}$.

Development of combinatorics for scenario without conjugations

As will have become clear by now, constructing and counting unique product terms in vectors like $\check{\mathbf{z}}_{D}^{(\circ)_l}$ resulting in vectors like $\check{\mathbf{z}}_{\mathrm{u},D}^{(\circ)_l}$ (see (6.2.27)) in essence is a combinatorial problem. In order to be able to perform this construction and counting task in a structured manner that can also be employed for the most general scenario in which also conjugation is involved, in the sequel we develop the combinatorial setting for solving this problem; see also Sections 5.3.3.1 and A.3. To start with, let $\mathcal{Z} \triangleq \{z_1,\ldots,z_D\}$ be a set containing D independent complex variables with cardinality $D \triangleq |\mathcal{Z}|$, and consider the following definition (see also Definitions 5.3.1 and A.3.8):

Definition 6.2.2. ***l-combination with repetition of the set $\mathcal{Z} = \{z_1,\ldots,z_D\}$ containing $D = |\mathcal{Z}|$ independent complex variables.*** An l-combination with repetition of the set $\mathcal{Z} = \{z_1,\ldots,z_D\}$ containing $D = |\mathcal{Z}|$ independent complex variables is an arbitrary unordered selection of l elements from this set, where the elements may be selected repeatedly. In other words, an l-combination with repetition of the set $\mathcal{Z}$ is an arbitrary l-multiset of $\mathcal{Z}$[‡].

The problem of constructing and counting the unique elements of Kronecker product vectors like $\check{\mathbf{z}}_{D}^{(\circ)_l}$ amounts to constructing and counting all possible unique l-combinations with repetition of $\mathcal{Z}$ because each l-combination can be associated bijectively with a product term. For example, let $\mathcal{Z} = \{z_1,z_2\}$ be a set containing $D = 2$ complex variables, $l = 3$, and let us construct and count the unique elements of the vector $\check{\mathbf{z}}_{2}^{(\circ)_3}$ defined in (6.2.27). All 3-combinations with repetition of the set $\mathcal{Z}$ are given by the set of 3-multisets $\mathcal{M}_{2}^{(\circ)_3} = \mathcal{M}_{2}^{\circ\circ\circ}\langle\mathcal{Z}\rangle = \{\langle z_1,z_1,z_1\rangle,\langle z_1,z_1,z_2\rangle,\langle z_1,z_2,z_2\rangle,\langle z_2,z_2,z_2\rangle\}$ with cardinality given by (6.2.26), (6.2.33), and (A.3.5) as $M_{\mathrm{u},2}^{3} \triangleq M_{\mathrm{u},2}^{(\circ)_3} = \frac{(3+2-1)!}{3!(2-1)!} = 4$ (we will derive this soon). Since order in multisets and multiplicity in sets are irrelevant, this set is the same as the 'full set' $\{\langle z_1,z_1,z_1\rangle, \langle z_1,z_1,z_2\rangle, \langle z_1,z_2,z_1\rangle, \langle z_1,z_2,z_2\rangle, \langle z_2,z_1,z_1\rangle, \langle z_2,z_1,z_2\rangle, \langle z_2,z_2,z_1\rangle, \langle z_2,z_2,z_2\rangle\}$ for example, which also has cardinality 4 because it has only 4 unique elements according to the definition of multisets. For example, elements like $\langle z_1,z_1,z_2\rangle$, $\langle z_1,z_2,z_1\rangle$

[‡]See Section A.3 for the definitions of l-combination, multiset, and other related combinatorial concepts.

and $\langle z_2, z_1, z_1 \rangle$ are considered to be the same because they belong to the same multiset-equivalence class.

Let $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}\langle \mathcal{Z} \rangle$ be the set containing all l-multisets of some set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ with cardinality D, i.e. it is the set of all l-combinations with repetition of $\mathcal{Z}$; see Definition 6.2.2. An unambiguous representation $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}(\mathcal{Z})$ of $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}\langle \mathcal{Z} \rangle$ that contains the different l-multisets of $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}\langle \mathcal{Z} \rangle$ exactly once can be defined for example by the set containing exactly one element (represented by an l-tuple) from each multiset-equivalence class. Although it is not necessary, for convenience the elements of $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}(\mathcal{Z})$ may be arranged according to some fixed ordering scheme of the element indices, e.g. ascending order. For the example above with $D = 2$ and $l = 3$ this results in the following representation for the set containing all unique 3-combinations with repetition of the set $\mathcal{Z} = \{z_1, z_2\}$: $\mathcal{M}_{\mathrm{u},2}^{(\circ)_3}(\mathcal{Z}) \triangleq \{(z_1, z_1, z_1)$, (z_1, z_1, z_2), (z_1, z_2, z_2), $(z_2, z_2, z_2)\}$. Clearly, this set can be indexed by an index set of the type in (6.2.25) with $l = 3$. Calling the index i for example, the index set associated with $\mathcal{M}_{\mathrm{u},2}^{(\circ)_2}(\mathcal{Z})$ is given by: $\mathcal{I}_{\mathrm{u},2}^{3} = \{(1,1,1), (1,1,2), (1,2,2), (2,2,2)\}$, the cardinality of which is given as 4 by (6.2.33) and (A.2.15). The product terms associated with the elements of $\mathcal{M}_{\mathrm{u},2}^{(\circ)_2}(\mathcal{Z})$ are $z_1z_1z_1, z_1z_1z_2, z_1z_2z_2$ and $z_2z_2z_2$. Clearly, there is a one-to-one correspondence between $\mathcal{M}_{\mathrm{u},2}^{(\circ)_3}(\mathcal{Z})$ and the uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},2}^{(\circ)_3}$ of $\check{\mathbf{z}}_{2}^{(\circ)_3}$ defined in (6.2.27). This also holds for the general situation with D sensors and l-th order statistics, i.e.:

> There is a one-to-one correspondence between the set $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}(\mathcal{Z})$ containing all unique l-multisets of $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}\langle \mathcal{Z} \rangle$ exactly once and the vector $\mathbf{z}_{\mathrm{u},D}^{(\circ)_l}$ of unique products.

In general, with Definition 6.2.2 in place we can *formulate the problem of constructing and counting all unique product terms of $\check{\mathbf{z}}_{D}^{(\circ)_l}$ as that of constructing and counting the set $\mathcal{M}_{\mathrm{u},D}^{(\circ)_l}(\mathcal{Z})$ containing all unique l-combinations with repetition of the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$.*

We will now investigate the problem that we have just described in detail; see also Sections 5.3.3.1 and A.3. Consider a set $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ containing $D \triangleq |\mathcal{Z}|$ independent variables, from which we are to construct and count all possible l-multisets, i.e. all l-combinations with repetition. As we have indicated above, the set containing these l-combinations is denoted by $\mathcal{M}_{D}^{(\circ)_l}\langle \mathcal{Z} \rangle$. To start with, consider a specific l-multiset containing m_1 times element z_1, m_2 times element z_2, and so on, till m_D times element z_D, i.e. m_i is the multiplicity of element z_i. Note that the sum $m_1 + \cdots + m_D$ of the multiplicities of $z_1, \ldots, z_D$ respectively equals l because each multiset contains exactly l elements. Hence, $m_i \in \{0, 1, \ldots, l\}$ for all $1 \leq i \leq D$. The l-multiset can be represented uniquely by the following set:

$$\underbrace{\big\{\{\underbrace{z_1, \ldots, z_1}_{m_1}\}, \{\underbrace{z_2, \ldots, z_2}_{m_2}\}, \ldots, \{\underbrace{z_D, \ldots, z_D}_{m_D}\}\big\}}_{m_1+\cdots+m_D = l \text{ elements from } \mathcal{Z}} . \tag{6.2.31}$$

As in Section 5.3.3.1, in turn this set can be represented uniquely by an ordered sequence of l arbitrary symbols denoting the elements of the set, e.g. bullets, and $D - 1$ other arbitrary

symbols denoting the 'transition to the next element', e.g. vertical bars, as follows:

$$\underbrace{\Big(\underbrace{\bullet\cdots\bullet}_{m_1}\overbrace{|}^{D-1\text{ bars}}\underbrace{\bullet\cdots\bullet}_{m_2}|\cdots|\underbrace{\bullet\cdots\bullet}_{m_D}\Big)}_{\substack{m_1+\cdots+m_D=l\text{ bullets}\\ l+(D-1)\text{ symbols in total}}}. \tag{6.2.32}$$

The number m_1 of bullets to the left of the first bar represents the multiplicity of z_1, the number of bullets m_2 between the first and second bar represents the multiplicity of z_2, and so on. Finally, the number of bullets m_D to the right of the last bar represents the multiplicity of z_D. The total number of bullets equals the length of the multiset, i.e. $m_1 + \cdots + m_D = l$. Since there are D different elements in $\mathcal{Z}$, the number of separating bars equals $D-1$. Hence, the total number of symbols in (6.2.32) including both bullets and bars equals $(m_1 + \cdots + m_D) + (D-1) = l + (D-1)$. Thus, we can conclude that the number of l-multisets of $\mathcal{Z}$ equals the number of ways to arrange the $D-1$ bars among the $l+D-1$ symbols in (6.2.32), which equals the number of subsets of size $D-1$ in a set of size $l+D-1$, which is given by the binomial coefficient $\binom{l+D-1}{D-1} = \binom{l+D-1}{l}$. In conclusion, *the number of l-multisets of a set $\mathcal{Z}$ with cardinality D is the same as the number of l-subsets of a set of size $l+D-1$*, which is expressed as follows (see also (A.3.5)):

$$M_{\mathrm{u},D}^{(\circ)_l} \triangleq M_{\mathrm{u},D}^{l} \triangleq \left\langle \begin{matrix} D \\ l \end{matrix} \right\rangle = \frac{(l+D-1)!}{l!(D-1)!} = \binom{l+D-1}{l} = \binom{l+D-1}{D-1}. \tag{6.2.33}$$

This function is sometimes termed D *multichoose* l.

Summarizing the results of this section, for the scenario without conjugations, with D sensors, and with statistical order l, the set of unique sensor cumulant functions and corresponding set of index tuples are given by:

$$\mathcal{K}_{\mathrm{u},D}^{x,(\circ)_l} \equiv \mathcal{K}_{\mathrm{u},D}^{x,l} = \left\{\kappa_{\mathbf{i}_l}^{x}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{l}} \tag{6.2.34}$$

and:

$$\mathcal{I}_{\mathrm{u},D}^{(\circ)_l} \equiv \mathcal{I}_{\mathrm{u},D}^{l} = \mathcal{I}_{\mathrm{a},D}^{l} \overset{(6.2.25)}{=} \{(i_1,\ldots,i_l) \mid 1 \le i_1 \le \cdots \le i_l \le D\}$$

respectively. Both sets have cardinality $M_{\mathrm{u},D}^{x,(\circ)_l} \triangleq M_{\mathrm{u},D}^{x,l} = M_{\mathrm{u},D}^{(\circ)_l} = M_{\mathrm{u},D}^{l}$ given by (6.2.26) and (6.2.33). Table 6.1 on the next page lists the values of $M_{\mathrm{u},D}^{l} = M_{\mathrm{u},D}^{x,(\circ)_l} = M_{\mathrm{u},D}^{x,(*)_l} \triangleq M_{\mathrm{u},D}^{x,l} = \left|\mathcal{K}_{\mathrm{u},D}^{x,(\circ)_l}\right|$ for $0 \le l, D \le 5$.

6.2.3.2 Symmetries for scenario with conjugations: most general case

Now we are in a position to consider the most general and most complex case of scenarios with arbitrary conjugation patterns. In this section, we will develop the combinatorial setting for solving the problem of constructing and counting the unique elements of a vector like $\check{\mathbf{z}}_D^{\mathbf{c}_l}$. That is, we examine the problem of finding and counting the unique terms in $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ for an arbitrary conjugation tuple $\mathbf{c}_l$ in a manner similar to the one presented at the end of the previous section and in Section 5.3.3.2, thereby constructing its uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},D}^{\mathbf{c}_l}$, the set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ (6.2.13) of unique sensor cumulant functions, the associated set $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ of index tuples, and the cardinality $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ of these sets. Similarly to the example discussed in the

Table 6.1: *Number $M^l_{\mathrm{u},D}$ of unique products and sensor cumulant functions.*

$l \backslash D$	1	2	3	4	5
0	1	1	1	1	1
1	1	2	3	4	5
2	1	3	6	10	15
3	1	4	10	20	35
4	1	5	15	35	70
5	1	6	21	56	126

previous section, the unique product terms in the vector $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ can be obtained by retaining only one element from each equivalence class containing the same product terms, i.e. by reducing the multiplicity of products that occur more than once to one. As a thinking framework and reference example for the coming developments, consider the scenario with $D = 2$ complex variables and conjugation tuple $\mathbf{c}_3 \triangleq (\circ, \circ, *)$, i.e. $l = 3$. Substituting these values into (6.2.22) gives $\check{\mathbf{z}}_2^{\circ\circ *} = \mathbf{w}_2^{\circ\circ *}(\mathbf{z})$ and its uniquiefied version $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ *} = \mathbf{w}_{\mathrm{u},2}^{\circ\circ *}(\mathbf{z})$ as:

$$\check{\mathbf{z}}_2^{\circ\circ *} \triangleq \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \otimes \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \otimes \begin{bmatrix} (z_1)^* \\ (z_2)^* \end{bmatrix} = \begin{bmatrix} z_1 z_1 (z_1)^* \\ z_1 z_1 (z_2)^* \\ z_1 z_2 (z_1)^* \\ z_1 z_2 (z_2)^* \\ z_2 z_1 (z_1)^* \\ z_2 z_1 (z_2)^* \\ z_2 z_2 (z_1)^* \\ z_2 z_2 (z_2)^* \end{bmatrix} \quad \text{and} \quad \check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ *} = \begin{bmatrix} z_1 z_1 (z_1)^* \\ z_1 z_1 (z_2)^* \\ z_1 z_2 (z_1)^* \\ z_1 z_2 (z_2)^* \\ z_2 z_2 (z_1)^* \\ z_2 z_2 (z_2)^* \end{bmatrix} \tag{6.2.35}$$

respectively. The number of unique product terms in $\check{\mathbf{z}}_2^{\circ\circ *}$, i.e. the number of elements in $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ *}$, equals 6 because on the one hand the third and fifth, and on the other the fourth and sixth elements of $\check{\mathbf{z}}_2^{\circ\circ *}$ are the same.

Development of combinatorics for scenario with conjugations

To start the analysis, let $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ be a set containing D independent complex variables with cardinality $D \triangleq |\mathcal{Z}|$ and let the set containing the complex conjugates of the elements of $\mathcal{Z}$ be denoted by $\mathcal{Z}^*$, i.e. $\mathcal{Z}^* = \{(z_1)^*, \ldots, (z_D)^*\}$. Then, we define the *compound set* $\mathcal{Z}^{\circ *}$ as the set containing the elements of both $\mathcal{Z}$ and $\mathcal{Z}^*$:

$$\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\} = \{z_1, \ldots, z_D, (z_1)^*, \ldots, (z_D)^*\}.$$

Furthermore, we denote the number of conjugations in the conjugation tuple $\mathbf{c}_l$ by $n_{\mathbf{c}_l}$:

$$n_{\mathbf{c}_l} \triangleq \#\{\text{conjugations in } \mathbf{c}_l\} \equiv \#_*\{\mathbf{c}_l\} \quad \in \quad \{0, \ldots, l\}. \tag{6.2.36}$$

Since each element of $\mathbf{c}_l$ can have two different values, there are $(2)^l$ different conjugation tuples; see Section A.5. Now, the generalization of Definition 6.2.2 to the current scenario can be given as follows:

Definition 6.2.3. ***l-combination with conjugation tuple $\mathbf{c}_l$ with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$ constructed from $\mathcal{Z} = \{z_1, \ldots, z_D\}$ containing $D = |\mathcal{Z}|$ independent complex variables.*** An l-combination with conjugation tuple $\mathbf{c}_l$ with repetition of the set

$\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$ constructed from the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ containing D independent complex variables is an arbitrary unordered selection of l elements from this set, where the elements may be selected repeatedly but $l - n_{\mathbf{c}_l}$ elements have to be chosen from $\mathcal{Z}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^*$. In other words, an l-combination with conjugation tuple $\mathbf{c}_l$ with repetition of the set $\mathcal{Z}^{\circ *}$ is an arbitrary l-multiset of $\mathcal{Z}^{\circ *}$ containing $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^*$.

With this definition in place, *the problem of finding and counting the unique terms in the vector* $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ *and/or set* $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l}$ *for an arbitrary conjugation tuple* $\mathbf{c}_l$ *can be formulated as that of constructing and counting all possible* l*-combinations with conjugation tuple* $\mathbf{c}_l$ *with repetition of the set* $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$*, where the set* $\mathcal{Z} = \{z_1, \ldots, z_D\}$ *contains* D *independent complex variables,* $l - n_{\mathbf{c}_l}$ *elements have to be chosen from* $\mathcal{Z}$*, and* $n_{\mathbf{c}_l}$ *elements have to be chosen from* $\mathcal{Z}^*$*. This can be done because each such* l*-combination with conjugation pair* $\mathbf{c}_l$ *can be associated bijectively with a product term.* For example, let $\mathcal{Z} = \{z_1, z_2\}$ be a set containing $D = 2$ complex variables and let the conjugation tuple be given by $\mathbf{c}_3 \triangleq (\circ, \circ, *)$, i.e. $l = 3$. This scenario corresponds to the vector $\check{\mathbf{z}}_2^{\circ\circ *}$ defined in (6.2.35). All 3-combinations with conjugation tuple $\mathbf{c}_3 = (\circ, \circ, *)$ with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\} = \{z_1, z_2, (z_1)^*, (z_2)^*\}$ are given by the set of 3-multisets $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}\langle\mathcal{Z}\rangle = \{\langle z_1, z_1, (z_1)^*\rangle, \langle z_1, z_1, (z_2)^*\rangle, \langle z_1, z_2, (z_1)^*\rangle, \langle z_1, z_2, (z_2)^*\rangle, \langle z_2, z_2, (z_1)^*\rangle, \langle z_2, z_2, (z_2)^*\rangle\}$ with cardinality given by (6.2.41) as 6 (we will derive this soon). Since order is irrelevant in multisets, this set is the same as the set $\{\langle z_1, (z_1)^*, z_1\rangle, \langle z_1, (z_2)^*, z_1\rangle, \langle z_1, z_2, (z_1)^*\rangle, \langle z_1, z_2, (z_2)^*\rangle, \langle z_2, z_1, (z_1)^*\rangle, \langle z_2, z_1, (z_2)^*\rangle, \langle z_2, z_2, (z_1)^*\rangle, \langle z_2, (z_2)^*, z_2\rangle\}$, for example. The unique product terms associated with the elements of $\mathcal{M}_3^{\circ\circ *}\langle\mathcal{Z}\rangle$ are $z_1 z_1 (z_1)^*$, $z_1 z_1 (z_2)^*$, $z_1 z_2 (z_1)^*$, $z_1 z_2 (z_2)^*$, $z_2 z_2 (z_1)^*$ and $z_2 z_2 (z_2)^*$.

Let $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}\langle\mathcal{Z}\rangle$ be the set containing all l-multisets of $\mathcal{Z}^{\circ *}$ containing $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^*$, i.e. it is the set of all l-combinations with conjugation tuple $\mathbf{c}_l$ with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains $D = |\mathcal{Z}|$ independent complex variables; see Definition 6.2.3. An unambiguous representation $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$ of $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}\langle\mathcal{Z}\rangle$ that contains the different l-multisets of $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}\langle\mathcal{Z}\rangle$ exactly once can be defined for example by the set containing exactly one element from each multiset-equivalence class, which is represented by an l-tuple. Although it is not necessary, for convenience the elements of $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$ may be arranged according to some fixed ordering scheme of the element indices, e.g. ascending order. For the example above with $D = 2$ and $l = 3$ this results in the following representation for $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}\langle\mathcal{Z}\rangle$: $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}(\mathcal{Z}) = \{z_1 z_1 (z_1)^*, z_1 z_1 (z_2)^*, z_1 z_2 (z_1)^*, z_1 z_2 (z_2)^*, z_2 z_2 (z_1)^*, z_2 z_2 (z_2)^*\}$. Calling the index i for example, the index set associated with $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}(\mathcal{Z})$ is given by: $\mathcal{I}_{\mathrm{u},2}^{\circ\circ *} = \{(1,1,1), (1,1,2), (1,2,1), (1,2,2), (2,2,1), (2,2,2)\}$, the cardinality of which will be derived soon and is given as 6 by (6.2.41). The product terms associated with the elements of $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}(\mathcal{Z})$ are $z_1 z_1 (z_1)^*$, $z_1 z_1 (z_2)^*$, $z_1 z_2 (z_1)^*$, $z_1 z_2 (z_2)^*$, $z_2 z_2 (z_1)^*$ and $z_2 z_2 (z_2)^*$. Clearly, there is a one-to-one correspondence between $\mathcal{M}_{\mathrm{u},2}^{\circ\circ *}(\mathcal{Z})$ and the uniquified version $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ *}$ of $\check{\mathbf{z}}_2^{\circ\circ *}$ given in (6.2.35). This also holds for the general situation with D sensors, l-th order statistics, and conjugation tuple $\mathbf{c}_l$, i.e.:

> There is a one-to-one correspondence between the set $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$ containing all unique l-multisets of $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}\langle\mathcal{Z}\rangle$ exactly once and the vector $\mathbf{z}_{\mathrm{u},D}^{\mathbf{c}_l}$ of unique products.

In general, with Definition 6.2.3 in place we can *formulate the problem of constructing and counting all unique terms of* $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ *for an arbitrary conjugation tuple* $\mathbf{c}_l$ *as that of constructing*

and counting the set $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$ containing all unique l-combinations with conjugation tuple $\mathbf{c}_l$ with repetition of the set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z}, \mathcal{Z}^*\}$, where the set $\mathcal{Z} = \{z_1, \ldots, z_D\}$ contains D independent complex variables, $l - n_{\mathbf{c}_l}$ elements have to be chosen from $\mathcal{Z}$, and $n_{\mathbf{c}_l}$ elements have to be chosen from $\mathcal{Z}^*$.

We will now investigate the problem that we have just described in detail; see also Chapter 5 and Section A.3. Consider again the set $\mathcal{Z} \triangleq \{z_1, \ldots, z_D\}$ containing $D \triangleq |\mathcal{Z}|$ complex variables, its 'conjugate set' $\mathcal{Z}^* \triangleq \{(z_1)^*, \ldots, (z_D)^*\}$, and the compound set $\mathcal{Z}^{\circ *} = \{z_1, \ldots, z_D, (z_1)^*, \ldots, (z_D)^*\}$. As we have explained above, our task is to construct and count the number of all unique l-multisets of $\mathcal{Z}^{\circ *}$ containing $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z}$ and $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^*$. To start with, consider a specific l-multiset containing m_1 times element z_1, and so on, till m_D times element z_D, m_1^* times element $(z_1)^*$, and so on, till m_D^* times element $(z_D)^*$, i.e. the multiplicities of the i-th element z_i of $\mathcal{Z}$ and the i-th element z_i^* of $\mathcal{Z}^*$ are denoted by m_i and m_i^* respectively. Since each multiset contains $l - n_{\mathbf{c}_l}$ elements from $\mathcal{Z}$, it follows that $m_1 + \cdots + m_D = l - n_{\mathbf{c}_l}$ with $m_i \in \{0, \ldots, l - n_{\mathbf{c}_l}\}$ for all $1 \leq i \leq D$. Likewise, since each multiset contains $n_{\mathbf{c}_l}$ elements from $\mathcal{Z}^*$, it follows that $m_1^* + \cdots + m_D^* = n_{\mathbf{c}_l}$ with $m_i^* \in \{0, \ldots, n_{\mathbf{c}_l}\}$ for all $1 \leq i \leq D$. Note that the sum of all multiplicities equals $(m_1 + \cdots + m_D) + (m_1^* + \cdots + m_D^*) = l$. The l-multiset can be represented uniquely by the following set:

$$\Big\{ \underbrace{\underbrace{\{z_1, \ldots, z_1\}}_{m_1}, \ldots, \underbrace{\{z_D, \ldots, z_D\}}_{m_D}}_{(l - n_{\mathbf{c}_l}) \text{ elements from } \mathcal{Z}}, \underbrace{\underbrace{\{(z_1)^*, \ldots, (z_1)^*\}}_{m_1^*}, \ldots, \underbrace{\{(z_D)^*, \ldots, (z_D)^*\}}_{m_D^*}}_{n_{\mathbf{c}_l} \text{ elements from } \mathcal{Z}^*} \Big\} \; . \tag{6.2.37}$$

l elements from $\mathcal{Z}^{\circ *} = \{\mathcal{Z}, \mathcal{Z}^*\}$

Our problem can be decomposed into two independent problems each of which is similar to the one discussed in the previous section for the situation without conjugation(s). In one of the problems all possible $(l - n_{\mathbf{c}_l})$-combinations with repetition of $\mathcal{Z} = \{z_1, \ldots, z_D\}$ have to be constructed and counted, whereas in the other the same has to be done for all possible $n_{\mathbf{c}_l}$-combinations with repetition of $\mathcal{Z}^* \triangleq \{(z_1)^*, \ldots, (z_D)^*\}$. Hence, constructing and counting all unique elements of the form (6.2.37) amounts to doing the same for each of the two sets:

$$\underbrace{\Big\{ \underbrace{\{z_1, \ldots, z_1\}}_{m_1}, \ldots, \underbrace{\{z_D, \ldots, z_D\}}_{m_D} \Big\}}_{(l - n_{\mathbf{c}_l}) \text{ elements from } \mathcal{Z}}, \underbrace{\Big\{ \underbrace{\{(z_1)^*, \ldots, (z_1)^*\}}_{m_1^*}, \ldots, \underbrace{\{(z_D)^*, \ldots, (z_D)^*\}}_{m_D^*} \Big\}}_{n_{\mathbf{c}_l} \text{ elements from } \mathcal{Z}^*}$$

separately, and then combining the results using the Cartesian product rule. Following the same reasoning as the one below (5.3.47) leading to results (5.3.48)-(5.3.52), and the one below (6.2.31) leading to result (6.2.33), these two sets are represented as:

$$\underbrace{\Big(\underbrace{\bullet \cdots \bullet}_{m_1} \overbrace{\big| \underbrace{\bullet \cdots \bullet}_{m_2} \big| \cdots \big|}^{D-1 \text{ bars}} \underbrace{\bullet \cdots \bullet}_{m_D} \Big)}_{\substack{m_1 + \cdots + m_D = (l - n_{\mathbf{c}_l}) \text{ bullets} \\ (D - n_{\mathbf{c}_l} + 1) \text{ symbols in total}}} \quad \text{and} \quad \underbrace{\Big(\underbrace{\bullet \cdots \bullet}_{m_1^*} \overbrace{\big| \underbrace{\bullet \cdots \bullet}_{m_2^*} \big| \cdots \big|}^{D-1 \text{ bars}} \underbrace{\bullet \cdots \bullet}_{m_D^*} \Big)}_{\substack{m_1^* + \cdots + m_D^* = n_{\mathbf{c}_l} \text{ bullets} \\ (D + n_{\mathbf{c}_l} - 1) \text{ symbols in total}}} \tag{6.2.38}$$

respectively (see also (6.2.32)). Since there are D different elements in both $\mathcal{Z}$ and $\mathcal{Z}^*$, the number of separating bars equals $D-1$ for each set. Hence, the total number of symbols including bullets and bars equals $(m_1+\cdots+m_D)+(D-1)=(l-n_{\mathbf{c}_l})+(D-1)$ for the first set, and $(m_1^*+\cdots+m_D^*)+(D-1)=n_{\mathbf{c}_l}+(D-1)$ for the second. Thus, the number of $(l-n_{\mathbf{c}_l})$-multisets of $\mathcal{Z}$ equals the number of ways to arrange $D-1$ bars among $((l-n_{\mathbf{c}_l})+(D-1))$ symbols, which equals the number of subsets of size $D-1$ or $l-n_{\mathbf{c}_l}$ in a set of size $((l-n_{\mathbf{c}_l})+(D-1))$ and is given by the binomial coefficient as follows (see also (6.2.33) and (A.3.5)):

$$M_{\mathrm{u},D}^{l-n_{\mathbf{c}_l}} \triangleq \left\langle \begin{matrix} D \\ l-n_{\mathbf{c}_l} \end{matrix} \right\rangle = \binom{(l-n_{\mathbf{c}_l})+(D-1)}{l-n_{\mathbf{c}_l}} = \binom{(l-n_{\mathbf{c}_l})+(D-1)}{D-1}. \tag{6.2.39}$$

Likewise, the number of $n_{\mathbf{c}_l}$-multisets of $\mathcal{Z}^*$ equals the number of ways to arrange $D-1$ bars among $n_{\mathbf{c}_l}+(D-1)$ symbols, which equals the number of subsets of size $D-1$ or $n_{\mathbf{c}_l}$ in a set of size $n_{\mathbf{c}_l}+(D-1)$ and is also given by the binomial coefficient as follows:

$$M_{\mathrm{u},D}^{n_{\mathbf{c}_l}} \triangleq \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle = \binom{n_{\mathbf{c}_l}+(D-1)}{n_{\mathbf{c}_l}} = \binom{n_{\mathbf{c}_l}+(D-1)}{D-1}. \tag{6.2.40}$$

Combining (6.2.39) and (6.2.40) using the standard product rule from combinatorics gives the number of l-multisets of a set $\mathcal{Z}^{\circ *} \triangleq \{\mathcal{Z},\mathcal{Z}^*\}$ containing $l-n_{\mathbf{c}_l}$ elements from $\mathcal{Z}$ (having cardinality D) and $n_{\mathbf{c}_l}$ from $\mathcal{Z}^*$ (also having cardinality D) as the product of the number of $(l-n_{\mathbf{c}_l})$-subsets of a set of size $((l-n_{\mathbf{c}_l})+(D-1))$ on the one hand, and the number of $n_{\mathbf{c}_l}$-subsets of a set of size $n_{\mathbf{c}_l}+(D-1)$ on the other hand:

$$M_{\mathrm{u},D}^{\mathbf{c}_l} = \left\langle \begin{matrix} D \\ l-n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle = \binom{(l-n_{\mathbf{c}_l})+(D-1)}{l-n_{\mathbf{c}_l}} \binom{n_{\mathbf{c}_l}+(D-1)}{n_{\mathbf{c}_l}}. \tag{6.2.41}$$

Hence, the number of unique elements in the vector $\check{\mathbf{z}}_D^{\mathbf{c}_l} = \mathbf{w}_D^{\mathbf{c}_l}(\mathbf{z})$ and/or set $\{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^l}$ can be expressed as follows in various ways:

$$\begin{aligned} M_{\mathrm{u},D}^{\mathbf{c}_l} &= M_{\mathrm{u},D}^{l-n_{\mathbf{c}_l}} \cdot M_{\mathrm{u},D}^{n_{\mathbf{c}_l}} = \left\langle \begin{matrix} D \\ l-n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle = \binom{(l-n_{\mathbf{c}_l})+(D-1)}{l-n_{\mathbf{c}_l}} \binom{n_{\mathbf{c}_l}+(D-1)}{n_{\mathbf{c}_l}} \\ &= \binom{(l-n_{\mathbf{c}_l})+(D-1)}{D-1} \binom{n_{\mathbf{c}_l}+(D-1)}{D-1} = \frac{\big((l-n_{\mathbf{c}_l})+(D-1)\big)!\big(n_{\mathbf{c}_l}+(D-1)\big)!}{(D-1)!(l-n_{\mathbf{c}_l})!(D-1)!(n_{\mathbf{c}_l})!} \\ &= \frac{\big((l-n_{\mathbf{c}_l})+(D-1)\big)\cdot\cdots\cdot D}{(l-n_{\mathbf{c}_l})!} \cdot \frac{\big(n_{\mathbf{c}_l}+(D-1)\big)\cdot\cdots\cdot D}{(n_{\mathbf{c}_l})!}, \end{aligned} \tag{6.2.42}$$

or, more compactly:

$$\boxed{M_{\mathrm{u},D}^{\mathbf{c}_l} = \left\langle \begin{matrix} D \\ l-n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle = \binom{(l-n_{\mathbf{c}_l})+(D-1)}{l-n_{\mathbf{c}_l}} \binom{n_{\mathbf{c}_l}+(D-1)}{n_{\mathbf{c}_l}}.} \tag{6.2.43}$$

Summarizing the results of this section, using the procedure for constructing $\mathcal{M}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathcal{Z})$ explained above and the various one-to-one correspondences, for the general scenario with an arbitrary conjugation tuple $\mathbf{c}_l$ we can construct the vector $\check{\mathbf{z}}_{\mathrm{u},D}^{\mathbf{c}_l}$ of unique products and the set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ of unique sensor cumulant functions, both with the associated set $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ of index tuples. The cardinality $M_{\mathrm{u},D}^{\mathbf{c}_l} = M_{\mathrm{u},D}^{x,\mathbf{c}_l} \triangleq \big|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big|$ of these sets is given by (6.2.41)-(6.2.43), and its values are listed in Table 6.2 on the following page for several combinations of $\mathbf{c}_l$ and D. Note that the values in one row of the table can obtained by multiplying two rows of

Table 6.2: *Number $M_{\mathrm{u},D}^{\mathbf{c}_l}$ of unique products and sensor cumulant functions.*

Conjugation tuple parameters			$M_{\mathrm{u},D}^{\mathbf{c}_l}$	Number of sensors D				
l	$\mathbf{c}_l$	$n_{\mathbf{c}_l}$	$\left\langle {D \atop n_{\mathbf{c}_l}} \right\rangle \left\langle {D \atop l-n_{\mathbf{c}_l}} \right\rangle$	1	2	3	4	5
1	$(\circ), (*)$	0, 1	$\left\langle {D \atop 0} \right\rangle \left\langle {D \atop 1} \right\rangle$	1	2	3	4	5
2	$(\circ,\circ), (*,*)$	0, 2	$\left\langle {D \atop 0} \right\rangle \left\langle {D \atop 2} \right\rangle$	1	3	6	10	15
	$(\circ,*), (*,\circ)$	1, 1	$\left\langle {D \atop 1} \right\rangle \left\langle {D \atop 1} \right\rangle$	1	4	9	16	25
3	$(\circ,\circ,\circ), (*,*,*)$	0, 3	$\left\langle {D \atop 0} \right\rangle \left\langle {D \atop 3} \right\rangle$	1	4	10	20	35
	$(\circ,\circ,*), (\circ,*,*)$	1, 2	$\left\langle {D \atop 1} \right\rangle \left\langle {D \atop 2} \right\rangle$	1	6	18	40	75
4	$(\circ,\circ,\circ,\circ), (*,*,*,*)$	0, 4	$\left\langle {D \atop 0} \right\rangle \left\langle {D \atop 4} \right\rangle$	1	5	15	35	70
	$(\circ,\circ,\circ,*), (\circ,*,*,*)$	1, 3	$\left\langle {D \atop 1} \right\rangle \left\langle {D \atop 3} \right\rangle$	1	8	30	80	175
	$(\circ,\circ,*,*)$	2, 2	$\left\langle {D \atop 2} \right\rangle \left\langle {D \atop 2} \right\rangle$	1	9	36	100	225

Table 6.1, as indicated in the fourth column of Table 6.2. Note that for each l the first row gives the corresponding row of Table 6.1.

Properties and table of values

Formula (6.2.43) shows that for a specific conjugation tuple $\mathbf{c}_l$, in addition to the number of sensors D the number $M_{\mathrm{u},D}^{\mathbf{c}_l}$ of unique elements in $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ is determined completely by the number $n_{\mathbf{c}_l}$ of conjugations and by the tuple length l. Hence, the order of conjugations '$*$' and no conjugations '$\circ$' in the tuple is not important. Let the complement of a conjugation tuple $\mathbf{c}_l$ be denoted by $\bar{\mathbf{c}}_l$, i.e. $\bar{\mathbf{c}}_l \triangleq (\bar{c}_1, \ldots, \bar{c}_l)$ with $\bar{*} = \circ$ and $\bar{\circ} = *$. Then, it is clear that the number of unconjugated elements in $\mathbf{c}_l$ equals $n_{\bar{\mathbf{c}}_l} = l - n_{\mathbf{c}_l}$, and also that $n_{\mathbf{c}_l} = l - n_{\bar{\mathbf{c}}_l}$. Hence, (6.2.43) also shows that for a specific conjugation tuple $\mathbf{c}_l$, in addition to D the number $M_{\mathrm{u},D}^{\mathbf{c}_l}$ is determined completely by the number $n_{\bar{\mathbf{c}}_l}$ of unconjugated elements and by the tuple length l. Formulated differently, $M_{\mathrm{u},D}^{\mathbf{c}_l}$ is determined completely by D, $n_{\mathbf{c}_l}$, and $n_{\bar{\mathbf{c}}_l}$ because $n_{\mathbf{c}_l} + n_{\bar{\mathbf{c}}_l} = l$. Evidently, $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ has the same number $M_{\mathrm{u},D}^{\mathbf{c}_l} = M_{\mathrm{u},D}^{\bar{\mathbf{c}}_l}$ of unique product terms as $\check{\mathbf{z}}_D^{\bar{\mathbf{c}}_l}$. Note that the terms of the latter vector are the conjugates of the corresponding ones in $\check{\mathbf{z}}_D^{\mathbf{c}_l}$. These observations inspire us to define the following notations equivalent to $M_{\mathrm{u},D}^{\mathbf{c}_l}$ in (6.2.43), from which the properties that we have just discussed follow immediately:

$$\boxed{M_{\mathrm{u},D}^{\mathbf{c}_l} = M_{\mathrm{u},D}^{\bar{\mathbf{c}}_l} = M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)} = M_{\mathrm{u},D}^{\langle l,n_{\bar{\mathbf{c}}_l}\rangle} = \left\langle {D \atop n_{\mathbf{c}_l}} \right\rangle \left\langle {D \atop n_{\bar{\mathbf{c}}_l}} \right\rangle .} \tag{6.2.44}$$

Together with the fact that both $n_{\mathbf{c}_l}$ and $n_{\bar{\mathbf{c}}_l}$ are elements of the set $\{0, 1, \ldots, l\}$ with cardinality $l+1$, equation (6.2.44) implies that $M_{\mathrm{u},D}^{\mathbf{c}_l} = M_{\mathrm{u},D}^{\bar{\mathbf{c}}_l} = M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)} = M_{\mathrm{u},D}^{\langle l,n_{\bar{\mathbf{c}}_l}\rangle}$ considered

as a function of $n_{\mathbf{c}_l}$ or $n_{\bar{\mathbf{c}}_l}$ is symmetric around $\frac{l+1}{2}$ for fixed D and l. Using the notations $M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)}$ and $M_{\mathrm{u},D}^{\langle l,n_{\bar{\mathbf{c}}_l}\rangle}$ with explicit dependence on $n_{\mathbf{c}_l}$, this follows directly from (6.2.44) by noting that:

$$M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)} = M_{\mathrm{u},D}^{(l-n_{\mathbf{c}_l},l)} \qquad \text{and} \qquad M_{\mathrm{u},D}^{\langle l,n_{\bar{\mathbf{c}}_l}\rangle} = M_{\mathrm{u},D}^{\langle l,l-n_{\bar{\mathbf{c}}_l}\rangle} .$$

Hence, for each fixed D and l we only need to consider the first or last $\lceil \frac{l+1}{2} \rceil$ values of $n_{\mathbf{c}_l}$, i.e. the values $0,\ldots,\lceil \frac{l+1}{2} \rceil - 1$ or $l - \lceil \frac{l+1}{2} \rceil + 1,\ldots,l$ respectively, because the other values are then determined completely. Furthermore, from (6.2.41)-(6.2.44), it can be proven that for each fixed combination of D and l, the maximum possible value of the multiset coefficient in (6.2.44) is attained for $n_{\mathbf{c}_l} = l/2$ if l is even, and for $n_{\mathbf{c}_l} = (l-1)/2$ or $n_{\mathbf{c}_l} = (l+1)/2$ if l is odd; this can also be seen from Table 6.2. Hence, using for example the notation $M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)}$ with explicit dependence on $n_{\mathbf{c}_l}$, for even l we have:

$$\max_{n_{\mathbf{c}_l}} M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)} = M_{\mathrm{u},D}^{(l/2,l)} = \left(\left\langle \begin{matrix} D \\ l/2 \end{matrix} \right\rangle \right)^2 = \binom{l/2+(D-1)}{l/2}^2 , \tag{6.2.45}$$

whereas for odd l:

$$\begin{aligned} \max_{n_{\mathbf{c}_l}} M_{\mathrm{u},D}^{(n_{\mathbf{c}_l},l)} &= M_{\mathrm{u},D}^{((l-1)/2,l)} = M_{\mathrm{u},D}^{((l+1)/2,l)} \triangleq \left\langle \begin{matrix} D \\ (l+1)/2 \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ (l-1)/2 \end{matrix} \right\rangle \\ &= \binom{(l+1)/2+(D-1)}{(l+1)/2} \binom{(l-1)/2+(D-1)}{(l-1)/2} , \end{aligned} \tag{6.2.46}$$

Evaluating these functions for $1 \leq D \leq 5$ for each fixed l yields the last row for that value of l given in Table 6.2. For example, the last row of the table can be computed by evaluating (6.2.45) with $l = 4$ and $1 \leq D \leq 5$. These *maximum possible values of $M_{\mathrm{u},D}^{\mathbf{c}_l}$ are important for constructing a subspace matrix in such a way that a maximum number of mixing matrix columns can be identified.*

Examples

We will now provide some examples illustrating the use and correctness of (6.2.41)-(6.2.44). As a first example, we consider the l-th order Kronecker product vector $\check{\mathbf{z}}_D^{(\circ)_l}$ containing products without conjugations, which is defined in (6.2.23) of the previous section. Substituting $n_{\mathbf{c}_l} = n_{(\circ)_l} = 0$ into one of the equations above yields:

$$M_{\mathrm{u},D}^{(\circ)_l} \triangleq M_{\mathrm{u},D}^{l} = \left\langle \begin{matrix} D \\ l \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ 0 \end{matrix} \right\rangle = M_{\mathrm{u},D}^{l-0} \cdot M_{\mathrm{u},D}^{0} = \binom{l+D-1}{l} , \tag{6.2.47}$$

which agrees with the result given by (6.2.33). For example, the length of $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ}$ (5.3.33) with $D = 2$ and $l = 2$ equals 3, the length of $\check{\mathbf{z}}_{\mathrm{u},3}^{\circ\circ}$ (5.3.34) with $D = 3$ and $l = 2$ equals 6, and the length of $\check{\mathbf{z}}_{\mathrm{u},2}^{(\circ)_3}$ (6.2.27) with $D = 2$ and $l = 3$ equals 4.

Now consider the example in (6.2.35), where $D = 2$, $l = 3$, and the conjugation tuple is given by $\mathbf{c}_3 \triangleq (\circ,\circ,*)$ with $n_{\circ\circ *} = 1$. From (6.2.44) or Table 6.2 we see that $M_{\mathrm{u},2}^{\circ\circ *} = M_{\mathrm{u},2}^{(1,3)} = M_{\mathrm{u},D}^{\langle 3,2\rangle} = \left\langle \begin{matrix} 2 \\ 1 \end{matrix} \right\rangle \left\langle \begin{matrix} 2 \\ 2 \end{matrix} \right\rangle = 6$, which is precisely the number of unique product terms in $\check{\mathbf{z}}_2^{\circ\circ *}$, i.e. the number of elements in $\check{\mathbf{z}}_{\mathrm{u},2}^{\circ\circ *}$ in (6.2.35).

6.2.3.3 Other symmetries

In this section, we briefly note some symmetries that are not relevant for determining the number of unique sensor cumulant functions, but nevertheless are useful on other occasions. Firstly, in general the product terms in $\check{\mathbf{z}}_D^{\mathbf{c}_l}$ and $\check{\mathbf{z}}_D^{\bar{\mathbf{c}}_l}$ satisfy the following symmetry:

$$\begin{aligned} &\left((z_{i_1})^{c_1} \cdots (z_{i_l})^{c_l}\right)^* = (z_{i_1})^{\bar{c}_1} \cdots (z_{i_l})^{\bar{c}_l} \qquad \forall\, 1 \leq i_1, \ldots, i_l \leq D \\ \equiv \quad &\left(\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}\right)^* = \check{z}_{\mathbf{i}_l}^{\bar{\mathbf{c}}_l} \qquad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}\,, \end{aligned} \tag{6.2.48}$$

which is a generalization of (6.2.30).

The next two symmetry properties pertain to the source auto-cumulant functions and follow directly from the definition of the cumulant function. The first one is given by:

$$\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] = \kappa_j^{s,\sigma(\mathbf{c}_l)}[\sigma(\mathbf{n}_l)] \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \qquad \forall\, 1 \leq j \leq S, \qquad \forall \sigma \in \Sigma_{\mathbf{c}_l}\,,$$

where $\Sigma_{\mathbf{c}_l}$ is the set of all different conjugation tuple permutation functions defined on $\mathcal{C}_l$, and the second by:

$$\left(\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l]\right)^* = \kappa_j^{s,\bar{\mathbf{c}}_l}[\mathbf{n}_l] \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \qquad \forall\, 1 \leq j \leq S\,.$$

6.2.3.4 Maximum rank of l-th order Khatri-Rao product with conjugation tuple $\mathbf{c}_l$

As we have made clear in the previous chapters, and will see in this chapter as well, the (maximum) rank of the involved Khatri-Rao product of the mixing matrix $\mathbf{A}$ is of paramount importance for determining the number of linearly independent equations in the system of homogeneous polyconjugal equations that will be derived in the coming sections, and thus also for determining the maximum number $S_{\max,D}^{\mathbf{c}_l}$ of columns and/or sources that can be identified; see also Section 6.6. The l-th order Khatri-Rao product $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l} = \mathbf{A}_{D,\diamond}^{c_1\cdots c_l}$ of $\mathbf{A}$ with conjugation tuple $\mathbf{c}_l = (c_1, \ldots, c_l)$ is defined as the matrix formed by stacking the column vectors $\check{\mathbf{a}}_D^{1,\mathbf{c}_l}, \ldots, \check{\mathbf{a}}_D^{S,\mathbf{c}_l}$ defined in (6.2.21) alongside each other as follows (see also Sections 5.3.3.4 and A.9, in particular Equations (A.9.3) and (A.9.4)):

$$\mathbf{A}_{D,\diamond}^{\mathbf{c}_l} \triangleq (\mathbf{A})^{c_1} \diamond \cdots \diamond (\mathbf{A})^{c_l} = \left[\left(\mathbf{a}^1\right)^{c_1} \otimes \cdots \otimes \left(\mathbf{a}^1\right)^{c_l} \;\; \cdots \;\; \left(\mathbf{a}^S\right)^{c_1} \otimes \cdots \otimes \left(\mathbf{a}^S\right)^{c_l}\right], \tag{6.2.49}$$

which can be written in expanded form as follows:

$$\mathbf{A}_{D,\diamond}^{\mathbf{c}_l} = \begin{bmatrix} \check{a}_{1\cdots1}^{1,c_1\cdots c_l} & & \check{a}_{1\cdots1}^{S,c_1\cdots c_l} \\ \vdots & \cdots & \vdots \\ \check{a}_{1\cdots D}^{1,c_1\cdots c_l} & & \check{a}_{1\cdots D}^{S,c_1\cdots c_l} \\ \vdots & \cdots & \vdots \\ \check{a}_{D\cdots1}^{1,c_1\cdots c_l} & & \check{a}_{D\cdots1}^{S,c_1\cdots c_l} \\ \vdots & \cdots & \vdots \\ \check{a}_{D\cdots D}^{1,c_1\cdots c_l} & & \check{a}_{D\cdots D}^{S,c_1\cdots c_l} \end{bmatrix} = \begin{bmatrix} \left(a_1^1\right)^{c_1} \cdots \left(a_1^1\right)^{c_l} & \cdots & \left(a_1^S\right)^{c_1} \cdots \left(a_1^S\right)^{c_l} \\ \vdots & \ddots & \vdots \\ \left(a_1^1\right)^{c_1} \cdots \left(a_D^1\right)^{c_l} & \cdots & \left(a_1^S\right)^{c_1} \cdots \left(a_D^S\right)^{c_l} \\ \vdots & \ddots & \vdots \\ \left(a_D^1\right)^{c_1} \cdots \left(a_1^1\right)^{c_l} & \cdots & \left(a_D^S\right)^{c_1} \cdots \left(a_1^S\right)^{c_l} \\ \vdots & \ddots & \vdots \\ \left(a_D^1\right)^{c_1} \cdots \left(a_D^1\right)^{c_l} & \cdots & \left(a_D^S\right)^{c_1} \cdots \left(a_D^S\right)^{c_l} \end{bmatrix}. \tag{6.2.50}$$

Following the same reasoning as in Section 5.3.3.4, it follows that the maximum possible rank of $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ is bounded as follows:

$$\mathrm{rank}(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}) \leq \min\left(M_{\mathrm{u},D}^{\mathbf{c}_l}, S\right), \tag{6.2.51}$$

where $M_{\mathrm{u},D}^{\mathbf{c}_l}$ is given by (6.2.41)-(6.2.44). Hence, for a randomly generated matrix $\mathbf{A}$ and without limitations on the number of columns the maximum possible rank of (a matrix with the same structure as) $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ for fixed D, viz. the maximum row rank of $\mathbf{A}_{\diamond}^{\mathbf{c}_l}$, equals:

$$\boxed{\max\left(\mathrm{rank}_{\mathrm{row}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right) = M_{\mathrm{u},D}^{\mathbf{c}_l} = \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\bar{\mathbf{c}}_l} \end{matrix} \right\rangle .} \tag{6.2.52}$$

In general, if the matrix $\mathbf{A}$ from which the l-th order Khatri-Rao product is 'sufficiently random' this maximum possible rank is achieved. See Table 6.2 on page 314 for the maximum achievable row rank of $\mathbf{A}_{\diamond}^{\mathbf{c}_l}$ for several combinations of D, l and $\mathbf{c}_l$.

6.2.4 Derivation of system of D-variate l-homogeneous polyconjugal equations; functional notation

Now, finally we are in a position to derive a system of D-variate homogeneous polyconjugal equations of degree l satisfied by the columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of the mixing matrix $\mathbf{A} \in \mathbb{C}_D^S$ for the most general scenario along the same lines as in Sections 4.3.4, 4.3.5 and 4.3.6 of Chapter 4, and Sections 5.3.4, 5.3.5 and 5.3.6 of Chapter 5. In this section, similarly to Sections 4.3.4 and 5.3.4 we start by developing the theory in functional notation.

6.2.4.1 Dimensions of signal and source subspaces

Again, we start our derivation by considering the dimension $d_D^{x,\mathbf{c}_l}$ of the linear space spanned by the sensor cumulant functions:

$$d_D^{x,\mathbf{c}_l} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}\right)\right) . \tag{6.2.53}$$

As in Chapters 4 and 5, in analogy with subspace method jargon the linear space $\mathcal{L}\big(\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}\big) = \mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)$ is called *signal subspace*. We also refer to this subspace as *sensor subspace*. Let $d_S^{s,\mathbf{c}_l}$ denote the dimension of the linear space spanned by the source auto-cumulant functions:

$$d_S^{s,\mathbf{c}_l} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,\mathbf{c}_l}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{1\leq j\leq S}\right)\right) . \tag{6.2.54}$$

We will refer to the linear space $\mathcal{L}\big(\mathcal{K}_S^{s,\mathbf{c}_l}\big)$ spanned by the source auto-cumulant functions as *source subspace*. From *AS2* on page 299, i.e. from the fact that the source auto-cumulant functions are linearly independent, it follows directly that the dimension of the source subspace equals:

$$d_S^{s,\mathbf{c}_l} \triangleq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,\mathbf{c}_l}\big)\right) = S\,. \tag{6.2.55}$$

Equations (6.2.3) and (6.2.16) state that each sensor cumulant function $\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ is a linear combination of the S source auto-cumulant functions $\kappa_1^{s,\mathbf{c}_l}[\mathbf{n}_l], \ldots, \kappa_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ with coefficients $\check{a}_{\mathbf{i}_l}^{1,\mathbf{c}_l}, \ldots, \check{a}_{\mathbf{i}_l}^{S,\mathbf{c}_l}$ respectively. Hence, using linear algebra [115, 150] the following 'dimensional relation' between $d_D^{x,\mathbf{c}_l}$ and $d_S^{s,\mathbf{c}_l}$ can be deduced:

$$d_D^{x,\mathbf{c}_l} = \dim\left(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) \leq \dim\left(\mathcal{L}\big(\mathcal{K}_S^{s,\mathbf{c}_l}\big)\right) = d_S^{s,\mathbf{c}_l} = S\,. \tag{6.2.56}$$

This is a logical and intuitive consequence of the fact that all sensor cumulant functions are linear combinations of the same S source auto-cumulant functions. In principle, depending on the values of the mixing matrix elements, $d_D^{x,\mathbf{c}_l}$ can be smaller than $d_S^{s,\mathbf{c}_l}$. However, in general $d_D^{x,\mathbf{c}_l}$ will be equal to $d_S^{s,\mathbf{c}_l}$ for a randomly generated mixing matrix $\mathbf{A}$. In the sequel, we will show how $d_D^{x,\mathbf{c}_l}$ depends on $\mathbf{A}$ and $d_S^{s,\mathbf{c}_l}$.

6.2.4.2 Linear dependence of sensor cumulant functions

Now we will present an essential part of our derivation that is based on a trivial linear algebra result that we have already stated in Section 5.3.4.2, viz. that in an arbitrary linear vector space of a certain dimension d any set containing more than d vectors is linearly dependent. From this property and (6.2.56) it follows that:

$$\dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big) \leq \min\Big(\big|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big|,\ S\Big) = \min\Big(M_{\mathrm{u},D}^{x,\mathbf{c}_l},\ S\Big)\ .$$

As we will see in the Section 6.2.6, this inequality also follows from (6.2.51) because $\dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big)$ is smaller than or equal to $\mathrm{rank}(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l})$. Furthermore, if the number $M_{\mathrm{u},D}^{x,\mathbf{c}_l} = \big|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big|$ of unique sensor cumulant functions given by (6.2.41)-(6.2.43) is larger than the dimension $d_D^{x,\mathbf{c}_l}$ of the signal subspace spanned by the functions in $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, i.e. if:

$$\big|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big| > \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,\mathbf{c}_l} > d_D^{x,\mathbf{c}_l}\ , \tag{6.2.57}$$

then *the sensor cumulant functions in $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ are linearly dependent*. The condition in (6.2.57) can be fulfilled by choosing the right number D of sensors in relation to the number S of sources, where in general 'right' depends on the employed order of the statistics and on the chosen conjugation pattern.

Equivalently to (5.3.65), if the sensor cumulant functions in $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ are linearly dependent, i.e. if (6.2.57) is satisfied, then by the definition of linear dependence there exist non-zero and non-unique sets of coefficients $\big\{\varphi_1^{\mathbf{i}_l}\big\}^{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$, ..., $\big\{\varphi_{Q_{D,S}^{\mathbf{c}_l}}^{\mathbf{i}_l}\big\}^{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$, indexed by an arbitrarily integer-valued index q, such that:

$$\sum_{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l}\ \kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] = 0[\mathbf{n}_l] \qquad \forall\ \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l},\quad \forall\ q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}\,, \tag{6.2.58}$$

where the set $\mathcal{Q}_{D,S}^{\mathbf{c}_l}$ is defined similarly to (4.3.23) and (5.3.66) as follows:

$$\mathcal{Q}_{D,S}^{\mathbf{c}_l} \triangleq \big\{1,\ldots,Q_{D,S}^{\mathbf{c}_l}\big\} \tag{6.2.59}$$

with $Q_{D,S}^{\mathbf{c}_l} \triangleq \big|\mathcal{Q}_{D,S}^{\mathbf{c}_l}\big|$ the maximum number of linearly independent equations. This number equals the dimension of the orthogonal complement of the signal subspace:

$$\boxed{Q_{D,S}^{\mathbf{c}_l} = \dim\Big(\big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\big)^{\perp}\Big) = \big|\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big| - \dim\Big(\mathcal{L}\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big) = M_{\mathrm{u},D}^{x,\mathbf{c}_l} - d_D^{x,\mathbf{c}_l}\,.} \tag{6.2.60}$$

Hence, $Q_{D,S}^{\mathbf{c}_l}$ equals the difference between the left and right hand sides of each of the equations in (6.2.57). Note the similarity to (4.3.24), (4.3.54) and (4.3.92) of Chapter 4, and (5.3.67), (5.3.95) and (5.3.127) of Chapter 6. In Section 6.2.8.3 we will show that a valid set of coefficients of the equations in (6.2.58) can be deduced from the Singular Value Decomposition (SVD) of a function-valued vector associated with the set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, or equivalently from the SVD of a properly defined subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. Thus, we can consider the coefficients of the equations as (approximately) known quantities. Similarly to (4.3.26) and (5.3.68), the set containing the different sets of coefficients $\big\{\varphi_q^{\mathbf{i}_l}\big\}^{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$ for each value of $q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}$ is denoted by $\Phi_{D,S}^{\mathbf{c}_l}$, i.e.:

$$\Phi_{D,S}^{\mathbf{c}_l} \triangleq \Big\{\big\{\varphi_q^{\mathbf{i}_l}\big\}^{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}\Big\}_{q\in\mathcal{Q}_{D,S}^{\mathbf{c}_l}}\ . \tag{6.2.61}$$

Note that all linear combinations of the sets in $\Phi^{\mathbf{c}_l}_{D,S}$ also satisfy (6.2.58). As in Sections 4.3.4.2 and 5.3.4.2, it should be noted again that completely similar and equivalent equations and results are obtained if we replace $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ by $\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{t},D}$ in the discussion above. Since it is trivial to work out this case along the lines of the remarks made in the paragraph containing (4.3.27), we will not discuss this approach here.

6.2.4.3 Deriving the system of equations by exploiting linear independence of source auto-cumulant functions

The next step of our derivation amounts to combining (6.2.16) and (6.2.58), and exploiting assumption *AS2* on page 299 once again; this directly leads us to the desired system of equations. Firstly, (6.2.58) is expressed in the source auto-cumulant functions by substituting (6.2.16):

$$\begin{aligned} 7 \sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \, \kappa^{x,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l] &= \sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \left(\sum_{j=1}^{S} \check{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l} \kappa^{s,\mathbf{c}_l}_j[\mathbf{n}_l] \right) = \sum_{j=1}^{S} \left(\sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \check{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l} \right) \kappa^{s,\mathbf{c}_l}_j[\mathbf{n}_l] \\ &= 0[\mathbf{n}_l], \qquad \forall\, \mathbf{n}_l \in \Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}, \quad \forall\, q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}\,. \end{aligned} \tag{6.2.62}$$

Now the essence of assumption *AS2*, viz. the linear independence of the source auto-cumulant functions, comes into play. Applying *AS2* to (6.2.62) by identifying:

$$\xi^j_q = \sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \, \check{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l} \qquad \forall\, q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}\,, \quad \forall\, 1 \le j \le S\,,$$

it follows that:

$$\sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \, \check{a}^{j,\mathbf{c}_l}_{\mathbf{i}_l} = 0 \qquad \forall\, q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}\,, \quad \forall\, 1 \le j \le S\,. \tag{6.2.63}$$

This system of equations describes the relation between the *unknown* coefficients of the mixing matrix $\mathbf{A}$ and the *known* set of coefficient sets $\Phi^{\mathbf{c}_l}_{D,S}$ (6.2.61). Let $\mathbf{z} \in \mathbb{C}_D$ be a vector of variables having the same size as a column of $\mathbf{A}$, and define the functions $\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z})\}_{q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ as follows:

$$\boxed{f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_q \, \check{z}^{\mathbf{c}_l}_{\mathbf{i}_l} \qquad \forall\, \mathbf{z} \in \mathbb{C}_D\,, \quad \forall\, q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}\,.} \tag{6.2.64}$$

Then, (6.2.63) states that all columns $\mathbf{a}^1, \ldots, \mathbf{a}^S$ of $\mathbf{A}$ satisfy:

$$f^{\mathbf{c}_l}_{D,q}(\mathbf{a}^j) = 0 \qquad \forall\, q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}\,, \qquad 1 \le j \le S\,. \tag{6.2.65}$$

Hence, at this point *the MIBI problem has been projected onto the problem of solving the following system of equations for the columns of the mixing matrix* $\mathbf{A}$:

$$\boxed{\left\{ f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}}\,.} \tag{6.2.66}$$

As in the previous two chapters, by 'projected' we mean that the system of equations follows from our MIBI problem definition and formulation, but not necessarily vice versa.

6.2.4.4 Properties and structure of functions and system

All functions in system (6.2.66) have the same specific form. Firstly, from the definition in (6.2.64) it is clear that each function $f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) = f_{D,q}^{\mathbf{c}_l}(z_1, \ldots, z_D)$ is a D-variate 'polynomial-like' function containing product terms of degree l. Strictly speaking, $f_{D,q}^{\mathbf{c}_l}(\mathbf{z})$ is only a polynomial in $z_1, \ldots, z_D$ if $\mathbf{c}_l = (\circ)_l$ and a polynomial in $(z_1)^*, \ldots, (z_D)^*$ if $\mathbf{c}_l = (*)_l$. Similarly to Section 5.3.4.4, in order to clearly make a distinction between a polynomial in certain variables on the one hand, and a polynomial-like function in the same variables as well as their conjugates on the other, we call the latter type of function *'polyconjugal'*. Hence, from (6.2.64) it follows that each function $f_{D,q}^{\mathbf{c}_l}(\mathbf{z})$ in (6.2.66) is a *D-variate polyconjugal of degree l, which is homogeneous of degree l with conjugation tuple $\mathbf{c}_l$*, also called *l-homogeneous with conjugation tuple $\mathbf{c}_l$*, meaning that (see also (A.2.8)):

$$\boxed{f_{D,q}^{\mathbf{c}_l}(\eta \mathbf{z}) = \breve{\eta}^{\mathbf{c}_l} f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) \qquad \forall\, \eta \in \mathbb{C}, \qquad \forall\, \mathbf{z} \in \mathbb{C}_D\,.} \tag{6.2.67}$$

Using (6.2.64) this property can easily be proven as follows:

$$f_{D,q}^{\mathbf{c}_l}(\eta \mathbf{z}) = f_{D,q}^{c_1 \cdots c_l}(\eta z_1, \ldots, \eta z_D) = \sum_{(i_1,\ldots,i_l) \in \mathcal{I}_{\mathrm{u},D}^{c_1 \cdots c_l}} \varphi_q^{i_1 \cdots i_l}\, (\eta z_{i_1})^{c_1} \cdots (\eta z_{i_l})^{c_l}$$

$$= \sum_{(i_1,\ldots,i_l) \in \mathcal{I}_{\mathrm{u},D}^{c_1 \cdots c_l}} (\eta)^{c_1} \cdots (\eta)^{c_l}\, \varphi_q^{i_1 \cdots i_l}\, (z_{i_1})^{c_1} \cdots (z_{i_l})^{c_l} = \breve{\eta}^{c_1 \cdots c_l} f_{D,q}^{c_1 \cdots c_l}(z_1, \ldots, z_D)\,. \quad \square$$

Since each function $f_{D,q}^{\mathbf{c}_l}(\mathbf{z})$ in system (6.2.66) possesses the type of homogeneity defined by (6.2.67), it has the following nice property:

$$\boxed{f_{D,q}^{\mathbf{c}_l}(\mathbf{v}) = 0 \qquad \Longrightarrow \qquad f_{D,q}^{\mathbf{c}_l}(\eta \mathbf{v}) = 0 \qquad \forall\, \eta \in \mathbb{C}\,.} \tag{6.2.68}$$

Hence, if $\mathbf{v}$ is a solution of the system $\{f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) = 0\}_{q \in \mathcal{Q}_{D,S}^{c_1 c_2}}$, then so is $\eta \mathbf{v}$ for all $\eta \in \mathbb{C}$. This is a logical result of the scaling indeterminacy inherent to MIBI (see Section 2.4). Note that combining the homogeneity property of all functions in the system yields:

$$f_{D,q}^{\mathbf{c}_l}(\mathbf{v}) = 0 \quad \forall\, q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l} \qquad \Longrightarrow \qquad \sum_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} \alpha_p^q\, f_{D,q}^{\mathbf{c}_l}(\mathbf{v}) = 0 \quad \forall\, \alpha_p^q \in \mathbb{C}\,, \quad \forall\, p \in \mathbb{N}\,.$$

Hence, applying a full rank linear transformation to system (6.2.66) and solving the resulting system is equivalent to solving (6.2.66) itself. Moreover, a new valid equation is obtained by taking any linear combination of the equations in (6.2.66). This is reminiscent of the remark made just after (6.2.61), viz. that all linear combinations of the sets in $\Phi_{D,S}^{\mathbf{c}_l}$ also give proper sets of coefficients. See the discussion after (4.3.37) on page 147 for more information and a mathematical formulation that can readily be adapted to the current scenario. More properties of the system, and the functions and equations it contains, will be investigated in Section 6.3, where we study several aspects of the algebraic and geometric structure of the problem induced by the system of equations.

6.2.5 Derivation of system of D-variate l-homogeneous polyconjugal equations; row vector notation

Similarly to Sections 4.3.5 and 5.3.5, in this section we briefly repeat the derivation in the previous section in terms of row vectors as an intuitive intermediate step towards the derivation in terms of a subspace matrix, which is discussed in Section 6.2.6. Again, we will see that it suffices to set up a proper one-to-one mapping between the cumulant functions considered in the previous section on the one hand, and corresponding cumulant row vectors on the other; see Fig. 1.11 on page 26. We do this by associating a row vector with each cumulant function defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$. This correspondence will also be used later on for the construction of the subspace matrix.

6.2.5.1 Bijective mapping between cumulant functions and cumulant row vectors

Here, we first generalize the results of Section 5.3.5.1. Suppose that we have specified a Noise-Free ROS $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ by a set of N time tuples as follows:

$$\begin{aligned}\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l} &= \left\{\mathbf{n}_l^1,\ldots,\mathbf{n}_l^N\right\} = \left\{(n_1,\ldots,n_l)^1,\ldots,(n_1,\ldots,n_l)^N\right\}\\ &= \left\{(n_1^1,\ldots,n_l^1),\ldots,(n_1^N,\ldots,n_l^N)\right\},\end{aligned}$$

where $\mathbf{n}_l^q = (n_1,\ldots,n_l)^q = (n_1^q,\ldots,n_l^q)$ is the q-th time tuple of $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$, and $n_1^q,\ldots,n_l^q$ are the first till the l-th elements of $\mathbf{n}_l^q$ respectively. Now we are in a position to define a bijective mapping between a function belonging to $\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$ on the one hand, and a row vector on the other hand, see also Sections 1.4, 4.3.5.1, and 5.3.5.1. For each tuple $\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l$, a row vector $\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}$ is associated with the sensor cumulant function $\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ by defining:

$$\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} \triangleq \left[\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l^1] \;\cdots\; \kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l^N]\right]. \tag{6.2.69}$$

The length of $\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}$ equals the cardinality N of the set $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ (see also (6.1.6)) and thus represents the number of time tuples contained in the Noise-Free ROS. Following the same procedure as described above, for each $1 \leq j \leq S$ a row vector $\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l}$ is associated with the j-th source auto-cumulant function $\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l]$ defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ by defining:

$$\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} \triangleq \left[\kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l^1] \;\cdots\; \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l^N]\right]. \tag{6.2.70}$$

Using these notational conventions, the row vector equivalent of expression (6.2.16) becomes:

$$\boxed{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} = \sum_{j=1}^{S} \check{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\, \tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} \qquad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l\,.} \tag{6.2.71}$$

The various sets of cumulant row vectors are denoted by symbols that are similar to those denoting the corresponding cumulant functions. The set containing all sensor cumulant row vectors associated with (6.2.11) is denoted by:

$$\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,\mathbf{c}_l} \triangleq \left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} \mid \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{\mathbf{c}_l}\right\} = \left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{\mathbf{c}_l}} = \left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}}. \tag{6.2.72}$$

Likewise, the set containing all unique sensor cumulant row vectors associated with (6.2.13) is denoted by:

$$\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l} \triangleq \left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} \mid \mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}\right\} = \left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}. \tag{6.2.73}$$

Finally, the set containing all source auto-cumulant row vectors associated with (6.2.15) is denoted by:

$$\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l} \triangleq \left\{\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} \mid 1 \leq j \leq S\right\} = \left\{\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l}\right\}_{1\leq j\leq S}. \tag{6.2.74}$$

Clearly, the cardinalities of $\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,\mathbf{c}_l}$, $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and $\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l}$ are the same as those of $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$, $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and $\mathcal{K}_S^{s,\mathbf{c}_l}$ respectively, which are given by $M_{\mathrm{t},D}^{x,\mathbf{c}_l}$ in (6.2.12), $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ in (6.2.41)-(6.2.44) and the number of sources S respectively.

6.2.5.2 Dimensions of signal and source subspaces

Now we are in a position to derive the same system of equations as in the previous section in terms of row vectors. Again, we start the derivation by considering the dimension of the signal subspace spanned by the sensor cumulant row vectors, which is expressed in (6.2.53) in functional notation. In row vector notation, this expression becomes:

$$d_D^{x,\mathbf{c}_l} \triangleq \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}\right\}_{\mathbf{i}_l\in\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}\right)\right). \tag{6.2.75}$$

Likewise, the dimension of the source subspace spanned by the source auto-cumulant functions, which is expressed in (6.2.54) in functional notation, can now be expressed in terms of the associated row vectors as follows:

$$d_S^{s,\mathbf{c}_l} \triangleq \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l}\big)\right) = \dim\left(\mathcal{L}\left(\left\{\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l}\right\}_{1\leq j\leq S}\right)\right). \tag{6.2.76}$$

The row vector equivalent of assumption *AS2* on page 299, which states that the source auto-cumulant functions are linearly independent, is given by:

$$\sum_{j=1}^{S}\xi^j\,\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} = \tilde{\mathbf{0}}^N \quad\Longrightarrow\quad \xi^j = 0 \quad \forall\, 1\leq j\leq S\,. \tag{6.2.77}$$

From this property it follows again that $d_S^{s,\mathbf{c}_l} = S$; see (6.2.55). Equation (6.2.71) states that each sensor cumulant row vector $\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}$ is a linear combination of the S source auto-cumulant row vectors $\tilde{\boldsymbol{\kappa}}_1^{s,\mathbf{c}_l},\ldots,\tilde{\boldsymbol{\kappa}}_S^{s,\mathbf{c}_l}$ with coefficients $\check{a}_{\mathbf{i}_l}^{1,\mathbf{c}_l},\ldots,\check{a}_{\mathbf{i}_l}^{S,\mathbf{c}_l}$ respectively. This again implies 'dimensional relation' (6.2.56) between $d_D^{x,\mathbf{c}_l}$ and $d_S^{s,\mathbf{c}_l}$.

6.2.5.3 Linear dependence of sensor cumulant row vectors

Following the same reasoning as in Section 6.2.4.2 it follows that:

$$\dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) \leq \min\left(\big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big|,\ S\right) = \min\left(M_{\mathrm{u},D}^{x,\mathbf{c}_l},\ S\right).$$

Now, if the number $M_{\mathrm{u},D}^{x,\mathbf{c}_l} = \big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big|$ of unique sensor cumulant row vectors given by (6.2.41)-(6.2.43) is larger than the dimension $d_D^{x,\mathbf{c}_l}$ of the signal subspace spanned by the functions in $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, i.e. if:

$$\big|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big| > \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,\mathbf{c}_l} > d_D^{x,\mathbf{c}_l}\,, \tag{6.2.78}$$

then *the sensor cumulant row vectors in $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ are linearly dependent*. Hence, equivalently to (4.3.53) and (5.3.94), if (6.2.78) is satisfied then by the definition of linear dependence

there exist non-zero and non-unique sets of coefficients $\{\varphi_1^{\mathbf{i}_l}\}^{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}, \ldots, \{\varphi_{Q_{D,S}^{\mathbf{c}_l}}^{\mathbf{i}_l}\}^{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$, indexed by an index q, such that:

$$\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l}\, \tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} = \tilde{\mathbf{0}}^N \qquad \forall\, q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}\,, \tag{6.2.79}$$

where the set $\mathcal{Q}_{D,S}^{\mathbf{c}_l}$ is defined in (6.2.59). Note that the number $Q_{D,S}^{\mathbf{c}_l} \triangleq \left|\mathcal{Q}_{D,S}^{\mathbf{c}_l}\right|$ of linearly independent equations is given by (6.2.60) and that it also equals the dimension of the orthogonal complement of the linear (signal sub)space spanned by the sensor cumulant row vectors in $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$:

$$\boxed{Q_{D,S}^{\mathbf{c}_l} = \dim\left(\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)^{\perp}\right)\right) = \left|\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right| - \dim\left(\mathcal{L}\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\right) = M_{\mathrm{u},D}^{x,\mathbf{c}_l} - d_D^{x,\mathbf{c}_l}\,.} \tag{6.2.80}$$

Hence, in the same way as $Q_{D,S}^{\mathbf{c}_l}$ is given by the difference between the left and right hand sides of (6.2.57), it is also given by the difference between the left and right hand sides of (6.2.78).

6.2.5.4 Deriving the system of equations by exploiting linear independence of source auto-cumulant row vectors

The main part of our argument leading to the desired system of equations amounts to combining (6.2.71) and (6.2.79), and exploiting assumption *AS2* in the form of (6.2.77). Hence, expressing (6.2.79) in terms of the source auto-cumulant row vectors by substituting (6.2.71) yields for all $q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}$:

$$\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l}\, \tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} = \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l} \left(\sum_{j=1}^{S} \breve{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\, \tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l}\right) = \sum_{j=1}^{S} \left(\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l} \breve{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\right) \tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} = \tilde{\mathbf{0}}^N\,. \tag{6.2.81}$$

Now exploiting the linear independence property of the source auto-cumulant row vectors, i.e. applying (6.2.77) to the last line in (6.2.81) in the same manner as we did in the previous section again gives system (6.2.63). Hence, from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in the previous section from (6.2.63) on.

6.2.6 Derivation of system of D-variate l-homogeneous polyconjugal equations; matrix-vector notation and subspace matrix

Similarly to Sections 4.3.6 and 5.3.6, in this section we finally present the derivation of the system of homogeneous polyconjugal equations in terms of matrix-vector notation and a 'uniquiefied' subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ along the same lines as Section 5.3.6. In the previous two sections we have shown that the derivation of this system is based on the properties of the signal and noise subspaces. In order to be able to naturally define a proper subspace matrix for our problem formulation and to derive the system of equations from this, in addition to the one-to-one mapping between cumulant functions and their corresponding cumulant row vectors that we used in the previous section, now we also need to set up a proper one-to-one mapping between function-valued vectors on the one hand and matrices on the other hand; see Fig. 1.11 on page 26.

6.2.6.1 Bijective mapping between function-valued vectors and matrices

Similarly to Sections 4.3.6.1 and 5.3.6.1, in this section we associate a matrix with each function-valued vector defined on the Noise-Free ROS $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$, or equivalently with the set containing the elements of the function-valued vector, by means of a bijective mapping. Firstly, with each set of functions defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ we associate a function-valued column vector that is also defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ by simply stacking the 'function elements' of the considered set in a column vector in a certain order. For example, the function-valued column vectors $\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]$ and $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ associated with the sets $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$ (6.2.11) and $\mathcal{K}_S^{s,\mathbf{c}_l}$ (6.2.15) respectively are defined by:

$$\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l] \triangleq \begin{bmatrix} \kappa_{1\cdots1}^{x,\mathbf{c}_l}[\mathbf{n}_l] \\ \vdots \\ \kappa_{D\cdots D}^{x,\mathbf{c}_l}[\mathbf{n}_l] \end{bmatrix} = \begin{bmatrix} \kappa_{(1)_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] \\ \vdots \\ \kappa_{(D)_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] \end{bmatrix} \quad \text{and} \quad \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l] \triangleq \begin{bmatrix} \kappa_1^{s,\mathbf{c}_l}[\mathbf{n}_l] \\ \vdots \\ \kappa_S^{s,\mathbf{c}_l}[\mathbf{n}_l] \end{bmatrix}, \tag{6.2.82}$$

where we have arranged the elements of $\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]$ in 'Kronecker order' and the elements of $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ in ascending index order. Let $\mathbb{C}_P[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$ denote the space of all function-valued column vectors of length P the elements of which are members of $\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$. Then, it is clear that:

$$\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l] \in \mathbb{C}_{M_{\mathrm{t},D}^{x,\mathbf{c}_l}}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}] \quad \text{and} \quad \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l] \in \mathbb{C}_S[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}],$$

where $M_{\mathrm{t},D}^{x,\mathbf{c}_l}$ is defined in (6.2.12). Secondly, following the same reasoning and procedure as in Sections 4.3.6.1 and 5.3.6.1 we associate a matrix with each function-valued column vector, and thus also with the associated set of functions and/or row vectors. For example, at the same time we can associate the three related quantities $\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]$, $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$ and $\tilde{\mathcal{K}}_{\mathrm{t},D}^{x,\mathbf{c}_l}$ with the *subspace matrix* $\mathbf{C}_D^{x,\mathbf{c}_l}$ defined as follows (see also Section 6.2.5.1 and Fig. 1.11):

$$\begin{aligned} \mathbf{C}_D^{x,\mathbf{c}_l} &\triangleq \left[\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l^1] \ \cdots \ \boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l^N]\right] = \begin{bmatrix} \tilde{\boldsymbol{\kappa}}_{(1)_l}^{x,\mathbf{c}_l} \\ \vdots \\ \tilde{\boldsymbol{\kappa}}_{(D)_l}^{x,\mathbf{c}_l} \end{bmatrix} \\ &= \begin{bmatrix} \kappa_{1\cdots1}^{x,c_1\cdots c_l}[n_1^1,\ldots,n_l^1] & \cdots & \kappa_{1\cdots1}^{x,c_1\cdots c_l}[n_1^N,\ldots,n_l^N] \\ \vdots & \ddots & \vdots \\ \kappa_{D\cdots D}^{x,c_1\cdots c_l}[n_1^1,\ldots,n_l^1] & \cdots & \kappa_{D\cdots D}^{x,c_1\cdots c_l}[n_1^N,\ldots,n_l^N] \end{bmatrix} \in \mathbb{C}_{M_{\mathrm{t},D}^{x,\mathbf{c}_l}}^N. \end{aligned} \tag{6.2.83}$$

Using the approach explained in Section 6.2.3 we can construct the corresponding uniquiefied quantities $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$, $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, and $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. As we have explained in the previous sections, the index set indexing the functions of $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ and $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, as well as the rows of $\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, (by definition) is $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$. Suppose that we number the $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ elements of $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ from 1 to $M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ in sequential order, i.e. $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ is specified as follows (see Section 4.3.7):

$$\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l} = \left\{\mathbf{i}_{l,1},\ldots,\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}\right\} = \left\{(i_1,\ldots,i_l)_1,\ldots,(i_1,\ldots,i_l)_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}\right\}. \tag{6.2.84}$$

Then, the uniquiefied subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ (with which we will mainly work in the sequel) can be expressed as follows:

$$\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} \triangleq \big[\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l^1] \;\cdots\; \boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l^N]\big] = \begin{bmatrix} \tilde{\boldsymbol{\kappa}}_{\mathbf{i}_{l,1}}^{x,\mathbf{c}_l} \\ \vdots \\ \tilde{\boldsymbol{\kappa}}_{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}}^{x,\mathbf{c}_l} \end{bmatrix}$$
$$= \begin{bmatrix} \kappa_{(i_1,\ldots,i_l)_1}^{x,c_1\cdots c_l}[n_1^1,\ldots,n_l^1] & \cdots & \kappa_{(i_1,\ldots,i_l)_1}^{x,c_1\cdots c_l}[n_1^N,\ldots,n_l^N] \\ \vdots & \ddots & \vdots \\ \kappa_{(i_1,\ldots,i_l)_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}}^{x,c_1\cdots c_l}[n_1^1,\ldots,n_l^1] & \cdots & \kappa_{(i_1,\ldots,i_l)_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}}^{x,c_1\cdots c_l}[n_1^N,\ldots,n_l^N] \end{bmatrix} \in \mathbb{C}_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}^N . \tag{6.2.85}$$

Finally, we associate the three related quantities $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$, $\mathcal{K}_S^{s,\mathbf{c}_l}$ and $\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l}$ with the source auto-cumulant matrix $\mathbf{C}_S^{s,c_1c_2}$ defined as follows:

$$\mathbf{C}_S^{s,\mathbf{c}_l} \triangleq \big[\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l^1] \;\cdots\; \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l^N]\big] = \begin{bmatrix} \tilde{\boldsymbol{\kappa}}_1^{s,\mathbf{c}_l} \\ \vdots \\ \tilde{\boldsymbol{\kappa}}_S^{s,\mathbf{c}_l} \end{bmatrix} = \begin{bmatrix} \kappa_1^{s,\mathbf{c}_l}[\mathbf{n}_l^1] & \cdots & \kappa_1^{s,\mathbf{c}_l}[\mathbf{n}_l^N] \\ \vdots & \ddots & \vdots \\ \kappa_S^{s,\mathbf{c}_l}[\mathbf{n}_l^1] & \cdots & \kappa_S^{s,\mathbf{c}_l}[\mathbf{n}_l^N] \end{bmatrix} \in \mathbb{C}_S^N . \tag{6.2.86}$$

Note that in definitions (6.2.83), (6.2.85), and (6.2.86) the time tuple index indexes the columns of the involved matrix. Therefore, it is in the superscript position.

The concepts associated with the various bijectively mapped quantities are related bijectively as well. Here, we generalize expressions (5.3.102)-(5.3.106) of Section 5.3.6.1. Suppose that the functions in some set $\mathcal{F} \triangleq \{f_{\mathbf{i}_l}^{\mathbf{c}_l}[\mathbf{n}_l]\}_{\mathbf{i}_l \in \mathcal{I}_f^{\mathbf{c}_l}}$ are defined on some ROS $\boldsymbol{\mathcal{T}}$ with $|\boldsymbol{\mathcal{T}}| \triangleq N$. As before, with these functions we associate the row vectors in the set $\tilde{\mathcal{F}} \triangleq \{\tilde{\mathbf{f}}_{\mathbf{i}_l}^{\mathbf{c}_l}\}_{\mathbf{i}_l \in \mathcal{I}_f^{\mathbf{c}_l}}$. The function-valued column vector $\mathbf{f}[\mathbf{n}_l]$ is defined by stacking the functions on top of each other in a vector. Likewise, the matrix $\mathbf{F}$ associated with $\mathbf{f}[\mathbf{n}_l]$ is defined by stacking the row vectors on top of each other in a matrix; see Fig. 1.11. Now we consider the relations between the fundamental subspaces of $\mathbf{f}[\mathbf{n}_l]$ and $\mathbf{F}$.

Firstly, the linear span $\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[\mathbf{n}_l]\big)$ of the conjugates of the functions in the function-valued column vector argument $\mathbf{f}[\mathbf{n}_l]$ of $\mathcal{L}_{\mathrm{r}}\big(\,\cdot\,\big)$, i.e. the 'linear span in the row direction' of the argument, can be mapped bijectively to the row range of $\mathbf{F}$:

$$\boxed{\mathcal{L}_{\mathrm{r}}\big(\mathbf{f}[\mathbf{n}_l]\big) \triangleq \mathcal{L}\big(\mathcal{F}^*\big) \doteq \mathcal{L}\big(\tilde{\mathcal{F}}^*\big) = \mathcal{R}_{\mathrm{r}}\left(\mathbf{F}\right) .} \tag{6.2.87}$$

where $\mathcal{F}^*$ and $\tilde{\mathcal{F}}^*$ denote the sets containing the conjugated elements of the sets $\mathcal{F}$ and $\tilde{\mathcal{F}}$ respectively.
Secondly, the linear span $\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[\mathbf{n}_l]\big)$ of the column vectors obtained by evaluating the argument $\mathbf{f}[\mathbf{n}_l]$ of $\mathcal{L}_{\mathrm{c}}\big(\,\cdot\,\big)$ at the N time pairs in $\boldsymbol{\mathcal{T}}$, i.e. the 'linear span in the column direction' equals the column range of $\mathbf{F}$:

$$\boxed{\mathcal{L}_{\mathrm{c}}\big(\mathbf{f}[\mathbf{n}_l]\big) \triangleq \mathcal{L}\big(\{\mathbf{f}[\mathbf{n}_l]\}_{\mathbf{n}_l \in \boldsymbol{\mathcal{T}}}\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{F}\right) .} \tag{6.2.88}$$

Thirdly, the right null space of $\mathbf{f}[\mathbf{n}_l]$ can be mapped bijectively to that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[\mathbf{n}_l]\big) \doteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{F}\big) .} \tag{6.2.89}$$

Finally, the left null space of $\mathbf{f}[\mathbf{n}_l]$ equals that of $\mathbf{F}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{F}\big)\,.} \tag{6.2.90}$$

Because of these relations the dimensions of the spaces in the various bijections are equal to each other, i.e.:

$$\boxed{\begin{aligned} &\dim\big(\mathcal{L}_{\mathrm{r}}(\mathbf{f}[\mathbf{n}_l])\big) = \dim\big(\mathcal{L}(\mathcal{F}^*)\big) = \dim\big(\mathcal{L}(\tilde{\mathcal{F}}^*)\big) = \dim\big(\mathcal{R}_{\mathrm{r}}(\mathbf{F})\big) \\ &= \dim\big(\mathcal{L}_{\mathrm{c}}(\mathbf{f}[\mathbf{n}_l])\big) = \dim\big(\mathcal{L}(\{\mathbf{f}[\mathbf{n}_l]\}_{\mathbf{n}_l \in \boldsymbol{\mathcal{T}}})\big) \\ &= \dim\big(\mathcal{R}_{\mathrm{c}}(\mathbf{F})\big) = \mathrm{rank}(\mathbf{F}) = \mathrm{rank}\,\big(\mathbf{f}[\mathbf{n}_l]\big)\,, \end{aligned}} \tag{6.2.91}$$

$$\boxed{\dim\big(\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[\mathbf{n}_l]\big)\big) = \dim\big(\mathcal{N}_{\mathrm{r}}\big(\mathbf{F}\big)\big)\,,} \tag{6.2.92}$$

and:

$$\boxed{\dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[\mathbf{n}_l]\big)\big) = \dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{F}\big)\big)\,.} \tag{6.2.93}$$

6.2.6.2 Expressing sensor cumulant function-valued vector and matrix in mixing matrix elements and source auto-cumulant function-valued vector and matrix

Similarly to Sections 4.3.6.2 and 5.3.6.2, in this section we will express the function-valued sensor cumulant vector in terms of the mixing matrix elements and the function-valued source auto-cumulant vector. Then, we do the same for the associated sensor cumulant matrix, i.e. the subspace matrix. Using (6.2.16) or a derivation that parallels that of (5.3.109) and starts from the basic assumptions we can express the function-valued sensor cumulant vector $\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]$ for all $\mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$ in terms of the elements of the mixing matrix $\mathbf{A}$ and the function-valued source auto-cumulant vector $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ in a very intuitive manner as follows:

$$\begin{aligned} \boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l] &\triangleq \mathrm{cum}\Big(\big(\mathbf{x}[n_1]\big)^{c_1}, \ldots, \big(\mathbf{x}[n_l]\big)^{c_l}\Big) \\ &\overset{(5.1.1)}{=} \mathrm{cum}\left(\left(\sum_{j_1=1}^{S} \mathbf{a}^{j_1} s_{j_1}[n_1] + \boldsymbol{\nu}[n_1]\right)^{c_1}, \ldots, \left(\sum_{j_l=1}^{S} \mathbf{a}^{j_l} s_{j_l}[n_l] + \boldsymbol{\nu}[n_l]\right)^{c_l}\right) \\ &= \sum_{j_1=1}^{S} \cdots \sum_{j_l=1}^{S} \Big[\big(\mathbf{a}^{j_1}\big)^{c_1} \otimes \cdots \otimes \big(\mathbf{a}^{j_l}\big)^{c_l}\Big]\ \mathrm{cum}\Big(\big(s_{j_1}[n_1]\big)^{c_1}, \ldots, \big(s_{j_l}[n_l]\big)^{c_l}\Big) \\ &= \sum_{j=1}^{S} \check{\mathbf{a}}_D^{j,\mathbf{c}_l}\, \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] \;=\; \mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\, \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l] \qquad \forall\, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}\,, \end{aligned} \tag{6.2.94}$$

where the l-th order Kronecker product vector $\check{\mathbf{a}}_D^{j,\mathbf{c}_l}$ is defined in (6.2.21) and the l-th order Khatri-Rao product matrix $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ in (6.2.49) and (6.2.50). Note that $\check{\mathbf{a}}_D^{1,\mathbf{c}_l}, \ldots, \check{\mathbf{a}}_D^{S,\mathbf{c}_l}$ serve as *generalized array response vectors*, whereas $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ serves as the *generalized array response matrix*; see also the previous chapters. Summarizing, we have derived the following function-valued column vector equivalent of (6.2.16) and (6.2.71):

$$\boxed{\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{j=1}^{S} \check{\mathbf{a}}_{\mathrm{u},D}^{j,\mathbf{c}_l}\, \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] = \mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\, \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\,,} \tag{6.2.95}$$

which is defined on $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$. Now using the bijective mappings established in Section 6.2.6.1, this expression can be written in matrix notation as follows:

$$\boxed{\mathbf{C}_D^{x,\mathbf{c}_l} = \mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\, \mathbf{C}_S^{s,\mathbf{c}_l}\,.} \tag{6.2.96}$$

Equations (6.2.95) and (6.2.96) are natural and compact ways of representing the structure of the set of sensor cumulant functions. For each time (index) tuple $\mathbf{n}_l$ the vector $\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]$ is a linear combination of the generalized array response vectors $\check{\mathbf{a}}_D^{1,\mathbf{c}_l}, \ldots, \check{\mathbf{a}}_D^{S,\mathbf{c}_l}$. The matrix $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ containing these vectors defines a linear transformation from $\mathbb{C}_S[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$ to $\mathbb{C}_{M_{\mathrm{t},D}^{x,\mathbf{c}_l}}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$, in particular from the function-valued source auto-cumulant vector for a certain time tuple to a function-valued sensor cumulant vector for the same time tuple. It also defines a linear transformation from $\mathbb{C}_S$ to $\mathbb{C}_{M_{\mathrm{t},D}^{x,\mathbf{c}_l}}$.

Similarly to Section 6.2.6.1, expressions (6.2.95) and (6.2.96) can equally well be expressed in terms of their uniquiefied versions as:

$$\boxed{\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{j=1}^{S} \check{\mathbf{a}}_{\mathrm{u},D}^{j,\mathbf{c}_l}\, \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] = \mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\, \boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]} \tag{6.2.97}$$

and:

$$\boxed{\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\, \mathbf{C}_S^{s,\mathbf{c}_l}} \tag{6.2.98}$$

respectively. Note that $\check{\mathbf{a}}_{\mathrm{u},D}^{j,\mathbf{c}_l}$ is a subvector of the Kronecker product $\check{\mathbf{a}}_D^{j,\mathbf{c}_l}$ that contains only the unique elements of the latter vector that are selected by the index tuples in $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$. Likewise, $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ is a submatrix of the Khatri-Rao product $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$ that contains only the unique rows of the latter matrix that are selected by the index tuples in $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$.

With these concepts in place we can once again derive the system of homogeneous polyconjugal equations in a way that enhances our insight and clearly shows the connections with conventional subspace methods. In fact, we have already done an important part of the job by defining a matrix that can serve naturally as the subspace matrix containing all relevant information in the proper way, viz. the matrix $\mathbf{C}_D^{x,\mathbf{c}_l}$ or $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. In the sequel we will mainly work with $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ for reasons that we have explained earlier, but as before in virtually all equations and results $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ can by replaced by $\mathbf{C}_D^{x,\mathbf{c}_l}$. We leave it to the reader to make this kind of translation if desired.

6.2.6.3 Dimensions of signal and source subspaces

From (6.2.87) it follows that the row range of the subspace matrix, which equals the linear span of the conjugated corresponding sensor cumulant row vectors, is mapped bijectively to the 'linear span in the row direction' of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{L}\big(\big(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)^*\big) \doteqdot \mathcal{L}\big(\big(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)^*\big) = \mathcal{L}_{\mathrm{r}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big)\,. \tag{6.2.99}$$

Likewise, from (6.2.88) it follows that the column range of the subspace matrix equals the linear span in the column direction of the associated function-valued column vector:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{L}\Big(\big\{\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big\}_{\mathbf{n}_l\in\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}}\Big) = \mathcal{L}_{\mathrm{c}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big)\,. \tag{6.2.100}$$

From (6.2.91) it follows that the dimensions of all these spaces are equal to each other:

$$\begin{aligned} d_D^{x,\mathbf{c}_l} &\triangleq \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) = \dim\left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)\right) = \dim\left(\mathcal{L}\left(\left(\tilde{\mathcal{K}}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)^*\right)\right) \\ &= \dim\left(\mathcal{L}\left(\left(\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)^*\right)\right) = \dim\left(\mathcal{L}_{\mathrm{r}}\left(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right)\right) = \dim\left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)\right) \\ &= \dim\left(\mathcal{L}_{\mathrm{c}}\left(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right)\right) = \operatorname{rank}\left(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right). \end{aligned} \tag{6.2.101}$$

Hence, naturally *the dimension of the signal subspace equals the rank of the subspace matrix*.

Similarly to (6.2.99), the row range of the source auto-cumulant matrix $\mathbf{C}_S^{s,\mathbf{c}_l}$ defined in (6.2.86) is mapped bijectively to the linear span of the associated function-valued column vector $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ in the row direction:

$$\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right) = \mathcal{L}\left(\left(\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l}\right)^*\right) \doteqdot \mathcal{L}\left(\left(\mathcal{K}_S^{s,\mathbf{c}_l}\right)^*\right) = \mathcal{L}_{\mathrm{r}}\left(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right). \tag{6.2.102}$$

Likewise, similarly to (6.2.100) the column range of $\mathbf{C}_S^{s,\mathbf{c}_l}$ equals the linear span of $\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ in the column direction:

$$\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right) = \mathcal{L}\left(\left\{\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}}\right) = \mathcal{L}_{\mathrm{c}}\left(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right). \tag{6.2.103}$$

Similarly to (6.2.101) the dimension of the source subspace can be expressed as follows:

$$\begin{aligned} d_S^{s,\mathbf{c}_l} &\triangleq \operatorname{rank}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right) = \dim\left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right)\right) = \dim\left(\mathcal{L}\left(\left(\tilde{\mathcal{K}}_S^{s,\mathbf{c}_l}\right)^*\right)\right) \\ &= \dim\left(\mathcal{L}\left(\left(\mathcal{K}_S^{s,\mathbf{c}_l}\right)^*\right)\right) = \dim\left(\mathcal{L}_{\mathrm{r}}\left(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right)\right) = \dim\left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right)\right) \\ &= \dim\left(\mathcal{L}_{\mathrm{c}}\left(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right)\right) = \operatorname{rank}\left(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\right); \end{aligned} \tag{6.2.104}$$

see also (6.2.54) and (6.2.76). Hence, naturally *the dimension of the source subspace equals the rank of the source auto-cumulant matrix*. The matrix equivalent of assumption *AS2* on page 299, which states that the source auto-cumulant functions and row vectors respectively are linearly independent (see also (6.2.77)), is given by:

$$\tilde{\boldsymbol{\xi}}\mathbf{C}_S^{s,\mathbf{c}_l} = \tilde{\mathbf{0}}^N \qquad \Longrightarrow \qquad \tilde{\boldsymbol{\xi}} = \begin{bmatrix}\xi^1 & \cdots & \xi^S\end{bmatrix} = \begin{bmatrix}0 & \cdots & 0\end{bmatrix}. \tag{6.2.105}$$

Hence, from this property it follows once more that $d_S^{s,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right) = S$, i.e. $\mathbf{C}_S^{s,\mathbf{c}_l} \in \mathbb{C}_S^N$ has full rank.

Now we will show how the signal subspace dimension $d_D^{x,\mathbf{c}_l}$ depends on the mixing matrix and the source signal properties. To start with, from (6.2.98) and linear algebra [115, 150] it follows that:

$$\operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) \leq \min\left(\operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\right), \operatorname{rank}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right)\right), \tag{6.2.106}$$

which again implies relation (6.2.56) between $d_D^{x,\mathbf{c}_l}$ and $d_S^{s,\mathbf{c}_l}$:

$$d_D^{x,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) \leq \operatorname{rank}\left(\mathbf{C}_S^{s,\mathbf{c}_l}\right) = d_S^{s,\mathbf{c}_l} = S. \tag{6.2.107}$$

Similar results are obtained when using (6.2.96) instead of (6.2.98). Equations (6.2.98) and (6.2.106) show that the dimension $d_D^{x,\mathbf{c}_l}$ of the signal subspace is determined by the Khatri-Rao product matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ on the one hand, and the source auto-cumulant matrix $\mathbf{C}_S^{s,\mathbf{c}_l}$ on the other. Now, using the fact that the rank of a matrix is unchanged upon either left or right multiplication by a nonsingular matrix [115] and that $\mathbf{C}_S^{s,\mathbf{c}_l}$ has full rank,

it follows immediately from (6.2.98) that (6.2.107) can be specialized further to:

$$d_D^{x,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\,\mathbf{C}_S^{s,\mathbf{c}_l}\right) = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\right). \tag{6.2.108}$$

Summarizing, $d_D^{x,\mathbf{c}_l}$ equals the rank of the linear transformation defined by $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ in (6.2.50), which in turn equals the dimension of the linear space spanned by the vectors $\check{\mathbf{a}}_D^{1,\mathbf{c}_l}, \ldots, \check{\mathbf{a}}_D^{S,\mathbf{c}_l}$ defined in (6.2.21):

$$\boxed{d_D^{x,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\right).} \tag{6.2.109}$$

Essentially our subspace approach to MIBI is based on the fact that we can compute or estimate the (properties of the) various subspaces of the unknown matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ from the known matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. Now we can understand the importance of studying the rank of the Khatri-Rao product [61, 95, 141, 141, 151, 181] $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ as we did in Section 6.2.3.4. As we have seen in the previous sections, the dimension $d_D^{x,\mathbf{c}_l}$ of the signal subspace is of paramount importance for the derivations in this thesis. Because $d_D^{x,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\right)$ as revealed by (6.2.109), the rank of the Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ is so important.

6.2.6.4 Linear dependence of subspace matrix rows

Following the same reasoning as in Section 6.2.4.2 it follows that:

$$d_D^{x,\mathbf{c}_l} = \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) = \operatorname{rank}\left(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\right) \leq \min\left(M_{\mathrm{u},D}^{\mathbf{c}_l},\ S\right). \tag{6.2.110}$$

Now, if the number $M_{\mathrm{u},D}^{x,\mathbf{c}_l} = \#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\}$ of rows of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ is larger than the dimension $d_D^{x,\mathbf{c}_l}$ of the signal subspace spanned by the row vectors of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ (this is the matrix equivalent of (6.2.57) and (6.2.78)), i.e. if:

$$\#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\} > \operatorname{rank}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right) \qquad \text{or} \qquad M_{\mathrm{u},D}^{x,\mathbf{c}_l} > d_D^{x,\mathbf{c}_l}, \tag{6.2.111}$$

then *the subspace matrix rows are linearly dependent*. Hence, equivalently to (4.3.91) and (5.3.126), if (6.2.111) is satisfied then by the definition of linear dependence there exists a non-zero and non-unique matrix $\mathbf{\Phi} \in \mathbb{C}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}_{Q_{D,S}^{\mathbf{c}_l}}$, which represents the complex conjugate $\left(\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)\right)^*$ of the left null space $\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)$ of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ (see Definition C.2.4), such that:

$$\mathbf{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}}^{N} \quad \equiv \quad \mathbf{\Phi}\,\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}} \quad \forall\ \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}. \tag{6.2.112}$$

The maximum number of linearly independent rows of $\mathbf{\Phi}$ equals the dimension $Q_{D,S}^{\mathbf{c}_l}$ of $\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)$ and is given by the difference between the number of rows of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and its rank [115, 150], which equals the number $Q_{D,S}^{\mathbf{c}_l}$ derived in (6.2.60) and (6.2.80):

$$\boxed{Q_{D,S}^{\mathbf{c}_l} = \dim\left(\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)\right) = \#_{\text{rows}}\{\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\} - \operatorname{rank}(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}) = M_{\mathrm{u},D}^{x,\mathbf{c}_l} - d_D^{x,\mathbf{c}_l}.} \tag{6.2.113}$$

Hence, in the same way as $Q_{D,S}^{\mathbf{c}_l}$ is given by (6.2.60) and (6.2.80) as the differences between the left and right hand sides of (6.2.57) and (6.2.78) respectively, it is also given by the difference between the left and right hand sides of (6.2.111).

Equation (6.2.112) is the matrix-vector equivalent of the equations in (6.2.58) and (6.2.79). In fact, using (6.2.84) to define the row vectors:

$$\tilde{\boldsymbol{\varphi}}_q \triangleq \begin{bmatrix} \varphi_q^{\mathbf{i}_{l,1}} & \cdots & \varphi_q^{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}} \end{bmatrix} \in \mathbb{C}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} \tag{6.2.114}$$

and stacking these vectors in the matrix $\boldsymbol{\Phi}$ as follows:

$$\boldsymbol{\Phi} \triangleq \begin{bmatrix} \tilde{\boldsymbol{\varphi}}_1 \\ \vdots \\ \tilde{\boldsymbol{\varphi}}_{Q_{D,S}^{\mathbf{c}_l}} \end{bmatrix} = \begin{bmatrix} \varphi_1^{\mathbf{i}_{l,1}} & \cdots & \varphi_1^{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}} \\ \vdots & \vdots & \vdots \\ \varphi_{Q_{D,S}^{\mathbf{c}_l}}^{\mathbf{i}_{l,1}} & \cdots & \varphi_{Q_{D,S}^{\mathbf{c}_l}}^{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}} \end{bmatrix} \in \mathbb{C}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}_{Q_{D,S}^{\mathbf{c}_l}}, \tag{6.2.115}$$

this equivalence is clear immediately. For example, the q-th row of $\boldsymbol{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ in (6.2.112) is given by $\tilde{\boldsymbol{\varphi}}_q\,\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, which equals the left hand side of (6.2.79). Clearly, all rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ belong to the conjugate of the left null space of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, i.e.:

$$\tilde{\boldsymbol{\varphi}}_q \in \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big)^* = \Big(\mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big)\Big)^* \qquad \forall\, q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}\,. \tag{6.2.116}$$

In order to obtain a system of equations with as many linearly independent equations as possible, the vectors in the set $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ should be chosen such that they span $\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\Big)^*$ completely. For a matrix $\boldsymbol{\Phi}$ that contains such as set of rows, we write symbolically:

$$\boldsymbol{\Phi}^* \circledcirc \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) \doteqdot \big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\big)^{\perp}. \tag{6.2.117}$$

The indicated isomorphism follows from property *FSP1* on page 450, which states that the left null space of a matrix is isomorphic to the orthogonal complement of its column space.

6.2.6.5 Deriving the system of equations by exploiting nonsingularity of source auto-cumulant matrix

Similarly to the previous sections, the main part of our argument leading to the desired system of polyconjugal equations amounts to combining (6.2.98) and (6.2.112), and exploiting assumption *AS2* on page 299, which implies that the source auto-cumulant matrix $\mathbf{C}_S^{s,\mathbf{c}_l}$ has full rank. Expressing (6.2.112) in terms of $\mathbf{C}_S^{s,\mathbf{c}_l}$ and the uniquiefied Khatri-Rao product matrix $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ by substituting (6.2.98) yields:

$$\boldsymbol{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \boldsymbol{\Phi}\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\,\mathbf{C}_S^{s,\mathbf{c}_l} = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}}^{N}\,. \tag{6.2.118}$$

Because $\mathbf{C}_S^{s,\mathbf{c}_l}$ has full rank due to assumption *AS2*, its pseudo-inverse $\big(\mathbf{C}_S^{s,\mathbf{c}_l}\big)^{\dagger}$ exists. Hence, similarly to Sections 4.3.6.5 and 5.3.6.5 we can multiply both sides (5.3.132) from the right by $\big(\mathbf{C}_S^{s,\mathbf{c}_l}\big)^{\dagger}$ in order to obtain the following system of equations:

$$\boldsymbol{\Phi}\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l} = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}}^{S} \qquad \equiv \qquad \boldsymbol{\Phi}\,\check{\mathbf{a}}_{\mathrm{u},D}^{j,\mathbf{c}_l} = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}} \quad \forall\, 1 \le j \le S\,, \tag{6.2.119}$$

which equals the system in (6.2.63). Hence, again from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in Section 6.2.4 from (6.2.63) on.

6.2.6.6 Concise enlightening derivation of system from subspace matrix

In this section, we present an enlightening derivation in the same spirit as the one given in Sections 4.3.6.6 and 5.3.6.6 of the previous chapters. Firstly, we prove that $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ and $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ have the same column spaces and thus also the same left null spaces. Then we use these properties to obtain the system of homogeneous polyconjugal equations in a very concise manner. We start by showing that the column range of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$, i.e. the signal subspace, equals the column range of the (uniquiefied) l-th order Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$, i.e. the linear span of the (uniquiefied) array response vectors, whenever the source auto-cumulant matrix $\mathbf{C}_S^{s,\mathbf{c}_l}$ has full rank (which is the case in our problem definition). For reference purposes, we summarize this important result explicitly in the following theorem.

Theorem 6.2.4. Column range of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ equals column range of $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$.
If the source cumulant matrix $\mathbf{C}_S^{s,\mathbf{c}_l}$ has full rank, then the column range of the subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ equals the column range of the l-th order Khatri-Rao product $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$ with conjugation tuple $\mathbf{c}_l$ of the mixing matrix $\mathbf{A}$:

$$\boxed{\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{L}_{\mathrm{c}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big)\,.} \tag{6.2.120}$$

Proof. We prove this assertion using (6.2.98). From this equation it follows directly that:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) \subseteq \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big)\,.$$

In addition, multiplying both sides of (6.2.98) from the right by $(\mathbf{C}_S^{s,\mathbf{c}_l})^\dagger$ yields the equality $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l} = \mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}(\mathbf{C}_S^{s,\mathbf{c}_l})^\dagger$, which implies that:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big) \subseteq \mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)\,.$$

These two equations immediately imply (6.2.120). □

Corollary 6.2.5. Left null space of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ equals left null space of $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$.
Under the assumption(s) of Theorem 6.2.4 the left null space of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ equals the left null space of $\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}$:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big)\,.} \tag{6.2.121}$$

Proof. Applying property *FSP1* on page 450 to both sides of (6.2.120) immediately proves the assertion. □

For the scenario considered in Chapter 5 we have shown in Section 5.3.6.6 that the system of homogeneous polyconjugal equations can directly be derived from two facts, viz. on the one hand the property that *the left null space of the unknown matrix containing the generalized array response vectors equals the left null space of the known subspace matrix*, and on the other hand the trivial property that the left null space of any matrix is orthogonal to the column range of the same matrix. Here, we do the same for deriving the system of homogeneous polyconjugal equations for the most general MIBI scenario. The first fact is formulated in Eq. (6.2.121) of Corollary 6.2.5, whereas applying the second to the uniquiefied generalized array response matrix yields:

$$\boxed{\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big) \quad \overset{(6.2.121)}{\Longrightarrow} \quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) \perp \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big)\,.} \tag{6.2.122}$$

Now realizing that the columns of $\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}$, i.e. the generalized array response vectors, form a basis for $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big)$, and defining the matrix $\boldsymbol{\Phi}$ such that its rows form a basis for $\Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big)\Big)^*$, it finally follows that:

$$\boldsymbol{\Phi}\,\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond} = \mathbf{0}^{S}_{Q^{\mathbf{c}_l}_{D,S}}\,,$$

which equals the systems in (6.2.63) and (6.2.119). Hence, once more from this point on the derivation of the system of homogeneous polyconjugal equations is exactly the same as that in Section 6.2.4 from (6.2.63) on.

The few steps outlined in the previous paragraph clearly reflect and constitute the essence of our subspace approach to MIBI. Again, it is important to note that this concise derivation of the system of equations is possible by virtue of the fact that we have imposed a fixed order on the set of sensor cumulant row vectors by stacking them in the chosen order on top of each other in the subspace matrix, and that we could have given similar concise derivations for the functional and row vector formulations in Sections 6.2.4 and 6.2.5 by also imposing a fixed order on the set of sensor cumulant functions and sensor cumulant row vectors respectively, and defining and employing appropriate concepts of left null spaces. We leave these insightful derivations to the reader. See also the remarks at the end of Sections 4.3.4 and 4.3.6.6. Finally we remark that the order itself, i.e. the order in which the cumulant functions and row vectors are stacked on top of each other in the source and sensor function-valued vectors and matrices respectively, is not important because for a different order a completely equivalent system can be derived and completely equivalent conclusions can be drawn. Obviously, if we would choose a different order, the transformation matrix from $\mathbf{C}^{s,\mathbf{c}_l}_{S}$ to $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$, or equivalently from $\boldsymbol{\kappa}^{s,\mathbf{c}_l}_{S}[\mathbf{n}_l]$ to $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$, would be different from the one in (6.2.49) in that its columns and/or rows would be ordered differently. Finally we note that *everywhere in the whole derivation above, we could have used* $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ *instead of* $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ *because of the bijective mapping defined between them.*

6.2.7 Writing the system of equations in matrix-vector notation

We conclude this section by writing the system of equations $\big\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) = 0\big\}_{q\in\mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ given by (6.2.66) in matrix-vector notation. Firstly, using (6.2.114) and (6.2.22) each function $f^{\mathbf{c}_l}_{D,q}(\mathbf{z})$ defined in (6.2.64) can we written as follows:

$$\boxed{f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) \triangleq \sum_{\mathbf{i}_l\in\mathcal{I}^{\mathbf{c}_l}_{\mathrm{u},D}} \varphi^{\mathbf{i}_l}_{q}\,\check{z}^{\mathbf{c}_l}_{\mathbf{i}_l} = \tilde{\boldsymbol{\varphi}}_q\,\check{\mathbf{z}}^{\mathbf{c}_l}_{\mathrm{u},D} = \tilde{\boldsymbol{\varphi}}_q\,\mathbf{w}^{\mathbf{c}_l}_{\mathrm{u},D}(\mathbf{z}) \qquad \forall\,\mathbf{z}\in\mathbb{C}_D\,,\quad \forall\,q\in\mathcal{Q}^{\mathbf{c}_l}_{D,S}.}$$
(6.2.123)

Defining the column vector $\mathbf{f}^{\mathbf{c}_l}_{D,S}(\mathbf{z})$ containing the functions $\big\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z})\big\}_{q\in\mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ by:

$$\mathbf{f}^{\mathbf{c}_l}_{D,S}(\mathbf{z}) \triangleq \begin{bmatrix} f^{\mathbf{c}_l}_{D,1}(\mathbf{z}) \\ \vdots \\ f^{\mathbf{c}_l}_{D,Q^{\mathbf{c}_l}_{D,S}}(\mathbf{z}) \end{bmatrix}, \tag{6.2.124}$$

system (6.2.66) can be written in matrix-vector notation in the following manner:

$$\boxed{\mathbf{f}^{\mathbf{c}_l}_{D,S}(\mathbf{z}) = \boldsymbol{\Phi}\,\check{\mathbf{z}}^{\mathbf{c}_l}_{\mathrm{u},D} = \boldsymbol{\Phi}\,\mathbf{w}^{\mathbf{c}_l}_{\mathrm{u},D}(\mathbf{z}) = \mathbf{0}_{Q^{\mathbf{c}_l}_{D,S}}\,.} \tag{6.2.125}$$

Hence, *the problem of solving the system* $\mathbf{f}^{\mathbf{c}_l}_{D,S}(\mathbf{z}) = \mathbf{0}$ *is equivalent to finding all solutions for* $\mathbf{z}$ *such that the structured vector* $\check{\mathbf{z}}^{\mathbf{c}_l}_{\mathrm{u},D}$*, which is a function* $\mathbf{w}^{\mathbf{c}_l}_{\mathrm{u},D}(\mathbf{z})$ *of* $\mathbf{z}$ *and has the same structure as the columns of* $\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}$*, belongs to the orthogonal complement of the left null space* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big)$ *of* $\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}$ *that can be computed from the known matrix* $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ *because* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big)$ *equals* $\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big)$*. Equivalently,* $\check{\mathbf{z}}^{\mathbf{c}_l}_{\mathrm{u},D}$ *belongs to the column range* $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big)$ *of* $\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}$*.*

6.2.8 Using the SVD to find proper coefficients of the polyconjugals

In this section, we generalize the results of Section 4.3.8 of Chapter 4. In the previous sections and chapters we have projected the MIBI problem onto the problem of solving a system of D-variate l-homogeneous polyconjugal equations for the columns of the mixing matrix $\mathbf{A}$. We also have remarked that the coefficients of the equations in the system can be considered as known because they can be deduced from the sensor cumulant functions by means of the Singular Value Decomposition (SVD). In this section, the procedure for doing this is explained along the lines of Sections 4.3.8 and 5.3.8 and amounts to performing a subspace decomposition that is similar to that performed by conventional subspace methods. Again, we provide both a functional and a matrix formulation of the SVD. In the sequel of the section, both the functional SVD of $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ and the equivalent standard matrix SVD of $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ are used to obtain a proper set $\Phi^{\mathbf{c}_l}_{D,S}$ (6.2.61) containing a set of coefficients for each equation in the system, or equivalently the matrix $\boldsymbol{\Phi}$ (6.2.115) containing a row for each equation. In Section 6.2.6, in particular Sections 6.2.6.4 and 6.2.6.6, we have seen that the rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ of $\Phi^{\mathbf{c}_l}_{D,S}$ belong to the conjugate left null space $\Big(\mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big)\Big)^* = \Big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big)\Big)^*$ of $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ and $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$, which can be determined directly from the SVD of $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ or $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ by choosing and conjugate transposing the left singular vectors that correspond to zero singular values. Hence, in principle it is sufficient to determine this left null space. However, for the sake of insight and because we will also use another fundamental subspace of $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ and $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ later on in this chapter (in Section 6.5, where the MIBI problem is projected onto a multi-matrix generalized eigenvalue problem complementary to the current projection onto the system of homogeneous polyconjugal equations), we will discuss the fully fledged SVD here. We start by formulating the functional SVD and then present the standard matrix SVD.

6.2.8.1 Functional SVD of function-valued sensor cumulant vector $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$

The following theorem presents the functional singular value decomposition[‡] of the function-valued vector $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ understood to be defined on $\Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}$:

Theorem 6.2.6. SVD of function-valued sensor cumulant vector.
Let $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l] \in \mathbb{C}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}[\Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}]$ *be a (uniquefied) function-valued sensor cumulant vector with rank* $d^{x,\mathbf{c}_l}_D \triangleq \operatorname{rank}\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big)^{\ddagger}$*, number of elements* $M^{x,\mathbf{c}_l}_{\mathrm{u},D} \triangleq \big|\mathcal{K}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big|$ *and ROS* $\Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}$ *with cardinality* $N \triangleq \big|\Omega^{s\backslash\nu,\mathbf{c}_l}_{\mathbf{n}_l}\big|$*. Then, there exist real-valued positive constants* $\sigma_p \in \mathbb{R}$ *for* $1 \leq p \leq P$*, column vectors* $\mathbf{u}^p \in \mathbb{C}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$ *for* $1 \leq p \leq M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ *and functions* $v_p[\mathbf{n}_l] \in$

[‡] See Section D.2 for the definitions of the functional singular value decomposition, the rank of a function-valued vector, the relevant inner product functions, etc.

$\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$ *for* $1 \leq p \leq N$*, such that* $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ *can be decomposed as follows:*

$$\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \, (v_p[\mathbf{n}_l])^* \qquad \forall \; \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l} \, ,$$

where:

- $P = \min(M_{\mathrm{u},D}^{x,\mathbf{c}_l}, N)$;
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall \; 1 \leq m, n \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}$;
- $\langle v_m[\mathbf{n}_l], v_n[\mathbf{n}_l] \rangle_{\mathrm{f}} = \delta_{mn} \quad \forall \; 1 \leq m, n \leq N$;
- $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,\mathbf{c}_l}} > 0, \; \sigma_{d_D^{x,\mathbf{c}_l}+1}, \ldots, \sigma_P = 0$.

This theorem is the result of the application of Theorem D.2.1 in Section D.2 to the most general form of the MIBI problem studied in this thesis. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}$, functions $v_1[\mathbf{n}_l], \ldots, v_N[\mathbf{n}_l]$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors*, *right singular functions*, and *singular values* respectively of $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$. The definitions of the employed inner product functions $\langle \cdot, \cdot \rangle_{\mathrm{v}}$ and $\langle \cdot, \cdot \rangle_{\mathrm{f}}$ are given by (D.2.3) and (D.2.4) respectively. For the current scenario these definitions can be given as follows. The first inner product is a function from $\mathbb{C}_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} \times \mathbb{C}_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}$ to $\mathbb{C}$ and is defined by:

$$\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} \triangleq \sum_{p=1}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} u_p^m \, (u_p^n)^* = \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} u_{\mathbf{i}_l}^m \, (u_{\mathbf{i}_l}^n)^* \, , \tag{6.2.126}$$

where the 'v' in the subscript position indicates that this inner product is of the '<u>v</u>ector type'. The second inner product is a function from $\mathbb{C}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}] \times \mathbb{C}[\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}]$ to $\mathbb{C}$ and is defined by:

$$\langle v_m[\mathbf{n}_l], v_n[\mathbf{n}_l] \rangle_{\mathrm{f}} \triangleq \sum_{\mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}} v_m[\mathbf{n}_l] \, (v_n[\mathbf{n}_l])^* \, , \tag{6.2.127}$$

where the 'f' in the subscript position indicates that this inner product is of the '<u>f</u>unctional type'. The theorem can be proven by using the standard SVD theorem [72, 115] described in Section D.1 and the various bijective mappings that we have discussed earlier, i.e. those between functions and row vectors, and function-valued vectors and matrices; see also Fig. 1.11 on page 26.

For convenience and insight the set of vectors and the set of functions in the SVD are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. Since this partitioning is particularly relevant for the subspace techniques used in this thesis, we explain the associated notation in detail. Firstly, we define the set $\mathcal{U}$ containing the left singular vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}$ of the decomposition as follows:

$$\mathcal{U} \triangleq \left\{ \mathbf{u}^1, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} \right\} .$$

This set is split into the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$, where:

$$\mathcal{U}^s \triangleq \left\{ \mathbf{u}^1, \ldots, \mathbf{u}^{d_D^{x,\mathbf{c}_l}} \right\}$$

contains the left singular vectors corresponding to the non-zero singular values, and:

$$\mathcal{U}^\nu \triangleq \left\{ \mathbf{u}^{d_D^{x,\mathbf{c}_l}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} \right\}$$

contains the left singular vectors corresponding to the zero singular values. In subspace decomposition terms the sets $\mathcal{U}^s$ and $\mathcal{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, we define the set $\mathcal{V}$ containing the right singular functions $v_1[\mathbf{n}_l], \ldots, v_N[\mathbf{n}_l]$ of the decomposition as follows:

$$\mathcal{V} \triangleq \left\{ v_1[\mathbf{n}_l], \ldots, v_N[\mathbf{n}_l] \right\}.$$

This set is split into the sets $\mathcal{V}_s$ and $\mathcal{V}_\nu$, where:

$$\mathcal{V}_s \triangleq \left\{ v_1[\mathbf{n}_l], \ldots, v_{d_D^{x,\mathbf{c}_l}}[\mathbf{n}_l] \right\}$$

contains the right singular functions corresponding to the non-zero singular values, and:

$$\mathcal{V}_\nu \triangleq \left\{ v_{d_D^{x,\mathbf{c}_l}+1}[\mathbf{n}_l], \ldots, v_N[\mathbf{n}_l] \right\}$$

contains the right singular functions corresponding to the zero singular values. Now splitting the SVD expression in Theorem 6.2.6 accordingly yields:

$$\begin{aligned} \boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l] &= \sum_{p=1}^{P} \sigma_p \, \mathbf{u}^p \, (v_p[\mathbf{n}_l])^* \\ &= \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \sigma_p \, \mathbf{u}^p \, (v_p[\mathbf{n}_l])^* + \sum_{p=d_D^{x,\mathbf{c}_l}+1}^{P} \sigma_p \, \mathbf{u}^p \, (v_p[\mathbf{n}_l])^* \qquad \forall \, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \end{aligned} \tag{6.2.128}$$

where the first summation contains only vectors and functions from $\mathcal{U}^s$ and $\mathcal{V}_s$ respectively, and the second contains only vectors and functions from $\mathcal{U}^\nu$ and $\mathcal{V}_\nu$ respectively.

Corollary 6.2.7. Reduced SVD of function-valued sensor cumulant vector.
Under the assumptions formulated in Theorem 6.2.6, $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ can be written in the following reduced form:

$$\boxed{\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \sigma_p \, \mathbf{u}^p \, (v_p[\mathbf{n}_l])^* \qquad \forall \, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}.} \tag{6.2.129}$$

Proof. Substituting $\sigma_{d_D^{x,\mathbf{c}_l}+1}, \ldots, \sigma_P = 0$ into (6.2.128) directly yields the Corollary. □

Using the notation introduced in Section 4.3.7 we can now write each sensor cumulant function $\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ in the following natural 'functional SVD-form':

$$\boxed{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \sigma_p \, u_{\mathbf{i}_l}^p \, (v_p[\mathbf{n}_l])^* \qquad \forall \, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \quad \forall \, \mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}.} \tag{6.2.130}$$

Hence, the function $\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ is a linear combination of the right singular functions $(v_1[\mathbf{n}_l])^*$, …, $(v_{d_D^{x,\mathbf{c}_l}}[\mathbf{n}_l])^*$ with coefficients $\sigma_1 \, u_{\mathbf{i}_l}^1, \ldots, \sigma_{d_D^{x,\mathbf{c}_l}} \, u_{\mathbf{i}_l}^{d_D^{x,\mathbf{c}_l}}$. Compare this expression with (6.2.16), which is repeated here for convenience:

$$\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l] = \sum_{j=1}^{S} \breve{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l} \kappa_j^{s,\mathbf{c}_l}[\mathbf{n}_l] \qquad \forall \, \mathbf{n}_l \in \Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}, \quad \forall \, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}. \tag{6.2.131}$$

This equation says that each sensor cumulant function $\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ is also a linear combination of the S source auto-cumulant functions $\kappa_1^{s,\mathbf{c}_l}[\mathbf{n}_l], \ldots, \kappa_S^{s,\mathbf{c}_l}[\mathbf{n}_l]$ with coefficients $\breve{a}_{\mathbf{i}_l}^{1,\mathbf{c}_l}, \ldots,$

$\check{a}^{S,\mathbf{c}_l}_{\mathbf{i}_l}$ respectively. The precise relations between the various (fundamental) subspaces can be formulated as follows (see also (D.2.6)-(D.2.9)):

$$\mathcal{L}_c\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big) = \mathcal{L}\left(\mathcal{U}^s\right); \tag{6.2.132}$$

$$\mathcal{L}_r\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big) = \mathcal{L}\big(\mathcal{V}_s\big) \doteqdot \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big) \subseteq \mathcal{L}_r\big(\boldsymbol{\kappa}^{s,\mathbf{c}_l}_{S}[\mathbf{n}_l]\big) \doteqdot \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}^{s,\mathbf{c}_l}_{S}\big); \tag{6.2.133}$$

$$\mathcal{N}_{\mathrm{r}}\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big) = \mathcal{L}\big(\mathcal{V}_\nu\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big) \supseteq \mathcal{N}_{\mathrm{r}}\big(\boldsymbol{\kappa}^{s,\mathbf{c}_l}_{S}[\mathbf{n}_l]\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}^{s,\mathbf{c}_l}_{S}\big); \tag{6.2.134}$$

$$\mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}^{\mathbf{c}_l}_{\mathrm{u},D,\diamond}\big) = \big(\mathcal{L}\left(\mathcal{U}^\nu\right)\big)^T, \tag{6.2.135}$$

where $\mathcal{L}_r(\cdot)$ denotes the linear span of the conjugates of the functions in argument of the function-valued vector. This concludes our discussion on the functional SVD. See Section 6.2.8.3 for explicit expressions of a proper set $\Phi^{\mathbf{c}_l}_{D,S}$ (6.2.61) containing sets of coefficients for the polyconjugal equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ (6.2.114), or a matrix $\boldsymbol{\Phi}$ (6.2.115) representing these coefficients, in terms of the left null space of $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ found by the SVD.

6.2.8.2 Matrix SVD of subspace matrix $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$

In this section, we present the matrix equivalent of Theorem 6.2.6, which is given by the standard Singular Value Decomposition of the subspace matrix $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$. Because the function-valued vector $\boldsymbol{\kappa}^{x,\mathbf{c}_l}_{\mathrm{u},D}[\mathbf{n}_l]$ and the subspace matrix $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ are related bijectively, their SVD's are related bijectively as well. For details about this relation, we refer the reader to Appendix D. The following theorem presents the SVD of the subspace matrix $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$:

Theorem 6.2.8. SVD of subspace matrix.
Let $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D} \in \mathbb{C}^N_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$ be a (uniquefied sensor cumulant) subspace matrix with rank $d^{x,\mathbf{c}_l}_D \triangleq \mathrm{rank}(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D})$, and with $M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ and N defined as in Theorem 6.2.6. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, column vectors $\mathbf{u}^p \in \mathbb{C}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$ for $1 \leq p \leq M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ and row vectors $\tilde{\mathbf{v}}_p \in \mathbb{C}^N$ for $1 \leq p \leq N$, such that $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ can be decomposed as follows:

$$\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D} = \sum_{p=1}^{P} \sigma_p\, \mathbf{u}^p\, (\tilde{\mathbf{v}}_p)^* \equiv \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^*,$$

where:

- $P = \min(M^{x,\mathbf{c}_l}_{\mathrm{u},D}, N)$;
- $\mathbf{U} \triangleq \begin{bmatrix}\mathbf{u}^1 & \cdots & \mathbf{u}^{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}\end{bmatrix} \in \mathbb{C}^{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$;
- $\langle \mathbf{u}^m, \mathbf{u}^n \rangle_{\mathrm{v}} = \delta^{mn} \quad \forall\, 1 \leq m,n \leq M^{x,\mathbf{c}_l}_{\mathrm{u},D} \quad \equiv \quad \mathbf{U}^H\mathbf{U} = \mathbf{U}\mathbf{U}^H = \mathbf{I}$;
- $\mathbf{V} \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_N\end{bmatrix} \in \mathbb{C}^N_N$;
- $\langle \tilde{\mathbf{v}}_m, \tilde{\mathbf{v}}_n \rangle_{\mathrm{v}} = \delta_{mn} \quad \forall\, 1 \leq m,n \leq N \quad \equiv \quad \mathbf{V}^H\mathbf{V} = \mathbf{V}\mathbf{V}^H = \mathbf{I}$;
- $\boldsymbol{\Sigma} \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_P\} \in \mathbb{R}^N_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$;
- $\sigma_1 \geq \cdots \geq \sigma_{d^{x,\mathbf{c}_l}_D} > 0,\ \sigma_{d^{x,\mathbf{c}_l}_D+1}, \ldots, \sigma_P = 0$.

This theorem is the result of the application of Theorem D.1.1 in Section D.1 to the most general form of the MIBI problem studied in this thesis. As is explained there, the column vectors $\mathbf{u}^1, \ldots, \mathbf{u}^{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}$, row vectors $\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_N$, and constants $\sigma_1, \ldots, \sigma_P$, are called the *left singular vectors*, *right singular vectors*, and *singular values* respectively of $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$. Note that now only vector type inner product functions are used. For a proof we refer the reader to standard textbooks on this topic, e.g. see [72]. The matrices $\mathbf{U}$ and $\mathbf{V}$ are square and unitary, i.e. $\mathbf{U}^H\mathbf{U} = \mathbf{U}\mathbf{U}^H = \mathbf{I}$ and $\mathbf{V}^H\mathbf{V} = \mathbf{V}\mathbf{V}^H = \mathbf{I}$. Although the matrix $\mathbf{\Sigma}$ containing the singular values is not a square matrix, notationally it is written as $\mathrm{diag}\{\sigma_1, \ldots, \sigma_P\}$, and its off-diagonal elements are zero. The singular values $\sigma_1 \geq \cdots \geq \sigma_P$ are written along the main diagonal in descending order, and rows or columns of zeros are appended as necessary to obtain the proper dimensions. Note that only the first $d^{x,\mathbf{c}_l}_D$ singular values are non-zero because $\mathrm{rank}(\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}) = d^{x,\mathbf{c}_l}_D$. See Appendix D for details and examples.

Similarly to the functional SVD in Section 6.2.8.1, for convenience and insight the various components of the matrix SVD of $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$ are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. To start with, let $\mathbf{\Sigma}^s_s$ represent the square diagonal block of $\mathbf{\Sigma}$ that contains these non-zero singular values, i.e. $\mathbf{\Sigma}^s_s$ is the upper left block of $\mathbf{\Sigma}$ of size $d^{x,\mathbf{c}_l}_D \times d^{x,\mathbf{c}_l}_D$. Furthermore, let $\mathbf{\Sigma}^\nu_\nu$ represent the (not necessarily square) block of zeros of $\mathbf{\Sigma}$ to the right and below $\mathbf{\Sigma}^s_s$, i.e. $\mathbf{\Sigma}^\nu_\nu$ is the lower right block of $\mathbf{\Sigma}$ of size $(M^{x,\mathbf{c}_l}_{\mathrm{u},D} - d^{x,\mathbf{c}_l}_D) \times (N - d^{x,\mathbf{c}_l}_D)$. Thus, we have:

$$\mathbf{\Sigma}^s_s \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_{d^{x,\mathbf{c}_l}_D}\} \in \mathbb{R}^{d^{x,\mathbf{c}_l}_D}_{d^{x,\mathbf{c}_l}_D} \qquad \text{and} \qquad \mathbf{\Sigma}^\nu_\nu \triangleq \mathbf{0}^{N-d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}-d^{x,\mathbf{c}_l}_D}, \tag{6.2.136}$$

and $\mathbf{\Sigma}$ can be written as follows:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}^s_s & \mathbf{0}^{N-d^{x,\mathbf{c}_l}_D}_{d^{x,\mathbf{c}_l}_D} \\ \mathbf{0}^{d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}-d^{x,\mathbf{c}_l}_D} & \mathbf{\Sigma}^\nu_\nu \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma}^s_s & \mathbf{0}^{N-d^{x,\mathbf{c}_l}_D}_{d^{x,\mathbf{c}_l}_D} \\ \mathbf{0}^{d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}-d^{x,\mathbf{c}_l}_D} & \mathbf{0}^{N-d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}-d^{x,\mathbf{c}_l}_D} \end{bmatrix}. \tag{6.2.137}$$

The matrix $\mathbf{U}$ is split into two parts as $\mathbf{U} = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix}$. The first part $\mathbf{U}^s$ consists of columns that are multiplied by the non-zero singular values in the SVD-expression of $\mathbf{C}^{x,\mathbf{c}_l}_{\mathrm{u},D}$, whereas the second part $\mathbf{U}^\nu$ consists of columns that are multiplied by the zero singular values. Hence:

$$\mathbf{U}^s \triangleq \begin{bmatrix}\mathbf{u}^1 \cdots \mathbf{u}^{d^{x,\mathbf{c}_l}_D}\end{bmatrix} \in \mathbb{C}^{d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}} \quad \text{and} \quad \mathbf{U}^\nu \triangleq \begin{bmatrix}\mathbf{u}^{d^{x,\mathbf{c}_l}_D+1} \cdots \mathbf{u}^{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}\end{bmatrix} \in \mathbb{C}^{M^{x,\mathbf{c}_l}_{\mathrm{u},D}-d^{x,\mathbf{c}_l}_D}_{M^{x,\mathbf{c}_l}_{\mathrm{u},D}}, \tag{6.2.138}$$

where $\mathbf{u}^p$ is the p-th column of $\mathbf{U}$. In subspace decomposition terms the matrices $\mathbf{U}^s$ and $\mathbf{U}^\nu$ represent the signal and noise subspaces respectively. Likewise, the matrix $\mathbf{V}$ is split as $\mathbf{V} = \begin{bmatrix}\mathbf{V}_s \\ \mathbf{V}_\nu\end{bmatrix}$ with:

$$\mathbf{V}_s \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_{d^{x,\mathbf{c}_l}_D}\end{bmatrix} \in \mathbb{C}^N_{d^{x,\mathbf{c}_l}_D} \quad \text{and} \quad \mathbf{V}_\nu \triangleq \begin{bmatrix}\tilde{\mathbf{v}}_{d^{x,\mathbf{c}_l}_D+1} \\ \vdots \\ \tilde{\mathbf{v}}_N\end{bmatrix} \in \mathbb{C}^N_{N-d^{x,\mathbf{c}_l}_D}, \tag{6.2.139}$$

where $\tilde{\mathbf{v}}_p$ is the p-th row of $\mathbf{V}$. Now splitting the SVD expression in Theorem 6.2.8

accordingly yields:

$$\begin{aligned}\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} &= \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^* = \begin{bmatrix}\mathbf{U}^s & \mathbf{U}^\nu\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_s^s & \mathbf{0}_{d_D^{x,\mathbf{c}_l}}^{N-d_D^{x,\mathbf{c}_l}} \\ \mathbf{0}_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}-d_D^{x,\mathbf{c}_l}}^{d_D^{x,\mathbf{c}_l}} & \boldsymbol{\Sigma}_\nu^\nu\end{bmatrix}\begin{bmatrix}(\mathbf{V}_s)^* \\ (\mathbf{V}_\nu)^*\end{bmatrix} \\ &= \mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* + \mathbf{U}^\nu\boldsymbol{\Sigma}_\nu^\nu(\mathbf{V}_\nu)^* = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}}\sigma_p\mathbf{u}^p\,(\tilde{\mathbf{v}}_p)^* + \sum_{p=d_D^{x,\mathbf{c}_l}+1}^{P}\sigma_p\mathbf{u}^p\,(\tilde{\mathbf{v}})^*,\end{aligned} \tag{6.2.140}$$

where the first summation in each of the last two expressions contains only vectors from $\mathbf{U}^s$ and $\mathbf{V}_s$ respectively, and the second only from $\mathbf{U}^\nu$ and $\mathbf{V}_\nu$ respectively. Note the resemblance of the partitioning described above to (6.2.128), and the partitioning into signal and noise subspaces that is reminiscent of the subspace methods explained in Chapters 3, 4 and 5.

Corollary 6.2.9. Reduced SVD of subspace matrix.
Under the assumptions formulated in Theorem 6.2.8, $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ can be written in the following reduced form:

$$\boxed{\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}}\sigma_p\,\mathbf{u}^p\,(\tilde{\mathbf{v}}_p)^*.} \tag{6.2.141}$$

Proof. Substituting $\sigma_{d_D^{x,\mathbf{c}_l}+1},\ldots,\sigma_P = 0$ into (6.2.140) directly yields the Corollary, which is the equivalent of Corollary (6.2.7). □

Using the notation introduced in Section 4.3.7 we can now write each sensor cumulant row vector $\tilde{\mathbf{r}}_{i_1 i_2}^x$ in the following natural SVD-form:

$$\boxed{\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}}\sigma_p\,u_{\mathbf{i}_l}^p\,(\tilde{\mathbf{v}}_p)^* \qquad \forall\,\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^l\,,} \tag{6.2.142}$$

which is the row vector equivalent of (6.2.130). Hence, the row vector $\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}$ is a linear combination of the row vectors $(\tilde{\mathbf{v}}_1)^*,\ldots,(\tilde{\mathbf{v}}_{d_D^{x,\mathbf{c}_l}})^*$ with coefficients $\sigma_1\,u_{\mathbf{i}_l}^1,\ldots,\sigma_{d_D^{x,\mathbf{c}_l}}\,u_{\mathbf{i}_l}^{d_D^{x,\mathbf{c}_l}}$, which is consistent with the results obtained in the previous section. Compare (6.2.142) with (6.2.71), which is repeated here for convenience:

$$\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l} = \sum_{j=1}^{S}\breve{a}_{\mathbf{i}_l}^{j,\mathbf{c}_l}\,\tilde{\boldsymbol{\kappa}}_j^{s,\mathbf{c}_l} \qquad \forall\,\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l\,. \tag{6.2.143}$$

This equation states that each sensor cumulant row vector $\tilde{\boldsymbol{\kappa}}_{\mathbf{i}_l}^{x,\mathbf{c}_l}$ is a linear combination of the S source auto-cumulant row vectors $\tilde{\boldsymbol{\kappa}}_1^{s,\mathbf{c}_l},\ldots,\tilde{\boldsymbol{\kappa}}_S^{s,\mathbf{c}_l}$ with coefficients $\breve{a}_{\mathbf{i}_l}^{1,\mathbf{c}_l},\ldots,\breve{a}_{\mathbf{i}_l}^{S,\mathbf{c}_l}$ respectively. Analogously to (4.3.122)-(4.3.125), the precise relations between the various (fundamental) subspaces can be formulated as follows (see also (6.2.132)-(6.2.135) and (D.1.8)-

(D.1.11)):

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{L}_c\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^s\big)\,; \tag{6.2.144}$$

$$\mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{R}_{\mathrm{r}}\big((\mathbf{V}_s)^*\big) \doteqdot \mathcal{L}_r\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) \subseteq \mathcal{R}_{\mathrm{r}}\big(\mathbf{C}_S^{s,\mathbf{c}_l}\big) \doteqdot \mathcal{L}_r\big(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\big)\,; \tag{6.2.145}$$

$$\mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{R}_{\mathrm{c}}\big((\mathbf{V}_\nu)^T\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) \supseteq \mathcal{N}_{\mathrm{r}}\big(\mathbf{C}_S^{s,\mathbf{c}_l}\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\boldsymbol{\kappa}_S^{s,\mathbf{c}_l}[\mathbf{n}_l]\big)\,; \tag{6.2.146}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big) = \mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{A}_{\mathrm{u},D,\diamond}^{\mathbf{c}_l}\big) = \big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^\nu\big)\big)^T\,. \tag{6.2.147}$$

This concludes our discussion on the matrix SVD of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. See the next section for explicit expressions of a proper set $\Phi_{D,S}^{\mathbf{c}_l}$ (6.2.61) containing sets of coefficients for the polyconjugal equations in our system, or equivalently a proper set of rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ (6.2.114), or a matrix $\boldsymbol{\Phi}$ (6.2.115) representing these coefficients, in terms of the left null space of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ found by the SVD.

6.2.8.3 Coefficients of polyconjugals in terms of SVD results

Finally, in this section we show how the coefficients of the equations in the system of homogeneous polyconjugal equations derived earlier can be expressed in terms of the left null space $\mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)$ of $\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]$ and $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. From (6.2.116), (6.2.117), (6.2.135) and (6.2.147) it follows directly how a proper set of sets of coefficients $\Phi_{D,S}^{\mathbf{c}_l}$, or equivalently rows $\{\tilde{\boldsymbol{\varphi}}_q\}_{q\in\mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ of the matrix $\boldsymbol{\Phi}$, can be expressed explicitly in terms of this left null space because these equations state that $\mathcal{N}_{\mathrm{l}}\big(\boldsymbol{\kappa}_{\mathrm{u},D}^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\big)$ is spanned by the transposed left singular vectors $\mathbf{u}^{d_D^{x,\mathbf{c}_l}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}$ that are contained in the set $\mathcal{U}^\nu$ and the matrix $\mathbf{U}^\nu$. Hence, proper nontrivial vectors $\tilde{\boldsymbol{\varphi}}_q$ satisfying (6.2.116) are given by:

$$\tilde{\boldsymbol{\varphi}}_q = \big(\mathbf{u}^{d_D^{x,\mathbf{c}_l}+q}\big)^H \qquad \forall\, 1 \le q \le Q_{D,S}^{\mathbf{c}_l}\,, \tag{6.2.148}$$

or by arbitrary linear combinations of these vectors:

$$\tilde{\boldsymbol{\varphi}}_q = \sum_{j=d_D^{x,\mathbf{c}_l}+1}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}} \alpha_q^j\,(\mathbf{u}^j)^H \qquad \forall\, \alpha_q^j \in \mathbb{C}\,. \tag{6.2.149}$$

Equivalently, a nontrivial matrix $\boldsymbol{\Phi}$ satisfying (6.2.117) is given by:

$$\boldsymbol{\Phi} = \begin{bmatrix} \tilde{\boldsymbol{\varphi}}_1 \\ \vdots \\ \tilde{\boldsymbol{\varphi}}_{Q_{D,S}^{\mathbf{c}_l}} \end{bmatrix} = (\mathbf{U}^\nu)^H = \begin{bmatrix} \big(\mathbf{u}^{d_D^{x,\mathbf{c}_l}+1}\big)^H \\ \vdots \\ \big(\mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}\big)^H \end{bmatrix}\,, \tag{6.2.150}$$

or any other matrix containing arbitrary linear combinations of the rows of $\boldsymbol{\Phi}$. Note that it can easily be shown that $\boldsymbol{\Phi}$ given by (6.2.150) satisfies (6.2.112):

$$\boldsymbol{\Phi}\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = (\mathbf{U}^\nu)^H\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} \overset{(6.2.141)}{=} (\mathbf{U}^\nu)^H\mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}}^N\,.$$

Examples of the computation of $\boldsymbol{\Phi}$ from the sensor cumulant values will be given in Section 6.3.

By realizing that we can set up a one-to-one mapping between the set $\left\{\varphi_q^{\mathbf{i}_l}\right\}^{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$ of coefficients corresponding to the q-th equation in the system and the row $\tilde{\boldsymbol{\varphi}}_q$, we can write the equivalent of (6.2.150) in terms of the set $\Phi_{D,S}^{\mathbf{c}_l}$ defined in (6.2.61) by means of two equivalent formulations. Firstly, from (6.2.148) it follows that for $1 \leq p \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}$:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{\mathbf{i}_{l,p}} = \left[\left(\mathbf{u}^{d_D^{x,\mathbf{c}_l}+q}\right)^H\right]^p = \left(u_p^{d_D^{x,\mathbf{c}_l}+q}\right)^*,$$

which yields:

$$\begin{aligned}\Phi_{D,S}^{\mathbf{c}_l} &\triangleq \left\{\left\{\varphi_q^{\mathbf{i}_l}\right\}^{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} = \left\{\left\{\left(u_p^{d_D^{x,\mathbf{c}_l}+q}\right)^*\right\}_{1 \leq p \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}}\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} \\ &= \left\{\left\{\left(u_1^{d_D^{x,\mathbf{c}_l}+1}\right)^*, \ldots, \left(u_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}^{d_D^{x,\mathbf{c}_l}+1}\right)^*\right\}, \ldots, \right. \\ &\qquad \left.\left\{\left(u_1^{d_D^{x,\mathbf{c}_l}+Q_{D,S}^{\mathbf{c}_l}}\right)^*, \ldots, \left(u_{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}^{d_D^{x,\mathbf{c}_l}+Q_{D,S}^{\mathbf{c}_l}}\right)^*\right\}\right\}.\end{aligned}$$

Secondly, using the mapping between the integer p and index tuple $\mathbf{i}_{l,p}$ for all $1 \leq p \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ (see (6.2.84)) it also follows for $1 \leq p \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ that:

$$[\tilde{\boldsymbol{\varphi}}_q]^p = \varphi_q^{\mathbf{i}_{l,p}} = \left[\left(\mathbf{u}^{d_D^{x,\mathbf{c}_l}+q}\right)^H\right]^p = \left(u_{\mathbf{i}_{l,p}}^{d_D^{x,\mathbf{c}_l}+q}\right)^*,$$

which yields:

$$\begin{aligned}\Phi_{D,S}^{\mathbf{c}_l} &\triangleq \left\{\left\{\varphi_q^{\mathbf{i}_l}\right\}^{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} = \left\{\left\{\left(u_{\mathbf{i}_{l,p}}^{d_D^{x,\mathbf{c}_l}+q}\right)^*\right\}_{1 \leq p \leq M_{\mathrm{u},D}^{x,\mathbf{c}_l}}\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} \\ &= \left\{\left\{\left(u_{\mathbf{i}_{l,1}}^{d_D^{x,\mathbf{c}_l}+1}\right)^*, \ldots, \left(u_{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}}^{d_D^{x,\mathbf{c}_l}+1}\right)^*\right\}, \ldots, \right. \\ &\qquad \left.\left\{\left(u_{\mathbf{i}_{l,1}}^{d_D^{x,\mathbf{c}_l}+Q_{D,S}^{\mathbf{c}_l}}\right)^*, \ldots, \left(u_{\mathbf{i}_{l,M_{\mathrm{u},D}^{x,\mathbf{c}_l}}}^{d_D^{x,\mathbf{c}_l}+Q_{D,S}^{\mathbf{c}_l}}\right)^*\right\}\right\}.\end{aligned}$$

This concludes our derivation of the coefficients of the polyconjugals in the system.

Note that the Singular Value Decomposition confirms once more that the number of linearly independent equations in the system is given by (6.2.60), (6.2.80), and (6.2.113). This follows directly by considering the dimension of the left null space $\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)$ of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. See also (6.2.149) for example. Since the singular vectors $\mathbf{u}^{d_D^{x,\mathbf{c}_l}+1}, \ldots, \mathbf{u}^{M_{\mathrm{u},D}^{x,\mathbf{c}_l}}$ correspond to the zero singular values, in subspace terminology we call them noise subspace vectors or 'zero subspace vectors'. Also note from the results above the notational consistency that the transpose operation effectively turns a superscript index into a subscript index, and vice versa. Finally, note that in practice only an approximate estimated version $\hat{\mathbf{C}}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ is available. In this case, the singular values are divided into two sets, one with the $d_D^{x,\mathbf{c}_l}$ largest ones, and the other with the remaining $M_{\mathrm{u},D}^{x,\mathbf{c}_l} - d_D^{x,\mathbf{c}_l}$ smallest ones. This implies that in the statements made above $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,\mathbf{c}_l}} > 0,\ \sigma_{d_D^{x,\mathbf{c}_l}+1}, \ldots, \sigma_P = 0$ has to be replaced with $\sigma_1 \geq \cdots \geq \sigma_{d_D^{x,\mathbf{c}_l}} > 0,\ \sigma_{d_D^{x,\mathbf{c}_l}+1}, \ldots, \sigma_P \approx 0$. This stage completes the derivation of the system of equations. The algorithmic part of the method is summarized in Alg. 6.2 on the next page, which is a generalization of Algorithms 4.3 on page 171 and 5.2 on page 268. The last step of the algorithm will be discussed in later sections.

Algorithm 6.2 High-level algorithm for $D \times S$ MIBI exploiting l-th order temporal structure with arbitrary conjugation tuple $\mathbf{c}_l$.

1: Compute/estimate sensor cumulant functions in set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}}$ *for time index tuples* $\mathbf{n}_l$ *in Noise-Free ROS* $\Omega_{\mathbf{n}_l}^{s\setminus\nu,\mathbf{c}_l}$*;*

2: Arrange these values in uniquiefied sensor cumulant subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$*;*

3: Compute Singular Value Decomposition (SVD) of $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$ *and split result into signal and null/noise subspace parts as follows:*

$$\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V})^* = \mathbf{U}^s\boldsymbol{\Sigma}_s^s(\mathbf{V}_s)^* + \mathbf{U}^\nu\boldsymbol{\Sigma}_\nu^\nu(\mathbf{V}_\nu)^*\,;$$

4: Compute matrix $\boldsymbol{\Phi}$ *whose rows span complex conjugate left null space* $\left(\mathcal{N}_{\mathrm{l}}\left(\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}\right)\right)^*$ *of* $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$*:*

$$\boldsymbol{\Phi} \triangleq (\mathbf{U}^\nu)^H\,;$$

5: With each row $\tilde{\boldsymbol{\varphi}}_q$ *of* $\boldsymbol{\Phi}$ *with* $q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}$ *associate a* D*-variate polyconjugal that is homogeneous of degree* l *with conjugation tuple* $\mathbf{c}_l$*:*

$$f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}} \varphi_q^{\mathbf{i}_l}\, \check{z}_{\mathbf{i}_l}^{\mathbf{c}_l} \;=\; \tilde{\boldsymbol{\varphi}}_q\, \check{\mathbf{z}}_{\mathrm{u},D}^{\mathbf{c}_l} \;=\; \tilde{\boldsymbol{\varphi}}_q\, \mathbf{w}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathbf{z})\,;$$

6: The following system remains to be solved for the columns of the mixing matrix:

$$\left\{f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) = 0\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}} \quad \equiv \quad \mathbf{f}_{D,S}^{\mathbf{c}_l}(\mathbf{z}) = \boldsymbol{\Phi}\,\check{\mathbf{z}}_{\mathrm{u},D}^{\mathbf{c}_l} = \boldsymbol{\Phi}\,\mathbf{w}_{\mathrm{u},D}^{\mathbf{c}_l}(\mathbf{z}) = \mathbf{0}_{Q_{D,S}^{\mathbf{c}_l}}\,.$$

6.3 Algebraic and geometric structure

The purpose of this section is to provide insight into the algebraic and geometric structure of the problem induced by the system $\left\{f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) = 0\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ in (6.2.66). As we have shown in Sections 6.2.4.3 and 6.2.7, the type of function we are considering is defined by (6.2.64) and (6.2.123). For convenience, we write the function in elaborate form as follows:

$$f_{D,q}^{c_1\cdots c_l}(z_1,\ldots,z_D) \triangleq \sum_{(i_1,\ldots,i_l)\in\mathcal{I}_{\mathrm{u},D}^{c_1\cdots c_l}} \varphi_q^{i_1\cdots i_l}\,(z_{i_1})^{c_1}\cdots(z_{i_l})^{c_l}$$

$$\forall\, z_1,\ldots,z_D \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}\,. \tag{6.3.1}$$

By definition, the set of values of the tuple $(z_1,\ldots,z_D)$ for which $f_{D,q}^{c_1\cdots c_l}(z_1,\ldots,z_D) = 0$ is the zero contour level of $f_{D,q}^{c_1\cdots c_l}(z_1,\ldots,z_D)$. Hence, geometrically, finding the different solutions of the system $\left\{f_{D,q}^{\mathbf{c}_l}(\mathbf{z}) = 0\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}}$ is equivalent to finding the intersections between the zero contour levels of the functions $\left\{f_{D,q}^{\mathbf{c}_l}(\mathbf{z})\right\}_{q \in \mathcal{Q}_{D,S}^{\mathbf{c}_l}}$. In Section 6.2.4.4 we have seen that the function $f_{D,q}^{\mathbf{c}_l}(\mathbf{z})$ defined in (6.3.1) is a D-variate homogeneous polyconjugal

of degree l with conjugation tuple $\mathbf{c}_l$, which possesses the important *homogeneity property* formulated in (6.2.67). This latter property implies that if $\mathbf{v} \in \mathbb{C}_D$ is a solution of the system $\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) = 0\}_{q \in \mathcal{Q}^{c_1 c_2}_{D,S}}$, then so is $\eta\mathbf{v}$ for all $\eta \in \mathbb{C}$; see (6.2.68). Therefore, only solution vectors can be distinguished that are different according to the equivalence relation defined in (5.4.2) on page 269. According to this relation, vectors that have the same or opposite directions and different lengths are considered to be equivalent, and all other vectors are different. This is a logical result of the scaling indeterminacy discussed in Section 2.4. Now, from Definition 4.4.1 it is clear that the zero contour level of each function $f^{c_1 \cdots c_l}_{D,q}(z_1, \ldots, z_D)$ defines a cone in the D-dimensional Euclidian space $\mathbb{C}_D$. Thus:

Geometrically, solving the system $\left\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z}) = 0\right\}_{q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}}$ is equivalent to finding the one-dimensional intersections between $Q^{\mathbf{c}_l}_{D,S}$ cones in a D-dimensional Euclidian space that represent the zero contour levels of the functions $\left\{f^{\mathbf{c}_l}_{D,q}(\mathbf{z})\right\}_{q \in \mathcal{Q}^{\mathbf{c}_l}_{D,S}}$.

Ideally, each intersection between the cones is a one-dimensional subspace that corresponds to a column of the mixing matrix. Our purpose in solving (6.2.66) is to find all non-zero solutions $\mathbf{z}^1, \ldots, \mathbf{z}^S$ that are different according to relation (5.4.2).

Now we are in a position to discuss several examples that will elucidate the theory that has been presented so far in this chapter. The examples focus on highlighting algebraic and geometric structure. The actual estimation of the mixing matrix columns and, if applicable, the separated signals will be presented in Section 6.4. For ease of visualization, here we will only consider real-valued systems and signals. Therefore, the conjugation tuple $\mathbf{c}_l$, which now is given by $(\circ)_l$, is irrelevant and will be omitted from the notation or replaced by the number l representing the employed statistical order. See Chapter 7 for examples with complex-valued scenarios with different conjugation tuples. In the following sections, we will give examples with higher order odd, viz. third, and even, viz. fourth, order statistics. Only the results obtained from estimated subspace matrices are presented.

6.3.1 Real-valued scenarios with two sensors, stationary Gamma AR(1) source signals, additive white noise, and third order statistics

In this section we examine and perform the various steps of Alg. 6.2 on the previous page for two real-valued MIBI examples with signal scenarios and Regions Of Support that comply with the assumptions described in Section 6.1, and using third order statistics. The first example deals with the 2×2 and the second with the 2×3 mixing case. Both examples involve stationary Gamma AR(1) source signals and independently and uniformly distributed additive sensor noise signals (see Section 6.3.1.1). Note that for third order statistics the employed sensor cumulant function definition in (6.2.1) becomes:

$$\kappa^{x,\circ\circ\circ}_{i_1 i_2 i_3}[n_1, n_2, n_3] \triangleq \operatorname{cum}\Big(x_{i_1}[n_1], x_{i_2}[n_2], x_{i_3}[n_3]\Big) = E\Big\{x_{i_1}[n_1]\, x_{i_2}[n_2]\, x_{i_3}[n_3]\Big\}\,.$$

Because all signals are assumed to be stationary the involved third order cumulant functions depend two lags, say k_1 and k_2 only. Hence, with abuse of notation we may define and use:

$$\kappa^{x,3}_{i_1 i_2 i_3}[k_1, k_2] \triangleq \kappa^{x,\circ\circ\circ}_{i_1 i_2 i_3}[k_1, k_2] = E\Big\{x_{i_1}[n]\, x_{i_2}[n-k_1]\, x_{i_3}[n-k_2]\Big\} \qquad \forall\, k_1, k_2 \in \mathbb{Z}\,. \tag{6.3.2}$$

Accordingly, the various regions of support can also be chosen in the domain of lag pairs (k_1, k_2) instead of in the domain of time index triples (n_1, n_2, n_3), and we will write $\Omega^{s\backslash\nu}_{k_1 k_2}$ instead of $\Omega^{s\backslash\nu}_{n_1 n_2 n_3}$ for the Noise-Free ROS. Computing the values in (6.3.2) on a properly defined Noise-Free ROS (see Section 6.3.1.1) yields the subspace matrix.

Substituting $D = 2$, $l = 3$, and $c_1 = c_2 = c_3 = \circ$ into (6.2.66) and (6.3.1) respectively it follows that the system of equations resulting from Alg. 6.2 is given by:

$$\Big\{ f^3_{2,q}(z_1, z_2) = 0 \Big\}_{q \in \mathcal{Q}^3_{2,S}} \tag{6.3.3}$$

and that its functions take the following *cubic form*:

$$f^3_{2,q}(z_1, z_2) = \sum_{(i_1,i_2,i_3) \in \mathcal{I}^3_{\mathrm{a},2}} \varphi^{i_1 i_2 i_3}_q z_{i_1} z_{i_2} z_{i_3} = \varphi^{111}_q z_1 z_1 z_1 + \varphi^{112}_q z_1 z_1 z_2 + \varphi^{122}_q z_1 z_2 z_2 + \varphi^{222}_q z_2 z_2 z_2 \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}^3_{2,S}\,. \tag{6.3.4}$$

The set $\mathcal{Q}^3_{2,S} \triangleq \mathcal{Q}^{\circ\circ\circ}_{2,S}$ is defined by (6.2.59) and contains the integers $1, \ldots, Q^3_{2,S}$, where the number $Q^3_{2,S} \triangleq Q^{\circ\circ\circ}_{2,S}$ of linearly independent equations in (6.3.3) is given by (6.2.60), (6.2.80), and/or (6.2.113). For scenarios with D sensors and order l without conjugations the number $M^{x,l}_{\mathrm{u},D} \triangleq M^{x,(\circ)_l}_{\mathrm{u},D} \equiv M^{x,l}_{\mathrm{a},D}$ of unique rows of the subspace matrix $\mathbf{C}^{x,l}_D \triangleq \mathbf{C}^{x,(\circ)_l}_D$, which equals the number $M^l_{\mathrm{u},D} \triangleq M^{(\circ)_l}_{\mathrm{u},D} \equiv M^l_{\mathrm{a},D}$ of unique rows of the l-th order Khatri-Rao product $\mathbf{A}^l_{D,\diamond} \triangleq \mathbf{A}^{(\circ)_l}_{D,\diamond}$, is given by (6.2.26) and (6.2.33) as $\left\langle {D \atop l} \right\rangle$. The corresponding index set $\mathcal{I}^l_{\mathrm{u},D} \equiv \mathcal{I}^l_{\mathrm{a},D}$ is given in (6.2.25). See (6.2.29) for the case with $D = 2$ and $l = 3$. In the MIBI examples that will be presented shortly the employed mixing matrices have full rank D, i.e. $\mathrm{rank}(\mathbf{A}) = D = 2$, while their corresponding Khatri-Rao products have full rank S, i.e. $\mathrm{rank}(\mathbf{A}^3_{\mathrm{u},2,\diamond}) = S$. Hence, according to (6.2.109) the signal subspace dimension $d^{x,3}_2 \triangleq d^{x,\circ\circ\circ}_2 = \mathrm{rank}(\mathbf{C}^{x,3}_{\mathrm{u},2}) = \mathrm{rank}(\mathbf{A}^3_{\mathrm{u},2,\diamond})$ also equals S. Substituting those results into (6.2.60), (6.2.80), or (6.2.113), it follows that the number of linearly independent equations in (6.3.3) equals:

$$Q^3_{2,S} = \left\langle {2 \atop 3} \right\rangle - d^{x,3}_2 = \binom{2+3-1}{3} - \mathrm{rank}(\mathbf{A}^3_{\mathrm{u},2,\diamond}) = 4 - S\,, \tag{6.3.5}$$

i.e. $\mathcal{Q}^3_{2,S} = \{1, \ldots, 4 - S\}$. Using (6.2.125) the system in (6.3.3) can be written as follows:

$$\mathbf{f}^3_{2,S}(\mathbf{z}) = \mathbf{\Phi}\, \check{\mathbf{z}}^3_{\mathrm{u},2} = \mathbf{\Phi}\, \mathbf{w}^3_{\mathrm{u},2}(\mathbf{z}) = \mathbf{0}_{Q^3_{2,S}}\,, \tag{6.3.6}$$

where $\check{\mathbf{z}}^3_{\mathrm{u},2} = \check{\mathbf{z}}^{\circ\circ\circ}_{\mathrm{u},2}$ and $\mathbf{w}^3_{\mathrm{u},2}(\mathbf{z}) = \mathbf{w}^{\circ\circ\circ}_{\mathrm{u},2}(\mathbf{z})$ are defined in (6.2.27), and $\mathbf{\Phi}$ is a $Q^3_{2,S} \times 4$ matrix. In the following sections we will demonstrate that (6.3.3) as derived according to Alg. 6.2 can determine up to $l = 3$ mixing matrix columns. In Section 6.3.1.1 we first describe the signal scenario is some detail. Then, in Sections 6.3.1.2 and 6.3.1.3 we analyze examples with two and three sources respectively.

6.3.1.1 Signal scenario

The source signals are generated in such a way that they possess the same temporal structure as in Section 5.4.1, i.e. they are generated according to the AR(1) model in (5.2.13) (also the same regression coefficients are used). Because now we want to exploit third order statistics the distribution of the source signal samples should be odd. In order to achieve this, the random variables of the sequences $w_1[n]$, ..., $w_S[n]$ are drawn from a pdf with odd symmetry, viz. the Gamma probability density function. The Gamma pdf of a zero-mean real-valued

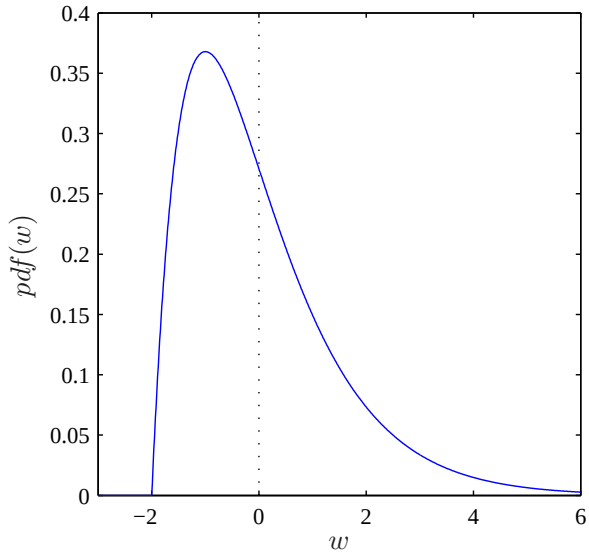

Figure 6.2: *Gamma probability density function for $\alpha = 2$ and $\lambda = 1$.*

random variable w with shape parameter α and scale parameter λ, both of which are real-valued positive numbers, is given by:

$$p_w(w) \triangleq \frac{1}{(\lambda)^{\alpha}\Gamma(\alpha)}\,\epsilon(w+\alpha\lambda)\,(w+\alpha\lambda)^{\alpha-1}\,e^{-\frac{w+\alpha\lambda}{\lambda}}\,, \tag{6.3.7}$$

where $\epsilon(\cdot)$ denotes the step and $\Gamma(\cdot)$ the Gamma function. The variance, normalized *skewness*[‡], and normalized *kurtosis*[‡] of this distribution equal $\alpha(\lambda)^2$, $2/\sqrt{\alpha}$, and $6/\alpha$ respectively. In this example we use the same value of α for generating all excitation sequences $w_j[n]$ in (5.2.13) with $1 \leq j \leq S$, in particular, we choose $\alpha = 2$. Note that the essential shape of the Gamma pdf is determined only by α. Here, we choose $\lambda = 1$ and normalize the source signals to unit variance afterwards. For $\alpha = 2$ the normalized skewness and kurtosis equal $\sqrt{2}$ and 3 respectively. As is explained in Section B.1.3.3, the skewness characterizes the degree of asymmetry of the pdf about its mean. Since the skewness is not zero, the Gamma pdf is an asymmetric function. Similarly, the kurtosis of a random variable characterizes the degree of flatness or peakedness of the pdf about its mean. Since the normalized kurtosis is positive, the Gamma distribution is *leptokurtic* or *Super-Gaussian* (see Section B.1.3.3). Fig. 6.2 depicts the probability density function for $\alpha = 2$ and $\lambda = 1$.

The noise signals $\nu_1[n]$ and $\nu_2[n]$ are statistically independent identically and uniformly distributed sequences with standard deviation 0.5. Because the third order cumulants of uniformly distributed random variables equal zero the Noise-Free ROS $\Omega^{s\setminus\nu}_{k_1k_2}$ may contain all possible lag pairs (k_1, k_2). Furthermore, in order to have a full rank source auto-cumulant matrix, the cardinality of the Noise-Free ROS should at least be equal to the number of sources S. In the following examples we will use $\Omega^{s\setminus\nu}_{k_1k_2} = \{(0,0),(0,1),(1,1)\}$ and compute and plot the zero contour levels for subspace matrices estimated from computer generated sensor signals.

[‡] See Section B.1.3.3.

From the sensor signals, we estimate the unique third order time-averaged sensor cumulant functions defined in (6.3.2) for all lag pairs $(k_1, k_2) \in \Omega_{k_1 k_2}^{s \setminus \nu}$ by averaging products of the form $x_{i_1}[n]\, x_{i_2}[n-k_1]\, x_{i_3}[n-k_2]$ over the available time samples:

$$\boxed{\hat{\kappa}_{i_1 i_2 i_3}^{x,3}[k_1, k_2] = \frac{1}{N_s} \sum_{n=1}^{N_s} x_{i_1}[n]\, x_{i_2}[n-k_1]\, x_{i_3}[n-k_2]\,.} \tag{6.3.8}$$

6.3.1.2 Two sensors, two sources and third order cumulants

Let the mixing matrix be given by:

$$\mathbf{A} = \left[\begin{bmatrix} 0 \\ 1 \end{bmatrix} \; \tfrac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right] \approx \begin{bmatrix} 0 & 0.7071 \\ 1 & 0.7071 \end{bmatrix}\,. \tag{6.3.9}$$

The source signals $s_1[n]$ and $s_2[n]$ are unit variance signals consisting of $N_s = 15000$ time samples and are generated as described in Section 6.3.1.1 with $\rho_1 = 0.9$, $\rho_2 = -0.8$, and the shape parameter α of the Gamma distribution (6.3.7) set to 2. The sensor signals $x_1[n]$ and $x_2[n]$ are computed according to (5.1.1)/(5.1.2), where $\mathbf{A}$ is given by (6.3.9) and the noise signals $\nu_1[n]$ and $\nu_2[n]$ are mutually statistically independent white uniformly distributed noise sequences with variance 0.25, as is also described in the previous section. This results in SNR's equal to 3.0 and 7.8 dB at the first and second sensors respectively. For a specific realization, Fig. 6.3 shows 200 samples of the source signals at the left side and their noisy mixtures at the right side. Computing the rank of $\mathbf{A}_{\mathrm{u},2,\diamond}^{3}$ for the values in (6.3.9) shows that $\mathrm{rank}(\mathbf{A}_{\mathrm{u},2,\diamond}^{3}) = S = 2$. Substituting this into (6.3.5) it follows that the number of linearly independent equations equals 2, i.e. $\mathcal{Q}_{2,2}^{3} = \{1, 2\}$. Thus, the ideal system contains the two equations $f_{2,1}^{3}(z_1, z_2) = 0$ and $f_{2,2}^{3}(z_1, z_2) = 0$, where the form of the functions is

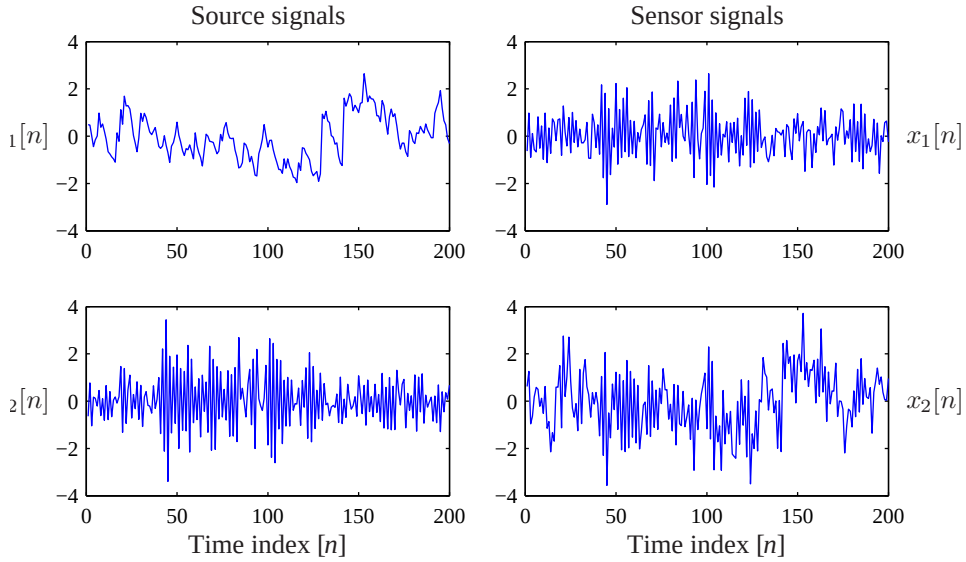

Figure 6.3: *Stationary Gamma $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 2×2 example.*

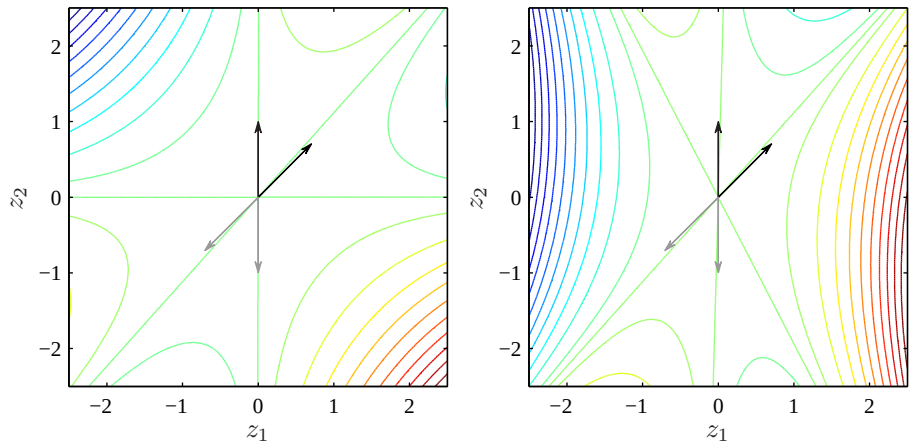

Figure 6.4: *Contour plots of $\hat{f}^3_{2,1}(z_1, z_2)$ (left) and $\hat{f}^3_{2,2}(z_1, z_2)$ (right) for $D = 2$, $S = 2$, and $l = 3$.*

specified by (6.3.4). Using formula (6.3.8) with $(k_1, k_2) \in \Omega^{s\setminus\nu}_{k_1 k_2} = \{(0,0),(0,1),(1,1)\}$ for estimating the subspace matrix from the sensor signals gives an estimate $\hat{\mathbf{C}}^{x,3}_{\mathrm{u},2}$ of $\mathbf{C}^{x,3}_{\mathrm{u},2}$ that can be used in Alg. 6.2 on page 341 from the second step on. Steps 3 and 4 of this algorithm yield an estimate of the matrix $\mathbf{\Phi}$ and thus also estimates of the functions $f^3_{2,1}(z_1, z_2)$ and $f^3_{2,2}(z_1, z_2)$. The corresponding estimated zero contour plots are shown in Fig. 6.4. The black and grey arrows denote the ideal columns of $\mathbf{A}$ and their opposites respectively. The three crossing straight lines through the middle of each of the figures correspond to the zero contour level of the corresponding function. It is clear from the two figures that the columns of the mixing matrix are defined uniquely and approximately by the intersections of the zero contour levels of the two functions. Two of the three lines defining the zero contour level of each function are approximately in the directions of the columns of the ideal mixing matrix and one line is 'superfluous', which is an indication of the fact that one degree of freedom is left unused. This degree of freedom can be exploited for estimating one extra column, as demonstrated in the next example.

6.3.1.3 Two sensors, three sources and third order cumulants

Now let the mixing matrix be given by:

$$\mathbf{A} = \left[\begin{bmatrix}0\\1\end{bmatrix} \ \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix} \ \frac{1}{\sqrt{13}}\begin{bmatrix}-2\\3\end{bmatrix}\right] \approx \begin{bmatrix}0 & 0.7071 & -0.5547\\ 1 & 0.7071 & 0.8321\end{bmatrix}, \tag{6.3.10}$$

i.e. one column is appended to the matrix in the previous example. The source signals $s_1[n]$, $s_2[n]$ and $s_3[n]$ are unit variance signals consisting of $N_s = 25000$ time samples and are also generated as described in Section 6.3.1.1 with $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$, and $\alpha = 2$. Again, the sensor signals $x_1[n]$ and $x_2[n]$ are computed according to (5.1.1)/(5.1.2), where $\mathbf{A}$ is now given by (6.3.10) and the noise signals $\nu_1[n]$ and $\nu_2[n]$ are the same as in the previous section. The SNR's at the first and second sensors now equal 5.1 and 9.5 dB respectively. For a specific realization, Fig. 6.5 shows 200 samples of the source signals at the left side and their noisy mixtures at the right side. Computing the rank of $\mathbf{A}^3_{\mathrm{u},2,\diamond}$ for the values in (6.3.10) shows that $\mathrm{rank}(\mathbf{A}^3_{\mathrm{u},2,\diamond}) = S = 3$. Substituting this into (6.3.5) it follows

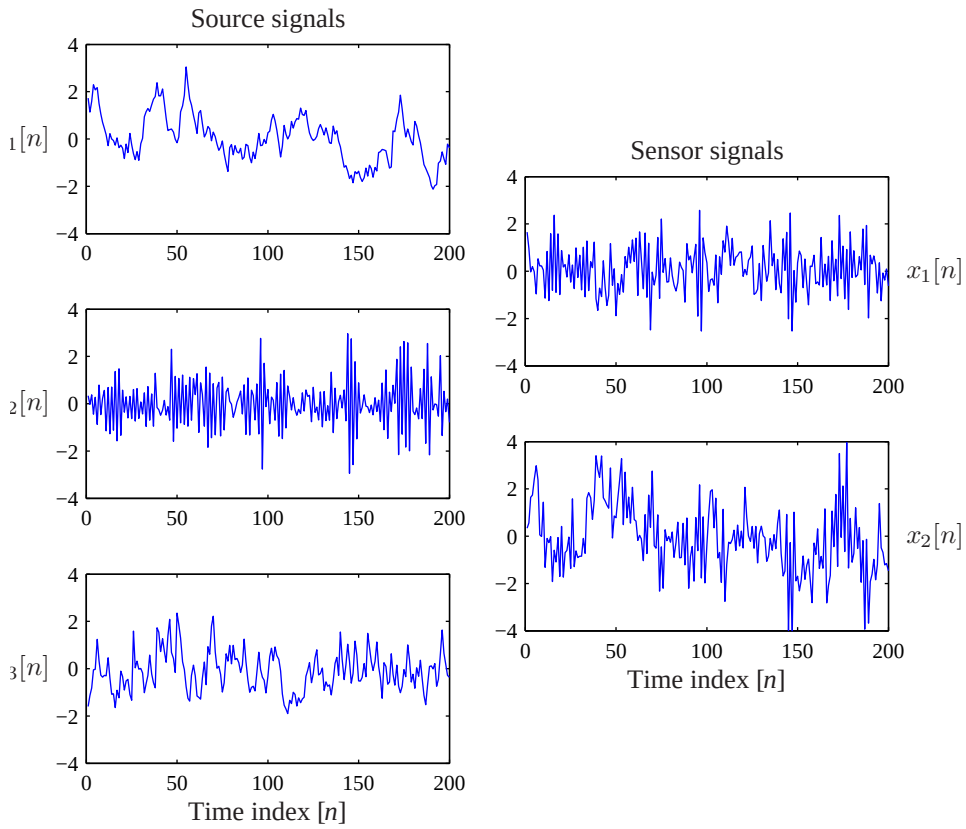

Figure 6.5: *Stationary Gamma $AR(1)$ source signals (left) and their noise-contaminated mixtures (right) for 2×3 example.*

that now the number of linearly independent equations equals one. This indicates that no more than three columns of the mixing matrix can be identified with two sensors and third order cumulants. The ideal system consists of the single equation $f^3_{2,1}(z_1, z_2) = 0$, whose form again is specified by (6.3.4). Using the subspace matrix $\hat{\mathbf{C}}^{x,3}_{\mathrm{u},2}$ with exactly the same structure as in the previous example in Alg. 6.2 yields an estimate $\hat{f}^3_{2,1}(z_1, z_2)$ of $f^3_{2,1}(z_1, z_2)$. The corresponding contour plot is shown in Fig. 6.6. The black and grey arrows denote the ideal columns of $\mathbf{A}$ and their opposites respectively. It is clear from the contour plot that the columns of the mixing matrix are defined uniquely and approximately by the zero contour level consisting of the three crossing straight lines through the middle of the figure. Note that because there is only one equation no intersections between cones need to be computed in order to estimate the columns.

Note that compared to the example with two sources in the previous section, now more samples have been used for estimating the subspace matrix. This is due to the fact that the sensor signals are (cor)related much stronger now because three instead of two source signals are mixed into two observed signals. In general, the more sources are present for a fixed number of sensors, the more samples are required for the reliable estimation of the statistics.

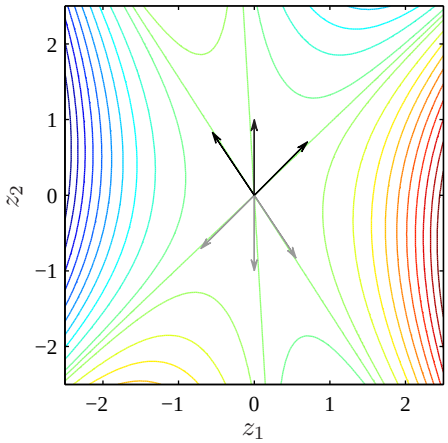

Figure 6.6: *Contour plot of $\hat{f}^{3}_{2,1}(z_1, z_2)$ for $D = 2$, $S = 3$, and $l = 3$.*

6.3.2 Real-valued scenarios with two sensors, stationary Gamma AR(1) source signals, additive white noise, and fourth order statistics

In this section we examine and perform the various steps of Alg. 6.2 on page 341 for three real-valued MIBI examples with signal scenarios similar to those in Section 6.3.1, but now using fourth order statistics. The first, second and third examples deal with the 2×2, 2×3 and 2×4 mixing cases respectively. All examples use the same signal types as in Section 6.3.1, viz. stationary Gamma AR(1) source signals and independently and uniformly distributed additive sensor noise signals. As in Section 5.4.1 and the previous examples the employed AR(1) regression coefficients are given by $\rho_1 = 0.9$, $\rho_2 = -0.8$, $\rho_3 = 0.6$, and $\rho_4 = -0.4$. As in the previous section, only the results obtained from estimated subspace matrices are presented. Furthermore, only contour levels and no signals are plotted. From (B.2.44d) it follows that for zero-mean random variables and fourth order statistics the employed sensor cumulant function definition in (6.2.1) becomes:

$$\begin{aligned}
&\kappa^{x,\circ\circ\circ\circ}_{i_1 i_2 i_3 i_4}[n_1, n_2, n_3, n_4] \triangleq \operatorname{cum}\Big(x_{i_1}[n_1], x_{i_2}[n_2], x_{i_3}[n_3], x_{i_4}[n_4]\Big) \\
&= E\big\{x_{i_1}[n_1]\, x_{i_2}[n_2]\, x_{i_3}[n_3]\, x_{i_4}[n_4]\big\} - E\big\{x_{i_1}[n_1]\, x_{i_2}[n_2]\big\}E\big\{x_{i_3}[n_3]\, x_{i_4}[n_4]\big\} \\
&\quad - E\big\{x_{i_1}[n_1]\, x_{i_3}[n_3]\big\}E\big\{x_{i_2}[n_2]\, x_{i_4}[n_4]\big\} - E\big\{x_{i_1}[n_1]\, x_{i_4}[n_4]\big\}E\big\{x_{i_2}[n_2]\, x_{i_3}[n_3]\big\}\,.
\end{aligned}$$

Because all signals are assumed to be stationary the involved fourth order cumulant functions depend three lags, say k_1, k_2 and k_3. Hence, with abuse of notation we may define and use:

$$\begin{aligned}
&\kappa^{x,4}_{i_1 i_2 i_3 i_4}[k_1, k_2, k_3] \triangleq \kappa^{x,(\circ)_4}_{i_1 i_2 i_3 i_4}[k_1, k_2, k_3] \\
&\quad \triangleq \operatorname{cum}\Big(x_{i_1}[n], x_{i_2}[n-k_1], x_{i_3}[n-k_2], x_{i_4}[n-k_3]\Big) \qquad \forall\, k_1, k_2, k_3 \in \mathbb{Z}\,. \qquad (6.3.11)
\end{aligned}$$

Accordingly, the various regions of support can also be chosen in the (k_1, k_2, k_3)-domain instead of in the domain of time index quadruples $\mathbf{n}_4 = (n_1, n_2, n_3, n_4)$, and we will write $\Omega^{s\setminus\nu}_{k_1 k_2 k_3}$ instead of $\Omega^{s\setminus\nu}_{n_1 n_2 n_3 n_4}$ for the Noise-Free ROS. Computing/estimating the values in (6.3.11) on a properly defined Noise-Free ROS (see further) yields the subspace matrix. The cumulant values can be estimated from the sensor signals by first estimating the individual

joint moments similarly to (4.4.18) and (6.3.8), and then using (6.3.11). Because the noise signals $\nu_1[n]$ and $\nu_2[n]$ are uniformly distributed sequences, their fourth order cumulants are non-zero. Therefore, contrary to the examples in Section 6.3.1, the Noise-Free ROS $\Omega^{s\setminus\nu}_{k_1k_2k_3}$ may not contain all possible lag-triples (k_1, k_2, k_3), but now $(0, 0, 0)$ is excluded. Again, in order to have a full rank source auto-cumulant matrix, the cardinality of the Noise-Free ROS should at least be equal to the number of sources S. In the following examples we will use $\Omega^{s\setminus\nu}_{k_1k_2k_3} = \{(0, 0, 1), (0, 0, 2), (0, 1, 2), (1, 1, 2)\}$ and compute and plot the zero contour levels for subspace matrices estimated from computer generated sensor signals.

Substituting $D = 2$, $l = 4$, and $c_1 = c_2 = c_3 = c_4 = \circ$ into (6.2.66) and (6.3.1) respectively it follows that the system of equations resulting from Alg. 6.2 is given by:

$$\left\{ f^4_{2,q}(z_1, z_2) = 0 \right\}_{q\in\mathcal{Q}^4_{2,S}} \tag{6.3.12}$$

and that its functions take the following *tetradic form*:

$$\begin{aligned} f^4_{2,q}(z_1, z_2) = {} & \varphi^{1111}_q z_1z_1z_1z_1 + \varphi^{1112}_q z_1z_1z_1z_2 + \varphi^{1122}_q z_1z_1z_2z_2 \\ & + \varphi^{1222}_q z_1z_2z_2z_2 + \varphi^{2222}_q z_2z_2z_2z_2 \qquad \forall\, z_1, z_2 \in \mathbb{C}\,, \quad \forall\, q \in \mathcal{Q}^4_{2,S}\,. \end{aligned} \tag{6.3.13}$$

The set $\mathcal{Q}^4_{2,S} \triangleq \mathcal{Q}^{(\circ)_4}_{2,S}$ is defined by (6.2.59) and contains the integers $1, \ldots, Q^4_{2,S}$, where the number $Q^4_{2,S} \triangleq Q^{(\circ)_4}_{2,S}$ of linearly independent equations in (6.3.12) is given by (6.2.60), (6.2.80), and/or (6.2.113). In the MIBI examples that will be presented shortly the employed mixing matrices have full rank D, i.e. $\mathrm{rank}(\mathbf{A}) = D = 2$, while their corresponding Khatri-Rao products have full rank S, i.e. $\mathrm{rank}(\mathbf{A}^4_{\mathrm{u},2,\diamond}) = S$. Hence, according to (6.2.109) the signal subspace dimension $d^{x,4}_2 \triangleq d^{x,(\circ)_4}_2 = \mathrm{rank}(\mathbf{C}^{x,4}_{\mathrm{u},2}) = \mathrm{rank}(\mathbf{A}^4_{\mathrm{u},2,\diamond})$ also equals S. Substituting those results into (6.2.60), (6.2.80), or (6.2.113), it follows that the number of linearly independent equations in (6.3.12) equals:

$$Q^4_{2,S} = \left\langle {2 \atop 4} \right\rangle - d^{x,4}_2 = \binom{2+4-1}{4} - \mathrm{rank}(\mathbf{A}^4_{\mathrm{u},2,\diamond}) = 5 - S\,, \tag{6.3.14}$$

i.e. $\mathcal{Q}^4_{2,S} = \{1, \ldots, 5 - S\}$. In the following sections we will demonstrate that (6.3.12) as derived according to Alg. 6.2 can determine up to $l = 4$ mixing matrix columns.

6.3.2.1 Two sensors, two sources and fourth order cumulants

Let the mixing matrix be given by (6.3.9). See Fig. 6.3 for a specific realization of the source and sensor signals. According to (6.3.14) the number of linearly independent equations equals 3, i.e. $\mathcal{Q}^4_{2,2} = \{1, 2, 3\}$. Thus, the system of equations consists of three equations with functions of the form specified by (6.3.13). Using $N_s = 20000$ time samples the sensor cumulant functions defined in (6.3.11) are estimated on the Noise-Free ROS $\Omega^{s\setminus\nu}_{k_1k_2k_3} = \{(0, 0, 1), (0, 0, 2), (0, 1, 2), (1, 1, 2)\}$. This gives an estimate $\hat{\mathbf{C}}^{x,4}_{\mathrm{u},2}$ of the subspace matrix $\mathbf{C}^{x,4}_{\mathrm{u},2}$ that can be used in Alg. 6.2 on page 341 from the second step on. Steps 3 and 4 of this algorithm yield an estimate of the matrix $\boldsymbol{\Phi}$ and thus also estimates of the functions $f^4_{2,1}(z_1, z_2)$, $f^4_{2,2}(z_1, z_2)$ and $f^4_{2,3}(z_1, z_2)$. The corresponding estimated zero contour plots are shown in Fig. 6.7. As before, the black and grey arrows denote the ideal columns of $\mathbf{A}$ and their opposites respectively. Now there are four crossing straight lines through the middle of each figure defining the zero contour level of the corresponding function. For the third function, two of the four lines defining the zero contour level are in 'complex directions' and therefore invisible in the real $z_1 - z_2$ plane. It is clear from the contour plots that the two

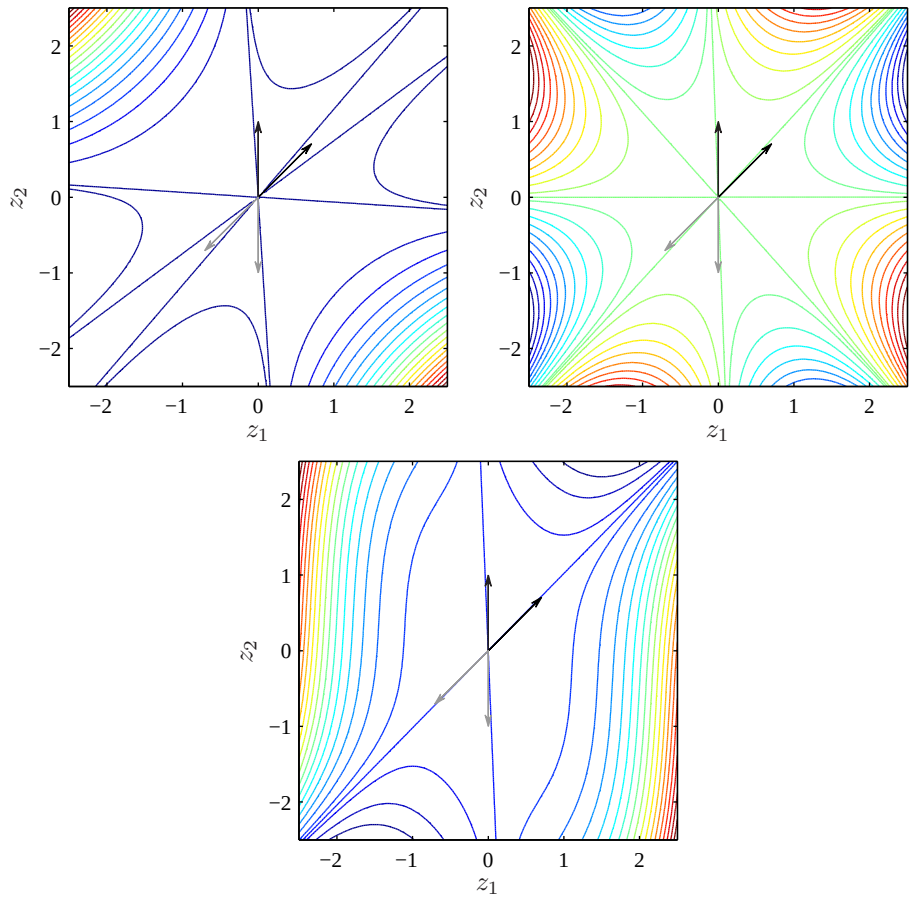

Figure 6.7: *Contour plots of $\hat{f}^4_{2,1}(z_1, z_2)$ (top left), $\hat{f}^4_{2,2}(z_1, z_2)$ (top right) and $\hat{f}^4_{2,3}(z_1, z_2)$ (bottom) for $D = 2$, $S = 2$, and $l = 4$.*

columns of the mixing matrix are defined uniquely and approximately by the intersections of the zero contour levels of the three functions. Two of the four lines defining the zero contour level of each function are approximately in the directions of the columns of the ideal mixing matrix and two lines are 'superfluous', which is an indication of the fact that two degrees of freedom are left unused. These can be exploited for estimating two additional columns, as is demonstrated in the next two examples.

6.3.2.2 Two sensors, three sources and fourth order cumulants

Now let the mixing matrix be given by (6.3.10). According to (6.3.14) the number of linearly independent equations now equals 2. Using the same subspace matrix definition as in the previous example, but now using $N_s = 40000$ time samples for its estimation, we obtain estimates $\hat{f}^4_{2,1}(z_1, z_2)$ and $\hat{f}^4_{2,2}(z_1, z_2)$ of the functions in the system of equations. The contour plots of these functions are depicted in Fig. 6.8. Again, it is clear from the contour plots

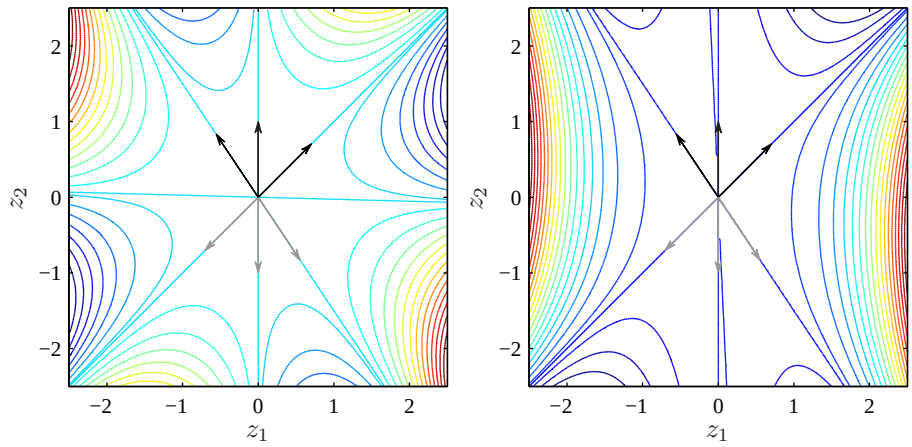

Figure 6.8: *Contour plots of $\hat{f}^4_{2,1}(z_1, z_2)$ (left) and $\hat{f}^4_{2,2}(z_1, z_2)$ (right) for $D = 2$, $S = 3$, and $l = 4$.*

that the three columns of the mixing matrix are defined uniquely and approximately by the intersections of the zero contour levels of the two functions. Three of the four lines defining the zero contour level of each function are approximately in the directions of the columns of the ideal mixing matrix and one line is 'superfluous', which is an indication of the fact that one degree of freedom is left unused. This can be exploited for estimating one more column, as is demonstrated in the next example.

6.3.2.3 Two sensors, four sources and fourth order cumulants

Appending another column to the mixing matrix in the previous example (see (6.3.10)) gives:

$$\mathbf{A} = \begin{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} & \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} & \frac{1}{\sqrt{13}} \begin{bmatrix} -2 \\ 3 \end{bmatrix} & \frac{1}{5} \begin{bmatrix} 4 \\ -3 \end{bmatrix} \end{bmatrix} \approx \begin{bmatrix} 0 & 0.7071 & -0.5547 & 0.8000 \\ 1 & 0.7071 & 0.8321 & -0.6000 \end{bmatrix} . \quad (6.3.15)$$

According to (6.3.14) the number of linearly independent equations now equals 1. Employing the same subspace matrix definition as in the previous two examples, but now using $N_s = 100000$ time samples for its estimation, we obtain an estimate $\hat{f}^4_{2,1}(z_1, z_2)$ of the single function in the system. The contour plot of this function is depicted in Fig. 6.9. Now the columns of the mixing matrix are defined uniquely and approximately by the zero contour level consisting of the four crossing straight lines through the middle of the figure. Note that because there is only one equation no intersections between cones need to be computed in order to estimate the columns.

From the examples presented above, we observe that the more sources are present, the samples have to be used for obtaining a reliable estimate of the subspace matrix, and thus of the sensor cumulants. This is due to the fact that the sensor signals are related much stronger if more source signals are mixed into the same number of observed signals. In general, the more sources are present for a fixed number of sensors, the more samples are required for the reliable estimation of the statistics, and thus also for the reliable estimation of the mixing matrix columns.

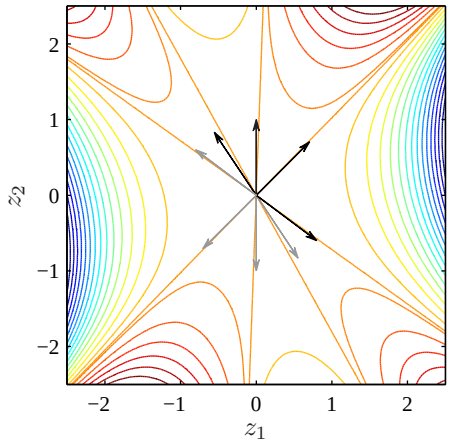

Figure 6.9: *Contour plot of $\hat{f}^4_{2,1}(z_1, z_2)$ for $D = 2$, $S = 4$, and $l = 4$.*

6.3.3 Generalizations and implications for identifiability

It is now straightforward to generalize the results of the examples in Sections 6.3.1 and 6.3.2. We fist consider a MIBI scenario with two sensors and arbitrary l-th order cumulants without conjugations. Then, we generalize the results further to a MIBI scenario with D sensors and finally we give some intuitive examples.

6.3.3.1 Two sensors and arbitrary l-th order cumulants without conjugations

Substituting $D = 2$ and $\mathbf{c}_l = (\circ)_l$ into (6.2.66) and (6.3.1) respectively it follows that the system of equations resulting from Alg. 6.2 is given by:

$$\left\{ f^l_{2,q}(z_1, z_2) = 0 \right\}_{q \in \mathcal{Q}^l_{2,S}} \tag{6.3.16}$$

and that its functions take the following form:

$$f^l_{D,q}(z_1, z_2) \triangleq \sum_{(i_1,\ldots,i_l) \in \mathcal{I}^l_{\mathrm{a},D}} \varphi_q^{i_1 \cdots i_l} \, z_{i_1} \cdots z_{i_l} \qquad \forall \, z_1, z_2 \in \mathbb{C} \,, \; \forall \, q \in \mathcal{Q}^l_{2,S} \,. \tag{6.3.17}$$

The set $\mathcal{Q}^l_{2,S} \triangleq \mathcal{Q}^{(\circ)_l}_{2,S}$ is defined by (6.2.59) and contains the integers $1, \ldots, Q^l_{2,S}$, where the number $Q^l_{2,S} \triangleq Q^{(\circ)_l}_{2,S}$ of linearly independent equations in (6.3.16) again is given by (6.2.60), (6.2.80), and/or (6.2.113). In general, a randomly generated mixing matrix of size $2 \times S$ has rank $\min(2, S)$ with probability one, while its corresponding l-th order Khatri-Rao product has full rank S, i.e. $\operatorname{rank}(\mathbf{A}^l_{\mathrm{u},2,\diamond}) = S$, if $S \leq l + 1$. In this case, according to (6.2.109) the signal subspace dimension $d_2^{x,l} \triangleq d_2^{x,(\circ)_l} = \operatorname{rank}(\mathbf{C}^{x,l}_{\mathrm{u},2}) = \operatorname{rank}(\mathbf{A}^l_{\mathrm{u},2,\diamond})$ also equals S. Substituting those results into (6.2.60), (6.2.80), or (6.2.113), it follows that the number of linearly independent equations in (6.3.16) equals:

$$Q^l_{2,S} = \left\langle {}^{2}_{l} \right\rangle - d_2^{x,l} = \binom{2+l-1}{l} - \operatorname{rank}(\mathbf{A}^l_{\mathrm{u},2,\diamond}) = (l+1) - S \,, \tag{6.3.18}$$

i.e. $\mathcal{Q}^l_{2,S} = \{1, \ldots, l+1-S\}$. It has been shown in [17] that the number of different solutions of an l-homogeneous bivariate polynomial equals l:

Theorem 6.3.1. Fundamental theorem of Algebra - Version 2.
Let $p(z_1, z_2)$ be a non-zero complex homogeneous bivariate polynomial of degree l. Then $p(z_1, z_2)$ has l complex solutions if the solutions are counted with their multiplicity.

For $S = l$ there is still one equation in (6.3.16), which can determine l different solutions. Hence, *with two sensors and l-th order statistics it is possible to determine up to l mixing matrix columns*. In Section 4.5.3 we have presented a method for finding all different solutions of a bivariate homogeneous polynomial equation of arbitrary degree. This implies that those columns can be estimated by means of a closed-form method. Note that the zero contour level of each bivariate l-homogeneous function in (6.3.17) consists of l lines passing through the origin. Also note that Theorem 6.3.1 in fact is just another version of the fundamental theorem of algebra, which states that every non-zero univariate polynomial with complex coefficients has exactly as many complex roots as its degree if repeated roots are counted with their multiplicity.

6.3.3.2 D sensors and arbitrary l-th order cumulants without conjugations

Now we consider a general MIBI scenario with D sensors and l-th order statistics without conjugations. Substituting $\mathbf{c}_l = (\circ)_l$ into (6.2.66) and (6.3.1) respectively it follows that the system of equations resulting from Alg. 6.2 is given by:

$$\Big\{ f^l_{D,q}(z_1, \ldots, z_D) = 0 \Big\}_{q \in \mathcal{Q}^l_{D,S}} \tag{6.3.19}$$

and that its functions take the following form:

$$f^l_{D,q}(z_1, \ldots, z_D) \triangleq \sum_{(i_1,\ldots,i_l) \in \mathcal{I}^l_{\mathrm{a},D}} \varphi_q^{i_1 \cdots i_l} \, z_{i_1} \cdots z_{i_l} \quad \forall \, z_1, \ldots, z_D \in \mathbb{C} \,, \; \forall \, q \in \mathcal{Q}^l_{D,S} \,. \tag{6.3.20}$$

The set $\mathcal{Q}^l_{D,S} \triangleq \mathcal{Q}^{(\circ)_l}_{D,S}$ now contains the integers $1, \ldots, Q^l_{D,S}$, where the number $Q^l_{D,S} \triangleq Q^{(\circ)_l}_{D,S}$ of linearly independent equations in (6.3.3) is given by (6.2.60), (6.2.80), and/or (6.2.113). As we have remarked at the beginning of Section 6.3.1, for scenarios with D sensors and l-th order statistics without conjugations the number of unique rows of the subspace matrix $\mathbf{C}^{x,l}_D$ and l-th order Khatri-Rao product matrix $\mathbf{A}^l_{D,\diamond}$ equals $M^l_{\mathrm{a},D} = \left\langle {D \atop l} \right\rangle$. In general, a randomly generated mixing matrix of size $D \times S$ has rank $\min(D, S)$ with probability one, while its corresponding l-th order Khatri-Rao product has full rank S, i.e. $\mathrm{rank}(\mathbf{A}^l_{\mathrm{u},D,\diamond}) = S$, if $S \leq \left\langle {D \atop l} \right\rangle$. In such a case, according to (6.2.109) the signal subspace dimension $d^{x,l}_D$ also equals S. Substituting those results into (6.2.60), (6.2.80), or (6.2.113), it follows that the number of linearly independent equations in (6.3.19) equals:

$$Q^l_{D,S} = \left\langle {D \atop l} \right\rangle - d^{x,l}_D = \binom{D+l-1}{l} - \mathrm{rank}(\mathbf{A}^l_{\mathrm{u},D,\diamond}) = \frac{(l+D-1)!}{l!(D-1)!} - S \,, \tag{6.3.21}$$

i.e. $\mathcal{Q}^l_{D,S} = \{1, \ldots, \left\langle {D \atop l} \right\rangle - S\}$. Solving the system $\big\{ f^l_{D,q}(\mathbf{z}) = 0 \big\}_{q \in \mathcal{Q}^l_{D,S}}$ consisting of $Q^l_{D,S}$ D-variate l-homogeneous polynomial equations is equivalent to finding the one-dimensional intersections between $Q^l_{D,S}$ 'cones of degree l' in a D-dimensional Euclidian space that represent the zero contour levels of the functions $\big\{ f^l_{D,q}(\mathbf{z}) \big\}_{q \in \mathcal{Q}^l_{D,S}}$ defined in (6.3.20). In general, i.e. in the non-degenerate case, the zero contour level of a D-variate homogenous function is a $(D-1)$-dimensional surface embedded in a D-dimensional Euclidian space.

Intuitively, the higher the 'homogeneity order' l of a homogeneous function, the more degrees of freedom the surface has, e.g. the more turns the surface can make. Generally, the intersection of two $(D-1)$-dimensional surfaces embedded in a D-dimensional Euclidian space has dimension $D-2$, i.e. the dimension drops by one (compared to the dimensions of the intersecting surfaces). Intersecting the resulting $(D-2)$-dimensional surface with a third $(D-1)$-dimensional surface drops the dimension by one again in general. This procedure can be repeated until the dimension of the resulting intersection(s) is one, which is the dimension of the desired solution set because each line through the origin that describes all scalar multiples of a column of the mixing matrix has dimension one. From this reasoning, it is clear that *in a D-dimensional Euclidian space at least $D-1$ surfaces of degree $D-1$ are required in order to define unique one-dimensional intersections*. If less surfaces are used, the dimension of the resulting intersections is larger than one in general. Algebraically, this means that *in general the dimension of the solution set of a system of $D-1$ homogeneous polynomials in D variables has dimension one*. Hence, since at least $D-1$ equations are required to uniquely define the one-dimensional solution sets corresponding to the linear subspaces defined by the columns of the mixing matrix, for the general MIBI case without conjugations we find the constraint $Q^l_{D,S} \geq D-1$. From this constraint it directly follows that (an upper bound on) the maximum number of identifiable mixing matrix columns for the considered scenario equals (see also (6.6.11)):

$$S^l_{\max}(D) = \binom{l+D-1}{l} - (D-1)\,. \tag{6.3.22}$$

See Section 6.6 for a structural approach for determining the maximum number of columns of the mixing matrix $\mathbf{A}$ that can be identified for a given number of sensors D and conjugation tuple $\mathbf{c}_l$ under assumptions *AS1-AS4* on page 299.

6.3.3.3 Examples of identifiability

Finally, we briefly consider some specific examples in order to develop an intuitive feeling for the results above. Firstly, for $D=1$ a cone is defined by any line through the origin, i.e. by any non-zero scalar. Since any two non-zero scalars define the same 'column' of $\mathbf{A}$ at most one source can be identified uniquely, no matter how large the order l is. This also follows from (6.3.22) because $S^l_{\max}(1) = \binom{l}{l} = 1$. Secondly, for $D=2$ a cone is defined by a line through the origin in the two-dimensional Euclidian plane. As we have derived already, with D sensors and l-th order statistics it is possible to determine up to l mixing matrix columns. Again, this also follows directly from (6.3.22) because $S^l_{\max}(2) = \binom{l+1}{l} - 1 = l$. Generally, because increasing the cumulant order l increases the degrees of freedom of the corresponding zero contour level of an l-homogeneous polynomial, the higher the order, the more columns can be identified for the same number of sensors. Contrary to fixing D and varying l, now consider some examples in which l is fixed and D is varied. To start with, let $l=1$. Then, from (6.3.22) it follows that $S^1_{\max}(D) = 1$ for all values of D. Hence, for any number of sensors at most one column can be identified for 'first order statistics', i.e. by using the data directly. This result can also be explained intuitively. For example, consider the case with $D=3$. For $l=1$ the cones in the three-dimensional Euclidian space are planes, and the lines through the origin that correspond to the columns of $\mathbf{A}$ are defined by the one-dimensional intersections between these planes. It can easily be seen that the maximum number of one-dimensional intersections between planes going through the origin, i.e. one-dimensional solution sets that lie in all planes, equals one. Note that if the mixing matrix

were parameterized, as in the case of Direction Of Arrival Estimation (see the next chapter), then it is possible to identify more than one column with 'first order statistics'. Finally, we consider the case with $l = 2$. From (6.3.22) it now follows that $S^2_{\max}(D) = \frac{1}{2}D(D-1)+1$ for all values of D. As an example, again consider the case with $D = 3$. It can easily be seen that the maximum number of one-dimensional intersections between 'conventional' quadric cones defined by quadratic equations in three-dimensional space equals four, which also follows by computing $\frac{1}{2}3(3-1)+1$. This confirms and explains the result obtained in Sections 5.4.1 and 5.4.2.

6.4 Solving the system of equations

In this section, we will outline the generalization of the homotopy method that was presented in Sections 4.5.6 and 5.5, and summarized in Alg. 4.4 on page 202, to the general MIBI problem considered in this chapter. Moreover, we will show that for the scenario with two sensors and the maximum number of identifiable mixing matrix columns/sources, a simple closed-form solution exists.

We consider system (6.3.19), which is obtained for a general MIBI scenario with D sensors and l-th order statistics without conjugations. Again, we start by formulating the appropriate start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ and target system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ for the system of equations under consideration. Similarly to (4.5.32) and (5.5.3), the target system comprises the system defined in (6.3.19) and a unit-norm constraint equation, i.e.:

$$\begin{cases} \left\{ f^l_{D,q}(\mathbf{z}) = 0 \right\}_{q \in \mathcal{Q}^l_{D,S}} ; \\ \qquad c(\mathbf{z}) = 0 \, , \end{cases} \tag{6.4.1}$$

where the unit-norm constraint function $c(\mathbf{z})$ is now defined as $c(\mathbf{z}) \triangleq \sum_{i=1}^{D}(z_i)^2 - 1$. Similarly to (5.5.3), we write this system in matrix-vector notation as follows:

$$\mathbf{p}(\mathbf{z}) = \begin{bmatrix} \boldsymbol{\Phi}_{f,\mathrm{h}} \, \mathbf{w}^l_{\mathrm{u},D}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} = \mathbf{0} \, , \tag{6.4.2}$$

where $\boldsymbol{\Phi}_{f,\mathrm{h}}$ is given by the matrix $\boldsymbol{\Phi}$ defined in (6.2.150). The start system is constructed in such a way that it has exactly the same structure as the target system. Similarly to (6.4.2) it can be expressed as:

$$\mathbf{g}(\mathbf{z}) \triangleq \begin{bmatrix} \boldsymbol{\Phi}_{g,\mathrm{h}} \, \mathbf{w}^l_{\mathrm{u},D}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} = \mathbf{0} \, . \tag{6.4.3}$$

The coefficients of the homogeneous part of the start system can be obtained by computing the left null space of the uniquiefied second order Khatri-Rao product $\mathbf{B}^l_{\mathrm{u},D,\diamond}$ of a known matrix $\mathbf{B}$ (which might be generated at random), i.e. $\boldsymbol{\Phi}^* \eqcirc \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}^l_{\mathrm{u},D,\diamond}\big)$ in the formulation of (6.2.117). The coefficient matrix $\boldsymbol{\Phi}_{h,\mathrm{h}}$ of the homogeneous part $\mathbf{h}_{\mathrm{h}}(\mathbf{z},\lambda)$ of the homotopy $\mathbf{h}(\mathbf{z},\lambda)$ defined in (4.5.25) is given by (5.5.5). Similarly to (5.5.6), because the non-homogeneous part of the start and target systems is not deformed by (4.5.25) it suffices to work with the following homotopy:

$$\mathbf{h}(\mathbf{z},\lambda) \triangleq \begin{bmatrix} \boldsymbol{\Phi}_{h,\mathrm{h}}(\lambda) \, \mathbf{w}^l_{\mathrm{u},D}(\mathbf{z}) \\ c(\mathbf{z}) \end{bmatrix} , \tag{6.4.4}$$

The Jacobian of $\mathbf{h}(\mathbf{z}, \lambda)$ is now given by:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}, \lambda) \triangleq \begin{bmatrix} \mathbf{\Phi}_{h,\mathrm{h}}(\lambda)\, \nabla_{\mathbf{z}}\{\mathbf{w}^{l}_{\mathrm{u},D}(\mathbf{z})\} \\ \nabla_{\mathbf{z}}\{c(\mathbf{z})\} \end{bmatrix} . \tag{6.4.5}$$

The derivative of $\mathbf{h}(\mathbf{z}, \lambda)$ w.r.t. λ is given by (4.5.31) with $\mathbf{g}(\mathbf{z})$ and $\mathbf{p}(\mathbf{z})$ defined in (6.4.3) and (6.4.2) respectively. We now have all ingredients required by Alg. 4.4. Again, the inverse $\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}$ in the sixth step of the algorithm has to be replaced by the pseudo-inverse $\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{\dagger}$. We have performed this algorithm for the examples in Sections 6.3.1.3 and 6.3.2.3, in which the number of sources equals the maximum number of identifiable columns, and found that all columns were identified correctly (i.e. they were in the directions of the lines defining the zero contour levels). For the examples containing more than one equation the solutions can be found by finding the zeros of the gradient of a cost function defined in terms of the functions in the system; see also Sections 4.5.1-4.5.4. Although it is considerably more difficult, the results above can also be generalized to systems of polyconjugal equations. This is evident already from the fact that we can split any polyconjugal equation into a real and an imaginary part. A homotopy method can then be designed that finds the real and imaginary parts of the different solutions.

As we have seen, for scenarios with two sensors in which the number of sources equals the maximum number of identifiable columns there is only one homogeneous polyconjugal equation in the system. Hence, for scenarios without conjugations such a MIBI problem comes down to finding the solutions of a single bivariate homogeneous equation. In Section 4.5.3 we have shown that a single bivariate homogeneous polynomial equation of arbitrary degree can easily be rewritten as a univariate non-homogeneous polynomial of the same degree. Thus, its solutions can be found by using a standard algorithm for finding the roots of a univariate polynomial equation. This yields a closed-form method for solving the MIBI problem with two sensors, l-th order statistics without conjugations, and $S = l$ sources. Using this method for the examples in Sections 6.3.1.3 and 6.3.2.3 yielded the same solutions as the homotopy method described above.

6.5 Writing MIBI as a Multi-Matrix Generalized Eigenvalue Decomposition problem

In this section, we generalize the results from Sections 4.6 and 5.6 and project the MIBI problem onto a *Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD)*. Again, this formulation is dual to the problem description in terms of a system of homogeneous polyconjugal equations because the derivation of system (6.2.66) was primarily based on (6.2.121), whereas the derivation we are about to present is primarily based on the complementary relation (6.2.120). More precisely, under the assumption that the source auto-cumulant matrix has full rank, system (6.2.66) was derived directly from the fact that the left null space of the (uniquiefied) subspace matrix equals the left null space of the (uniquiefied) Khatri-Rao product with conjugation tuple $\mathbf{c}_l$ of the mixing matrix, which was proven in Section 6.2.6.6. Likewise, in the sequel of this section we will derive the dual MMGEVD problem from the fact that the column range of the (uniquiefied) subspace matrix equals the column range of the (uniquiefied) Khatri-Rao product, which was also proven in Section 6.2.6.6. For convenience, and *without any loss of generality, the results in this section are derived from the 'full subspace matrix'* $\mathbf{C}_{D}^{x,\mathbf{c}_l}$ *instead of the uniquiefied matrix* $\mathbf{C}_{\mathrm{u},D}^{x,c_1c_2}$.

6.5.1 MGEVD structure

To start with, from the 'full versions' of Theorems 6.2.4, 6.2.6, and 6.2.8 it follows that:

$$\check{\mathbf{a}}_D^{j,\mathbf{c}_l} = \check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{a}^j) \in \mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_D^{x,\mathbf{c}_l}\right) = \mathcal{L}_{\mathrm{c}}\big(\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right) , \tag{6.5.1}$$

where $\check{\mathbf{a}}_D^{j,\mathbf{c}_l}$ and $\check{\mathbf{w}}_D^{\mathbf{c}_l}(\cdot)$ are defined in (6.2.21) and (6.2.22) respectively. Note that now the matrix $\mathbf{U}^s$ contains the left singular vectors of the 'full subspace matrix' $\mathbf{C}_D^{x,\mathbf{c}_l}$ and that the 'full versions' of (6.2.132) and (6.2.144) have been used:

$$\mathcal{R}_{\mathrm{c}}\big(\mathbf{C}_D^{x,\mathbf{c}_l}\big) = \mathcal{L}_c\big(\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big) = \mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^s\big) = \mathcal{L}\left(\mathcal{U}^s\right) .$$

Eq. (6.5.1) implies that $\check{\mathbf{a}}_D^{j,\mathbf{c}_l}$ can be written as a linear combination of the columns of $\mathbf{U}^s$:

$$\check{\mathbf{a}}_D^{j,\mathbf{c}_l} = \check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{a}^j) = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p^j\, \mathbf{u}^p = \mathbf{U}^s\, \boldsymbol{\mu}^j \qquad \forall\, 1 \leq j \leq S\,, \tag{6.5.2}$$

where $\boldsymbol{\mu}^j \triangleq \begin{bmatrix} \mu_1^j & \cdots & \mu_{d_D^{x,\mathbf{c}_l}}^j \end{bmatrix}^T$ with $\mu_p^j \in \mathbb{R}$ for $1 \leq p \leq d_D^{x,\mathbf{c}_l}$ and $1 \leq j \leq S$. Thus, the MIBI problem can be projected onto the problem of finding the solutions for the unknown vector $\mathbf{z}$ that satisfy the following equation:

$$\boxed{\check{\mathbf{z}}_D^{\mathbf{c}_l} = \check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{z}) \triangleq (\mathbf{z})^{c_1} \otimes \cdots \otimes (\mathbf{z})^{c_l} = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p\, \mathbf{u}^p = \mathbf{U}^S\, \boldsymbol{\mu}\,,} \tag{6.5.3}$$

where to each solution $\mathbf{z}$ there corresponds a different vector $\boldsymbol{\mu}$ of coefficients with $\boldsymbol{\mu}$ defined in (5.6.9). The latter two equations say that the problem of finding the array response vectors is equivalent to finding vectors $\mathbf{z}$ such that the structured vectors $\check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{z}) = \check{\mathbf{z}}_D^{\mathbf{c}_l}$ belong to the column range $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big)$ of $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}$, which can be computed from the known matrix $\mathbf{C}_D^{x,\mathbf{c}_l}$ because $\mathcal{R}_{\mathrm{c}}\big(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big)$ equals $\mathcal{R}_{\mathrm{c}}\left(\mathbf{C}_D^{x,\mathbf{c}_l}\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^s\right)$. In order to find the solutions for $\mathbf{z}$, or equivalently the solutions for $\boldsymbol{\mu}$, we exploit the specific structure of $\check{\mathbf{z}}_D^{\mathbf{c}_l} = \check{\mathbf{w}}_D^{\mathbf{c}_l}(\mathbf{z})$ given by (6.2.22). Firstly, from (6.2.22) and (6.5.3) it follows directly that each product of the form $\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l} \triangleq (z_{i_1})^{c_1} \cdots (z_{i_l})^{c_l}$ can be expressed as follows:

$$\boxed{\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l} \;=\; \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p\, u_{\mathbf{i}_l}^p \;=\; \begin{bmatrix} u_{\mathbf{i}_l}^1 & \cdots & u_{\mathbf{i}_l}^{d_D^{x,\mathbf{c}_l}} \end{bmatrix} \boldsymbol{\mu} \;\triangleq\; \tilde{\mathbf{g}}_{\mathbf{i}_l}\, \boldsymbol{\mu} \qquad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}\,.} \tag{6.5.4}$$

Based on this equation, we can derive several systems of equations that are similar to (5.6.6).

To start with, similarly to (5.6.6) we derive the following systems of equations from (6.5.4) by varying the last index i_l of the tuple $\mathbf{i}_l$ from 1 till D:

$$(z_{i_1})^{c_1} \cdots (z_{i_{l-1}})^{c_{l-1}} (\mathbf{z})^{c_l} = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p \begin{bmatrix} u_{i_1 \cdots i_{l-1} 1}^p \\ \vdots \\ u_{i_1 \cdots i_{l-1} D}^p \end{bmatrix} = \mathbf{G}_{\mathbf{i}_{l-1}}\, \boldsymbol{\mu} \quad \forall\, \mathbf{i}_{l-1} \in \mathcal{I}_{\mathrm{t},D}^{l-1}\,, \tag{6.5.5}$$

where:

$$\mathbf{G}_{\mathbf{i}_{l-1}} \triangleq \begin{bmatrix} u_{i_1 \cdots i_{l-1} 1}^1 & \cdots & u_{i_1 \cdots i_{l-1} 1}^{d_D^{x,\mathbf{c}_l}} \\ \vdots & \ddots & \vdots \\ u_{i_1 \cdots i_{l-1} D}^1 & \cdots & u_{i_1 \cdots i_{l-1} D}^{d_D^{x,\mathbf{c}_l}} \end{bmatrix} = \begin{bmatrix} \mathbf{u}_{\mathbf{i}_{l-1}}^1 & \cdots & \mathbf{u}_{\mathbf{i}_{l-1}}^{d_D^{x,\mathbf{c}_l}} \end{bmatrix} \in \mathbb{C}_D^{d_D^{x,\mathbf{c}_l}} \quad \forall\, \mathbf{i}_{l-1} \in \mathcal{I}_{\mathrm{t},D}^{l-1}\,, \tag{6.5.6}$$

and:

$$\mathbf{u}^{p}_{\mathbf{i}_{l-1}} \triangleq \begin{bmatrix} u^{p}_{i_1 \cdots i_{l-1} 1} \\ \vdots \\ u^{p}_{i_1 \cdots i_{l-1} D} \end{bmatrix} \in \mathbb{C}_D \qquad \forall\, 1 \leq p \leq d_D^{x,\mathbf{c}_l} \,. \tag{6.5.7}$$

Note that there are $(D)^{l-1}$ matrices $\mathbf{G}_{\mathbf{i}_{l-1}}$ because all indices $i_1, \ldots, i_{l-1}$ ranges from 1 till D. In addition notice that $\mathbf{G}_{\mathbf{i}_{l-1}}$ is the '$\mathbf{i}_{l-1}$-th' submatrix of size $D \times d_D^{x,\mathbf{c}_l}$ of $\mathbf{U}^s$:

$$\mathbf{U}^s = \begin{bmatrix} \mathbf{G}_{(1)_{l-1}} \\ \vdots \\ \mathbf{G}_{(D)_{l-1}} \end{bmatrix} .$$

For an example with $l = 2$ see (5.6.10) on page 291. With the definitions made above, we can write (6.5.5) concisely as follows:

$$\tilde{z}^{\mathbf{c}_{l-1}}_{\mathbf{i}_{l-1}} \cdot (\mathbf{z})^{c_l} = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p \, \mathbf{u}^{p}_{\mathbf{i}_{l-1}} = \mathbf{G}_{\mathbf{i}_{l-1}} \boldsymbol{\mu} \qquad \forall\, \mathbf{i}_{l-1} \in \mathcal{I}^{l-1}_{\mathrm{t},D} \,. \tag{6.5.8}$$

Instead of i_l we could also have varied another index, but this does not make an essential difference. However, we can obtain different systems with different sizes by varying more than one index. As can directly be derived from (6.5.3) and/or (6.5.4), in general we have for each $0 \leq m \leq l$:

$$\boxed{\tilde{z}^{\mathbf{c}_{l-m}}_{\mathbf{i}_{l-m}} \cdot \left((\mathbf{z})^{c_{l-m+1}} \otimes \cdots \otimes (\mathbf{z})^{c_l} \right) = \sum_{p=1}^{d_D^{x,\mathbf{c}_l}} \mu_p \, \mathbf{u}^{p}_{\mathbf{i}_{l-m}} = \mathbf{G}_{\mathbf{i}_{l-m}} \boldsymbol{\mu} \qquad \forall\, \mathbf{i}_{l-m} \in \mathcal{I}^{l-m}_{\mathrm{t},D} \,.} \tag{6.5.9}$$

where:

$$\mathbf{G}_{\mathbf{i}_{l-m}} \triangleq \begin{bmatrix} \mathbf{u}^{1}_{\mathbf{i}_{l-m}} & \cdots & \mathbf{u}^{d_D^{x,\mathbf{c}_l}}_{\mathbf{i}_{l-m}} \end{bmatrix} \in \mathbb{C}^{d_D^{x,\mathbf{c}_l}}_{(D)^m} \qquad \forall\, \mathbf{i}_{l-m} \in \mathcal{I}^{l-m}_{\mathrm{t},D} \tag{6.5.10}$$

and:

$$\mathbf{u}^{p}_{\mathbf{i}_{l-m}} \triangleq \begin{bmatrix} u^{p}_{i_1 \cdots i_{l-m} (1)_m} \\ \vdots \\ u^{p}_{i_1 \cdots i_{l-m} (D)_m} \end{bmatrix} \in \mathbb{C}_{(D)^m} \qquad \forall\, 1 \leq p \leq d_D^{x,\mathbf{c}_l} . \tag{6.5.11}$$

See (6.5.8) for an example with $m = 1$, and (6.5.15) through (6.5.20) for an example with $l = 3$ and $0 \leq m \leq 3$. There are $(D)^{l-m}$ matrices $\mathbf{G}_{\mathbf{i}_{l-m}}$ of size $(D)^m \times d_D^{x,\mathbf{c}_l}$ in system (6.5.9). Stacking these matrices on top of each other in the right order yields $\mathbf{U}^s$:

$$\mathbf{U}^s = \underbrace{\begin{bmatrix} \tilde{\mathbf{g}}_{(1)_l} \\ \vdots \\ \tilde{\mathbf{g}}_{(D)_l} \end{bmatrix}}_{m=0} = \underbrace{\begin{bmatrix} \mathbf{G}_{(1)_{l-1}} \\ \vdots \\ \mathbf{G}_{(D)_{l-1}} \end{bmatrix}}_{m=1} = \cdots = \underbrace{\begin{bmatrix} \mathbf{G}_{11} \\ \vdots \\ \mathbf{G}_{DD} \end{bmatrix}}_{m=l-2} = \underbrace{\begin{bmatrix} \mathbf{G}_{1} \\ \vdots \\ \mathbf{G}_{D} \end{bmatrix}}_{m=l-1} = \underbrace{\mathbf{G}}_{m=l} . \tag{6.5.12}$$

Although (6.5.9) defines a different (MM)GEVD problem for each $0 \leq m \leq l$, these problems are equivalent. Thus, if we can solve one of them, we can solve them all. Therefore, symbolically we write:

$$\boxed{\begin{array}{c} \mathrm{gevd}\left(\tilde{\mathbf{g}}_{(1)_l}, \ldots, \tilde{\mathbf{g}}_{(D)_l}\right) \equiv \mathrm{gevd}\left(\mathbf{G}_{(1)_{l-1}}, \ldots, \mathbf{G}_{(D)_{l-1}}\right) \equiv \cdots \\ \equiv \mathrm{gevd}\left(\mathbf{G}_{11}, \ldots, \mathbf{G}_{DD}\right) \equiv \mathrm{gevd}\left(\mathbf{G}_{1}, \ldots, \mathbf{G}_{D}\right) \equiv \mathrm{gevd}\left(\mathbf{G}\right) . \end{array}} \tag{6.5.13}$$

Note that for each value of m system (6.5.9) contains $(D)^l$ (non-unique) equations. Also note that a whole range of equivalent descriptions follow from (6.5.9) by varying m from 0 till l. At one end of the spectrum, for $m = 0$ we obtain expression (6.5.4), whereas at the other end of the spectrum for $m = l$ we obtain expression (6.5.3). For $0 < m < l$ we obtain expressions that are in between these two.

6.5.2 Example with $D = 2$, $S = 3$, $l = 3$, and $\mathbf{c}_3 = (\circ, \circ, \circ)$

Before we examine the structure of the systems in (6.5.9) in more detail, we consider an example with $D = 2$, $S = 3$, $l = 3$, and $\mathbf{c}_3 = (\circ)_3 = (\circ, \circ, \circ)$, i.e. there are no conjugations. Furthermore, let $\mathbf{A}$ be a mixing matrix whose third order Khatri-Rao product has rank 3. Then, the dimension of the signal subspace equals $d_2^{x,(\circ)_3} \triangleq d_2^{x,3} = \text{rank}\left(\mathbf{A}_{\text{u},2,\diamond}^{(\circ)_3}\right) = \text{rank}\left(\mathbf{A}_{2,\diamond}^{(\circ)_3}\right) = \text{rank}\left(\mathbf{A}_{2,\diamond}^{3}\right) = 3$ and the signal subspace matrix $\mathbf{U}^s$ can be written as:

$$\mathbf{U}^s = \begin{bmatrix} u^1_{111} & u^2_{111} & u^3_{111} \\ u^1_{112} & u^2_{112} & u^3_{112} \\ u^1_{121} & u^2_{121} & u^3_{121} \\ u^1_{122} & u^2_{122} & u^3_{122} \\ u^1_{211} & u^2_{211} & u^3_{211} \\ u^1_{212} & u^2_{212} & u^3_{212} \\ u^1_{221} & u^2_{221} & u^3_{221} \\ u^1_{222} & u^2_{222} & u^3_{222} \end{bmatrix} . \tag{6.5.14}$$

Now we will write (6.5.9) explicitly for $0 \leq m \leq 3$. To start with, writing out (6.5.9) for $l = 3$ and $m = 0$ yields:

$$\begin{aligned} z_{i_1} z_{i_2} z_{i_3} &= \sum_{p=1}^{3} \mu_p \, u^p_{i_1 i_2 i_3} = \tilde{\mathbf{g}}_{i_1 i_2 i_3} \, \boldsymbol{\mu} \\ &= \mu_1 \, u^1_{i_1 i_2 i_3} + \mu_2 \, u^2_{i_1 i_2 i_3} + \mu_3 \, u^3_{i_1 i_2 i_3} = \begin{bmatrix} u^1_{i_1 i_2 i_3} & u^2_{i_1 i_2 i_3} & u^3_{i_1 i_2 i_3} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} \\ &\forall \, 1 \leq i_1, i_2, i_3 \leq 2 \, , \end{aligned} \tag{6.5.15}$$

which equals (6.5.4) with $l = 3$. Note that the matrices $\mathbf{G}_{\mathbf{i}_l}$ have degenerated to row vectors (signifying the fact that the type of expression specified in (6.5.15) is at one end of the spectrum) and are given by:

$$\tilde{\mathbf{g}}_{i_1 i_2 i_3} = \begin{bmatrix} u^1_{i_1 i_2 i_3} & u^2_{i_1 i_2 i_3} & u^3_{i_1 i_2 i_3} \end{bmatrix} \qquad \forall \, 1 \leq i_1, i_2, i_3 \leq 2 \, . \tag{6.5.16}$$

Because $1 \leq i_1, i_2, i_3 \leq 2$, there are $(D)^{l-m} = (2)^3 = 8$ expressions of the form (6.5.15), of which $M^3_{\text{u},2} = 4$ are different (6.2.33); see also (6.2.27). Next, writing out (6.5.9) for $l = 3$ and $m = 1$ yields:

$$\begin{aligned} z_{i_1} z_{i_2} \, \mathbf{z} = z_{i_1} z_{i_2} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} &= \sum_{p=1}^{3} \mu_p \, \mathbf{u}^p_{i_1 i_2} = \mathbf{G}_{i_1 i_2} \, \boldsymbol{\mu} \\ &= \mu_1 \begin{bmatrix} u^1_{i_1 i_2 1} \\ u^1_{i_1 i_2 2} \end{bmatrix} + \mu_2 \begin{bmatrix} u^2_{i_1 i_2 1} \\ u^2_{i_1 i_2 2} \end{bmatrix} + \mu_3 \begin{bmatrix} u^3_{i_1 i_2 1} \\ u^3_{i_1 i_2 2} \end{bmatrix} = \begin{bmatrix} u^1_{i_1 i_2 1} & u^2_{i_1 i_2 1} & u^3_{i_1 i_2 1} \\ u^1_{i_1 i_2 2} & u^2_{i_1 i_2 2} & u^3_{i_1 i_2 2} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} \\ &\forall \, 1 \leq i_1, i_2 \leq 2 \, , \end{aligned} \tag{6.5.17}$$

which equals (6.5.8) with $l = 3$. Because $1 \leq i_1, i_2 \leq 2$, there are $(2)^2 = 4$ expressions of the form (6.5.17), of which $M^2_{\mathrm{u},2} = \frac{(2+2-1)!}{2!(2-1)!} = 3$ are different (6.2.33). Writing out (6.5.9) for $l = 3$ and $m = 2$ yields:

$$z_i\,\mathbf{z}\otimes\mathbf{z} = z_i \begin{bmatrix} z_1z_1 \\ z_1z_2 \\ z_2z_1 \\ z_2z_2 \end{bmatrix} = \sum_{p=1}^{3} \mu_p\,\mathbf{u}_i^p = \mathbf{G}_i\,\boldsymbol{\mu}$$

$$= \mu_1 \begin{bmatrix} u^1_{i11} \\ u^1_{i12} \\ u^1_{i21} \\ u^1_{i22} \end{bmatrix} + \mu_2 \begin{bmatrix} u^2_{i11} \\ u^2_{i12} \\ u^2_{i21} \\ u^2_{i22} \end{bmatrix} + \mu_3 \begin{bmatrix} u^3_{i11} \\ u^3_{i12} \\ u^3_{i21} \\ u^3_{i22} \end{bmatrix} = \begin{bmatrix} u^1_{i11} & u^2_{i11} & u^3_{i11} \\ u^1_{i12} & u^2_{i12} & u^3_{i12} \\ u^1_{i21} & u^2_{i21} & u^3_{i21} \\ u^1_{i22} & u^2_{i22} & u^3_{i22} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix},\ i = 1, 2\,. \quad (6.5.18)$$

Because $1 \leq i \leq 2$, there are $(2)^1 = 2$ different expressions of the form (6.5.18). Finally, writing out (6.5.9) for $l = 3$ and $m = 3$ (or (6.5.3) for $l = 3$) yields:

$$\mathbf{z}\otimes\mathbf{z}\otimes\mathbf{z} = \begin{bmatrix} z_1z_1z_1 \\ z_1z_1z_2 \\ z_1z_2z_1 \\ z_1z_2z_2 \\ z_2z_1z_1 \\ z_2z_1z_2 \\ z_2z_2z_1 \\ z_2z_2z_2 \end{bmatrix} = \sum_{p=1}^{3} \mu_p\,\mathbf{u}^p_{i_1i_2i_3} = \mathbf{G}\,\boldsymbol{\mu} = \mathbf{U}^S\,\boldsymbol{\mu}$$

$$= \mu_1 \begin{bmatrix} u^1_{111} \\ u^1_{112} \\ u^1_{121} \\ u^1_{122} \\ u^1_{211} \\ u^1_{212} \\ u^1_{221} \\ u^1_{222} \end{bmatrix} + \mu_2 \begin{bmatrix} u^2_{111} \\ u^2_{112} \\ u^2_{121} \\ u^2_{122} \\ u^2_{211} \\ u^2_{212} \\ u^2_{221} \\ u^2_{222} \end{bmatrix} + \mu_3 \begin{bmatrix} u^3_{111} \\ u^3_{112} \\ u^3_{121} \\ u^3_{122} \\ u^3_{211} \\ u^3_{212} \\ u^3_{221} \\ u^3_{222} \end{bmatrix} = \begin{bmatrix} u^1_{111} & u^2_{111} & u^3_{111} \\ u^1_{112} & u^2_{112} & u^3_{112} \\ u^1_{121} & u^2_{121} & u^3_{121} \\ u^1_{122} & u^2_{122} & u^3_{122} \\ u^1_{211} & u^2_{211} & u^3_{211} \\ u^1_{212} & u^2_{212} & u^3_{212} \\ u^1_{221} & u^2_{221} & u^3_{221} \\ u^1_{222} & u^2_{222} & u^3_{222} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}. \quad (6.5.19)$$

Hence, for this example decomposition (6.5.12) of $\mathbf{U}^s$ becomes:

$$\mathbf{U}^s = \begin{bmatrix} \tilde{\mathbf{g}}_{111} \\ \tilde{\mathbf{g}}_{112} \\ \tilde{\mathbf{g}}_{121} \\ \tilde{\mathbf{g}}_{122} \\ \tilde{\mathbf{g}}_{211} \\ \tilde{\mathbf{g}}_{212} \\ \tilde{\mathbf{g}}_{221} \\ \tilde{\mathbf{g}}_{222} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{11} \\ \mathbf{G}_{12} \\ \mathbf{G}_{21} \\ \mathbf{G}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \end{bmatrix} = \mathbf{G}\,. \quad (6.5.20)$$

Although (6.5.15) through (6.5.18) define different (MM)GEVD problems, clearly they are equivalent. Thus, if we can solve one of them, we can solve them all. Hence, symbolically

we can write (see (6.5.13)):

$$\begin{aligned}&\mathrm{gevd}\left(\tilde{\mathbf{g}}_{111},\tilde{\mathbf{g}}_{112},\tilde{\mathbf{g}}_{121},\tilde{\mathbf{g}}_{122},\tilde{\mathbf{g}}_{211},\tilde{\mathbf{g}}_{212},\tilde{\mathbf{g}}_{221},\tilde{\mathbf{g}}_{222}\right)\equiv\mathrm{gevd}\left(\mathbf{G}_{11},\mathbf{G}_{12},\mathbf{G}_{21},\mathbf{G}_{22}\right)\\&\equiv\mathrm{gevd}\left(\mathbf{G}_1,\mathbf{G}_2\right)\equiv\mathrm{gevd}\left(\mathbf{G}\right).\end{aligned}\tag{6.5.21}$$

For this example we can find the solutions for $\boldsymbol{\mu}$ by means of a standard GEVD algorithm for computing $\mathrm{gevd}\left(\mathbf{G}_1,\mathbf{G}_2\right)$ defined by (6.5.18). The matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ have size 4×3, but can be reduced to the square matrices $\mathbf{G}_{\mathrm{r},1}$ and $\mathbf{G}_{\mathrm{r},2}$ of size 3×3 because one of the rows of $\mathbf{G}_1$ and $\mathbf{G}_2$ is redundant and can be eliminated, viz. the second or the third. Eliminating the third rows for example, the correct solutions of $\boldsymbol{\mu}$ can be found by computing the GEVD of $\mathbf{G}_{\mathrm{r},1}$ and $\mathbf{G}_{\mathrm{r},2}$ defined as follows:

$$\mathbf{G}_{\mathrm{r},1}\triangleq\begin{bmatrix}u^1_{111}&u^2_{i11}&u^3_{111}\\u^1_{112}&u^2_{i12}&u^3_{112}\\u^1_{122}&u^2_{i22}&u^3_{122}\end{bmatrix}\qquad\text{and}\qquad\mathbf{G}_{\mathrm{r},2}\triangleq\begin{bmatrix}u^1_{211}&u^2_{i11}&u^3_{211}\\u^1_{212}&u^2_{i12}&u^3_{212}\\u^1_{222}&u^2_{i22}&u^3_{222}\end{bmatrix}.\tag{6.5.22}$$

6.5.3 Generalized matrix pencil

In the previous sections we have shown that the MIBI problem can be projected onto the problem of solving different equivalent MMGEVD problems. In this section, we examine the MMGEVD section in more detail and define a *generalized matrix pencil*. In practice, the only known quantities are the matrices $\mathbf{G}_{\mathbf{i}_{l-m}}$ with $\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}$ occurring in (6.5.9), which can be computed from the SVD of the subspace matrix. Our goal in solving the MMGEVD problems is to find S different vectors $\boldsymbol{\mu}$ that satisfy (6.5.9), say $\boldsymbol{\mu}^1$, …, $\boldsymbol{\mu}^S$. Each solution vector $\boldsymbol{\mu}^j$ corresponds to an array response vector, say $\mathbf{z}^j$. We will show later in this section how the array response vectors $\mathbf{z}^1$, …, $\mathbf{z}^S$ corresponding to $\boldsymbol{\mu}^1$, …, $\boldsymbol{\mu}^S$ respectively can also be determined from (6.5.9) once the latter vectors are known.

From (6.5.9) the MMGEVD structure follows immediately from the fact that all matrix-vector products $\mathbf{G}_{\mathbf{i}_{l-m}}\boldsymbol{\mu}$ with $\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}$ yield a vector in the same direction, viz. $(\mathbf{z})^{c_{l-m+1}}\otimes\cdots\otimes(\mathbf{z})^{c_l}$. Similarly to (5.6.12) in Section 5.6, the MMGEVD structure also follows directly by assuming that $\check{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}\neq 0$ and writing (6.5.9) as follows:

$$(\mathbf{z})^{c_{l-m+1}}\otimes\cdots\otimes(\mathbf{z})^{c_l}=\frac{1}{\check{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}}\mathbf{G}_{\mathbf{i}_{l-m}}\boldsymbol{\mu}\qquad\forall\,\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}.\tag{6.5.23}$$

Hence, the vector $\boldsymbol{\mu}$ is a generalized eigenvector of the matrices $\mathbf{G}_{\mathbf{i}_{l-m}}$ with $\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}$. For each pair of length-$(l-m)$ index tuples $(\mathbf{i}_{l-m},\mathbf{j}_{l-m})$ a standard GEVD problem exists:

$$\frac{1}{\check{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}}\mathbf{G}_{\mathbf{i}_{l-m}}\boldsymbol{\mu}=\frac{1}{\check{z}_{\mathbf{j}_{l-m}}^{\mathbf{c}_{l-m}}}\mathbf{G}_{\mathbf{j}_{l-m}}\boldsymbol{\mu}\qquad\forall\,\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m},\quad\forall\,\mathbf{j}_{l-m}\in\mathcal{J}_{\mathrm{t},D}^{l-m}.\tag{6.5.24}$$

That is, for each pair $(\mathbf{i}_{l-m},\mathbf{j}_{l-m})$ the vector $\boldsymbol{\mu}$ is a generalized eigenvector of the matrix pencil $\mathbf{G}_{\mathbf{i}_{l-m}}-\lambda\mathbf{G}_{\mathbf{j}_{l-m}}$ with generalized eigenvalue $\lambda=\check{z}_{\mathbf{j}_{l-m}}^{\mathbf{c}_{l-m}}/\check{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}$. The intersections of the generalized eigenspaces of the GEVD's of all possible matrix pencils defined by (6.5.24) give S possible solutions for the vector $\boldsymbol{\mu}$, each of which corresponds to a column of the mixing matrix. Hence, $\boldsymbol{\mu}$ is a generalized eigenvector for all pairs $(\mathbf{i}_{l-m},\mathbf{j}_{l-m})$. Similarly to (6.5.24), we can consider MMGEVD problems consisting of an arbitrary number of matrices. As in Section 5.6, we do not need to make assumptions like $\check{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}\neq 0$, and a much more natural description of the problem is obtained by multiplying (6.5.23) by the product of the

terms occurring in the denominators pre-multiplying the matrix-vector products at the right hand side of the equation:

$$\left(\prod_{\mathbf{j}_{l-m} \in \mathcal{J}_{\mathrm{t},D}^{l-m}} \check{z}_{\mathbf{j}_{l-m}}^{\mathbf{c}_{l-m}} \right) \cdot \left((\mathbf{z})^{c_{l-m+1}} \otimes \cdots \otimes (\mathbf{z})^{c_l} \right)$$

$$= \left(\prod_{\substack{\mathbf{j}_{l-m} \in \mathcal{J}_{\mathrm{t},D}^{l-m} \\ \mathbf{j}_{l-m} \neq \mathbf{i}_{l-m}}} \check{z}_{\mathbf{j}_{l-m}}^{\mathbf{c}_{l-m}} \right) \cdot \mathbf{G}_{\mathbf{i}_{l-m}} \boldsymbol{\mu} \qquad \forall\, \mathbf{i}_{l-m} \in \mathcal{I}_{\mathrm{t},D}^{l-m}, \quad \forall\, 0 \leq m \leq l\,. \tag{6.5.25}$$

Equivalently, we can multiply (6.5.9) by the term between parentheses at the right hand side of (6.5.25). Using formulation (6.5.25) the problem comes down to finding all generalized eigenvectors and *eigenvalue tuples* of the *generalized matrix pencil* that is defined similarly to (5.6.16) as follows:

$$\boxed{\mathbf{G}_m^l(\boldsymbol{\lambda}_m^l) \triangleq \sum_{\mathbf{i}_{l-m} \in \mathcal{I}_{\mathrm{t},D}^{l-m}} \lambda_{\mathbf{i}_{l-m}} \mathbf{G}_{\mathbf{i}_{l-m}} \qquad 0 \leq m \leq l\,,} \tag{6.5.26}$$

where the generalized eigenvalue tuple $\boldsymbol{\lambda}_m^l$ is defined by:

$$\boldsymbol{\lambda}_m^l \triangleq \left(\lambda_{\mathbf{i}_{l-m,1}}, \ldots, \lambda_{\mathbf{i}_{l-m,(D)^{l-m}}} \right), \tag{6.5.27}$$

Note that the summation runs over $(D)^{l-m}$ matrices and that an ideal solution for $\boldsymbol{\lambda}_m^l$ is given by (6.5.27) with the individual generalized eigenvalues defined as follows:

$$\lambda_{\mathbf{i}_{l-m}} = \prod_{\substack{\mathbf{j}_{l-m} \in \mathcal{J}_{\mathrm{t},D}^{l-m} \\ \mathbf{j}_{l-m} \neq \mathbf{i}_{l-m}}} \check{z}_{\mathbf{j}_{l-m}}^{\mathbf{c}_{l-m}} \qquad \forall\, \mathbf{i}_{l-m} \in \mathcal{I}_{\mathrm{t},D}^{l-m}\,. \tag{6.5.28}$$

Summarizing, we have shown that finding all generalized eigenvectors and eigenvalue tuples of the generalized matrix pencil defined in (6.5.26) means solving a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem. This is a well-structured mathematical problem that to our knowledge has not yet been formulated in this general form. We formulate it explicitly as follows. Assume that the generalized eigenvectors and eigenvalues resulting from the MMGEVD are stored in the matrix $\mathbf{M}_{G_m^l}$ and set $\Lambda_{G_m^l}$ of generalized eigenvalue tuples respectively, then we write:

$$[\mathbf{M}_{G_m^l}, \Lambda_{G_m^l}] = \mathrm{gevd}\left(\left\{ \mathbf{G}_{\mathbf{i}_{l-m}} \right\}_{\mathbf{i}_{l-m} \in \mathcal{I}_{\mathrm{t},D}^{l-m}} \right) \qquad 0 \leq m \leq l\,. \tag{6.5.29}$$

This system has $d_D^{x,\mathbf{c}_l}$ generalized eigenvectors $\boldsymbol{\mu}$ and eigenvalue tuples $\boldsymbol{\lambda}_m^l$, i.e. the generalized eigenvector matrix $\mathbf{M}_{G_m^l}$ and generalized eigenvalue tuple set $\Lambda_{G_m^l}$ can be written as follows:

$$\mathbf{M}_{G_m^l} = \left[\boldsymbol{\mu}^1 \;\; \cdots \;\; \boldsymbol{\mu}^{d_D^{x,\mathbf{c}_l}} \right] \qquad \text{and} \quad \Lambda_{G_m^l} = \left\{ \boldsymbol{\lambda}_{m,1}^l, \ldots, \boldsymbol{\lambda}_{m,d_D^{x,\mathbf{c}_l}}^l \right\} \tag{6.5.30}$$

respectively. After having computed this decomposition, which may be a difficult mathematical problem in itself, the columns of $\mathbf{A}$ can be recovered straightforwardly from the generalized eigenvectors, see (6.5.33).

Observe that $\boldsymbol{\mu}$ also is a generalized eigenvector of all linear combinations of the matrices $\left\{\mathbf{G}_{\mathbf{i}_{l-m}}\right\}_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}}$. This can easily be proven as follows. Assume that $\boldsymbol{\mu}$ is a solution of (6.5.9) and for some index $p\in\mathbb{N}^{+}$ define the arbitrary scalar coefficients $\{\xi_p^{\mathbf{i}_{l-m}}\}_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}}$ as the coefficients of the linear combination $\boldsymbol{\Gamma}_p$ of the matrices $\left\{\mathbf{G}_{\mathbf{i}_{l-m}}\right\}_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}}$, i.e.:

$$\boldsymbol{\Gamma}_p \triangleq \sum_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}} \xi_p^{\mathbf{i}_{l-m}}\mathbf{G}_{\mathbf{i}_{l-m}} \qquad 0\leq m\leq l\,. \tag{6.5.31}$$

Using (6.5.9) it then follows that:

$$\boldsymbol{\Gamma}_p\,\boldsymbol{\mu} = \left(\sum_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}} \xi_p^{\mathbf{i}_{l-m}}\,\tilde{z}_{\mathbf{i}_{l-m}}^{\mathbf{c}_{l-m}}\right)\cdot\left((\mathbf{z})^{c_{l-m+1}}\otimes\cdots\otimes(\mathbf{z})^{c_l}\right), \tag{6.5.32}$$

which shows that $\boldsymbol{\Gamma}_p\,\boldsymbol{\mu}$ is a vector in the direction of $\left((\mathbf{z})^{c_{l-m+1}}\otimes\cdots\otimes(\mathbf{z})^{c_l}\right)$. Hence, $\boldsymbol{\mu}$ is also a generalized eigenvector of all matrices $\boldsymbol{\Gamma}_p$ consisting of linear combinations of the matrices $\left\{\mathbf{G}_{\mathbf{i}_{l-m}}\right\}_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}}$. In general, the matrices of size $(D)^m\times d_D^{x,c_1c_2}$ (6.5.10) that are involved in the MMGEVD decompositions are rectangular. In some cases, depending on the number of sources, number of sensors, and the conjugation tuple, one of the equivalent MMGEVD problems can be reduced to the standard GEVD of two square matrices; for example, see (6.5.22) and Section 5.6. As we have also remarked in Section 5.6, if the matrices in a MMGEVD problem have less rows than columns, it cannot be solved immediately by traditional techniques. This leads us to the conclusion that general MMGEVD algorithms that can also handle rectangular matrices need to be developed.

After having computed one of the (MM)GEVD decompositions, the columns of the mixing matrix $\mathbf{A}$ can be recovered from the generalized eigenvectors $\boldsymbol{\mu}^1,\ldots,\boldsymbol{\mu}^S$. To start with, suppose that a decomposition of the form (6.5.29) is available for some $0\leq m\leq l$. For convenience, it is also assumed that the array response vectors are no scaled versions of each other. Then, S different vectors $\boldsymbol{\mu}^1,\ldots,\boldsymbol{\mu}^S$ that satisfy (6.5.9) are available and the matrix $\mathbf{M}_{G_m^l}$ in (6.5.30) is of size $S\times S$. In order to compute the corresponding array response vectors $\mathbf{z}^1,\ldots,\mathbf{z}^S$ we employ (6.5.9) with $m=1$, i.e. (6.5.8). This latter equation shows that for a generalized solution eigenvector $\boldsymbol{\mu}^j$ the vectors $\left\{\mathbf{G}_{\mathbf{i}_{l-1}}\,\boldsymbol{\mu}^j\right\}_{\mathbf{i}_{l-1}\in\mathcal{I}_{\mathrm{t},D}^{l-1}}$ lie in the direction of $(\mathbf{z}^j)^{c_l}$, where $\mathbf{z}^j$ is an array response vector. Hence, given $\boldsymbol{\mu}^j$ the corresponding solution for $\mathbf{z}^j$ can be computed as an arbitrary linear combination of $\left\{\mathbf{G}_{\mathbf{i}_{l-1}}\,\boldsymbol{\mu}^j\right\}_{\mathbf{i}_{l-1}\in\mathcal{I}_{\mathrm{t},D}^{l-1}}$ (see (6.5.31)) that is conjugated according to c_l, i.e.:

$$\mathbf{z}^j = \frac{\left(\sum\limits_{\mathbf{i}_{l-1}\in\mathcal{I}_{\mathrm{t},D}^{l-1}} \xi^{\mathbf{i}_{l-1}}\mathbf{G}_{\mathbf{i}_{l-1}}\,\boldsymbol{\mu}^j\right)^{c_l}}{\left\|\sum\limits_{\mathbf{i}_{l-1}\in\mathcal{I}_{\mathrm{t},D}^{l-1}} \xi^{\mathbf{i}_{l-1}}\mathbf{G}_{\mathbf{i}_{l-1}}\,\boldsymbol{\mu}^j\right\|} \qquad 1\leq j\leq S\,, \tag{6.5.33}$$

where $\{\xi^{\mathbf{i}_{l-m}}\}_{\mathbf{i}_{l-m}\in\mathcal{I}_{\mathrm{t},D}^{l-m}}$ is a set of arbitrary non-zero real- or complex-valued scalars.

6.6 Maximum number of columns or minimum number of sensors for given conjugation tuple

Let $S^{\mathbf{c}_l}_{\max}(D)$ be the maximum number of source array response vectors, i.e. columns of the mixing matrix $\mathbf{A}$, that can be identified for a given number of sensors D and conjugation tuple $\mathbf{c}_l$ under assumptions *AS1-AS4* on page 299. Conversely, let $D^{\mathbf{c}_l}_{\min}(S)$ be the minimum number of sensors that is required for uniquely identifying S source array response vectors for a given conjugation tuple $\mathbf{c}_l$ under assumptions *AS1-AS4*. In this section, we provide insight for determining (bounds on) $S^{\mathbf{c}_l}_{\max}(D)$ and $D^{\mathbf{c}_l}_{\min}(S)$ by using properties of the system of equations (6.2.66), as well as other results obtained in the previous sections. In particular, the number of variables D, the conjugation tuple $\mathbf{c}_l$, and the number $Q^{\mathbf{c}_l}_{D,S}$ of linearly independent equations in the system are of paramount importance.

6.6.1 Number of linearly independent equations

To start with, we consider the number of linearly independent equations $Q^{\mathbf{c}_l}_{D,S}$ as given by (6.2.60), (6.2.80) and (6.2.113). The latter equations show that $Q^{\mathbf{c}_l}_{D,S}$ equals the difference between the number $M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ of unique sensor cumulant functions and the dimension $d^{x,\mathbf{c}_l}_{D}$ of the signal subspace:

$$Q^{\mathbf{c}_l}_{D,S} = M^{x,\mathbf{c}_l}_{\mathrm{u},D} - d^{x,\mathbf{c}_l}_{D} \,. \tag{6.6.1}$$

As we have shown in Section 6.2.3, on the one hand $M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ equals the number $M^{\mathbf{c}_l}_{\mathrm{u},D}$ of unique elements of the generalized array response vector that is given by (6.2.41)-(6.2.44), whereas on the other hand $M^{x,\mathbf{c}_l}_{\mathrm{u},D}$ equals the maximum possible row rank of the generalized array response matrix $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}$ that is given by (6.2.52). Hence:

$$M^{x,\mathbf{c}_l}_{\mathrm{u},D} = M^{\mathbf{c}_l}_{\mathrm{u},D} = \max\Big(\operatorname{rank}_{\mathrm{row}}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\right)\Big) \,. \tag{6.6.2}$$

In addition, we have proven in Section 6.2.6.3 that $d^{x,\mathbf{c}_l}_{D}$ equals the rank of $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}$, see (6.2.109). Substituting (6.6.2) and (6.2.109) into (6.6.1) yields the following expression for $Q^{\mathbf{c}_l}_{D,S}$:

$$\boxed{Q^{\mathbf{c}_l}_{D,S} = \max\left(\operatorname{rank}_{\mathrm{row}}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\right)\right) - \operatorname{rank}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\right) .} \tag{6.6.3}$$

As is clear from algebraic and geometric observations, as well as from intuition, this expression is always true, also if $\max\left(\operatorname{rank}_{\mathrm{row}}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\right)\right)$ does not equal the specific expression for $M^{\mathbf{c}_l}_{\mathrm{u},D}$ given by (6.2.41)-(6.2.44).

6.6.2 Derivation of condition for identifiability

Let $Q^{\mathbf{c}_l}_{\min,D}$ be the minimum number of equations that is required for uniquely determining the solution vectors of a system of equations in D complex variables such as the one obtained in the previous sections for a general MIBI scenario with D sensors and conjugation tuple $\mathbf{c}_l$. Then, for a MIBI scenario with S sources the following condition should be satisfied for identifiability:

$$\boxed{Q^{\mathbf{c}_l}_{\min,D} \leq Q^{\mathbf{c}_l}_{D,S} \,.} \tag{6.6.4}$$

From (6.2.51) it follows that for a given number of sensors D and conjugation tuple $\mathbf{c}_l$ the maximum possible rank of $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}$ is bounded by $M^{\mathbf{c}_l}_{\mathrm{u},D}$ and the number of sources S, where

$M_{\mathrm{u},D}^{\mathbf{c}_l}$ bounds the row rank and S bounds the column rank. *In order to determine as many source array response vectors as possible, the generalized array response matrix should have as many linearly independent columns as possible.* In other words, its column rank should be maximum in case there are no limitations on its row rank due to the Kronecker structure. Following the same reasoning as in Section 6.2.3.4 it follows directly that the maximum column rank of $\mathbf{A}_{\diamond}^{\mathbf{c}_l}$ equals:

$$\max\left(\operatorname{rank}_{\text{col}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right) = S\,. \tag{6.6.5}$$

Hence, if:

$$\operatorname{rank}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right) = \max\left(\operatorname{rank}_{\text{col}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right) = S\,, \tag{6.6.6}$$

which in general is the case if $\mathbf{A}$ is randomly generated and $S \leq M_{\mathrm{u},D}^{\mathbf{c}_l}$, then $Q_{D,S}^{\mathbf{c}_l}$ can be expressed in the following way:

$$\boxed{Q_{D,S}^{\mathbf{c}_l} = \max\left(\operatorname{rank}_{\text{row}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right) - \max\left(\operatorname{rank}_{\text{col}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right)\,.} \tag{6.6.7}$$

See Sections 6.3.1-6.3.3 for examples and discussion.

Combining the results above it follows that for given $Q_{\text{min},D}^{\mathbf{c}_l}$ and $\mathbf{c}_l$ (bounds on) the values of $S_{\text{max}}^{\mathbf{c}_l}(D)$ and $D_{\text{min}}^{\mathbf{c}_l}(S)$ can be obtained from the following inequality:

$$Q_{\text{min},D}^{\mathbf{c}_l} \leq \max\left(\operatorname{rank}_{\text{row}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right) - S\,. \tag{6.6.8}$$

In the next two sections we will determine $S_{\text{max}}^{\mathbf{c}_l}(D)$ and $D_{\text{min}}^{\mathbf{c}_l}(S)$ respectively for the general MIBI scenario considered in this chapter, i.e. with $\max\left(\operatorname{rank}_{\text{row}}\left(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\right)\right)$ equal to $M_{\mathrm{u},D}^{\mathbf{c}_l}$ given by (6.2.41)-(6.2.44). Thus, we will employ the following inequality:

$$Q_{\text{min},D}^{\mathbf{c}_l} \leq M_{\mathrm{u},D}^{\mathbf{c}_l} - S = \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\bar{\mathbf{c}}_l} \end{matrix} \right\rangle - S\,, \tag{6.6.9}$$

where we have substituted (6.2.44). As an example of (heuristically) determining $Q_{\text{min},D}^{\mathbf{c}_l}$, from the reasoning in Section 6.3.3.2 it follows that for $\mathbf{c}_l = (\circ)_l$ at least $D-1$ equations are required for uniquely defining the one-dimensional solution sets corresponding to the mixing matrix columns, which implies that $Q_{\text{min},D}^{(\circ)_l} \equiv Q_{\text{min},D}^{l} = D-1$.

6.6.3 Determining the maximum number of columns $S_{\text{max}}^{\mathbf{c}_l}(D)$

From (6.6.9) it follows that for given values of D, $\mathbf{c}_l$, and $Q_{\text{min},D}^{\mathbf{c}_l}$ the value of $S_{\text{max}}^{\mathbf{c}_l}(D)$ is:

$$S_{\text{max}}^{\mathbf{c}_l}(D) = M_{\mathrm{u},D}^{\mathbf{c}_l} - Q_{\text{min},D}^{\mathbf{c}_l} = \left\langle \begin{matrix} D \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D \\ n_{\bar{\mathbf{c}}_l} \end{matrix} \right\rangle - Q_{\text{min},D}^{\mathbf{c}_l}\,. \tag{6.6.10}$$

As an example, consider a real-valued MIBI scenario with $\mathbf{c}_l = (\circ)_l$ and $Q_{\text{min},D}^{(\circ)_l} = D-1$. Substituting (6.2.33) into (6.6.10), it follows that for a given value of D:

$$S_{\text{max}}^{\mathbf{c}_l}(D) \triangleq S_{\text{max}}^{(\circ)_l}(D) = \binom{l+D-1}{l} - (D-1) = \frac{(l+D-1)!}{l!(D-1)!} - (D-1)\,, \tag{6.6.11}$$

which agrees with (6.3.22). Table 6.3 on the next page lists the values of $S_{\text{max}}^{(\circ)_l}(D)$ for $1 \leq l \leq 4$ and $1 \leq D \leq 5$.

Table 6.3: *Values of $S_{\max}^{(\circ)_l}(D)$ for $Q_{\min,D}^{(\circ)_l} = D-1$, $1 \le l \le 4$, and $1 \le D \le 5$.*

$l \backslash D$	1	2	3	4	5
1	1	1	1	1	1
2	1	2	4	7	11
3	1	3	8	17	31
4	1	4	13	32	66

6.6.4 Determining the minimum number of sensors $D_{\min}^{\mathbf{c}_l}(S)$

From (6.6.9) it follows that for given values of S and $\mathbf{c}_l$, and a given function $Q_{\min,D}^{\mathbf{c}_l}$ of D, the value of $D_{\min}^{\mathbf{c}_l}(S)$ can be obtained by solving the following inequality:

$$Q_{\min,D_{\min}^{\mathbf{c}_l}(S)}^{\mathbf{c}_l} \le M_{\mathrm{u},D_{\min}^{\mathbf{c}_l}(S)}^{\mathbf{c}_l} - S = \left\langle \begin{matrix} D_{\min}^{\mathbf{c}_l}(S) \\ n_{\mathbf{c}_l} \end{matrix} \right\rangle \left\langle \begin{matrix} D_{\min}^{\mathbf{c}_l}(S) \\ n_{\bar{\mathbf{c}}_l} \end{matrix} \right\rangle - S\,. \tag{6.6.12}$$

Depending on the cumulant order and conjugation tuple this inequality may be complicated to solve. However, it can easily be solved for the relevant range of values by simply using tabulated values as in Table 6.2, for example.

For the real-valued MIBI scenario with $\mathbf{c}_l = (\circ)_l$ and $Q_{\min,D}^{(\circ)_l} = D-1$ it follows by substituting (6.2.33) into (6.6.12) that for a given value of S, the value of $D_{\min}^{(\circ)_l}(S)$ can be obtained by solving the following inequality:

$$D_{\min}^{(\circ)_l}(S) - 1 \le \binom{l + D_{\min}^{(\circ)_l}(S) - 1}{l} - S\,. \tag{6.6.13}$$

We now show how this inequality can be solved for the first few values of l, viz. $l = 1, 2, 3$. For convenience we temporarily write D instead of $D_{\min}^{(\circ)_l}(S)$.

To start with, substituting $l = 1$ into (6.6.13) yields:

$$D - 1 \le D - S \qquad \Longrightarrow \qquad S \le 1\,. \tag{6.6.14}$$

This result states that for $l = 1$ at most one source array response vector can be identified, no matter how many sensors are used. Hence, there is no unique value for $D_{\min}^{\circ}(S)$. This agrees with the results in the first row of Table 6.3, as well as with intuition.

Substituting $l = 2$ into (6.6.13) gives:

$$D - 1 \le \tfrac{1}{2}(D+1)D - S\,. \tag{6.6.15}$$

Solving this inequality gives the following formula for $D_{\min}^{(\circ)_2}(S)$ as a function of S:

$$D_{\min}^{(\circ)_2}(S) = \left\lceil \tfrac{1}{2}\left(1 + \sqrt{8S-7}\right) \right\rceil, \tag{6.6.16}$$

which agrees with the results in the second row of Table 6.3.

Finally, substituting $l = 3$ into (6.6.13) yields:

$$D - 1 \le \tfrac{1}{6}(D+2)(D+1)D - S\,. \tag{6.6.17}$$

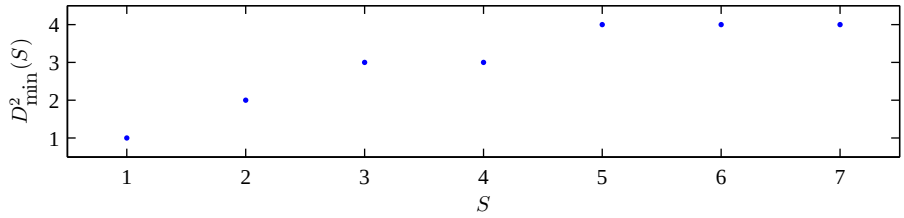

Figure 6.10: *Minimum number of sensors* $D^{(\circ)_2}_{\min}(S)$.

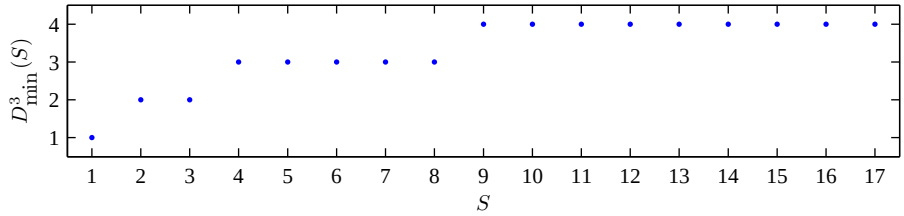

Figure 6.11: *Minimum number of sensors* $D^{(\circ)_3}_{\min}(S)$.

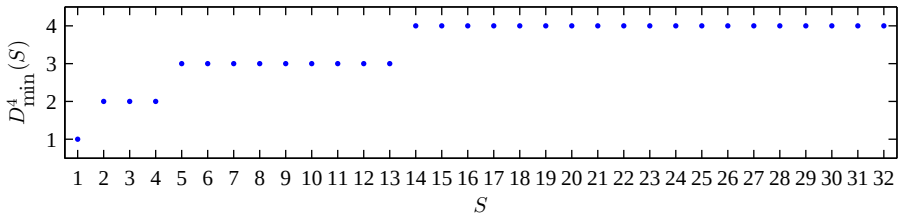

Figure 6.12: *Minimum number of sensors* $D^{(\circ)_4}_{\min}(S)$.

Solving this inequality on the appropriate domain gives the following formula for $D^{(\circ)_3}_{\min}(S)$ as a function of S:

$$D^{(\circ)_3}_{\min}(S) = \left\lceil \frac{\sqrt[3]{3}\, r(S)}{3} + \frac{7}{\sqrt[3]{3}\, r(S)} - 1 \right\rceil, \tag{6.6.18}$$

where the function $r(S)$ is defined by:

$$r(S) \triangleq \sqrt[3]{27(S-2) + \sqrt{3}\sqrt{243(S-4)S + 629}}\ .$$

This result agrees with the third row of Table 6.3.

Using the values in tables like Table 6.3, the solution of inequality (6.6.13) for any value of l can easily be obtained. For example, for cumulant order $l = 4$, at least $D^{(\circ)_4}_{\min}(S) = 3$ sensors are required if $5 \leq S \leq 13$. In Figures 6.10, 6.11 and 6.12, $D^{(\circ)_l}_{\min}(S)$ is plotted as a function of S for $l = 2, 3$ and $l = 4$ respectively.

6.7 Conclusions and discussion

The theory that we have developed in this chapter is unifying in several senses. Firstly, it is general with respect to the order of the exploited temporal structure in the sense that the mathematical problem formulation has the same structure for any order. This is reflected in the uniformity of our notation. Secondly, all types of statistical variability in the data, such as arbitrary order non-stationarity and non-whiteness, are exploited in a unified manner. Finally, for complex-valued signals, the conjugation pattern of the arguments of the involved cumulant functions can be chosen arbitrarily. In practice, this should be done in accordance with the characteristics of the involved signals. Depending on the number of sensors, the order of the exploited temporal structure, and the chosen conjugation pattern, a large number of mixing matrix columns can be determined that exceeds the number of sensors for orders larger than one. We have provided insight into the computation of the maximum number of identifiable columns and the minimum number of sensors. Our method allows to annihilate the influence of noise signals that have a simpler temporal structure than the source signals, provides insight into the geometric and algebraic structure of the problem, allows to make trade-offs between various quantities such as the number of sources or columns of the mixing matrix that can be identified, number of sensors, exploited order of temporal structure, exploited type of temporal structure, and so on.

CHAPTER 7

Semi-Blind Source Localization

In this chapter, we apply the theory that we have developed in the previous chapters to the problem of *Instantaneous Semi-Blind Source Localization (ISBSL)* that we have introduced in Chapter 1 and elaborated on in Chapter 3. As we have explained in Section 1.1.3, narrowband source localization deals with the extraction of position related parameters such as Direction Of Arrival (DOA) and range from the sensor array data. This is possible due to the available a priori knowledge about the propagation medium characteristics, source signals, and sensor positions, from which the array response vector as a function of the source position parameters can be deduced. We have also seen that ISBSL can be considered as a parameterized version of MIBI because the columns of the mixing matrix have a known functional dependence on the source position parameters. The fact that the array response vector is a function of the source position parameters implies that it has less degrees of freedom than a general MIBI array response vector. In turn, this implies that the array response matrix also has less degrees of freedom and that its maximum possible rank as a function of the source position parameters may be smaller than that derived in Section 6.2.3.4 for the general MIBI problem. On the one hand this means that there are less independent equations in the system, but on the other hand there is also more structure in the problem and less variables to be solved for. As we will see in this chapter, it depends on the specific source parameters and array configuration how many 'true' equations the relevant system of equations contains and how many source positions can be estimated. The number of unique elements in a generalized array response vector, which equals the number of unique sensor cumulant functions, is not given any more by expressions (6.2.41)-(6.2.44). However, except for those points, all equations and results essentially remain the same if we use the full Kronecker subspace matrix $\mathbf{C}_D^{x,\mathbf{c}_l}$ instead of the uniquiefied subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$. Therefore, for the sake of generality we will mainly express the results obtained in this chapter in terms of $\mathbf{C}_D^{x,\mathbf{c}_l}$, $\mathcal{K}_{\mathrm{t},D}^{x,\mathbf{c}_l}$, $\mathcal{I}_{\mathrm{t},D}^{l}$, and $M_{\mathrm{t},D}^{x,\mathbf{c}_l}$.

In Section 7.1 we develop a MUSIC-like method called TIME-MUSIC for DOA estimation of far field sources with a two-dimensional array of arbitrary geometry by substituting the parameterized array response vector into the expressions obtained in the previous chapter for the general MIBI case. In particular, the spectral version of this method, called TIME-SPECTRAL-MUSIC, is considered. Then, in Section 7.2 we specialize the results to the case of a Uniform Linear Array (ULA), which results in a method called ULA-TIME-SPECTRAL-MUSIC. In Section 7.3, a ROOT-MUSIC-like method called ULA-TIME-ROOT-MUSIC is designed. Finally, conclusions are drawn in Section 7.4.

7.1 One-parameter Direction Of Arrival estimation of far field sources with TIME-SPECTRAL-MUSIC

In this section, we develop a MUSIC-like method called TIME-MUSIC for DOA estimation of far field sources with a two-dimensional array of arbitrary geometry. This scenario is depicted in Fig. 3.1. As in Section 3.1, it is assumed that S statistically independent complex narrowband signals emitted by far field sources are incident on a planar array of D sensors, that the propagation medium is linear, homogeneous and isotropic, and that all sensors are identical and have omnidirectional unit transfer. By 'arbitrary geometry' we mean that the sensors in principle may be positioned anywhere in the two-dimensional plane, i.e. that their arrangement is not necessarily restricted to a ULA, for example. By 'one-parameter DOA estimation' we mean that there is only one parameter characterizing the position of each source w.r.t. the chosen coordinate system, viz. the DOA that is denoted by θ and defined as the angle between the line from the source to the origin and the positive vertical axis (see Fig. 3.1). See also Sections 1.1.3 and 3.1 for more information on the model, derivation of array response vectors, array response matrix, complex transfer coefficients, and so on.

7.1.1 Structure of system and equations

As we have derived in the previous chapter, for the general MIBI problem the system of equations is given by (6.2.66), where the functions are defined in (6.2.64) and (6.2.123). As can be seen in equations (3.1.1)-(3.1.7) the array response vectors are parameterized by the DOA θ. Hence, the unknown vector of variables $\mathbf{z}$ now depends on θ and thus can be replaced by $\mathbf{z}(\theta)$ with the specific functional dependence on θ that we have derived in Section 3.1. In the next sections, we will tailor the form of the equations and system that have been derived in the previous chapters to the current scenario. Firstly, with abuse of notation we can write the system and function definitions from the previous chapter as:

$$\big\{ f_{D,q}^{\mathbf{c}_l}(\theta) = 0 \big\}_{q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}} \tag{7.1.1}$$

and:

$$f_{D,q}^{\mathbf{c}_l}(\theta) \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}} \varphi_q^{\mathbf{i}_l}\, \check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}(\theta) = \tilde{\boldsymbol{\varphi}}_q\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) \qquad \forall\, \theta \in \Theta\,, \qquad \forall\, q \in \mathcal{T}_{D,S}^{\mathbf{c}_l} \tag{7.1.2}$$

respectively, where Θ is the parameter space of interest that has been defined in Section 3.1, and the set $\mathcal{T}_{D,S}^{\mathbf{c}_l}$ is defined similarly to (4.3.23), (5.3.66), and (6.2.59) as follows:

$$\mathcal{T}_{D,S}^{\mathbf{c}_l} \triangleq \big\{1, \ldots, T_{D,S}^{\mathbf{c}_l}\big\} \tag{7.1.3}$$

with $T_{D,S}^{\mathbf{c}_l} \triangleq \big|\mathcal{T}_{D,S}^{\mathbf{c}_l}\big|$ the maximum number of linearly independent equations, which analogously to (6.2.60), (6.2.80), and (6.2.113), is given by:

$$\boxed{T_{D,S}^{\mathbf{c}_l} = \dim\big(\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}_D^{x,\mathbf{c}_l}\big)\big) = \#_{\text{rows}}\{\mathbf{C}_D^{x,\mathbf{c}_l}\} - \operatorname{rank}(\mathbf{C}_D^{x,\mathbf{c}_l}) = (D)^l - \operatorname{rank}\big(\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big(\tilde{\boldsymbol{\theta}}\big)\big)\,.} \tag{7.1.4}$$

Note that the summation in (7.1.2) now runs over all possible length-l tuples $\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^{l}$ with $1 \leq i_m \leq D$ instead of over the tuples in $\mathcal{I}_{\mathrm{u},D}^{\mathbf{c}_l}$ only. Recall that $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big(\tilde{\boldsymbol{\theta}}\big)$ is the l-th order Khatri-Rao product with conjugation tuple $\mathbf{c}_l$ defined by (6.2.49) and (6.2.50). Note also that the number $T_{D,S}^{\mathbf{c}_l}$ by definition gives the number of independent equations of system (7.1.1)

for the current formulation. A formulation in terms of uniquified quantities would yield a system with ${}_{\theta}Q^{\mathbf{c}_l}_{D,S}$ essentially different equations, where in general ${}_{\theta}Q^{\mathbf{c}_l}_{D,S}$ is different from $T^{\mathbf{c}_l}_{D,S}$ and the number $Q^{\mathbf{c}_l}_{D,S}$ defined in the previous chapters for general MIBI. In this context, it is important to realize that similarly to (6.6.3) the maximum number ${}_{\theta}S^{\mathbf{c}_l}_{\max,D}$ of columns and/or sources that can be identified is determined by the difference:

$$ {}_{\theta}Q^{\mathbf{c}_l}_{D,S} \triangleq \max\Big(\operatorname{rank}_{\text{row}}\big(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}(\tilde{\boldsymbol{\theta}})\big)\Big) - \operatorname{rank}\Big(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}(\tilde{\boldsymbol{\theta}})\Big) \tag{7.1.5}$$

between the maximum possible row rank and the actual rank of $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}(\tilde{\boldsymbol{\theta}})$ respectively. See also Section 7.1.3.

Now we will examine the specific form that the functions in (7.1.2) have by virtue of the fact that the array response vectors are parameterized according to the DOA scenario. From (3.1.5) it follows that the i-th element $z_i(\theta)$ of $\mathbf{z}(\theta)$ can be written as:

$$ z_i(\theta) = \exp\big(\jmath\, 2\pi \tilde{\mathbf{p}}_{i,\text{n}} \mathbf{u}(\theta)\big) = \exp\big(\jmath\, \phi_i(\theta)\big) \qquad \forall\, \theta \in \Theta, \quad \forall\, 1 \leq i \leq D\,, \tag{7.1.6}$$

where the phase $\phi_i(\theta)$ is defined in (3.1.3), which is repeated here for convenience:

$$ \phi_i(\theta) = 2\pi \frac{\tilde{\mathbf{p}}_i}{\lambda_{\text{c}}} \mathbf{u}(\theta) = 2\pi \tilde{\mathbf{p}}_{i,\text{n}}\, \mathbf{u}(\theta) \qquad \forall\, \theta \in \Theta, \quad \forall\, 1 \leq i \leq D\,. \tag{7.1.7}$$

As is explained in Section 3.1.1, the row vector $\tilde{\mathbf{p}}_i = \begin{bmatrix} p^1_i & p^2_i \end{bmatrix}$ denotes the position of the i-th sensor, whereas $\tilde{\mathbf{p}}_{i,\text{n}}$ denotes the position vector of the i-th sensor that is normalized w.r.t. the carrier wavelength λ_{c}, and finally $\mathbf{u}(\theta) = \begin{bmatrix} \sin(\theta) & \cos(\theta) \end{bmatrix}^T$ denotes the unit column vector pointing in DOA θ. Using (7.1.6), the product $\check{z}^{\mathbf{c}_l}_{\mathbf{i}_l}(\theta)$ arising in (7.1.2) can be expressed as follows:

$$\begin{aligned} \check{z}^{\mathbf{c}_l}_{\mathbf{i}_l}(\theta) &= \prod_{m=1}^{l} \big(z_{i_m}(\theta)\big)^{c_m} = \prod_{m=1}^{l} \Big(\exp\big(\jmath\, \phi_{i_m}(\theta)\big)\Big)^{c_m} = \prod_{m=1}^{l} \exp\big((\jmath)^{c_m}\, \phi_{i_m}(\theta)\big) \\ &= \exp\left(\sum_{m=1}^{l} (\jmath)^{c_m}\, \phi_{i_m}(\theta)\right) = \exp\left(\jmath \sum_{m=1}^{l} \langle -1\rangle^{c_m}\, \phi_{i_m}(\theta)\right), \end{aligned} \tag{7.1.8}$$

where by definition:

$$ \langle -1\rangle^{*} \triangleq -1 \qquad \text{and} \qquad \langle -1\rangle^{\circ} \triangleq 1\,. \tag{7.1.9}$$

Now substituting (7.1.7) into (7.1.8) yields:

$$ \check{z}^{\mathbf{c}_l}_{\mathbf{i}_l}(\theta) = \exp\left(\jmath\, 2\pi \sum_{m=1}^{l} \langle -1\rangle^{c_m} \tilde{\mathbf{p}}_{i_m,\text{n}}\, \mathbf{u}(\theta)\right) = \exp\Big(\jmath\, 2\pi\, \tilde{\mathbf{t}}^{\mathbf{c}_l}_{\mathbf{i}_l,\text{n}}\, \mathbf{u}(\theta)\Big)\,, \tag{7.1.10}$$

where:

$$ \tilde{\mathbf{t}}^{\mathbf{c}_l}_{\mathbf{i}_l,\text{n}} \triangleq \sum_{m=1}^{l} \langle -1\rangle^{c_m} \tilde{\mathbf{p}}_{i_m,\text{n}}\,. \tag{7.1.11}$$

Note the resemblance of (7.1.6) and (7.1.10), due to which $\tilde{\mathbf{t}}_{\mathbf{i}_l,\text{n}}$ can be seen as a *'conjugation-compensated mean normalized position vector'*. Using (7.1.10) the functions defined in (7.1.2) can be expressed as follows:

$$ f^{\mathbf{c}_l}_{D,q}(\theta) \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}^{l}_{\iota,D}} \varphi^{\mathbf{i}_l}_q\, \exp\Big(\jmath\, 2\pi\, \tilde{\mathbf{t}}^{\mathbf{c}_l}_{\mathbf{i}_l,\text{n}}\, \mathbf{u}(\theta)\Big) \qquad \forall\, \theta \in \Theta\,, \qquad \forall\, q \in \mathcal{T}^{\mathbf{c}_l}_{D,S}\,. \tag{7.1.12}$$

From (6.2.65) it follows that:

$$f_{D,q}^{\mathbf{c}_l}(\theta^j) = 0 \qquad \forall\, q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}, \quad 1 \leq j \leq S\,. \tag{7.1.13}$$

Hence, the source DOA's can be estimated by locating the zeros of the system $\{f_{D,q}^{\mathbf{c}_l}(\theta) = 0\}_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}}$, which is a one-dimensional search problem. Observe that $\mathbf{k} \triangleq -\frac{2\pi}{\lambda_{\mathrm{c}}}\mathbf{u}(\theta)$ is the wave vector of a signal coming from direction θ and having wavenumber $\|\mathbf{k}\| = \frac{2\pi}{\lambda_{\mathrm{c}}}$. Hence, the array response matrix is given by (see also (7.1.20)):

$$\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \triangleq \big[\mathbf{z}(\theta^1) \;\cdots\; \mathbf{z}(\theta^S)\big] = \begin{bmatrix} \exp\big(-\jmath\,\tilde{\mathbf{p}}_1\mathbf{k}^1\big) & \cdots & \exp\big(-\jmath\,\tilde{\mathbf{p}}_1\mathbf{k}^S\big) \\ \vdots & \ddots & \vdots \\ \exp\big(-\jmath\,\tilde{\mathbf{p}}_D\,\mathbf{k}^1\big) & \cdots & \exp\big(-\jmath\,\tilde{\mathbf{p}}_D\,\mathbf{k}^S\big) \end{bmatrix}, \tag{7.1.14}$$

where $\mathbf{k}^j \triangleq -\frac{2\pi}{\lambda_{\mathrm{c}}}\mathbf{u}(\theta^j)$ denotes the wavenumber vector corresponding to the j-th source. Likewise, observe that the Khatri-Rao product $\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big(\tilde{\boldsymbol{\theta}}\big)$ is given by:

$$\mathbf{A}_{D,\diamond}^{\mathbf{c}_l}\big(\tilde{\boldsymbol{\theta}}\big) \triangleq \big[\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta^1) \;\cdots\; \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta^S)\big] = \begin{bmatrix} \exp\big(-\jmath\,\tilde{\mathbf{t}}_{\mathbf{i}_{l,1}}^{\mathbf{c}_l}\,\mathbf{k}^1\big) & \cdots & \exp\Big(-\jmath\,\tilde{\mathbf{t}}_{\mathbf{i}_{l,1}}^{\mathbf{c}_l}\,\mathbf{k}^S\Big) \\ \vdots & \ddots & \vdots \\ \exp\big(-\jmath\,\tilde{\mathbf{t}}_{\mathbf{i}_{l,(D)^l}}^{\mathbf{c}_l}\,\mathbf{k}^1\big) & \cdots & \exp\big(-\jmath\,\tilde{\mathbf{t}}_{\mathbf{i}_{l,(D)^l}}^{\mathbf{c}_l}\,\mathbf{k}^S\big) \end{bmatrix}, \tag{7.1.15}$$

where:

$$\tilde{\mathbf{t}}_{\mathbf{i}_l}^{\mathbf{c}_l} \triangleq \sum_{m=1}^{l} \langle -1\rangle^{c_m}\tilde{\mathbf{p}}_{i_m} = \lambda_{\mathrm{c}}\,\tilde{\mathbf{t}}_{\mathbf{i}_l,\mathrm{n}}^{\mathbf{c}_l} \tag{7.1.16}$$

is the '*conjugation-compensated mean position vector*' (see also (7.1.11)), and the index tuples $\mathbf{i}_{l,1},\ldots,\mathbf{i}_{l,(D)^l}$ are arranged vertically in 'Kronecker order'.

7.1.2 TIME-MUSIC null- and pseudo-spectra

Similarly to the MUSIC algorithms presented in Chapter 3, based on system $\{f_{D,q}^{\mathbf{c}_l}(\theta) = 0\}_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}}$ we define an appropriate *null-spectrum*, called the *l-th order TIME-MUSIC null-spectrum*, as follows:

$$\boxed{J_D^{\mathbf{c}_l}(\theta) \triangleq \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \Big|f_{D,q}^{\mathbf{c}_l}(\theta)\Big|^2 = \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \Bigg|\sum_{\mathbf{i}_l\in\mathcal{I}_{\mathrm{t},D}^{l}} \varphi_q^{\mathbf{i}_l}\,\exp\Big(\jmath\,2\pi\,\tilde{\mathbf{t}}_{\mathbf{i}_l,\mathrm{n}}^{\mathbf{c}_l}\mathbf{u}(\theta)\Big)\Bigg|^2.} \tag{7.1.17}$$

See also the similar functions defined in (3.2.27), (3.3.34) and (3.3.35). Clearly, due to (6.2.65) the zeros or minima of this function are located exactly at the source DOA's for exactly known statistics, and approximately for estimated statistics. Hence, if we have an estimate $\hat{\mathbf{C}}_D^{x,\mathbf{c}_l}$ of $\mathbf{C}_D^{x,\mathbf{c}_l}$ that is sufficiently accurate, then the null-spectrum $J_D^{\mathbf{c}_l}(\theta)$ in (7.1.17) is approximately zero for the true source DOA's (compare this with (3.2.28) and (3.3.36)):

$$J_D^{\mathbf{c}_l}(\theta)\big|_{\theta=\theta^j} \to 0 \qquad \forall\, 1 \leq j \leq S \quad \text{if} \quad \hat{\mathbf{C}}_D^{x,\mathbf{c}_l} \to \mathbf{C}_D^{x,\mathbf{c}_l}\,. \tag{7.1.18}$$

Hence, the source DOA's can be estimated by locating the minima or approximate zeros of (7.1.17). Obviously, a DOA estimator based on this principle is statistically consistent because $\hat{\mathbf{C}}_D^{x,\mathbf{c}_l} \to \mathbf{C}_D^{x,\mathbf{c}_l}$ when the number of samples used for its estimation goes to infinity.

Similarly to the SOS-SPECTRAL-MUSIC and HOS-SPECTRAL-MUSIC pseudo-spectra defined in Chapter 3, we define a pseudo-spectrum called the *l-th order TIME-MUSIC pseudo-spectrum* that is proportional to the reciprocal of (7.1.17):

$$\boxed{P_D^{\mathbf{c}_l}(\theta) \triangleq \frac{\|\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2}{\sum\limits_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \left|f_{D,q}^{\mathbf{c}_l}(\theta)\right|^2} = \frac{(D)^l}{\sum\limits_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \left|\sum\limits_{\mathbf{i}_l\in\mathcal{I}_{\mathfrak{l},D}^l} \varphi_q^{\mathbf{i}_l}\, \exp\Big(\jmath\, 2\pi\, \tilde{\mathbf{t}}_{\mathbf{i}_l,\mathfrak{n}}^{\mathbf{c}_l}\mathbf{u}(\theta)\Big)\right|^2} .} \tag{7.1.19}$$

Note that the nominator is just a normalization constant that may be omitted. Similarly to the conventional MUSIC pseudo-spectrum, the TIME-MUSIC pseudo-spectrum $P_D^{\mathbf{c}_l}(\theta)$ exhibits sharp peaks at the source DOA's $\theta^1, \ldots, \theta^S$ and thus these DOA's can be estimated by locating the peaks of the pseudo-spectrum. The above procedure is called TIME-MUSIC because it exploits the temporal/time structure in the data and due to its resemblance with the conventional MUSIC algorithm. Notice the resemblance of (7.1.19) to (3.2.30) and (3.3.38).

For convenience of notation we now express the null- and pseudo-spectra defined above in matrix-vector notation. Firstly, expressing $\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)$ (see (6.2.22)) as follows:

$$\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) = \mathbf{w}_D^{\mathbf{c}_l}(\theta) = \begin{bmatrix} \check{z}_{\mathbf{i}_{l,1}}^{\mathbf{c}_1}(\theta) \\ \vdots \\ \check{z}_{\mathbf{i}_{l,(D)^l}}^{\mathbf{c}_1}(\theta) \end{bmatrix} = \begin{bmatrix} \exp\Big(\jmath\, 2\pi\, \tilde{\mathbf{t}}_{\mathbf{i}_{l,1},\mathfrak{n}}^{\mathbf{c}_l}\, \mathbf{u}(\theta)\Big) \\ \vdots \\ \exp\Big(\jmath\, 2\pi\, \tilde{\mathbf{t}}_{\mathbf{i}_{l,(D)^l,\mathfrak{n}}}^{\mathbf{c}_l}\, \mathbf{u}(\theta)\Big) \end{bmatrix} \tag{7.1.20}$$

and writing $\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta)$ for the column vector $\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta)$ containing the functions $\big\{f_{D,q}^{\mathbf{c}_l}(\theta)\big\}_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}}$ (see also (6.2.124)), system (7.1.1) becomes:

$$\boxed{\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta) = \mathbf{\Phi}\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) = \mathbf{\Phi}\, \mathbf{w}_D^{\mathbf{c}_l}(\theta) = \mathbf{0}_{T_{D,S}^{\mathbf{c}_l}} .} \tag{7.1.21}$$

As above, note from (6.2.65) that $\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta^j) = \mathbf{0}$ for all $1 \le j \le S$. Now (7.1.17) and (7.1.19) can be written as:

$$J_D^{\mathbf{c}_l}(\theta) = \|\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta)\|^2 = \|\mathbf{\Phi}\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2 = \big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{\Phi}^H \mathbf{\Phi}\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) \tag{7.1.22}$$

and:

$$P_D^{\mathbf{c}_l}(\theta) = \frac{\|\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2}{\|\mathbf{f}_{D,S}^{\mathbf{c}_l}(\theta)\|^2} = \frac{\|\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2}{\|\mathbf{\Phi}\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2} = \frac{\|\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2}{\big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{\Phi}^H \mathbf{\Phi}\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)} \tag{7.1.23}$$

respectively. Let $\mathbf{\Phi}$ be defined as in (6.2.117), i.e. its rows span the conjugate left null space $\big(\mathcal{N}_l\big(\mathbf{C}_D^{x,\mathbf{c}_l}\big)\big)^* = \big(\mathcal{N}_l\big(\boldsymbol{\kappa}_D^{x,\mathbf{c}_l}[\mathbf{n}_l]\big)\big)^* = \big(\mathcal{R}_{\mathrm{c}}\big(\mathbf{U}^\nu\big)\big)^H$ of the subspace matrix $\mathbf{C}_D^{x,\mathbf{c}_l}$. Using equation (6.2.150) from Section 6.2.8.3 it finally follows that:

$$J_D^{\mathbf{c}_l}(\theta) = \big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{U}^\nu (\mathbf{U}^\nu)^H\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) = \big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{P}^\nu\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) \tag{7.1.24}$$

and:

$$P_D^{\mathbf{c}_l}(\theta) = \frac{\|\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\|^2}{\big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{U}^\nu (\mathbf{U}^\nu)^H\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)} = \frac{(D)^l}{\big(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\big)^H \mathbf{P}^\nu \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)} \tag{7.1.25}$$

respectively, where $\mathbf{P}^\nu$ is the orthogonal projection matrix onto the noise/zero subspace defined in (3.2.17). Recall that $\mathbf{P}^\nu = (\mathbf{I} - \mathbf{P}^s)$, where $\mathbf{P}^s$ is the orthogonal projection matrix

Algorithm 7.1 Arbitrary order TIME-SPECTRAL-MUSIC algorithm for $D \times S$ DOA estimation scenario with arbitrary conjugation tuple $\mathbf{c}_l$.

1: Compute/estimate sensor cumulant functions in set $\mathcal{K}_{\mathrm{u},D}^{x,\mathbf{c}_l} = \left\{\kappa_{\mathbf{i}_l}^{x,\mathbf{c}_l}[\mathbf{n}_l]\right\}_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{u},D}^{l}}$ *for time index tuples* $\mathbf{n}_l$ *in Noise-Free ROS* $\Omega_{\mathbf{n}_l}^{s\backslash\nu,\mathbf{c}_l}$*;*

2: Arrange these values in sensor cumulant subspace matrix $\mathbf{C}_{\mathrm{u},D}^{x,\mathbf{c}_l}$*;*

3: Compute Singular Value Decomposition (SVD) of $\mathbf{C}_{D}^{x,\mathbf{c}_l}$ *and split the result into signal and null/noise subspace parts as follows:*

$$\mathbf{C}_{D}^{x,\mathbf{c}_l} = \mathbf{U}\mathbf{\Sigma}(\mathbf{V})^* = \mathbf{U}^s\mathbf{\Sigma}_s^s(\mathbf{V}_s)^* + \mathbf{U}^\nu\mathbf{\Sigma}_\nu^\nu(\mathbf{V}_\nu)^*\,;$$

4: Compute matrix for projection onto noise subspace:

$$\mathbf{P}^\nu \triangleq \mathbf{U}^\nu(\mathbf{U}^\nu)^H;$$

5: Evaluate l*-th order TIME-SPECTRAL-MUSIC pseudo-spectrum on fine grid in* θ*-domain:*

$$P_D^{\mathbf{c}_l}(\theta) = \frac{1}{\left(\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)\right)^H \mathbf{P}^\nu\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)} \qquad \forall\, \theta \in \Theta\,;$$

6: Determine source DOA's by locating the S *sharpest peaks of* $P_D^{\mathbf{c}_l}(\theta)$*, e.g. by grid search method.*

onto the signal subspace that is also defined in (3.2.17). The different steps of the TIME-SPECTRAL-MUSIC algorithm are summarized in Alg. 7.1, which is formulated in terms of ideal statistics but is completely the same for estimated statistics. Note that we have set the normalization constant in the nominator to one. Compare this algorithm with Algorithms 3.1 and 3.5 from Chapter 3, and note that it is a specific instance of Alg. 3.6 on page 129. Examples are provided in Section 7.1.4.

7.1.3 Maximum number of source DOA's or minimum number of sensors for given array configuration and conjugation tuple

In this section we show how the results obtained in Section 6.6, which hold for the general MIBI scenario, can be adapted to the currently considered ISBSL scenario with array response vectors that are parameterized by the Direction Of Arrival parameter $\theta \in \Theta$. To indicate the difference between the symbols used for general MIBI and those used for ISBSL, we write θ in the left subscript position of the symbols involved. For example, now ${}_\theta S_{\max}^{\mathbf{c}_l}(D)$ denotes the maximum number of identifiable source array response vectors and ${}_\theta D_{\min}^{\mathbf{c}_l}(S)$ the minimum required number of sensors. Likewise, ${}_\theta M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ denotes the number of unique sensor cumulant functions, ${}_\theta M_{\mathrm{u},D}^{\mathbf{c}_l} = {}_\theta M_{\mathrm{u},D}^{x,\mathbf{c}_l}$ the number of unique products in a generalized array response vector $\check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta)$ (7.1.20), ${}_\theta Q_{D,S}^{\mathbf{c}_l}$ the number of linearly independent equations in the system, and so on. Due to the dependence of the array response vectors on θ, both ${}_\theta S_{\max}^{\mathbf{c}_l}(D)$ and ${}_\theta D_{\min}^{\mathbf{c}_l}(S)$ now are also functions of the array configuration. Now we are in a position to formulate the ISBSL versions of the most important results from Section 6.6. To

start with, (6.6.7) becomes:

$$\begin{aligned} {}_{\theta}Q^{\mathbf{c}_l}_{D,S} &= \max\left(\operatorname{rank}_{\text{row}}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}(\tilde{\boldsymbol{\theta}})\right)\right) - \max\left(\operatorname{rank}_{\text{col}}\left(\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}(\tilde{\boldsymbol{\theta}})\right)\right) \\ &= {}_{\theta}M^{\mathbf{c}_l}_{\text{u},D} - S\,. \end{aligned} \tag{7.1.26}$$

Due to the inherent parameterization it directly follows that:

$$ {}_{\theta}M^{\mathbf{c}_l}_{\text{u},D} \;\leq\; M^{\mathbf{c}_l}_{\text{u},D}\,. \tag{7.1.27}$$

Given ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$ and $\mathbf{c}_l$, similarly to (6.6.8) and (6.6.9) (bounds on) the values of ${}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D)$ and ${}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)$ can be obtained from the following inequality:

$$\boxed{ {}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D} \;\leq\; {}_{\theta}M^{\mathbf{c}_l}_{\text{u},D} - S\,.} \tag{7.1.28}$$

For a given array configuration, number of sensors D, conjugation tuple $\mathbf{c}_l$, and function ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$, this yields the maximum number of identifiable array response vectors:

$$ {}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D) = {}_{\theta}M^{\mathbf{c}_l}_{\text{u},D} - {}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}\,. \tag{7.1.29}$$

Likewise, for a given array configuration, number of sources S, conjugation tuple $\mathbf{c}_l$, and function ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$, the minimum required number of sensors ${}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)$ can be obtained by solving the following inequality:

$$ {}_{\theta}Q^{\mathbf{c}_l}_{\text{min},{}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)} \;\leq\; {}_{\theta}M^{\mathbf{c}_l}_{\text{u},{}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)} - S\,. \tag{7.1.30}$$

Depending on the cumulant order and the form of the functions ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$ and ${}_{\theta}M^{\mathbf{c}_l}_{\text{u},D}$ this may be complicated to solve. However, as in Section 6.6, it can often easily be solved for the relevant range of values by using tabulated values.

We now give a plausibility argument for deriving the function ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$. Because the system (7.1.1) consists of nonlinear univariate equations, intuitively it is expected that one equation suffices for uniquely determining the solutions of the system, viz. the source DOA's. In fact, the simulations in the next sections (and many more simulations) show that this indeed is the case. Hence, if this heuristic is true, the function ${}_{\theta}Q^{\mathbf{c}_l}_{\text{min},D}$ identically equals 1, which means that one function $f^{\mathbf{c}_l}_{D,q}(\theta)$ suffices for defining the cost function $J^{\mathbf{c}_l}_D(\theta)$ and pseudo-spectrum $P^{\mathbf{c}_l}_D(\theta)$ in such a manner that all source DOA's can be determined uniquely from them. Under this assumption, it follows from (7.1.29) that ${}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D)$ is given by:

$$\boxed{ {}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D) = {}_{\theta}M^{\mathbf{c}_l}_{\text{u},D} - 1\,.} \tag{7.1.31}$$

Likewise, from (7.1.30) we see that ${}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)$ can be obtained from the following inequality:

$$\boxed{ S+1 \;\leq\; {}_{\theta}M^{\mathbf{c}_l}_{\text{u},{}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)}\,.} \tag{7.1.32}$$

Finally, we remark that in case ${}_{\theta}M^{\mathbf{c}_l}_{\text{u},D}$ is unknown or difficult to determine, by employing (7.1.27) the following upper bound for ${}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D)$ can be found:

$$ {}_{\theta}S^{\mathbf{c}_l}_{\text{max}}(D) \;\leq\; M^{\mathbf{c}_l}_{\text{u},D} - 1\,. \tag{7.1.33}$$

Likewise, a lower bound on ${}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)$ can be computed from the following inequality:

$$ S+1 \;\leq\; M^{\mathbf{c}_l}_{\text{u},{}_{\theta}D^{\mathbf{c}_l}_{\text{min}}(S)}\,. \tag{7.1.34}$$

7.1.4 TIME-SPECTRAL-MUSIC examples

In this section we will provide several examples of applying the TIME-SPECTRAL-MUSIC algorithm in Alg.7.1 to different DOA estimation scenarios.

7.1.4.1 Example 1: Ideal and estimated pseudo-spectra for scenario with 3 sensors, 6 sources, second order statistics, conjugation pair $(\circ, *)$, and employing 6 lags

In this example we consider a DOA estimation scenario with $D = 3$ sensors and $S = 6$ sources, and we employ complex second order statistics with conjugation tuple $\mathbf{c}_2 = (\circ, *)$. We compare the pseudo-spectra obtained from the ideal and estimated cumulant (correlation) matrices. The (randomly generated) DOA's of the six sources are given by $\tilde{\boldsymbol{\theta}} = [-95, -5, 38, 93, 139, 160\,]$ degrees and the source signals are assumed to be stationary. The signals and system are generated in such a way that all assumptions listed on page 299 of Section 6.1 are satisfied; in the next paragraphs, some specifics will be highlighted. As in Sections 3.2.2 and 3.3, two pseudo-spectra are computed for the considered scenario, viz. one using the ideal sensor correlation matrix $\mathbf{C}_3^{x,\circ *}$ computed using e.g. (6.2.96) with ideally known source DOA's and auto-cumulants, and the other using its estimate $\hat{\mathbf{C}}_3^{x,\circ *}$ obtained from $T = 40000$ samples of $\mathbf{x}[n]$. Note that the definition of the cumulant function pertaining to the current scenario reduces to the conventional correlation function for complex-valued signals. In the column- or lag-direction, the sensor correlation matrices contain the sensor correlation function values for $N = 6$ lags. The estimated correlation values are obtained by averaging over the available $T = 40000$ samples.

We will first explain how the source and noise signals are generated. Each complex-valued source signal is forced to have an 'AR(2) temporal structure' by generating it according to the following AR(2) model:

$$s_j[n] = -c_j^1\, s_j[n-1] - c_j^2\, s_j[n-2] + w_j[n] \qquad \forall\, 1 \leq j \leq 6\,, \tag{7.1.35}$$

where c_j^1 and c_j^2 are AR(2) regression coefficients and $w_j[n]$ is a sequence of independent uniformly and identically distributed complex random variables (see Section 3.3.6). The variance of $w_j[n]$ is chosen such that both the real and imaginary parts of $s_j[n]$ have variance 0.5, implying that each source signal $s_j[n]$ has unit variance. The regression coefficients are computed from the pole positions of the all-pole minimum-phase coloring filters, where the two poles of each filter are chosen such that they form a complex conjugated pair. Hence, the j-th filter can be written as:

$$H_j(z) = \frac{1}{(1 - p_j\, z^{-1})(1 - p_j^*\, z^{-1})} \triangleq \frac{1}{1 + c_j^1\, z^{-1} + c_j^2\, z^{-2}}$$

with the AR(2) regression coefficients defined by:

$$c_j^1 \triangleq -2\Re\{p_j\} \quad \text{and} \quad c_j^2 \triangleq |p_j|^2\,. \tag{7.1.36}$$

Note that the power spectral density function of $s_j[n]$ in the z-domain is given by:

$$P_{s_j}(z) = \sigma_{w_j}^2 P_{w_j}(z) = \sigma_{w_j}^2 H_j(z) H_j^*\left(\frac{1}{z^*}\right)\,. \tag{7.1.37}$$

The pole pairs of the six source signals are chosen as depicted in Fig. 7.1. From the fact that the pairs of AR(2) coefficients are different, it follows that their corresponding auto-correlation functions and power spectral densities are linearly independent.

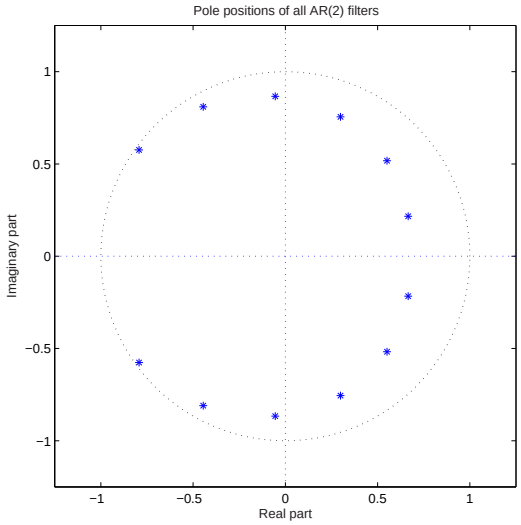

Figure 7.1: *Example 1: Source pole positions for $S = 6$.*

As in Sections 3.2.2 and 3.3.6, the noise signals are generated as realizations of white complex circular Gaussian stochastic processes. We have chosen the noise variance equal to the source variance, i.e. $(\sigma^\nu)^2 = (\sigma^s)^2 = 1$. Note that this results in a signal-to-noise power ratio S at each sensor, i.e. the ratio between the noise-free and noise components in each sensor signal equals $10\log_{10}(S) = 7.8$ dB. Since all noise signals are white, the Noise-Free ROS $\Omega_k^{s\backslash\nu,\circ *}$ in the lag domain equals $\Omega_k^{s\backslash\nu,\circ *} = \{k \in \mathbb{Z} \mid k \neq 0\}$. In the sequel of this example we choose $\Omega_k^{s\backslash\nu,\circ *} = \{1,\ldots,6\}$.

Now we will apply the theory developed in Section 7.1.3 to the current example without firstly determining an explicit expression for the number ${}_\theta M_{\mathrm{u},D}^{\circ *}$ of unique elements of the generalized array response vector. Since we do not determine ${}_\theta M_{\mathrm{u},D}^{\circ *}$ beforehand, we cannot employ (7.1.32) and have to resort to (7.1.34) for obtaining a lower bound on ${}_\theta D_{\min}^{\circ *}(S)$. Firstly, from (6.2.43) it follows that $M_{\mathrm{u},D}^{\circ *} = (D)^2$. Substituting this into (7.1.34) gives the inequality $\left({}_\theta D_{\min}^{\circ *}(S)\right)^2 \geq S+1$, yielding the following minimum bound on ${}_\theta D_{\min}^{\circ *}(S)$:

$$ {}_\theta D_{\min}^{\circ *}(S) = \left\lceil \sqrt{S+1} \right\rceil \ . $$

For the current example with $S = 6$ this gives ${}_\theta D_{\min}^{\circ *}(S) \geq 3$. Simulations reveal that indeed this is a correct and tight minimum bound on the required number of sensors, i.e. ${}_\theta D_{\min}^{\circ *}(S) = 3$. In addition, simulations reveal that for this number of sensors the maximum achievable row rank $\max\left(\mathrm{rank}_{\mathrm{row}}\left(\mathbf{A}_{3,\diamond}^{\circ *}(\tilde{\boldsymbol{\theta}})\right)\right)$ equals 7. Hence, there can be at most 7 unique elements in the generalized array response vector, in which case there are also 7 unique sensor cumulant functions. This implies that with 3 sensors and 6 sources the number ${}_\theta Q_{3,6}^{\circ *}$ is given by (7.1.26) as ${}_\theta Q_{3,6}^{\circ *} = 1$ (note that $\mathrm{rank}\left(\mathbf{A}_{3,\diamond}^{\circ *}(\tilde{\boldsymbol{\theta}})\right) = S = 6$ because the source DOA's are different and the array response matrix is assumed to be unambiguous), which means that essentially there is only one underlying function available for defining the null- and pseudo-spectra. Therefore, ${}_\theta S_{\max,D}^{\circ *} = 6$ for $D = 3$, and ${}_\theta Q_{3,6}^{\circ *} = {}_\theta Q_{\min,3}^{\circ *} = 1$.

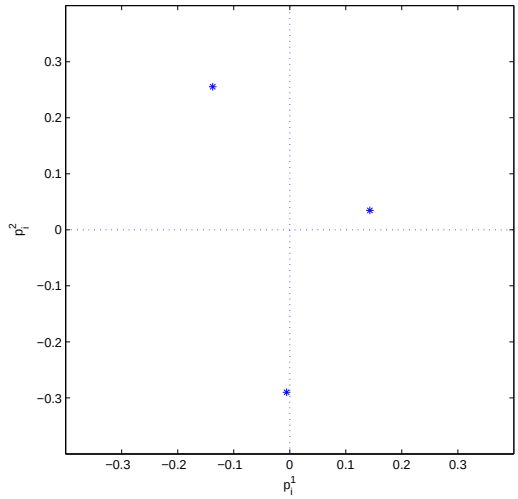

Figure 7.2: *Example 1: Normalized sensor positions for $D = 3$.*

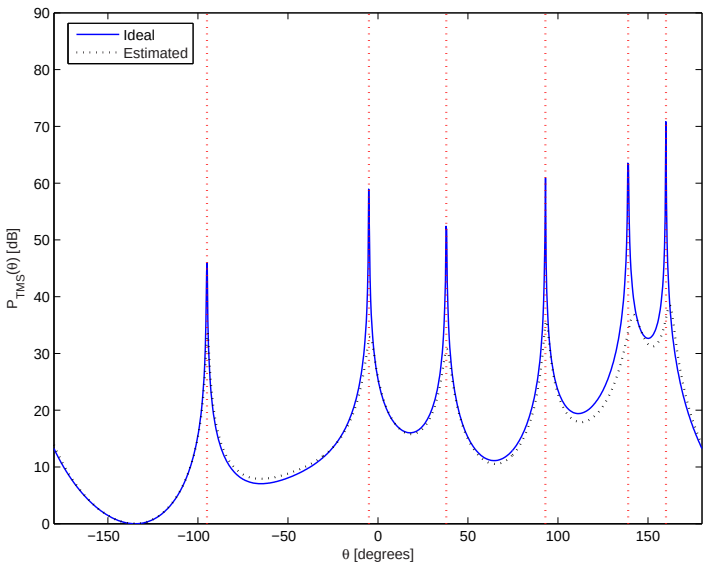

Figure 7.3: *Example 1: Ideal TIME-SPECTRAL-MUSIC spectrum $P_3^{\circ *}(\theta)$ and its estimate $\hat{P}_3^{\circ *}(\theta)$ for $D = 3$, $S = 6$ and $\mathbf{c}_2 = (\circ, *)$.*

A proper array configuration yielding an unambiguous array response matrix is depicted in Fig. 7.2. For $1 \leq i \leq 3$ its normalized sensor positions $\tilde{\mathbf{p}}_{i,\mathrm{n}} = \left[p_{i,\mathrm{n}}^1 \;\; p_{i,\mathrm{n}}^2\right]$ are given by:

$$\tilde{\mathbf{p}}_{i,\mathrm{n}} = r\left[\sin\left(\tfrac{4}{3}(i-1)\right) - \tfrac{\sqrt{3}\sin(2)\sin(4/3)}{18\sin 2/3} \quad \cos\left(\tfrac{4}{3}(i-1)\right) - \tfrac{\sqrt{3}\sin(2)\cos(4/3)}{18\sin 2/3}\right] \qquad (7.1.38)$$

with $r = \frac{1}{6}\sqrt{3}$. The mean of these positions, i.e. the 'center of gravity' lies in the origin. We remark that the configuration of an array is very important. For a ULA with $D = 3$ sensors, proper cost functions and pseudo-spectra cannot be obtained because for the considered

scenario a ULA does not utilize all its possible degrees of freedom sufficiently (the sensors are positioned on a line having dimension one whereas two dimensions are available). The ideal and estimated pseudo-spectra are computed using Alg. 7.1 with $\mathbf{C}_3^{x,\circ *}$ and $\hat{\mathbf{C}}_3^{x,\circ *}$ respectively. The resulting spectra are shown in Fig. 7.3. We leave it to the reader to assess their properties.

7.1.4.2 Example 2: Comparing ideal pseudo-spectra for cumulant orders $l = 1, 2, 3$ and conjugation tuple $\mathbf{c}_l = (\circ)_l$, i.e. no conjugations

In this section our main goal is to provide theoretical insight. For this reason, here we only present results computed from ideal subspace matrices under the assumption that the source signals have sufficient temporal structure, e.g. AR(2) temporal structure with different pole pairs, and sufficient statistical structure for the considered order, e.g. for $l = 3$ they are obtained from AR(2) filtering white i.i.d. excitation sequences with an odd probability density function such as a Rayleigh or Gamma pdf. Again, the noise signals are assumed to be realizations of white complex circular Gaussian stochastic processes. Although we only present the results obtained from exactly known sensor cumulants, it can be shown that the pseudo-spectra obtained by using estimated cumulants converge to the ideal pseudo-spectra when consistent estimators are used. In addition, for simplicity, we assume that the arguments of the cumulant functions are not conjugated, i.e. the conjugation tuple for order l is given by $\mathbf{c}_l = (\circ)_l$. In order to compare the spectra for different cumulant orders l, two sets of simulations employing the same sensor configurations are performed with $l = 1, 2$ and 3. In the first set of simulations, the number of sources is the same for each $1 \leq l \leq 3$ and equals $S = 8$ with the source DOA's given by $\tilde{\boldsymbol{\theta}} = [-173, -97, -16, -5, 38, 94, 141, 162]$ degrees. In the second set of simulations, the number of sources is also the same for each $1 \leq l \leq 3$ but now equals $S = 9$ with the source DOA's given by $\tilde{\boldsymbol{\theta}} = [-173, -97, -16, -5, 38, 94, 116, 141, 162]$ degrees.

As we have explained in Section 7.1.3 and the previous example, depending on the array configuration and chosen conjugation tuple a certain minimum number of sensors is required for identifying a given number of source array response vectors. Since for the current example the conjugation tuple is given by $(\circ)_l$ this number is denoted by ${}_{\theta}D_{\text{min}}^{(\circ)_l}(S)$. We will now derive this number for a particular array configuration, viz. similarly to (7.1.38) for all $1 \leq i \leq D$ the normalized sensor positions $\tilde{\mathbf{p}}_{i,\text{n}} = \begin{bmatrix} p_{i,\text{n}}^1 & p_{i,\text{n}}^2 \end{bmatrix}$ are chosen as:

$$\tilde{\mathbf{p}}_{i,\text{n}} = r \left[\sin\left(\tfrac{4}{D}\left(i - \tfrac{1}{2}\right) - 2\right) \quad \cos\left(\tfrac{4}{D}\left(i - \tfrac{1}{2}\right) - 2\right) - \tfrac{r \sin(2)}{D \sin(2/D)} \right], \tag{7.1.39}$$

where $r = \frac{1}{4}$. Again, the mean (normalized) position vector $\frac{1}{D}\sum_{i=1}^{D} \tilde{\mathbf{p}}_i$ lies in the origin. It turns out that for the considered range of D, this array configuration yields an unambiguous (generalized) array response matrix. Simulations reveal that with this array configuration for $1 \leq l \leq 4$ the number ${}_{\theta}M_{\text{u},D}^{(\circ)_l}$ of unique elements of the generalized array response vector for the current Source Localization scenario equals the number $M_{\text{u},D}^{(\circ)_l}$ of unique elements of the generalized array response vector for a general MIBI scenario. Hence, equation (7.1.27) holds with equality, i.e. ${}_{\theta}M_{\text{u},D}^{(\circ)_l} = M_{\text{u},D}^{(\circ)_l}$, where $M_{\text{u},D}^{(\circ)_l}$ is given by (6.2.33) or any of equations (6.2.41) through (6.2.44).

Table 7.1: *${}_\theta S_{\text{max},D}^{(\circ)_l}$ values for $1 \leq l, D \leq 4$.*

$l \backslash D$	1	2	3	4
1	0	1	2	3
2	0	2	5	9
3	0	3	9	19
4	0	4	14	34

Before determining ${}_\theta D_{\text{min}}^{(\circ)_l}(S) \equiv {}_\theta D_{\text{min}}^{l}(S)$, we first note from (7.1.31) that for $1 \leq l \leq 4$:

$$ {}_\theta S_{\text{max},D}^{(\circ)_l} = \binom{l+D-1}{l} - 1 = \frac{(l+D-1)!}{l!(D-1)!} - 1 \,. \tag{7.1.40} $$

Table 7.1 lists the values of ${}_\theta S_{\text{max},D}^{(\circ)_l}$ for $1 \leq l, D \leq 4$. Comparing (7.1.40) and Table 7.1 on the one hand with (6.6.11) and Table 6.3 obtained for the example at the end of Section 6.6 on the other hand, reveals the differences with the equivalent general MIBI case. Since $Q_{\text{min},D}^{(\circ)_l} = D - 1$ for the example in Section 6.6, whereas ${}_\theta Q_{\text{min},D}^{(\circ)_l} = 1$ for the current example, the difference between the maximum numbers of source array response vectors that can be identified for the current example and the example at the end of Section 6.6 equals $D - 2$. This implies that for $D \geq 3$, ${}_\theta S_{\text{max},D}^{(\circ)_l}$ for the current DOA scenario is larger than $S_{\text{max},D}^{(\circ)_l}$ for the general MIBI scenario.

Now we turn our attention to the computation of ${}_\theta D_{\text{min}}^{(\circ)_l}(S)$. Because ${}_\theta M_{\text{u},D}^{(\circ)_l} = M_{\text{u},D}^{(\circ)_l}$ for $1 \leq l, D \leq 4$, ${}_\theta D_{\text{min}}^{(\circ)_l}(S)$ can be computed from either (7.1.32) or (7.1.34), which are now equivalent. If satisfied, these inequalities guarantee that at least one function $f_{D,q}^{(\circ)_l}(\theta)$ will be available for defining the cost function $J_D^{(\circ)_l}(\theta)$ and pseudo-spectrum $P_D^{(\circ)_l}(\theta)$. Substituting (6.2.33) into (7.1.32), it follows that for $1 \leq l \leq 4$ and for a given value of S, the value of ${}_\theta D_{\text{min}}^{(\circ)_l}(S)$ can be obtained by solving the following inequality:

$$ 1 \leq \binom{l + {}_\theta D_{\text{min}}^{(\circ)_l}(S) - 1}{l} - S \,. \tag{7.1.41} $$

We now show how this can be done for $1 \leq l \leq 4$; see also the example at the end of Section 6.6. For convenience, at some places we will temporarily write D instead of ${}_\theta D_{\text{min}}^{(\circ)_l}(S)$. To start with, substituting $l = 1$ into (7.1.41) yields:

$$ 1 \leq D - S \qquad \Longrightarrow \qquad D \geq S + 1 \,. $$

Hence:

$$ {}_\theta D_{\text{min}}^{(\circ)_1}(S) = S + 1 \,. \tag{7.1.42} $$

which agrees with the results in the first row of Table 7.1. Now substituting $l = 2$ into (7.1.41) gives the inequality:

$$ 1 \leq \tfrac{1}{2}(D+1)D - S \,. \tag{7.1.43} $$

Solving this inequality gives the following formula for ${}_\theta D_{\text{min}}^{(\circ)_2}(S)$ as a function of S:

$$ {}_\theta D_{\text{min}}^{(\circ)_2}(S) = \left\lceil \tfrac{1}{2}\big(-1 + \sqrt{8S+9}\big) \right\rceil , \tag{7.1.44} $$

which agrees with the results in the second row of Table 7.1. Substituting $l = 3$ into (7.1.41) yields:

$$1 \leq \tfrac{1}{6}(D+2)(D+1)D - S\,. \tag{7.1.45}$$

Solving this inequality gives the following formula for ${}_\theta D^{(\circ)_3}_{\min}(S)$ as a function of S:

$${}_\theta D^{(\circ)_3}_{\min}(S) = \left\lceil \frac{\sqrt[3]{3}\, r(S)}{3} + \frac{1}{\sqrt[3]{3}\, r(S)} - 1 \right\rceil, \tag{7.1.46}$$

where the function $r(S)$ is defined by:

$$r(S) \triangleq \sqrt[3]{27(S+1) + \sqrt{3}\sqrt{243(S+2)S + 242}}\,.$$

This result agrees with the third row of Table 7.1. Finally, substituting $l = 4$ into (7.1.41) yields the inequality:

$$1 \leq \tfrac{1}{24}(D+3)(D+2)(D+1)D - S\,. \tag{7.1.47}$$

Solving this gives the following formula for ${}_\theta D^{(\circ)_4}_{\min}(S)$ as a function of S:

$${}_\theta D^{(\circ)_4}_{\min}(S) = \left\lceil \tfrac{1}{2}\left(-3 + \sqrt{5 + 4\sqrt{24S+25}}\right) \right\rceil, \tag{7.1.48}$$

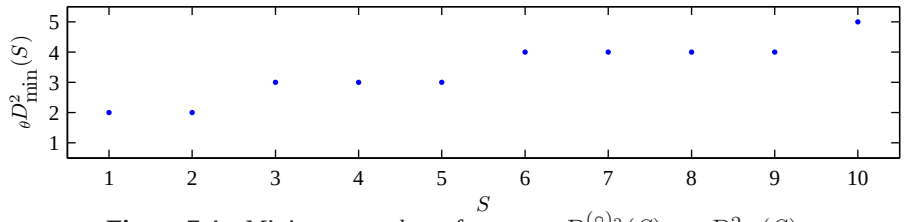

Figure 7.4: *Minimum number of sensors* ${}_\theta D^{(\circ)_2}_{\min}(S) \equiv {}_\theta D^2_{\min}(S)$.

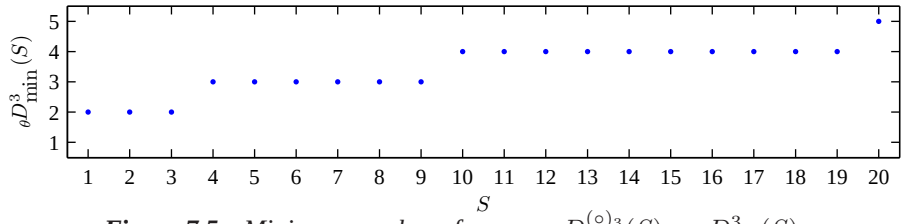

Figure 7.5: *Minimum number of sensors* ${}_\theta D^{(\circ)_3}_{\min}(S) \equiv {}_\theta D^3_{\min}(S)$.

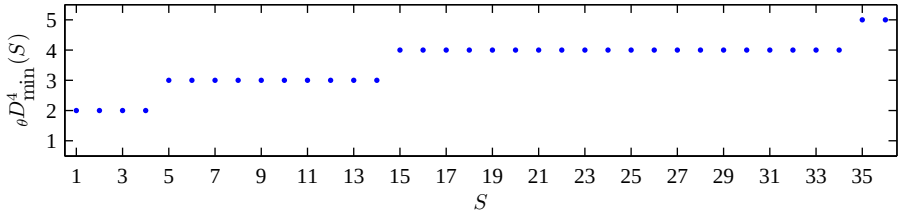

Figure 7.6: *Minimum number of sensors* ${}_\theta D^{(\circ)_4}_{\min}(S) \equiv {}_\theta D^4_{\min}(S)$.

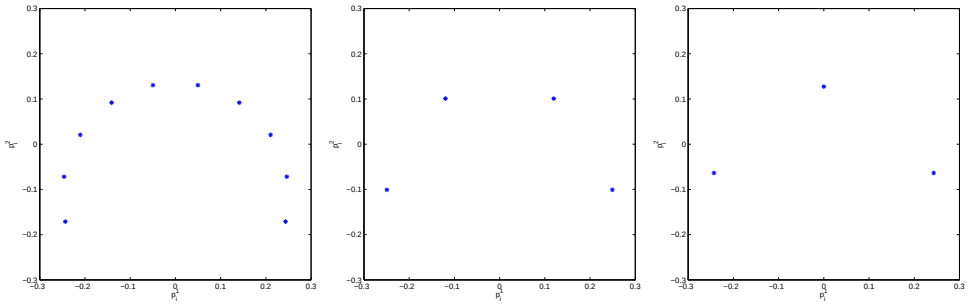

Figure 7.7: *Example 2: Normalized array configuration with $D = 10,\ 4$ and 3 sensors for $l = 1,\ 2$ and 3 respectively.*

which agrees with the fourth row of Table 7.1. In Figures 7.4, 7.5 and 7.6, ${}_{\theta}D_{\min}^{(\circ)_l}(S)$ is plotted as a function of S for $l = 2, 3$ and 4 respectively.

For the first set of simulations with $S = 8$, (7.1.42), (7.1.44) and (7.1.46), or Figures 7.4, 7.5 and 7.6, or Table 7.1 yield:

$$ {}_{\theta}D_{\min}^{(\circ)_1}(S) = 9, \qquad {}_{\theta}D_{\min}^{(\circ)_2}(S) = 4, \qquad {}_{\theta}D_{\min}^{(\circ)_3}(S) = 3\,. $$

Simulations reveal that for the chosen array configuration these numbers indeed are tight bounds. From (7.1.40), Figures 7.4, 7.5 and 7.6, or Table 7.1 we see that with $D = 4$ for $l = 2$ and with $D = 3$ for $l = 3$ we can identify a maximum of ${}_{\theta}S_{\max,4}^{(\circ)_2} = {}_{\theta}S_{\max,3}^{(\circ)_3} = 9$ source DOA's. Since we are considering a scenario with $S = 8$, this means that two underlying functions are employed for defining the null- and pseudo-spectra. In order to make a fair comparison, we choose $D = 10$ for $l = 1$. Hence, this way for $l = 1, 2, 3$ the number of underlying linearly independent functions available for averaging in the definitions of the cost functions and pseudo-spectra are given by ${}_{\theta}Q_{D,S}^{(\circ)_l} = \binom{l+D-1}{l} - S$ (7.1.26) as ${}_{\theta}Q_{10,8}^{(\circ)_1} = {}_{\theta}Q_{4,8}^{(\circ)_2} = {}_{\theta}Q_{3,8}^{(\circ)_3} = 2$. The normalized sensor array configurations for the considered combinations of l and D can now be computed from (7.1.39) and are shown in Fig. 7.7. The TIME-MUSIC pseudo-spectra $P_{10}^{(\circ)_1}(\theta)$, $P_{4}^{(\circ)_2}(\theta)$ and $P_{3}^{(\circ)_3}(\theta)$ computed by means of Alg. 7.1 are plotted in Fig. 7.8. The source DOA's are indicated by the dotted vertical lines. For easy comparison, the minima of all spectra have been normalized to 0 dB. The figure shows that all spectra exhibit sharp peaks at the source DOA's and that no spurious peaks are present.

For the second set of simulations with $S = 9$, the TIME-MUSIC pseudo-spectra $P_{10}^{(\circ)_1}(\theta)$, $P_{4}^{(\circ)_2}(\theta)$ and $P_{3}^{(\circ)_3}(\theta)$ are depicted in Fig. 7.9. Note that the sensor positions again are given by (7.1.39) and depicted in Fig. 7.7. Now, for $l = 1, 2, 3$ the number of underlying linearly independent functions available for averaging in the definitions of the cost functions and pseudo-spectra are given by (7.1.26) as ${}_{\theta}Q_{10,9}^{(\circ)_1} = {}_{\theta}Q_{4,9}^{(\circ)_2} = {}_{\theta}Q_{3,9}^{(\circ)_3} = 1$. It can be seen that still all source DOA's are resolved correctly. From many other similar simulations, it can be concluded that in general one equation $f_{D,q}^{(\circ)_l}(\theta)$ is sufficient to resolve ${}_{\theta}S_{\max,D}^{(\circ)_l} = \binom{D+l-1}{l} - 1$ DOA's uniquely from the pseudo-spectrum $P_{D}^{(\circ)_l}(\theta)$.

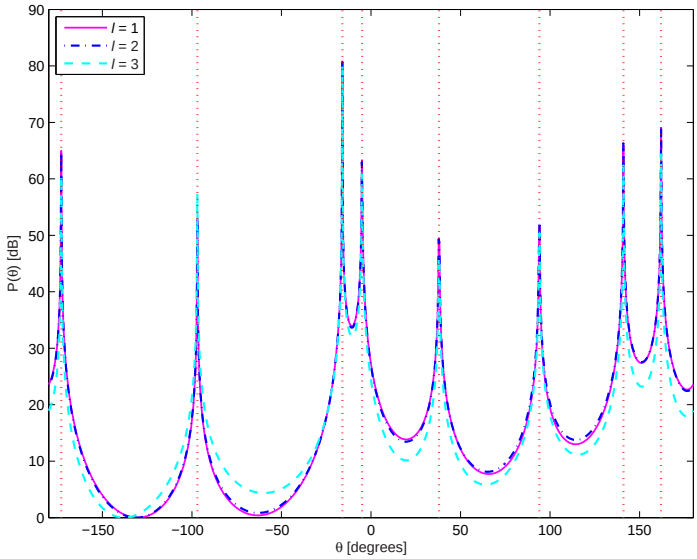

Figure 7.8: *Example 2: Ideal TIME-MUSIC spectra* $P_{10}^{(\circ)_1}(\theta)$, $P_{4}^{(\circ)_2}(\theta)$ *and* $P_{3}^{(\circ)_3}(\theta)$ *for* $S = 8$.

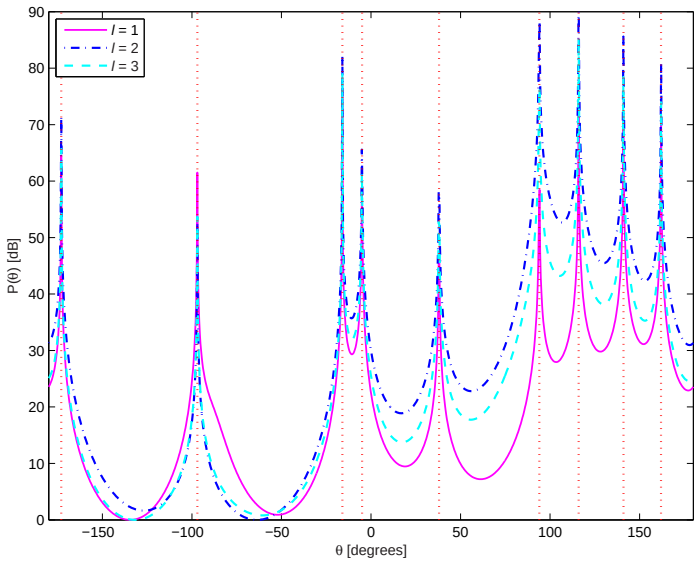

Figure 7.9: *Example 2: Ideal TIME-MUSIC spectra* $P_{10}^{(\circ)_1}(\theta)$, $P_{4}^{(\circ)_2}(\theta)$ *and* $P_{3}^{(\circ)_3}(\theta)$ *for* $S = 9$.

7.1.4.3 Example 3: Ideal and estimated pseudo-spectra for scenario with 2 sensors, 3 sources, third order statistics, conjugation tuple $(*,\circ,\circ)$, and employing 4 lags

Similarly to Example 1 of this section, here we compare the pseudo-spectra obtained from ideal and estimated sensor cumulant matrices. Now we consider a DOA estimation scenario with $D = 2$ sensors, $S = 3$ zero-mean stationary sources, and we employ third order statistics with conjugation tuple $(*,\circ,\circ)$. We will demonstrate later on in this example that two sensors suffice to resolve three source DOA's. Note that for zero-mean signals the employed sensor cumulant functions defined in (6.2.1) now become:

$$\kappa_{\mathbf{i}_3}^{x,\mathbf{c}_3}[\mathbf{n}_3] \triangleq \operatorname{cum}\Big(x_{i_1}[n_1], x_{i_2}[n_2], (x_{i_3}[n_3])^*\Big) = E\Big\{x_{i_1}[n_1]\, x_{i_2}[n_2]\, (x_{i_3}[n_3])^*\Big\}\,.$$

The DOA's of the three sources are given by $\tilde{\boldsymbol{\theta}} = [-35, -11, 47\,]$ degrees. Furthermore, the normalized sensor positions are given by $\tilde{\mathbf{p}}_{1,\mathrm{n}} = \big[-1/4 \;\; 0\big]$ and $\tilde{\mathbf{p}}_{2,\mathrm{n}} = \big[1/4 \;\; 0\big]$, i.e. the sensors are positioned symmetrically w.r.t. the origin on the horizontal axis at half a wavelength distance from each other. Again, the signals and system are generated in such a way that the assumptions listed on page 299 of Section 6.1 are satisfied; in the next paragraphs, some specifics will be highlighted. As in Example 1 two pseudo-spectra will be computed for the considered scenario, viz. one using the ideal sensor cumulant matrix $\mathbf{C}_2^{x,\circ\circ *}$ computed using e.g. (6.2.96) with ideally known source DOA's and auto-cumulants, and the other using its estimate $\hat{\mathbf{C}}_2^{x,\circ\circ *}$ obtained by averaging over $T = 100000$ samples of $\mathbf{x}[n]$. In the column-direction the sensor cumulant matrices contain $N = 4$ sensor cumulant function values.

We will first point out how the source and noise signals are generated. Each complex-valued source signal is forced to have unit variance and an 'ALLPASS(1) temporal structure' by generating it according to the following ALLPASS(1) model:

$$s_j[n] = p_j\, s_j[n-1] + p_j^* w_j[n] - w_j[n-1] \qquad \forall\, 1 \leq j \leq 3\,, \tag{7.1.49}$$

where p_j and $1/p_j^*$ are the ALLPASS(1) pole and zero associated with the j-th source signal respectively, and $w_j[n]$ is a sequence of independent and identically distributed complex random variables that will be described shortly. Note that the j-th transfer function used for generating $s_j[n]$ from $w_j[n]$ can be written as follows:

$$H_j(z) = \frac{p_j^* - z^{-1}}{1 - p_j\, z^{-1}}\,.$$

The pole-zero pairs of the three source signals are chosen as depicted in Fig. 7.10 on the next page. From the fact that these pairs are (sufficiently) different, it follows that their corresponding auto-cumulant functions are linearly independent. Because we want to exploit third order statistics the distribution of the source signal samples should be odd. In order to achieve this, the random variables of the real and imaginary parts of each sequence $w_j[n]$ are drawn from a pdf with odd symmetry, viz. the Gamma distribution, with the means removed. See Section 6.3.1.1 for the definition and properties of the Gamma pdf. The real and imaginary parts $w_{\mathrm{r}}[n]$ and $w_{\mathrm{i}}[n]$ respectively are generated independently of each other, and the variance of the imaginary part equals four times that of the real part in order to further increase the statistical diversity of the source signals. Because the source signal $s_j[n]$ corresponding to the sequence $w_j[n]$ is obtained by exciting a unit-gain all-pass filter with $w_j[n]$, the variance of $s_j[n]$ equals that of $w_j[n]$. Since we have chosen to make the variance of the imaginary part of each sequence $w_j[n]$ four times as large as that of the real part, the scale parameter λ_{i} required for generating the imaginary part equals twice the scale parameter λ_{r} required for

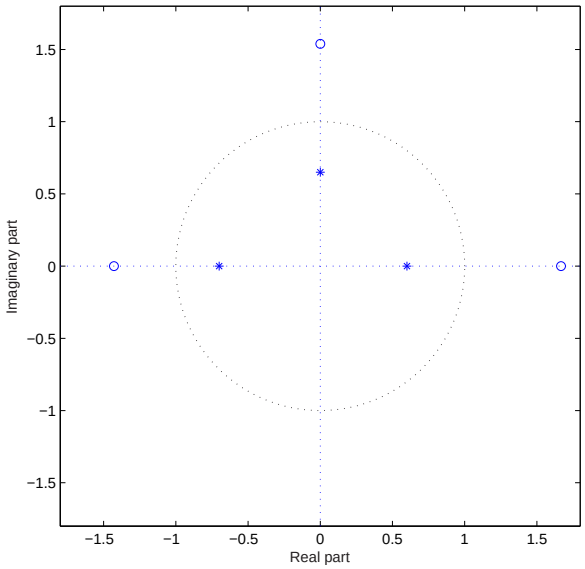

Figure 7.10: *Example 3: Pole-zero pairs of 'ALLPASS(1)' source signals for $S = 3$.*

generating the real part. Hence, in order to obtain unit variance sequences $w_j[n]$ and $s_j[n]$, the equality $\alpha(\lambda_\mathrm{r})^2 + \alpha(\lambda_\mathrm{i})^2 = 5\alpha(\lambda_\mathrm{r})^2 = 1$ must hold. As in Section 6.3.1.1, we choose $\alpha = 2$. It then follows that $\lambda_\mathrm{r} = 1/\sqrt{10}$ and $\lambda_\mathrm{i} = 2/\sqrt{10}$. Substituting these values into (6.3.7) gives the pdfs of w_r and w_i as follows:

$$p_{w_\mathrm{r}}(w_\mathrm{r}) = \sqrt{10}\,\epsilon\big(w_\mathrm{r} + \tfrac{2}{\sqrt{10}}\big)\,\big(\sqrt{10}\,w_\mathrm{r} + 2\big)\;e^{-\left(\sqrt{10}\,w_\mathrm{r}+2\right)}$$

and:

$$p_{w_\mathrm{i}}(w_\mathrm{i}) = \tfrac{1}{2}\sqrt{10}\,\epsilon\big(w_\mathrm{i} + \tfrac{4}{\sqrt{10}}\big)\,\big(\tfrac{1}{2}\sqrt{10}\,w_\mathrm{i} + 2\big)\;e^{-\left(\frac{1}{2}\sqrt{10}\,w_\mathrm{i}+2\right)}$$

respectively. Fig. 7.11 depicts these probability density functions.

As in Example 1, the noise signals are generated as realizations of white complex circular Gaussian stochastic processes. Again, we have chosen the noise variance equal to the source variance, which now results in a signal-to-noise power ratio of 4.8 dB at each sensor. Since all noise signals are white, the Noise-Free ROS $\Omega_k^{s\setminus\nu,\circ *}$ in the domain of lag pairs equals $\Omega_{k_1 k_2}^{s\setminus\nu,\circ\circ *} = \{(k_1, k_2) \in \mathbb{Z} \mid k_1, k_2 \neq 0\}$. For this example we have chosen $\Omega_{k_1 k_2}^{s\setminus\nu,\circ\circ *} = \{(0,0),(0,1),(1,0),(1,1)\}$. Hence, with this Noise-Free ROS the '(i_1, i_2, i_3)-th' sensor cumulant row vector $\tilde{\boldsymbol{\kappa}}_{i_1 i_2 i_3}^{x,\circ\circ *}$ of $\mathbf{C}_2^{x,\circ\circ *}$ contains the values $\kappa_{i_1 i_2 i_3}^{x,\circ\circ *}[n,n,n]$, $\kappa_{i_1 i_2 i_3}^{x,\circ\circ *}[n,n,n-1]$, $\kappa_{i_1 i_2 i_3}^{x,\circ\circ *}[n,n-1,n]$, and $\kappa_{i_1 i_2 i_3}^{x,\circ\circ *}[n,n-1,n-1]$ (and similarly for $\hat{\mathbf{C}}_2^{x,\circ\circ *}$).

Similarly to the previous two examples, we will now apply the theory developed in Section 7.1.3 to the current example. In Section 7.3, it is shown that for a ULA the number of unique elements ${}_\theta M_{\mathrm{u},D}^{\mathbf{c}_l}$ of the generalized array response vector for any conjugation tuple equals $l(D-1)+1$. Hence, for the current scenario ${}_\theta M_{\mathrm{u},D}^{\circ\circ *} = 3(D-1)+1$. Employing (7.1.32) yields the inequality $S+1 \leq 3(D-1)+1$. From this we easily obtain the following minimum bound on ${}_\theta D_{\min}^{\circ\circ *}(S)$:

$${}_\theta D_{\min}^{\circ\circ *}(S) \geq \left\lceil \frac{S+3}{3} \right\rceil .$$

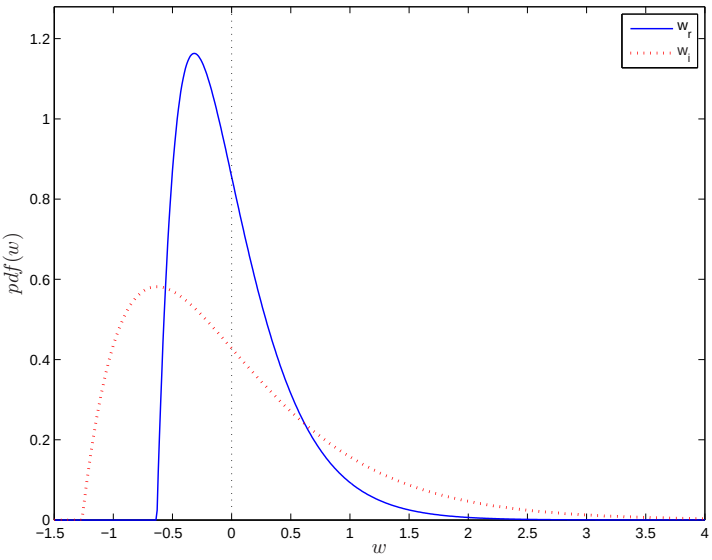

Figure 7.11: *Example 3: Probability density functions of* w_{r} *and* w_{r}.

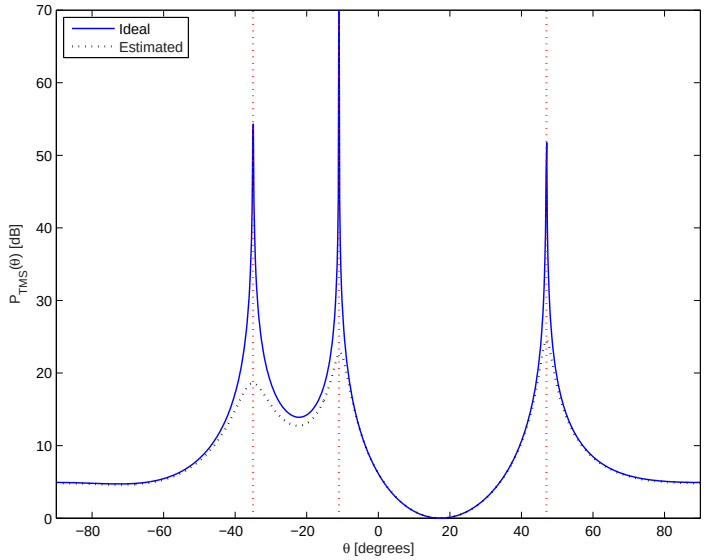

Figure 7.12: *Example 3: Ideal TIME-SPECTRAL-MUSIC spectra* $P_2^{\circ\circ *}(\theta)$ *and its estimate* $\hat{P}_2^{\circ\circ *}(\theta)$ *for* $D = 2$, $S = 3$ *and* $\mathbf{c}_3 = (*, \circ, \circ)$.

For $S = 3$ this gives ${}_{\theta}D_{\min}^{\circ\circ *}(S) \geq 2$. Simulations reveal that this is a correct and tight minimum bound, i.e. ${}_{\theta}D_{\min}^{\circ\circ *}(S) = 2$, and that for this value of D the maximum achievable row rank $\max\left(\operatorname{rank}_{\text{row}}\left(\mathbf{A}_{2,\diamond}^{\circ\circ *}(\tilde{\boldsymbol{\theta}})\right)\right)$ equals 4. Note that this means that there can be at most 4 unique elements in the generalized array response vector, in which case there are also 4 unique sensor cumulant functions. This implies that with 2 sensors and 3 sources the number

${}_{\theta}Q_{2,3}^{\circ\circ *}$ is given by (7.1.26) as ${}_{\theta}Q_{2,3}^{\circ\circ *} = 1$ ($\text{rank}\left(\mathbf{A}_{2,\diamond}^{\circ\circ *}(\tilde{\boldsymbol{\theta}})\right) = S = 3$ because the source DOA's are different), which means that essentially there is only one 'true' underlying function available for defining the cost functions and pseudo-spectra pertaining to the current example, and this turns out to be sufficient. The ideal and estimated pseudo-spectra are computed using Alg. 7.1 with $\mathbf{C}_2^{x,\circ\circ *}$ and $\hat{\mathbf{C}}_2^{x,\circ\circ *}$ respectively, and are shown in Fig. 7.12.

7.2 One-parameter Direction Of Arrival estimation of far field sources with ULA-TIME-SPECTRAL-MUSIC

With the results of the previous section in place, we can now straightforwardly and concisely derive and formulate the equivalent results for a Uniform Linear Array (ULA). As we have explained in Section 3.1.2, a ULA can be considered as a special case of the planar array because it is degenerate version of it. The sensors are assumed to be located on the positive p^1-axis with the position of the i-th sensor given by $\tilde{\mathbf{p}}_i = \begin{bmatrix} p_i^1 & p_i^2 \end{bmatrix} = \begin{bmatrix} (i-1)d & 0 \end{bmatrix}$ with $1 \leq i \leq D$. For more details, see Section 3.1.2.

7.2.1 Structure of system and equations

From (3.1.12) it follows that now the i-th element $z_i(\theta)$ of $\mathbf{z}(\theta)$ is given by:

$$z_i(\theta) = \exp\big(\jmath\, \phi_i(\theta)\big) = \exp\big(\jmath\,(i-1)\xi(\theta)\big) \qquad \forall\, \theta \in \Theta, \quad \forall\, 1 \leq i \leq D\,, \tag{7.2.1}$$

where $\xi(\theta)$ is defined in (3.1.10), which is repeated here for convenience:

$$\xi(\theta) \triangleq 2\pi \frac{d}{\lambda_{\text{c}}} \sin(\theta) = 2\pi d_{\text{n}} \sin(\theta) \qquad \forall\, \theta \in \Theta\,. \tag{7.2.2}$$

Substituting $\phi_i(\theta) = (i-1)\xi(\theta)$ into (7.1.8) and using (7.2.2) yields:

$$\check{z}_{\mathbf{i}_l}^{\mathbf{c}_l}(\theta) = \exp\left(\jmath \sum_{m=1}^{l} \langle -1 \rangle^{c_m} \,(i_m - 1)\,\xi(\theta)\right) = \exp\Big(\jmath\, 2\pi\, t_{\mathbf{i}_l,\text{n}}^{\mathbf{c}_l} \sin(\theta)\Big)\,, \tag{7.2.3}$$

where:

$$t_{\mathbf{i}_l,\text{n}}^{\mathbf{c}_l} \triangleq \sum_{m=1}^{l} \langle -1 \rangle^{c_m} (i_m - 1)\, d_{\text{n}} = \left(\sum_{m=1}^{l} \langle -1 \rangle^{c_m} (i_m - 1)\right)\, d_{\text{n}}\,. \tag{7.2.4}$$

Note the resemblance between (7.2.3) on the one hand, and (7.1.6) and (7.1.10) on the other hand, which implies that $t_{\mathbf{i}_l,\text{n}}^{\mathbf{c}_l}$ can be seen as a conjugation-compensated mean normalized position. Similarly to (7.1.12), the functions in the 'ULA system' can now be written as:

$$\boxed{f_{D,q}^{\mathbf{c}_l}(\theta) \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}_{\text{t},D}^{l}} \varphi_q^{\mathbf{i}_l} \, \exp\Big(\jmath\, 2\pi\, t_{\mathbf{i}_l,\text{n}}^{\mathbf{c}_l} \sin(\theta)\Big) = \tilde{\boldsymbol{\varphi}}_q\, \check{\mathbf{z}}_D^{\mathbf{c}_l}(\theta) \qquad \forall\, \theta \in \Theta\,, \quad \forall\, q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}\,.} \tag{7.2.5}$$

Again, since (6.2.65) states that $f_{D,q}^{\mathbf{c}_l}(\theta^j) = 0$ for all $q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}$ and all $1 \leq j \leq S$, the source DOA's can be estimated by locating the zeros of the system $\{f_{D,q}^{\mathbf{c}_l}(\theta) = 0\}_{q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}}$. Observe

that the array response matrix $\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) \triangleq \big[\mathbf{z}(\theta^1) \ \cdots \ \mathbf{z}(\theta^S)\big]$ is given by (see also (7.2.10)):

$$\mathbf{A}\big(\tilde{\boldsymbol{\theta}}\big) = \begin{bmatrix} 1 & \cdots & 1 \\ \exp\Big(\jmath\, 2\pi d_{\mathrm{n}} \sin(\theta^1)\Big) & \cdots & \exp\Big(\jmath\, 2\pi d_{\mathrm{n}} \sin(\theta^S)\Big) \\ \vdots & \ddots & \vdots \\ \exp\Big(\jmath\, 2\pi d_{\mathrm{n}} (D-1) \sin(\theta^1)\Big) & \cdots & \exp\Big(\jmath\, 2\pi d_{\mathrm{n}} (D-1) \sin(\theta^S)\Big) \end{bmatrix} \tag{7.2.6}$$

and the generalized array response matrix $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\big(\tilde{\boldsymbol{\theta}}\big) \triangleq \big[\check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta^1) \ \cdots \ \check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta^S)\big]$ by:

$$\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\big(\tilde{\boldsymbol{\theta}}\big) = \begin{bmatrix} \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,1},\mathrm{n}} \sin(\theta^1)\Big) & \cdots & \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,1},\mathrm{n}} \sin(\theta^S)\Big) \\ \vdots & \ddots & \vdots \\ \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,(D)^l},\mathrm{n}} \sin(\theta^1)\Big) & \cdots & \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,(D)^l},\mathrm{n}} \sin(\theta^S)\Big) \end{bmatrix}. \tag{7.2.7}$$

7.2.2 ULA TIME-MUSIC null- and pseudo-spectra

The definitions of the null- and pseudo-spectra formulated in equations (7.1.17)–(7.1.25) of Section 7.1.2, as well as Alg. 7.1 on page 374, also apply to the current scenario. However, they may be simplified by replacing the term $\tilde{\mathbf{t}}^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}}\mathbf{u}(\theta)$ by $t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}} \sin(\theta)$. Hence, *$l$-th order ULA TIME-MUSIC null- and pseudo-spectra* are defined as:

$$\boxed{J^{\mathbf{c}_l}_D(\theta) \triangleq \sum_{q \in \mathcal{T}^{\mathbf{c}_l}_{D,S}} \Big| f^{\mathbf{c}_l}_{D,q}(\theta) \Big|^2 = \sum_{q \in \mathcal{T}^{\mathbf{c}_l}_{D,S}} \Big| \sum_{\mathbf{i}_l \in \mathcal{I}^l_{\iota,D}} \varphi^{\mathbf{i}_l}_q \, \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}} \sin(\theta)\Big) \Big|^2}} \tag{7.2.8}$$

and:

$$\boxed{P^{\mathbf{c}_l}_{D,S}(\theta) \triangleq \frac{\|\check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta)\|^2}{\sum\limits_{q \in \mathcal{T}^{\mathbf{c}_l}_{D,S}} \Big| f^{\mathbf{c}_l}_{D,q}(\theta) \Big|^2} = \frac{(D)^l}{\sum\limits_{q \in \mathcal{T}^{\mathbf{c}_l}_{D,S}} \Big| \sum\limits_{\mathbf{i}_l \in \mathcal{I}^l_{\iota,D}} \varphi^{\mathbf{i}_l}_q \, \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}} \sin(\theta)\Big) \Big|^2}.} \tag{7.2.9}$$

respectively. Similarly to the conventional MUSIC and TIME-SPECTRAL-MUSIC pseudo-spectrum, the ULA TIME-MUSIC pseudo-spectrum exhibits sharp peaks at the source DOA's $\theta^1, \ldots, \theta^S$, and thus the source DOA's can be estimated by locating the peaks of the pseudo-spectrum. Definition (7.1.20) used for expressing the null- and pseudo-spectra in matrix-vector notation now becomes:

$$\check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta) = \mathbf{w}^{\mathbf{c}_l}_D(\theta) = \begin{bmatrix} \check{z}^{\mathbf{c}_1}_{\mathbf{i}_{l,1}}(\theta) \\ \vdots \\ \check{z}^{\mathbf{c}_1}_{\mathbf{i}_{l,(D)^l}}(\theta) \end{bmatrix} = \begin{bmatrix} \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,1},\mathrm{n}} \sin(\theta)\Big) \\ \vdots \\ \exp\Big(\jmath\, 2\pi\, t^{\mathbf{c}_l}_{\mathbf{i}_{l,(D)^l},\mathrm{n}} \sin(\theta)\Big) \end{bmatrix}. \tag{7.2.10}$$

Note that $f^{\mathbf{c}_l}_{D,q}(\theta)$ in (7.2.5) can be considered as a *nonuniform* Discrete Fourier Transform (DFT) of the set of coefficients $\{\varphi^{\mathbf{i}_l}_q\}^{\mathbf{i}_l \in \mathcal{I}^l_{\iota,D}}$ if θ is taken as the transform variable, whereas it can be considered as a *uniform* DFT if ξ defined in (7.2.2) is taken as the transform variable. Hence, given the set $\Phi^{\mathbf{c}_l}_{D,S}$ defined in (6.2.61) a Fast Fourier Transform (FFT) algorithm can be used for computing the functions in (7.2.5), the null-spectrum in (7.2.8), and the pseudo-spectrum in (7.2.9) on a uniform grid in the ξ-domain that corresponds to a nonuniform grid in the θ-domain according to relation (7.2.2).

7.2.3 Maximum number of source DOA's or minimum number of sensors for given ULA and conjugation tuple

In this section we apply the theory developed in Section 7.1.3 for arrays with arbitrary geometry to the ULA scenario. In Section 7.1.3 we have shown that for a given value of D the value of ${}_{\theta}S^{\mathbf{c}_l}_{\max,D}$ is given by (7.1.31), and vice versa that for a given value of S the value of ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ can be obtained by solving inequality (7.1.32). Hence, in order to be able to compute ${}_{\theta}S^{\mathbf{c}_l}_{\max,D}$ and ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ for the ULA scenario, we need to derive the formula for ${}_{\theta}M^{\mathbf{c}_l}_{\mathrm{u},D}$ for a ULA with a given number of sensors D, normalized sensor distance d_{n}, and conjugation tuple $\mathbf{c}_l$. Recall that ${}_{\theta}M^{\mathbf{c}_l}_{\mathrm{u},D}$ represents the maximum number of unique elements of the generalized array response vector $\check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta)$ over all possible source DOA's θ, and equals the maximum number $M^{x,\mathbf{c}_l}_{\mathrm{u},D}(\theta)$ of unique sensor cumulant functions, as well as the maximum possible row rank of the generalized array response matrix $\mathbf{A}^{\mathbf{c}_l}_{D,\diamond}\big(\tilde{\boldsymbol{\theta}}\big)$ over all possible source DOA's combinations $\tilde{\boldsymbol{\theta}}$. As we have explained and will see in the sequel, the fact that the array response vector $\mathbf{z}(\theta)$ is a function of a single variable θ implies that it has less degrees of freedom than a general MIBI array response vector. Because of this, also the generalized array response vector and generalized array response matrix have less degrees of freedom.

Firstly, we will derive how ${}_{\theta}M^{\mathbf{c}_l}_{\mathrm{u},D}$ depends on the number of sensors D and conjugation tuple $\mathbf{c}_l$ under the assumption that the normalized sensor distance d_{n} has been chosen properly such that no spatial aliasing or other undesired effects occur, i.e. we assume that the array is unambiguous (see Section 3.1.1). As is clear from (7.2.10), in order to determine the maximum number of unique elements of the generalized array response vector $\check{\mathbf{z}}^{\mathbf{c}_l}_D(\theta)$ over all possible source DOA's θ, it suffices to determine the number of different values that the term $t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}}$ in (7.2.4) can assume when the index tuple $\mathbf{i}_l$ runs through all its possible values in $\mathcal{I}^l_{\mathrm{t},D}$. For convenience of proving this, we consider the equivalent problem of determining all possible integer values of $t^{\mathbf{c}_l}_{\mathbf{i}_l} \triangleq t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}}/d_{\mathrm{n}} = \sum_{m=1}^{l}\langle -1\rangle^{c_m}(i_m-1)$. As in the previous chapters, let the number of conjugations in conjugation tuple $\mathbf{c}_l$ be denoted by $n_{\mathbf{c}_l}$, see (6.2.36). Without loss of generality, it is assumed that the first $n_{\mathbf{c}_l}$ elements of $\mathbf{c}_l$ are conjugations and thus equal $*$. Hence, the last $l-n_{\mathbf{c}_l}$ elements are no conjugations and equal $\circ$. We first consider the term $\sum_{m=1}^{l}\langle -1\rangle^{c_m} i_m$, which due to our assumption can be written as:

$$\sum_{m=1}^{n_{\mathbf{c}_l}}\langle -1\rangle^{*} i_m \;+\; \sum_{m=n_{\mathbf{c}_l}+1}^{l}\langle -1\rangle^{\circ} i_m = -\sum_{m=1}^{n_{\mathbf{c}_l}} i_m + \sum_{m=n_{\mathbf{c}_l}+1}^{l} i_m\,. \tag{7.2.11}$$

The lowest possible value this term can assume occurs for the index tuple $\mathbf{i}_l$ whose first $n_{\mathbf{c}_l}$ elements equal the maximum possible index value D and whose last $l-n_{\mathbf{c}_l}$ elements equal the minimum possible index value 1. Hence, the lowest possible value of $\sum_{m=1}^{l}\langle -1\rangle^{c_m} i_m$ equals $-n_{\mathbf{c}_l}D + (l-n_{\mathbf{c}_l})$. The largest possible value occurs for the index tuple $\mathbf{i}_l$ whose first $n_{\mathbf{c}_l}$ elements equal 1 and whose last $l-n_{\mathbf{c}_l}$ elements equal D. Hence, the largest possible value of $\sum_{m=1}^{l}\langle -1\rangle^{c_m} i_m$ equals $-n_{\mathbf{c}_l} + (l-n_{\mathbf{c}_l})D$. This implies that in total the term $\sum_{m=1}^{l}\langle -1\rangle^{c_m} i_m$ assumes $l(D-1)+1$ different values. Now we consider the second term occurring in $t^{\mathbf{c}_l}_{\mathbf{i}_l}$, viz. the term $\sum_{m=1}^{l}\langle -1\rangle^{c_m}(-1)$. Clearly, this term has no influence on the number of different possible value of $t^{\mathbf{c}_l}_{\mathbf{i}_l}$ because no index from the tuple $\mathbf{i}_l$ is involved. It only influences or 'offsets' the values of $t^{\mathbf{c}_l}_{\mathbf{i}_l}$ by $-\sum_{m=1}^{l}\langle -1\rangle^{c_m} = 2n_{\mathbf{c}_l} - l$. Concluding,

Table 7.2: *Uniform Linear Array:* ${}_{\theta}S^{l}_{\max,D} = l(D-1)$ *for* $1 \leq l \leq 4$, $1 \leq D \leq 14$.

$l \backslash D$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13
2	0	2	4	6	8	10	12	14	16	18	20	22	24	26
3	0	3	6	9	12	15	18	21	24	27	30	33	36	39
4	0	4	8	12	16	20	24	28	32	36	40	44	48	52

Table 7.3: *Uniform Linear Array:* ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S) = 1 + \lceil S/l \rceil$ *for* $1 \leq l \leq 4$, $1 \leq S \leq 14$.

$l \backslash S$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	2	2	3	3	4	4	5	5	6	6	7	7	8	8
3	2	2	2	3	3	3	4	4	4	5	5	5	6	6
4	2	2	2	2	3	3	3	3	4	4	4	4	5	5

for any conjugation tuple $\mathbf{c}_l$ the term $t^{\mathbf{c}_l}_{\mathbf{i}_l}$ assumes all values in the following range:

$$[-n_{\mathbf{c}_l}(D-1), -n_{\mathbf{c}_l}(D-1) + l(D-1)] = [-n_{\mathbf{c}_l}(D-1), (l-n_{\mathbf{c}_l})(D-1)] \,, \quad (7.2.12)$$

which contains:

$$\boxed{{}_{\theta}M^{\mathbf{c}_l}_{\mathrm{u},D} = l(D-1)+1} \quad (7.2.13)$$

different values. Hence, the term $t^{\mathbf{c}_l}_{\mathbf{i}_l}$ assumes all values in the following set:

$$\mathcal{T}^{l}_{D} \triangleq \left\{ -n_{\mathbf{c}_l}(D-1) \,, \ldots, \, -n_{\mathbf{c}_l}(D-1) + l(D-1) \right\}. \quad (7.2.14)$$

Likewise, for any conjugation tuple $\mathbf{c}_l$ the term $t^{\mathbf{c}_l}_{\mathbf{i}_l,\mathrm{n}}$ defined in (7.2.4) assumes all values in the following set when the index tuple $\mathbf{i}_l$ runs through all its possible values in $\mathcal{I}^{l}_{\mathrm{t},D}$:

$$\mathcal{T}^{l}_{D,\mathrm{n}} \triangleq \left\{ -n_{\mathbf{c}_l}(D-1)\, d_{\mathrm{n}} \,, \ldots, \, \big(-n_{\mathbf{c}_l}(D-1) + l(D-1)\big)\, d_{\mathrm{n}} \right\}. \quad (7.2.15)$$

Now we are in a position to compute ${}_{\theta}S^{\mathbf{c}_l}_{\max,D}$ and ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ for a ULA. Substituting (7.2.13) into (7.1.31), it follows that ${}_{\theta}S^{\mathbf{c}_l}_{\max,D}$ is given by:

$${}_{\theta}S^{\mathbf{c}_l}_{\max,D} \triangleq {}_{\theta}S^{l}_{\max,D} = l(D-1) \,. \quad (7.2.16)$$

Table 7.2 lists the values of ${}_{\theta}S^{l}_{\max,D}$ for $1 \leq l \leq 4$ and $1 \leq D \leq 14$. Likewise, it follows from (7.1.32) that the minimum required number of sensors ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ for a ULA can be obtained by solving:

$$S + 1 \;\leq\; l(D-1) + 1 \,,$$

which yields:

$${}_{\theta}D^{\mathbf{c}_l}_{\min}(S) \triangleq {}_{\theta}D^{l}_{\min}(S) = 1 + \lceil S/l \rceil \,. \quad (7.2.17)$$

Table 7.3 on the current page lists the values of ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ for $1 \leq l \leq 4$ and $1 \leq S \leq 14$. In addition, in Figures 7.13-7.16, ${}_{\theta}D^{\mathbf{c}_l}_{\min}(S)$ is plotted as a function of S for $l = 1, 2, 3$ and 4 respectively. Compare Figures 7.14-7.16 with Figures 6.10-6.12 from Chapter 6.

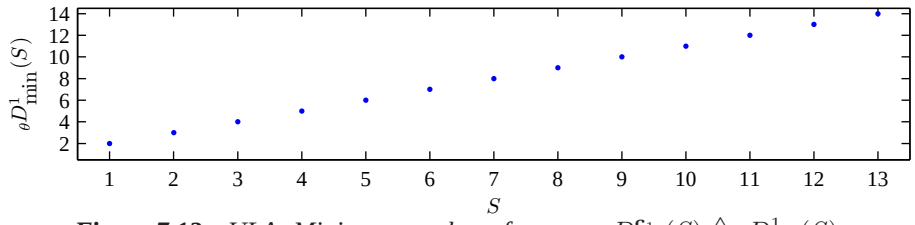

Figure 7.13: *ULA: Minimum number of sensors* $_\theta D^{\mathbf{c}_1}_{\min}(S) \triangleq {}_\theta D^1_{\min}(S)$.

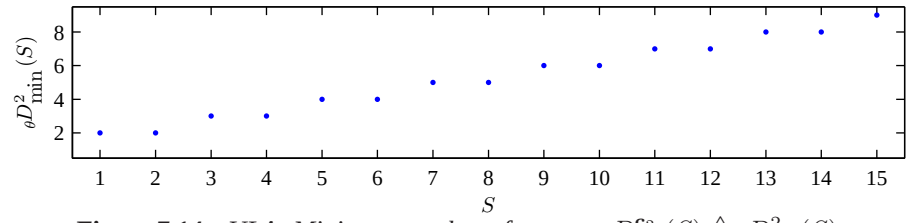

Figure 7.14: *ULA: Minimum number of sensors* $_\theta D^{\mathbf{c}_2}_{\min}(S) \triangleq {}_\theta D^2_{\min}(S)$.

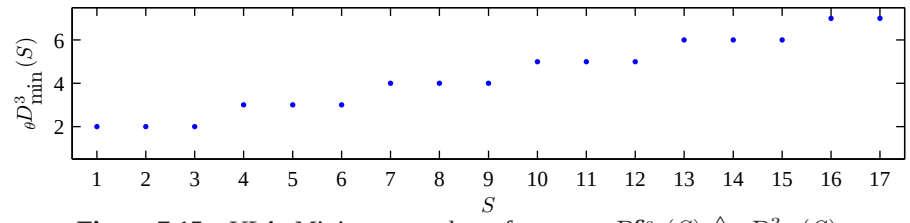

Figure 7.15: *ULA: Minimum number of sensors* $_\theta D^{\mathbf{c}_3}_{\min}(S) \triangleq {}_\theta D^3_{\min}(S)$.

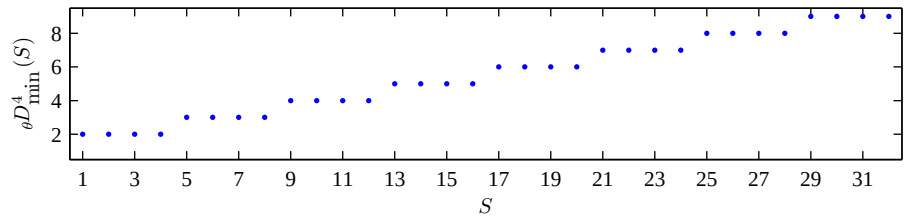

Figure 7.16: *ULA: Minimum number of sensors* $_\theta D^{\mathbf{c}_4}_{\min}(S) \triangleq {}_\theta D^4_{\min}(S)$.

7.2.4 Examples of ULA-TIME-SPECTRAL-MUSIC

In this section we provide several examples of ULA-TIME-SPECTRAL-MUSIC with normalized sensor spacing $d_{\text{n}} = 1/2$ by applying the algorithm in Alg.7.1 to signal scenarios that are similar to the ones presented in Section 7.1.4 but adapted to the ULA scenario. In particular, the source and noise signals are generated in the same way, but now the sensors are placed on a ULA and their number is chosen as required by (7.2.17).

7.2.4.1 ULA Example 1: Ideal and estimated pseudo-spectra for scenario with 4 sensors, 6 sources, second order statistics, conjugation pair $(\circ, *)$, and 6 lags

The example we present in this section employs exactly the same source and noise signals as used in the example of Section 7.1.4.1. Hence, for details about these signals and how they are generated, see that section. The only differences are that now the array is a ULA that requires a different number of sensors and also that the parameter space of interest Θ, to which the source DOA's belong, now has a size of 180 degrees. We choose $\Theta \triangleq [-90, 90]$ degrees and the DOA's of the six sources are now $\tilde{\boldsymbol{\theta}} = [-57, -17, -5, 18, 29, 51\,]$ degrees. Because a ULA does not utilize all its possible degrees of freedom, it generally requires more sensors for identifying the same number of source DOA's than an array with arbitrary geometry; see the remark made at the end of Section 7.1.4.1. The number of sensors that is required now can be found from equation (7.2.17), Table 7.3, or Fig. 7.14, and equals ${}_{\theta}D^{\circ *}_{\min}(S) \triangleq {}_{\theta}D^{2}_{\min}(S) = 1 + \lceil 6/2 \rceil = 4$. The normalized sensor distance d_{n} is chosen as $1/2$, which guarantees that the array is unambiguous. Note that essentially there is only one underlying function available for defining the null- and pseudo-spectra because ${}_{\theta}Q^{\circ *}_{4,6} = 1$ according to (7.1.26). For $1 \leq i \leq 4$ the normalized sensor positions of the uniform linear array are given by:

$$p_{i,\text{n}} = (i-1)d_{\text{n}} = \tfrac{1}{2}(i-1)\,. \tag{7.2.18}$$

If desired, the positions can be shifted such that the 'center of gravity' lies in the origin, but this does not influence the results. The ideal and estimated pseudo-spectra are computed using Alg. 7.1 with $\mathbf{C}_4^{x,\circ *}$ and $\hat{\mathbf{C}}_4^{x,\circ *}$ respectively. The resulting spectra are shown in Fig. 7.17.

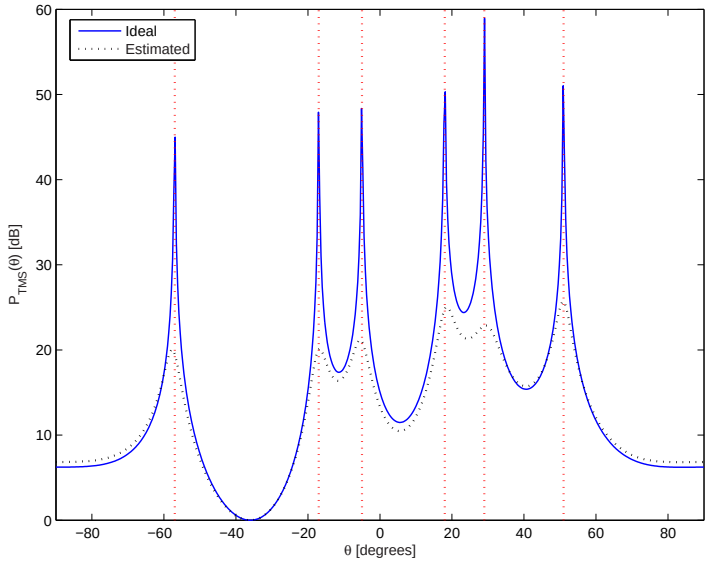

Figure 7.17: *ULA Example 1: Ideal ULA-TIME-SPECTRAL-MUSIC spectrum $P_4^{\circ *}(\theta)$ and its estimate $\hat{P}_4^{\circ *}(\theta)$ $D = 4$, $S = 6$ and $\mathbf{c}_2 = (\circ, *)$.*

7.2.4.2 ULA Example 2: Comparing ideal pseudo-spectra for cumulant orders $l = 1, 2, 3, 4$ and conjugation tuple $\mathbf{c}_l = (\circ)_l$, i.e. no conjugations

As in Section 7.1.4.2, in this section our main goal is to provide theoretical insight and we only present results computed from ideal subspace matrices under the assumptions that the source signals have sufficient temporal structure and that the conjugation tuple for order l is given by $\mathbf{c}_l = (\circ)_l$. Now, we compare pseudo-spectra for cumulant orders $l = 1, 2, 3$ and 4 for two sets of simulations employing the same number of sensors. Furthermore, as in the previous example we choose the parameter space of interest as $\Theta \triangleq [-90, 90]$ degrees. In the first set of simulations, the number of sources equals $S = 11$ with the source DOA's given by $\tilde{\boldsymbol{\theta}} = [-65, -51, -35, -17, -5, 4, 18, 29, 51, 63, 71]$ degrees, implying that two underlying functions are employed for defining the null- and pseudo-spectra. In the second set of simulations, the number of sources equals $S = 12$ with the source DOA's given by $\tilde{\boldsymbol{\theta}} = [-65, -51, -35, -17, -5, 4, 11, 18, 29, 51, 63, 71]$ degrees. Because this set employs the same number of sensors as the first, only one underlying function is employed for defining the null- and pseudo-spectra. For both sets of simulations, this way a fair comparison is made between the different cumulant orders because ${}_{\theta}S^{(\circ)_l}_{\text{max},D} \triangleq S^{l}_{\text{max},D} = 12$ occurs in each row of Table 7.2 on page 390, which implies that the definitions of the null- and pseudo-spectra involve the same number of functions for each $l \in \{1, 2, 3, 4\}$.

Firstly, we consider the first set of simulations with $S = 11$. The minimum number of sensors ${}_{\theta}D^{(\circ)_l}_{\text{min}}(S)$ required for each value of l can be found from equation (7.2.17), Table 7.3, or Figures 7.14-7.16. Hence, for $S = 11$ we have:

$$ {}_{\theta}D^{(\circ)_1}_{\text{min}}(S) = 12, \quad {}_{\theta}D^{(\circ)_2}_{\text{min}}(S) = 7, \quad {}_{\theta}D^{(\circ)_3}_{\text{min}}(S) = 5, \quad {}_{\theta}D^{(\circ)_4}_{\text{min}}(S) = 4\,. $$

From (7.2.16), Figures 7.13-7.16, or Table 7.2 we see that with $D = 13$ for $l = 1$, $D = 7$ for $l = 2$, $D = 5$ for $l = 3$ and with $D = 4$ for $l = 4$ we can identify a maximum of ${}_{\theta}S^{(\circ)_1}_{\text{max},13} = {}_{\theta}S^{(\circ)_2}_{\text{max},7} = {}_{\theta}S^{(\circ)_3}_{\text{max},5} = {}_{\theta}S^{(\circ)_4}_{\text{max},4} = 12$ source DOA's. Since we are considering a scenario with $S = 11$, this means that two underlying functions are employed for defining the null- and pseudo-spectra. Hence, this way for $l = 1, 2, 3, 4$ the number of underlying linearly independent functions available for averaging in the definitions of the cost functions and pseudo-spectra are given by ${}_{\theta}Q^{(\circ)_l}_{D,S} = l(D-1) + 1 - S$ (see (7.1.26) and (7.2.13)) as ${}_{\theta}Q^{(\circ)_1}_{13,11} = {}_{\theta}Q^{(\circ)_2}_{7,11} = {}_{\theta}Q^{(\circ)_3}_{5,11} = {}_{\theta}Q^{(\circ)_4}_{4,11} = 2$. The normalized sensor array configurations are again given by (7.2.18) with $d_{\text{n}} = 1/2$, i.e. for each D the ULA is unambiguous. The TIME-MUSIC pseudo-spectra $P^{(\circ)_1}_{13}(\theta)$, $P^{(\circ)_2}_{7}(\theta)$, $P^{(\circ)_3}_{5}(\theta)$ and $P^{(\circ)_4}_{4}(\theta)$ computed by means of Alg. 7.1 are plotted in Fig. 7.18. The source DOA's are indicated by the dotted vertical lines. For easy comparison, the minima of all spectra have been normalized to 0 dB. The figure shows that all spectra exhibit sharp peaks at the source DOA's and that no spurious peaks are present. Also notice that the spectra do not differ much.

For the second set of simulations with $S = 12$, the TIME-MUSIC pseudo-spectra $P^{(\circ)_1}_{13}(\theta)$, $P^{(\circ)_2}_{7}(\theta)$, $P^{(\circ)_3}_{5}(\theta)$ and $P^{(\circ)_4}_{4}(\theta)$ are depicted in Fig. 7.19. Note that the sensor positions again are given by (7.2.18). Now, for $l = 1, 2, 3, 4$ the number of underlying linearly independent functions available for averaging in the definitions of the cost functions and pseudo-spectra are given by (7.1.26) as ${}_{\theta}Q^{(\circ)_1}_{13,12} = {}_{\theta}Q^{(\circ)_2}_{7,12} = {}_{\theta}Q^{(\circ)_3}_{5,12} = {}_{\theta}Q^{(\circ)_4}_{4,12} = 1$. As in Section 7.1.4.2, it can be seen that all source DOA's are correctly resolved. Fig. 7.19 also clearly shows that in case only one function is involved the pseudo-spectra for different cumulant

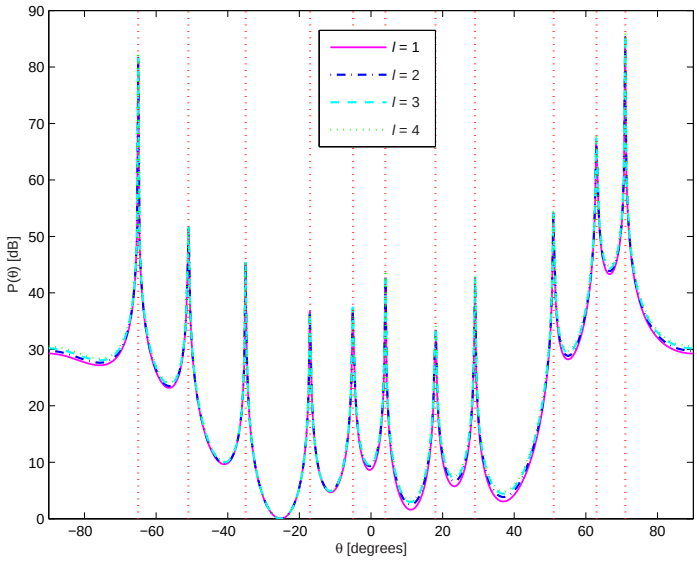

Figure 7.18: *ULA Example 2: Ideal ULA-TIME-MUSIC spectra* $P_{13}^{(\circ)_1}(\theta)$, $P_{7}^{(\circ)_2}(\theta)$, $P_{5}^{(\circ)_3}(\theta)$ *and* $P_{4}^{(\circ)_4}(\theta)$ *for* $S = 11$.

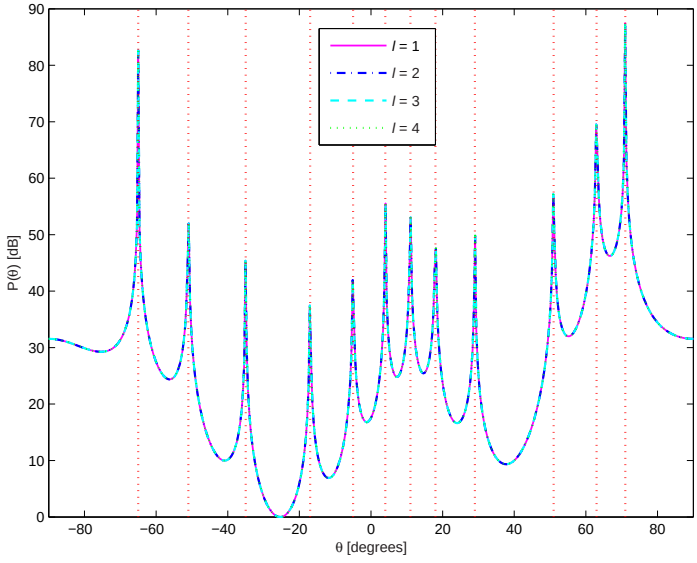

Figure 7.19: *ULA Example 2: Ideal ULA-TIME-MUSIC spectra* $P_{13}^{(\circ)_1}(\theta)$, $P_{7}^{(\circ)_2}(\theta)$, $P_{5}^{(\circ)_3}(\theta)$ *and* $P_{4}^{(\circ)_4}(\theta)$ *for* $S = 12$.

orders become the same. Again, from many other similar simulations, it can be concluded that in general one equation $f_{D,q}^{(\circ)_l}(\theta)$ is sufficient to resolve ${}_{\theta}S_{\max,D}^{(\circ)_l} = \binom{D+l-1}{l} - 1$ source DOA's uniquely from the pseudo-spectrum $P_{D}^{(\circ)_l}(\theta)$.

7.3 ULA-TIME-ROOT-MUSIC

In this section, we derive the ROOT version of ULA-TIME-MUSIC, which is denoted by *ULA-TIME-ROOT-MUSIC (UTRM)*. Hence, instead of computing and plotting a pseudo-spectrum as we did for the ULA-TIME-SPECTRAL-MUSIC algorithm in Section 7.2, we directly compute the zeros of the null-spectrum; see also Section 3.2.3 of Chapter 3. To a large extent ULA-TIME-ROOT-MUSIC is the same as ULA-TIME-SPECTRAL-MUSIC. However, instead of determining the source DOA's by searching the pseudo-spectrum $P_{D,S}^{\mathbf{c}_l}(\theta)$ defined in (7.2.9) for its maxima, the ULA-ROOT-TIME-MUSIC algorithm is based on finding the roots of a polynomial that is closely related to the denominator $P_{D,S}^{\mathbf{c}_l}(\theta)$, i.e. the null-spectrum $J_D^{\mathbf{c}_l}(\theta)$ defined in (7.2.8). Similarly to the conventional ROOT-MUSIC algorithm, the source DOA's are derived from these roots. We can take different approaches for developing an UTRM algorithm. For example, we can follow almost exactly the same approach as in Section 3.2.3. However, here we take a slightly different approach and start from functions that are still defined in terms of general coefficients instead of zero/noise eigenvectors.

To start with, recall from Section 7.2.3 that for any conjugation tuple $\mathbf{c}_l$ the term $t_{\mathbf{i}_l,\mathrm{n}}^{\mathbf{c}_l}$ defined in (7.2.4) and occurring in the exponent of each function $f_{D,q}^{\mathbf{c}_l}(\theta)$ defined in (7.2.5) assumes all values in the set $\mathcal{T}_{D,\mathrm{n}}^l$ defined in (7.2.15) when the index tuple $\mathbf{i}_l$ runs through all its possible values in $\mathcal{I}_{\mathrm{t},D}^l$. Equivalently, for any conjugation tuple $\mathbf{c}_l$ the term $t_{\mathbf{i}_l}^{\mathbf{c}_l} \triangleq t_{\mathbf{i}_l,\mathrm{n}}^{\mathbf{c}_l}/d_{\mathrm{n}}$ assumes all values in the set $\mathcal{T}_D^l$ defined in (7.2.14) when the index tuple $\mathbf{i}_l$ runs through all elements of $\mathcal{I}_{\mathrm{t},D}^l$. Hence, each function $f_{D,q}^{\mathbf{c}_l}(\theta)$ in (7.2.5) can we written as follows:

$$\begin{aligned} f_{D,q}^{\mathbf{c}_l}(\theta) &= \sum_{t_{\mathrm{n}} \in \mathcal{T}_{D,\mathrm{n}}^l} \left(\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l \,:\, t_{\mathbf{i}_l,\mathrm{n}}^{\mathbf{c}_l} = t_{\mathrm{n}}} \varphi_q^{\mathbf{i}_l} \right) \exp\Big(\jmath\, 2\pi\, t_{\mathrm{n}}\, \sin(\theta)\Big) \\ &= \sum_{t \in \mathcal{T}_D^l} \left(\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l \,:\, t_{\mathbf{i}_l}^{\mathbf{c}_l} = t} \varphi_q^{\mathbf{i}_l} \right) \exp\Big(\jmath\, t\, \xi(\theta)\Big), \end{aligned} \tag{7.3.1}$$

where $\xi(\theta)$ is defined in (7.2.2). By defining:

$$\varphi_q^t \triangleq \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},D}^l \,:\, t_{\mathbf{i}_l}^{\mathbf{c}_l} = t} \varphi_q^{\mathbf{i}_l}, \tag{7.3.2}$$

we can write (7.3.1) as:

$$\begin{aligned} f_{D,q}^{\mathbf{c}_l}(\theta) &= \sum_{t \in \mathcal{T}_D^l} \varphi_q^t \exp\Big(\jmath\, t\, \xi(\theta)\Big) = \sum_{t=-n_{\mathbf{c}_l}(D-1)}^{-n_{\mathbf{c}_l}(D-1)+l(D-1)} \varphi_q^t \exp\Big(\jmath\, t\, \xi(\theta)\Big) \\ &= \sum_{t \in \mathcal{T}_D^l} \varphi_q^t \left(z(\theta)\right)^t = \sum_{t=-n_{\mathbf{c}_l}(D-1)}^{-n_{\mathbf{c}_l}(D-1)+l(D-1)} \varphi_q^t \left(z(\theta)\right)^t, \end{aligned} \tag{7.3.3}$$

where:

$$z(\theta) = \exp\Big(\jmath\, \xi(\theta)\Big). \tag{7.3.4}$$

Leaving out the dependence of $z(\theta)$ on θ, with abuse of notation we define:

$$f_{D,q}^{\mathbf{c}_l}(z) \triangleq \sum_{t \in \mathcal{T}_D^l} \varphi_q^t\, (z)^t = \sum_{t=-n_{\mathbf{c}_l}(D-1)}^{-n_{\mathbf{c}_l}(D-1)+l(D-1)} \varphi_q^t\, (z)^t. \tag{7.3.5}$$

Now, the l-th order ULA-TIME-MUSIC null- spectrum defined in (7.2.8) can be written as:

$$J_D^{\mathbf{c}_l}(\theta) \triangleq \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \left| f_{D,q}^{\mathbf{c}_l}\Big((e)^{\jmath\xi(\theta)}\Big)\right|^2 = \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} f_{D,q}^{\mathbf{c}_l}(z)\left(f_{D,q}^{\mathbf{c}_l}(z)\right)^*\Big|_{z=(e)^{\jmath\xi(\theta)}}$$
$$= \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} f_{D,q}^{\mathbf{c}_l}(z)\left(f_{D,q}^{\mathbf{c}_l}(1/z^*)\right)^*\Big|_{z=(e)^{\jmath\xi(\theta)}}, \tag{7.3.6}$$

where the latter equality holds because $z = \frac{1}{z^*}$ for z on the unit circle. With abuse of notation we can now write:

$$J_D^{\mathbf{c}_l}(z) = \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} f_{D,q}^{\mathbf{c}_l}(z)\left(f_{D,q}^{\mathbf{c}_l}(1/z^*)\right)^*. \tag{7.3.7}$$

Because:

$$\left(f_{D,q}^{\mathbf{c}_l}(1/z^*)\right)^* = \sum_{t\in\mathcal{T}_D^l} \left(\varphi_q^t\right)^*(z)^{-t} = \sum_{t=-n_{\mathbf{c}_l}(D-1)}^{-n_{\mathbf{c}_l}(D-1)+l(D-1)} \left(\varphi_q^t\right)^*(z)^{-t}$$

the null-spectrum $J_D^{\mathbf{c}_l}(z)$ only contains terms with negative and positive powers of z, and no terms with powers of z^*. It is this property of $J_D^{\mathbf{c}_l}(z)$ that makes it easy to compute its roots. Since the powers of z in $J_D^{\mathbf{c}_l}(z)$ range from $-l(D-1)$ till $l(D-1)$, multiplying $J_D^{\mathbf{c}_l}(z)$ by $(z)^{l(D-1)}$ yields the following *'ULA-TIME-ROOT-MUSIC polynomial'* $p_{\text{UTRM}}(z)$ of degree $2l(D-1)$:

$$p_{\text{UTRM}}(z) \triangleq (z)^{l(D-1)} J_D^{\mathbf{c}_l}(z) = (z)^{l(D-1)} \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} f_{D,q}^{\mathbf{c}_l}(z)\left(f_{D,q}^{\mathbf{c}_l}(1/z^*)\right)^*, \tag{7.3.8}$$

which can be written explicitly as follows:

$$p_{\text{UTRM}}(z) = \sum_{q\in\mathcal{T}_{D,S}^{\mathbf{c}_l}} \sum_{\tau=0}^{2l(D-1)} \sum_{t=0}^{l(D-1)} \varphi_q^{t-n_{\mathbf{c}_l}(D-1)} \left(\varphi_q^{t-n_{\mathbf{c}_l}(D-1)-(\tau-l(D-1))}\right)^* (z)^{\tau}. \tag{7.3.9}$$

Hence, now we have constructed a polynomial function that is zero at the locations $z^j = (e)^{\jmath\xi(\theta^j)}\ \forall\, 1 \le j \le S$ in the z-plane that correspond to the source DOA's:

$$p_{\text{URRM}}(z)\big|_{z=(e)^{\jmath\xi(\theta^j)}} = p_{\text{RM}}\big((e)^{\jmath\xi(\theta^j)}\big) = 0 \qquad \forall\, 1 \le j \le S\,. \tag{7.3.10}$$

As in Section 3.2.3, we will call the roots $\{z^j \triangleq (e)^{\jmath\xi(\theta^j)}\}^{1\le j\le S}$ *source roots*. Again, because the function $J_D^{\mathbf{c}_l}(z)$ in (7.3.7) is self-reciprocal its roots appear in conjugate reciprocal pairs $(z, 1/z^*)$, and thus each root on the unit circle is double. Hence, all source roots are double roots. Since $p_{\text{UTRM}}(z)$ is of degree $2l(D-1)$, it has $2l(D-1)$ roots, $2S$ of which are the double source roots. Consequently, there are $2(l(D-1)-S)$ other roots that in general do not lie on the unit circle. These roots are called *spurious roots* or *noise roots*. See the discussion in Section 3.2.3 about this kind of roots. There are essentially two different kinds of spurious roots. Firstly, spurious roots are always present if $l(D-1)-S > 0$, i.e. if the dimension of the generalized noise subspace is larger than one. If they lie close to the unit circle such noise roots may give rise to spurious peaks in a TIME-SPECTRAL-MUSIC pseudo-spectrum, or may cause confusion between source and noise roots in TIME-ROOT-MUSIC. Especially if an estimated cumulant matrix is used, the noise roots may move close

Algorithm 7.2 ULA-TIME-ROOT-MUSIC.

1-3: See steps 1–3 of Alg. 7.1;

4: Use left null space of sensor cumulant subspace matrix $\mathbf{C}_D^{x,\mathbf{c}_l}$ *for constructing ULA-TIME-ROOT-MUSIC polynomial:*

$$p_{\,\mathrm{UTRM}}(z) = (z)^{l(D-1)} \sum_{q \in \mathcal{T}_{D,S}^{\mathbf{c}_l}} f_{D,q}^{\mathbf{c}_l}(z) \left(f_{D,q}^{\mathbf{c}_l}(1/z^*)\right)^* ;$$

5: Compute roots of $p_{\,\mathrm{UTRM}}(z)$*;*

6: Select S *unique roots of* $p_{\,\mathrm{UTRM}}(z)$ *that lie on or closest to unit circle;*

7: For each source root z*, compute corresponding source DOA:*

$$\theta = \arcsin\Big(\tfrac{\lambda_c}{2\pi d} \arg(z)\Big) .$$

to the unit circle, thereby causing this kind of problem. If all source roots are closer to the unit circle than the noise roots, the latter play no particular role in determining the source DOA's for the TIME-ROOT-MUSIC algorithm. Secondly, spurious roots close to or on the unit circle are present if the array is ambiguous. Obviously, this kind of spurious roots always causes confusion of source and noise roots. Therefore, the array should always be designed in such a way that it is unambiguous. The roots of $p_{\mathrm{UTRM}}(z)$ in (7.3.8) or (7.3.9), and thus of the ULA null-spectrum $J_D^{\mathbf{c}_l}(z)$, can easily be obtained by any polynomial rooting method. Once these roots have been obtained, in the ideal case the S unique (double) roots that lie on the unit circle are selected, whereas in the estimated case the ones closest to the unit circle are selected. From these roots, the source DOA's can easily be recovered by inverting the function $z(\theta) = \exp(\jmath\xi(\theta))$ with the 'coordinate transformation' $\xi(\theta)$ given by (3.1.10) on page 87. The different steps of the ULA-TIME-ROOT-MUSIC algorithm are summarized in Alg. 7.2. It is formulated in terms of ideal statistics but is completely the same for estimated statistics. Figure 7.20 shows the unique roots of the ULA-TIME-ROOT-MUSIC polynomial defined in (7.3.9) for Example 3 of Section 7.1.4.3.

7.4 Conclusions

In this chapter, we have applied the theory that we have developed in the previous chapters to the problem of Instantaneous Semi-Blind Source Localization (ISBSL). In particular, we have generalized the various MUSIC-based algorithms described in Chapter 3 to our method, resulting in what call the TIME-MUSIC approach, which consists of the so-called TIME-SPECTRAL-MUSIC, ULA-TIME-SPECTRAL-MUSIC, and ULA-TIME-ROOT-MUSIC algorithms. In the course of the chapter, we have made plausible that in general for ISBSL problems with one-parameter array response vectors only one equation suffices to define appropriate null- and pseudo-spectra that uniquely determine the source DOA's. We have also seen that the configuration of the sensor array is of paramount importance. If the number of identifiable source DOA's has to be maximized, the array should not be too structured. for example in the case of a ULA, not all available degrees of freedom may be sufficiently

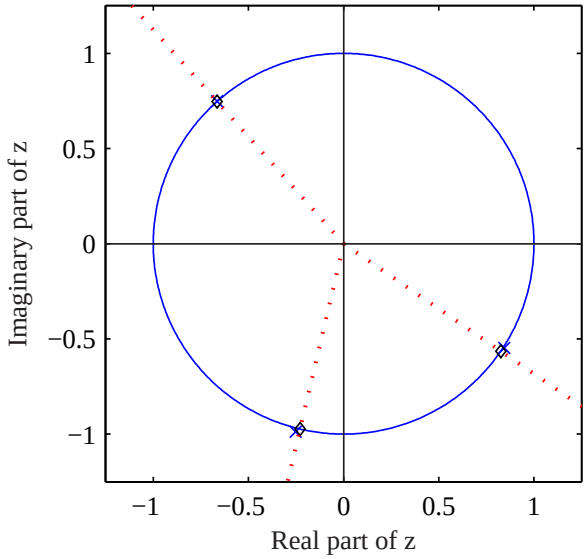

Figure 7.20: *Ideal ($\diamond$) and estimated ($\times$) roots of the ULA-TIME-ROOT-MUSIC polynomial $p_{\mathrm{UTRM}}(z)$ for Example 3 of Section 7.1.4.3.*

exploited. In general, the sensor positions should be chosen such that the determinant of the considered Khatri-Rao matrix as function of the DOA in the parameter space of interest is always non-zero. Finally, as in the previous chapter, we have provided insight into the computation of the maximum number of identifiable source DOA's and the minimum number of sensors.

CHAPTER 8

Conclusions, trade-offs and future research

8.1 Conclusions

In this thesis we have developed a general method for solving three closely related blind instantaneous identification problems, viz. Multiple-Input Multiple-Output (MIMO) Instantaneous Blind Identification (MIBI), Instantaneous Blind Signal Separation (IBSS) and Instantaneous Semi-Blind Source Localization (ISBSL). In those problems a number of mutually statistically independent source signals are mixed by a MIMO instantaneous mixing system and only the mixed signals are observed. The goals of MIBI, IBSS and ISBSL are the identification of the mixing system, the recovery of the source signals, and the localization of narrowband sources, respectively. Until now, mainly the following three different approaches have been followed, which are based on the exploitation of non-Gaussianity, second order spatial uncorrelatedness combined with temporal non-whiteness, and second order spatial uncorrelatedness combined with second order non-stationarity. What is lacking from those approaches is the exploitation of Higher Order Temporal Structure (HOTS) in the data, such as higher order non-whiteness and non-stationarity. Some methods for exploiting HOTS exist, but usually these are quite specific. In addition, most blind methods described in the literature cannot deal with a MIBI scenario of great interest, viz. the so-called underdetermined scenario in which there are more sources than sensors. Like many other signal processing problems, blind identification problems are commonly formulated and approached as linear algebra problems. Although this certainly has the significant advantage that off-the-shelf linear algebra algorithms can be employed, it also has the disadvantage that the underlying more general and possibly nonlinear structure of the problem often is overlooked or neglected. In our opinion, always trying to formulate a problem in linear algebra terms may disguise the true nature and structure of the problem and impedes the development and use of nonlinear algebra tools in signal processing. In this work we have shown that it is also possible to formulate and solve a blind identification problem in an intuitive manner by employing its natural nonlinear description.

All those motivations have led us to the main purpose of the research presented in this thesis, viz. the development of a unifying framework and method for exploiting the temporal structure in the data of any kind and any order for performing MIBI, IBSS and ISBSL. We have achieved this goal for physically plausible assumptions on the temporal structure of the source and noise signals by projecting the encompassing MIBI problem onto two dual mathematical problems. This projection has been accomplished by applying subspace techniques to a subspace matrix containing sensor cumulant values arranged in a specific manner that allows to incorporate any kind and order of statistical variability, such as arbitrary order non-stationarity and non-whiteness, in a unified way. In the first problem, MIBI has been projected onto the problem of solving a system of multivariate homogeneous polynomial/polyconjugal equations, the solutions of which yield (possibly scaled) estimates of the columns of the mix-

ing matrix. The number of variables and the degree (of homogeneity) of the functions in the system equal the number of sensors and the order of the considered statistics, respectively. Possible approaches for solving the nonlinear system of equations have been described. In particular, we have employed the so-called homotopy method, which is relatively unknown and hence unused in the signal processing community. Because in our opinion homotopy techniques are very powerful and useful for several signal processing problems, one of our goals also has been to advocate and adapt homotopy methods for use in the signal processing research area. In the second problem, MIBI has been projected onto the problem of solving a Multi-Matrix Generalized Eigenvalue Decomposition (MMGEVD) problem, the solutions of which are related to the columns of the mixing matrix by a linear transformation. We have shown that under certain conditions the MMGEVD problem can be solved using existing mathematical tools, whereas for some other conditions appropriate tools still need to be developed. We have clarified the sense in which the two MIBI problem formulations are dual and provided enhanced insight into the close relationship between solving systems of polynomial/polyconjugal equations on the one hand, and solving generalized eigenvalue problems on the other hand. Because of this close connection, solving one of the two mathematical problems in principle also solves the other.

The physically plausible assumptions underlying our problem formulations primarily serve to ensure that sufficient temporal structure is present in the source signals, that the source and noise signals are mutually unrelated, and that the noise signals have a simpler statistical and/or temporal structure than the source signals. We have shown how this latter assumption allows one to annihilate the influence of additive sensor noise on the estimation of the mixing system by considering the sensor cumulant functions on a Noise-Free Region Of Support (ROS). This principle is a major advantage of exploiting the temporal structure in the data. The condition on the required temporal structure of the source signals has been formulated in terms of the linear independence of the source auto-cumulant functions on the Noise-Free ROS. This means that the spectral diversity for the considered statistical order is exploited. Depending on the number of sensors, the order and type of the exploited temporal structure, the conjugation pattern, the arrangement of the statistics in the subspace matrix, and the array configuration for problems with a parameterized mixing system, a certain maximum number of mixing matrix columns can be determined that exceeds the number of sensors for orders larger than one. Our approach allows the identification of an underdetermined system in a natural manner that does not require such a strong assumption like sparsity that is usually employed by other methods. We have provided insight into the computation of (an upper bound on) the maximum number of mixing matrix columns that can be identified under the adopted assumptions as a function of the number of sensors, the employed statistical order, and the conjugation pattern. Conversely, we have also developed new insight into the derivation of (a lower bound on) the minimum number of sensors that is required for uniquely identifying a certain number of mixing matrix columns as a function of the employed statistical order and conjugation pattern. For example, we have proven that for two sensors the maximum number of identifiable columns equals the order of the employed statistics, and that when using three sensors and second order statistics up to four columns can be identified. Contrary to what is commonly believed, we have proven that it is possible to deal with underdetermined scenarios by using Second Order Statistics only. We have achieved this by exploiting the temporal structure in the source signals. If the number of sources is unknown, it can be computed or estimated as the (effective) rank of the subspace matrix. It should be noted that our method can deal with MIBI scenarios in which the mixing matrix

is rank-deficient. Moreover, over-estimation of the number of sources forms no problem and is handled in a natural manner. We have also demonstrated and emphasized that the specific arrangement of sensor cumulant values in the subspace matrix is crucial and allows different trade-offs (see next section). In fact, different subspace methods use different mappings from a certain sensor cumulant tensor to the subspace matrix. One of the goals of this work also was to contribute to the subspace research area by providing a unifying approach that allows natural generalizations and provides enhanced insight. This has resulted in a general algorithm for designing subspace algorithms employing any kind and order of statistics. We have also derived the maximum possible rank of the Khatri-Rao product with an arbitrary conjugation pattern, which is an important result for many array signal processing problems.

The rationale behind most of the work in this thesis has been to provide conceptual insight and highlighting the geometric and algebraic structure of the different problem formulations. For this reason we have developed and employed a notational device that in our opinion is natural and intuitive, and directly allows several generalizations. We have shown that geometrically, solving the system of homogeneous equations is equivalent to finding the one-dimensional intersections between cones in an Euclidian space, which represent the zero contour levels of the functions in the system. The MIBI theory that we have developed in this thesis is unifying in several senses. Firstly, it is general with respect to the order of the exploited temporal structure in the sense that the mathematical problem formulation has the same structure for any order. This is reflected in the uniformity of our notation. Secondly, it is unifying with respect to the type of temporal statistical variability in the data because it allows to exploit all types of statistical variability in the source signals in a unified manner. Thirdly, the theory is unifying with respect to the chosen conjugation pattern implying that for complex-valued signals the conjugation pattern of the arguments of the involved cumulant functions can be chosen arbitrarily. Finally, it is unifying with respect to the kind of signals involved because it does not make a difference whether the data to be analyzed consists of 1D time signals, 2D images, 3D volumetric data, and so on. Although we have mainly developed the theory in this thesis for a fixed arbitrary statistical order, it should also be noted that it is possible to exploit several orders and even several conjugation patterns simultaneously. Algorithms that employ such a mixture of orders may provide additional robustness or flexibility, for example. In summary, we have presented a method for performing MIBI that is very general and in theory is limited only by the errors in the estimation of the sensor cumulant functions.

8.2 Trade-offs

In the course of the developments in this thesis, we have shown that our theory naturally reveals which kind of trade-offs can be made, as well as how they can be made. In this section we summarize the various trade-offs qualitatively in terms of the quantities involved, such as the maximum number of identifiable mixing matrix columns or array response vectors (number of sources), minimum required number of sensors, exploited type(s) of temporal structure, exploited order(s) of temporal structure, employed conjugation pattern(s), number of samples required for reliable estimation of the involved statistics, required amount of temporal structure, required size of the Noise-Free ROS, the arrangement of the statistics in the subspace matrix.

To start with, we have shown that under the adopted assumptions the maximum number of identifiable mixing matrix columns is a function of the number of sensors and the employed conjugation pattern (which also specifies the order). From the functional dependence that we have found several general trade-offs follow immediately. Firstly, the larger the number of sensors for a given cumulant order is, the larger the number of identifiable columns, and vice versa. Likewise, the higher the cumulant order for a given number of sensors is, the larger the number of identifiable columns, and vice versa. Conversely, the larger the number of sources is, the larger the number of sensors and/or the higher the cumulant order need to be. In addition, for a given (maximum) number of sources there is a trade-off between the number of sensors on the one hand, and the cumulant order on the other hand. We have also seen that as a function of the employed conjugation pattern the maximum number of identifiable columns is maximal for even order statistics if exactly half of the cumulant function arguments are conjugated, and for odd order statistics if half minus or plus one of the cumulant function arguments are conjugated. Furthermore, depending on the number of sensors and the conjugation pattern, there are also some computational trade-offs. In general, the higher the cumulant order for a given number of sensors is, the more cumulant functions need to be estimated in order to exploit all information and possibilities. Hence, there exists a trade-off between the order, number of columns that can or need to be estimated, and the amount of cumulant values to be computed. Relating to this issue we remark that the derivation presented in the thesis is based on the exploitation of all unique sensor cumulant functions. However, a similar derivation can be given for the case in which only a specific subset of those functions is used. One consequence is that less columns can be identified. Another kind of trade-off concerns the structure of the mixing matrix versus the number of identifiable columns. The more structured or parameterized the mixing matrix is, the more columns or associated parameters can be estimated. For ISBSL problems care should be taken that the array configuration exploits the available degrees of freedom as much as possible and that it is not made too simple or structured because otherwise the maximum number of identifiable columns might be reduced unnecessarily. Hence, in order to maximize the number of identifiable source locations or Directions Of Arrival, array configurations that are too structured, such as Uniform Linear or Rectangular arrays, should be avoided. In fact, this comes down to a trade-off between computational and algorithmic simplicity on the one hand, and exploiting all information under the parameterization restrictions on the other hand.

It is well-known that the higher the cumulant order is, the more data samples are required for the reliable estimation of the cumulants. Hence, to be able to deal with a certain maximum number of columns/sources, from a practical viewpoint a compromise needs to be made between the number of sensors on the one hand and the employed cumulant order on the other hand. For example, if it is impossible or undesirable to gather many samples or if there are much outliers in the data, one might prefer using more sensors instead of increasing the cumulant order in order to cope with a certain number of sources. However, if the employed sensors are very expensive, it may be desirable to increase the cumulant order instead of using more sensors. Apart from the cumulant order and the statistical properties of the data, for a given number of sensors and cumulant order, the number of samples required for the reliable estimation of the sensor cumulants also depends on the number of sources. This is due to the fact that the (higher order) correlation between the sensor signals increases when the number of sources increases. Finally, the number of samples required for reliably estimating the statistics also depends on the temporal structure of the data. For example, if the temporal diversity of the source signals is large, which is the case when the auto-cumulant

functions or spectra are quite different/diverse, then usually less samples are required than for a small temporal diversity. In the first case, the source auto-cumulant matrix is well-conditioned, whereas in the second it is not. It should be noted that the more source signals are present and the more columns need to be identified, the more temporal structure needs to be present and exploited. Conversely, the more temporal structure is present in the source signals, the more columns can be identified. Those trade-offs follow readily from the linear independence assumption on the source auto-cumulant functions, which for example implies that the cardinality of the Noise-Free ROS at least needs to be as large as the number of sources. In general, the minimum required number of time, lag, or time-lag index tuples depends on the specific problem, in particular the number of sources and sensors, the temporal structure in the data, and the employed conjugation pattern.

We have shown that the way the sensor cumulant samples are arranged in a matrix is crucial and may be chosen in several manners. In the literature usually a well-known fixed arrangement is considered and little or no attention is paid to this issue. However, much insight and possibilities are revealed by considering several arrangements incorporating more or less temporal information. The choice/definition of the subspace matrix (statistics) reflects a trade-off in the amount of temporal structure that is exploited. In general, temporal structure can be exploited for several purposes, such as increasing robustness, coping with underdetermined scenarios, etcetera. In fact, our method for exploiting the temporal structure in the data to the full extend is at one end of a whole range or spectrum of methods, whereas the methods, such as MUSIC, exploiting only lag-zero cumulants are at the other end.

8.3 Future research

In this section we discuss several open issues and directions for future research. Firstly, in the course of this thesis we have identified several unsolved or incompletely solved mathematical problems. Among other things, these include the generalization of GEVD algorithms to MMGEVD algorithms, the development of general MMGEVD algorithms that can also handle rectangular matrices, robust approximate MMGEVD algorithms, robust joint approximate diagonalization algorithms for higher order tensors, robust higher order tensor decomposition algorithms, and robust solution methods for systems of multivariate polynomial/polyconjugal equations with approximated coefficients. In our view, solution methods for those problems are particularly relevant for many signal processing problems and also for other engineering areas; hence they need to be addressed. Secondly, the algorithms that we have presented in this work operate in batch mode. A useful extension would be make them work in an online or adaptive manner. This would also enable the tracking of slowly varying mixing systems. Thirdly, more insight into the statistical performance and robustness of the various algorithms, and comparison to currently available algorithms, are needed. In this respect, it is useful to examine how the number of samples required for good performance depends on the problem properties, such as the number of sensors and sources, statistical order, the type of temporal of the source signals, and so on. Also, we still need to investigate manners for reducing the computational complexity for several MIBI scenarios. Finally, because many practical problems can be described more adequately by more complex mixing models, such as convolutive or nonlinear models, the MIBI method presented in this work needs to be extended to those models.

APPENDIX A

Notation and definitions

The notation used in this work has been set up in such a way that it allows a uniform description of the theory for any considered order of the temporal structure. It closely follows the conventions used in tensor calculus and uses both sub- and superscript indices, see [111], for example. According to these conventions, summations are typically performed over a pair of quantities one of which has a subscript index and the other a superscript index, e.g. $\sum_i v_i w^i$. A quantity that has to be raised to some power is put between parentheses and the power is put in the superscript position, e.g. v to the power i is denoted by $(v)^i$. We sometimes only make an exception to this rule for the power of the natural exponent e (i.e. the base of the natural logarithm). In connection with the use of sub- and superscript indices, a distinction is made between column vectors and row vectors.

A.1 Vectors, matrices and vector spaces

Column vectors are denoted by lower case boldface letters, e.g. $\mathbf{v}$, and their elements are indexed by subscript indices, e.g. $\mathbf{v} = [v_i]$. The vector space of real-valued (complex-valued) column vectors of length M is denoted by $\mathbb{R}_M\big(\mathbb{C}_M\big)$. If the length of $\mathbf{v}$ needs to be clear from its notation, we will write $\mathbf{v}_M$. The standard basis of $\mathbb{R}_M$ is given by the set of column vectors $\{\mathbf{e}^i\}^{1\leq i\leq M}$, where $\mathbf{e}^i$ is a column vector with all zeros except for a 1 at the i-th position. Thus a column vector $\mathbf{v} \in \mathbb{R}_M\big(\mathbb{C}_M\big)$ can be decomposed as $\mathbf{v} = \sum_{i=1}^M v_i \mathbf{e}^i$. Note that the column vector containing the basis coefficients $\{v_i\}_{i=1}^M$ of a column vector $\mathbf{v}$ with respect to the standard basis is exactly the same as the column vector itself. This does not hold for other bases.

Row vectors are denoted by lower case boldface letters with a tilde, e.g. $\tilde{\mathbf{v}}$, and their elements are indexed by superscript indices, e.g. $\tilde{\mathbf{v}} = [v^j]$. The vector space of real-valued (complex-valued) row vectors of length N is denoted by $\mathbb{R}^N\big(\mathbb{C}^N\big)$. If the length of $\tilde{\mathbf{v}}$ needs to be clear from its notation, we will write $\tilde{\mathbf{v}}^N$. The standard basis of $\mathbb{R}^N$ is given by the set of row vectors $\{\tilde{\mathbf{e}}_j\}_{1\leq j\leq N}$, where $\tilde{\mathbf{e}}_j$ is a row vector with all zeros except for a 1 at the j-th position. Thus a row vector $\tilde{\mathbf{v}}$ can be written as $\tilde{\mathbf{v}} = \sum_{j=1}^N v^j \tilde{\mathbf{e}}_j$. Again, note that the row vector containing the basis coefficients $\{v^j\}^{1\leq j\leq N}$ of a row vector $\tilde{\mathbf{v}}$ with respect to the standard basis is exactly the same as the row vector itself. Again, this does not hold for other bases.

Matrices are denoted by upper case boldface letters, e.g. $\mathbf{V}$. The elements of matrices are denoted by lower case letters with both sub- and superscript indices, e.g. $\mathbf{V} = \big[v_i^j\big]$, where the subscript indices correspond to row indices and the superscript indices to column indices. The vector space of matrices of size M by N, i.e. with M rows and N columns, containing real-valued (complex-valued) elements is denoted by $\mathbb{R}_M^N\big(\mathbb{C}_M^N\big)$. If the size of $\mathbf{V}$ needs to be clear from its notation, we will write $\mathbf{V}_M^N$. The i-th row of a matrix $\mathbf{V}$ is denoted by $\tilde{\mathbf{v}}_i$ and

the j-th column by $\mathbf{v}^j$. Usually, the dimensions of matrices and vectors are clear from the context, otherwise they are stated explicitly. The M by N zero matrix will be denoted by $\mathbf{0}_M^N$. *In general, the notation in this work is arranged in such a way that quantities with subscripts are stacked on top of each other when constructing a matrix or vector, whereas quantities with superscripts are stacked besides each other.* For example, this implies that the rows of a matrix $\mathbf{V}$ are denoted by $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2$, etc., whereas its columns are denoted by $\mathbf{v}^1, \mathbf{v}^2$, etc. A big advantage of using both sub- and superscripts is that the type of the involved quantity follows immediately from the notation. For example, v_i is an element of a column vector (because the index 'runs' in the 'row indexing direction'), whereas v^j is an element of a row vector (because the index 'runs' in the 'column indexing direction'). Likewise, $\tilde{\mathbf{v}}_i$ is a 'row vector element' of a matrix $\mathbf{V}$, whereas $\mathbf{v}^j$ is a 'column vector element', and so on.

The transpose, Hermitian (i.e. conjugate transpose) and pseudo-inverse of a matrix are denoted by $\mathbf{V}^T$, $\mathbf{V}^H$ and $\mathbf{V}^\dagger$ respectively. The inverse of a square matrix will be denoted by $\mathbf{V}^{-1}$. The Frobenius norm and trace [72] of a matrix $\mathbf{V}$ are denoted by $\|\mathbf{V}\|_{\mathrm{F}}$ and $\mathrm{tr}(\mathbf{V})$ respectively. Note that $\|\mathbf{V}\|_{\mathrm{F}}^2 = \mathrm{tr}(\mathbf{V}^H\mathbf{V}) = \sum_i \sum_j |v_i^j|^2$. The function that constructs the ordered set $\mathcal{D}_{\mathbf{V}}$ containing the diagonal elements of $\mathbf{V}$ is denoted by $\mathrm{diag}(\mathbf{V})$, whereas $\mathrm{diag}(\mathcal{D}_{\mathbf{V}})$ denotes the function that constructs a square diagonal matrix from the elements of $\mathcal{D}_{\mathbf{V}}$. Thus, the notation $\mathrm{diag}(\cdot)$ has a double use and its function is determined by its argument. The function that constructs the matrix $\mathrm{diag}\big(\mathrm{diag}(\mathbf{V})\big)$, i.e. the diagonal matrix built from the diagonal elements of $\mathbf{V}$, is denoted by $\mathrm{ddiag}(\mathbf{V})$. The notation $\mathrm{offd}(\mathbf{V})$ denotes the matrix $\mathbf{V}$ with zeroed diagonal elements, i.e. the off-diagonal part is extracted. Hence, apart from the zero diagonal elements $\mathrm{offd}(\mathbf{V})$ equals $\mathbf{V}$. Note that $\mathrm{offd}(\mathbf{V}) = \mathbf{V} - \mathrm{ddiag}(\mathbf{V})$. The sum of squared absolute values of the off-diagonal elements of $\mathbf{V}$ is denoted by $\mathrm{off}(\mathbf{V})$, i.e. $\mathrm{off}(\mathbf{V}) \triangleq \sum_{i\neq j} \left|v_j^i\right|^2$. The 'operator' $\mathrm{off}(\mathbf{V})$ is often used as a measure of diagonality of $\mathbf{V}$, and is also referred to as the Frobenius norm of the off-diagonal elements of $\mathbf{V}$. Note that $\mathrm{off}(\mathbf{V}) = \|\mathrm{offd}(\mathbf{V})\|_{\mathrm{F}}$.

The linear span of a set of vectors $\mathcal{V}$ in a vector space is denoted by $\mathcal{L}(\mathcal{V})$. E.g. suppose that $\mathcal{V} \triangleq \{\mathbf{v}^1, \ldots, \mathbf{v}^P\}$, then:

$$\mathcal{L}(\mathcal{V}) \triangleq \mathrm{span}\left(\mathbf{v}^1, \ldots, \mathbf{v}^P\right) = \left\{ \sum_{j=1}^{P} \lambda_j \mathbf{v}^j \;\middle|\; \lambda_1, \ldots, \lambda_S \in \mathbb{R}\,(\mathbb{C}) \right\}. \tag{A.1.1}$$

A.2 Tuples and sets

For uniformity and compactness of notation, some additional definitions concerning tuples and sets are made (see also next section). Let p and G be positive integers and define:

- $\mathbf{i}_p \triangleq (i_1, \ldots, i_p) \in \mathbb{Z}_p$ (A.2.1)
- $(i)_p \triangleq (i, \ldots, i) \in \mathbb{Z}_p$ (A.2.2)
- $\check{v}_{\mathbf{i}_p} \triangleq \prod_{m=1}^{p} v_{i_m} = v_{i_1} \cdots v_{i_p}$ (A.2.3)
- $\check{v}^{\mathbf{j}_p} \triangleq \prod_{m=1}^{p} v^{j_m} = v^{j_1} \cdots v^{j_p}$ (A.2.4)

- $\breve{v}_{\mathbf{i}_p}^{\mathbf{j}_p} \triangleq \prod_{m=1}^{p} v_{i_m}^{j_m} = v_{i_1}^{j_1} \cdots v_{i_p}^{j_p}$ (A.2.5)
- $\breve{v}_{i}^{\mathbf{j}_p} \triangleq \breve{v}_{(i)_p}^{\mathbf{j}_p} \overset{\text{(A.2.5)}}{=} \prod_{m=1}^{p} v_{i}^{j_m} = v_{i}^{j_1} \cdots v_{i}^{j_p}$ (A.2.6)
- $\breve{v}_{\mathbf{i}_p}^{j} \triangleq \breve{v}_{\mathbf{i}_p}^{(j)_p} \overset{\text{(A.2.5)}}{=} \prod_{m=1}^{p} v_{i_m}^{j} = v_{i_1}^{j} \cdots v_{i_p}^{j}$ (A.2.7)
- $\breve{v}^{\mathbf{c}_p} \triangleq \prod_{m=1}^{p} \left(v\right)^{c_m} = (v)^{c_1} \cdots (v)^{c_p}$ (A.2.8)
- $\breve{v}_{\mathbf{i}_p}^{\mathbf{c}_p} \triangleq \prod_{m=1}^{p} \left(v_{i_m}\right)^{c_m} = \left(v_{i_1}\right)^{c_1} \cdots \left(v_{i_p}\right)^{c_p}$ (A.2.9)
- $\breve{v}^{\mathbf{j}_p,\mathbf{c}_p} \triangleq \prod_{m=1}^{p} \left(v^{j_m}\right)^{c_m} = \left(v^{j_1}\right)^{c_1} \cdots \left(v^{j_p}\right)^{c_p}$ (A.2.10)
- $\breve{v}_{\mathbf{i}_p}^{\mathbf{j}_p,\mathbf{c}_p} \triangleq \prod_{m=1}^{p} \left(v_{i_m}^{j_m}\right)^{c_m} = \left(v_{i_1}^{j_1}\right)^{c_1} \cdots \left(v_{i_p}^{j_p}\right)^{c_p}$ (A.2.11)
- $\breve{v}_{i}^{\mathbf{j}_p,\mathbf{c}_p} \triangleq \breve{v}_{(i)_p}^{\mathbf{j}_p,\mathbf{c}_p} \overset{\text{(A.2.11)}}{=} \prod_{m=1}^{p} \left(v_{i}^{j_m}\right)^{c_m} = \left(v_{i}^{j_1}\right)^{c_1} \cdots \left(v_{i}^{j_p}\right)^{c_p}$ (A.2.12)
- $\breve{v}_{\mathbf{i}_p}^{j,\mathbf{c}_p} \triangleq \prod_{m=1}^{p} \left(v_{i_m}^{j}\right)^{c_m} = \left(v_{i_1}^{j}\right)^{c_1} \cdots \left(v_{i_p}^{j}\right)^{c_p}$ (A.2.13)
- $\mathcal{I}_{\text{t},G}^{p} \triangleq \{\mathbf{i}_p \mid 1 \leq i_1, \ldots, i_p \leq G\}, \quad \left|\mathcal{I}_{\text{t},G}^{p}\right| = \left(G\right)^{p}$ (A.2.14)
- $\mathcal{I}_{\text{a},G}^{p} \triangleq \{\mathbf{i}_p \mid 1 \leq i_1 \leq \cdots \leq i_p \leq G\}, \quad \left|\mathcal{I}_{\text{a},G}^{p}\right| = \binom{p+G-1}{p}$ (A.2.15)
- $\mathcal{I}_{\text{d},G}^{p} \triangleq \{\mathbf{i}_p \mid \mathbf{i}_p \in \mathcal{I}_{\text{a},G}^{p},\ i_m \neq i_n\ \forall\, m \neq n, 1 \leq m, n \leq p\}, \quad \left|\mathcal{I}_{\text{d},G}^{p}\right| = \binom{G}{p}$ (A.2.16)
- $\mathcal{I}_{\text{e},G}^{p} \triangleq \{(i)_p \mid 1 \leq i \leq G\}, \quad \left|\mathcal{I}_{\text{e},G}^{p}\right| = G$ (A.2.17)
- $\mathcal{I}_{\text{c},G}^{p} \triangleq \mathcal{I}_{\text{t},G}^{p} \setminus \mathcal{I}_{\text{e},G}^{p}, \quad \left|\mathcal{I}_{\text{c},G}^{p}\right| = (G)^{p} - G$ (A.2.18)
- $\mathcal{I}_{\text{o},G}^{p} \triangleq \{\mathbf{i}_p \in \mathcal{I}_{\text{t},G}^{p} \mid \exists$ at least one index value that occurs an odd number of times$\}$, (A.2.19)

where $|\mathcal{V}|$ denotes the cardinality of the set $\mathcal{V}$. An ordered tuple of integers like $\mathbf{i}_p$ will be used for three different purposes. Firstly, the indices in a tuple can be used as running variables in summations or products. Secondly, the indices in a tuple can be used as free indices. Finally, a tuple can contain discrete time indices, in which case it is used as a multi-dimensional argument to functions. Typically, the symbols $\mathbf{i}_p$ and $\mathbf{j}_p$ denote tuples used for the first two purposes, while the symbol $\mathbf{n}_p$ always denotes a tuple of discrete time indices. The tuple $(i)_p$ is a shorthand notation for $\mathbf{i}_p$ when all indices are equal. The symbols $\breve{v}_{\mathbf{i}_p}$, $\breve{v}^{\mathbf{j}_p}$ and $\breve{v}_{\mathbf{i}_p}^{\mathbf{j}_p}$ are merely shorthand notations for the products $v_{i_1} \cdots v_{i_p}$, $v^{j_1} \cdots v^{j_p}$ and $v_{i_1}^{j_1} \cdots v_{i_p}^{j_p}$ respectively. Likewise, the symbols $\breve{v}_{\mathbf{i}_p}^{\mathbf{c}_p}$, $\breve{v}^{\mathbf{j}_p,\mathbf{c}_p}$ and $\breve{v}_{\mathbf{i}_p}^{\mathbf{j}_p,\mathbf{c}_p}$ are merely shorthand notations for the

products $\left(v_{i_1}\right)^{c_1}\cdots\left(v_{i_p}\right)^{c_p}$, $\left(v^{j_1}\right)^{c_1}\cdots\left(v^{j_p}\right)^{c_p}$ and $\left(v_{i_1}^{j_1}\right)^{c_1}\cdots\left(v_{i_p}^{j_p}\right)^{c_p}$ respectively, where $\mathbf{c}_p \triangleq (c_1,\ldots,c_p)$ is a so-called *conjugation tuple* that is defined in Section A.5 and each of the symbols $c_1,\ldots,c_p$ can either be '$*$' or '$\circ$', meaning 'conjugation' and 'no conjugation' respectively.

Notations like $\mathcal{I}^p_{\text{t},G}, \mathcal{I}^p_{\text{a},G}, \mathcal{I}^p_{\text{d},G}$, etc. denote specific sets of index tuples. Each letter in such a notation has a specific unambiguous meaning, which we will explain by means of an example, viz. the set $\mathcal{I}^p_{\text{a},G}$ in (A.2.15) that is defined as $\mathcal{I}^p_{\text{a},G} \triangleq \{(i_1,\ldots,i_p) \mid 1 \le i_1 \le \cdots \le i_p \le G\}$. The meanings of the different parts of the notation $\mathcal{I}^p_{\text{a},G}$ are illustrated in Fig. A.1. The

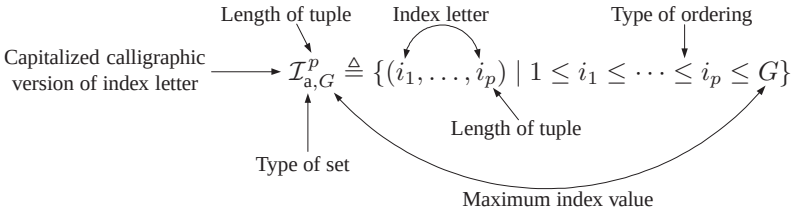

Figure A.1: *Index set notation.*

capitalized calligraphic letter $\mathcal{I}$ indicates that we are considering tuples with index letter i, i.e. tuples of the form $(i_1, i_2, \ldots)$. The superscript index letter p indicates the length of the tuples in the set, i.e. the set only contains tuples of the form $\mathbf{i}_p = (i_1,\ldots,i_p)$; see (A.2.1). The second subscript letter G indicates that each index $i_1,\ldots,i_p$ of $\mathbf{i}_p$ in principle can range from 1 to G, i.e. $1 \le i_1,\ldots,i_p \le G$. Finally, the first subscript index letter 'a' indicates the 'type' of the set that, in addition to the constraint $1 \le i_1,\ldots,i_p \le G$, possibly specifies other constraints on the values of the indices. In this example, the subscript 'a' stands for 'ascending' because the set $\mathcal{I}^p_{\text{a},G}$ contains only those tuples whose indices, in addition to the constraint $1 \le i_1,\ldots,i_p \le G$, are ordered in ascending order; see (A.2.15) and the boxed text on page 412. Likewise, the subscript 't' in $\mathcal{I}^p_{\text{t},G}$ stands for 'total' because the set contains all tuples obtained by varying all indices in a tuple from 1 till G, i.e. there are no further constraints; see (A.2.14). The subscript 'd' in $\mathcal{I}^p_{\text{d},G}$ stands for 'different' because the set contains only those tuples whose indices are different; see (A.2.16). The subscript 'e' in $\mathcal{I}^p_{\text{e},G}$ stands for 'equal' because the set contains only those tuples whose indices are equal; see (A.2.17). The subscript 'c' in $\mathcal{I}^p_{\text{c},G}$ stands for 'cross' because the set contains only those tuples whose indices are unequal or 'cross'; see (A.2.18). Finally, the subscript 'o' in $\mathcal{I}^p_{\text{o},G}$ stands for 'odd' because the set contains only those tuples that have at least one index value that occurs an odd number of times; see (A.2.19). *The advantage of this notation is that it makes it possible to compactly and uniformly specify sets of indices.* Also note that whenever a set of indices is denoted by $j_1,\ldots,j_k$ or a product by $j_1\cdots j_k$ it is meant that the subscripts run in ascending order from 1 to k. As an example, consider a situation in which we need the set of all index pairs (j_1, j_2) such that $1 \le j_1, j_2 \le 3$ and $j_1 \le j_2$, i.e. the set $\{(1,1),(1,2),(1,3),(2,2),(2,3),(3,3)\}$ consisting of six different elements. Then, in our notation this set is denoted by $\mathcal{J}^2_{\text{a},3}$, which is obtained by setting $G = 3$, $p = 2$ in (A.2.15) and using the capitalized letter $\mathcal{J}$. Note that the number of elements is given correctly by the second formula in (A.2.15): $\binom{G+p-1}{p} = \binom{3+2-1}{2} = \binom{4}{2} = 6$.

A.3 Definitions and results from combinatorics

Combinatorics concerns the calculation of the number of ways in which certain patterns can be formed. Here we recall some basic definitions and results used in the thesis [15, 74, 134].

Definition A.3.1. Set. A *set* is a collection of objects in which order and multiplicity are irrelevant.

We denote sets by curly braces. For example, $\{1,2,3\}$, $\{3,2,1\}$, $\{1,2,2,3\}$ and $\{2,1,3,3,2,3,1\}$ are the same according to the set-definition, whereas $\{1,2,3\}$ and $\{1,2,4\}$ are different.

Definition A.3.2. Multiset. A *multiset* is a collection of objects in which order is irrelevant, but multiplicity is explicitly significant.

We denote multisets by angle brackets. For example, $\langle 1,2,3,3\rangle$, $\langle 3,2,1,3\rangle$ and $\langle 3,2,3,1\rangle$ are the same according to the multiset-definition and the members 1, 2 and 3 have multiplicities 1, 1 and 2 respectively, whereas $\langle 1,2,3\rangle$, $\langle 1,1,2,3\rangle$ and $\langle 1,2,2,3\rangle$ are all different from each other.

Definition A.3.3. Tuple or ordered set. A *tuple* or *ordered set* is a collection of objects in which order and multiplicity are relevant.

We denote tuples by parentheses. For example, $(1,2,3)$, $(1,3,2)$ and $(1,2,2,3)$ are all different from each other according to the tuple-definition.

Now, we will define various concepts relating to permutations and combinations of elements in a set. For convenience of definition, we consider a set $\mathcal{G} \triangleq \{g_1,\ldots,g_G\}$ that contains G different elements, i.e. it has cardinality $G \triangleq |\mathcal{G}|$, from which we are to select p elements. As we will see, the selection of p elements can be done in various manners, each of which yields a certain number of possibilities.

Definition A.3.4. p-permutation without repetition of a set $\mathcal{G}$ with G different objects. A p-permutation without repetition of the set $\mathcal{G}$ with G different objects and $G \geq p$ is an arbitrary ordered selection of p different elements from this set. Hence, it is determined uniquely by a p-tuple constructed from different elements of $\mathcal{G}$. The number of different p-permutations without repetition of a set with G different elements is denoted by $P_{p,G}$ and equals:

$$P_{p,G} = G\cdot(G-1)\cdots(G-p+1) = \frac{G!}{(G-p)!}\,. \tag{A.3.1}$$

For example, all 2-permutations without repetition of the set $\{1,2,3,4\}$ are given by the set of 2-tuples $\{(1,2),(1,3),(1,4),(2,1),(2,3),(2,4),(3,1),(3,2),(3,4),(4,1),(4,2),(4,3)\}$ with cardinality $P_{2,4} = \frac{4!}{(4-2)!} = 12$.

Definition A.3.5. Permutation without repetition of a set $\mathcal{G}$ with G different objects, or simply permutation. A permutation without repetition of the set $\mathcal{G}$ with G different objects is an arbitrary bijection of $\mathcal{G}$ onto itself. This is equivalent to saying that it is a G-permutation of $\mathcal{G}$. Hence, it is determined uniquely by a G-tuple constructed from different elements of $\mathcal{G}$. The number of different permutations without repetition of a set with G different elements is denoted by P_G and equals:

$$P_G = P_{G,G} \overset{(A.3.1)}{=} G\cdot(G-1)\cdots 1 = G!\,. \tag{A.3.2}$$

If the word 'permutation' is used on its own, i.e. without the qualification 'without repetition', this type of permutation is meant.

For example, all possible permutations of the set $\{1, 2, 3\}$ are given by the set of 3-tuples $\{(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)\}$ with cardinality $P_3 = 3! = 6$.

Definition A.3.6. ***p-permutation with repetition of a set $\mathcal{G}$ with G different objects.*** A p-permutation with repetition of the set $\mathcal{G}$ with G different objects is an arbitrary ordered selection of p elements from this set, where the elements may be selected repeatedly. Hence, it is determined uniquely by a p-tuple constructed from the elements of $\mathcal{G}$, where the elements may be selected repeatedly. The number of different p-permutations with repetition of a set with G different elements is denoted by $P^{\mathrm{r}}_{p,G}$ and equals:

$$P^{\mathrm{r}}_{p,G} = (G)^p \,. \tag{A.3.3}$$

For example, all 2-permutations with repetition of the elements of $\{1, 2, 3\}$ are given by the set of 2-tuples $\{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)\}$ with cardinality $P^{\mathrm{r}}_{2,3} = (3)^2 = 9$.

Definition A.3.7. ***p-combination without repetition of a set $\mathcal{G}$ with G different objects.*** A p-combination without repetition of the set $\mathcal{G}$ with G different objects and $G \geq p$ is an arbitrary unordered selection of p different elements from this set. In other words, *a p-combination without repetition of the set $\mathcal{G}$ is an arbitrary p-subset of $\mathcal{G}$*. The number of different p-combinations without repetition of a set with G different elements is denoted by $C_{p,G}$ and equals the *binomial coefficient*:

$$C_{p,G} = \frac{G!}{p!(G-p)!} = \binom{G}{p}, \tag{A.3.4}$$

which is commonly termed *G choose p*.

For example, all 2-combinations without repetition of the set $\{1, 2, 3, 4\}$ are given by the set of 2-sets $\{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}\}$ with cardinality $C_{2,4} = \frac{4!}{2!(4-2)!} = 6$.

Definition A.3.8. ***p-combination with repetition of a set $\mathcal{G}$ with G different objects.*** A p-combination with repetition of a set $\mathcal{G}$ with G different objects is an arbitrary unordered selection of p elements from this set, where the elements may be selected repeatedly. In other words, *a p-combination with repetition of the set $\mathcal{G}$ is an arbitrary p-multiset of $\mathcal{G}$*. The number of different p-combinations with repetition of a set with G different elements, i.e. the number of p-multisets of $\mathcal{G}$, is denoted by $M_{p,G}$ and equals the so-called *multiset coefficient*:

$$M_{p,G} \triangleq \left\langle \begin{matrix} G \\ p \end{matrix} \right\rangle = \frac{(p+G-1)!}{p!(G-1)!} = \binom{p+G-1}{p} = \binom{p+G-1}{G-1}, \tag{A.3.5}$$

which is sometimes termed *G multichoose p*. See Table A.1 on the next page for values of $M_{p,G}$ for several combinations of p and G. A proof of (A.3.5) is given further in this section.

For example, all 2-combinations with repetition of the set $\{1, 2, 3, 4\}$ are given by the set of 2-multisets $\{\langle 1, 1\rangle, \langle 1, 2\rangle, \langle 1, 3\rangle, \langle 1, 4\rangle, \langle 2, 2\rangle, \langle 2, 3\rangle, \langle 2, 4\rangle, \langle 3, 3\rangle, \langle 3, 4\rangle, \langle 4, 4\rangle\}$ with cardinality $M_{2,4} = \frac{(2+4-1)!}{2!(4-1)!} = 10$. Since order is irrelevant in multisets, this set is the same as e.g. $\{\langle 1, 1\rangle, \langle 1, 2\rangle, \langle 1, 3\rangle, \langle 1, 4\rangle, \langle 2, 1\rangle, \langle 2, 2\rangle, \langle 2, 3\rangle, \langle 2, 4\rangle, \langle 3, 1\rangle, \langle 3, 2\rangle, \langle 3, 3\rangle, \langle 3, 4\rangle, \langle 4, 1\rangle, \langle 4, 2\rangle, \langle 4, 3\rangle, \langle 4, 4\rangle\}$, which also has cardinality 10 because it has only 10 different elements according to the definition of multisets (for example, elements like $\langle 1, 2\rangle$ and $\langle 2, 1\rangle$ belong to the same equivalence class).

Table A.1: *Multiset coefficients* $M_{p,G} = \left\langle {G \atop p} \right\rangle = \binom{p+G-1}{p} = \binom{p+G-1}{G-1}$.

$p \backslash G$	1	2	3	4	5
1	1	2	3	4	5
2	1	3	6	10	15
3	1	4	10	20	35
4	1	5	15	35	70
5	1	6	21	56	126

Because sets consisting of all possible p-multisets of some set $\mathcal{G}$ are very important in this thesis (see also further in this section), we will examine them in more detail by proving (A.3.5). Consider a set $\mathcal{G} \triangleq \{g_1, \ldots, g_G\}$ with cardinality $G \triangleq |\mathcal{G}|$, from which we are to construct and count the number of all possible p-multisets, i.e. all p-combinations with repetition. To start with, consider a specific p-multiset containing m_1 times element g_1, m_2 times element g_2, etcetera, i.e. m_i is the multiplicity of element g_i. Note that $m_1 + m_2 + \cdots + m_G = p$. This multiset can be represented uniquely by the following set:

$$\underbrace{\big\{\{\underbrace{g_1, \ldots, g_1}_{m_1}\}, \{\underbrace{g_2, \ldots, g_2}_{m_2}\}, \ldots, \{\underbrace{g_G, \ldots, g_G}_{m_G}\}\big\}}_{m_1+m_2+\cdots+m_G=p \text{ elements}} . \tag{A.3.6}$$

In turn this set can be uniquely represented by an ordered sequence of p arbitrary symbols denoting the elements of the set, e.g. bullets, and $G-1$ other arbitrary symbols denoting the 'transition to the next element', e.g. vertical bars, as follows:

$$\underbrace{\big(\underbrace{\bullet \cdots \bullet}_{m_1} \overbrace{|}^{G-1 \text{ bars}} \underbrace{\bullet \cdots \bullet}_{m_2} | \cdots | \underbrace{\bullet \cdots \bullet}_{m_G}\big)}_{m_1+\cdots+m_G=p \text{ bullets}} . \tag{A.3.7}$$

The number of bullets m_1 to the left of the first bar represents the number of repetitions (multiplicity) of g_1, the number of bullets m_2 between the first and second bars represents the number of repetitions of g_2, and so on. Finally, the number of bullets m_G to the right of the last bar represents the number of repetitions of g_G. The total number of bullets equals $m_1+m_2+\cdots+m_G = p$. Since there are G different elements in $\mathcal{G}$, the number of separating bars equals $G-1$. Hence, the total number of symbols in (A.3.7) including both bullets and bars equals $p+(G-1) = p+G-1$. Thus, we can conclude that the number of p-multisets of the set $\mathcal{G}$ equals the number of ways to arrange the $G-1$ bars among the $p+G-1$ symbols in (A.3.7), which equals the number of subsets of size $G-1$ in a set of size $p+G-1$ given by (A.3.4) as the binomial coefficient $p+G-1$ choose $G-1$. Equivalently, it is the number of ways to arrange the p bullets among the $p+G-1$ symbols, which is the number of subsets of size p of a set of size $p+G-1$ given by (A.3.4) as the binomial coefficient $p+G-1$ choose p. In conclusion, *the number of p-multisets of a set $\mathcal{G}$ with cardinality G is the same as the number of p-subsets of a set of size $p+G-1$*. This conclusion is expressed in various forms in (A.3.5).

A unique representation of a set containing all different p-multisets of some set $\mathcal{G}$ with cardinality G can be defined for example by the set containing exactly one element from each

multiset-equivalence class that is represented by a tuple whose elements are arranged according to some fixed ordering scheme, e.g. ascending order. Note that the tuple-representation is crucial because order is relevant now. This way, we obtain a set of the type in (A.2.15), which is very important for the work in this thesis. For the example given below Definition A.3.8, this results in the following unique set representation for the set of all *unique* 2-combinations with repetition of the set $\{1,2,3,4\}$ with 4 different objects (calling the index i for convenience): $\mathcal{I}^2_{\mathrm{a},4} = \{(1,1),(1,2),(1,3),(1,4),(2,2),(2,3),(2,4),(3,3),(3,4),(4,4)\}$. Obviously, the number of elements in such a set is given by (A.3.5) (see also (A.2.15)). Hence, in general we have:

> The set $\mathcal{I}^p_{\mathrm{a},G}$ with cardinality $M_{p,G}$ is a unique representation of the set containing all p-combinations with repetition (i.e. p-multisets) of some set $\mathcal{G}$ with cardinality G, where each multiset-equivalence class is represented by exactly one tuple whose elements are arranged in ascending order.

Definition A.3.9. Permutation without repetition of a p-multiset or p-tuple of a set $\mathcal{G} \triangleq \{g_1,\ldots,g_G\}$ with cardinality $G \triangleq |\mathcal{G}|$, where the multiset contains m_i times element g_i for all $1 \leq i \leq G$. A permutation without repetition of a p-multiset of a set $\mathcal{G} \triangleq \{g_1,\ldots,g_G\}$ with cardinality $G \triangleq |\mathcal{G}|$, where the multiset contains m_i times element g_i for all $1 \leq i \leq G$ (i.e. it is of the type in (A.3.6) with $m_1 + \cdots + m_G = p$) is an arbitrary ordered selection of the elements from that multiset. Hence, it is uniquely determined by a (p-)tuple constructed from the elements of the (p-)multiset. The number of different p-permutations of the type explained above is denoted by $P_{m_1,\ldots,m_G}$ and equals the *multinomial coefficient*:

$$P_{m_1,\ldots,m_G} = \frac{p!}{m_1!\cdots m_G!} = \frac{(m_1+\cdots+m_G)!}{m_1!\cdots m_G!} \triangleq \binom{p}{m_1,\ldots,m_G} = \binom{m_1+\cdots+m_G}{m_1,\ldots,m_G}. \tag{A.3.8}$$

Hence, $P_{m_1,\ldots,m_G}$ is simply the number of different permutations that can be formed from the elements of the multiset in (A.3.6). In other words, $P_{m_1,\ldots,m_G}$ is the number of distinct permutations in a multiset of G distinct elements of multiplicity m_i with $1 \leq i \leq G$. For example, let $\mathcal{G} = \{1,2,3\}$ and consider the 4-multiset $\langle 1,2,3,3\rangle$ of $\mathcal{G}$ where the multiplicities of the elements 1, 2 and 3 are given by $m_1 = 1$, $m_2 = 1$ and $m_3 = 2$ respectively. Then all distinct permutations without repetition of this multiset are given by the set of 4-tuples $\{(1,2,3,3),(1,3,2,3),(1,3,3,2),(2,1,3,3),(2,3,1,3),(2,3,3,1),(3,1,2,3),(3,1,3,2),(3,2,1,3),(3,2,3,1),(3,3,1,2),(3,3,2,1)\}$ with cardinality $P_{1,1,2} = \frac{(1+1+2)!}{1!1!2!} = 12$.

Finally, we introduce some notation regarding permutation functions that is used in the thesis. In fact, we have already defined the permutation function in Definition A.3.5. Here we define it more explicitly as a function.

Definition A.3.10. Permutation function on a set $\mathcal{G}$ with cardinality $G \triangleq |\mathcal{G}|$. A permutation function $\sigma_{\mathcal{G}}(\cdot)$ of order G on a set $\mathcal{G}$ with cardinality $G \triangleq |\mathcal{G}|$ is defined as an arbitrary bijective function from this set onto itself. The set of all different permutation functions of order G is denoted by $\Sigma_{\mathcal{G}}$ and contains $G!$ elements.

A.4 Kronecker delta functions

As usual, a Kronecker delta function [115] is used to denote a function whose value equals one if and only if all or several of its arguments are equal, and zero otherwise. Depending on the position of the arguments (indices in our case), it can take several forms, each of which can easily be generalized, e.g.:

- δ_{ij} equals 1 iff $i = j$ and 0 elsewhere;
- $\delta_{\mathbf{i}_p} = \delta_{i_1 i_2 \cdots i_p}$ equals 1 iff $i_1 = i_2 = \cdots = i_p$ and 0 elsewhere;
- δ^{ij} equals 1 iff $i = j$ and 0 elsewhere;
- $\delta^{\mathbf{i}_p} = \delta^{i_1 i_2 \cdots i_p}$ is 1 iff $i_1 = i_2 = \cdots = i_p$ and 0 elsewhere;
- δ_i^j equals 1 iff $i = j$ and 0 elsewhere;
- $\delta_{\mathbf{i}_p}^{\mathbf{j}_p} = \delta_{i_1 i_2 \cdots i_p}^{j_1 j_2 \cdots j_p}$ equals 1 iff $i_1 = j_1, i_2 = j_2, \ldots, i_p = j_p$ and 0 elsewhere;
- $\delta[m, n]$ equals 1 iff $m = n$ and 0 elsewhere;
- $\delta[\mathbf{n}_p] = \delta[n_1, n_2, \ldots, n_p]$ equals 1 iff $n_1 = n_2 = \cdots = n_p$ and 0 elsewhere.

It will always be clear which indices have to be equal.

A.5 Conjugation patterns and notation by tuples

The imaginary unit is denoted by $\jmath$ and complex conjugation by $(\cdot)^*$. On several occasions, it will turn out to be useful to explicitly denote conjugation or no conjugation by a symbol. The symbol c (which may be indexed) is used for this purpose. Its value can either be '$*$', which means conjugation, or '$\circ$', which means no conjugation. When $c = *$, the notation $(v)^c = (v)^*$ means that the variable v is conjugated. Similarly, when $c = \circ$, the notation $(v)^c = (v)^\circ$ means that the variable v is not conjugated, i.e. $(v)^\circ = v$. A length-p *conjugation tuple* $\mathbf{c}_p$ is defined as follows:

$$\mathbf{c}_p \triangleq (c_1, \ldots, c_p) \quad \text{with} \quad c_1, \ldots, c_p \in \{\circ, *\}\,, \tag{A.5.1}$$

i.e. each of the symbols $c_1, \ldots, c_p$ can either be $*$ or $\circ$. A length-p conjugation tuple acts upon a set of p variables in such a way that the j-th conjugation symbol c_j acts upon the j-th variable for all $1 \leq j \leq p$. For example, if $\mathbf{c}_p$ acts upon the set $\{v_{i_1}, \ldots, v_{i_p}\}$, the result is the set $\{(v_{i_1})^{c_1}, \ldots, (v_{i_p})^{c_p}\}$. When a certain operation is performed on a set of variables which are to be conjugated according to a certain conjugation tuple, this tuple is written in the right-most position of the upper-right corner of the symbol denoting the operation, separated from a possible index tuple by a comma. For example, the product $(v_{i_1})^{c_1} \cdots (v_{i_p})^{c_p}$ is denoted by $\breve{v}_{\mathbf{i}_p}^{\mathbf{c}_p}$. Likewise, the products $(v^{j_1})^{c_1} \cdots (v^{j_p})^{c_p}$ and $(v_{i_1}^{j_1})^{c_1} \cdots (v_{i_p}^{j_p})^{c_p}$ are denoted by $\breve{v}^{\mathbf{j}_p, \mathbf{c}_p}$ and $\breve{v}_{\mathbf{i}_p}^{\mathbf{j}_p, \mathbf{c}_p}$ respectively. Since a dedicated symbol $\mathbf{c}_p$ is used to denote a conjugation tuple, it cannot be confused with an index tuple in a symbol like $\breve{v}_{\mathbf{i}_p}^{\mathbf{c}_p}$ for example. The set of all possible length-p conjugation tuples is defined as follows:

$$\mathcal{C}_p \triangleq \{\mathbf{c}_p \mid c_1, \ldots, c_p \in \{\circ, *\}\} \quad \text{with} \quad |\mathcal{C}_p| = (2)^p\,. \tag{A.5.2}$$

The complement $\bar{\mathbf{c}}_p \triangleq (\bar{c}_1, \ldots, \bar{c}_p)$ of a conjugation tuple $\mathbf{c}_p \triangleq (c_1, \ldots, c_p)$ is obtained by replacing conjugations by 'no conjugations' and vice versa:

$$\bar{\mathbf{c}}_p \triangleq (\bar{c}_1, \ldots, \bar{c}_p), \quad \text{where} \quad \bar{\circ} = * \quad \text{and} \quad \bar{*} = \circ\,. \tag{A.5.3}$$

For example, $\left(\check{v}^{\mathbf{j}_p}_{\mathbf{i}_p}\right)^* = \check{v}^{\mathbf{j}_p,\bar{\mathbf{c}}_p}_{\mathbf{i}_p}$. Note that both $\mathbf{c}_p$ and $\bar{\mathbf{c}}_p$ contain 2^p different length-p conjugation tuples because each element can have two different values.

A.6 Signals, cumulant and moment functions, conjugation of arguments

Functions or signals with continuous arguments have their variable(s) between parentheses. For example, $\mathbf{v}(t)$ represents a signal column vector containing signal values at continuous time t and $v_i(t)$ denotes its i-th component. Discrete-time functions or signals have their time index (indices) between square brackets. For example, $\mathbf{v}[n]$ represents a signal column vector containing signal values at discrete time n and $v_i[n]$ denotes its i-th component. The operator that gives the expected value of a random variable is denoted by $E\{\cdot\}$. The MIBI problem considered in this thesis is tackled by exploiting the temporal structure in the data of some arbitrary order l by using l-th order *cumulant* and *moment functions*. Therefore, cumulants and moments play a key role in this work. In order to understand the theory developed in this thesis, it is necessary that the reader is familiar with the theory of moments and cumulants. Detailed explanations can be found in many books and articles, see [111] and [112], for example. See also Appendix B for a survey. Now, we introduce our notation for cumulant and moment functions. Given a length-G time dependent random vector $\mathbf{v}[n]$, the l-th order signal cumulant function $\kappa^v_{\mathbf{i}_l}[\mathbf{n}_l]$ and moment function $r^v_{\mathbf{i}_l}[\mathbf{n}_l]$ of a subset $\{v_{i_1}[n_1], \ldots, v_{i_l}[n_l]\}$ of l components indexed by $i_1, \ldots, i_l$ at possibly different times $n_1, \ldots, n_l$ are defined by:

$$\kappa^v_{\mathbf{i}_l}[\mathbf{n}_l] \triangleq \mathrm{cum}\Big(v_{i_1}[n_1], \ldots, v_{i_l}[n_l]\Big) \qquad \forall\, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},G}, \quad \forall\, \mathbf{n}_l \in \mathbb{Z}_l \tag{A.6.1}$$

and:

$$r^v_{\mathbf{i}_l}[\mathbf{n}_l] \triangleq \mathrm{mom}\Big(v_{i_1}[n_1], \ldots, v_{i_l}[n_l]\Big) \qquad \forall\, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},G}, \quad \forall\, \mathbf{n}_l \in \mathbb{Z}_l\,, \tag{A.6.2}$$

respectively, where $\mathrm{cum}(\cdot, \ldots, \cdot)$ and $\mathrm{mom}(\cdot, \ldots, \cdot)$ denote the multi-argument cumulant and moment functions (see Appendix B). Actually, the arguments of these functions can be conjugated in $(2)^l$ different ways. The most suitable *conjugation tuple* for a particular application depends on the type of the involved signals. See Section B.3 for a justification of this statement. In the sequel, it will be assumed that a specific length-l conjugation tuple $\mathbf{c}_l = (c_1, \ldots, c_l)$ has been chosen. As explained above, this tuple is written in the upper-right corner of the considered symbol or function (right after a possibly present index tuple). For example, when the arguments $v_{i_1}[n_1], \ldots, v_{i_l}[n_l]$ of the function $\mathrm{cum}(\cdot, \ldots, \cdot)$ in the definition of the signal cumulant function $\kappa^v_{\mathbf{i}_l}[\mathbf{n}_l]$ are conjugated according to $(c_1, \ldots, c_l)$, the resulting function is denoted by $\kappa^{v,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l]$ and written as:

$$\kappa^{v,\mathbf{c}_l}_{\mathbf{i}_l}[\mathbf{n}_l] \triangleq \mathrm{cum}\Big(\big(v_{i_1}[n_1]\big)^{c_1}, \ldots, \big(v_{i_l}[n_l]\big)^{c_l}\Big) \qquad \forall\, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},G}, \quad \forall\, \mathbf{n}_l \in \mathbb{Z}_l\,, \quad \forall\, \mathbf{c}_l \in \mathcal{C}_l\,, \tag{A.6.3}$$

where $\mathcal{C}_l$ is the set of all possible length-l conjugation tuples, i.e.:

$$\mathcal{C}_l \triangleq \big\{\mathbf{c}_l \mid c_1, \ldots, c_l \in \{\circ, *\}\big\} \quad \text{with} \quad |\mathcal{C}_l| = (2)^l\,.$$

Examples

In general, the two-dimensional (cross-)correlation or (cross-)moment function of the i_1-th and i_2-th components of a length-G time dependent random vector $\mathbf{v}[n]$ at two possibly different time indices n_1 and n_2 respectively, i.e. $v_{i_1}[n_1]$ and $v_{i_2}[n_2]$, is denoted by $r^{v}_{i_1 i_2}[n_1, n_2]$ and defined as follows:

$$\begin{aligned} &r^{v}_{i_1 i_2}[n_1, n_2] \triangleq \mathrm{mom}\left(v_{i_1}[n_1], v_{i_2}[n_2]\right) = E\big\{v_{i_1}[n_1] v_{i_2}[n_2]\big\} \\ &\forall\, 1 \leq i_1, i_2 \leq G, \quad \forall\, n_1, n_2 \in \mathbb{Z}\,, \end{aligned} \tag{A.6.4}$$

where $\mathrm{mom}(\cdot,\cdot)$ denotes the two-argument moment function discussed in Section B.2.6.2 (see (B.2.28)). Likewise, the two-dimensional (cross-)correlation or (cross-)moment function of the i-th component of a length-G time dependent random vector $\mathbf{v}[n]$ at time (index) n, i.e. $v_i[n]$, and the j-th component of another length-P random vector $\mathbf{w}[m]$ at time m, i.e. $w_j[m]$, is denoted by $r^{vw}_{ij}[n, m]$ and defined as follows:

$$\begin{aligned} &r^{vw}_{ij}[n, m] \triangleq \mathrm{mom}\left(v_i[n], w_j[m]\right) = E\big\{v_i[n] w_j[m]\big\} \\ &\forall\, 1 \leq i \leq G,\ 1 \leq j \leq P, \quad \forall\, n, m \in \mathbb{Z}\,. \end{aligned} \tag{A.6.5}$$

As we have explained above, since the arguments of the moment function $\mathrm{mom}(\cdot,\cdot)$ can be conjugated in different ways, the signal correlation function can also be defined in different ways. For the two-argument case a *conjugation tuple* is also called *conjugation pair*. For example, if the arguments $v_{i_1}[n_1]$ and $v_{i_2}[n_2]$ of the function $\mathrm{mom}(\cdot,\cdot)$ in definition (A.6.4) of the moment function $r^{v}_{i_1 i_2}[n_1, n_2]$ are conjugated according to (c_1, c_2), the resulting function is denoted by $r^{v,c_1 c_2}_{i_1 i_2}[n_1, n_2]$ and written as:

$$\begin{aligned} &r^{v,c_1 c_2}_{i_1 i_2}[n_1, n_2] \triangleq \mathrm{mom}\Big(\big(v_{i_1}[n_1]\big)^{c_1}, \big(v_{i_2}[n_2]\big)^{c_2}\Big) = E\Big\{\big(v_{i_1}[n_1]\big)^{c_1} \big(v_{i_2}[n_2]\big)^{c_2}\Big\} \\ &\forall\, 1 \leq i_1, i_2 \leq G, \quad \forall\, n_1, n_2 \in \mathbb{Z}, \quad \forall\, (c_1, c_2) \in \mathcal{C}_2\,, \end{aligned} \tag{A.6.6}$$

where the set $\mathcal{C}_2$ of all four possible conjugation pairs is defined in (A.5.2). For example, using the conjugation pair $(c_1, c_2) = (\circ, *)$ that is commonly used when employing second order statistics for complex-valued data, definition (A.6.6) becomes:

$$\begin{aligned} &r^{v,\circ *}_{i_1 i_2}[n_1, n_2] \triangleq \mathrm{mom}\Big(\big(v_{i_1}[n_1]\big)^{\circ}, \big(v_{i_2}[n_2]\big)^{*}\Big) = E\Big\{\big(v_{i_1}[n_1]\big)^{\circ} \big(v_{i_2}[n_2]\big)^{*}\Big\} \\ &\forall\, 1 \leq i_1, i_2 \leq G, \quad \forall\, n_1, n_2 \in \mathbb{Z}, \end{aligned} \tag{A.6.7}$$

Similarly, if the arguments $v_i[n]$ and $w_j[m]$ of the function $\mathrm{mom}(\cdot,\cdot)$ in definition (A.6.5) of the moment function $r^{vw}_{ij}[n, m]$ are conjugated according to (c_1, c_2), the resulting function is denoted by $r^{vw,c_1 c_2}_{ij}[n, m]$ and written as:

$$\begin{aligned} &r^{vw,c_1 c_2}_{ij}[n, m] \triangleq \mathrm{mom}\Big(\big(v_i[n]\big)^{c_1}, \big(w_j[m]\big)^{c_2}\Big) = E\Big\{\big(v_i[n]\big)^{c_1} \big(w_j[m]\big)^{c_2}\Big\} \\ &\forall\, 1 \leq i_1, i_2 \leq G, \quad \forall\, n, m \in \mathbb{Z}, \quad \forall\, (c_1, c_2) \in \mathcal{C}_2\,. \end{aligned} \tag{A.6.8}$$

A.7 Hadamard product

The *Hadamard product* $\mathbf{A} \odot \mathbf{B}$ of two matrices $\mathbf{A}$ and $\mathbf{B}$ with the same size $M \times N$ is the matrix of size $M \times N$ defined by their element-wise product [77, 115, 147, 179]:

$$\mathbf{A} \odot \mathbf{B} \triangleq \begin{bmatrix} a_1^1 b_1^1 & a_1^2 b_1^2 & \cdots & a_1^N b_1^N \\ a_2^1 b_2^1 & a_2^2 b_2^2 & \cdots & a_2^N b_2^N \\ \vdots & \vdots & \ddots & \vdots \\ a_M^1 b_M^1 & a_M^2 b_M^2 & \cdots & a_M^N b_M^N \end{bmatrix}. \tag{A.7.1}$$

It satisfies the following properties (assuming that all involved matrices have the proper sizes for the involved operations):

HP1: $\mathbf{A} \odot (\mathbf{B} \odot \mathbf{C}) = (\mathbf{A} \odot \mathbf{B}) \odot \mathbf{C} \triangleq \mathbf{A} \odot \mathbf{B} \odot \mathbf{C}$

HP2: $(\mathbf{A} + \mathbf{B}) \odot \mathbf{C} = (\mathbf{A} \odot \mathbf{C}) + (\mathbf{B} \odot \mathbf{C})$

HP3: $\mathbf{A} \odot (\mathbf{B} + \mathbf{C}) = (\mathbf{A} \odot \mathbf{B}) + (\mathbf{A} \odot \mathbf{C})$

HP4: $(\mathbf{A} \odot \mathbf{B})^H = \mathbf{A}^H \odot \mathbf{B}^H$.

HP5: $\mathbf{A} \odot \mathbf{B} = \mathbf{B} \odot \mathbf{A}$

HP6: $\mathbf{A} \geq 0, \mathbf{B} \geq 0 \implies \mathbf{A} \odot \mathbf{B} \geq 0$,

where $\mathbf{V} \geq 0$ means that the matrix $\mathbf{V}$ is positive semi-definite. Furthermore, for $\mathbf{a}, \mathbf{b} \in \mathbb{R}_M(\mathbb{C}_M)$ and $\tilde{\mathbf{c}}, \tilde{\mathbf{d}} \in \mathbb{R}^N(\mathbb{C}^N)$:

HP7: $(\mathbf{a} \odot \mathbf{b})(\tilde{\mathbf{c}} \odot \tilde{\mathbf{d}}) = (\mathbf{a}\tilde{\mathbf{c}}) \odot (\mathbf{b}\tilde{\mathbf{d}})$.

For convenience, we also define the *p-th order Hadamard product* $\mathbf{A}_{\odot}^{p}$ of a matrix $\mathbf{A}$ with size $M \times N$ as the following matrix of size $M \times N$:

$$\mathbf{A}_{\odot}^{p} \triangleq \underbrace{\mathbf{A} \odot \cdots \odot \mathbf{A}}_{p \text{ terms}} = \mathbf{A} \odot \mathbf{A}_{\odot}^{p-1} \tag{A.7.2}$$

with $\mathbf{A}_{\odot}^{1} = \mathbf{A}$. From $\mathbf{A}$ we also define the *p-th order Hadamard product with conjugation tuple* $\mathbf{c}_p \triangleq (c_1, \ldots, c_p)$, which is denoted by $\mathbf{A}_{\odot}^{\mathbf{c}_p}$ and defined as:

$$\mathbf{A}_{\odot}^{\mathbf{c}_p} \triangleq (\mathbf{A})^{c_1} \odot \cdots \odot (\mathbf{A})^{c_p}, \tag{A.7.3}$$

where each of the symbols $c_1, \ldots, c_p$ can either be '$*$' or '$\circ$', meaning 'conjugation' and 'no conjugation' respectively (see Section A.5).

A.8 Kronecker product

The *Kronecker product* $\mathbf{A} \otimes \mathbf{B}$ of two matrices $\mathbf{A}$ and $\mathbf{B}$ with sizes $M \times N$ and $P \times Q$ respectively is the matrix of size $MP \times NQ$ defined by [13, 21, 73, 115]:

$$\mathbf{A} \otimes \mathbf{B} \triangleq \begin{bmatrix} a_1^1 \mathbf{B} & a_1^2 \mathbf{B} & \cdots & a_1^N \mathbf{B} \\ a_2^1 \mathbf{B} & a_2^2 \mathbf{B} & \cdots & a_2^N \mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_M^1 \mathbf{B} & a_M^2 \mathbf{B} & \cdots & a_M^N \mathbf{B} \end{bmatrix}. \tag{A.8.1}$$

It satisfies the following properties (assuming that all involved matrices have the proper sizes for the involved operations):

KP1: $\mathbf{A} \otimes (\mathbf{B} \otimes \mathbf{C}) = (\mathbf{A} \otimes \mathbf{B}) \otimes \mathbf{C} \triangleq \mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}$

KP2: $(\mathbf{A} + \mathbf{B}) \otimes \mathbf{C} = (\mathbf{A} \otimes \mathbf{C}) + (\mathbf{B} \otimes \mathbf{C})$

KP3: $\mathbf{A} \otimes (\mathbf{B} + \mathbf{C}) = (\mathbf{A} \otimes \mathbf{B}) + (\mathbf{A} \otimes \mathbf{C})$

KP4: $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD})$

KP5: $(\mathbf{A}_1 \otimes \mathbf{B}_1)(\mathbf{A}_2 \otimes \mathbf{B}_2) \cdots (\mathbf{A}_p \otimes \mathbf{B}_p) = (\mathbf{A}_1\mathbf{A}_2 \cdots \mathbf{A}_p) \otimes (\mathbf{B}_1\mathbf{B}_2 \cdots \mathbf{B}_p)$

KP6: $(\mathbf{A} \otimes \mathbf{B})^H = \mathbf{A}^H \otimes \mathbf{B}^H$.

Furthermore, for $\mathbf{A} \in \mathbb{R}_M^M(\mathbb{C}_M^M)$ and $\mathbf{B} \in \mathbb{R}_N^N(\mathbb{C}_N^N)$:

KP7: $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$

KP8: $\mathrm{tr}(\mathbf{A} \otimes \mathbf{B}) = \mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{B})$

KP9: $\det(\mathbf{A} \otimes \mathbf{B}) = (\det(\mathbf{A}))^M (\det(\mathbf{B}))^N$

KP10: $\mathrm{rank}(\mathbf{A} \otimes \mathbf{B}) = \mathrm{rank}(\mathbf{A})\,\mathrm{rank}(\mathbf{B})$.

For convenience, we also define the *p-th order Kronecker product (power)* $\mathbf{A}_\otimes^p$ of a matrix $\mathbf{A}$ with size $M \times N$ as the following matrix of size $(M)^p \times (N)^p$:

$$\mathbf{A}_\otimes^p \triangleq \underbrace{\mathbf{A} \otimes \cdots \otimes \mathbf{A}}_{p \text{ terms}} = \mathbf{A} \otimes \mathbf{A}_\otimes^{p-1} \tag{A.8.2}$$

with $\mathbf{A}_\otimes^1 = \mathbf{A}$. From $\mathbf{A}$ we also define the *p-th order Kronecker product with conjugation tuple* $\mathbf{c}_p \triangleq (c_1, \ldots, c_p)$, which is denoted by $\mathbf{A}_\otimes^{\mathbf{c}_p}$ and defined as:

$$\mathbf{A}_\otimes^{\mathbf{c}_p} \triangleq (\mathbf{A})^{c_1} \otimes \cdots \otimes (\mathbf{A})^{c_p}\,. \tag{A.8.3}$$

A.9 Khatri-Rao product

The *Khatri-Rao product* $\mathbf{A} \diamond \mathbf{B}$ of two matrices $\mathbf{A}$ and $\mathbf{B}$ with sizes $M \times N$ and $P \times N$ respectively is the matrix of size $MP \times N$ defined as the column-wise Kronecker product of $\mathbf{A}$ and $\mathbf{B}$ [21, 95, 141], i.e.

$$\mathbf{A} \diamond \mathbf{B} \triangleq \begin{bmatrix} \mathbf{a}^1 \otimes \mathbf{b}^1 & \cdots & \mathbf{a}^N \otimes \mathbf{b}^N \end{bmatrix} = \begin{bmatrix} a_1^1\mathbf{b}^1 & \cdots & a_1^N\mathbf{b}^N \\ a_2^1\mathbf{b}^1 & \cdots & a_2^N\mathbf{b}^N \\ \vdots & \ddots & \vdots \\ a_M^1\mathbf{b}^1 & \cdots & a_M^N\mathbf{b}^N \end{bmatrix}. \tag{A.9.1}$$

It satisfies the following properties (assuming that all involved matrices have the proper sizes for the involved operations):

KRP1: $\mathbf{A} \diamond (\mathbf{B} \diamond \mathbf{C}) = (\mathbf{A} \diamond \mathbf{B}) \diamond \mathbf{C} \triangleq \mathbf{A} \diamond \mathbf{B} \diamond \mathbf{C}$

KRP2: $(\mathbf{A} + \mathbf{B}) \diamond \mathbf{C} = (\mathbf{A} \diamond \mathbf{C}) + (\mathbf{B} \diamond \mathbf{C})$

KRP3: $\mathbf{A} \diamond (\mathbf{B} + \mathbf{C}) = (\mathbf{A} \diamond \mathbf{B}) + (\mathbf{A} \diamond \mathbf{C})$

KRP4: $(\mathbf{A} \diamond \mathbf{B})^H (\mathbf{C} \diamond \mathbf{D}) = (\mathbf{A}^H\mathbf{C}) \odot (\mathbf{B}^H\mathbf{D})$

KRP5: $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \diamond \mathbf{D}) = (\mathbf{AC}) \diamond (\mathbf{BD})$.

KRP6: $(\mathbf{A} \diamond \mathbf{B})^H = \mathbf{A}^H \diamond \mathbf{B}^H$

For convenience, we also define the *p-th order Khatri-Rao product (power)* $\mathbf{A}_{\diamond}^{p}$ of a matrix $\mathbf{A}$ with size $M \times N$ as the following matrix of size $(M)^p \times N$:

$$\mathbf{A}_{\diamond}^{p} \triangleq \underbrace{\mathbf{A} \diamond \cdots \diamond \mathbf{A}}_{p \text{ terms}} = \mathbf{A} \diamond \mathbf{A}_{\diamond}^{p-1} \tag{A.9.2}$$

with $\mathbf{A}_{\diamond}^{1} = \mathbf{A}$. From $\mathbf{A}$ we also define the *p-th order Khatri-Rao product with conjugation tuple* $\mathbf{c}_p \triangleq (c_1, \ldots, c_p)$, which is denoted by $\mathbf{A}_{\diamond}^{\mathbf{c}_p}$ and defined as:

$$\mathbf{A}_{\diamond}^{\mathbf{c}_p} \triangleq (\mathbf{A})^{c_1} \diamond \cdots \diamond (\mathbf{A})^{c_p} . \tag{A.9.3}$$

Note that this latter product is equal to the following expression:

$$\mathbf{A}_{\diamond}^{\mathbf{c}_p} \triangleq \left[\left(\mathbf{a}^1\right)^{c_1} \otimes \cdots \otimes \left(\mathbf{a}^1\right)^{c_p} \quad \cdots\cdots \quad \left(\mathbf{a}^N\right)^{c_1} \otimes \cdots \otimes \left(\mathbf{a}^N\right)^{c_p} \right] . \tag{A.9.4}$$

For example, if $p = 2$, $\mathbf{c}_2 = (c_1, c_2) = (\circ, *)$, and $\mathbf{A} \in \mathbb{C}_D^S$ (see equations (3.3.11) and (3.3.13) in Section 3.3.2), then:

$$\mathbf{A}_{\diamond}^{\mathbf{c}_2} = \mathbf{A}_{\diamond}^{c_1 c_2} \triangleq \left[\mathbf{a}^1 \otimes \left(\mathbf{a}^1\right)^* \quad \cdots \quad \mathbf{a}^S \otimes \left(\mathbf{a}^S\right)^* \right] . \tag{A.9.5}$$

A.10 Inner products

In this section we define two essentially different inner product functions, viz. one between vectors and one between matrices. We only give complex-valued versions because the corresponding real-valued versions can easily be obtained by omitting conjugations. Let $\mathbf{v}, \mathbf{w} \in \mathbb{C}_P$ be two vectors in a complex inner product space. Then, the inner product is a function from $\mathbb{C}_P \times \mathbb{C}_P$ to $\mathbb{C}$ defined by:

$$\langle \mathbf{v}, \mathbf{w} \rangle \triangleq \sum_{p=1}^{P} v_p \, (w_p)^* . \tag{A.10.1}$$

Since the difference between a column vector and a row vector essentially is only a notational difference, we can also compute inner products between them as long as they have the same size. For example, let $\tilde{\mathbf{v}} \in \mathbb{R}^P$, $\mathbf{w} \in \mathbb{C}_P$ be two vectors in a complex inner product space. Then, the inner product is a function from $\mathbb{C}^P \times \mathbb{C}_P$ to $\mathbb{C}$ defined by:

$$\langle \tilde{\mathbf{v}}, \mathbf{w} \rangle \triangleq \sum_{p=1}^{P} v^p \, (w_p)^* , \tag{A.10.2}$$

and so on; see also Section D.2. In signal processing, the following definition of an inner product function between two dimensionally consistent matrices $\mathbf{V}$ and $\mathbf{W}$ belonging to an inner product space of complex-valued matrices is commonly employed:

$$\langle \mathbf{V}, \mathbf{W} \rangle \triangleq \mathrm{tr}(\mathbf{V}\mathbf{W}^H) . \tag{A.10.3}$$

APPENDIX B

Statistical prerequisites

In this appendix, a review is given of the definitions and concepts of statistics, probability theory, and stochastic processes, that are relevant for this thesis. Firstly, the concepts related to a single real-valued random variable are discussed, followed by those of a real-valued random vector. Then, the extension of the concepts to real-valued stochastic processes is presented and finally some comments regarding the extension to complex-valued random variables and processes are made. W assume that the reader is familiar with the basics of statistics and probability theory, so that (axiomatic) definitions of probability, probability spaces, events, random variables, probability density functions, stochastic processes, etcetera, are known. In this appendix we only discuss continuous-valued random variables. The material for this appendix has been compiled from several sources, mainly [80, 111, 112, 122, 154].

B.1 Characterizations of real-valued random variables

A continuous random variable can be characterized in several ways. Common descriptions employ the concept of a *cumulative distribution function (cdf)* and/or a *probability density function (pdf)*. These functions characterize a random variable completely. Another way of specifying the properties of a random variable completely is by giving its so-called *characteristic functions*. These functions uniquely determine the probability density function, and vice versa. Alternatively, some of the key aspects of a cdf or pdf concerning the mean, dispersion, (a)symmetry, etc, can be summarized by using a discrete set of parameters. These parameters can either be *moments* or *cumulants*, and are typically derived from the characteristic functions. All those concepts will be reviewed briefly in this section for real-valued continuous random variables.

B.1.1 Cumulative distribution and probability density functions

5 Suppose that a real-valued random variable v is given, and let $v_0 \in \mathbb{R}$ denote a specific value of v. The probability of the event $v \leq v_0$ is denoted by $\Pr\{v \leq v_0\}$. Clearly, $\Pr\{v \leq v_0\}$ is a function of v_0. This function is called the *cumulative distribution function (cdf)* of the random variable v and is defined by:

$$P_v(v_0) \triangleq \Pr\{v \leq v_0\} \qquad \forall \quad -\infty \leq v_0 \leq \infty\,. \tag{B.1.1}$$

The letter v in the subscript position of $P_v(v_0)$ denotes the considered random variable, whereas the variable v_0 represents the argument of $P_v(v_0)$ representing a specific value of v. In the sequel, v_0 will often be replaced by v, i.e. $P_v(v)$ is considered. In such a case, the difference between the random variable v in the subscript position on the one hand, and the argument v between parentheses on the other hand should be kept in mind. For continuous random variables, the cdf is a nonnegative nondecreasing continuous function whose values

lie in the interval $[0, 1]$. From the definition it also follows directly that $P_v(-\infty) = 0$ and $P_v(\infty) = 1$.

Usually, a random variable v is characterized in terms of its *probability density function (pdf)* $p_v(v_0)$ rather than its cdf $P_v(v_0)$. This function is defined as the derivative of the cdf:

$$p_v(v_0) = \left.\frac{dP_v(v)}{dv}\right|_{v=v_0} . \tag{B.1.2}$$

Hence, $P_v(v_0)$ can be (re-)obtained from $p_v(v_0)$ by integration:

$$P_v(v_0) = \int_{-\infty}^{v_0} p_v(v)\, dv\,. \tag{B.1.3}$$

Similarly to the cdf we usually write $p_v(v)$ instead of $p_v(v_0)$. Note that $\int_{-\infty}^{\infty} p_v(v)\, dv = 1$ because $P_v(\infty) = 1$. The probability density function $p_v(v_0)$ can be interpreted as a measure of the probability of the event $v_0 \leq v \leq v_0 + \Delta v$, where Δv is a small increment of the random variable v. Indeed, from (B.1.1) and (B.1.3) it follows that:

$$\begin{aligned}\Pr\{v_0 \leq v \leq v_0 + \Delta v\} &= \Pr\{v \leq v_0 + \Delta v\} - \Pr\{v \leq v_0\} \\ &= \int_{v_0}^{v_0+\Delta v} p_v(v)\, dv \approx p_v(v_0) \cdot \Delta v\,.\end{aligned} \tag{B.1.4}$$

Hence, the probability that the random variable v falls in a small region $v_0 \leq v \leq v_0 + \Delta v$ around v_0 equals (approximately) $p_v(v_0) \cdot \Delta v$, i.e. it is proportional to the pdf evaluated at v_0, i.e. $p_v(v_0)$. It also follows from (B.1.4) that the probability that v is in any arbitrary region $\mathcal{V} \in \mathbb{R}$ can be obtained by integrating the pdf over that region.

B.1.2 Characteristic functions

In this section, two functions are defined that will prove useful for the definition or derivation of moments and cumulants. The *First Characteristic Function (FCF)* $\phi^v(\omega)$ of the random variable v is defined as the Fourier transform of the pdf $p_v(v)$:

$$\phi^v(\omega) \triangleq \int_{-\infty}^{\infty} p_v(v)\, \exp(\jmath\omega v)\, dv\,. \tag{B.1.5}$$

where ω is the real-valued transformed variable corresponding to v (recall that $\jmath$ denotes the imaginary unit). Every pdf is uniquely specified by its FCF and vice versa [122]. It can be shown that the FCF exists for all pdf's [130]. As we will explain in the next section, the FCF $\phi^v(\omega)$ can be used to derive the moments of v. The *Second Characteristic Function (SCF)* $\psi^v(\omega)$ of the random variable v is defined as the natural logarithm of the FCF of v:

$$\psi^v(\omega) \triangleq \ln(\phi^v(\omega))\,. \tag{B.1.6}$$

The SCF $\psi^v(\omega)$ can be used to define the so-called *cumulants* of v, as will also be shown in the next section. Several properties of the FCF and SCF can be found in [122].

B.1.3 Expectations, moments, central moments, and cumulants

A random variable is completely characterized by its probability density function. However, in many cases of practical interest, the (exact) pdf of a random variable is unknown. Nevertheless, it is often possible to use *(mathematical) expectations* of certain functions of that

random variable for performing useful analyses and processing. Although these expectations are defined theoretically in terms of the pdf, they can usually be estimated from the data without explicit knowledge of the pdf.

B.1.3.1 Expectations

Let $g(v)$ denote a function of the random variable v. The (mathematical) *expectation* of $g(v)$ is denoted by $E\{g(v)\}$ and defined by:

$$E\{g(v)\} \triangleq \int_{-\infty}^{\infty} g(v)\, p_v(v)\, dv\,. \tag{B.1.7}$$

The usefulness of the expectation notation lies in the fact that $E\{\cdot\}$ can be considered as an operator in much the same way as the derivation operator $\frac{d\{\cdot\}}{dv}$ and integration operator $\int\{\cdot\}\, dv$ are considered as operators. Considering expectation as an operator $E\{\cdot\}$, the following important properties of $E\{\cdot\}$ may be listed, which hold irrespective of the nature of the particular random variable under study:

- *Linearity of Type I.* Let $g_1(v), \ldots, g_P(v)$ be a set of P functions of v whose expectations exist and let $c^1, \ldots, c^P$ be a set of constant coefficients, then:

$$E\left\{\sum_{i=1}^{P} c^i g_i(v)\right\} = \sum_{i=1}^{P} c^i E\left\{g_i(v)\right\}\,. \tag{B.1.8}$$

- *Linearity of Type II.* Let $v_1, \ldots, v_Q$ be a set of Q random variables whose expectations exist and let $c^1, \ldots, c^Q$ be a set of constant coefficients, then:

$$E\left\{\sum_{i=1}^{Q} c^i v_i\right\} = \sum_{i=1}^{Q} c^i E\left\{v_i\right\}\,. \tag{B.1.9}$$

- *Transformation invariance.* Let $y = g(v)$ be a function of v, then:

$$E\{y\} = E\{g(v)\} \qquad \text{or} \qquad \int_{-\infty}^{\infty} y\, p_y(y)\, dy = \int_{-\infty}^{\infty} g(v)\, p_v(v)\, dv\,. \tag{B.1.10}$$

As can be seen from the list, there are two essentially different types of linearity. In the first type the individual expectation terms at the right hand side of (B.1.8) are all computed with the same pdf $p_v(v)$. On the contrary, in the second type the individual expectation terms at the right hand side of (B.1.9) are computed with the different pdf's $p_{v_1}(v_1), \ldots, p_{v_G}(v_G)$ respectively. *Hence, the same notation $E\{\cdot\}$ is used for different variables and the probability density function that is relevant for a particular expectation operator is implicitly understood from the variable in its argument.* Note that this is also the case for the last property since the integrations are carried out over different probability density functions. In case of doubt, the variable that determines which pdf has to be taken for the computation of the expectation can be added as a subscript to the expectation operator, e.g. the notation $E_{v_i}\{v_i\}$ is equivalent to $E\left\{v_i\right\}$. As a final remark, we note that the two different types of linearity can be combined into one statement as follows. Let $g_1(\cdot), \ldots, g_P(\cdot)$ be a set of P functions, let $v_1, \ldots, v_Q$ be a set of Q random variables, and let $\{c^{ij}\}^{1\leq i\leq P, 1\leq j\leq Q}$ be a set of constant coefficients. Assuming that the expectation of $g_i(v_j)$ exists for each possible combination of i and j, the

following holds:

$$E\left\{\sum_{i=1}^{P}\sum_{j=1}^{Q} c^{ij}\, g_i(v_j)\right\} = \sum_{i=1}^{P}\sum_{j=1}^{Q} c^{ij}\, E\left\{g_i(v_j)\right\}, \tag{B.1.11}$$

where $E\left\{g_i(v_j)\right\} = E_{v_j}\{g_i(v_j)\}$.

The properties listed above can be proven using the definition of the expectation operator and the properties of probability density functions. They are useful because in practice they allow expressions containing expectations to be simplified without actually computing the involved integrals.

B.1.3.2 Moments

Moments of a random variable v are typical expectations used to characterize it. They are defined by taking $g(v) = (v)^l$ in (B.1.7), i.e. $g(v)$ is a power of v. Hence, the l-th order moment $r^{v,l}$ of the random variable v is defined by:

$$r^{v,l} \triangleq E\big\{(v)^l\big\} = \int_{-\infty}^{\infty} (v)^l\, p_v(v)\, dv \qquad \forall\, l \in \mathbb{N}. \tag{B.1.12}$$

In particular, the first-order moment $r^{v,1}$ is the *mean* of v, and the second order moment $r^{v,2}$ is the *mean-squared value* of v. Moments of order larger than two are usually called *higher order moments*.

Deriving the moments from the FCF

The moments of v can also be derived from the Taylor series expansion of the First Characteristic Function $\phi^v(\omega)$ of v around $\omega = 0$. Define the l-th derivative $\phi^{v,(l)}(\omega)$ of $\phi^v(\omega)$ w.r.t. ω as follows:

$$\phi^{v,(l)}(\omega) \triangleq \frac{d^{(l)}\phi^v(\omega)}{d\omega^{(l)}} = \frac{d\cdots d\phi^v(\omega)}{d\omega\cdots d\omega}.$$

Assuming that differentiation under the integral sign in (B.1.5) is allowed, it follows that:

$$\phi^{v,(l)}(\omega) \triangleq \frac{d^{(l)}\phi^v(\omega)}{d\omega^{(l)}} = \int_{-\infty}^{\infty} \frac{d^{(l)}\exp(\jmath\omega v)}{d\omega^{(l)}}\, p_v(v)\, dv = \int_{-\infty}^{\infty} (\jmath v)^l \exp(\jmath\omega v)\, p_v(v)\, dv.$$

Evaluating $\phi^{v,(l)}(\omega)$ at $\omega = 0$ under the assumption that all moments exist yields:

$$\phi^{v,(l)}(0) = \int_{-\infty}^{\infty} (\jmath v)^l\, p_v(v)\, dv \overset{\text{(B.1.12)}}{=} (\jmath)^l E\big\{(v)^l\big\} = (\jmath)^l\, r^{v,l}. \tag{B.1.13}$$

Hence, whenever all moments are finite the Taylor series expansion of the FCF around the origin is given by:

$$\phi^v(\omega) = \sum_{l=0}^{\infty} \frac{\phi^{v,(l)}(0)}{l!}\,(\omega)^l = \sum_{l=0}^{\infty} \frac{(\jmath)^l\, r^{v,l}}{l!}\,(\omega)^l = \sum_{l=0}^{\infty} r^{v,l}\,\frac{(\jmath\omega)^l}{l!}. \tag{B.1.14}$$

Thus, the l-th order Taylor coefficient of this expansion, i.e. the coefficient of the term $\frac{(\jmath\omega)^l}{l!}$, is the l-th order moment $r^{v,l} = E\big\{(v)^l\big\}$ of v. For this reason, the First Characteristic Function is also called *Moment Generating Function (MGF)*. Note that the result in (B.1.14) can also be derived in the following insightful manner. Firstly, from (B.1.5) and (B.1.7) it is noted

that $\phi^v(\omega) = E\{\exp(\jmath\omega v)\}$ by taking $g(v) = \exp(\jmath\omega v)$. Then, using the linearity property of the expectation operator and the standard Taylor expansion of the exponential function it follows that:

$$\phi^v(\omega) = E\{\exp(\jmath\omega v)\} = E\left\{\sum_{l=0}^{\infty} \frac{(\jmath\omega v)^l}{l!}\right\} = \sum_{l=0}^{\infty} E\left\{\frac{(\jmath\omega v)^l}{l!}\right\} = \sum_{l=0}^{\infty} r^{v,l}\,\frac{(\jmath\omega)^l}{l!}\,.$$

From the development above, in particular from (B.1.13), it is clear that the moments can also be defined formally as follows:

$$r^{v,l} = E\big\{(v)^l\big\} \triangleq \frac{\phi^{v,(l)}(0)}{(\jmath)^l} = (-\jmath)^l\,\frac{d^{(l)}\phi^v(\omega)}{d\omega^{(l)}}\bigg|_{\omega=0} \qquad \forall\, l \in \mathbb{N}\,. \tag{B.1.15}$$

B.1.3.3 Central moments; variance, skewness and kurtosis

For non-zero mean random variables usually another set of moments, the so-called *central moments* or *moments about the mean*, are used instead of the 'normal' moments defined above. These central moments are the moments of a random variable computed after its mean has been removed. They are obtained when $g(v)$ in (B.1.7) is defined as $g(v) = \big(v - E\{v\}\big)^l = \big(v - r^{v,1}\big)^l$. Hence, the l-th order central moment $\gamma^{v,l}$ of the random variable v is defined by:

$$\gamma^{v,l} \triangleq E\Big\{\big(v - E\{v\}\big)^l\Big\} = \int_{-\infty}^{\infty} \big(v - E\{v\}\big)^l\, p_v(v)\, dv \quad \forall\, l \in \mathbb{N}\,. \tag{B.1.16}$$

In particular, the first-order central moment $\gamma^{v,1} = 0$. The second order central moment $\gamma^{v,2}$ is the *variance* of v and is often denoted by $(\sigma^v)^2$. The quantity $\sigma^v = \sqrt{\gamma^{v,2}}$ is called the *standard deviation* of v and is a measure of the spread of the observed values of v around its mean $r^{v,1} = E\{v\}$. The third-order central moment $\gamma^{v,3}$ is called the *skewness* of v and characterizes the degree of asymmetry of the pdf $p_v(v)$ about its mean. The skewness is zero if the pdf is symmetric about its mean, positive if the shape skews to the right, and negative if the shape skews to the left. Often, the *normalized skewness*, defined as the dimensionless number $\gamma^{v,3}/(\sigma^v)^3$ is used to characterize the asymmetry of a distribution. The fourth order central moment $\gamma^{v,4}$ of v characterizes the degree of *flatness* or *peakedness* of the pdf $p_v(v)$ about its mean. Instead of the fourth order central moment, a related dimensionless quantity, called *normalized kurtosis*, is typically employed to measure the flatness or peakedness of a distribution w.r.t. a Gaussian distribution. The normalized kurtosis of v is denoted by $\mathrm{kurt}(v)$ and defined as follows:

$$\mathrm{kurt}(v) \triangleq E\left\{\left(\frac{v - E\{v\}}{\sigma^v}\right)^4\right\} - 3 = \frac{\gamma^{v,4}}{(\sigma^v)^4} - 3\,. \tag{B.1.17}$$

For a Gaussian pdf it can be shown that $\gamma^{v,4} = 3(\sigma^v)^4$, from which it follows that $\mathrm{kurt}(v) = 0$ when v is Gaussian distributed. It is in this sense that the kurtosis is 'normalized' when compared to the fourth order central moment, which is non-zero for a Gaussian random variable. The (normalized) kurtosis is a measure of the *non-Gaussianity* of a random variable. A distribution having zero kurtosis is called *mesokurtic* or *Gaussian*. A distribution having negative kurtosis is called *platykurtic* or *Sub-Gaussian*. Finally, a distribution having positive kurtosis is called *leptokurtic* or *Super-Gaussian*. Examples of platykurtic, mesokurtic, and leptokurtic distributions are the uniform distribution, Gaussian distribution, and Laplacian distributions, respectively, which are illustrated in Fig. B.1. Higher than fourth order

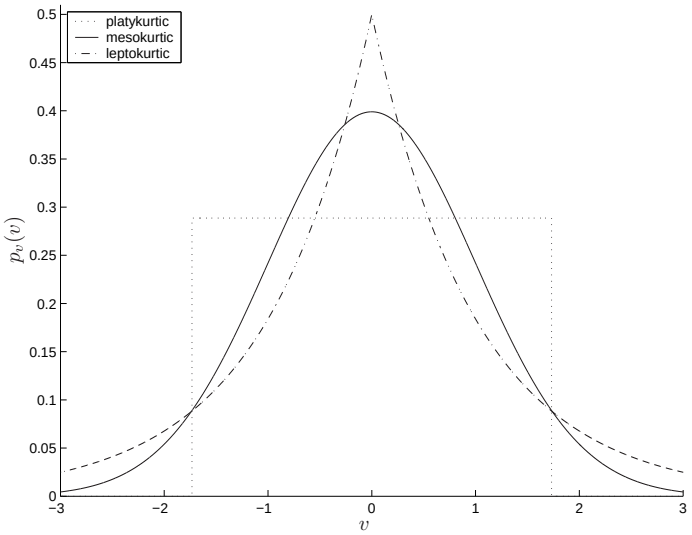

Figure B.1: *Platykurtic (uniform), mesokurtic (Gaussian) and leptokurtic (Super-Gaussian) distributions.*

statistics are rarely used in practice because usually large estimation errors are made, or large numbers of samples are necessary for a reasonably accurate estimation. Therefore, we will not discuss them in detail here.

We conclude this section by pointing out the relationship between moments and central moments. Using the binomial theorem, it can be shown that the moments and central moments of a random variable are related as follows:

$$\gamma^{v,l} = \sum_{k=0}^{l} \binom{l}{k} (-1)^k \left(r^{v,1}\right)^k r^{v,l-k}. \tag{B.1.18}$$

Throughout this thesis it is assumed that all random variables have zero mean. From (B.1.12) and (B.1.16) it then easily follows that moments and central moments are identical. The same result follows from (B.1.18) because only the term in the summation with $k = 0$ 'survives'. Consequently, moments and central moments can be used interchangeably in the work presented in the thesis.

B.1.3.4 Cumulants

Cumulants of a random variable v are another set of parameters that can be used to characterize the pdf $p_v(v)$ of v. Due to their specific properties cumulants are more convenient for higher order statistical analysis than higher order moments. The l-th order *cumulant* $\kappa^{v,l}$ of the random variable v is defined as the coefficient of the term $\frac{(\jmath\omega)^l}{l!}$ in the Taylor series expansion of the *Second Characteristic Function (SCF)* $\psi^v(\omega)$ around the origin:

$$\psi^v(\omega) \triangleq \ln(\phi^v(\omega)) = \sum_{l=0}^{\infty} \frac{\psi^{v,(l)}(0)}{l!} (\omega)^l \triangleq \sum_{l=0}^{\infty} \kappa^{v,l} \frac{(\jmath\omega)^l}{l!}, \tag{B.1.19}$$

where $\psi^{v,(l)}(\omega)$ is the l-th derivative of $\psi^v(\omega)$ w.r.t. ω, i.e. $\psi^{v,(l)}(\omega) \triangleq \frac{d^{(l)}\psi^v(\omega)}{d\omega^{(l)}} = \frac{d\cdots d\psi^v(\omega)}{d\omega\cdots d\omega}$. Hence, the l-th order cumulant $\kappa^{v,l}$ is defined as (note the similarity to (B.1.15)):

$$\kappa^{v,l} \triangleq \frac{\psi^{v,(l)}(0)}{(\jmath)^l} = (-\jmath)^l \left.\frac{d^{(l)}\psi^v(\omega)}{d\omega^{(l)}}\right|_{\omega=0} \qquad \forall\, l \in \mathbb{N}\,. \tag{B.1.20}$$

For this reason, the Second Characteristic Function is also called *Cumulant Generating Function (CGF)*.

Usually cumulants are preferred over moments when considering higher order statistics because they present in a clearer and better way the higher order statistical information. It can be shown that cumulants have the following important properties not shared by moments:

UCP1: *Cumulants are additive in their argument. Let $v_1, \ldots, v_P$ be P statistically independent random variables[‡]. Then, the cumulant of their sum $w \triangleq \sum_{i=1}^{P} v_i$ is equal to the sum of the individual cumulants, i.e. $\kappa^{w,l} = \sum_{i=1}^{P} \kappa^{v_i,l}$.*

UCP2: *Cumulants measure deviation from non-Gaussianity. If the pdf of a random variable v is Gaussian, all cumulants of order larger than two are identically zero.*

The first property justifies the name 'cumulant', whereas the second property implies that cumulants of order larger than two measure the departure of a random variable from a Gaussian random variable with the same mean and variance. It is in this sense that cumulants measure non-Gaussianity. Cumulants of order larger than two are usually called *higher order cumulants*.

In the next section, we will see that cumulants are closely related to (central) moments and thus they also characterize the shape of the involved probability density function by features like mean, variance, skewness, peakedness, etcetera, see Section B.1.3.3.

B.1.3.5 Expressing cumulants in moments

It is possible to express cumulants in terms of sums of products of moments, and vice versa. Hence, cumulants and moments convey the same statistical information. In this thesis, we are only interested in the conversion of moments to cumulants, and not vice versa. Therefore, only the moment-to-cumulant formula is discussed. In order to express the l-th order cumulant $\kappa^{v,l}$ of v in the moments $r^{v,k}$ up to order l, i.e. $1 \leq k \leq l$, all possible partitions of the set $\mathcal{I} \triangleq \{1, \ldots, l\}$ have to be considered.

Definition B.1.1. Partition of a set of integers. A partition of the set $\mathcal{I} \triangleq \{1, \ldots, l\}$ is an unordered collection of disjoint non-empty sets $\mathcal{I}^j$ such that $\cup_j \mathcal{I}^j = \mathcal{I}$.

For example, for $l = 3$ all possible partitions of the set $\mathcal{I} = \{1, 2, 3\}$ are given by $\{(1, 2, 3)\}$, $\{(1), (2, 3)\}, \{(2), (1, 3)\}, \{(3), (1, 2)\}$ and $\{(1), (2), (3)\}$. The set $\mathcal{P}$ is defined as the set containing all those partitions. Hence, in this case $\mathcal{P} \triangleq \big\{\{(1, 2, 3)\}, \{(1), (2, 3)\}, \{(2), (1, 3)\}, \{(3), (1, 2)\}, \{(1), (2), (3)\}\big\}$. Each partition in a set of partitions $\mathcal{P}$ is denoted by a set $\mathcal{P}_i$, where $1 \leq i \leq |\mathcal{P}|$ (remember that the order is not important). In the current example, there are five possible partitions, and thus $|\mathcal{P}| = 5$ and $\mathcal{P}$ can be written as $\mathcal{P} \triangleq \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4, \mathcal{P}_5\}$. For example, we may choose $\mathcal{P}_1 = \{(1, 2, 3)\}, \mathcal{P}_2 = \{(1), (2, 3)\}$, $\mathcal{P}_3 = \{(2), (1, 3)\}, \mathcal{P}_4 = \{(3), (1, 2)\}, \mathcal{P}_5 = \{(1), (2), (3)\}$, see Table B.1. The number

[‡]See Section B.2.3 for the definition of statistical independence.

of elements in each partition $\mathcal{P}_i$ is denoted by its cardinality $|\mathcal{P}_i|$. Hence, in the example $|\mathcal{P}_1| = 1$, $|\mathcal{P}_2| = |\mathcal{P}_3| = |\mathcal{P}_4| = 2$ and $|\mathcal{P}_5| = 3$. The j-th set of the i-th partition $\mathcal{P}_i$ is denoted by $\mathcal{I}_i^j$ (see Definition B.1.1 and Table B.1), i.e. $\mathcal{P}_i$ is written as $\mathcal{P}_i = \left\{\mathcal{I}_i^1, \ldots, \mathcal{I}_i^{|\mathcal{P}_i|}\right\}$. For the example, $\mathcal{P}_1 = \left\{\mathcal{I}_1^1\right\}$ with $\mathcal{I}_1^1 = (1,2,3)$, $\mathcal{P}_2 = \left\{\mathcal{I}_2^1, \mathcal{I}_2^2\right\}$ with $\mathcal{I}_2^1 = (1)$ and $\mathcal{I}_2^2 = (2,3)$, $\mathcal{P}_3 = \left\{\mathcal{I}_3^1, \mathcal{I}_3^2\right\}$ with $\mathcal{I}_3^1 = (2)$ and $\mathcal{I}_3^2 = (1,3)$, $\mathcal{P}_4 = \left\{\mathcal{I}_4^1, \mathcal{I}_4^2\right\}$ with $\mathcal{I}_4^1 = (3)$ and $\mathcal{I}_4^2 = (1,2)$, and finally $\mathcal{P}_5 = \left\{\mathcal{I}_5^1, \mathcal{I}_5^2, \mathcal{I}_5^3\right\}$ with $\mathcal{I}_5^1 = (1), \mathcal{I}_5^2 = (2)$ and $\mathcal{I}_5^3 = (3)$. The number of elements in each set $\mathcal{I}_i^j$ is denoted by its cardinality $|\mathcal{I}_i^j|$. Hence, $\left|\mathcal{I}_1^1\right| = 3$, $\left|\mathcal{I}_2^1\right| = 1$, $\left|\mathcal{I}_2^2\right| = 2$, $\left|\mathcal{I}_3^1\right| = 1$, $\left|\mathcal{I}_3^2\right| = 2$, $\left|\mathcal{I}_4^1\right| = 1$, $\left|\mathcal{I}_4^2\right| = 2$, and finally $\left|\mathcal{I}_5^1\right| = \left|\mathcal{I}_5^2\right| = \left|\mathcal{I}_5^3\right| = 1$. All results obtained so far for the example above are summarized in Table B.1. Now we are in a position to formulate the moment-to-cumulant conversion formula [112] as follows:

$$\kappa^{v,l} = \sum_{i=1}^{|\mathcal{P}|} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} r^{v,|\mathcal{I}_i^j|}\,. \tag{B.1.21}$$

Note that $r^{v,|\mathcal{I}_i^j|}$ denotes the moment of v of order equal to the cardinality of $\mathcal{I}_i^j$. Applying this formula to the example above yields:

$$\begin{aligned}
\kappa^{v,3} &= \sum_{i=1}^{5} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} r^{v,|\mathcal{I}_i^j|} = (-1)^{1-1}(1-1)! \prod_{j=1}^{1} r^{v,|\mathcal{I}_1^j|} \\
&\quad + (-1)^{2-1}(2-1)! \prod_{j=1}^{2} r^{v,|\mathcal{I}_2^j|} + (-1)^{2-1}(2-1)! \prod_{j=1}^{2} r^{v,|\mathcal{I}_3^j|} \\
&\quad + (-1)^{2-1}(2-1)! \prod_{j=1}^{2} r^{v,|\mathcal{I}_4^j|} + (-1)^{3-1}(3-1)! \prod_{j=1}^{3} r^{v,|\mathcal{I}_5^j|} \\
&= r^{v,|\mathcal{I}_1^1|} - r^{v,|\mathcal{I}_2^1|}\, r^{v,|\mathcal{I}_2^2|} - r^{v,|\mathcal{I}_3^1|}\, r^{v,|\mathcal{I}_3^2|} - r^{v,|\mathcal{I}_4^1|}\, r^{v,|\mathcal{I}_4^2|} + 2\, r^{v,|\mathcal{I}_5^1|}\, r^{v,|\mathcal{I}_5^2|}\, r^{v,|\mathcal{I}_5^3|} \\
&= r^{v,3} - 3\, r^{v,1}\, r^{v,2} + 2\, (r^{v,1})^3.
\end{aligned}$$

Hence, if v has zero mean, then $\kappa^{v,3} = r^{v,3}$.

Table B.1: *Example of partitioning* $\mathcal{I} = \{1,2,3\}$.

i	$\mathcal{P}_i$	$\lvert\mathcal{P}_i\rvert$	j	$\mathcal{I}_i^j$	$\lvert\mathcal{I}_i^j\rvert$
1	$\{(1,2,3)\}$	1	1	$(1,2,3)$	3
2	$\{(1),(2,3)\}$	2	1	(1)	1
			2	$(2,3)$	2
3	$\{(2),(1,3)\}$	2	1	(2)	1
			2	$(1,3)$	2
4	$\{(3),(1,2)\}$	2	1	(3)	1
			2	$(1,2)$	2
5	$\{(1),(2),(3)\}$	3	1	(1)	1
			2	(2)	1
			3	(3)	1

As another example, consider $l = 4$. All relevant information about the partitioning of the set $\mathcal{I} = \{1, \ldots, l\} = \{1, 2, 3, 4\}$ is given in Table B.2 on the following page. The last column gives the value of the i-th product term $\prod_{j=1}^{|\mathcal{P}_i|} r^{v,|\mathcal{I}_i^j|}$ in (B.1.21). Note that $|\mathcal{P}| = 15$ and thus i ranges from 1 to 15. Computing (B.1.21) using the results in the table yields:

$$\begin{aligned}\kappa^{v,4} &= \sum_{i=1}^{15} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} r^{v,|\mathcal{I}_i^j|} \\ &= r^{v,4} - 4\, r^{v,1}\, r^{v,3} - 3\left(r^{v,2}\right)^2 + 12\left(r^{v,1}\right)^2 r^{v,2} - 6\left(r^{v,1}\right)^4, \end{aligned} \tag{B.1.22}$$

or, in terms of expectations:

$$\begin{aligned}\kappa^{v,4} = E\{(v)^4\} - 4\, E\{v\}\, E\{(v)^3\} - 3\, \left(E\{(v)^2\}\right)^2 \\ + 12\left(E\{v\}\right)^2 E\{(v)^2\} - 6\left(E\{v\}\right)^4. \end{aligned} \tag{B.1.23}$$

Applying (B.1.21) for $l = 1$ and $l = 2$ yields the familiar expressions

$$\kappa^{v,1} = r^{v,1} = E\{v\}, \tag{B.1.24}$$

respectively:

$$\kappa^{v,2} = r^{v,2} - \left(r^{v,1}\right)^2 = E\{(v)^2\} - \left(E\{v\}\right)^2. \tag{B.1.25}$$

For zero mean variables, the expressions for the first four cumulants of v become respectively:

$$\kappa^{v,1} = 0\,; \tag{B.1.26a}$$

$$\kappa^{v,2} = r^{v,2} = E\{(v)^2\}\,; \tag{B.1.26b}$$

$$\kappa^{v,3} = r^{v,3} = E\{(v)^3\}\,; \tag{B.1.26c}$$

$$\kappa^{v,4} = r^{v,4} - 3\left(r^{v,2}\right)^2 = E\{(v)^4\} - 3\, \left(E\{(v)^2\}\right)^2\,. \tag{B.1.26d}$$

Dividing (B.1.26d) by $\left(r^{v,2}\right)^2$, recalling that $r^{v,l} = \gamma^{v,l}$ for zero mean variables, and comparing the result to (B.1.17), we see that the kurtosis of a zero mean random variable v is actually the fourth order cumulant of v normalized by the squared variance of v:

$$\frac{\kappa^{v,4}}{\left(r^{v,2}\right)^2} = \frac{r^{v,4}}{\left(r^{v,2}\right)^2} - 3 = \frac{E\{(v)^4\}}{\left(E\{(v)^2\}\right)^2} - 3 = \frac{\gamma^{v,4}}{(\sigma^v)^4} - 3 = \mathrm{kurt}(v)\,. \tag{B.1.27}$$

Formula (B.1.21) for expressing cumulants in terms of moments is needed when cumulants have to be estimated from data. In such a case, first the moments are estimated and then (B.1.21) is used to compute the estimated cumulants.

Table B.2: *Example of partitioning* $\mathcal{I} = \{1, 2, 3, 4\}$.

i	$\mathcal{P}_i$	$\lvert\mathcal{P}_i\rvert$	j	$\mathcal{I}_i^j$	$\lvert\mathcal{I}_i^j\rvert$	$\prod_{j=1}^{\lvert\mathcal{P}_i\rvert} r^{v,\lvert\mathcal{I}_i^j\rvert}$
1	$\{(1,2,3,4)\}$	1	1	$(1,2,3,4)$	4	$r^{v,4}$
2	$\{(1),(2,3,4)\}$	2	1	(1)	1	
			2	$(2,3,4)$	3	$r^{v,1}\, r^{v,3}$
3	$\{(2),(1,3,4)\}$	2	1	(2)	1	
			2	$(1,3,4)$	3	$r^{v,1}\, r^{v,3}$
4	$\{(3),(1,2,4)\}$	2	1	(3)	1	
			2	$(1,2,4)$	3	$r^{v,1}\, r^{v,3}$
5	$\{(4),(1,2,3)\}$	2	1	(4)	1	
			2	$(1,2,3)$	3	$r^{v,1}\, r^{v,3}$
6	$\{(1,2),(3,4)\}$	2	1	$(1,2)$	2	
			2	$(3,4)$	2	$\left(r^{v,2}\right)^2$
7	$\{(1,3),(2,4)\}$	2	1	$(1,3)$	2	
			2	$(2,4)$	2	$\left(r^{v,2}\right)^2$
8	$\{(1,4),(2,3)\}$	2	1	$(1,4)$	2	
			2	$(2,3)$	2	$\left(r^{v,2}\right)^2$
9	$\{(1),(2),(3,4)\}$	3	1	(1)	1	
			2	(2)	1	
			3	$(3,4)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
10	$\{(1),(3),(2,4)\}$	3	1	(1)	1	
			2	(3)	1	
			3	$(2,4)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
11	$\{(1),(4),(2,3)\}$	3	1	(1)	1	
			2	(4)	1	
			3	$(2,3)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
12	$\{(2),(3),(1,4)\}$	3	1	(2)	1	
			2	(3)	1	
			3	$(1,4)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
13	$\{(2),(4),(1,3)\}$	3	1	(2)	1	
			2	(4)	1	
			3	$(1,3)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
14	$\{(3),(4),(1,2)\}$	3	1	(3)	1	
			2	(4)	1	
			3	$(1,2)$	2	$\left(r^{v,1}\right)^2 r^{v,2}$
15	$\{(1),(2),(3),(4)\}$	4	1	(1)	1	
			2	(2)	1	
			3	(3)	1	
			4	(4)	1	$\left(r^{v,1}\right)^4$

B.1.3.6 Estimation of expectations, (central) moments and cumulants

The estimation or computation of expectations, moments, central moments, cumulants and other statistical characterizations from data is the subject of estimation theory and is treated in several books. Suppose that the expectation $E\{g(v)\}$ in (B.1.7) has to be estimated. We assume that the involved probability density function $p_v(v)$ is unknown, but that instead a set $\{v_1, \ldots, v_K\}$ of K samples of v is available. Then, a simple standard estimator is given by [154]:

$$\hat{g}(v) = \frac{1}{K} \sum_{k=1}^{K} g(v_k)\,, \tag{B.1.28}$$

where the hat over g in $\hat{g}(v)$ is a standard notation for the estimator/estimate of a quantity. In general, if the estimator is consistent and the number of samples sufficiently large, then $E\{g(v)\} \approx \hat{g}(v)$. Since moments are obtained for special choices of $g(v)$, viz. $g(v) = (v)^k$ for the k-th moment of v, they can be estimated using (B.1.28). Then, as has been remarked at the end of the previous section, the estimated value of a cumulant can be computed by means of (B.1.21), where the ideal moments $r^{v,k}$ are replaced by their estimates $\hat{r}^{v,k}$:

$$\hat{\kappa}^{v,l} = \sum_{i=1}^{|\mathcal{P}|} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} \hat{r}^{v,|\mathcal{I}_i^j|}\,. \tag{B.1.29}$$

B.2 Characterizations of real-valued random vectors

In this thesis we often consider groups of signal observations. One such a group can be modeled as a collection of random variables. For convenience, these variables are usually stacked in a vector, thus forming a *random vector*. This section discusses the generalizations and extensions of the concepts treated in Section B.1.

B.2.1 Joint cumulative distribution and probability density functions

Suppose that a length-G real-valued random (column) vector $\mathbf{v}$ is given:

$$\mathbf{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_G \end{bmatrix},$$

where the *components* $v_1, \ldots, v_G$ are real-valued random variables. Let $\mathbf{v}_0 \in \mathbb{R}_G$ denote a specific value of $\mathbf{v}$. Hence, the components $v_{0,1}, \ldots, v_{0,G}$ of $\mathbf{v}_0$ are considered as fixed real numbers at this stage. The probability of the event $\mathbf{v} \leq \mathbf{v}_0$, which means that $v_1 \leq v_{0,1}, \ldots, v_G \leq v_{0,G}$, is denoted by $\Pr\{\mathbf{v} \leq \mathbf{v}_0\}$. Clearly, $\Pr\{\mathbf{v} \leq \mathbf{v}_0\}$ is a function of $\mathbf{v}_0$. This function is called the *cumulative distribution function (cdf)* of the random vector $\mathbf{v}$ (i.e. of the random variables $v_1, \ldots, v_G$) and is defined by:

$$P_{\mathbf{v}}(\mathbf{v}_0) = P_{v_1,\ldots,v_G}(v_{0,1}, \ldots, v_{0,G}) \triangleq \Pr\{\mathbf{v} \leq \mathbf{v}_0\} \qquad \forall\ -\infty \leq \mathbf{v}_0 \leq \infty\,, \tag{B.2.1}$$

where $-\infty \leq \mathbf{v}_0 \leq \infty$ means that each component of $\mathbf{v}_0$ can range from $-\infty$ to ∞. Note that the cdf for a random vector describes the *joint* distribution of its components. Therefore, it is also called the *joint cumulative distribution function (jcdf)* of the random variables

$v_1, \ldots, v_G$. The notation $\mathbf{v}$ in the subscript position of $P_{\mathbf{v}}(\mathbf{v}_0)$ denotes the considered random vector, whereas the variable $\mathbf{v}_0$ represents the argument of $P_{\mathbf{v}}(\mathbf{v}_0)$ representing a specific value of $\mathbf{v}$. Usually, $\mathbf{v}_0$ will be replaced by $\mathbf{v}$, i.e. $P_{\mathbf{v}}(\mathbf{v})$ is considered. In such a case, the difference between the random vector $\mathbf{v}$ in the subscript position on the one hand, and the argument $\mathbf{v}$ between parentheses on the other hand should be kept in mind. The multivariate cdf in (B.2.1) has properties similar to that of the univariate cdf in (B.1.1). For continuous random variables, it is a nonnegative nondecreasing continuous function of each component whose values lie in the interval $[0, 1]$, i.e. $0 \leq P_{\mathbf{v}}(\mathbf{v}_0) \leq 1$. From the definition, it also follows directly that $P_{\mathbf{v}}(-\boldsymbol{\infty}) = 0$ and $P_{\mathbf{v}}(\boldsymbol{\infty}) = 1$.

The *probability density function (pdf)* of $\mathbf{v}$ (i.e. of the random variables $v_1, \ldots, v_G$) is defined as the derivative of the cdf w.r.t. all vector components:

$$p_{\mathbf{v}}(\mathbf{v}_0) \triangleq \left. \frac{\partial^{(G)} P_{\mathbf{v}}(\mathbf{v})}{\partial v_1 \cdots \partial v_G} \right|_{\mathbf{v}=\mathbf{v}_0} = \left. \frac{\partial}{\partial v_1} \cdots \frac{\partial}{\partial v_G} P_{\mathbf{v}}(\mathbf{v}) \right|_{\mathbf{v}=\mathbf{v}_0} . \tag{B.2.2}$$

This function is also called the *joint probability density function (jpdf)* or *multivariate probability density function* of the random variables $v_1, \ldots, v_G$. From (B.2.2), it follows that $P_{\mathbf{v}}(\mathbf{v}_0)$ can be re-obtained from $p_{\mathbf{v}}(\mathbf{v}_0)$ by integration:

$$P_{\mathbf{v}}(\mathbf{v}_0) = \int_{-\boldsymbol{\infty}}^{\mathbf{v}_0} p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \triangleq \int_{-\infty}^{v_{0,1}} \cdots \int_{-\infty}^{v_{0,G}} p_{\mathbf{v}}(\mathbf{v})\, dv_G \cdots dv_1 . \tag{B.2.3}$$

Note that $\int_{-\infty}^{\infty} p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} = 1$ because $P_{\mathbf{v}}(\boldsymbol{\infty}) = 1$. The probability density function $p_{\mathbf{v}}(\mathbf{v}_0)$ can be interpreted in a similar way as $p_v(v_0)$ for a (scalar) random variable. In order to do so, consider the event $\mathbf{v}_0 \leq \mathbf{v} \leq \mathbf{v}_0 + \Delta\mathbf{v}$ defined by:

$$\mathbf{v}_0 \leq \mathbf{v} \leq \mathbf{v}_0 + \Delta\mathbf{v} \qquad \equiv \qquad v_{0,1} \leq v_1 \leq v_{0,1} + \Delta v_1, \ldots, v_{0,G} \leq v_G \leq v_{0,G} + \Delta v_G ,$$

where $\Delta v_1, \ldots, \Delta v_G$ are small increments of the components of $\mathbf{v}$. From (B.2.1) and (B.2.3) it then follows that:

$$\Pr\{\mathbf{v}_0 \leq \mathbf{v} \leq \mathbf{v}_0 + \Delta\mathbf{v}\} = \int_{\mathbf{v}_0}^{\mathbf{v}_0+\Delta\mathbf{v}} p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \approx p_{\mathbf{v}}(\mathbf{v}_0) \cdot \Delta v_1 \cdots \Delta v_G . \tag{B.2.4}$$

Hence, the probability that the random vector $\mathbf{v}$ falls in a small region $\mathbf{v}_0 \leq \mathbf{v} \leq \mathbf{v}_0 + \Delta\mathbf{v}$ around $\mathbf{v}_0$ equals (approximately) $p_{\mathbf{v}}(\mathbf{v}_0) \cdot \Delta v_1 \cdots \Delta v_G$, i.e. it is proportional to the pdf evaluated at $\mathbf{v}_0$. It also follows from (B.2.4) that the probability that $\mathbf{v}$ is in any arbitrary region $\mathcal{V} \in \mathbb{R}_G$ can be obtained by integrating the pdf over that region.

The univariate pdf of an individual component of $\mathbf{v}$ can be obtained by integrating out all other components. For example, the pdf of the i-th component v_i can be computed as:

$$p_{v_i}(v_i) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p_{\mathbf{v}}(\mathbf{v})\, dv_G \cdots dv_{i+1}\, dv_{i-1} \cdots dv_1 , \tag{B.2.5}$$

i.e. the integration is performed over all variables except v_i. The result obtained in this way is called a *marginal probability density function (mpdf)*. Similarly, the joint probability density function of a subset of the components of $\mathbf{v}$ is obtained by integrating out the components of $\mathbf{v}$ that are not in the selected subset. For example, the jpdf of v_1 and v_2 can be computed as

follows:

$$p_{v_1,v_2}(v_1,v_2) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p_{\mathbf{v}}(\mathbf{v})\, dv_G \cdots dv_3\,. \tag{B.2.6}$$

A joint probability density function obtained in this way is called a *marginal joint probability density function (mjpdf)* or simply marginal probability density function. The same concepts can be defined for groups of random variables in a completely similar manner. Suppose that different groups or collections $\mathcal{G}_1, \ldots, \mathcal{G}_P$ of random variables are given. For convenience of notation the variables in each group are collected in a random vector. The vector $\mathbf{v}_i$ denotes the random vector containing the variables from the i-th group (note that the different vectors may have different lengths). For example, the jpdf of the random vectors $\mathbf{v}_1, \ldots, \mathbf{v}_P$ can be written as:

$$p_{\mathbf{v}_1,\ldots,\mathbf{v}_P}\big(\mathbf{v}_1,\ldots,\mathbf{v}_P\big)\,. \tag{B.2.7}$$

Likewise, the marginal probability density function of the i-th vector can be written as:

$$p_{\mathbf{v}_i}(\mathbf{v}_i) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p_{\mathbf{v}_1,\ldots,\mathbf{v}_P}\big(\mathbf{v}_1,\ldots,\mathbf{v}_P\big)\, d\mathbf{v}_P \cdots d\mathbf{v}_{i+1}\, d\mathbf{v}_{i-1} \cdots d\mathbf{v}_1\,. \tag{B.2.8}$$

The generalization to other concepts, such as the joint cumulative distribution function and marginal joint probability density functions, etcetera, is straightforward and left to the reader.

B.2.2 Conditional probability density functions

An important type of probability density function is the so-called *conditional probability density function (cpdf)*. Consider two sets of random variables, say $v_1, \ldots, v_G$ and $w_1, \ldots, w_P$, or their corresponding random vectors $\mathbf{v}$ and $\mathbf{w}$. Furthermore, assume that the joint density:

$$p_{\mathbf{v},\mathbf{w}}(\mathbf{v},\mathbf{w}) \triangleq p_{v_1,\ldots,v_G,w_1,\ldots,w_P}(v_1,\ldots,v_G,w_1,\ldots,w_P)\,,$$

the (joint) marginal densities:

$$p_{\mathbf{v}}(\mathbf{v}) \triangleq p_{v_1,\ldots,v_G}(v_1,\ldots,v_G) = \int_{-\infty}^{\infty} p_{\mathbf{v},\mathbf{w}}(\mathbf{v},\mathbf{w})\, d\mathbf{w}\,,$$

and:

$$p_{\mathbf{w}}(\mathbf{w}) \triangleq p_{w_1,\ldots,w_P}(w_1,\ldots,w_P) = \int_{-\infty}^{\infty} p_{\mathbf{v},\mathbf{w}}(\mathbf{v},\mathbf{w})\, d\mathbf{v}$$

exist. Then, the *conditional probability density function (cpdf) of* $\mathbf{v}$ *given* $\mathbf{w}$ is defined as:

$$p_{\mathbf{v}|\mathbf{w}}(\mathbf{v}|\mathbf{w}) \triangleq \frac{p_{\mathbf{v},\mathbf{w}}(\mathbf{v},\mathbf{w})}{p_{\mathbf{w}}(\mathbf{w})}\,. \tag{B.2.9}$$

This formula can be interpreted as follows. Assuming that the random vector $\mathbf{w}$ falls in a small region $\mathbf{w}_0 \leq \mathbf{w} \leq \mathbf{w}_0 + \Delta\mathbf{w}$ around $\mathbf{w}_0$, the probability that the random vector $\mathbf{v}$ falls in a small region $\mathbf{v}_0 \leq \mathbf{v} \leq \mathbf{v}_0 + \Delta\mathbf{v}$ around $\mathbf{v}_0$ equals (approximately) $p_{\mathbf{v}|\mathbf{w}}(\mathbf{v}_0|\mathbf{w}_0) \cdot \Delta v_1 \cdots \Delta v_G$, i.e. it is proportional to $p_{\mathbf{v}|\mathbf{w}}(\mathbf{v}_0|\mathbf{w}_0)$. As an example of (B.2.9), in case $v_1 = x_1, w_1 = x_2, w_2 = x_3$ (i.e. $G = 1$, $P = 2$) Eq. (B.2.9) becomes:

$$p_{x_1|x_2,x_3}(x_1|x_2,x_3) \triangleq \frac{p_{x_1,x_2,x_3}(x_1,x_2,x_3)}{p_{x_2,x_3}(x_2,x_3)}\,.$$

Note that from (B.2.9) it follows that:

$$\int_{-\infty}^{\infty} p_{\mathbf{v}|\mathbf{w}}(\mathbf{v}|\mathbf{w})\, d\mathbf{v} = 1\,.$$

This emphasizes the fact that $p_{\mathbf{v}|\mathbf{w}}(\mathbf{v}|\mathbf{w})$ is a true (joint) probability density function for $\mathbf{v}$ that depends on $\mathbf{w}$ as if it were a parameter. The conditioning quantity $\mathbf{w}$ can be thought of as a nonrandom parameter vector, even though it is actually a random vector.

A number of useful statistical quantities can be defined using the cpdf. For example, conditional expectations, moments and cumulants are defined similarly to the conventional way, with the only difference that the pdf's in the integrals are conditional. For more information on, and properties of, conditional densities, see [122].

B.2.3 Statistical independence

A very important concept that constitutes the foundation of many signal processing problems is *statistical independence*. The random variables $v_1, \ldots, v_G$ are (mutually) statistically independent if their joint pdf factorizes into marginal pdf's:

$$p_{v_1,\ldots,v_G}\big(v_1, \ldots, v_G\big) = \prod_{m=1}^{G} p_{v_m}(v_m)\,. \tag{B.2.10}$$

This means that knowing something about one of the variables does not provide any information about the others. For example, this is the case when the variables originate from quite different physical processes. This can also be stated as follows:

$$p_{v_i|v_1,\ldots,v_{i-1},v_{i+1},\ldots,v_G}\big(v_i|v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_G\big) = p_{v_i}\big(v_i\big) \tag{B.2.11}$$

because v_i does not depend in any way on $v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_G$. Independent random variables satisfy a basic property that can be expressed in terms of expectations as follows (multivariate expectation is defined in Section B.2.6.1).

Theorem B.2.1. Nonlinear joint expectations and statistical independence. *Let* $\mathbf{v}$ *be a length-*G *random vector with statistically independent components and let* $(i_1, \ldots, i_l)$ *be a length-*l *tuple of indices. Consider a subset* $\{v_{i_1}, \ldots, v_{i_l}\}$ *of* l *components of* $\mathbf{v}$*, i.e. each index* $1 \leq i_k \leq G$ *with* $1 \leq k \leq l$ *indexes a component of* $\mathbf{v}$*. In addition, let* $g_1(\cdot), \ldots, g_P(\cdot)$ *be a set of* $P \leq G$ *smooth functions from* $\mathbb{R}$ *to* $\mathbb{R}$ *whose expectations exist. Now, if the indices* $i_1, \ldots, i_l$ *are all different from each other, then:*

$$E\left\{\prod_{m=1}^{P} g_m\big(v_{i_m}\big)\right\} = \prod_{m=1}^{P} E\left\{g_m\big(v_{i_m}\big)\right\}\,. \tag{B.2.12}$$

Proof. The proof easily follows from (B.2.10) and the definition of the multivariate expectation operator $E\{\cdot\}$ in (B.2.16). □

The same concepts can be defined for groups of random variables in a completely similar manner. Suppose that different groups or collections $\mathcal{G}_1, \ldots, \mathcal{G}_P$ of random variables are given. For convenience of notation the variables in each group are collected in a random vector. The vector $\mathbf{v}_i$ denotes the random vector containing the variables from the i-th group

(note that the different vectors may have different lengths). Now, the groups are said to be statistically independent of each other if:

$$p_{\mathbf{v}_1,\ldots,\mathbf{v}_P}\big(\mathbf{v}_1,\ldots,\mathbf{v}_P\big) = \prod_{m=1}^{P} p_{\mathbf{v}_m}\big(\mathbf{v}_m\big)\,. \tag{B.2.13}$$

The generalizations of (B.2.11) and (B.2.13) to groups of random variables are obvious.

B.2.4 The Central Limit Theorem

A theorem from statistics and probability theory that is particularly important for blind signal processing problems such as Independent Component Analysis and Instantaneous Blind Signal Separation is the so-called *Central Limit Theorem (CLT)*:

Theorem B.2.2. Central Limit Theorem.
Let $\{v_k\}_{1\leq k\leq n}$ be a sequence of independent and identically distributed (i.i.d.) random variables, each having finite mean μ and finite variance $(\sigma)^2$. In addition, let S_n denote the partial sum defined by:

$$S_n \triangleq \sum_{k=1}^{n} v_k \qquad \forall\, n \in \mathbb{N}\,.$$

Then:

$$\lim_{n\to\infty} \frac{S_n - n\mu}{\sigma\sqrt{n}} \;\to\; \boldsymbol{\mathcal{N}}(0,1)\,,$$

where $\boldsymbol{\mathcal{N}}(0,1)$ denotes the Gaussian distribution with zero mean and unit variance.

This means that the distribution of the random variable $\frac{S_n-n\mu}{\sigma\sqrt{n}}$ converges to the standardized Gaussian distribution when $n\to\infty$. In other words, the sum of several independent random variables has a distribution that is closer to Gaussian than any of the individual variables. The CLT generalizes readily to statistically independent and identically distributed random vectors. Also, several (stronger) different forms of the theorem exist, where the assumptions on the statistical independence and identical distributions are weaker.

B.2.5 Multivariate characteristic functions

As in the scalar case, two characteristic functions are defined that will prove useful for the definition or derivation of moments and cumulants in the multivariate case. We will call them *First Joint Characteristic Function (FJCF)* and *Second Joint Characteristic Function (SJCF)* in order to distinguish them from their univariate counterparts. The FJCF $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ of the length-G random vector $\mathbf{v}$ is defined as the (continuous) multi-dimensional Fourier transform of the pdf $p_{\mathbf{v}}(\mathbf{v})$[‡]:

$$\begin{aligned}\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) &\triangleq E\{\exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})\} = \int_{-\boldsymbol{\infty}}^{\boldsymbol{\infty}} p_{\mathbf{v}}(\mathbf{v})\,\exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})\,d\mathbf{v} \\ &= \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} p_{v_1,\ldots,v_G}\big(v_1,\ldots,v_G\big)\,\exp\Big(\jmath\left(\omega^1 v_1+\cdots+\omega^G v_G\right)\Big)\,dv_G\cdots dv_1\,. \end{aligned}\tag{B.2.14}$$

[‡]See Section B.2.6.1 for the definition of the multivariate expectation operator $E\{\cdot\}$.

where $\tilde{\boldsymbol{\omega}} \triangleq \begin{bmatrix} \omega^1 & \cdots & \omega^G \end{bmatrix}$ is a length-G real-valued row vector of the transformed variables corresponding to the components of $\mathbf{v}$. As will be shown in the next section, the FJCF $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ can be used to derive the joint moments of the components of $\mathbf{v}$. The SJCF $\psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ of the random vector $\mathbf{v}$ is defined as the natural logarithm of the FJCF of $\mathbf{v}$:

$$\psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) \triangleq \ln(\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})) \,. \tag{B.2.15}$$

The SJCF $\psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ can be used to define the *(joint) cumulants* of the components of $\mathbf{v}$, as will also be shown in Section B.2.6.5.

B.2.6 Multivariate expectations, (central) moments and cumulants

The definitions of expectations, multivariate moments and multivariate cumulants are straightforward generalizations of the corresponding univariate definitions. Since usually a large number of indices is required to denote joint (multivariate) moments, cumulants, etcetera, a compact and convenient notation for tuples of indices will be employed (see also Appendix A). This notation is heavily used throughout the thesis.

B.2.6.1 Expectations

Let $g(\mathbf{v})$ denote a function of the random vector $\mathbf{v}$. The (mathematical) *expectation* of $g(\mathbf{v})$ is denoted by $E\{g(\mathbf{v})\}$ and defined by:

$$E\{g(\mathbf{v})\} \triangleq \int_{-\infty}^{\infty} g(\mathbf{v})\, p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \,. \tag{B.2.16}$$

Considering again expectation as an operator $E\{\cdot\}$, the following important properties of $E\{\cdot\}$, which hold irrespective of the nature of the particular random vector(s) under study, can be derived:

- *Linearity of Type I:* Let $g_1(\mathbf{v}), \ldots, g_P(\mathbf{v})$ be a set of P functions whose expectations exist and let $c^1, \ldots, c^P$ be a set of constant coefficients, then:

$$E\left\{\sum_{i=1}^{P} c^i g_i(\mathbf{v})\right\} = \sum_{i=1}^{P} c^i E\left\{g_i(\mathbf{v})\right\} \,. \tag{B.2.17}$$

  Note that the same statement holds when function-valued vectors $\mathbf{g}_1(\mathbf{v}), \ldots, \mathbf{g}_P(\mathbf{v})$ are considered instead of the scalar functions $g_1(\mathbf{v}), \ldots, g_P(\mathbf{v})$.

- *Linearity of Type II:* Let $\mathbf{v}_1, \ldots, \mathbf{v}_Q$ be a set of Q random vectors whose expectations exist and let $c^1, \ldots, c^Q$ be a set of constant coefficients, then:

$$E\left\{\sum_{i=1}^{Q} c^i \mathbf{v}_i\right\} = \sum_{i=1}^{Q} c^i E\left\{\mathbf{v}_i\right\} \,. \tag{B.2.18}$$

- *Transformation invariance:* Let $\mathbf{y} = \mathbf{g}(\mathbf{v})$ be a function-valued vector of $\mathbf{v}$, then:

$$\int_{-\infty}^{\infty} \mathbf{y}\, p_{\mathbf{y}}(\mathbf{y})\, d\mathbf{y} = \int_{-\infty}^{\infty} \mathbf{g}(\mathbf{v})\, p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \,, \tag{B.2.19}$$

  i.e. $E\{\mathbf{y}\} = E\{\mathbf{g}(\mathbf{v})\}$, even though the integrations are carried out over different probability density functions.

The same remarks as made for the univariate case in Section B.1.3.1 also apply to properties (B.2.17)-(B.2.19). The types of linearity defined in (B.2.17) and (B.2.18) can be combined into one formulation as follows. Let $g_1(\cdot), \ldots, g_P(\cdot)$ be a set of P functions, let $\mathbf{v}_1, \ldots, \mathbf{v}_Q$ be a set of Q random vectors, and let $\{c^{ij}\}^{1 \leq i \leq P, 1 \leq j \leq Q}$ be a set of constant coefficients. Assuming that the expectation of $g_i(\mathbf{v}_j)$ exists for each possible combination of i and j, the following holds:

$$E\left\{\sum_{i=1}^{P}\sum_{j=1}^{Q} c^{ij} g_i(\mathbf{v}_j)\right\} = \sum_{i=1}^{P}\sum_{j=1}^{Q} c^{ij} E\left\{g_i(\mathbf{v}_j)\right\}\,, \tag{B.2.20}$$

where $E\left\{g_i(\mathbf{v}_j)\right\} = E_{\mathbf{v}_j}\{g_i(\mathbf{v}_j)\}$. Again, the same statement holds when function-valued vectors $\mathbf{g}_1(\mathbf{v}), \ldots, \mathbf{g}_P(\mathbf{v})$ are considered instead of the scalar functions $g_1(\mathbf{v}), \ldots, g_P(\mathbf{v})$.

B.2.6.2 Joint moments

In this section we will define the joint moment of a set of random variables in two different ways, as in Section B.1.3.2. The first involves the conventional definition and the second involves the Taylor expansion of the FJCF. Along the way, a convenient notational device is introduced. Let $\mathbf{v}$ be a length-G random vector with jpdf $p_{\mathbf{v}}(\mathbf{v})$ and let $(i_1, \ldots, i_l)$ be a length-l tuple of indices. Consider a subset $\{v_{i_1}, \ldots, v_{i_l}\}$ of l components of $\mathbf{v}$. Then, the l-th order joint moment $r^{v,l}_{i_1 \cdots i_l}$ of the random variables $v_{i_1}, \ldots, v_{i_l}$ is defined as the expectation of their product. Hence, substituting $g(\mathbf{v}) = v_{i_1} \cdots v_{i_l}$ into (B.2.16) gives for all $l \in \mathbb{N}$:

$$\begin{aligned} r^{v,l}_{i_1 \cdots i_l} \equiv r^{v}_{i_1 \cdots i_l} &\triangleq E\{v_{i_1} \cdots v_{i_l}\} \triangleq \int_{-\infty}^{\infty} v_{i_1} \cdots v_{i_l}\, p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} v_{i_1} \cdots v_{i_l}\, p_{v_1,\ldots,v_G}\big(v_1, \ldots, v_G\big)\, dv_G \cdots dv_1 \quad \forall\, 1 \leq i_1, \ldots, i_l \leq G\,. \end{aligned} \tag{B.2.21}$$

This joint moment is said to be of *tuple-order* $(i_1, \ldots, i_l)$. The *total order* of the moment is l because there are l indices and the expectation of a product of l variables is computed. Since the total order equals the number of subscript indices, it is usually omitted from the superscript position. Eq. (B.2.21) involves the complete jpdf of all components of $\mathbf{v}$. It can also be evaluated with the marginal joint probability density function in which only the different components selected by the tuple $(i_1, \ldots, i_l)$ are involved. As an example, consider the length-3 random vector $\mathbf{v}$ and $l = 2$. All possible different tuples of length two are listed in Table B.3, along with the product of variables $v_{i_1} v_{i_2}$. According to (B.2.21), the (i_1, i_2)-th moment $r^{v}_{i_1 i_2}$ can be computed by:

$$r^{v}_{i_1 i_2} \triangleq E\{v_{i_1} v_{i_2}\} \triangleq \int_{-\infty}^{\infty} v_{i_1} v_{i_2}\, p_{v_1,v_2,v_3}(v_1, v_2, v_3)\, dv_1 dv_2 dv_3 \qquad \forall\, 1 \leq i_1, i_2 \leq 3\,,$$

which involves the 'full' jpdf $p_{v_1,v_2,v_3}(v_1, v_2, v_3)$. However, in case the two selected components of $\mathbf{v}$ are different, i.e. when $i_1 \neq i_2$, the mjpdf $p_{v_{i_1},v_{i_2}}(v_{i_1}, v_{i_2})$ can be determined and used for computing $r^{v}_{i_1 i_2}$:

$$r^{v}_{i_1 i_2} = \int_{-\infty}^{\infty} v_{i_1} v_{i_2}\, p_{v_{i_1},v_{i_2}}(v_{i_1}, v_{i_2})\, dv_{i_1} dv_{i_2} \qquad \forall\, 1 \leq i_1 \neq i_2 \leq 3\,.$$

Table B.3: *Example of joint moment definition for $G = 3$ and $l = 2$.*

(i_1, i_2)	$v_{i_1} v_{i_2}$
$(1,1)$	$v_1 v_1 = (v_1)^2$
$(1,2)$	$v_1 v_2$
$(1,3)$	$v_1 v_2$
$(2,2)$	$v_2 v_2 = (v_2)^2$
$(2,3)$	$v_2 v_3$
$(3,3)$	$v_3 v_3 = (v_3)^2$

In case the two selected components of $\mathbf{v}$ are equal, i.e. when $i_1 = i_2 \triangleq i$, the computation can be simplified even more by determining the mpdf $p_{v_i}(v_i)$ and using the result for computing r^v_{ii}:

$$r^v_{ii} = \int_{-\infty}^{\infty} (v_i)^2 \, p_{v_i}(v_i) \, dv_i \qquad \forall \, 1 \leq i \leq 3 \, .$$

When all indices in a tuple $(i_1, \ldots, i_l)$ are equal (such a tuple is also written as $(i)_l$, as explained in Appendix A), the corresponding moment is called *auto-moment*. In the other case, the corresponding moment is called *cross-moment*. The term *joint moment* can mean either auto- or cross-moment. Using the notation explained in Section A.2, Eq. (B.2.21) can be written compactly as follows:

$$r^{v,l}_{\mathbf{i}_l} \equiv r^v_{\mathbf{i}_l} \triangleq E\{\check{v}_{\mathbf{i}_l}\} = \int_{-\infty}^{\infty} \check{v}_{\mathbf{i}_l} \, p_{\mathbf{v}}(\mathbf{v}) \, d\mathbf{v} \qquad \forall \, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},G} \, . \tag{B.2.22}$$

As in the univariate case, multivariate moments of order larger than two are usually called *higher order moments*.

Deriving the joint moments from the FJCF

The joint moments of all possible subsets of components of $\mathbf{v}$ can also be derived from the Taylor series expansion of the First Joint Characteristic Function $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ of $\mathbf{v}$ around $\tilde{\boldsymbol{\omega}} = \tilde{\mathbf{0}}$. We proceed in a similar way as in Section B.1.3.2. Assume that (higher order) partial differentiation under the integral signs in (B.2.14) is allowed and define the partial derivative $\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}})$ of tuple-order $\mathbf{i}_l$ (i.e. of total order l) as the higher order partial derivative of $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ w.r.t. $\omega^{i_1}, \ldots, \omega^{i_l}$:

$$\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}}) = \phi^{\mathbf{v},(i_1 \cdots i_l)}(\omega^1, \ldots, \omega^G) \triangleq \frac{\partial^{(l)} \phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})}{\partial \omega^{i_1} \cdots \partial \omega^{i_l}} \, . \tag{B.2.23}$$

Then, from (B.2.14) it follows that:

$$\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}}) = \int_{-\infty}^{\infty} \frac{\partial^{(l)} \exp(\jmath \tilde{\boldsymbol{\omega}} \mathbf{v})}{\partial \omega^{i_1} \cdots \partial \omega^{i_l}} \, p_{\mathbf{v}}(\mathbf{v}) \, d\mathbf{v} = (\jmath)^l \int_{-\infty}^{\infty} \check{v}_{\mathbf{i}_l} \exp(\jmath \tilde{\boldsymbol{\omega}} \mathbf{v}) \, p_{\mathbf{v}}(\mathbf{v}) \, d\mathbf{v} \, ,$$

where $\check{v}_{\mathbf{i}_l} \triangleq v_{i_1} \cdots v_{i_l}$. Evaluating $\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}})$ at $\tilde{\boldsymbol{\omega}} = \tilde{\mathbf{0}}$ under the assumption that all moments exist yields:

$$\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\mathbf{0}}) \;=\; \phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}})\Big|_{\tilde{\boldsymbol{\omega}}=\tilde{\mathbf{0}}} \;=\; (\jmath)^l \int_{-\boldsymbol{\infty}}^{\boldsymbol{\infty}} \check{v}_{\mathbf{i}_l}\, p_{\mathbf{v}}(\mathbf{v})\, d\mathbf{v} \;\overset{\text{(B.2.22)}}{=}\; (\jmath)^l\, E\{\check{v}_{\mathbf{i}_l}\} \;=\; (\jmath)^l\, r^v_{\mathbf{i}_l}\,. \tag{B.2.24}$$

Hence, whenever all moments are finite the multivariate Taylor series expansion of the FJCF around the origin is given by:

$$\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = \sum_{l=0}^{\infty} \sum_{\mathbf{i}_l \in \mathcal{I}^l_{\mathfrak{t},G}} \frac{\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\mathbf{0}})}{l!}\, \check{\omega}^{\mathbf{i}_l} = \sum_{l=0}^{\infty} \sum_{\mathbf{i}_l \in \mathcal{I}^l_{\mathfrak{t},G}} r^v_{\mathbf{i}_l}\, \frac{(\jmath)^l \check{\omega}^{\mathbf{i}_l}}{l!}\,, \tag{B.2.25}$$

where $\check{\omega}^{\mathbf{i}_l} \triangleq \omega^{i_1}\cdots\omega^{i_l}$ and the summation $\sum_{\mathbf{i}_l \in \mathcal{I}^l_{\mathfrak{t},G}}$ denotes the l-fold summation $\sum_{i_1=1}^{G}\cdots\sum_{i_l=1}^{G}$. Our notation results in a formulation that is different from the conventional one, but is quite natural. From (B.2.25) it can be concluded that the Taylor coefficient of tuple-order $\mathbf{i}_l = (i_1,\ldots,i_l)$ (and total order l), i.e. the coefficient of the term $\frac{(\jmath)^l\check{\omega}^{\mathbf{i}_l}}{l!} = \frac{(\jmath)^l\omega^{i_1}\cdots\omega^{i_l}}{l!} = \frac{(\jmath\,\omega^{i_1})\cdots(\jmath\,\omega^{i_l})}{l!}$, is the joint moment $r^v_{i_1\cdots i_l}$ of tuple-order $\mathbf{i}_l$ of the random variables $v_{i_1},\ldots,v_{i_l}$. For this reason, the First Joint Characteristic Function is also called *(Joint) Moment Generating Function*. Similarly to the univariate case, the result in (B.2.25) can also be derived in the following insightful manner. Firstly, from (B.2.16) it is noted that $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = E\{\exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})\}$ by taking $g(\mathbf{v}) = \exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})$. Then, using the linearity property of the expectation operator and the standard Taylor expansion of the exponential function, it follows that:

$$\begin{aligned}
\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = E\{\exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})\} &= E\left\{\sum_{l=0}^{\infty}\frac{(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})^l}{l!}\right\} = E\left\{\sum_{l=0}^{\infty}\frac{(\jmath)^l}{l!}\left(\sum_{i=1}^{G}\omega^i v_i\right)^l\right\}\\
&= E\left\{\sum_{l=0}^{\infty}\frac{(\jmath)^l}{l!}\left(\sum_{i_1=1}^{G}\omega^{i_1}v_{i_1}\right)\cdots\left(\sum_{i_l=1}^{G}\omega^{i_l}v_{i_l}\right)\right\}\\
&= E\left\{\sum_{l=0}^{\infty}\frac{(\jmath)^l}{l!}\sum_{i_1=1}^{G}\cdots\sum_{i_l=1}^{G}\omega^{i_1}\cdots\omega^{i_l}\,v_{i_1}\cdots v_{i_l}\right\}\\
&= \sum_{l=0}^{\infty}\sum_{i_1=1}^{G}\cdots\sum_{i_l=1}^{G} r^v_{i_1\cdots i_l}\,\frac{(\jmath\,\omega^{i_1})\cdots(\jmath\,\omega^{i_l})}{l!}\,.
\end{aligned}$$

Hence, as in (B.2.25) the joint moment $r^v_{i_1\cdots i_l}$ is the coefficient of the term $\frac{(\jmath\,\omega^{i_1})\cdots(\jmath\,\omega^{i_l})}{l!}$ in the Taylor series expansion of $\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$. With the compact notation explained in Appendix A the equation above can be written concisely as:

$$\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = E\{\exp(\jmath\tilde{\boldsymbol{\omega}}\mathbf{v})\} = \sum_{l=0}^{\infty}\sum_{\mathbf{i}_l\in\mathcal{I}^l_{\mathfrak{t},G}} r^v_{\mathbf{i}_l}\,\frac{(\jmath)^l\check{\omega}^{\mathbf{i}_l}}{l!}\,.$$

From the development above, in particular from (B.2.24), it is clear that the joint moments can also be defined formally as follows:

$$r^v_{\mathbf{i}_l} = E\big\{\check{v}_{\mathbf{i}_l}\big\} \triangleq \frac{\phi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\mathbf{0}})}{(\jmath)^l} = (-\jmath)^l\,\frac{\partial^{(l)}\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})}{\partial\omega^{i_1}\cdots\partial\omega^{i_l}}\bigg|_{\tilde{\boldsymbol{\omega}}=\tilde{\mathbf{0}}} \qquad \forall\,\mathbf{i}_l\in\mathcal{I}^l_{\mathfrak{t},G},\;\forall\,l\in\mathbb{N}\,. \tag{B.2.26}$$

The moment function $\mathrm{mom}(\cdot,\ldots,\cdot)$

The computation of the (joint) moment $r^v_{i_1\cdots i_l} \equiv r^{v,l}_{i_1\cdots i_l}$ of the random variables $v_{i_1},\ldots,v_{i_l}$ can be considered as computing a certain function of these variables, namely the expectation of their product. This function is denoted by $\mathrm{mom}(\cdot,\ldots,\cdot)$ and defined by:

$$\mathrm{mom}\big(v_{i_1},\ldots,v_{i_l}\big) \triangleq E\{v_{i_1}\cdots v_{i_l}\}\,. \tag{B.2.27}$$

Hence, the number of arguments of this function equals the total order l of the considered joint moment. As has been explained earlier, the result of this function is denoted by $r^v_{\mathbf{i}_l} = r^v_{i_1\cdots i_l}$. In fact, the notations $r^v_{i_1\cdots i_l}$ and $\mathrm{mom}\big(v_{i_1},\ldots,v_{i_l}\big)$ are completely equivalent. We state this notation explicitly for reference purposes:

$$\boxed{r^v_{\mathbf{i}_l} \equiv r^v_{i_1\cdots i_l} \equiv r^{v,l}_{i_1\cdots i_l} \triangleq \mathrm{mom}\big(v_{i_1},\ldots,v_{i_l}\big) = E\{v_{i_1}\cdots v_{i_l}\} \quad \forall\, \mathbf{i}_l \in \mathcal{I}^l_{\mathrm{t},G}\,.} \tag{B.2.28}$$

For example, in the case that $l=3$ and $\mathbf{i}_3=(3,1,4)$ we have $r^v_{\mathbf{i}_3} = r^v_{314} = \mathrm{mom}(v_3,v_1,v_4)$ $= E\{v_3v_1v_4\}$ $\Big(= E\{v_1v_3v_4\} = r^v_{134}\Big)$. If all indices in $\mathbf{i}_l$ are equal, the corresponding moment function is called *auto-moment function*. In the other case, the corresponding moment function is called *cross-moment function*. The term *joint moment function* can mean either an auto- or a cross-moment function.

B.2.6.3 Properties of joint moments

The following general properties of joint moments, which are very important for manipulating general expressions without actually computing (multivariate) integrals, can be derived directly from the definition of the moment function:

MP1: For all constant coefficients $c^1,\ldots,c^l \in \mathbb{R}$:

$$\mathrm{mom}\left(c^1w_1,\ldots,c^lw_l\right) = c^1\cdots c^l\,\mathrm{mom}\left(w_1,\ldots,w_l\right). \tag{B.2.29}$$

MP2: Moments are symmetric in their arguments:

$$\mathrm{mom}\left(w_{i_1},\ldots,w_{i_l}\right) = \mathrm{mom}\left(w_1,\ldots,w_l\right) \tag{B.2.30}$$

for any permutation $(i_1,\ldots,i_l)$ of $(1,\ldots,l)$.

MP3: For statistically independent variables the moment function factorizes:

$$\mathrm{mom}\left(w_1,\ldots,w_l\right) = \prod_{m=1}^{l}\mathrm{mom}\left(w_m\right). \tag{B.2.31}$$

B.2.6.4 Central joint moments

In this section we provide the extension of the definition of univariate central moments to *joint* or *multivariate central moments*. Let $\mathbf{v}$ be a length-G random vector with jpdf $p_{\mathbf{v}}(\mathbf{v})$ and let $(i_1,\ldots,i_l)$ be a length-l tuple of indices. Consider a subset $\{v_{i_1},\ldots,v_{i_l}\}$ of l components of $\mathbf{v}$. Hence, each index $1\le i_k\le G$ indexes a component of $\mathbf{v}$. Then, the *joint central moment* $\gamma^v_{i_1\cdots i_l}$ of tuple-order $(i_1,\ldots,i_l)$ and total order l of the random variables $v_{i_1},\ldots,v_{i_l}$ is defined as follows for all $l\in\mathbb{N}$:

$$\gamma^v_{i_1\cdots i_l} \;\triangleq\; E\Big\{\big(v_{i_1} - E\{v_{i_1}\}\big)\cdots\big(v_{i_l} - E\{v_{i_l}\}\big)\Big\} \qquad \forall\;\; 1 \;\le\; i_1,\ldots,i_l \;\le\; G\,.$$

As in the univariate case, central joint moments are equal to joint moments for zero mean random variables.

B.2.6.5 Joint cumulants

Joint or *multivariate cumulants* are defined analogously to the univariate case, namely in terms of the higher order partial derivatives of the *Second Joint Characteristic Function (SJCF)*. The *joint cumulant* $\kappa_{\mathbf{i}_l}^{v,l} \equiv \kappa_{\mathbf{i}_l}^{v} = \kappa_{i_1 \cdots i_l}^{v}$ of tuple-order $\mathbf{i}_l = (i_1, \ldots, i_l)$ and total order l of the random variables $v_{i_1}, \ldots, v_{i_l}$ is defined as the coefficient of the term $\frac{(\jmath)^l \check{\boldsymbol{\omega}}^{\mathbf{i}_l}}{l!}$ in the Taylor series expansion of the SJCF $\psi^{\mathbf{v}}(\check{\boldsymbol{\omega}})$ defined in (B.2.15) around the origin:

$$\psi^{\mathbf{v}}(\check{\boldsymbol{\omega}}) \triangleq \ln(\phi^{\mathbf{v}}(\check{\boldsymbol{\omega}})) = \sum_{l=0}^{\infty} \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^{l}} \frac{\psi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\mathbf{0}})}{l!}\, \check{\omega}^{\mathbf{i}_l} \triangleq \sum_{l=0}^{\infty} \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^{l}} \kappa_{\mathbf{i}_l}^{v}\, \frac{(\jmath)^l \check{\omega}^{\mathbf{i}_l}}{l!}\,, \tag{B.2.32}$$

where $\psi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\boldsymbol{\omega}})$ is the partial derivative of tuple-order $\mathbf{i}_l$ of $\psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})$ that is defined similarly to (B.2.23). Compare the Taylor expansion in (B.2.32) with (B.2.25). From (B.2.32) it follows that the cumulant $\kappa_{\mathbf{i}_l}^{v}$ is defined as (note the similarity to (B.2.26)):

$$\kappa_{\mathbf{i}_l}^{v} \triangleq \frac{\psi^{\mathbf{v},(\mathbf{i}_l)}(\tilde{\mathbf{0}})}{(\jmath)^l} = (-\jmath)^l \left. \frac{\partial^{(l)} \psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}})}{\partial \omega^{i_1} \cdots \partial \omega^{i_l}} \right|_{\tilde{\boldsymbol{\omega}} = \tilde{\mathbf{0}}} \quad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^{l}\,, \quad \forall\, l \in \mathbb{N}\,. \tag{B.2.33}$$

For this reason, the Second Joint Characteristic Function is also called *(Joint) Cumulant Generating Function*. When all indices in a tuple $\mathbf{i}_l$ are equal, the corresponding cumulant is called *auto-cumulant*. In the other case, the corresponding cumulant is called *cross-cumulant*. The term *joint cumulant* can mean either auto- or cross-cumulant. As in the univariate case, multivariate cumulants of order larger than two are usually called *higher order cumulants*.

The cumulant function $\mathrm{cum}(\cdot, \ldots, \cdot)$
Like the moment function discussed on page 438, the computation of the (joint) cumulant $\kappa_{i_1 \cdots i_l}^{v} \equiv \kappa_{\mathbf{i}_l}^{v,l}$ of the random variables $v_{i_1}, \ldots, v_{i_l}$ can be considered as computing a certain function of these variables. This function is denoted by $\mathrm{cum}(\cdot, \ldots, \cdot)$ and defined by (B.2.33). Like the moment function in (B.2.28), the number of arguments of this function equals the total order l of the considered joint cumulant. Also, the notations $\kappa_{\mathbf{i}_l}^{v} = \kappa_{i_1 \cdots i_l}^{v}$ and $\mathrm{cum}\big(v_{i_1}, \ldots, v_{i_l}\big)$ are completely equivalent. We state this notation explicitly for reference purposes:

$$\boxed{\kappa_{\mathbf{i}_l}^{v} \equiv \kappa_{i_1 \cdots i_l}^{v} \equiv \kappa_{i_1 \cdots i_l}^{v,l} \triangleq \mathrm{cum}\big(v_{i_1}, \ldots, v_{i_l}\big) \quad \forall\, \mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^{l}\,.} \tag{B.2.34}$$

Several important properties of the cumulant functions are listed in Section B.2.6.7. Similarly to the previous paragraph, when all indices in $\mathbf{i}_l$ are equal the corresponding cumulant function is called *auto-cumulant function*. In the other case, the corresponding cumulant function is called *cross-cumulant function*. The term *joint cumulant function* can mean either an auto- or a cross-cumulant function.

B.2.6.6 Expressing multivariate cumulants in multivariate moments

As in the univariate case, it is possible to express multivariate cumulants in terms of sums of products of multivariate moments, and vice versa. We first consider the joint cumulant

$\kappa^w_{1\cdots l} = \mathrm{cum}(w_1,\ldots,w_l)$ of the l components $w_1,\ldots,w_l$ of the random vector $\mathbf{w} \triangleq \big[w_1 \ \cdots \ w_l\big]^T$ and express this cumulant in terms of the joint moments up to total order l. Then, the joint cumulant of the components $v_{i_1},\ldots,v_{i_l}$ of $\mathbf{v}$ can be obtained by substituting $w_1 = v_{i_1},\ldots,w_l = v_{i_l}$ in the resulting expression. To express the joint cumulant $\kappa^w_{1\cdots l}$ in the joint moments up to total order l a formula that is very similar to (B.1.21), but slightly more general, is employed:

$$\kappa^w_{1\cdots l} \equiv \kappa^{w,l}_{1\cdots l} = \sum_{i=1}^{|\mathcal{P}|}(-1)^{|\mathcal{P}_i|-1}(|\mathcal{P}_i|-1)!\prod_{j=1}^{|\mathcal{P}_i|} r^w_{\mathcal{I}_i^j}\,. \tag{B.2.35}$$

The meaning of the symbols is exactly the same as in Section B.1.3.5. The only difference between (B.2.35) and (B.1.21) is that $r^{w,|\mathcal{I}_i^j|}$ has been replaced by $r^w_{\mathcal{I}_i^j} \equiv r^{w,|\mathcal{I}_i^j|}_{\mathcal{I}_i^j}$. Considering $\mathcal{I}_i^j$ as an index tuple, the meaning of the notation $r^w_{\mathcal{I}_i^j}$ is obvious from (B.2.22):

$$r^w_{\mathcal{I}_i^j} \equiv r^{w,|\mathcal{I}_i^j|}_{\mathcal{I}_i^j} \triangleq E\Bigg\{\prod_{m\in\mathcal{I}_i^j} w_m\Bigg\}.$$

Hence, $r^w_{\mathcal{I}_i^j}$ denotes the joint/multivariate moment of total order $|\mathcal{I}_i^j|$ of the components of $\mathbf{w}$ indexed by the elements of the set $\mathcal{I}_i^j$. In order to demonstrate the use of (B.2.35), we will now give examples for $l = 1, 2, 3$ and 4 respectively. The procedure and results should be compared to those in Section B.1.3.5.

Example for $l = 1$
For $l = 1$, the only possible partition for the set $\mathcal{I} = \{1\}$ is the set itself. Thus, $\mathcal{P} = \{\{1\}\}$, $|\mathcal{P}| = 1$, $\mathcal{P}_1 = \{(1)\}$, $|\mathcal{P}_1| = 1$, $\mathcal{I}_1^1 = (1)$ and $\big|\mathcal{I}_1^1\big| = 1$. Hence, (B.2.35) gives:

$$\kappa^w_1 \equiv \kappa^{w,1}_1 \triangleq \mathrm{cum}(w_1) = \sum_{i=1}^{1}(-1)^{|\mathcal{P}_i|-1}(|\mathcal{P}_i|-1)!\prod_{j=1}^{|\mathcal{P}_i|} r^w_{\mathcal{I}_i^j} = r^w_1 = E\{w_1\}\,. \tag{B.2.36}$$

This result is the same as that in (B.1.24) for the univariate case. Substituting $w_1 = v_i$ yields:

$$\kappa^v_i \equiv \kappa^{v,1}_i \triangleq \mathrm{cum}(v_i) = r^v_i \equiv r^{v,1}_i = E\{v_i\}\,. \tag{B.2.37}$$

Example for $l = 2$
For $l = 2$, the 'partitioning table' (similar to Tables B.1 and B.2 in Section B.1.3.5, except that the last column has been eliminated) is given in Table B.4. Substituting the values in this table into (B.2.35) yields:

$$\begin{aligned}\kappa^w_{12} \equiv \kappa^{w,2}_{12} &= \mathrm{cum}(w_1,w_2) = \sum_{i=1}^{2}(-1)^{|\mathcal{P}_i|-1}(|\mathcal{P}_i|-1)!\prod_{j=1}^{|\mathcal{P}_i|} r^w_{\mathcal{I}_i^j}\\ &= r^w_{\mathcal{I}_1^1} - r^w_{\mathcal{I}_2^1}\,r^w_{\mathcal{I}_2^2} = r^w_{12} - r^w_1\,r^w_2 = E\big\{w_1w_2\big\} - E\big\{w_1\big\}\,E\big\{w_2\big\}\,.\end{aligned} \tag{B.2.38}$$

This is the familiar expression for the cross-covariance of w_1 and w_2. Substituting $w_1 = v_{i_1}$ and $w_2 = v_{i_2}$ yields:

$$\kappa^v_{i_1i_2} \equiv \kappa^{v,2}_{i_1i_2} \triangleq \mathrm{cum}(v_{i_1},v_{i_2}) = r^v_{i_1i_2} - r^v_{i_1}\,r^v_{i_2} = E\big\{v_{i_1}v_{i_2}\big\} - E\big\{v_{i_1}\big\}\,E\big\{v_{i_2}\big\}\,. \tag{B.2.39}$$

Table B.4: *Partitioning of* $\mathcal{I} = \{1, 2\}$.

i	$\mathcal{P}_i$	$\lvert\mathcal{P}_i\rvert$	j	$\mathcal{I}_i^j$
1	$\{(1,2)\}$	1	1	$(1,2)$
2	$\{(1),(2)\}$	2	1	(1)
			2	(2)

Example for $l = 3$
For $l = 3$, the partitioning of the set $\mathcal{I} \triangleq \{1, 2, 3\}$ is given in Table B.1 of Section B.1.3.5. Substituting the values in this table into (B.2.35) yields:

$$\begin{aligned}
\kappa_{123}^w \equiv \kappa_{123}^{w,3} = \operatorname{cum}(w_1, w_2, w_3) &= \sum_{i=1}^{5} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} r_{\mathcal{I}_i^j}^w \\
&= r_{\mathcal{I}_1^1}^w - r_{\mathcal{I}_2^1}^w r_{\mathcal{I}_2^2}^w - r_{\mathcal{I}_3^1}^w r_{\mathcal{I}_3^2}^w - r_{\mathcal{I}_4^1}^w r_{\mathcal{I}_4^2}^w + 2\, r_{\mathcal{I}_5^1}^w r_{\mathcal{I}_5^2}^w r_{\mathcal{I}_5^3}^w \\
&= r_{123}^w - r_1^w r_{23}^w - r_2^w r_{13}^w - r_3^w r_{12}^w + 2\, r_1^w r_2^w r_3^w \\
&= E\{w_1 w_2 w_3\} - E\{w_1\} E\{w_2 w_3\} - E\{w_2\} E\{w_1 w_3\} \\
&\quad - E\{w_3\} E\{w_1 w_2\} + 2\, E\{w_1\} E\{w_2\} E\{w_3\}\,. \qquad \text{(B.2.40)}
\end{aligned}$$

Substituting $w_1 = v_{i_1}, w_2 = v_{i_2}$ and $w_3 = v_{i_3}$ yields:

$$\begin{aligned}
\kappa_{i_1 i_2 i_3}^v \equiv \kappa_{i_1 i_2 i_3}^{v,3} \triangleq \operatorname{cum}(v_{i_1}, v_{i_2}, v_{i_3}) &= r_{i_1 i_2 i_3}^v - r_{i_1}^v r_{i_2 i_3}^v - r_{i_2}^v r_{i_1 i_3}^v - r_{i_3}^v r_{i_1 i_2}^v + r_{i_1}^v r_{i_2}^v r_{i_3}^v \\
&= E\{v_{i_1} v_{i_2} v_{i_1}\} - E\{v_{i_1}\} E\{v_{i_2} v_{i_3}\} - E\{v_{i_2}\} E\{v_{i_1} v_{i_3}\} \\
&\quad - E\{v_{i_3}\} E\{v_{i_1} v_{i_2}\} + 2\, E\{v_{i_1}\} E\{v_{i_2}\} E\{v_{i_3}\}\,. \qquad \text{(B.2.41)}
\end{aligned}$$

Example for $l = 4$
For $l = 4$, the partitioning of the set $\mathcal{I} \triangleq \{1, 2, 3, 4\}$ is given in Table B.2 of Section B.1.3.5. Substituting the values in this table into (B.2.35) yields:

$$\begin{aligned}
\kappa_{1234}^w \equiv \kappa_{1234}^{w,4} = \operatorname{cum}(w_1, w_2, w_3, w_4) &= \sum_{i=1}^{15} (-1)^{|\mathcal{P}_i|-1} (|\mathcal{P}_i| - 1)! \prod_{j=1}^{|\mathcal{P}_i|} r_{\mathcal{I}_i^j}^w \\
&= r_{\mathcal{I}_1^1}^w - r_{\mathcal{I}_2^1}^w r_{\mathcal{I}_2^2}^w - r_{\mathcal{I}_3^1}^w r_{\mathcal{I}_3^2}^w - r_{\mathcal{I}_4^1}^w r_{\mathcal{I}_4^2}^w - r_{\mathcal{I}_5^1}^w r_{\mathcal{I}_5^2}^w - r_{\mathcal{I}_6^1}^w r_{\mathcal{I}_6^2}^w - r_{\mathcal{I}_7^1}^w r_{\mathcal{I}_7^2}^w - r_{\mathcal{I}_8^1}^w r_{\mathcal{I}_8^2}^w \\
&\quad + 2\, r_{\mathcal{I}_9^1}^w r_{\mathcal{I}_9^2}^w r_{\mathcal{I}_9^3}^w + 2\, r_{\mathcal{I}_{10}^1}^w r_{\mathcal{I}_{10}^2}^w r_{\mathcal{I}_{10}^3}^w + 2\, r_{\mathcal{I}_{11}^1}^w r_{\mathcal{I}_{11}^2}^w r_{\mathcal{I}_{11}^3}^w + 2\, r_{\mathcal{I}_{12}^1}^w r_{\mathcal{I}_{12}^2}^w r_{\mathcal{I}_{12}^3}^w \\
&\quad + 2\, r_{\mathcal{I}_{13}^1}^w r_{\mathcal{I}_{13}^2}^w r_{\mathcal{I}_{13}^3}^w + 2\, r_{\mathcal{I}_{14}^1}^w r_{\mathcal{I}_{14}^2}^w r_{\mathcal{I}_{14}^3}^w - 6\, r_{\mathcal{I}_{15}^1}^w r_{\mathcal{I}_{15}^2}^w r_{\mathcal{I}_{15}^3}^w r_{\mathcal{I}_{15}^4}^w \\
&= r_{1234}^w - r_1^w r_{234}^w - r_2^w r_{134}^w - r_3^w r_{124}^w - r_4^w r_{123}^w - r_{12}^w r_{34}^w - r_{13}^w r_{24}^w - r_{14}^w r_{23}^w \\
&\quad + 2\, r_1^w r_2^w r_{34}^w + 2\, r_1^w r_3^w r_{24}^w + 2\, r_1^w r_4^w r_{23}^w + 2\, r_2^w r_3^w r_{14}^w \\
&\quad + 2\, r_2^w r_4^w r_{13}^w + 2\, r_3^w r_4^w r_{12}^w - 6\, r_1^w r_2^w r_3^w r_4^w \\
&= E\{w_1 w_2 w_3 w_4\} - E\{w_1\} E\{w_2 w_3 w_4\} - E\{w_2\} E\{w_1 w_3 w_4\} \\
&\quad - E\{w_3\} E\{w_1 w_2 w_4\} - E\{w_4\} E\{w_1 w_2 w_3\} - E\{w_1 w_2\} E\{w_3 w_4\} \\
&\quad - E\{w_1 w_3\} E\{w_2 w_4\} - E\{w_1 w_4\} E\{w_2 w_3\} + 2E\{w_1\} E\{w_2\} E\{w_3 w_4\} \\
&\quad + 2\, E\{w_1\} E\{w_3\} E\{w_2 w_4\} + 2\, E\{w_1\} E\{w_4\} E\{w_2 w_3\}
\end{aligned}$$

$$
\begin{aligned}
&+ 2\,E\{w_2\}\,E\{w_3\}\,E\{w_1 w_4\} + 2\,E\{w_2\}\,E\{w_4\}\,E\{w_1 w_3\} \\
&+ 2\,E\{w_3\}\,E\{w_4\}\,E\{w_1 w_2\} - 6\,E\{w_1\}\,E\{w_2\}\,E\{w_3\}\,E\{w_4\}\,. \qquad \text{(B.2.42)}
\end{aligned}
$$

Substituting $w_1 = v_{i_1}, w_2 = v_{i_2}, w_3 = v_{i_3}$ and $w_4 = v_{i_4}$ yields:

$$
\begin{aligned}
&\kappa^v_{i_1 i_2 i_3 i_4} \equiv \kappa^{v,4}_{i_1 i_2 i_3 i_4} \triangleq \operatorname{cum}(v_{i_1}, v_{i_2}, v_{i_3}, v_{i_4}) \\
&= r^v_{i_1 i_2 i_3 i_4} - r^v_{i_1}\, r^v_{i_2 i_3 i_4} - r^v_{i_2}\, r^v_{i_1 i_3 i_4} - r^v_{i_3}\, r^v_{i_1 i_2 i_4} - r^v_{i_4}\, r^v_{i_1 i_2 i_3} - r^v_{i_1 i_2}\, r^v_{i_3 i_4} \\
&\quad - r^v_{i_1 i_3}\, r^v_{i_2 i_4} - r^v_{i_1 i_4}\, r^v_{i_2 i_3} + 2\, r^v_{i_1}\, r^v_{i_2}\, r^v_{i_3 i_4} + 2\, r^v_{i_1}\, r^v_{i_3}\, r^v_{i_2 i_4} + 2\, r^v_{i_1}\, r^v_{i_4}\, r^v_{i_2 i_3} \\
&\quad + 2\, r^v_{i_2}\, r^v_{i_3}\, r^v_{i_1 i_4} + 2\, r^v_{i_2}\, r^v_{i_4}\, r^v_{i_1 i_3} + 2\, r^v_{i_3}\, r^v_{i_4}\, r^v_{i_1 i_2} - 6\, r^v_{i_1}\, r^v_{i_2}\, r^v_{i_3}\, r^v_{i_4} \\
&= E\{v_{i_1} v_{i_2} v_{i_3} v_{i_4}\} - E\{v_{i_1}\}\,E\{v_{i_2} v_{i_3} v_{i_4}\} - E\{v_{i_2}\}\,E\{v_{i_1} v_{i_3} v_{i_4}\} \\
&\quad - E\{v_{i_3}\}\,E\{v_{i_1} v_{i_2} v_{i_4}\} - E\{v_{i_4}\}\,E\{v_{i_1} v_{i_2} v_{i_3}\} - E\{v_{i_1} v_{i_2}\}\,E\{v_{i_3} v_{i_4}\} \\
&\quad - E\{v_{i_1} v_{i_3}\}\,E\{v_{i_2} v_{i_4}\} - E\{v_{i_1} v_{i_4}\}\,E\{v_{i_2} v_{i_3}\} + 2\,E\{v_{i_1}\}\,E\{v_{i_2}\}\,E\{v_{i_3} v_{i_4}\} \\
&\quad + 2\,E\{v_{i_1}\}\,E\{v_{i_3}\}\,E\{v_{i_2} v_{i_4}\} + 2\,E\{v_{i_1}\}\,E\{v_{i_4}\}\,E\{v_{i_2} v_{i_3}\} \\
&\quad + 2\,E\{v_{i_2}\}\,E\{v_{i_3}\}\,E\{v_{i_1} v_{i_4}\} + 2\,E\{v_{i_2}\}\,E\{v_{i_4}\}\,E\{v_{i_1} v_{i_3}\} \\
&\quad + 2\,E\{v_{i_3}\}\,E\{v_{i_4}\}\,E\{v_{i_1} v_{i_2}\} - 6\,E\{v_{i_1}\}\,E\{v_{i_2}\}\,E\{v_{i_3}\}\,E\{v_{i_4}\}\,. \qquad \text{(B.2.43)}
\end{aligned}
$$

Example for $l = 1, 2, 3, 4$ when all variables have zero mean
When all involved random variables have zero mean, the multivariate moment-to-cumulant conversion formulas (B.2.37), (B.2.39), (B.2.41) and (B.2.43) become respectively:

$$
\begin{aligned}
\kappa^v_i \equiv \kappa^{v,1}_i &\triangleq \operatorname{cum}(v_i) = 0\,; && \text{(B.2.44a)} \\
\kappa^v_{i_1 i_2} \equiv \kappa^{v,2}_{i_1 i_2} &\triangleq \operatorname{cum}(v_{i_1}, v_{i_2}) = r^v_{i_1 i_2} = E\{v_{i_1} v_{i_2}\}\,; && \text{(B.2.44b)} \\
\kappa^v_{i_1 i_2 i_3} \equiv \kappa^{v,3}_{i_1 i_2 i_3} &\triangleq \operatorname{cum}(v_{i_1}, v_{i_2}, v_{i_3}) = r^v_{i_1 i_2 i_3} = E\{v_{i_1} v_{i_2} v_{i_1}\}\,; && \text{(B.2.44c)} \\
\kappa^v_{i_1 i_2 i_3 i_4} \equiv \kappa^{v,4}_{i_1 i_2 i_3 i_4} &\triangleq \operatorname{cum}(v_{i_1}, v_{i_2}, v_{i_3}, v_{i_4}) \\
&= r^v_{i_1 i_2 i_3 i_4} - r^v_{i_1 i_2}\, r^v_{i_3 i_4} - r^v_{i_1 i_3}\, r^v_{i_2 i_4} - r^v_{i_1 i_4}\, r^v_{i_2 i_3} \\
&= E\{v_{i_1} v_{i_2} v_{i_3} v_{i_4}\} - E\{v_{i_1} v_{i_2}\}\,E\{v_{i_3} v_{i_4}\} \\
&\quad - E\{v_{i_1} v_{i_3}\}\,E\{v_{i_2} v_{i_4}\} - E\{v_{i_1} v_{i_4}\}\,E\{v_{i_2} v_{i_3}\}\,. && \text{(B.2.44d)}
\end{aligned}
$$

Finally, note that the univariate expressions for a single component, say v_i, can be obtained by substituting $w_1 = \cdots w_l = v_i$ into (B.2.35), or $v_{i_1} = \cdots v_{i_l} = v_i$ into the expressions involving $v_{i_1}, \ldots, v_{i_l}$. For example, substituting $w_1 = w_2 = w_3 = w_4 = v_i$ into (B.2.42) or $v_{i_1} = v_{i_2} = v_{i_3} = v_{i_4} = v_i$ into (B.2.43) yields:

$$
\kappa^v_{iiii} \equiv \kappa^{v,4}_{iiii} \triangleq \operatorname{cum}(v_i, v_i, v_i, v_i) = r^v_{iiii} - 4 r^v_i\, r^v_{iii} - 3\left(r^v_{ii}\right)^2 + 12\left(r^v_i\right)^2 r^v_{ii} - 6\left(r^v_i\right)^4 .
$$

By noting that $\kappa^v_{iiii} \equiv \kappa^{v,4}_{iiii} \equiv \kappa^{v_i,4}$ it can be seen that this result is exactly equal to that in (B.1.22) and (B.1.23).

B.2.6.7 Properties of joint cumulants

The following general properties of joint cumulants are very important for manipulating general expressions:

CP1: For all constant coefficients $c^1, \ldots, c^l \in \mathbb{R}$:

$$\operatorname{cum}\left(c^1 w_1, \ldots, c^l w_l\right) = c^1 \cdots c^l \operatorname{cum}\left(w_1, \ldots, w_l\right). \tag{B.2.45}$$

CP2: Cumulants are symmetric in their arguments:

$$\operatorname{cum}\left(w_{i_1}, \ldots, w_{i_l}\right) = \operatorname{cum}\left(w_1, \ldots, w_l\right) \tag{B.2.46}$$

for any permutation $(i_1, \ldots, i_l)$ of $(1, \ldots, l)$.

CP3: Cumulants are insensitive to changes in the mean values except for the first-order cumulant. Hence for all constant coefficients $\mu_1, \ldots, \mu_l \in \mathbb{R}$ and $l \geq 2$:

$$\operatorname{cum}\left(\mu_1 + w_1, \ldots, \mu_l + w_l\right) = \operatorname{cum}\left(w_1, \ldots, w_l\right). \tag{B.2.47}$$

The first order cumulant of a random variable is equal to the mean, and thus:

$$\operatorname{cum}\left(\mu + w\right) = \mu + \operatorname{cum}\left(w\right).$$

CP4: Cumulants are additive in their arguments (also for variables that are not statistically independent):

$$\begin{aligned}&\operatorname{cum}\left(w_1, \ldots, w_{i-1}, w_i + v_i, w_{i+1}, \ldots, w_l\right)\\ &= \operatorname{cum}\left(w_1, \ldots, w_l\right) + \operatorname{cum}\left(w_1, \ldots, w_{i-1}, v_i, w_{i+1}, \ldots, w_l\right).\end{aligned} \tag{B.2.48}$$

CP5: If a subset of the l random variables $w_1, \ldots, w_l$ is statistically independent of the rest, then:

$$\operatorname{cum}\left(w_1, \ldots, w_l\right) = 0. \tag{B.2.49}$$

CP6: If the random variables $v_1, \ldots, v_l$ are statistically independent of the random variables $w_1, \ldots, w_l$, then:

$$\operatorname{cum}\left(v_1 + w_1, \ldots, v_l + w_l\right) = \operatorname{cum}\left(v_1, \ldots, v_l\right) + \operatorname{cum}\left(w_1, \ldots, w_l\right). \tag{B.2.50}$$

CP7: Cumulants are multilinear in their arguments. If the random vector $\mathbf{w} \in \mathbb{R}_P$ is a linear transformation of the random vector $\mathbf{v} \in \mathbb{R}_G$, i.e. $\mathbf{w} = \mathbf{C}\mathbf{v}$ for some matrix $\mathbf{C} = [c_i^j] \in \mathbb{R}_P^G$, then:

$$\begin{aligned}\operatorname{cum}\left(w_{i_1}, \ldots, w_{i_l}\right) &= \sum_{j_1=1}^{G} \cdots \sum_{j_l=1}^{G} c_{i_1}^{j_1} \cdots c_{i_l}^{j_l} \operatorname{cum}\left(v_{j_1}, \ldots, v_{j_l}\right)\\ \forall\, (i_1, \ldots, i_l) &\in \mathcal{I}_{\mathrm{t},P}^{l}, \quad \forall\, l \in \mathbb{N}.\end{aligned} \tag{B.2.51}$$

CP8: Cumulants measure deviation from non-Gaussianity. If the random vector $\mathbf{w}$ is multivariate Gaussian distributed, then all cumulants of (total) order larger than two are identically zero:

$$\operatorname{cum}\left(w_{i_1}, \ldots, w_{i_l}\right) = 0 \qquad \forall\, (i_1, \ldots, i_l) \in \mathbb{N}_l, \quad \forall\, l \geq 3. \tag{B.2.52}$$

All those properties can be derived from the definition of the cumulants and/or the results in Section B.2.6.6. Most proofs can be found in [112]. Comparing these properties to the properties of moments listed in Section B.2.6.3, we see that cumulants have many nice properties that moments do not have. This is the main reason for working with cumulants. Several of

the properties above are closely related. For example, *CP6* can be derived from *CP4* and *CP5* only. Likewise, *CP7* can be derived from *CP1* and *CP4*. Because *CP7* plays a very important role in this work and to give some feeling for the manipulation of cumulants, we briefly derive *CP7* from *CP1* and *CP4* as follows:

$$\begin{aligned}
&\operatorname{cum}\left(w_{i_1},\ldots,w_{i_l}\right) = \operatorname{cum}\left(\sum_{j_1=1}^{G} c_{i_1}^{j_1} v_{j_1},\ldots,\sum_{j_l=1}^{G} c_{i_l}^{j_l} v_{j_l}\right)\\
&\overset{CP4}{=} \sum_{j_1=1}^{G}\cdots\sum_{j_l=1}^{G} \operatorname{cum}\left(c_{i_1}^{j_1} v_{j_1},\ldots,c_{i_l}^{j_l} v_{j_l}\right)\\
&\overset{CP1}{=} \sum_{j_1=1}^{G}\cdots\sum_{j_l=1}^{G} c_{i_1}^{j_1}\cdots c_{i_l}^{j_l} \operatorname{cum}\left(v_{j_1},\ldots,v_{j_l}\right)\\
&\forall\, 1\leq i_1,\ldots,i_l\leq P,\quad \forall\, l\in\mathbb{N}\,.
\end{aligned}$$

Note that in our compact notation, *CP7* reads as follows:

$$\boxed{\kappa_{\mathbf{i}_l}^{w} = \sum_{\mathbf{j}_l\in\mathcal{J}_{\mathfrak{t},G}^{l}} \check{c}_{\mathbf{i}_l}^{\mathbf{j}_l}\,\kappa_{\mathbf{j}_l}^{v} \qquad \forall\,\mathbf{i}_l\in\mathcal{I}_{\mathfrak{t},P}^{l}\,.} \tag{B.2.53}$$

B.2.6.8 Statistical independence criterion based on joint cumulants

The statistical independence criterion in (B.2.10) can also be expressed in terms of joint cumulants. This is often very useful for practical purposes. The following theorem deals with this issue:

Theorem B.2.3. Statistical independence criterion based on joint cumulants.
The random variables $v_1,\ldots,v_G$ are mutually statistically independent if and only if their cross-cumulants of all possible orders are zero:

$$\kappa_{\mathbf{i}_l}^{v} \triangleq \operatorname{cum}\left(v_{i_1},\ldots,v_{i_l}\right) = 0 \qquad \forall\,(i_1,\ldots,i_l)\in\mathcal{I}_{\mathfrak{c},G}^{l}\,,\quad \forall\, l\in\mathbb{N}\,. \tag{B.2.54}$$

See Appendix A for the definition and explanation of the notation $\mathcal{I}_{\mathfrak{c},G}^{l}$. Note that by 'order' we mean 'total order'.

Proof. By definition (B.2.10), the random variables $v_1,\ldots,v_G$ are mutually statistically independent if and only if their joint pdf factorizes into marginal pdf's:

$$p_{v_1,\ldots,v_G}\left(v_1,\ldots,v_G\right) = \prod_{m=1}^{G} p_{v_m}(v_m)\,.$$

In terms of the first characteristic functions defined by (B.2.14), this is equivalent to:

$$\phi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = \prod_{m=1}^{G} \phi^{v_m}(\omega^m)\,, \tag{B.2.55}$$

where $\phi^{v_m}(\omega^m)$ denotes the (marginal) characteristic function of v_m (which is univariate). From (B.2.15) it follows that this can be written in terms of the second characteristic functions as follows:

$$\psi^{\mathbf{v}}(\tilde{\boldsymbol{\omega}}) = \sum_{m=1}^{G} \psi^{v_m}(\omega^m)\,. \tag{B.2.56}$$

Now, by writing both sides of this equation in terms of Taylor expansions (see (B.2.32) and (B.1.19)), the following equation is obtained:

$$\sum_{l=0}^{\infty} \sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^l} \kappa_{\mathbf{i}_l}^v \frac{(\jmath)^l}{l!} \omega^{i_1} \cdots \omega^{i_l} = \sum_{m=1}^{G} \sum_{l=0}^{\infty} \kappa_{(m)_l}^v \frac{(\jmath)^l}{l!} (\omega^m)^l \,. \tag{B.2.57}$$

Note that $\kappa_{(m)_l}^v \equiv \kappa^{v_m,l}$ denotes the l-th order univariate cumulant of v_m (see (A.2.2) for the definition of $(m)_l$). In order for this expression to hold for all $(\omega^1, \ldots, \omega^G) \in \mathbb{R}^G$, the coefficient of each product term of ω's on the left hand side must equal the coefficient of the same product term on the right hand side, i.e.:

$$\sum_{\mathbf{i}_l \in \mathcal{I}_{\mathrm{t},G}^l} \kappa_{\mathbf{i}_l}^v \frac{(\jmath)^l}{l!} \omega^{i_1} \cdots \omega^{i_l} = \sum_{m=1}^{G} \kappa_{(m)_l}^v \frac{(\jmath)^l}{l!} (\omega^m)^l \qquad \forall\, l \in \mathbb{N} \,.$$

Hence, only left-hand terms with $i_1 = \cdots = i_l = m$, i.e. with equal indices, have non-zero coefficients and all other terms are zero. The non-zero coefficients correspond to the auto-cumulants (or marginal cumulants) of $v_1, \ldots, v_G$, whereas the zero coefficients correspond to their cross-cumulants. The index tuples $\mathbf{i}_l \triangleq (i_1, \ldots, i_l)$ that index these cross-cumulants are precisely the elements of the set $\mathcal{I}_{\mathrm{c},G}^l$ defined in (A.2.18). Now, since $\kappa_{\mathbf{i}_l}^v \triangleq \mathrm{cum}\big(v_{i_1}, \ldots, v_{i_l}\big)$, the theorem is proven. □

A similar version of this theorem is proven in [189]. Here, we presented a version that is consistent with our notation and requirements. Two more useful theorems in the same spirit as Theorem B.2.3 are now given.

Definition B.2.4. Statistical independence up to order l**.** The random variables $v_1, \ldots, v_G$ are said to be mutually statistically independent *up to* order l if and only if all their cross-cumulants of orders smaller than or equal to l are zero:

$$\kappa_{\mathbf{i}_k}^v \triangleq \mathrm{cum}\big(v_{i_1}, \ldots, v_{i_k}\big) = 0 \qquad \forall\, (i_1, \ldots, i_k) \in \mathcal{I}_{\mathrm{c},G}^k \,, \quad \forall\, k \leq l \,. \tag{B.2.58}$$

Definition B.2.5. Statistical independence at order l**.** The random variables v_1, ..., v_G are said to be mutually statistically independent *at* order l if and only if all their cross-cumulants of order l are zero:

$$\kappa_{\mathbf{i}_l}^v \triangleq \mathrm{cum}\big(v_{i_1}, \ldots, v_{i_l}\big) = 0 \qquad \forall\, (i_1, \ldots, i_l) \in \mathcal{I}_{\mathrm{c},G}^l \,. \tag{B.2.59}$$

For example, mutual statistical independence *at* order two means uncorrelatedness.

B.2.6.9 Estimation of multivariate expectations, moments and cumulants

The estimation or computation of multivariate/joint expectations, moments, cumulants and other statistical characterizations is completely similar to the univariate case discussed in Section B.1.3.6. Suppose that the expectation $E\{g(\mathbf{v})\} = E\big\{g\big(v_1, \ldots, v_G\big)\big\}$ in (B.2.16) has to be estimated. We assume that the involved joint probability density function $p_{\mathbf{v}}(\mathbf{v})$ is unknown, but that instead a set $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\}$ of K samples of $\mathbf{v}$ is available. Then, a simple standard estimator is given by [154]:

$$\hat{g}(\mathbf{v}) = \frac{1}{K} \sum_{k=1}^{K} g(\mathbf{v}_k) \,. \tag{B.2.60}$$

B.3 Complex-valued statistical quantities

The definitions and results presented above for the real-valued case can be generalized to the complex-valued case in a structural and straightforward manner. The main difference is that now we need to combine the statistics of real and imaginary parts. In the thesis several examples of this are provided. We leave the various generalizations to the reader and finish this appendix with some remarks concerning the *choice of the conjugation pattern*. As we mention several times in this thesis, the conjugation pattern of a moment or cumulant function can be chosen in several ways, which may or may not be sensible depending on the complex-valued statistical structure of the source signals. For example, see Section 3.3.6. As another example, consider the two arguments of the moment function $\mathrm{mom}(\cdot,\cdot)$ in (A.6.4), which can be conjugated in four ways, viz. $\mathrm{mom}\big(\cdot,\cdot\big)$, $\mathrm{mom}\big(\cdot,(\cdot)^*\big)$, $\mathrm{mom}\big((\cdot)^*,\cdot\big)$ and $\mathrm{mom}\big((\cdot)^*,(\cdot)^*\big)$. Since the fourth function value can be obtained from the first by conjugation, and likewise the third from the second, two moment functions are redundant. For example, taking the random variables v and w as the first and second arguments of $\mathrm{mom}(\cdot,\cdot)$ respectively, it is easy to see that $\big(E\left\{(v)^*w^*\right\}\big)^* = \big(\mathrm{mom}\big((v)^*,(w)^*\big)\big)^* = E\left\{vw\right\} = \mathrm{mom}\big(v,w\big)$, and similarly $\mathrm{mom}\big((v)^*,w\big) = E\left\{(v)^*w\right\} = \big(E\left\{v\,(w)^*\right\}\big)^* = \big(\mathrm{mom}\big((v,(w)^*\big)\big)^*$. Depending on the properties of the joint probability density function of v and w (such as even or odd symmetries), either $\mathrm{mom}\,(v,w) = 0$ and $\mathrm{mom}\,(v,w^*) = 0$, or $\mathrm{mom}\,(v,w) = 0$ and $\mathrm{mom}\,(v,w^*) \neq 0$, or $\mathrm{cum}\,(v,w) \neq 0$ and $\mathrm{mom}\,(v^*,w) = 0$, or $\mathrm{mom}\,(v,w) \neq 0$ and $\mathrm{mom}\,(v,w^*) \neq 0$. Similarly, for applications dealing with vector random processes, the most suitable conjugation pattern depends on the type of the involved (stochastic) signals. For example, vector noise processes like $\boldsymbol{\nu}[n]$ in (1.1.1) are often assumed to be zero-mean stationary white circularly complex, which means that $E\big\{\nu_{i_1}[n_1]\,(\nu_{i_2}[n_2])^*\big\} = (\sigma^\nu)^2\,\delta_{i_1 i_2}\delta_{n_1 n_2}$ and $E\big\{\nu_{i_1}[n_1]\,\nu_{i_2}[n_2]\big\} = 0$, where $\delta_{i_1 i_2}$ and $\delta_{n_1 n_2}$ are Kronecker delta functions (see Appendix A) and $(\sigma^\nu)^2$ is a real-valued positive number. The point we want to make here is that those expressions are merely a result of the specific assumptions on the joint pdf $p_{\boldsymbol{\nu}[n]}(\boldsymbol{\nu}[n])$ of the signals. If a pdf with different properties is assumed, the reverse may hold, i.e. $E\big\{\nu_{i_1}[n_1]\,(\nu_{i_2}[n_2])^*\big\} = 0$, while $E\big\{\nu_{i_1}[n_1]\,\nu_{i_2}[n_2]\big\} = (\sigma^\nu)^2\,\delta_{i_1 i_2}\delta_{n_1 n_2}$. The same reasoning applies to higher order moments and cumulants.

APPENDIX C

Basic linear algebra facts

In this appendix some important basic results from linear algebra [72, 115, 150] are listed.

C.1 Definitions

Definition C.1.1. Rank; preliminary definition. The *rank* $d(\mathbf{B}) \triangleq \mathrm{rank}(\mathbf{B})$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$ is defined as the number of linearly independent columns or rows of $\mathbf{B}$ (recall that for any matrix the number of linearly independent columns equals the number of linearly independent rows). The rank of $\mathbf{B}$ equals the dimension $\dim\left(\mathcal{L}\left(\{\mathbf{b}^i\}^{1\leq i\leq N}\right)\right)$ of the linear subspace $\mathcal{L}\left(\{\mathbf{b}^i\}^{1\leq i\leq N}\right)$ spanned by the columns of $\mathbf{B}$, or the dimension $\dim\left(\mathcal{L}\left(\{\tilde{\mathbf{b}}_i\}_{1\leq i\leq M}\right)\right)$ of the linear subspace $\mathcal{L}\left(\{\tilde{\mathbf{b}}_i\}_{1\leq i\leq M}\right)$ spanned by the rows of $\mathbf{B}$. Note that $\dim(\mathbf{B}) \leq \min(M, N)$.

Definition C.1.2. Orthogonal or perpendicular vectors. Let S be an *inner product space*, i.e. a vector space equipped with an inner product. Then, two vectors $\mathbf{v} \in S$ and $\mathbf{w} \in S$ are said to be *orthogonal* or *perpendicular* if their inner product equals zero:

$$\langle \mathbf{v}, \mathbf{w} \rangle = 0\,. \tag{C.1.1}$$

Notationally, this is usually stated as $\mathbf{v} \perp \mathbf{w}$.

Definition C.1.3. Orthogonal complement. Let V be a subspace of an inner product space S. Then, the *orthogonal complement* $V^\perp$ is defined as the space of all vectors in S that are orthogonal to V, i.e.:

$$V^\perp \triangleq \{\mathbf{w} \in S \mid \mathbf{w} \perp \mathbf{v} \quad \forall\, \mathbf{v} \in V\}\,. \tag{C.1.2}$$

From now on, we assume that all vector spaces we work with are closed and equipped with an inner product. Therefore, we will use the terms 'vector space' and 'inner product space' interchangeably to denote a closed inner product space. All inner products used in this thesis are equal or similar to the following inner product that is defined on the familiar vector space $\mathbb{C}_N$ of length-N (column) vectors:

$$\langle \mathbf{v}, \mathbf{w} \rangle \triangleq \sum_{i=1}^{N} v_i (w_i)^*\,. \tag{C.1.3}$$

Definition C.1.4. Orthogonal or perpendicular subspaces. Let S be a vector space, i.e. a closed inner product space, and let V and W be subspaces of S. Then, V and W are said to be *orthogonal* or *perpendicular* if every vector $\mathbf{v} \in V$ is orthogonal to every vector $\mathbf{w} \in W$:

$$V \perp W \qquad \Longleftrightarrow \qquad \mathbf{v} \perp \mathbf{w} \quad \forall\, \mathbf{v} \in V,\ \mathbf{w} \in W\,. \tag{C.1.4}$$

Some important properties of orthogonal complements and subspaces can be formulated as follows. Let S be a vector space and let V and W be closed linear subspaces of S. Then, the following properties hold:

OCP1: $V = V^{\perp\perp}$;

OCP2: $\mathbf{v} \in V \cap V^{\perp} \quad \Longrightarrow \quad \mathbf{v} = \mathbf{0}$;

OCP3: $V \subseteq W \quad \Longrightarrow \quad W^{\perp} \subseteq V^{\perp}$;

OCP4: $\{\mathbf{0}\}^{\perp} = S$;

OCP5: $S^{\perp} = \{\mathbf{0}\}$;

OCP6: $S = V \oplus V^{\perp}$.

C.2 The four fundamental subspaces of a matrix

This section discusses the four fundamental subspaces of a matrix and various relations between them.

C.2.1 Definitions

Definition C.2.1. Column space; also called range space or simply range. The *column space* $\mathcal{R}_{\mathrm{c}}(\mathbf{B})$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$, also called the *range space* or simply *range* of $\mathbf{B}$, is the set of all column vectors $\mathbf{w} \in \mathbb{C}_M$ such that $\mathbf{w} = \mathbf{B}\mathbf{v}$ for some $\mathbf{v} \in \mathbb{C}_N$:

$$\mathcal{R}_{\mathrm{c}}(\mathbf{B}) \triangleq \{\mathbf{w} = \mathbf{B}\mathbf{v} \mid \mathbf{v} \in \mathbb{C}_N\}, \tag{C.2.1}$$

which can also be written as follows:

$$\mathcal{R}_{\mathrm{c}}(\mathbf{B}) \triangleq \left\{ \sum_{i=1}^{N} v_i \mathbf{b}^i \;\middle|\; v_i \in \mathbb{C} \quad \forall\, 1 \leq i \leq N \right\} = \mathcal{L}\left(\mathbf{b}^1, \ldots, \mathbf{b}^N\right). \tag{C.2.2}$$

Hence, $\mathcal{R}_{\mathrm{c}}(\mathbf{B})$ is the set of all linear combinations of the columns of $\mathbf{B}$, i.e. the linear subspace $\mathcal{L}\left(\mathbf{b}^1, \ldots, \mathbf{b}^N\right)$ spanned by the columns of $\mathbf{B}$. It is a subspace of $\mathbb{C}_M$ with dimension equal to the rank of $\mathbf{B}$, i.e.:

$$\dim(\mathcal{R}_{\mathrm{c}}(\mathbf{B})) = \mathrm{rank}(\mathbf{B}). \tag{C.2.3}$$

Definition C.2.2. Row space. The *row space* $\mathcal{R}_{\mathrm{r}}(\mathbf{B})$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$ is the set of all row vectors $\tilde{\mathbf{w}} \in \mathbb{C}^N$ such that $\tilde{\mathbf{w}} = \tilde{\mathbf{v}}\mathbf{B}^*$ for some $\tilde{\mathbf{v}} \in \mathbb{C}^M$ ($\mathbf{B}^*$ is the conjugate of $\mathbf{B}$):

$$\mathcal{R}_{\mathrm{r}}(\mathbf{B}) \triangleq \{\tilde{\mathbf{w}} = \tilde{\mathbf{v}}\mathbf{B}^* \mid \tilde{\mathbf{v}} \in \mathbb{C}^M\}, \tag{C.2.4}$$

which can also be written as follows:

$$\mathcal{R}_{\mathrm{r}}(\mathbf{B}) \triangleq \left\{ \sum_{i=1}^{M} v^i \left(\tilde{\mathbf{b}}_i\right)^* \;\middle|\; v^i \in \mathbb{C} \quad \forall\, 1 \leq i \leq M \right\} = \mathcal{L}\left(\left(\tilde{\mathbf{b}}_1\right)^*, \ldots, \left(\tilde{\mathbf{b}}_M\right)^*\right). \tag{C.2.5}$$

Hence, $\mathcal{R}_{\mathrm{r}}(\mathbf{B})$ is the set of all linear combinations of the conjugated rows of $\mathbf{B}$, i.e. the linear subspace $\mathcal{L}\left(\left(\tilde{\mathbf{b}}_1\right)^*, \ldots, \left(\tilde{\mathbf{b}}_M\right)^*\right)$ spanned by the conjugates of the rows of $\mathbf{B}$. It is a subspace of $\mathbb{C}^N$ with dimension equal to the rank of $\mathbf{B}^H$ and $\mathbf{B}$, i.e.:

$$\dim(\mathcal{R}_{\mathrm{r}}(\mathbf{B})) = \mathrm{rank}(\mathbf{B}^H) = \mathrm{rank}(\mathbf{B}) = \dim(\mathcal{R}_{\mathrm{c}}(\mathbf{B})). \tag{C.2.6}$$

Although a more natural name for $\mathcal{R}_{\text{r}}(\mathbf{B})$ would be '*conjugate* row space', the mathematical convention is to use the term 'row space'. Note that $\mathcal{R}_{\text{r}}(\mathbf{B}) = \left(\mathcal{R}_{\text{c}}\left(\mathbf{B}^H\right)\right)^T$. In words, the row space $\mathcal{R}_{\text{r}}(\mathbf{B})$ of $\mathbf{B}$ is *isomorphic* to the column space $\mathcal{R}_{\text{c}}\left(\mathbf{B}^H\right)$ of $\mathbf{B}^H$. Using the symbol '$\div$' to denote isomorphism, we denote this fact by $\mathcal{R}_{\text{r}}(\mathbf{B}) \div \mathcal{R}_{\text{c}}\left(\mathbf{B}^H\right)$.

Definition C.2.3. Null space and nullity. The (right) *null space* $\mathcal{N}_{\text{r}}\big(\mathbf{B}\big)$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$, also called the *kernel* of $\mathbf{B}$, is the set of all column vectors $\mathbf{v} \in \mathbb{C}_N$ such that $\mathbf{B}\mathbf{v} = \mathbf{0}_M$:

$$\mathcal{N}_{\text{r}}\big(\mathbf{B}\big) \triangleq \{\mathbf{v} \in \mathbb{C}_N \mid \mathbf{B}\mathbf{v} = \mathbf{0}_M\}\,. \tag{C.2.7}$$

$\mathcal{N}_{\text{r}}\big(\mathbf{B}\big)$ is a subspace of $\mathbb{C}_N$. Its dimension is called the *(right) nullity* of $\mathbf{B}$ and equals $\dim(\mathcal{N}_{\text{r}}\big(\mathbf{B}\big)) = N - \text{rank}(\mathbf{B})$. Note the following equality:

$$\begin{aligned}\mathcal{N}_{\text{r}}\big(\mathbf{B}\big) &\triangleq \{\mathbf{v} \in \mathbb{C}_N \mid \tilde{\mathbf{b}}_i\mathbf{v} = <\tilde{\mathbf{b}}_i, (\mathbf{v})^* > = 0 \quad \forall\, 1 \le i \le M\} \\ &= \{\mathbf{v} \in \mathbb{C}_N \mid (\mathbf{v})^* \perp \tilde{\mathbf{b}}_i = 0 \quad \forall\, 1 \le i \le M\}\,,\end{aligned} \tag{C.2.8}$$

where the definition of the inner product function $\langle\cdot,\cdot\rangle$ is given by (C.1.3). Hence, $\mathcal{N}_{\text{r}}\big(\mathbf{B}\big)$ consists of all vectors $\mathbf{v} \in \mathbb{C}_N$ whose conjugate is orthogonal to all rows of $\mathbf{B}$.

Definition C.2.4. Left null space and left nullity. The *left null space* $\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$ is the set of all row vectors $\tilde{\mathbf{v}} \in \mathbb{C}^M$ such that $\tilde{\mathbf{v}}\mathbf{B}^* = \tilde{\mathbf{0}}^N$:

$$\mathcal{N}_{\text{l}}\big(\mathbf{B}\big) \triangleq \{\tilde{\mathbf{v}} \in \mathbb{C}^M \mid \tilde{\mathbf{v}}\mathbf{B}^* = \tilde{\mathbf{0}}^N\}\,. \tag{C.2.9}$$

$\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)$ is a subspace of $\mathbb{C}^M$. We will call its dimension the *left nullity* of $\mathbf{B}$. This dimension equals $\dim(\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)) = M - \text{rank}(\mathbf{B})$. The fact that $\mathbf{B}^*$ in (C.2.9) is multiplied by $\tilde{\mathbf{v}}$ from the *left* explains the name 'left null space'. Note that the condition $\tilde{\mathbf{v}}\mathbf{B}^* = \tilde{\mathbf{0}}^N$ is equivalent to the condition $\mathbf{B}^H\mathbf{w} = \mathbf{0}_N$ for $\mathbf{w} = (\tilde{\mathbf{v}})^T$. In other words, the left null space $\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)$ of $\mathbf{B}$ is isomorphic to the right null space $\mathcal{N}_{\text{r}}\big(\mathbf{B}^H\big)$ of $\mathbf{B}^H$, i.e. $\mathcal{N}_{\text{l}}\big(\mathbf{B}\big) \div \mathcal{N}_{\text{r}}\big(\mathbf{B}^H\big)$ (also see the remark made at the end of Definition C.2.2). Note the following equality:

$$\begin{aligned}\mathcal{N}_{\text{l}}\big(\mathbf{B}\big) &\triangleq \{\tilde{\mathbf{v}} \in \mathbb{C}^M \mid \tilde{\mathbf{v}}\,(\mathbf{b}^j)^* = <\tilde{\mathbf{v}}, \mathbf{b}^j> = 0 \quad \forall\, 1 \le j \le N\} \\ &= \{\tilde{\mathbf{v}} \in \mathbb{C}^M \mid \tilde{\mathbf{v}} \perp \mathbf{b}^j = 0 \quad \forall\, 1 \le j \le N\}\,,\end{aligned} \tag{C.2.10}$$

where again the definition of the inner product function $\langle\cdot,\cdot\rangle$ is given by (C.1.3). Hence, $\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)$ consists of all vectors $\tilde{\mathbf{v}} \in \mathbb{C}^M$ that are orthogonal to all columns of $\mathbf{B}$.

Based on definitions C.2.1 and C.2.2, we can now the refine informal Definition C.1.1 to the following formal definition:

Definition C.2.5. Rank. The *rank* $\text{rank}(\mathbf{B})$ of a matrix $\mathbf{B} \in \mathbb{C}_M^N$ equals the dimension of the column or row space of $\mathbf{B}$:

$$d(\mathbf{B}) \triangleq \text{rank}(\mathbf{B}) = \dim(\mathcal{R}_{\text{c}}(\mathbf{B})) = \dim(\mathcal{R}_{\text{r}}(\mathbf{B})) = \text{rank}(\mathbf{B}^H)\,. \tag{C.2.11}$$

For reference purposes we summarize the four fundamental subspaces of $\mathbf{B} \in \mathbb{C}_M^N$ and their dimensions:

Column space $\mathcal{R}_{\text{c}}(\mathbf{B}) \in \mathbb{C}_M$:	$\dim(\mathcal{R}_{\text{c}}(\mathbf{B})) = \text{rank}(\mathbf{B})$;
Row space $\mathcal{R}_{\text{r}}(\mathbf{B}) \in \mathbb{C}^N$:	$\dim(\mathcal{R}_{\text{r}}(\mathbf{B})) = \text{rank}(\mathbf{B})$;
Null space $\mathcal{N}_{\text{r}}\big(\mathbf{B}\big) \in \mathbb{C}_N$:	$\dim(\mathcal{N}_{\text{r}}\big(\mathbf{B}\big)) = N - \text{rank}(\mathbf{B})$;
Left null space $\mathcal{N}_{\text{l}}\big(\mathbf{B}\big) \in \mathbb{C}^M$:	$\dim(\mathcal{N}_{\text{l}}\big(\mathbf{B}\big)) = M - \text{rank}(\mathbf{B})$.

C.2.2 Properties

The four fundamental subspaces defined in the previous section are closely related by certain orthogonality relations. We summarize these important relations using among other things the symbol '$\doteqdot$' to denote *isomorphism*:

FSP1: *The column space is isomorphic to the orthogonal complement of the left null space:*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right) \doteqdot \left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{B}\big)\right)^{\perp} \quad \text{or} \quad \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right)\right)^{\perp} \doteqdot \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}\big)\,;} \tag{C.2.12}$$

FSP2: *The row space is isomorphic to the orthogonal complement of the (right) null space:*

$$\boxed{\mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right) \doteqdot \left(\mathcal{N}_{\mathrm{r}}\big(\mathbf{B}\big)\right)^{\perp} \quad \text{or} \quad \left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right)\right)^{\perp} \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{B}\big)\,;} \tag{C.2.13}$$

FSP3: *The column space is isomorphic to the row space of the Hermitian:*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right) \doteqdot \mathcal{R}_{\mathrm{r}}\left(\mathbf{B}^{H}\right) \quad \text{or} \quad \mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right) \doteqdot \mathcal{R}_{\mathrm{c}}\left(\mathbf{B}^{H}\right)\,;} \tag{C.2.14}$$

FSP4: *The (right) null space is isomorphic to the left null space of the Hermitian:*

$$\boxed{\mathcal{N}_{\mathrm{r}}\big(\mathbf{B}\big) \doteqdot \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}^{H}\big) \quad \text{or} \quad \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}\big) \doteqdot \mathcal{N}_{\mathrm{r}}\big(\mathbf{B}^{H}\big)\,;} \tag{C.2.15}$$

FSP5: *The column space equals the orthogonal complement of the (right) null space of the Hermitian:*

$$\boxed{\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right) = \left(\mathcal{N}_{\mathrm{r}}\big(\mathbf{B}^{H}\big)\right)^{\perp} \quad \text{or} \quad \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right)\right)^{\perp} = \mathcal{N}_{\mathrm{r}}\big(\mathbf{B}^{H}\big)\,;} \tag{C.2.16}$$

FSP6: *The row space equals the orthogonal complement of the left null space of the Hermitian:*

$$\boxed{\mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right) = \left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{B}^{H}\big)\right)^{\perp} \quad \text{or} \quad \left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right)\right)^{\perp} = \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}^{H}\big)\,.} \tag{C.2.17}$$

From these properties and property *OCP6* on page 448 it follows that for any matrix $\mathbf{B} \in \mathbb{C}_{M}^{N}$ the vector spaces $\mathbb{C}_{M}$ and $\mathbb{C}^{N}$ can be decomposed as follows:

$$\mathbb{C}_{M} = \mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right) \oplus \left(\mathcal{N}_{\mathrm{l}}\big(\mathbf{B}\big)\right)^{T}; \tag{C.2.18}$$

$$\mathbb{C}^{N} = \mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right) \oplus \left(\mathcal{N}_{\mathrm{r}}\big(\mathbf{B}\big)\right)^{T}. \tag{C.2.19}$$

Equivalently (or, 'isomorphically'), the vector spaces $\mathbb{C}^{M}$ and $\mathbb{C}_{N}$ can be decomposed as follows:

$$\mathbb{C}^{M} = \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{B}\right)\right)^{T} \oplus \mathcal{N}_{\mathrm{l}}\big(\mathbf{B}\big)\,; \tag{C.2.20}$$

$$\mathbb{C}_{N} = \left(\mathcal{R}_{\mathrm{r}}\left(\mathbf{B}\right)\right)^{T} \oplus \mathcal{N}_{\mathrm{r}}\big(\mathbf{B}\big)\,. \tag{C.2.21}$$

APPENDIX D

Singular and Eigen Value Decompositions

In this section, we briefly recapitulate the standard Singular Value Decomposition (SVD) theorem and express it in our own notation, along with some clarifying examples. Since the theorem is a well-known result in linear algebra, for proofs we refer the reader to standard textbooks on this topic, e.g. see [72]. In addition to the standard matrix representation of the SVD theorem, we also describe its 'functional' equivalent because of its importance for the results in this thesis.

D.1 SVD in standard matrix notation

We first formulate the standard 'matrix-SVD' theorem and then examine its contents.

Theorem D.1.1. SVD of matrix.
Let $\mathbf{C} \in \mathbb{C}_M^N$ be a matrix with rank $r \triangleq \mathrm{rank}(\mathbf{C})$. Then, there exist a 'diagonal' matrix $\mathbf{\Sigma} = \mathrm{diag}\{\sigma_1, \ldots, \sigma_P\}$ whose entries are real-valued and positive, and unitary matrices $\mathbf{U} \in \mathbb{C}_M^M$ and $\mathbf{V} \in \mathbb{C}_N^N$, such that $\mathbf{C}$ can be decomposed as follows:

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}(\mathbf{V})^*,$$

where:

- $P = \min(M, N)$;
- $\mathbf{U}^H\mathbf{U} = \mathbf{U}\mathbf{U}^H = \mathbf{I}_M^M$;
- $\mathbf{V}^H\mathbf{V} = \mathbf{V}\mathbf{V}^H = \mathbf{I}_N^N$;
- $\sigma_1 \geq \cdots \geq \sigma_r > 0,\ \sigma_{r+1}, \ldots, \sigma_P = 0$.

The columns of $\mathbf{U}$ are called the *left singular vectors* of $\mathbf{C}$. Likewise, the rows of $\mathbf{V}$ are called the *right singular vectors* of $\mathbf{C}$. As stated in the theorem, the diagonal elements of $\mathbf{\Sigma}$ are real-valued and positive. They are called the *singular values* of $\mathbf{C}$ and are ordered in such a way that $\sigma_1 \geq \cdots \geq \sigma_P \geq 0$. Since $\mathrm{rank}(\mathbf{C}) = r$, only the first r singular values are non-zero. Hence, $\sigma_{r+1}, \ldots, \sigma_P = 0$. Although the matrix $\mathbf{\Sigma}$ is not a square matrix, notationally it is written as $\mathrm{diag}\{\sigma_1, \ldots, \sigma_P\}$, and its off-diagonal elements are zero. The singular values $\sigma_1 \geq \cdots \geq \sigma_P$ are written along the main diagonal, and rows or columns of zeros are appended as necessary to obtain the proper dimensions. As an example, consider the following real-valued matrix with $M = 3$ rows, $N = 4$ columns, and rank $r = 2$:

$$\mathbf{C} = \begin{bmatrix} -2.6296 & 4.3493 & 0.8462 & 4.3311 \\ 1.8464 & -1.1257 & 1.1587 & -0.0612 \\ 0.8667 & -2.9928 & -1.6964 & -3.8373 \end{bmatrix} \tag{D.1.1}$$

Then the SVD of $\mathbf{C}$ is given by:

$$\mathbf{C} = \underbrace{\begin{bmatrix} -0.7898 & 0.2207 & 0.5722 \\ 0.1373 & -0.8457 & 0.5157 \\ 0.5978 & 0.4858 & 0.6377 \end{bmatrix}}_{\mathbf{U}} \cdot \underbrace{\begin{bmatrix} 8.4923 & 0 & 0 & 0 \\ 0 & 2.5535 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{\Sigma}} \cdot \underbrace{\begin{bmatrix} 0.3354 & -0.6739 & 0.5583 & -0.3487 \\ -0.6334 & 0.1794 & 0.1345 & -0.7407 \\ -0.1794 & -0.6334 & -0.7407 & -0.1345 \\ -0.6739 & -0.3354 & 0.3487 & 0.5583 \end{bmatrix}}_{\mathbf{V}} . \tag{D.1.2}$$

Note that $P = 3$, $\mathbf{\Sigma} = \mathrm{diag}\{\sigma_1, \sigma_2, \sigma_3\}$ with $\sigma_1 = 8.4923, \sigma_2 = 2.5535, \sigma_3 = 0$.

For convenience and insight, the matrices in the SVD are usually split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. Since this partitioning is particularly relevant for the subspace techniques that are heavily used in this thesis, we explain the associated notation in some detail. Let $\mathbf{\Sigma}_1^1$ represent the square diagonal block of $\mathbf{\Sigma}$ that contains the non-zero singular values, i.e. $\mathbf{\Sigma}_1^1$ is the upper left block of $\mathbf{\Sigma}$ of size $r \times r$. Furthermore, let $\mathbf{\Sigma}_2^2$ represent the (not necessarily square) block of zeros of $\mathbf{\Sigma}$ to the right and below $\mathbf{\Sigma}_1^1$, i.e. $\mathbf{\Sigma}_2^2$ is the lower right block of $\mathbf{\Sigma}$ of size $(M-r) \times (N-r)$. Thus, we have:

$$\mathbf{\Sigma}_1^1 \triangleq \mathrm{diag}\{\sigma_1, \ldots, \sigma_r\} \in \mathbb{R}_r^r \qquad \text{and} \qquad \mathbf{\Sigma}_2^2 \triangleq \mathbf{0}_{M-r}^{N-r} , \tag{D.1.3}$$

and $\mathbf{\Sigma}$ can be written as follows:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1^1 & \mathbf{0}_r^{N-r} \\ \mathbf{0}_{M-r}^r & \mathbf{\Sigma}_2^2 \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma}_1^1 & \mathbf{0}_r^{N-r} \\ \mathbf{0}_{M-r}^r & \mathbf{0}_{M-r}^{N-r} \end{bmatrix} . \tag{D.1.4}$$

For the example in (D.1.2) we have:

$$\mathbf{\Sigma}_1^1 = \begin{bmatrix} 8.4923 & 0 \\ 0 & 2.5535 \end{bmatrix} \qquad \text{and} \qquad \mathbf{\Sigma}_2^2 = \begin{bmatrix} 0 & 0 \end{bmatrix} .$$

The matrix $\mathbf{U}$ is split into two parts as $\mathbf{U} = \begin{bmatrix} \mathbf{U}^1 & \mathbf{U}^2 \end{bmatrix}$. The first part $\mathbf{U}^1$ consists of columns that are multiplied by the non-zero singular values in the SVD-expression of $\mathbf{C}$, whereas the second part $\mathbf{U}^2$ consists of columns that are multiplied by the zero singular values. Hence:

$$\mathbf{U}^1 \triangleq \begin{bmatrix} \mathbf{u}^1 \cdots \mathbf{u}^r \end{bmatrix} \in \mathbb{C}_M^r \quad \text{and} \quad \mathbf{U}^2 \triangleq \begin{bmatrix} \mathbf{u}^{r+1} \cdots \mathbf{u}^M \end{bmatrix} \in \mathbb{C}_M^{M-r}, \tag{D.1.5}$$

where $\mathbf{u}^p$ is the p-th column of $\mathbf{U}$. Likewise, the matrix $\mathbf{V}$ is split as $\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix}$, with

$$\mathbf{V}_1 \triangleq \begin{bmatrix} \tilde{\mathbf{v}}_1 \\ \vdots \\ \tilde{\mathbf{v}}_r \end{bmatrix} \in \mathbb{C}_r^N \quad \text{and} \quad \mathbf{V}_2 \triangleq \begin{bmatrix} \tilde{\mathbf{v}}_{r+1} \\ \vdots \\ \tilde{\mathbf{v}}_N \end{bmatrix} \in \mathbb{C}_{N-r}^N , \tag{D.1.6}$$

where $\tilde{\mathbf{v}}_p$ is the p-th row of $\mathbf{V}$. Now, the SVD in Theorem D.1.1 can be written as follows:

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}(\mathbf{V})^* = \begin{bmatrix} \mathbf{U}^1 & \mathbf{U}^2 \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_1^1 & \mathbf{0}_r^{N-r} \\ \mathbf{0}_{M-r}^r & \mathbf{0}_{M-r}^{N-r} \end{bmatrix} \begin{bmatrix} (\mathbf{V}_1)^* \\ (\mathbf{V}_2)^* \end{bmatrix} = \mathbf{U}^1 \mathbf{\Sigma}_1^1 (\mathbf{V}_1)^* = \sum_{p=1}^{r} \sigma_p \mathbf{u}^p (\tilde{\mathbf{v}}_p)^* . \tag{D.1.7}$$

For the example in (D.1.2), we have:

$$\mathbf{U}^1 = \begin{bmatrix} -0.7898 & 0.2207 \\ 0.1373 & -0.8457 \\ 0.5978 & 0.4858 \end{bmatrix}, \quad \mathbf{U}^2 = \begin{bmatrix} 0.5722 \\ 0.5157 \\ 0.6377 \end{bmatrix},$$

and

$$\mathbf{V}_1 = \begin{bmatrix} 0.3354 & -0.6739 & 0.5583 & -0.3487 \\ -0.6334 & 0.1794 & 0.1345 & -0.7407 \end{bmatrix},$$
$$\mathbf{V}_2 = \begin{bmatrix} -0.1794 & -0.6334 & -0.7407 & -0.1345 \\ -0.6739 & -0.3354 & 0.3487 & 0.5583 \end{bmatrix}.$$

Since the relations between the fundamental subspaces of a matrix and the various parts of its Singular Value Decomposition in (D.1.7) are of paramount importance for this work, whose rationale to a large extent is based on the subspace approach, we now formulate these relations in mathematical terms. First note that due to (D.1.7) the i-th row of $\mathbf{C}$ can be expressed as follows:

$$\tilde{\mathbf{c}}_i = \sum_{p=1}^{r} \sigma_p u_i^p (\tilde{\mathbf{v}}_p)^* ,$$

where u_i^p is the i-th element of the column vector $\mathbf{u}^p$. Likewise, the j-the column is expressed as:

$$\mathbf{c}^j = \sum_{p=1}^{r} \sigma_p (v_p^j)^* \mathbf{u}^p ,$$

where v_p^j is the j-th element of the row vector $\tilde{\mathbf{v}}_P$. Writing $\mathbf{C} \in \mathbb{C}_M^N$ in terms of its columns and rows as follows:

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}^1 & \cdots & \mathbf{c}^N \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{c}}_1 \\ \vdots \\ \tilde{\mathbf{c}}_M \end{bmatrix},$$

we can derive the following relations between the various linear spaces:

$$\begin{aligned}
&\mathcal{R}_{\mathrm{c}}(\mathbf{C}) = \mathcal{L}\left(\mathbf{c}^1, \ldots, \mathbf{c}^N\right) = \mathcal{L}\left(\mathbf{u}^1, \ldots, \mathbf{u}^r\right) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^1\right); \\
&\mathcal{R}_{\mathrm{r}}(\mathbf{C}) = \mathcal{L}\left((\tilde{\mathbf{c}}_1)^*, \ldots, (\tilde{\mathbf{c}}_M)^*\right) = \mathcal{L}\left(\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_r\right) = \mathcal{R}_{\mathrm{r}}\big((\mathbf{V}_1)^*\big); \\
&\mathcal{N}_{\mathrm{r}}\big(\mathbf{C}\big) = \mathcal{L}\left((\tilde{\mathbf{v}}_{r+1})^T, \ldots, (\tilde{\mathbf{v}}_N)^T\right) = \mathcal{R}_{\mathrm{c}}\left((\mathbf{V}_2)^T\right); \\
&\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}\big) = \mathcal{L}\left((\mathbf{u}^{r+1})^T, \ldots, (\mathbf{u}^M)^T\right) = \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^2\right)\right)^T.
\end{aligned}$$

Summarizing, we have:

$$\mathcal{R}_{\mathrm{c}}(\mathbf{C}) = \mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^1\right); \tag{D.1.8}$$
$$\mathcal{R}_{\mathrm{r}}(\mathbf{C}) = \mathcal{R}_{\mathrm{r}}\big((\mathbf{V}_1)^*\big); \tag{D.1.9}$$
$$\mathcal{N}_{\mathrm{r}}\big(\mathbf{C}\big) = \mathcal{R}_{\mathrm{c}}\left((\mathbf{V}_2)^T\right); \tag{D.1.10}$$
$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{C}\big) = \left(\mathcal{R}_{\mathrm{c}}\left(\mathbf{U}^2\right)\right)^T. \tag{D.1.11}$$

D.2 SVD in functional notation

As we have shown in Section 1.4, we can always define a bijective mapping between a function $f[\mathbf{n}] \in \mathbb{C}[\boldsymbol{\mathcal{T}}]$ $(\mathbb{R}[\boldsymbol{\mathcal{T}}])$ defined on some discrete possibly multi-dimensional Region Of Support (ROS) $\boldsymbol{\mathcal{T}} \triangleq \{\mathbf{n}_1, \ldots, \mathbf{n}_N\}$ with cardinality $N \triangleq |\boldsymbol{\mathcal{T}}|$, and the row vector given by:

$$\tilde{\mathbf{f}} \triangleq \begin{bmatrix} f[\mathbf{n}_1] & \cdots & f[\mathbf{n}_N] \end{bmatrix} \in \mathbb{C}^N \, (\mathbb{R}^N) \, . \tag{D.2.1}$$

Likewise, we can define a bijective mapping between a function-valued vector $\mathbf{f}[\mathbf{n}] \triangleq \begin{bmatrix} f_1[\mathbf{n}] & \cdots & f_M[\mathbf{n}] \end{bmatrix}^T \in \mathbb{C}_M[\boldsymbol{\mathcal{T}}]$ $(\mathbb{R}_M[\boldsymbol{\mathcal{T}}])$ and the matrix $\mathbf{F} \in \mathbb{C}_M^N$ $(\mathbb{R}_M^N)$ given by:

$$\mathbf{F} \triangleq \begin{bmatrix} \tilde{\mathbf{f}}_1 \\ \vdots \\ \tilde{\mathbf{f}}_M \end{bmatrix} = \begin{bmatrix} \mathbf{f}[\mathbf{n}_1] & \cdots & \mathbf{f}[\mathbf{n}_N] \end{bmatrix} \in \mathbb{C}_M^N \, (\mathbb{R}_M^N) \, .$$

Because of these bijective mappings, we can formulate an SVD theorem for function-valued vectors, which will be done in the sequel. For convenience, we first consider functions and function-valued vectors that depend on a univariate argument $n \in \mathcal{T}$, where $\mathcal{T}$ is a set $\mathcal{T} \triangleq \{n_1, \ldots, n_N\}$ with cardinality $N \triangleq |\mathcal{T}|$. Then, the obtained results are briefly extended to multi-variable functions.

Consider the function-valued vector $\mathbf{f}[n] \triangleq \begin{bmatrix} f_1[n] & \cdots & f_M[n] \end{bmatrix}^T \in \mathbb{C}_M[\mathcal{T}]$ $(\mathbb{R}_M[\mathcal{T}])$ whose elements $f_1[n], \ldots, f_M[n]$ belong to $\mathbb{C}[\mathcal{T}]$ $(\mathbb{R}[\mathcal{T}])$ and span a linear vector space of dimension r. If for clarity we want to denote the cardinality of $\mathcal{T}$ explicitly in the employed notation, we can write $\mathcal{T}^N$ instead of $\mathcal{T}$. Then we say that $\mathbf{f}[n] \in \mathbb{C}_M[\mathcal{T}^N]$ $(\mathbb{R}_M[\mathcal{T}^N])$ and it is more clear that it can be associated bijectively with a matrix $\mathbf{F} \in \mathbb{C}_M^N$ $(\mathbb{R}_M^N)$ having the appropriate size. Note that due to the bijective mapping, we can also speak of the 'rank' of $\mathbf{f}[n]$, which may be defined as follows:

$$r \triangleq \operatorname{rank}\left(\mathbf{f}[n]\right) = \dim\left(\mathcal{L}\left(\left\{f_p[n]\right\}_{1 \leq p \leq M}\right)\right) = \dim\left(\mathcal{L}\left(\left\{\mathbf{f}[n_p]\right\}_{1 \leq p \leq N}\right)\right) . \tag{D.2.2}$$

In order to be able to define the SVD of the function-valued vector $\mathbf{f}[n] \in \mathbb{C}_M[\mathcal{T}^N]$ $(\mathbb{R}_M[\mathcal{T}^N])$, two different inner product functions need to be defined. We only give the complex-valued versions because the corresponding real-valued versions can be obtained by omitting the conjugations. The first inner product is a function from $\mathbb{C}_P \times \mathbb{C}_P$ to $\mathbb{C}$ and is defined by:

$$\boxed{\langle \mathbf{v}, \mathbf{w} \rangle_{\mathrm{v}} \triangleq \sum_{p=1}^{P} v_p (w_p)^* \, ,} \tag{D.2.3}$$

where the 'v' in the subscript position indicates that this inner product is of the '<u>v</u>ector type'. The second inner product is a function from $\mathbb{C}[\mathcal{T}] \times \mathbb{C}[\mathcal{T}]$ to $\mathbb{C}$ and is defined by:

$$\boxed{\langle v[n], w[n] \rangle_{\mathrm{f}} \triangleq \sum_{n \in \mathcal{T}} v[n] (w[n])^* \, ,} \tag{D.2.4}$$

where the 'f' in the subscript position indicates that this inner product is of the '<u>f</u>unctional type'. For multi-variate functions, this definition generalizes easily to a function from $\mathbb{C}[\boldsymbol{\mathcal{T}}] \times \mathbb{C}[\boldsymbol{\mathcal{T}}]$ to $\mathbb{C}$ as follows:

$$\boxed{\langle v[\mathbf{n}], w[\mathbf{n}] \rangle_{\mathrm{f}} \triangleq \sum_{\mathbf{n} \in \boldsymbol{\mathcal{T}}} v[\mathbf{n}] (w[\mathbf{n}])^* \, .} \tag{D.2.5}$$

The following theorem presents the functional singular value decomposition of $\mathbf{f}[n] \in \mathbb{C}_M[\mathcal{T}^N]$ and is the functional equivalent of Theorem D.1.1:

Theorem D.2.1. SVD of univariate function-valued vector.
Let $\mathbf{f}[n] \in \mathbb{C}_M[\mathcal{T}^N]$ be a univariate function-valued vector with rank $r \triangleq \operatorname{rank}\big(\mathbf{f}[n]\big)$. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, vectors $\mathbf{u}^p \in \mathbb{C}_M$ for $1 \leq p \leq M$ and functions $v_p[n] \in \mathbb{C}[\mathcal{T}^N]$ for $1 \leq p \leq N$, such that $\mathbf{f}[n]$ can be decomposed as follows:

$$\mathbf{f}[n] = \sum_{p=1}^{P} \sigma_p \mathbf{u}^p (v_p[n])^* \qquad \forall\, n \in \mathcal{T}\,,$$

where:

- $P = \min(M, N)\,;$
- $\big\langle \mathbf{u}^i, \mathbf{u}^j \big\rangle_{\mathrm{v}} = \delta^{ij} \quad \forall\, 1 \leq i, j \leq M\,;$
- $\langle v_i[n], v_j[n] \rangle_{\mathrm{f}} = \delta_{ij} \quad \forall\, 1 \leq i, j \leq N\,;$
- $\sigma_1 \geq \cdots \geq \sigma_r > 0,\ \sigma_{r+1}, \ldots, \sigma_P = 0\,.$

Proof. The theorem can be proven by using the standard SVD theorem described in the previous section, and the bijective mappings between functions and row vectors on the one hand, and function-valued vectors and matrices on the other. □

By analogy with Theorem D.1.1, the columns $\mathbf{u}^1, \ldots, \mathbf{u}^M$ are called the *left singular vectors* of $\mathbf{f}[n]$. Likewise, we call the functions $v_1[n], \ldots, v_N[n]$ the *right singular functions* of $\mathbf{f}[n]$.

The following corollary directly follows from the theorem:

Corollary D.2.2. Reduced SVD of univariate function-valued vector.
Under the assumptions formulated in Theorem D.2.1, $\mathbf{f}[n]$ can be written in the following reduced form:

$$\mathbf{f}[n] = \sum_{p=1}^{r} \sigma_p \mathbf{u}^p (v_p[n])^* \qquad \forall\, n \in \mathcal{T}\,.$$

Theorem D.2.1 can be generalized to multi-variable function-valued vectors as follows:

Theorem D.2.3. SVD of multi-variate function-valued vector.
Let $\mathbf{f}[\mathbf{n}] \in \mathbb{C}_M[\boldsymbol{\mathcal{T}}^N]$ be a multivariate function-valued vector with rank $r \triangleq \operatorname{rank}\big(\mathbf{f}[\mathbf{n}]\big)$. Then, there exist real-valued positive constants $\sigma_p \in \mathbb{R}$ for $1 \leq p \leq P$, vectors $\mathbf{u}^p \in \mathbb{C}_M$ for $1 \leq p \leq M$ and functions $v_p[\mathbf{n}] \in \mathbb{C}[\boldsymbol{\mathcal{T}}^N]$ for $1 \leq p \leq N$, such that $\mathbf{f}[\mathbf{n}]$ can be decomposed as follows:

$$\mathbf{f}[\mathbf{n}] = \sum_{p=1}^{P} \sigma_p \mathbf{u}^p (v_p[\mathbf{n}])^* \qquad \forall\, \mathbf{n} \in \boldsymbol{\mathcal{T}}\,,$$

where:

- $P = \min(M, N)\,;$
- $\big\langle \mathbf{u}^i, \mathbf{u}^j \big\rangle_{\mathrm{v}} = \delta^{ij} \quad \forall\, 1 \leq i, j \leq M\,;$

- $\langle v_i[\mathbf{n}], v_j[\mathbf{n}] \rangle_{\mathrm{f}} = \delta_{ij} \quad \forall\, 1 \leq i, j \leq N\,;$
- $\sigma_1 \geq \cdots \geq \sigma_r > 0,\ \sigma_{r+1}, \ldots, \sigma_P = 0\,.$

Again, the columns $\mathbf{u}^1, \ldots, \mathbf{u}^M$ and functions $v_1[n], \ldots, v_N[n]$ are called the left singular vectors and right singular functions respectively of $\mathbf{f}[\mathbf{n}]$.

The following corollary directly follows from Theorem D.2.3:

Corollary D.2.4. Reduced SVD of multi-variate function-valued vector.
Under the assumptions formulated in Theorem D.2.3, $\mathbf{f}[\mathbf{n}]$ *can be written in the following reduced form:*

$$\mathbf{f}[\mathbf{n}] = \sum_{p=1}^{r} \sigma_p \mathbf{u}^p (v_p[\mathbf{n}])^* \qquad \forall\, \mathbf{n} \in \boldsymbol{\mathcal{T}}\,.$$

Note from this corollary that the i-th function of $\mathbf{f}[\mathbf{n}]$ can be expressed as follows:

$$f_i[\mathbf{n}] = \sum_{p=1}^{r} \sigma_p u_i^p (v_p[\mathbf{n}])^*\,.$$

Similarly to the previous section, for convenience and insight the set of vectors and the set of functions in the functional SVD are split into two parts, one of which corresponds to the non-zero singular values and the other to the zero singular values. Firstly, we define the set $\mathcal{U}$ containing the left singular vectors $\mathbf{u}^1, \ldots, \mathbf{u}^M$ of the decomposition as follows:

$$\mathcal{U} \triangleq \{\mathbf{u}^1, \ldots, \mathbf{u}^M\}\,.$$

This set is split into the sets $\mathcal{U}^1$ and $\mathcal{U}^2$, where

$$\mathcal{U}^1 \triangleq \{\mathbf{u}^1, \ldots, \mathbf{u}^r\}$$

contains the left singular vectors corresponding to the non-zero singular values, and

$$\mathcal{U}^2 \triangleq \{\mathbf{u}^{r+1}, \ldots, \mathbf{u}^M\}$$

contains the left singular vectors corresponding to the zero singular values. Likewise, define the set $\mathcal{V}$ containing the right singular functions $v_1[\mathbf{n}], \ldots, v_N[\mathbf{n}]$ of the decomposition as follows:

$$\mathcal{V} \triangleq \big\{v_1[\mathbf{n}], \ldots, v_N[\mathbf{n}]\big\}\,.$$

This set is split into the sets $\mathcal{V}_1$ and $\mathcal{V}_2$, where

$$\mathcal{V}_1 \triangleq \big\{v_1[\mathbf{n}], \ldots, v_r[\mathbf{n}]\big\}$$

contains the right singular functions corresponding to the non-zero singular values, and

$$\mathcal{V}_2 \triangleq \big\{v_{r+1}[\mathbf{n}], \ldots, v_N[\mathbf{n}]\big\}$$

contains the right singular functions corresponding to the zero singular values. Now splitting the SVD expression in Theorem D.2.3 accordingly yields:

$$\mathbf{f}[\mathbf{n}] = \sum_{p=1}^{P} \sigma_p \mathbf{u}^p (v_p[\mathbf{n}])^* = \sum_{p=1}^{r} \sigma_p \mathbf{u}^p (v_p[\mathbf{n}])^* + \sum_{p=r+1}^{P} \sigma_p \mathbf{u}^p (v_p[\mathbf{n}])^* \qquad \forall\, \mathbf{n} \in \boldsymbol{\mathcal{T}}\,,$$

where the first summation contains only vectors and functions from $\mathcal{U}^1$ and $\mathcal{V}_1$ respectively, and the second contains only vectors and functions from $\mathcal{U}^2$ and $\mathcal{V}_2$ respectively.

By analogy with (D.1.8)–(D.1.11) of the previous section, we formulate the following relations between the fundamental subspaces of a function-valued vector and the various parts of its 'split' functional Singular Value Decomposition:

$$\mathcal{L}\left(\mathbf{f}[\mathbf{n}]\right) = \mathcal{L}\left(\mathcal{U}^1\right) = \mathcal{L}\left(\mathbf{u}^1, \ldots, \mathbf{u}^r\right) \; ; \tag{D.2.6}$$

$$\mathcal{L}_r\big(\mathbf{f}[\mathbf{n}]\big) = \mathcal{L}\big(\mathcal{V}_1\big) \triangleq \mathcal{L}\big(v_1[\mathbf{n}], \ldots, v_r[\mathbf{n}]\big) \, ; \tag{D.2.7}$$

$$\mathcal{N}_{\mathrm{r}}\big(\mathbf{f}[\mathbf{n}]\big) = \mathcal{L}\big(\mathcal{V}_2\big) \triangleq \mathcal{L}\big(v_{r+1}[\mathbf{n}], \ldots, v_N[\mathbf{n}]\big) \, ; \tag{D.2.8}$$

$$\mathcal{N}_{\mathrm{l}}\big(\mathbf{f}[\mathbf{n}]\big) = \left(\mathcal{L}\left(\mathcal{U}^2\right)\right)^T \triangleq \mathcal{L}\left((\mathbf{u}^{r+1})^T, \ldots, (\mathbf{u}^M)^T\right) \, , \tag{D.2.9}$$

where $\mathcal{L}_r(\cdot)$ denotes the linear span of the conjugates of the functions in the argument of te function-valued column vector.

D.3 Generalized Eigenvalue Decomposition

In this section we briefly recall the basic theory about the GEVD of two matrices $\mathbf{B}$ and $\mathbf{C}$ of size $D \times D$. The GEVD of $\mathbf{B}$ and $\mathbf{C}$ involves the determination of the set of generalized eigenvectors $\{\mathbf{u}_k\}_{1 \leq k \leq D}$ and corresponding (generalized) eigenvalues $\{\sigma_k\}_{1 \leq k \leq D}$ that satisfy the following equation:

$$\mathbf{B}\mathbf{u}_k = \sigma_k \mathbf{C}\,\mathbf{u}_k \quad \forall\, 1 \leq k \leq D \qquad \text{or} \qquad (\mathbf{B} - \sigma_k \mathbf{C})\mathbf{u}_k = \mathbf{0} \quad \forall\, 1 \leq k \leq D. \tag{D.3.1}$$

The set of matrices $\{\mathbf{B} - \sigma\mathbf{C} \mid \sigma \in \mathbb{R}\}$ is said to form a *matrix pencil*. The eigenvalues of the pencil, denoted by $\sigma(\mathbf{B}, \mathbf{C})$, are those values of σ for which $\det(\mathbf{B} - \sigma\mathbf{C}) = 0$. For an eigenvalue $\sigma \in \sigma(\mathbf{B}, \mathbf{C})$ of the pencil, a vector $\mathbf{u} \neq \mathbf{0}$ such that $\mathbf{B}\mathbf{u} = \sigma\mathbf{C}\,\mathbf{u}$ is said to be an (generalized) eigenvector of the pencil. Obviously, when $\mathbf{C}$ is nonsingular $\sigma(\mathbf{B}, \mathbf{C}) = \sigma(\mathbf{C}^{-1}\mathbf{B})$, where $\sigma(\mathbf{C}^{-1}\mathbf{B})$ denotes the set of (standard) eigenvalues of $\mathbf{C}^{-1}\mathbf{B}$. By defining $\mathbf{U} \triangleq [\mathbf{u}_1 \cdots \mathbf{u}_D]$ and $\mathbf{\Sigma} \triangleq \mathrm{diag}(\sigma_1, \ldots, \sigma_D)$, it follows from (D.3.1) that the GEVD of $\mathbf{B}$ and $\mathbf{C}$ can be written as:

$$\mathbf{B}\mathbf{U} = \mathbf{C}\mathbf{U}\mathbf{\Sigma}. \tag{D.3.2}$$

Note that in general $\mathbf{U}$ is not orthogonal. The GEVD of $\mathbf{B}$ and $\mathbf{C}$ can be computed using standard tools from linear algebra (for example, in Matlab the function `eig` can be used). In functional form, we can write the results as $[\mathbf{U}, \mathbf{\Sigma}] = \mathrm{gevd}(\mathbf{B}, \mathbf{C})$. It can be shown that under mild conditions, the solution is unique [72] up to some indeterminacies similar to the ones inherent to MIBI (see Section 2.4).

APPENDIX E

Homotopy methods for solving nonlinear systems

Homotopy methods provide a deterministic means for solving a system of nonlinear equations by smoothly deforming the known solutions of a simple start system into the desired solutions of the target system [2, 144]. They are based on the so-called *path following* or *continuation* techniques. In this appendix we give an overview of the basic principles of homotopy methods and summarize their essentials. Excellent discussions can be found in several articles and books [2, 105, 116, 144]. In the context of homotopy methods, the system of equations to be solved is commonly called *target system*. We will denote a general target system by $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ and assume that $\mathbf{p} : \mathbb{C}_P \rightarrow \mathbb{C}_P$. Furthermore, its solution set is denoted by $\mathcal{P}$, i.e. $\mathcal{P} \triangleq \{\mathbf{z}_p \in \mathbb{C}_P \mid \mathbf{p}(\mathbf{z}_p) = \mathbf{0}\}$. The rationale behind homotopy methods is to deform the known solutions of a known and simple *start system*, which in a sense is compatible with the target system, into the solutions of the target system. The start and target systems are embedded in a family of systems, the homotopy, and then all members in the family are solved in a sequential and iterative manner. In the sequel, we will denote the start system by $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, where $\mathbf{g} : \mathbb{C}_P \rightarrow \mathbb{C}_P$. Furthermore, its solution set is denoted by $\mathcal{G}$, i.e. $\mathcal{G} \triangleq \{\mathbf{z}_g \in \mathbb{C}_P \mid \mathbf{g}(\mathbf{z}_g) = \mathbf{0}\}$. Hence, a homotopy continuation procedure starts with a simple system of equations of which the solutions are known, i.e. the start system. Then this system is deformed slightly in the 'direction' of the target system. Since this new system is close to the previous system, under some mild conditions its solutions deviate only slightly from those of the previous system. Hence, each solution of the new system can be found by a local iterative optimization method that uses the corresponding solution of the previous system as its initial solution. For each solution of the start system, these steps are repeated until the target system is reached. In theory, under some mild uniqueness and smoothness assumptions each path converges to a geometrically isolated solution [2]. In the following sections we will describe homotopy methods in more detail and give some examples.

E.1 Basic principles of homotopy methods

Homotopy methods operate in two stages. Firstly, the (expected) number of solutions and the structure of the target system are exploited to construct the start system. Then, both systems are embedded in the *convex homotopy* defined as follows:

$$\mathbf{h}(\mathbf{z}, \lambda) = \gamma(1 - \lambda)\, \mathbf{g}(\mathbf{z}) + \lambda\, \mathbf{p}(\mathbf{z}) \qquad \forall\, \lambda \in [0, 1]\,, \tag{E.1.1}$$

where $\lambda \in \mathbb{R}$ is called the so-called *continuation parameter*, $\gamma \in \mathbb{C}$ is a randomly chosen fixed constant, and $\mathbf{h} : \mathbb{C}_P \times \mathbb{R} \rightarrow \mathbb{C}_P$. Secondly, for each solution $\mathbf{z}_g$ of the start system $\mathbf{g}(\mathbf{z})$, numerical path following or continuation methods attempt to trace the path/curve $(\mathbf{z}, \lambda) \in \mathbf{h}^{-1}(\mathbf{0})$ defined implicitly by $\mathbf{h}\big(\mathbf{z}, \lambda\big) = \mathbf{0}$ from the starting point $(\mathbf{z}_g, 0)$, which is a solution of the start system (note that $\mathbf{h}(\mathbf{z}, 0) = \gamma \mathbf{g}(\mathbf{z})$), to a solution point $(\mathbf{z}_p, 1)$ of the target system, where $\mathbf{p}(\mathbf{z}_p) = \mathbf{0}$ (note that $\mathbf{h}(\mathbf{z}, 1) = \mathbf{p}(\mathbf{z})$). For general nonlinear systems, various numerical problems may occur depending on the behavior of this curve. One

important problem occurs if the curve folds back (at a so-called turning point), i.e. if the values of λ decrease as the path progresses. In fact, this implies that the Jacobian matrix of $\mathbf{h}(\mathbf{z}, \lambda)$ is singular along the curve. This kind of difficulty is usually coped with by parameterizing the curve by a new parameter $s \in [0, s_e]$, e.g. the arc length, and tracking the curve $(\mathbf{z}(s), \lambda(s)) \in \mathbf{h}^{-1}(\mathbf{0})$ on $s \in [0, s_e]$ defined implicitly by $\mathbf{h}\big(\mathbf{z}(s), \lambda(s)\big) = \mathbf{0}$ from the starting point $(\mathbf{z}(0), \lambda(0)) = (\mathbf{z}_g, 0)$ to a solution point $(\mathbf{z}_p, 1)$ of the target system, where $\lambda(s_e) = 1$. Another important problem is the crossing of different solution paths, which can also occur for points on the curve(s) where the Jacobian is singular. It can be shown [116,144] that this problem is usually solved by the 'gamma trick', which is the incorporation of the randomly chosen constant γ in (E.1.1). For most systems the presence of this constant, which is often chosen as $\gamma = \exp(\jmath\theta)$ with θ picked at random from a uniform distribution on the interval $[0, 2\pi]$, avoids path crossing singularities with probability one. In [116, 144] it is also shown that for most polynomial systems employing the gamma trick the solution curve is parameterized well by the continuation parameter λ, i.e. it can described by the pair $(\mathbf{z}(\lambda), \lambda)$ for $\lambda \in [0, 1]$ without singularities or crossings along the curve. In fact, this means that the Jacobian matrix of $\mathbf{h}(\mathbf{z}, \lambda)$ is nonsingular along each solution curve. Since our main interest in this thesis is in polynomial and polyconjugal (i.e. polynomial-like, see Chapter 5 and 6) systems we assume that the homotopy curves arising in our problems are without singularities, and we only employ the curve parameterization with monotonically increasing continuation parameter λ (possibly along a path in the complex plane, see further). For more general problems that involve curves with turning points and require the use of an additional parameterization parameter, see [2] for example. Notwithstanding the aforesaid, simulations (see following sections) have revealed that for some types of polynomial problems the homotopy defined in (E.1.1) is not adequate because of path merging and crossing. We have solved the involved problems by constructing a new homotopy, viz. the one in (E.1.17), that is slightly more complicated and 'randomized' than the one in (E.1.1). After discussing a method for tracing the curves defined by (E.1.1), we briefly extend the results to this homotopy.

The main problem to be solved is to find the curve $(\mathbf{z}(\lambda), \lambda)$ that solves the following system as λ moves from 0 to 1:

$$\mathbf{h}\big(\mathbf{z}(\lambda), \lambda\big) = \mathbf{0} \qquad \forall\, \lambda \in [0, 1]\,. \tag{E.1.2}$$

Before explaining in detail how the curve defined by these equations can be found, we first highlight the basic rationale behind the operation of the homotopy method described in the remainder of this section. Given a certain point on the path, we first consider a single iteration of the method. Suppose that we have a solution $\mathbf{z}(\lambda)$ of $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}$ for a certain (known) value of λ. Then, this solution is an approximate solution of the slightly deformed system $\mathbf{h}(\mathbf{z}(\lambda + \Delta\lambda), \lambda + \Delta\lambda) = \mathbf{0}$, where $\Delta\lambda$ is a small increment. Then, the solution of the deformed system can be found in two steps. Firstly, in the *predictor* step an approximation of the solution $\mathbf{z}(\lambda + \Delta\lambda)$ of the deformed system is predicted, e.g. by using an Euler step (see further). Secondly, the predicted approximate solution is corrected to lie 'exactly' on the solution curve by applying a *corrector* step, which can be any local zero finding method such as Newton's method [2]. Performing these two steps for each solution of the start system and each iteration from $\lambda = 0$ to $\lambda = 1$, we finally end up at the solutions of the target system. Although several more sophisticated and optimized methods employing more optimal parameterizations (e.g. employing arc length) exist, for our purposes the above described method suffices. Now we will motivate and describe the method in more detail. Since the value of λ is imposed at each iteration, in the sequel by *solution curve* we mean the curve $\mathbf{z}(\lambda),\ \lambda \in [0, 1]$ defined by $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}\ \forall\ \lambda \in [0, 1]$. By computing the derivative of

$\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)=\mathbf{0}$ w.r.t. λ we observe that the curve $\mathbf{z}(\lambda)$ also satisfies the following system of ordinary differential equations (ODE's):

$$\frac{d}{d\lambda}\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)=\mathbf{0}\qquad\forall\,\lambda\in[0,1]\,, \tag{E.1.3}$$

which has to be solved separately for each of the initial conditions $\mathbf{z}(0)=\mathbf{z}_g\in\mathcal{G}$ in order to find all solutions. Given $\mathbf{z}(0)$, solving for $\mathbf{z}(\lambda)$ in this problem is a standard ODE initial value problem and is equivalent to solving (E.1.2). Hence, in principle the curve $\mathbf{z}(\lambda)$ can be found by using a standard software package for numerically solving systems of differential equations with initial values. This would come down to only performing the predictor step in the scheme described above. Although this is possible, in general it is not a good and efficient approach because it can lead to the same accumulation of errors occurring in numerical differential equation solvers and ignores the strong contractive properties which the curve has relative to the corrector steps in view of the fact that it satisfies $\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)=\mathbf{0}$. In the sequel, we first show how we can exploit (E.1.3) to find an Euler predictor, i.e. a predictor that estimates the solution $\mathbf{z}(\lambda+\Delta\lambda)$ of $\mathbf{h}(\mathbf{z}(\lambda+\Delta\lambda),\lambda+\Delta\lambda)=\mathbf{0}$ by moving from $\mathbf{z}(\lambda)$ in the direction of the tangent $\frac{d\mathbf{z}(\lambda)}{d\lambda}$ to the curve at $\mathbf{z}(\lambda)$. Then we show how we can use the first-order Taylor expansion of (E.1.2) to find a Newton corrector for correcting this estimated solution for fixed $\lambda+\Delta\lambda$.

To derive the Euler predictor, we consider system (E.1.3) and write it as follows:

$$\frac{d}{d\lambda}\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)=\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda),\lambda)\frac{d\mathbf{z}(\lambda)}{d\lambda}+\partial_\lambda\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)=\mathbf{0}\qquad\forall\,\lambda\in[0,1]\,, \tag{E.1.4}$$

where:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda),\lambda)=\begin{bmatrix}\frac{\partial h_1(\mathbf{z},\lambda)}{\partial z_1}&\cdots&\frac{\partial h_1(\mathbf{z},\lambda)}{\partial z_P}\\ \vdots&\ddots&\vdots\\ \frac{\partial h_P(\mathbf{z},\lambda)}{\partial z_1}&\cdots&\frac{\partial h_P(\mathbf{z},\lambda)}{\partial z_P}\end{bmatrix}\quad\text{and}\quad\partial_\lambda\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)\triangleq\frac{\partial\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)}{\partial\lambda}\,, \tag{E.1.5}$$

which are the Jacobian and partial derivative of $\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)$ w.r.t. $\mathbf{z}$ and λ respectively. This is equivalent to:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda),\lambda)\frac{d\mathbf{z}(\lambda)}{d\lambda}=-\frac{\partial\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)}{\partial\lambda}\qquad\forall\,\lambda\in[0,1]\,. \tag{E.1.6}$$

Solving this system for $\frac{d\mathbf{z}(\lambda)}{d\lambda}$ yields:

$$\frac{d\mathbf{z}(\lambda)}{d\lambda}=-\Big[\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda),\lambda)\Big]^{-1}\frac{\partial\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)}{\partial\lambda}\qquad\forall\,\lambda\in[0,1]\,. \tag{E.1.7}$$

Hence, the Euler predictor step is given by:

$$\hat{\mathbf{z}}(\lambda+\Delta\lambda)=\mathbf{z}(\lambda)+\Delta\lambda\frac{d\mathbf{z}(\lambda)}{d\lambda}=\mathbf{z}(\lambda)-\Delta\lambda\Big[\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}(\lambda),\lambda)\Big]^{-1}\frac{\partial\mathbf{h}\big(\mathbf{z}(\lambda),\lambda\big)}{\partial\lambda}\,, \tag{E.1.8}$$

which we briefly write as:

$$\mathbf{z}:=\mathbf{z}+\Delta\mathbf{z}=\mathbf{z}-\Delta\lambda\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}\frac{\partial\mathbf{h}}{\partial\lambda}\,, \tag{E.1.9}$$

where '$:=$' denotes the assignment operation. Note from (E.1.1) that:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda)=\gamma(1-\lambda)\nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z})+\lambda\nabla_{\mathbf{z}}\mathbf{p}(\mathbf{z})\,, \tag{E.1.10}$$

and:

$$\partial_\lambda \mathbf{h}(\mathbf{z},\lambda) = -\gamma \mathbf{g}(\mathbf{z}) + \mathbf{p}(\mathbf{z}) = \mathbf{p}(\mathbf{z}) - \gamma \mathbf{g}(\mathbf{z})\,. \tag{E.1.11}$$

To derive the Newton corrector, we consider system (E.1.2) for continuation parameter value $\lambda + \Delta\lambda$ and write it in terms of its first-order Taylor expansion in $\mathbf{z}$ around the point $\hat{\mathbf{z}}(\lambda + \Delta\lambda)$ predicted by the Euler step:

$$\begin{aligned}\mathbf{h}\big(\hat{\hat{\mathbf{z}}}(\lambda+\Delta\lambda),\lambda+\Delta\lambda\big) &\approx \mathbf{h}\big(\hat{\mathbf{z}}(\lambda+\Delta\lambda),\lambda+\Delta\lambda\big)\\ &+ \nabla_{\mathbf{z}}\mathbf{h}\big(\hat{\mathbf{z}}(\lambda+\Delta\lambda),\lambda+\Delta\lambda\big)\big(\hat{\hat{\mathbf{z}}}(\lambda+\Delta\lambda)-\hat{\mathbf{z}}(\lambda+\Delta\lambda)\big) \qquad \forall\,\lambda\in[0,1]\,.\end{aligned} \tag{E.1.12}$$

Since the left hand side of this equation should be zero due to (E.1.2), we obtain the following Newton corrector step:

$$\hat{\hat{\mathbf{z}}}(\lambda+\Delta\lambda) = \hat{\mathbf{z}}(\lambda+\Delta\lambda) - \Big[\nabla_{\mathbf{z}}\mathbf{h}\big(\hat{\mathbf{z}}(\lambda+\Delta\lambda),\lambda+\Delta\lambda\big)\Big]^{-1}\mathbf{h}\big(\hat{\mathbf{z}}(\lambda+\Delta\lambda),\lambda+\Delta\lambda\big)\,. \tag{E.1.13}$$

Realizing that the left hand side is an estimate of $\mathbf{z}(\lambda+\Delta\lambda)$, and that we can repeatedly apply the corrector, we briefly write (E.1.13) as follows:

$$\mathbf{z} := \mathbf{z} + \Delta\mathbf{z} = \mathbf{z} - \big[\nabla_{\mathbf{z}}\mathbf{h}\big]^{-1}\mathbf{h}\,. \tag{E.1.14}$$

Hence, for each fixed value of the continuation parameter the solution can be refined iteratively by performing several Newton corrections. Path following methods that keep the continuation parameter fixed during the correction of the predicted solutions are often called *increment-and-fix* methods [143, 182]. At each fixed value of λ we refine the solution until a certain (combined) stopping criterion is satisfied. For example, we could stop applying the corrector steps after the Newton correction becomes smaller than a certain tolerance, or after a fixed number of Newton corrections has been applied, or a combination of those criteria. As with numerical differential equation solvers the key to efficiency of the whole scheme lies in the use of a variable step size $\Delta\lambda$ for the continuation parameter λ. Depending on the outcome of the corrector the step size is increased or decreased. Although we have developed several working algorithms using this principle, for simplicity, and because our main interest is not in efficiency but rather in the rationale behind the methods, in this thesis we work with fixed step sizes.

Summarizing, we see that the predictor is obtained from the system of equations resulting from taking the derivative of (E.1.2) w.r.t. λ, whereas the corrector is obtained from the system of equations resulting from taking the derivative of (E.1.2) w.r.t. $\mathbf{z}$. In fact, both can be derived by considering a local model of the homotopy function via its first order Taylor expansion in both $\mathbf{z}$ and λ:

$$\mathbf{h}(\mathbf{z}+\Delta\mathbf{z},\lambda+\Delta\lambda) \approx \mathbf{h}(\mathbf{z},\lambda) + \nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda)\Delta\mathbf{z} + \partial_\lambda\mathbf{h}(\mathbf{z},\lambda)\Delta\lambda \qquad \forall\,\lambda\in[0,1]\,.$$

If we have a point $(\mathbf{z},\lambda)$ on or near the path, we can predict a new approximate solution $\mathbf{z}+\Delta\mathbf{z}$ of $\mathbf{z}$ at $\lambda+\Delta\lambda$ by setting $\mathbf{h}(\mathbf{z},\lambda)$ and $\mathbf{h}(\mathbf{z}+\Delta\mathbf{z},\lambda+\Delta\lambda)$ to zero and solving the first-order terms to obtain:

$$\Delta\mathbf{z} \approx -\big[\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda)\big]^{-1}\partial_\lambda\mathbf{h}(\mathbf{z},\lambda)\,\Delta\lambda \qquad \forall\,\lambda\in[0,1]\,. \tag{E.1.15}$$

This is equivalent to the Euler predictor step in (E.1.9). On the other hand, if we have a point $(\mathbf{z},\lambda)$ near the path for which $\mathbf{h}(\mathbf{z},\lambda)$ is not (yet) sufficiently small, we can correct to a new approximate solution $\mathbf{z}+\Delta\mathbf{z}$ of $\mathbf{z}$ at λ by setting $\Delta\lambda$ and $\mathbf{h}(\mathbf{z}+\Delta\mathbf{z},\lambda+\Delta\lambda)$ to zero and solving the first-order terms to obtain:

$$\Delta\mathbf{z} \approx -\big[\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda)\big]^{-1}\mathbf{h}(\mathbf{z},\lambda) \qquad \forall\,\lambda\in[0,1]\,. \tag{E.1.16}$$

This is equivalent to the Newton corrector step in (E.1.14). The key ingredients of the resulting algorithm are summarized in Alg. E.1. Note that at each continuation iteration of the homotopy algorithm the Euler predictor is evaluated at $(\hat{\mathbf{z}}(\lambda), \lambda)$ according to (E.1.8), whereas

Algorithm E.1 Overview of homotopy continuation method for solving a nonlinear system of equations by an increment-and-fix method with constant step size.

1: Initialize γ, $MaxNnwtIt$, $NwtTol$ and $NwtNoConvergenceTol$;

2: Construct a start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ that has the same structure and number of solutions as the target system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ (or more), and whose solutions are known;

3: Embed $\mathbf{g}(\mathbf{z})$ and $\mathbf{p}(\mathbf{z})$ in the following convex homotopy system:

$$\mathbf{h}(\mathbf{z}, \lambda) = \gamma(1-\lambda)\,\mathbf{g}(\mathbf{z}) + \lambda\,\mathbf{p}(\mathbf{z}) \qquad \forall\ \lambda \in [0,1]\,;$$

4: Compute the solutions of the start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, which are the solutions of the homotopy system for $\lambda = 0$, and store them in a set $\mathcal{G}$;

5: Choose a step size $\Delta\lambda$ for $\lambda \in [0,1]$ that is 'sufficiently small';

6: For each solution $\mathbf{z}_g \in \mathcal{G}$ of $\mathbf{h}(\mathbf{z}, 0) = \mathbf{g}(\mathbf{z}) = \mathbf{0}$, trace the curve $(\mathbf{z}(\lambda), \lambda) \in \mathbf{h}^{-1}(\mathbf{0})$ defined implicitly by $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}$ from the starting point $(\mathbf{z}_g, 0)$ to a solution point $(\mathbf{z}_p, 1)$ of $\mathbf{h}(\mathbf{z}, 1) = \mathbf{p}(\mathbf{z}) = \mathbf{0}$ by the following continuation method:

for all** $\mathbf{z}_g \in \mathcal{G}$ **do
  $\mathbf{z} := \mathbf{z}_g$;
  for** $\lambda = 0 \rightarrow 1$ in steps of $\Delta\lambda$ **do
    % Euler predictor step:
    $\frac{d\mathbf{z}}{d\lambda} = -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}\partial_\lambda \mathbf{h}$;
    $\mathbf{z} := \mathbf{z} + \Delta\lambda\,\frac{d\mathbf{z}}{d\lambda}$;

    % Newton corrector steps:
    $\lambda := \lambda + \Delta\lambda$;
    for** $m = 1 \rightarrow MaxNnwtIt$ **do
      $NwtCorr := -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1}\mathbf{h}$;
      $\mathbf{z} := \mathbf{z} + NwtCorr$;
      if** $norm(NwtCorr) < NwtTol$ **then
        Leave corrector loop;
      end if
    end for
    if** $norm(NwtCorr) > NwtNoConvergenceTol$ **then
      Print warning or break because Newton corrector did not converge;
    end if
  end for
  Store solution $\mathbf{z}_p = \mathbf{z}$ in set $\mathcal{P}$;
end for
Return $\mathcal{P}$;

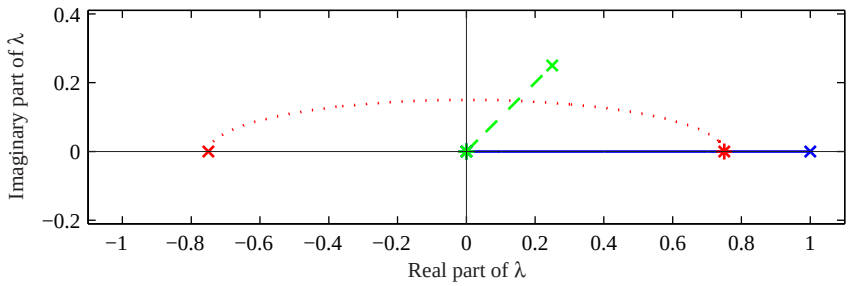

Figure E.1: *Possible continuation parameter paths in the complex plane.*

the Newton corrector is evaluated at $(\hat{\mathbf{z}}(\lambda+\Delta\lambda), \lambda+\Delta\lambda)$ according to (E.1.13). Also notice that at each fixed value of λ we stop applying the corrector steps either if the Newton correction becomes smaller than a certain tolerance $NwtTol$, or when a certain maximum number $MaxNnwtIt$ of Newton corrections has been applied. See the next sections for examples.

Now we will extend the results obtained above to the new homotopy referred to at the end of the first paragraph of Section E.1. We define this more natural and intuitive homotopy as:

$$\mathbf{h}(\mathbf{z},\lambda) = \gamma_g(\lambda-\lambda_e)\,\mathbf{g}(\mathbf{z}) + \gamma_p(\lambda-\lambda_0)\,\mathbf{p}(\mathbf{z}) \qquad \forall\ \lambda \in \mathcal{C}\,, \tag{E.1.17}$$

where $\mathcal{C}$ is a certain curve in the complex plane that has to be chosen by the user. Similarly to γ in (E.1.1), the constants $\gamma_g \in \mathbb{C}$ and $\gamma_p \in \mathbb{C}$ are randomly chosen fixed constants that serve to avoid singularities and crossings along the different paths. Now, the continuation parameter $\lambda \in \mathbb{R}$ $(\mathbb{C})$ that parameterizes each solution path $(\mathbf{z}(\lambda), \lambda)$ of (E.1.17) is allowed to be complex. In the course of tracing such a path by a homotopy continuation method, λ follows the curve $\mathcal{C}$ from its starting point $\lambda_0 \in \mathbb{R}$ $(\mathbb{C})$ to its end point $\lambda_e \in \mathbb{R}$ $(\mathbb{C})$, where $\lambda_0 \neq \lambda_e$. Some examples of possible paths that may be traversed in both directions are depicted in Fig. E.1. The end points of each trajectory are indicated by a solid star and a circle. Note that $\mathbf{h}(\mathbf{z},\lambda_0) = \gamma_g(\lambda_0-\lambda_e)\,\mathbf{g}(\mathbf{z})$ and $\mathbf{h}(\mathbf{z},\lambda_e) = \gamma_p(\lambda_e-\lambda_0)\,\mathbf{p}(\mathbf{z})$ with $\lambda_0-\lambda_e \neq 0$. Hence, solving $\mathbf{h}(\mathbf{z},\lambda_0) = \mathbf{0}$ is equivalent to solving $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, and solving $\mathbf{h}(\mathbf{z},\lambda_e) = \mathbf{0}$ is equivalent to solving $\mathbf{p}(\mathbf{z}) = \mathbf{0}$. From our simulations, we have noticed that our new homotopy (E.1.17) with the 'generalized gamma trick' and 'generalized continuation parameter' significantly outperforms that in (E.1.1) in avoiding crossings of different solution paths. Contrary to homotopy (E.1.1), for the considered systems we did not encounter any problems with homotopy (E.1.17). Useful and common choices for γ_g and γ_p are given by $\gamma_g = \exp(\jmath\theta_g)$ and $\gamma_p = \exp(\jmath\theta_p)$ with θ_g and θ_p picked at random from a uniform distribution on the interval $[0, 2\pi]$. Since the homotopy in (E.1.17) contains two different randomly chosen fixed constants, it is slightly more 'randomized' than the one in (E.1.1). Virtually all results developed above remain valid for the new homotopy, possibly in a slightly modified form. In fact, essentially only the expressions for $\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda)$ and $\partial_\lambda\mathbf{h}(\mathbf{z},\lambda)$ in (E.1.10) and (E.1.11) respectively change and become:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z},\lambda) = \gamma_g(\lambda-\lambda_e)\nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z}) + \gamma_p(\lambda-\lambda_0)\nabla_{\mathbf{z}}\mathbf{p}(\mathbf{z})\,, \tag{E.1.18}$$

and:

$$\partial_\lambda\mathbf{h}(\mathbf{z},\lambda) = \gamma_g\mathbf{g}(\mathbf{z}) + \gamma_p\mathbf{p}(\mathbf{z})\,. \tag{E.1.19}$$

Adapting Alg. E.1 to the new homotopy in (E.1.17) yields Alg. E.2 on the facing page.

Algorithm E.2 Overview of homotopy continuation method with 'generalized gamma trick' and 'generalized continuation parameter'.

1: Initialize γ_g, γ_p, $MaxNnwtIt$, $NwtTol$ and $NwtNoConvergenceTol$;

2: Construct a start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ that has the same structure and number of solutions as the target system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ (or more), and whose solutions are known;

3: Embed $\mathbf{g}(\mathbf{z})$ and $\mathbf{p}(\mathbf{z})$ in the following convex homotopy system:

$$\mathbf{h}(\mathbf{z}, \lambda) = \gamma_g(\lambda - \lambda_e)\,\mathbf{g}(\mathbf{z}) + \gamma_p(\lambda - \lambda_0)\,\mathbf{p}(\mathbf{z}) \qquad \forall\, \lambda \in \mathcal{C}\,,$$

where the continuation parameter $\lambda \in \mathbb{C}$ follows the predefined curve $\mathcal{C}$ in the complex plane from λ_0 to λ_e with $\lambda_0 \neq \lambda_e$;

4: Compute the solutions of the start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, which are the solutions of the homotopy system for $\lambda = \lambda_0$, and store them in a set $\mathcal{G}$;

5: Choose an ordered sequence $\mathcal{C}_o$ of values for λ lying on $\mathcal{C}$ such that λ progresses sequentially from λ_0 to λ_e in 'sufficiently small' steps;

6: For each solution $\mathbf{z}_g \in \mathcal{G}$ of $\mathbf{h}(\mathbf{z}, \lambda_0) = \mathbf{0} \equiv \mathbf{g}(\mathbf{z}) = \mathbf{0}$, trace the curve $(\mathbf{z}(\lambda), \lambda) \in \mathbf{h}^{-1}(\mathbf{0})$ defined implicitly by $\mathbf{h}(\mathbf{z}(\lambda), \lambda) = \mathbf{0}$ from the starting point $(\mathbf{z}_g, \lambda_0)$ to a solution point $(\mathbf{z}_p, \lambda_e)$ of $\mathbf{h}(\mathbf{z}, \lambda_e) = \mathbf{0} \equiv \mathbf{p}(\mathbf{z}) = \mathbf{0}$ by the following continuation method:

for all** $\mathbf{z}_g \in \mathcal{G}$ **do
  $\mathbf{z} := \mathbf{z}_g$;
  for** $\lambda = \lambda_0 \rightarrow \lambda_e$ along $\mathcal{C}_o$ **do
    % Euler predictor step:
    $\frac{d\mathbf{z}}{d\lambda} = -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1} \partial_\lambda \mathbf{h}$;
    $\mathbf{z} := \mathbf{z} + \Delta\lambda \frac{d\mathbf{z}}{d\lambda}$;
    % Newton corrector steps:
    $\lambda :=$ *next* λ *from* $\mathcal{C}_o$;
    for** $m = 1 \rightarrow MaxNnwtIt$ **do
      $NwtCorr := -\left[\nabla_{\mathbf{z}}\mathbf{h}\right]^{-1} \mathbf{h}$;
      $\mathbf{z} := \mathbf{z} + NwtCorr$;
      if** $norm(NwtCorr) < NwtTol$ **then
        Leave corrector loop;
      end if
    end for
    if** $norm(NwtCorr) > NwtNoConvergenceTol$ **then
      Print warning or break because Newton corrector did not converge;
    end if
  end for
  Store solution $\mathbf{z}_p = \mathbf{z}$ in set $\mathcal{P}$;
end for
Return $\mathcal{P}$;

E.2 Example for univariate polynomials

In order to develop a feeling for the operation of the homotopy method described in the previous section, we apply it to one of the simplest possible problems in this context, viz. that of solving one equation in one variable. More specifically, we consider the problem of finding all roots of a univariate polynomial $p : \mathbb{C} \to \mathbb{C}$ of degree P defined as follows:

$$p(z) = \sum_{p=0}^{P} a_p (z)^p = a_0 + a_1 z + \cdots + a_{P-1}(z)^{P-1} + a_P (z)^P \,. \tag{E.2.1}$$

Without loss of generality, we assume that $a_P = 1$ and choose the start system $g : \mathbb{C} \to \mathbb{C}$:

$$g(z) = (z)^P - 1 \,, \tag{E.2.2}$$

which has the following known solutions lying on the unit circle:

$$z = \exp(\jmath 2\pi p/P), \quad 0 \le p \le P - 1 \,. \tag{E.2.3}$$

Note that $g(z)$ is 'compatible' with $p(z)$ in the sense that it has the same structure, i.e. it is also a univariate polynomial of degree P, and has the same number of solutions P. Now, the target system (more precisely, the target function) $p(z)$ is embedded in the convex homotopy according to (E.1.1):

$$h(z, \lambda) = \gamma(1 - \lambda)\, g(z) + \lambda\, p(z) \qquad \forall\, \lambda \in [0, 1] \,, \tag{E.2.4}$$

where $\lambda \in \mathbb{R}$ is the continuation parameter, γ is the randomly chosen fixed constant used for avoiding path crossing singularities, and $h : \mathbb{C} \times \mathbb{R} \to \mathbb{C}$ is the homotopy function. In order to derive the Euler predictor and Newton corrector steps, we first compute $\frac{\partial h}{\partial z}$ and $\frac{\partial h}{\partial \lambda}$ as follows:

$$\frac{\partial h(z, \lambda)}{\partial z} = \gamma(1 - \lambda)\frac{dg(z)}{dz} + \lambda \frac{dp(z)}{dz} \triangleq \gamma(1 - \lambda) g'(z) + \lambda p'(z) \tag{E.2.5}$$

respectively:

$$\frac{\partial h(z, \lambda)}{\partial \lambda} = -\gamma g(z) + p(z) \,, \tag{E.2.6}$$

where:

$$g'(z) \triangleq \frac{dg(z)}{dz} = P(z)^{P-1} \tag{E.2.7}$$

and

$$p'(z) \triangleq \frac{dp(z)}{dz} = \sum_{p=1}^{P} p\, a_p (z)^{p-1} = a_1 + 2a_2 z + \cdots + P a_P (z)^{P-1} \,. \tag{E.2.8}$$

From (E.1.7) it follows that:

$$\frac{dz(\lambda)}{d\lambda} = -\left[\frac{\partial h(z, \lambda)}{\partial z}\right]^{-1} \frac{\partial h(z, \lambda)}{\partial \lambda} = \frac{\gamma g(z) - p(z)}{\gamma(1 - \lambda) g'(z) + \lambda p'(z)} \,. \tag{E.2.9}$$

Substituting this result into (E.1.8) gives the Euler predictor at $(z(\lambda), \lambda)$:

$$\hat{z}(\lambda + \Delta\lambda) = z(\lambda) + \Delta\lambda \frac{dz(\lambda)}{d\lambda} = z(\lambda) + \Delta\lambda \frac{\gamma g(z) - p(z)}{\gamma(1 - \lambda) g'(z) + \lambda p'(z)} \,. \tag{E.2.10}$$

Substituting (E.2.5) into (E.1.13) or (E.1.16) gives the Newton corrector at $(z(\lambda), \lambda)$:

$$\hat{z}(\lambda) := z(\lambda) - \left[\frac{\partial h(z, \lambda)}{\partial z}\right]^{-1} h\big(z, \lambda\big) = z(\lambda) - \frac{\gamma(1 - \lambda) g(z) + \lambda p(z)}{\gamma(1 - \lambda) g'(z) + \lambda p'(z)} \,. \tag{E.2.11}$$

Algorithm E.3 Homotopy continuation method for finding all roots of the univariate polynomial $p(z) = \sum_{p=0}^{P} a_p (z)^p$ by an increment-and-fix method with constant step size.

1: Initialize γ, $MaxNnwtIt$, $NwtTol$ and $NwtNoConvergenceTol$;

2: Choose the start system $g(z) = (z)^P - 1 = 0$;

3: Embed $g(z)$ and $p(z)$ in the following convex homotopy system:

$$h(z, \lambda) = \gamma(1 - \lambda)\, g(z) + \lambda\, p(z) \qquad \forall\ \lambda \in [0, 1]\,;$$

4: Compute the solutions of the start system:

$$\mathcal{G} = \{1, \exp(\jmath 2\pi/P), \exp(\jmath 4\pi/P), \ldots, \exp(\jmath 2\pi(P-1)/P)\}\,;$$

5: Choose a step size $\Delta\lambda$ for $\lambda \in [0, 1]$;

6: For each solution $z_g \in \mathcal{G}$ of $h(z, 0) = \gamma\, g(z) = 0$, trace the curve $(z(\lambda), \lambda) \in h^{-1}(0)$ defined implicitly by $h(z(\lambda), \lambda) = 0$ from the starting point $(z_g, 0)$ to a solution point $(z_p, 1)$ of $h(z, 1) = p(z) = 0$ by the following continuation method:

for $p = 0 \to P - 1$ ***do***
  $z := \exp(\jmath 2\pi p/P)$;
  for $\lambda = 0 \to 1$ *in steps of* $\Delta\lambda$ ***do***
    % Euler predictor step:
    $z := z + \Delta\lambda \frac{\gamma g(z) - p(z)}{\gamma(1-\lambda)g'(z) + \lambda p'(z)}$;
    % Newton corrector steps:
    $\lambda := \lambda + \Delta\lambda$;
    for $m = 1 \to MaxNnwtIt$ ***do***
      $NwtCorr := -\frac{\gamma(1-\lambda)g(z) + \lambda p(z)}{\gamma(1-\lambda)g'(z) + \lambda p'(z)}$;
      $z := z + NwtCorr$;
      if $abs(NwtCorr) < NwtTol$ ***then***
        Leave corrector loop;
      end if
    end for
    if $norm(NwtCorr) > NwtNoConvergenceTol$ ***then***
      Print warning or break because Newton corrector did not converge;
    end if
  end for
  Store solution $z_p = z$ in set $\mathcal{P}$;
end for
Return $\mathcal{P}$;

Substituting the results derived in this section into Alg. E.1 yields an algorithm tailored to the current problem, see Alg. E.3. Note again that at each iteration of this algorithm, i.e. for each value of λ, the Euler predictor (E.2.10) is evaluated at $(\hat{z}(\lambda), \lambda)$, whereas the Newton corrector (E.2.11) is evaluated at $(\hat{z}(\lambda + \Delta\lambda), \lambda + \Delta\lambda)$.

Now, we will provide a specific example for a univariate polynomial of degree $P = 6$. The coefficients $a_0, \ldots, a_P$ of the polynomial $p(z)$ in (E.2.1) are generated by firstly drawing $P+1$ independent samples from a uniform probability distribution on the interval $[-0.5, 0.5]$, and then normalizing this sequence in such a way that the coefficient for $(z)^P$ equals one. The resulting polynomial is given by:

$$p(z) = 0.0917 + 1.6506z - 0.2802(z)^2 + 1.6698(z)^3 + 0.4810(z)^4 + 2.2242(z)^5 + (z)^6 \,. \tag{E.2.12}$$

The solutions of the start system $g(z) = (z)^6 - 1 = 0$ are given by:

$$\mathcal{G} = \{1, \exp(\jmath 1/3\pi), \exp(\jmath 2/3\pi), \exp(\jmath\pi), \exp(\jmath 4/3\pi), \exp(\jmath 5/3\pi)\} \,. \tag{E.2.13}$$

The randomly generated constant γ used in (E.2.4) equals $0.9563 - 0.2923\jmath$. Furthermore, the other parameters used in Alg. E.3 are chosen as $\Delta\lambda = 10^{-2}$, $MaxNnwtIt = 2$, $NwtTol = 10^{-4}$ and $NwtNoConvergenceTol = 10^{-2}$ respectively. The real and imaginary parts of each of the six solution paths of the homotopy system in (E.2.4) with $p(z)$ and $g(z)$ as defined above are depicted in Fig. E.2 as a function of the continuation parameter λ running from 0 to 1. The beginnings of the paths, i.e. the roots of $g(z)$ are denoted by the crosses '$\times$', whereas the endpoints, i.e. the estimated roots of $p(z)$, are denoted by the stars '$*$'. With four decimals accuracy, the six roots of $p(z)$ found by the algorithm are:

$$\mathcal{P} = \{-2.3862, -0.0549, -0.4015 \pm 0.9128\jmath, 0.5100 \pm 0.6664\jmath\} \,.$$

Any algebra package can be used to verify that these are the correct roots. The figure shows that the solution paths smoothly run from the roots of $g(z)$ to those of $p(z)$ without crossing each other, i.e. without going through the same point. Using an ODE solver we also verified that the solutions of differential equation (E.2.9) for each of the initial conditions in $\mathcal{G}$ (E.2.13) yielded the same roots, but with less accuracy.

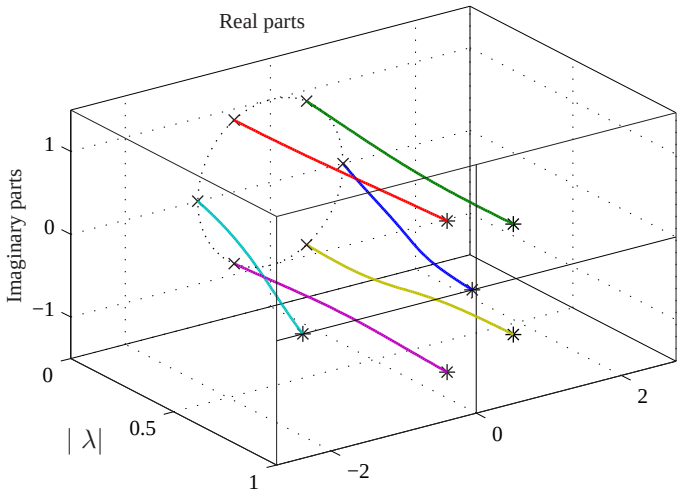

Figure E.2: *Solution paths of homotopy system $h(z, \lambda) = 0$ with $h(z, \lambda)$ defined in* (E.2.4) *and $p(z)$ in* (E.2.12).

E.3 Example 1 for bivariate polynomials

In this section we further develop our feeling for the operation of the homotopy method described in Section E.1 by applying it to an example that is more general and more complicated than the one presented in the previous section, viz. that of solving a system of two bivariate polynomial equations of degree two. Although our focus is on solving systems numerically, to provide more insight we choose a system that can also be solved algebraically. More specifically, we consider the problem of finding all solutions of the system $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ with $\mathbf{p} : \mathbb{C}_2 \rightarrow \mathbb{C}_2$ defined as follows:

$$\mathbf{p}(\mathbf{z}) = \begin{bmatrix} p_1(z_1, z_2) \\ p_2(z_1, z_2) \end{bmatrix} \triangleq \begin{bmatrix} (z_1 - \frac{1}{3}z_2)(z_1 - 2z_2) \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} = \begin{bmatrix} (z_1)^2 - 2\frac{1}{3}z_1 z_2 + \frac{2}{3}(z_2)^2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix}. \quad \text{(E.3.1)}$$

Note that in general we do not know the factorization of the first function in the upper equation and that in fact the purpose of solving system (E.3.1) is to compute such a factorization; see also Section 4.4.1. System (E.3.1) is reminiscent of the type of systems that arise in this thesis, see Chapter 4 for example, where we need to solve a system of homogeneous bivariate polynomial equations subject to a unit-norm constraint on the solutions. Geometrically, solving a system of equations means finding the intersections between the curves defined by the equations. Now, we first examine the structure of the solution set. Firstly, from (E.3.1) it is clear that the curves defined by the homogeneous polynomial equation $p_1(z_1, z_2) = 0$ are straight lines through the origin that in the real $z_1 - z_2$ plane can be represented by the following parameterizations:

$$\mathbf{z}(\eta) \triangleq \begin{bmatrix} z_1(\eta) \\ z_2(\eta) \end{bmatrix} = \eta \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \quad \mathbf{z}(\eta) \triangleq \begin{bmatrix} z_1(\eta) \\ z_2(\eta) \end{bmatrix} = \eta \begin{bmatrix} 2 \\ 1 \end{bmatrix} \qquad \forall\, \eta \in \mathbb{R}\,. \quad \text{(E.3.2)}$$

Secondly, it is also clear from (E.3.1) that the equation $p_2(z_1, z_2) = 0$ corresponds to a circle with center in the origin and radius one that can be parameterized as follows:

$$\mathbf{z}(\theta) \triangleq \begin{bmatrix} z_1(\theta) \\ z_2(\theta) \end{bmatrix} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} \qquad \forall\, \theta \in [0, 2\pi]\,. \quad \text{(E.3.3)}$$

Obviously, (E.3.2) and (E.3.3) represent the zero contour levels of the functions $p_1(z_1, z_2)$ and $p_2(z_1, z_2)$ respectively, see Fig. 4.16. Obviously, there are four intersection points, two of which are essentially different because points that are opposite w.r.t. the origin are (projectively) equivalent according to (4.5.4). Using simple algebra, it can easily be shown that these points are given by:

$$\mathbf{z} = \pm\frac{1}{\sqrt{10}} \begin{bmatrix} 1 \\ 3 \end{bmatrix} \qquad \text{and} \qquad \mathbf{z} = \pm\frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \quad \text{(E.3.4)}$$

which are simply the vectors in (E.3.2) normalized to unit Euclidian norm. In Fig. E.3 they are indicated by the solid black asterisks. With five decimals accuracy, the solution set of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ is given by:

$$\mathcal{P} = \left\{ \begin{bmatrix} 0.31623 \\ 0.94868 \end{bmatrix}, -\begin{bmatrix} 0.31623 \\ 0.94868 \end{bmatrix}, \begin{bmatrix} 0.89443 \\ 0.44721 \end{bmatrix}, -\begin{bmatrix} 0.89443 \\ 0.44721 \end{bmatrix} \right\}. \quad \text{(E.3.5)}$$

Now we will demonstrate how the same solutions can be obtained numerically by means of the homotopy method discussed in Section E.1 and summarized in Alg. E.1.

We first compute the quantities needed for applying Alg. E.1. The Jacobian of $\mathbf{p}(\mathbf{z})$ is:

$$\nabla_{\mathbf{z}}\mathbf{p}(\mathbf{z}) = \begin{bmatrix} \frac{\partial p_1(z_1,z_2)}{\partial z_1} & \frac{\partial p_1(z_1,z_2)}{\partial z_2} \\ \frac{\partial p_2(z_1,z_2)}{\partial z_1} & \frac{\partial p_2(z_1,z_2)}{\partial z_2} \end{bmatrix} = \begin{bmatrix} 2z_1 - 2\frac{1}{3}z_2 & -2\frac{1}{3}z_1 + 1\frac{1}{3}z_2 \\ 2z_1 & 2z_2 \end{bmatrix}. \quad \text{(E.3.6)}$$

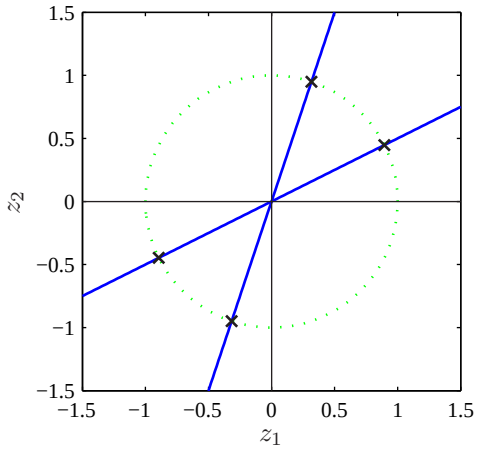

Figure E.3: *Intersecting zero contour curves of bivariate homogeneous polynomial (solid blue line) and unit-norm constraint function (dashed green line).*

We choose the start system $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ with $\mathbf{g} : \mathbb{C}_2 \to \mathbb{C}_2$ defined as follows:

$$\mathbf{g}(\mathbf{z}) = \begin{bmatrix} g_1(z_1, z_2) \\ g_2(z_1, z_2) \end{bmatrix} \triangleq \begin{bmatrix} (z_1)^2 - (\beta_1)^2 \\ (z_2)^2 - (\beta_2)^2 \end{bmatrix}, \tag{E.3.7}$$

where β_1 and β_2 are randomly chosen fixed complex constants [2, 116]. We choose β_1 and β_2 as $\beta_k = \exp(\jmath\theta_k)$ with θ_k picked at random from the interval $[0, 2\pi]$. This system has the following four solutions:

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \pm\beta_1 \\ \pm\beta_2 \end{bmatrix}. \tag{E.3.8}$$

Hence:

$$\mathcal{G} = \left\{ \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}, -\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}, \begin{bmatrix} -\beta_1 \\ \beta_2 \end{bmatrix}, -\begin{bmatrix} -\beta_1 \\ \beta_2 \end{bmatrix} \right\}. \tag{E.3.9}$$

The Jacobian of $\mathbf{g}(\mathbf{z})$ is given by:

$$\nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z}) = \begin{bmatrix} \frac{\partial g_1(z_1,z_2)}{\partial z_1} & \frac{\partial g_1(z_1,z_2)}{\partial z_2} \\ \frac{\partial g_2(z_1,z_2)}{\partial z_1} & \frac{\partial g_2(z_1,z_2)}{\partial z_2} \end{bmatrix} = \begin{bmatrix} 2z_1 & 0 \\ 0 & 2z_2 \end{bmatrix}. \tag{E.3.10}$$

Note that $\mathbf{g}(\mathbf{z})$ is compatible with $\mathbf{p}(\mathbf{z})$ in the sense that it has a similar structure, i.e. it also consists of two bivariate polynomials of degree 2 and has the same number of solutions 4. Moreover, the solution sets have a similar structure in that each solution is paired by its negative version. Now, the target function vector $\mathbf{p}(\mathbf{z})$ is embedded in the convex homotopy according to (E.1.1):

$$\begin{aligned} \mathbf{h}(\mathbf{z}, \lambda) &= \gamma(1-\lambda)\,\mathbf{g}(\mathbf{z}) + \lambda\,\mathbf{p}(\mathbf{z}) \\ &= \gamma(1-\lambda)\begin{bmatrix} (z_1)^2 - (\beta_1)^2 \\ (z_2)^2 - (\beta_2)^2 \end{bmatrix} + \lambda\begin{bmatrix} (z_1)^2 - 2\frac{1}{3}z_1z_2 + \frac{2}{3}(z_2)^2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix}. \end{aligned} \tag{E.3.11}$$

To compute the Euler predictor and Newton corrector steps in Alg. E.1 we need the quantities $\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}, \lambda)$ and $\partial_\lambda \mathbf{h}(\mathbf{z}, \lambda)$. According to (E.1.10), (E.3.6), and (E.3.10), the first is given by:

$$\nabla_{\mathbf{z}}\mathbf{h}(\mathbf{z}, \lambda) = \gamma(1-\lambda)\begin{bmatrix} 2z_1 & 0 \\ 0 & 2z_2 \end{bmatrix} + \lambda\begin{bmatrix} 2z_1 - 2\frac{1}{3}z_2 & -2\frac{1}{3}z_1 + 1\frac{1}{3}z_2 \\ 2z_1 & 2z_2 \end{bmatrix}, \tag{E.3.12}$$

and according to (E.1.11) the second is given by:

$$\partial_\lambda \mathbf{h}(\mathbf{z},\lambda) = \begin{bmatrix} (z_1)^2 - 2\frac{1}{3}z_1 z_2 + \frac{2}{3}(z_2)^2 \\ (z_1)^2 + (z_2)^2 - 1 \end{bmatrix} - \gamma \begin{bmatrix} (z_1)^2 - (\beta_1)^2 \\ (z_2)^2 - (\beta_2)^2 \end{bmatrix}. \tag{E.3.13}$$

We now have all ingredients required by Alg. E.1. The parameters γ, $\Delta\lambda$, $MaxNnwtIt$, $NwtTol$ and $NwtNoConvergenceTol$ are the same as in Section E.2. Furthermore, picking β_1 and β_2 as described above yielded $\beta_1 = -0.2846 - 0.9586\jmath$ and $\beta_2 = 0.9583 - 0.2858\jmath$. For each of the four solution paths of homotopy system (E.1.1), the real and imaginary parts of $z_1(\lambda)$ and $z_2(\lambda)$ are depicted in Fig. E.4 as a function of the continuation parameter λ in the interval $[0, 1]$. The z_1-components of the solution paths are represented by the solid lines, whereas the z_2-components are represented by the dotted lines. A pair of lines consisting of a solid and a dotted line of the same color or gray shade constitutes one solution path. The z_1-components of the beginnings of the paths, i.e. the z_1-components of the solutions of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$, are denoted by the crosses '$\times$', whereas the corresponding z_2-components are denoted by the diamonds '$\diamond$'. Likewise, the z_1-components of the endpoints of the paths, i.e. the z_1-components of the solutions of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$, are denoted by the stars '$*$', whereas the corresponding z_2-components are denoted by the dots '$\cdot$'. For reference, we have drawn the unit circle in the $z_1 - z_2$ plane for $\lambda = 0$, and the z_1- and z_2-axes in the $z_1 - z_2$ plane for $\lambda = 1$. With five decimals accuracy, the solution set of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ found by the algorithm is:

$$\mathcal{P} = \left\{ \begin{bmatrix} 0.31623 \\ 0.94868 \end{bmatrix}, -\begin{bmatrix} 0.31623 \\ 0.94868 \end{bmatrix}, \begin{bmatrix} 0.89443 \\ 0.44721 \end{bmatrix}, -\begin{bmatrix} 0.89443 \\ 0.44721 \end{bmatrix} \right\}. \tag{E.3.14}$$

As can be verified by comparing this set to that in (E.3.5), all roots have been found correctly.

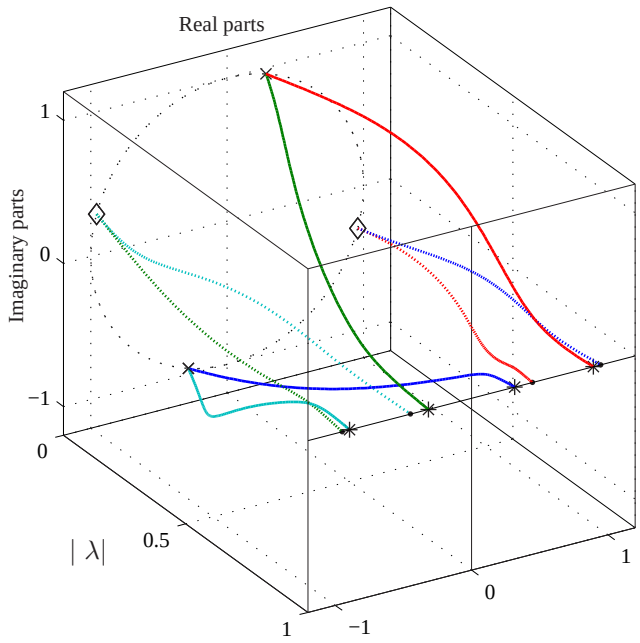

Figure E.4: *Real and imaginary parts of $z_1(\lambda)$-components (solid lines) and $z_2(\lambda)$-components (dotted lines) of solution paths of homotopy system $\mathbf{h}(\mathbf{z},\lambda) = \mathbf{0}$ with $\mathbf{h}(\mathbf{z},\lambda)$ defined in* (E.3.11).

Fig. E.4 shows that the solution paths run smoothly from those of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ to those of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$ without going through the same point. Our algorithm has computed the solution paths in such a way that the j-th solution of the start system, i.e. the j-th element of $\mathcal{G}$ in (E.3.9), leads to the j-th solution of the target system, i.e. the j-th element of $\mathcal{P}$ in (E.3.14). This can also be seen in the figure. Hence, for the current problem we see that if a solution $\mathbf{z}_g$ of $\mathbf{g}(\mathbf{z}) = \mathbf{0}$ leads to a solution $\mathbf{z}_p$ of $\mathbf{p}(\mathbf{z}) = \mathbf{0}$, then $-\mathbf{z}_g$ leads to $-\mathbf{z}_p$. More generally, it can be shown that this also holds for homotopies with a start and target system of the same type as in (E.3.1) with a zero contour level set similar to that in Fig. E.3. This implies that in order to obtain only the solutions that are essentially different according to (4.5.4), we only need to trace two solution paths of the homotopy system that start from two essentially different solutions of the start system, e.g. $\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$ and $\begin{bmatrix} -\beta_1 \\ \beta_2 \end{bmatrix}$, or $-\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$ and $\begin{bmatrix} \beta_1 \\ -\beta_2 \end{bmatrix}$, thereby reducing the computational complexity.

Scientific publications by the author

- J. van de Laar and P. Sommen, "Comparison of Discrete Polar Fourier Transform to 2D Cartesian FFT for Nonuniform Polar Input Data," in *Proc. 11th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2000)*, Veldhoven, The Netherlands, 30 Nov. – 1 Dec. 2000, pp. 355–363.
- J. van de Laar, E. Habets, J. Peters, and P. Lokkart, "Adaptive Blind Audio Signal Separation on a DSP," in *Proc. 12th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2001)*, Veldhoven, The Netherlands, Nov. 2001, pp. 475–479.
- N. Suditu, J. van de Laar, and P. Sommen, "Evaluation of Dereverberation Capabilities of a Nearfield Broadband Beamformer," in *Proc. 12th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2001)*, Veldhoven, The Netherlands, Nov. 2001, pp. 656–661.
- J. van de Laar and P. Sommen, "Exploiting Non-Stationarity in Multi-Channel Acoustic Echo Cancellation," in *Proc. 3rd IEEE Benelux Signal Processing Symposium (SPS 2002)*, Leuven, Belgium, Mar. 2002, pp. 9–12.
- J. van de Laar and P. Sommen, "Performance Analysis of Multi-Channel Adaptive Algorithms Operating in a Non-Stationary Acoustic Teleconferencing Setup," in *Proc. 2nd IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2002)*, Washington D.C., USA, Aug. 2002, pp. 234–238.
- J. van de Laar and P. Sommen, "Closed-form Instantaneous Blind Signal Separation based on Second Order Statistics and Non-Stationarity," in *Proc. 13th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2002)*, Veldhoven, The Netherlands, Nov. 2002, pp. 334–338.
- J. van de Laar and P. Sommen, "Conditions on Sources and Mixing Matrix for Solving the Permutation Indeterminacy in 2×2 Instantaneous Blind Signal Separation," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003)*, Hong Kong, China, Apr. 2003, pp. 317–320.
- J. van de Laar, "Instantaneous Blind Source Separation based on the Exploitation of Temporal Correlations and Nonstationarity," in *Proc. 14th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2003)*, Veldhoven, The Netherlands, Nov. 2003, pp. 391–398.
- J. van de Laar, "ULA TIME-MUSIC DOA Estimation based on the Exploitation of Some Arbitrary-Order Temporal Structure in the Data," in *Proc. 4th IEEE Benelux Signal Processing Symposium (SPS 2004)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 179–182.
- J. van de Laar, "On MIMO Instantaneous Blind Identification based on the Exploitation of the Time Structure of Signals using Arbitrary-Order Cumulants," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2004)*, Montreal, Quebec, Canada, May 2004, pp. 549–552.

- J. van de Laar, "TIME-MUSIC DOA Estimation based on the Exploitation of Some Arbitrary-Order Temporal Structure in the Data," in *Proc. 3rd IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2004)*, Barcelona, Spain, July 2004, pp. 308–312.
- J. van de Laar, "MIMO Instantaneous Blind Identification based on Second Order Temporal Structure and Homotopy Method," in *Proc. 9th International Workshop on Acoustic Echo and Noise Control (IWAENC 2005)*, Eindhoven, The Netherlands, Sept. 2005, pp. 241–244.
- J. van de Laar, M. Moonen, and P. Sommen, "MIMO Instantaneous Blind Identification Based on Second Order Temporal Structure," *IEEE Trans. Signal Processing*, submitted for publication in Aug. 2007.

Bibliography

[1] A. Ahmed, "Signal Separation," Ph.D. dissertation, University of Cambridge, Cambridge, United Kingdom, 2000.

[2] E. Allgower and K. Georg, *Introduction to Numerical Continuation Methods*, ser. Classics in Applied Mathematics. Philadelphia: SIAM, 2003.

[3] S. Alliot, E. Deprettere, and C. de Vos, "A Model for Architecture Exploration, Scaling Generic Blocks for Very Large Phased Array Telescopes," in *SPIE workshop on integrated modeling*, Lund, Sweden, Feb. 2002.

[4] S.-I. Amari, "Natural Gradient Works Efficiently in Learning," *Neural Computation*, vol. 10, no. 2, pp. 251–276, Feb. 1998.

[5] S.-I. Amari and A. Cichocki, "Adaptive Blind Signal Processing—Neural Network Approaches," *Proc. IEEE*, vol. 86, no. 10, pp. 2026–2048, 1998.

[6] S.-I. Amari, A. Cichocki, and H. Yang, "A New Learning Algorithm for Blind Source Separation," in *Advances in Neural Information Processing Systems*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds. Cambridge, MA, 1996: MIT Press, 1995, vol. 8, pp. 757–763.

[7] ——, "Blind Signal Separation and Extraction: Neural and Information-Theoretic Approaches," in *Unsupervised Adaptive Filtering, Volume I: Blind Source Separation*, S. Haykin, Ed. John Wiley and Sons, Inc., 2000, ch. 3, pp. 63–138.

[8] H. Anton, *Elementary Linear Algebra*, 7th ed. New York: John Wiley and Sons, Inc., 1994.

[9] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, Eds., *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. Philadelphia: SIAM, 2000.

[10] A. Barabell, "Improving the Resolution Performance of Eigenstructure-Based Direction-Finding Algorithms," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1983)*, vol. 8, Boston, MA, Apr. 1983, pp. 336–339.

[11] A. Bell, "Information Theory, Independent Component Analysis and Applications," in *Unsupervised Adaptive Filtering, Volume I: Blind Source Separation*, S. Haykin, Ed. John Wiley and Sons, Inc., 2000, ch. 6, pp. 237–264.

[12] A. Bell and T. Sejnowski, "An Information-Maximization Approach to Blind Separation and Blind Deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[13] R. Bellman, *Introduction to Matrix Analysis*, 2nd ed., ser. Classics in Applied Mathematics. Philadelphia: SIAM, 1997.

[14] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A Blind Source Separation Technique using Second Order Statistics," *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 434–444, Feb. 1997.

[15] A. Benjamin and J. Quinn, *Proofs that Really Count: The Art of Combinatorial Proof*, ser. Dolciani Mathematical Expositions. The Mathematical Association of America, 2003.

[16] G. Bienvenu and L. Kopp, "Optimality of High Resolution Array Processing using the Eigensystem Approach," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, no. 5, pp. 1235–1248, Oct. 1983.

[17] L. Blum, F. Cucker, M. Shub, and S. Smale, *Complexity and Real Computation*. Springer-Verlag New York, Inc., 1997.

[18] P. Bofill and M. Zibulevsky, "Blind Separation of More Sources Than Mixtures using Sparsity of Their ShortTime Fourier Fransform," in *Proc. Int. Workshop on Independent Component Analysis and Blind Source Separation (ICA 2000)*, Helsinki, Finland, June 2000, pp. 33–38.

[19] ——, "Underdetermined Blind Source Separation using Sparse Representations," *Signal Processing, Elsevier*, vol. 81, no. 11, pp. 2353–2362, 2001.

[20] D. Brandwood, "A Complex Gradient Operator and its Application in Adaptive Array Theory," *IEE Proceedings*, vol. 130, no. 1, pp. 11–16, Feb. 1983.

[21] J. Brewer, "Kronecker Products and Matrix Calculus in System Theory," *IEEE Trans. Circuits Syst.*, vol. 25, no. 9, pp. 772–781, Sept. 1978.

[22] D. Brillinger, *Time Series: Data Analysis and Theory*. Holt, Rinehart and Winston, 1975.

[23] A. Bronstein, M. Bronstein, M. Zibulevsky, and Y. Zeevi, "Separation of Reflections via Sparse ICA," in *Proc. Int. Conf. on Image Processing (ICIP 2003)*, Barcelona, Spain, Sept. 2003, pp. 313–316.

[24] R. Buccigrossi. (1997, Mar.) Breaking Up is Hard to Do: Blind Signal Separation. University of Pennsylvania, Department of Computer and Information Science. Philadelphia PA. [Online]. Available: http://www.tcg.com/WPEII.ps

[25] A. Bunse-Gerstner, R. Byers, and V. Mehrmann, "Numerical Methods for Simultaneous Diagonalization," *SIAM Journal on Matrix Analysis and Applications*, vol. 14, no. 4, pp. 927–949, 1993.

[26] J.-F. Cardoso, "Source Separation using Higher Order Moments," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1989)*, vol. 4, Glasgow, UK, 1989, pp. 2109–2112.

[27] ——, "Eigen-Structure of the Fourth-Order Cumulant Tensor with Application to the Blind Source Separation Problem," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1990)*, Albuquerque, New Mexico, USA, Apr. 1990, pp. 2655–2658.

[28] ——, "Higher-Order Narrow-Band Array Processing," in *Proc. Int. Workshop on Higher-Order Stat.*, Chamrousse, France, 1991, pp. 121–130.

[29] ——, "Super-Symmetric Decomposition of the Fourth-Order Cumulant Tensor. Blind Identification of More Sources than Sensors," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1991)*, vol. 5, Toronto, Canada, 1991, pp. 3109–3112.

[30] ——, "Iterative Techniques for Blind Source Separation using only Fourth-Order Cumulants," in *Proc. EUSIPCO*, Brussels, Belgium, 1992, pp. 739–742.

[31] ——, "On the Performance of Orthogonal Source Separation Algorithms," in *Proc. EUSIPCO*, Edinburgh, Scotland, 1994, pp. 776–779.

[32] ——, "A Tetradic Decomposition of 4th-Order Tensors: Application to the Source Separation Problem," in *Algorithms, Architectures and Applications*, ser. SVD and Signal Processing, M. Moonen and B. de Moor, Eds. Elsevier, 1995, vol. III, pp. 375–382.

[33] ——, "Infomax and Maximum Likelihood for Source Separation," *IEEE Signal Processing Lett.*, vol. 4, no. 4, pp. 112–114, Apr. 1997.

[34] ——, "Blind Signal Separation: Statistical Principles," *Proc. IEEE*, vol. 9, no. 10, pp. 2009–2025, 1998.

[35] ——, "High-Order Contrasts for Independent Component Analysis," *Neural Computation*, vol. 11, no. 1, pp. 157–192, 1999.

[36] J.-F. Cardoso and P. Comon, "Independent Component Analysis, A Survey of Some Algebraic Methods," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS 1996)*, vol. 2, Atlanta, May 1996, pp. 93–96.

[37] J.-F. Cardoso, H. Snoussi, J. Delabrouille, and G. Patanchon, "Blind Separation of Noisy Gaussian Stationary Sources. Application to Cosmic Microwave Background Imaging," in *Proc. EUSIPCO*, Toulouse, France, 2002, pp. 561–564.

[38] J.-F. Cardoso and A. Souloumiac, "Blind Beamforming for Non Gaussian Signals," *IEE Proceedings F*, vol. 140, no. 6, pp. 362–370, 1993.

[39] ——, "Jacobi Angles for Simultaneous diagonalization," *SIAM Journal on Matrix Analysis and Applications*, vol. 17, no. 1, pp. 161–164, 1996.

[40] R. Challa and S. Shamsunder, "3-D Spherical Localization of Multiple Non-Gaussian Sources using Cumulants," in *Proc. 8th IEEE Signal Processing Workshop on Statistical Signal and Array Processing*, Corfu, Greece, June 1996, pp. 101–104.

[41] C. Chang, Z. Ding, S. Yau, and F. Chan, "A Matrix-Pencil Approach to Blind Separation of Colored Nonstationary Signals," *IEEE Trans. Signal Processing*, vol. 48, no. 3, pp. 900–907, 2000.

[42] S. Choi and A. Cichocki, "Blind Separation of Nonstationary and Temporally Correlated Sources from Noisy Mixtures," in *Proc. IEEE Workshop on Neural Networks for Signal Processing (NNSP 2000)*, Sydney, Australia, Dec. 2000, pp. 405–414.

[43] ——, "Blind Separation of Nonstationary Sources in Noisy Mixtures," *Electronics Letters*, vol. 36, no. 9, pp. 848–849, Apr. 2000.

[44] S. Choi, A. Cichocki, and S. Amari, "Equivariant Nonstationary Source Separation," *Neural Networks*, vol. 15, no. 1, pp. 121–130, Jan. 2002.

[45] S. Choi, A. Cichocki, and A. Belouchrani, "Blind Separation of Second-Order Nonstationary and Temporally Colored Sources," in *Proc. IEEE Workshop on Statistical Signal Processing*, Singapore, 2001, pp. 444–447.

[46] ——, "Second Order Nonstationary Source Separation," *Journal of VLSI Signal Processing*, vol. 32, no. 1-2, pp. 93–104, Aug. 2002.

[47] A. Cichocki and S.-I. Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*. John Wiley and Sons, Inc., 2002.

[48] 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*, vol. 12, pp. 91–98, 2000.

[49] A. Cichocki and R. Unbehauen, "Robust Neural Networks with On-Line Learning for Blind Identification and Blind Separation of Sources," *IEEE Trans. Circuits Syst.*, vol. 43, no. 11, pp. 894–906, 1996.

[50] A. Cichocki, R. Unbehauen, L. Moszczynski, and E. Rummert, "A New On-Line Adaptive Algorithm for Blind Separation of Source Signals," in *Proc. Int. Symp. on Artificial Neural Networks (ISANN 1994)*, Tainan, Taiwan, 1994, pp. 406–411.

[51] A. Cichocki, R. Unbehauen, and E. Rummert, "Robust Learning Algorithm for Blind Separation of Signals," *Electronics Letters*, vol. 30, no. 17, pp. 1386–1387, 1994.

[52] P. Comon, "Independent Component Analysis, A New Concept?" *Signal Processing, Elsevier*, vol. 36, no. 3, pp. 287–314, Apr. 1994.

[53] ——, "Tensor Decompositions: State of the Art and Applications," in *Mathematics in Signal Processing*, J. McWhirter and I. Proudler, Eds. Oxford, UK: Oxford University Press, 2001, vol. V, pp. 1–24.

[54] P. Comon and P. Chevalier, "Blind Source Separation: Models, Concepts, Algorithms and Performance," in *Unsupervised Adaptive Filtering, Volume I: Blind Source Separation*, S. Haykin, Ed. John Wiley and Sons, Inc., 2000, ch. 5, pp. 191–236.

[55] P. Comon, C. Jutten, and J. Hérault, "Blind Separation of Sources, Part II: Problems Statement," *Signal Processing, Elsevier*, vol. 24, pp. 11–20, 1991.

[56] P. Comon and B. Mourrain, "Decomposition of Quantics in Sums of Powers of Linear Forms," *Signal Processing, Elsevier*, vol. 53, no. 2, pp. 93–107, Sept. 1996.

[57] R. Compton, *Adaptive Antennas: Concepts and Performance*. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1988.

[58] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley and Sons, Inc., 1991.

[59] D. Cox, J. Little, and D. O'Shea, *Ideals, Varietes, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 2nd ed. Springer-Verlag New York, Inc., 1996.

[60] J. de Graaf, "Tensorrekening en Differentiaalmeetkunde," Lecture Notes (in Dutch), Technische Universiteit Eindhoven, Faculteit Wiskunde en Informatica, Eindhoven, The Netherlands, 2002.

[61] L. de Lathauwer, "Signal Processing Based on Multilinear Algebra," Ph.D. dissertation, Katholieke Universiteit Leuven, Leuven, België, 1997.

[62] J. Deller, Ed., *Array Processing*, ser. IEEE Signal Processing Mag., vol. 13, no. 4. IEEE, July 1996.

[63] S. Douglas, "Blind Signal Separation and Blind Deconvolution," in *Handbook of Neural Networks Signal Processing*, Y.-H. Hu and J.-N. Hwang, Eds. Boca Raton, FL: CRC Press, 2001, ch. 7.

[64] ——, "Blind Source Separation and Independent Component Analysis: A Crossroads of Tools and Ideas," in *Proc. 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA 2003)*, Nara, Japan, Apr. 2003, pp. 1–10.

[65] S. Douglas and S.-I. Amari, "Natural-Gradient Adaptation," in *Unsupervised Adaptive Filtering, Volume I: Blind Source Separation*, S. Haykin, Ed. John Wiley and Sons, Inc., 2000, ch. 2, pp. 13–62.

[66] S. Ellingson and W. Cazemier, "Efficient Multibeam Synthesis with Interference Nulling for Large Arrays," *IEEE Trans. Antennas Propagat.*, vol. 51, no. 3, pp. 503–511, Mar. 2003.

[67] S. Ellingson and G. Hampson, "A Subspace-Tracking Approach to Interference Nulling for Phased Array-Based Radio Telescopes," *IEEE Trans. Antennas Propagat.*, vol. 50, no. 1, pp. 25–30, Jan. 2002.

[68] H. Farid and E. Adelson, "Separating Reflections and Lighting Using Independent Components Analysis," in *Proc. IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, Fort Collins, CO, 1999, pp. 262–267.

[69] W. Feller, *Probability Theory and Its Applications*, 3rd ed. John Wiley and Sons, Inc., 1968.

[70] M. Gaeta and J.-L. Lacoume, "Source Separation Without Prior Knowledge: The Maximum Likelihood Solution," in *Proc. EUSIPCO*, 1990, pp. 621–624.

[71] S. V. Gerven and D. van Compernolle, "On the use of Decorrelation in Scalar Signal Separation," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1994)*, Adelaide, Australia, 1994, pp. 57–60.

[72] G. Golub and C. van Loan, *Matrix computations*, 3rd ed. The Johns Hopkins university press, Baltimore and London, 1996.

[73] A. Graham, *Kronecker Products and Matrix Calculus With Applications*. New York, USA: John Wiley and Sons, Inc., 1981.

[74] R. Grimaldi, *Discrete and Combinatorial Mathematics: An Applied Introduction*, 3rd ed. Addison-Wesley, 1994.

[75] S. Haykin, Ed., *Unsupervised Adaptive Filtering, Volume 1: Blind Source Separation*, ser. Wiley Series on Adaptive and Learning Systems for Signal Processing. John Wiley and Sons, Inc., 2000, vol. 1.

[76] S. Haykin, *Adaptive Filter Theory*, 4th ed., ser. Prentice Hall Information and System Sciences Series, H. Kailath, Ed. Upper Saddle River, New Jersey: Prentice-Hall Inc., 2002, prentice-Hall Inc.

[77] R. Horn, "The Hadamard Product," in *Matrix Theory and Applications*, ser. Proc. of Symp. Appl. Math., C. Johnson, Ed., vol. 40. Providence, Rhode Island: American Mathematical Society, 1990, pp. 87–169.

[78] A. Hyvärinen, "New Approximations of Differential Entropy for Independent Component Analysis and Projection Pursuit," in *Advances in Neural Information Processing Systems*, M. Jordan, M. Kearns, and S. Solla, Eds. Cambridge, MA, 1996: MIT Press, 1997, vol. 10, pp. 273–279.

[79] A. Hyvärinen, "Survey on Independent Component Analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.

[80] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, ser. Wiley Series on Adaptive and Learning Systems for Signal Processing, S. Haykin, Ed. New York: John Wiley and Sons, Inc., 2001.

[81] ——, *Independent Component Analysis*, ser. Wiley Series on Adaptive and Learning Systems for Signal Processing. New York: John Wiley and Sons, Inc., 2001, ch. 8.

[82] ——, *Independent Component Analysis*, ser. Wiley Series on Adaptive and Learning Systems for Signal Processing. New York: John Wiley and Sons, Inc., 2001, ch. 5.

[83] A. Hyvärinen and E. Oja, "Independent Component Analysis: Algorithms and Applications," *Neural Networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[84] D. Johnson and D. Dudgeon, *Array Signal Processing: Concepts and Techniques*. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1993.

[85] M. Joho, R. H. Lambert, and H. Mathis, "Elementary Cost Functions for Blind Separation of Non-Stationary Source Signals," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2001)*, Salt Lake City, UT, May 2001, pp. 2793–2796.

[86] M. Joho and H. Mathis, "Joint Diagonalization of Correlation Matrices by using Gradient Methods with Application to Blind Signal Separation," in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2002)*, Washington D.C., USA, Aug. 2002, pp. 273–277.

[87] M. Joho and K. Rahbar, "Joint Diagonalization of Correlation Matrices by using Newton Methods with Application to Blind Signal Separation," in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2002)*, Washington D.C., USA, Aug. 2002, pp. 403–407.

[88] I. Jolliffe, *Principal Component Analysis*. Springer-Verlag, New York, 1986.

[89] M. Jones and R. Sibson, "What is Projection Pursuit?" *Journal of the Royal Statistical Society, ser. A*, vol. 150, pp. 1–36, 1987.

[90] J. Justice, N. Owsley, and J. Yen, *Array Signal Processing*, S. Haykin, Ed. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1985.

[91] C. Jutten and J. Hérault, "Blind Separation of Sources, part I: An Adaptive Algorithm based on Neuromimetic Architecture," *Signal Processing, Elsevier*, vol. 24, pp. 1–10, 1991.

[92] C. Jutten and A. Taleb, "Source Separation: From Dusk Till Dawn," in *Proc. Int. Workshop on Independent Component Analysis and Blind Source Separation (ICA 2000)*, Helsinki, Finland, June 2000, pp. 15–26.

[93] M. Kaveh and A. Barabell, "The Statistical Performance of the MUSIC and the Minimum-Norm Algorithms in Resolving Plane Waves in Noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 2, pp. 331–341, Apr. 1986.

[94] S. Kay, *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*, ser. Prentice Hall Signal Processing Series, A. Oppenheim, Ed. Prentice Hall Inc., 1998.

[95] C. Khatri and C. Rao, "Solutions to Some Functional Equations and Their Applications to Characterization of Probability Distributions," *Sankhya: The Indian Journal of Statistics, Series A, Part 2*, vol. 30, pp. 167–180, 1968.

[96] K. Kreutz-Delgado and B. Rao, "Sparse Basis Selection, ICA and Majorization: Towards a Unified Perspective," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1999)*, San Diego, La Jolla, CA, USA, Mar. 1999, pp. 1081–1084.

[97] H. Krim and M. Viberg, "Two Decades of Array Signal Processing Research: the Parametric Approach," *IEEE Signal Processing Mag.*, vol. 13, no. 4, pp. 67–94, July 1996.

[98] R. Kumaresan, "On the Zeros of the Linear Prediction-Error Filter for Deterministic Signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, no. 1, pp. 217–220, Feb. 1983.

[99] R. Kumaresan and D. Tufts, "Estimating the Angles of Arrival of Multiple Plane Waves," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 19, pp. 134–139, Jan. 1983.

[100] T.-W. Lee, M. Lewicki, M. Girolami, and T. Sejnowski, "Blind Source Separation of More Sources Than Mixtures using Overcomplete Representations," vol. 6, no. 4, 1999, pp. 87–90.

[101] A. Leshem and A.-J. van der Veen, "Radio-Astronomical Imaging in the Presence of Strong Radio Interference," *IEEE Trans. Inform. Theory*, vol. 46, no. 5, pp. 1730–1747, Aug. 2000.

[102] M. Lewicki and T. Sejnowski, "Learning Overcomplete Representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.

[103] F. Li and R. Vaccaro, "Analysis of Min-Norm and MUSIC with Arbitrary Array Geometry," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 126, no. 6, pp. 976–985, Nov. 1990.

[104] F. Li, R. Vaccaro, and D. Tufts, "Unified Performance Analysis of Subspace-Based Estimation Algorithms," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1990)*, vol. 5, Albuquerque, New Mexico, Apr. 1990, pp. 2575–2578.

[105] T. Li, "Numerical Solution of Multivariate Polynomial Systems by Homotopy Continuation Methods," *Acta Numerica*, no. 6, pp. 399–436, 1997.

[106] U. Lindgren and A.-J. van der Veen, "Source Separation based on Second Order Statistics - An Algebraic Approach," in *Proc. IEEE SP Workshop on Stat. Signal and Array Proc.*, Corfu, Greece, June 1996, pp. 324–327.

[107] U. Lindgren, H. Sahlin, and H. Broman, "Source Separation using Second Order Statistics," in *Proc. EUSIPCO*. Trieste, Italy: Elsevier, Sept. 1996.

[108] D. Luenberger, *Optimization by Vector Space Methods*. John Wiley and Sons, Inc., 1969.

[109] ——, *Linear and Nonlinear Programming*, 2nd ed. Addison-Wesley, 1984.

[110] K. Matsuoka, M. Ohya, and M. Kawamoto, "A Neural Net for Blind Separation of Nonstationary Signals," *Neural Networks*, vol. 8, no. 3, pp. 411–419, 1995.

[111] P. McCullagh, *Tensor Methods in Statistics*, ser. Monographs on Statistics and Applied Probability. Chapman and Hall, 1987.

[112] J. Mendel, "Tutorial on Higher-Order Statistics (Spectra) in Signal Processing and System Theory: Theoretical Results and Some Applications," *Proc. IEEE*, vol. 79, no. 3, pp. 278–305, Mar. 1991.

[113] G. K. Miley, "Powerful Radio Sources: Westerbork and Beyond," in *The Westerbork Observatory, Continuing Adventure in Radio Astronomy*, E. Raimond and R. O. Genee, Eds. Dordrecht, The Netherlands: Kluwer, 1996.

[114] L. Molgedey and H. Schuster, "Separation of Independent Signals using Time-Delayed Correlations," *Physical Review Letters*, vol. 72, no. 23, pp. 3634–3637, June 1994.

[115] T. Moon and W. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, 2000.

[116] A. Morgan, *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems*. Englewood Cliffs, New Jersey: Prentice Hall, 1987.

[117] B. Mourrain, "Solving Polynomial Systems by Matrix Computations," INRIA Sophia-Antipolis, France, 1997, pp. 1–25.

[118] ——, "An Introduction to Linear Algebra Methods for Solving Polynomial Equations," in *HERCMA 1998*, 1999, pp. 179–200. [Online]. Available: ftp://ftp-sop.inria.fr/saga/mourrain/9806-BM-hercma.ps.gz

[119] P. Naidu, *Sensor Array Signal Processing*. London: CRC Press, 2001.

[120] R. Nielsen, *Sonar Signal Processing*. Norwood, MA: Artech House, 1991.

[121] N. Owsley, "Sonar Array Processing," S. Haykin, Ed. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1985, ch. 3.

[122] A. Papoulis, *Probability, Random variables, and Stochastic Processes*. McGraw-Hill, 1984.

[123] S.-C. Pei and M.-H. Yeh, "An Introduction to Discrete Dinite Frames," *IEEE Signal Processing Mag.*, vol. 14, no. 6, pp. 84–96, Nov. 1997.

[124] D. Pham, "Joint Approximate Diagonalization of Positive Definite Hermitian Matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 22, no. 4, pp. 1136–1152, 2001.

[125] D. Pham and J.-F. Cardoso, "Blind Separation of Instantaneous Mixtures of Nonstationary Sources," *IEEE Trans. Signal Processing*, vol. 9, no. 9, pp. 1837–1848, 2001.

[126] S. Pillai, *Array Signal Processing*, C. Burrus, Ed. Springer-Verlag New York, Inc., 1989.

[127] V. Pisarenko, "The Retrieval of Harmonics from a Covariance Function," *Geophysics J. Royal Astron. Soc.*, vol. 33, pp. 347–366, 1973.

[128] K. Pope and R. Bogner, "Blind Signal Separation I - Linear, Instantaneous Combinations," *Digital Signal Processing*, vol. 6, no. 1, pp. 5–16, Jan. 1996.

[129] B. Porat and B. Friedlander, "Direction Finding Algorithms Based on High-Order Statistics," *IEEE Trans. Signal Processing*, vol. 39, no. 9, pp. 2016–2024, Sept. 1991.

[130] M. Priestley, *Spectral Analysis and Time Series*. Academic Press, 1981.

[131] B. Rao and K. Hari, "Performance Analysis of Root-Music," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 12, pp. 1939–1949, Dec. 1989.

[132] ——, "Statistical Performance Analysis of the Minimum-Norm Method," *Radar and Signal Processing, IEE Proceedings F*, vol. 136, no. 3, pp. 125–134, 1989.

[133] B. Rao and K. Kreutz-Delgado, "An Affine Scaling Methodology for Best Basis Selection," *IEEE Trans. Signal Processing*, vol. 47, no. 1, pp. 187–200, Jan. 1999.

[134] K. Rosen, Ed., *Handbook Of Discrete And Combinatorial Mathematics*. CRC Press, 2000.

[135] M. Ryle, "The 5-km Telescope at Cambridge," *Nature*, vol. 239, pp. 435–438, Oct. 1973.

[136] R. Schilizzi, "The Square Kilometre Array," in *Proc. of the SPIE Conference on Astronomical Telescopes and Instrumentation*, vol. 5489. Glasgow, Scotland: The International Society for Optical Engineering, June 2004.

[137] R. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation," in *Proc. RADC Spectrum Estimation Workshop*, Griffiths AFB, Rome, New York, 1979, pp. 243–258.

[138] ——, "A Signal Subspace Approach to Multiple Emitter Location and Spectral Estimation," Ph.D. dissertation, Standford University, USA, 1981.

[139] D. Schobben, "Efficient Adaptive Multi-Channel Concepts in Acoustics," Ph.D. dissertation, Eindhoven university of technology, Eindhoven, the Netherlands, 1999.

[140] S. Shamsunder and G. Giannakis, "Modeling of Non-Gaussian Array Data using Cumulants: DOA Estimation of More Sources with Less Sensors," *Signal Processing, Elsevier*, vol. 30, no. 3, pp. 279–297, 1993.

[141] N. D. Sidiropoulos and R. Bro, "On the uniqueness of Multilinear Decomposition of N-way Arrays," *Journal of chemometrics*, vol. 14, pp. 229–239, 2000.

[142] F. Smits, "The Square Kilometre Array Interferometer - The Radio Telescope for the 21st Century," in *Proc. Symp. Toekomst Zonder Grenzen*. Eindhoven, The Netherlands: Netherlands Foundation for Research in Astronomy, Mar. 1998.

[143] A. Sommese, J. Verschelde, and C. Wampler, "Introduction to Numerical Algebraic Geometry," *Report*, pp. 1–46, Nov. 2003.

[144] A. Sommese and C. Wampler, *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. World Scientific, 2005.

[145] E. Sorouchyari, "Blind Separation of Sources, Part III: Stability Analysis," *Signal Processing, Elsevier*, vol. 24, pp. 21–29, 1991.

[146] H. Stetter, "Matrix Eigenproblems are at the Heart of Polynomial System Solving," *ACM SIGSAM Bulletin*, vol. 30, no. 4, pp. 22–25, Dec. 1997.

[147] P. Stoica and R. Moses, *Introduction to Spectral Analysis*. Upper Saddle River, New Jersey: Prentice Hall, Inc., 1997.

[148] P. Stoica and A. Nehorai, "MUSIC, Maximum Likelihood, and Cramer-Rao Bound," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[149] ——, "MUSIC, Maximum Likelihood, and Cramer-Rao Bound: Further Results and Comparisons," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 12, pp. 2140–2150, Dec. 1990.

[150] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Brooks Cole, 1988.

[151] N. D. S. T. Jiang and J. M. F. ten Berge, "Almost-Sure Identifiability of Multidimensional Harmonic Retrieval," *IEEE Trans. Signal Processing*, vol. 48, no. 3, pp. 1849–1859, 2000.

[152] F. Theis and E. Lang, "Formalization of the Two-Step Approach to Overcomplete BSS," in *Proc. 4th Int. Conf. on Signal and Image Processing (SIP 2002)*, N. Younan, Ed., Hawaii, 2000, pp. 207–212.

[153] ——, "Geometric Overcomplete ICA," in *Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN 2002)*, Bruges, Belgium, Apr. 2002, pp. 217–222.

[154] C. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Prentice-Hall Inc., 1992.

[155] A. Thompson, B. Clark, C. Wade, and P. Napier, "The Very Large Array," *Astrophys. J. Suppl.*, vol. 44, pp. 151–167, Apr. 1980.

[156] L. Tong, Y. Inouye, and R. Liu, "Waveform-Preserving Blind Estimation of Multiple Independent Sources," *IEEE Trans. Signal Processing*, vol. 41, no. 7, pp. 2461–2470, July 1993.

[157] L. Tong, R. Liu, V. Soon, and Y. Huang, "Indeterminacy and Identifiability of Blind Identification," *IEEE Trans. Circuits Syst.*, vol. 38, no. 5, pp. 499–509, May 1991.

[158] L. Tong, V. Soon, Y. Huang, and R. Liu, "AMUSE: A New Blind Identification Algorithm," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS 1990)*, vol. 3, New Orleans, LA, 1990, pp. 1784–1787.

[159] R. Tweg and B. Porat, "Numerical Optimization Method for Array Signal Processing," *IEEE Trans. Signal Processing*, vol. 35, no. 2, pp. 549–565, Apr. 1999.

[160] A. van Ardenne, "Phased Arrays of Large Antennas: R&D Aspects," in *Proc. 3rd meeting of the Large Tel. Working Group*. Guizhou, China: Int. Acad. Publ., Dec. 1995, pp. 40–50.

[161] J. van de Laar, "Instantaneous Blind Source Separation based on the Exploitation of Temporal Correlations and Nonstationarity," in *Proc. 14th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2003)*, Veldhoven, The Netherlands, Nov. 2003, pp. 391–398.

[162] ——, "On MIMO Instantaneous Blind Identification based on the Exploitation of the Time Structure of Signals using Arbitrary-Order Cumulants," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2004)*, Montreal, Quebec, Canada, May 2004, pp. 549–552.

[163] ——, "TIME-MUSIC DOA Estimation based on the Exploitation of Some Arbitrary-Order Temporal Structure in the Data," in *Proc. 3rd IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2004)*, Barcelona, Spain, July 2004, pp. 308–312.

[164] ——, "ULA TIME-MUSIC DOA Estimation based on the Exploitation of Some Arbitrary-Order Temporal Structure in the Data," in *Proc. 4th IEEE Benelux Signal Processing Symposium (SPS 2004)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 179–182.

[165] ——, "MIMO Instantaneous Blind Identification based on Second Order Temporal Structure and Homotopy Method," in *Proc. 9th International Workshop on Acoustic Echo and Noise Control (IWAENC 2005)*, Eindhoven, The Netherlands, Sept. 2005, pp. 241–244.

[166] J. van de Laar, M. Moonen, and P. Sommen, "MIMO Instantaneous Blind Identification Based on Second Order Temporal Structure," *IEEE Trans. Signal Processing*, submitted for publication in Aug. 2007.

[167] J. van de Laar and P. Sommen, "Closed-form Instantaneous Blind Signal Separation based on Second Order Statistics and Non-Stationarity," in *Proc. 13th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2002)*, Veldhoven, The Netherlands, Nov. 2002, pp. 334–338.

[168] ——, "Exploiting Non-Stationarity in Multi-Channel Acoustic Echo Cancellation," in *Proc. 3rd IEEE Benelux Signal Processing Symposium (SPS 2002)*, Leuven, Belgium, Mar. 2002, pp. 9–12.

[169] ——, "Performance Analysis of Multi-Channel Adaptive Algorithms Operating in a Non-Stationary Acoustic Teleconferencing Setup," in *Proc. 2nd IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2002)*, Washington D.C., USA, Aug. 2002, pp. 234–238.

[170] ——, "Conditions on Sources and Mixing Matrix for Solving the Permutation Indeterminacy in 2×2 Instantaneous Blind Signal Separation," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003)*, Hong Kong, China, Apr. 2003, pp. 317–320.

[171] A.-J. van der Veen, "Algebraic Methods for Deterministic Blind Beamforming," *Proc. IEEE*, vol. 86, no. 10, pp. 1987–2008, Oct. 1998.

[172] ——, "Joint Diagonalization via Subspace Fitting Techniques," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2001)*, Salt Lake City, UT, May 2001, pp. 2773–2776.

[173] A.-J. van der Veen, E. Deprettere, and A. Swindlehurst, "Subspace Based Signal Analysis using Singular Value Decomposition," *Proc. IEEE*, vol. 81, no. 9, pp. 1277–1308, Sept. 1993.

[174] A.-J. van der Veen, A. Leshem, and A.-J. Boonstra, "Signal Processing for Radio Astronomical Arrays," in *Sensor Array and Multichannel Signal Processing Workshop Proceedings*, Delft, Netherlands, July 2004, pp. 1–10.

[175] A.-J. van der Veen, S. Talwar, and A. Paulraj, "A Subspace Approach to Blind Space-Time Signal Processing for Wireless Communication Systems," *IEEE Trans. Signal Processing*, vol. 45, no. 1, pp. 173–190, Jan. 1997.

[176] H. van der Vorst and G. Golub, "150 Years Old and Still Alive: Eigenproblems," vol. 63, 1997, pp. 93–120.

[177] S. van Gerven and D. van Compernolle, "Signal Separation in a Symmetric Adaptive Noise Canceler by Output Decorrelation," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 1992)*, Adelaide, Australia, 1992, pp. 221–224.

[178] ——, "Signal Separation by Symmetric Adaptive Decorrelation: Stability, Convergence, and Uniqueness," *IEEE Trans. Signal Processing*, vol. 43, no. 7, pp. 1602–1612, July 1995.

[179] H. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. New York, USA: John Wiley and Sons, Inc., 2002.

[180] B. Van Veen and K. Buckley, "Beamforming: A Versatile Approach to Spatial Filtering," *IEEE Signal Processing Mag.*, vol. 5, no. 2, pp. 4–24, Apr. 1988.

[181] M. Vanderveen, A.-J. van der Veen, and A. Paulraj, "Estimation of Multipath Parameters in Wireless Communications," *IEEE Trans. Signal Processing*, vol. 46, no. 3, pp. 682–690, Mar. 1998.

[182] J. Verschelde, "Homotopy Continuation Methods for Solving Polynomial Systems," Ph.D. dissertation, Katholieke Universiteit Leuven, Leuven, België, 1996.

[183] M. Viberg, "Subspace Fitting Concepts in Sensor Array Processing," Ph.D. dissertation, Linköping University, Linköping, Sweden, Oct. 1989.

[184] M. Viberg, B. Ottersten, and T. Kailath, "Detection and Astimation in Sensor Arrays using Weighted Subspace Fitting," *IEEE Trans. Signal Processing*, vol. 39, no. 11, pp. 2436–2449, Nov. 1991.

[185] H.-C. Wu and J. Principe, "A Unifying Criterion for Blind Source Separation and Decorrelation: Simultaneous Diagonalization of Correlation Matrices," in *Proc. Workshop on Neural Networks for Signal Processing (NNSP 1997)*, Amelia Island, FL USA, 1997, pp. 496–505.

[186] J. Yen, "Image Reconstruction in Synthesis Radio Telescope Arrays," S. Haykin, Ed. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1985, ch. 5.

[187] A. Yeredor, "Approximate Joint Diagonalization Using Non-Orthogonal Matrices," in *Proc. Int. Workshop on Independent Component Analysis and Blind Source Separation (ICA 2000)*, Helsinki, Finland, June 2000, pp. 33–38.

[188] ——, "Non-Orthogonal Joint Diagonalization in the Least-Squares Sense with Application in Blind Source Separation," *IEEE Trans. Signal Processing*, vol. 50, no. 7, pp. 1545–1553, July 2002.

[189] V. Zarzoso, "Closed-Form Higher-Order Estimators for Blind Separation of Independent Source Signals in Instantaneous Linear Mixtures," Ph.D. dissertation, The University of Liverpool, Liverpool, United Kingdom, 1999.

[190] S. Zhenghao and F. Fairman, "Harmonic Retrieval via State Space and Fourth-Order Cumulants," *IEEE Trans. Signal Processing*, vol. 42, no. 5, pp. 1109–1119, 1994.

[191] A. Ziehe, P. Laskov, G. Nolte, and K.-R. Müller, "A Fast Algorithm for Joint Diagonalization with Non-orthogonal Transformations and its Application to Blind Source Separation," *Journal of Machine Learning Research*, vol. 5, pp. 777–800, July 2004.

Index

spatial
  aliasing, 88
2D-scenario, 84

AC-DC algorithm, 81
  Alternating Columns phase, 81
  Diagonal Centers phase, 81
activation function, 53
algebraic algorithms
  based on cross-cumulants, 62–71
algorithm
  AMUSE, 74–75
  based on information-theoretic measure, 53–54
  based on kurtosis, 62
  based on nonlinear de-correlation, 59–61
  Cichocki-Unbehauen, 60–61
  for producing subspace algorithms, 128
  for producing subspace based algorithms, 83
  Hérault-Jutten, 59–60
  ingredients of subspace algorithm, 129
  Molgedey and Schuster, 73–74
  on-line, 54
  roller-coaster, 98
  SOBI, 75
aliasing
  spatial, 88
ALLPASS(1)
  model, 384
  temporal structure, 384
Amari, 2
ambiguities, *see* indeterminacies, 4, 133, 217
  DOA estimation, 88
approximate average matrix decomposition, 79
AR(1)
  model, 135, 223, 343
  temporal structure, 6, 135
AR(2)
  model, 376
  temporal structure, 376
arrangement
  of sensor correlation values, 139
  of sensor statistics in matrix, 83, 112, 127
array
  aperture, 85
  cylindrical, 15
  hydrophone, 15
  planar, 84, 370
  response, 84–89
  response matrix, 85
  response vector, 85
  towed, 15
  unambiguous, 86, 88
array aperture, 10, 13
array geometry, 128
array manifold, 86
  generalized, 114, 117
  parameterization, 86
array response matrix, 14, 85, 133, 153, 217
  generalized, 117, 254, 326
    unambiguous, 117
  unambiguous, 89
array response vector, 12, 85, 93, 133, 153, 217
  generalized, 117, 254, 326
array response vectors
  generalized, 119
Array Signal Processing, 1, 83
ASP, 83
assignment
  symbol for, xxi
assumptions
  MIBI, 220, 299
  on mixing matrix
    HOS, 112–114
  on mixing system, 31
  on noise signals, 33
    HOS, 112–114
  on source signals, 32
    HOS, 112–114

auto-cumulant, 439
auto-cumulant function, 439
auto-moment, 436
auto-moment function, 438
auto-statistics, 33
autoregressive, 6, 135, 376
azimuth, 11, 13

baseband, 12
beamforming, 2
Bell, 2
Belouchrani, 75
Bienvenu, 83, 97
bijective, 88, 148, 151, 251, 324
bijective mapping, 25
    between function-valued vectors and matrices, 25, 150, 251, 323
    between functions and vectors, 25
binomial coefficient, 410
bivariate, 138
blind, 1, 3
blind methods, 30
blind problems
    overcomplete, 18
    underdetermined, 18, 34
Blind Signal Extraction, 48, 61
Blind Signal Processing, 2, 29, 39, 48, 82
blind signal processing
    instantaneous, 29
Blind Signal Separation, 5
Brillinger, 18, 126
BSE, 48, 61
BSP, 2, 29

Cardoso, 2, 64, 68
carrier, 12, 85
    angular frequency, 85
    frequency, 85
    wavelength, 85
CBSS, 6
Central Limit Theorem
    application to ICA, 48
    definition of, 433
    non-Gaussianity, 16, 48
central moment
    univariate, 423
central moments
    joint, *see* multivariate
    multivariate, 438–439
    univariate, 424
characteristic functions
    multivariate, 433–434
    univariate, 420
Cholesky decomposition, 44
choose, 410
Cichocki, 2, 60
Cichocki-Unbehauen algorithm, 60–61
CLT, 48
coloredness, 7
column space, 448
Comon, 42, 64
complement
    orthogonal, 93
complex circular, 21
complex circular Gaussian, 99, 113, 117, 125, 377, 379, 385
complex circular uniform, 124
complex envelope, 12
compound set, 238, 239, 310, 312
cone, 172, 269, 342
    elliptic, 272, 279
cones, 23
conjugate
    of row, 448
conjugation
    pair, 415
    pair, 218
    pattern, 21, 218, 297
    tuple, 113–115, 125, 218, 297, 408, 413, 415
constrast function, 41–43
continuation, 199
continuation method, 200
continuation parameter, 200, 459, 460, 462
    generalized, 464
coordinate transformation, 103, 397
coplanar, 84
corrector, 200, 201, 460
    Newton, 204
correlatedness, 2, 7, 27
    temporal, 16
correlation
    nonlinear, 57–59
cost function, 41, 62, 78, 190, 191, 193

for DOA estimation, 96, 122
criterion
ICA, 47
identification, 30
separation, 30
cross-cumulant, 43, 439
cross-cumulant function, 439
cross-cumulants
as measures of statistical independence, 62
joint, 62
cross-moment, 436
cross-moment function, 438
cross-structure
temporal, 221
cubic form, 343
$\mathrm{cum}(\cdot)$, 439
cumulant, 16, 48, 62–71, 424
criteria, 68
fourth order, 62–71, 125
function, *see* $\mathrm{cum}(\cdot)$, 21, 297
higher order, 34, 48, 61
definition of, 425, 439
joint, *see* multivariate, 115
lag-zero fourth order, 112
matrix, 65, 67, 69, 114
multivariate, 439–445
definition of, 439
expressing in moments, 439–442
properties of, 443–444
nice properties, 443
nice properties of, 58
of complex signal, 125
tensor, 64, 115
univariate, 424–427
definition of, 425
expressing in moments, 425–427
properties of, 425
Cumulant Generating Function
joint, *see* multivariate
multivariate, 439
univariate, 425
cumulants
Generalized Eigenvalue Decomposition, 69
cumulative density function
joint, *see* multivariate
multivariate, 429–431
univariate, 419–420
cylindrical array, 15

D-variate, 227, 300
data reduction, 45
de-correlation
nonlinear, 59–61
decomposition, 91
into subspaces, 83, 112
tetradic, 63
deflation, 48, 62
diagonalization, 67–68
joint, 71
diagonalizer, 67–68
joint, 67, 75, 77
joint non-orthogonal, 71
joint orthogonal, 67, 75
non-orthogonal, 70, 74, 76
orthogonal, 66
differential entropy, **49**, 49
measure of non-Gaussianity, 49
differential equation
solver, 462
dimension, 35
Direction Of Arrival, 2, 11, 83
estimation, 369
observation model, 84–89
discriminating over a set, 42
diversity, 2, 27
spatial, 2, 4, 27
statistical, 2, 27
temporal, 2
d_{n}, 87
DOA, 85, 369, 371
observation model, 84–89
DOA estimation
purpose, 86
dyad, **63**

Edgeworth expansion of pdf, 52, 53
EFOBI, 68
eigen-decomposition, 83
eigen-set, 68
eigen-space, 20, 89–96
eigenfilter, 102
eigenmatrices

diagonalization of, 67–68
eigenmatrix, 65
eigenproblem, 73
generalized, 290
eigenspace, 361
eigenstructure, 73, 89–96, 118–119
eigenvalue
generalized, 361
noise, 89–96, 118–119
sensitivity of, 99
sensor, 119
signal, 89–96, 118–119
tuple, 292, 362
zero, 118
Eigenvalue Decomposition, 44, 65, 73, 91
Generalized, 69
of tensor, 66
eigenvalue tuple, 293, 362
eigenvalues and eigenvectors
sensitivity of, 99
eigenvector
generalized, 362
noise, 89–96, 118–119
sensitivity of, 99
signal, 89–96, 118–119
eigenvectors
generalized, 70, 73
elevation, 11, 13
entropy, 20, 48, 49
approximating, **51–52**
differential, 49
individual, 53
joint, 50
marginal, 50
equivalence
relation, 190, 269, 342
equivalence class
multiset, 311
of index tuples, 306
ergodic, 18, 112
ergodicity, 62, 96, 134
essentially different
sensor correlation functions, 142
estimation of
multivariate cumulants, 445
multivariate expectations, 445
multivariate moments, 445
univariate cumulants, 429
univariate expectations, 429
univariate moments, 429
EVD, 44, 65, 73, 91, 92, 98, 99
of sensor cumulant matrix, 66
tensor, 66
expansion
of pdf, 52
expectation
multivariate, 434–435
definition of, 434
estimation of, *see* estimation
properties of, 434–435
nonlinear, 57–59
univariate, 421–422
definition of, 421
estimation of, *see* estimation
properties of, 421–422
expectation operator, 43
neglecting, 43
exponential power family of pdf's, 36
Extended Fourth Order Blind Identification, 68

far field, 10, 84, 370
feedforward, 60
First Characteristic Function, 420
Taylor expansion of, 422
First Joint Characteristic Function, 433
Taylor expansion of, 436
flatness, 344, 423
FOBI, 68
FOSS, **65**
Fourth Order Blind Identification, 68
fourth order cumulants
concepts related to, 64–66
Fourth Order Signal Subspace, **65**
frame operator, 96, 97, 122
Friedlander, 112, 113, 124
Frobenius norm, 78
function-valued vector
functional SVD of, 261, 334
reduced functional SVD of, 263, 335
functional notation, 25

Gamma, 343
gamma trick, 460

generalized, 464
Gaussian, 6, 16, 36, 423
Gaussian variable
standardized, 52
general subspace method, 128
rationale, 128
generalized array manifold, 114, 127
with conjugation pair, 117
generalized array response matrix, 127
generalized array response vector, 124, 127
Generalized Eigenvalue Decomposition
cumulants, 69
generalized eigenvalue problem, 213
multi-matrix, 22, 131, 164, 215, 261, 294, 333
generalized eigenvalues, 70
generalized eigenvectors, 22, 70, 73
genetic algorithms, 96
geometry of DOA model, 88–89
geometry of observation model, 88–89
GEVD, 69, 210
Givens rotation, 79
Givens rotations, 67
global optimization, 96
gradient
complex, 80
gradient ascent algorithm, 62
gradient ascent method, 42
gradient descent, 80
gradient descent method, 42
Gram-Chalier, 53
Gram-Chalier expansion of pdf, 53
Gram-Charlier expansion of pdf, 52
grid search
exhaustive, 98
guiding principle, 33
guiding principle of statistical independence, 29

Hérault, 2
Hadamard product, 416
p-th order, 416
with conjugation tuple, 416
harmonics, 97
Hérault, 59
Hérault-Jutten algorithm, 59–60
Hermitian, 91, 118
higher order, 48
higher order statistics, 16
Higher Order Temporal Structure, 17
holor, 64, 115, 163
definition of, **63**
homogeneity, 247, 320
property, 190, 269, 342
homogeneity order, 354
homogeneous, 17, 22, 131, 146, 213, 215, 247, 268, 269, 294, 295, 320, 341
function, 146, 247, 320
property of, 146, 247, 320
homogeneous polyconjugal equations
system of, 215, 294
homogeneous polynomial equations
derivation of system of, 138–172
system of, 131, 213
homotopy, 18, 191, 199, 285, 459
method, 199
HOS, 83, 124
disadvantage of, 18, 126
HOTS, 17
hydrophone array, 15
hyperbolic paraboloid, 172, 176, 181, 184
hypersurface
quadric, 272, 279
Hyvärinen, 2

i.i.d., xviii, 6, 32, 135, 223, 379
$\mathcal{I}^p_{\mathrm{c},G}$, 55–56, 63, 112
definition of, 408
$\mathcal{I}^p_{\mathrm{d},G}$, 57–58
definition of, 408
$\mathcal{I}^p_{\mathrm{e},G}$
definition of, 408
$\mathcal{I}^p_{\mathrm{o},G}$, 58
definition of, 408
$\mathcal{I}^p_{\mathrm{a},G}$
definition of, 408
$\mathcal{I}^p_{\mathrm{t},G}$, 58, 63, 115
definition of, 408
IBSS, 1, 5, 27
underdetermined, 34
ICA, 37, 38, 43
images, 9
increment-and-fix, 462

Independent Component Analysis, 29, 37, 82
independent identically distributed, 6, 178, 274
indeterminacies, 4, 40–41, 133, 217
  IBSS, 5, 40–41
  MIBI, 4, 40–41, 133, 217
indeterminacy
  norm, 40
  order, 40
  scaling, 146, 189, 192, 247
information-theoretic measure
  algorithm based on, 53–54
information-theoretic measures, 48
initial value problem, 461
injective, 86, 88, 89
inner product, 95, 97, 101
  between matrices, 65
inner product space, 447
instantaneous transfer coefficient
  complex, 85, 216
  real, 132
ISBSL, 1, 27, 369
isomorphism, 449
  symbol for, xxi, 93, 449, 450
isotropic, 84, 370

Jacobi
  method, 79
  rotation, 79
Jacobi technique, 67
Jacobian, 200
JAD, 75–81
  based on Frobenius norm, 78
  based on model fitting formulation, 78
  based on positive definite formulation, 78
  criteria, 79
JADE, 67–68
JAND, 71
JAOD, 75
joint
  cumulant, 115
joint (orthogonal) diagonalizer, *see* diagonalizer
Joint Approximate Diagonalization, 75
joint approximate diagonalization, 76–81
Joint Approximate Diagonalization of Eigenmatrices, 67–68
Joint Approximate Non-Orthogonal Diagonalization, 71
Joint Approximate Orthogonal Diagonalization, 75
joint central moment, 438
joint cumulant, 439
joint cumulant function, 439
joint diagonalization, 71
  algorithm, 71
  approximate, 75
  non-orthogonal joint, 74
joint diagonalizer, 77
joint expectation
  nonlinear, 57–59
joint moment, 57
joint moment function, 438
joint non-orthogonal diagonalizer, *see* diagonalizer
joint orthogonal diagonalization
  approximate, 67
joint temporal statistics, 39
jpdf, 35, 37, 125
  a posteriori, 35
Jutten, 2, 59

kernel, 449
Khatri-Rao product, 113, 116, 153, 154, 156, 254–256, 326, 327, 417
  p-th order, 418
  higher order, 316
  second order, 243
  with conjugation pair, 116
  with conjugation tuple, 316, 418
Kopp, 83, 97
Kronecker product, 416
  p-th order, 417
  with conjugation tuple, 417
Kullback-Leibner divergence, **51**
kurtosis, 35, 61
  absolute, 125
  as measure of non-Gaussianity, 61–62
  negative, 125
  normalized, 61, 423

Lagrange multipliers, 107
Laplacian, 35
learning rule
based on information-theoretic measure, 54
Cichocki-Unbehauen, 60
Hérault-Jutten, 59
left null space, 157, 184, 257, 268, 329, 341, 449
left singular vector, 451, 455, 456
leptokurtic, 35, 344, 423
linear algebra, 17, 143, 244, 317
linear dependence
of sensor correlation functions, 144
linear space, 143, 244, 317
linear span, 93, 119, 158, 258, 331
linearly independence
of auto-correlation functions, 136, 223, 376
of auto-cumulant functions, 384

magnitude, 133, 217
main assumptions, 23
Main Subspace Assumption, 127
manifold
array, 86
mapping
bijective, 148, 151, 251, 324
matrix-to-matrix, 64
mathematical problems
complementary, 131, 213, 215, 294
matrix
unitary, 74
matrix pencil, 70, 73, 292, 361
generalized, 292, 361, 362
matrix-to-matrix
mapping, 64
Matsuaoka, 78
maximum likelihood, 68
maximum-likelihood, 35
mean, 422
mean-squared value, 422
measure
uncorrelatedness, 72
measures
information-theoretic, 48, **49–54**
of non-Gaussianity, 48, 61–62
of statistical independence, 48
signal processing, **55–62**
signal processing oriented, 49
mesokurtic, 423
method
blind identification, 30
blind separation, 30
methods
exploiting temporal structure, 29
not exploiting temporal structure, 29
MIBI, 1, 3, 27, 131, 213, 215, 294
2×2 mixing case, 131–213
as stepping stone, 4
assumptions, 220, 299
$D \times S$ mixing case, 215
indeterminacies, 133, 217
parameterized, 14, 369
MIBI model assumptions
2×2 mixing case, 134–138
$D \times S$ mixing case, 218–226
MIBI model structure
2×2 mixing case, 132–133
$D \times S$ mixing case, 216–217
MIMO, 1, 3
MIN-NORM, 95
Min-Norm
design problem, 107
pseudo-spectrum, 106–108
Min-Norm method, 106–108
minimal simplex, 35
Minimum-Norm method, 106–108
mixing matrix, 4
mjpdf, 39
MMGEVD, 290, 356
model
assumptions, 31
indeterminacies, **40**
model fitting, 78
advantages of, 79
Molgedey, 73
$\mathrm{mom}(\cdot)$, 438
moment
function, *see* $\mathrm{mom}(\cdot)$, 218
higher order
definition of, 422, 436
joint, *see* multivariate, 57
multivariate, 435–438

compact notation of, 436
conversion to cumulants, 439–442
definition of, 435
derivation from FJCF, 436
properties of, 438
tuple-order of, 435
nonlinear, 57
univariate, 422–423
conversion to cumulants, 425–427
definition of, 422
derivation from FCF, 422
Moment Generating Function
joint, *see* multivariate
multivariate, 437
univariate, 422
mpdf, 37
multi-matrix generalized eigenvalue problem, 131, 164, 215, 261, 294, 333
multichoose, 309, 410
multilinear
transformation, 64
multilinearity, 295
multinomial coefficient, 412
multiset
equivalence class, 311
multiset coefficient, 410
multivariate, 295
MUSIC, 24, 95, 97
SPECTRAL, 97
statistical performance analyses, 101
mutual information, 20, **50–51**
minimizing, 50
relation with negentropy, **52–53**
mutual statistical independence, 37
mutually statistically independent, 16

narrowband, 10–16, 84, 85, 370
natural gradient, 42, 54
near field, 13
negentropy, **50**, 49–50, 52–53
invariance, 52
maximizing, 50
relation with mutual information, **52–53**
neural feedback system, 59
noise
eigenvalue, 89–96, 118–119
eigenvector, 89–96, 118–119
Region Of Support, 134, 137, 219, 225, 226, 298
subspace, 89–96, 118–119
noise projection matrix, 93, 99
noise roots, 103, 106, 396
noise subspace, 21, 83, 150, 166, 168, 250, 251, 262, 265, 335, 337
noise subspace vector, 171, 269, 340
noise subspace vectors
appropriate set of, 95
noise-free, 140
Region Of Support, 135, 137, 219, 226, 298
noise-source
Region Of Support, 135, 137, 219, 226
non-blind, 1
non-Euclidian, 54
non-Gaussian, 37, 112
non-Gaussianity, 16, 29, 34, 38, 47–49, 61, 423
information-theoretic measures of, 49–53
marginal, 16
maximizing, 48, 61, 62
measuring by kurtosis, 61
relation with statistical independence, **52–53**
signal processing measures of, 55–62
non-orthogonal
diagonalizer, 70
non-stationarity, 2, 27, 223, 280
second order, 16, 215, 218, 294
non-stationary, 8, 216
non-white, 8
non-whiteness, 2, 7, 27, 223, 280
second order, 131, 213, 215, 294
nonlinear
moment, 57
nonlinear algebra, 17
nonlinear correlation, 57–59
nonlinear de-correlation, 59–61
nonlinear equations
system of, 95, 121
nonlinear joint expectation, 57–59

norm indeterminacy, 40
normalized kurtosis, 61
notation
    and generalization, 131
    and uniform description of theory, 131
    row vector, 147, 248, 321
nuisance parameter, 73
null space, 449
    left, 449
    right, 449
null-spectrum, 96, 122, 372
nullity, 449
    left, 449
    right, 449

objective function, 30, 41–43, 62
    based on cumulant criteria, 68
    instantaneous, 43
    JADE, 67
ODE, *see* ordinary differential equation, 200, 461
omnidirectional, 10, 12, 84, 370
one-to-one, 86
one-unit algorithm, 48
operator
    number of rows, 157
optimization algorithm, 30
optimization methods, 41–43
order, 5, 133, 217
    of columns, 133, 217
    of source signals, 217
    of sources, 133
    total, 435
order indeterminacy, 40
ordinary differential equation, 200, 461
orthogonal, 45, 447
    complement, 93
    projection matrix, 93
    projection operator, 96, 97
    subspace, 447
orthogonal diagonalizer, 66
orthonormal, 91, 168
overcomplete, *see* blind problems

parabolic valley, 184
parameter estimation, 83
parameter space, 85, 370
parameterization, 11
partition, 425
path following, 199, 200
PCA, 44
    relation to ICA, 45
pdf, 35
    prior, 35, 36
peakedness, 344, 423
peaks
    in pseudo-spectrum, 97, 123
penalty term, 81
permutation, 133
permutation function, 304
perpendicular, 447
    subspace, 447
Pham, 78
phase factor, 85
PHD, *see* Pisarenko Harmonic Decomposition
Pisarenko, 97, 105
Pisarenko Harmonic Decomposition, 97
planar, 84
plane wave
    complex, 85
platykurtic, 125, 423
polyconjugal, 215, 227, 244, 247, 268, 294, 295, 300, 317, 320, 341
polynomial, 139, 295
    LA-TIME-ROOT-MUSIC, 396
    Root-Min-Norm, 106
    ROOT-MUSIC, 103, 104, 109
Porat, 112, 113, 124
position
    sensor, 84–89
position vector, 84, 85, 371
    normalized, 85, 371
positive definite, 91, 118
predictor, 200, 201, 460
    Euler, 204
preprocessing, 29
Principal Component Analysis, 44–45
principle
    identification, 47, 71
    separation, 47, 71
probability density function
    conditional, 431–432
    joint, *see* multivariate, 37, 38, 40

marginal, 38, 39, 51, 430
marginal joint, 37, 431
multivariate, 429–431
posterior, 19
univariate, 419–420
problem statement
IBSS, 5
ICA, 38
ISBSL, 12
MIBI, 3
whitening, 44
projection matrix
noise, 93, 99
orthogonal, 93
signal, 93
projection onto constraint set, 80
projection operator
orthogonal, 96, 97
propagation
medium, 84, 370
time difference, 85
velocity, 85
pseudo-inverse, 36
pseudo-spectrum, 97, 123
Min-Norm, 106–108
SPECTRAL-MUSIC, 97–101
statistically consistent, 101

quadratic, 272
quadratic form, 270
quadric surface, 172
quasi-stationary, 72, 216

radio astronomy, 15
radio telescope array, 15
random variable
real-valued, 419–429
random vector
real-valued, 429–445
range, 13, 448
range space, 448
rank, 447, 449
ratio, 192
reduced row echelon form, 275
Region Of Support, 134, 218, 219, 298
Noise, 134, 137, 219, 225, 226, 298
Noise-Free, 135, 137, 219, 226, 298
Noise-Source, 135, 137, 219, 226
Source, 134, 136, 219, 226, 298
Source-Noise, 134, 137, 219, 226, 298
regression coefficient, 6
resolution, 99
loss of, 101
resolution of identity, 93
Riemannian, 42, 54
right singular function, 455, 456
right singular vector, 451
roller-coaster algorithm, 98
root-finding algorithm, 30
Root-Min-Norm, 106, 107
ROOT-Min-Norm polynomial, 106
ROOT-MUSIC, 95, 102–105
example, 104
ROOT-MUSIC polynomial, 103, 104, 109
ROS, 134, 138, 139, 219, 298
row space, 448
conjugate, 449
row vector
notation, 147, 248, 321

sampling
spatial, 10
temporal, 10
scaling, 5, 133
inverse, 133
scaling indeterminacy, 32, 33, 40, 45, 146, 247, 269, 320, 342
Schmidt, 83, 97
Schuster, 73
score function, 53
Second Characteristic Function, 420
Taylor expansion of, 424
Second Joint Characteristic Function, 433
Taylor expansion of, 439
second order statistics, 16
Second Order Temporal Structure, 17
Sejnowski, 2
self-reciprocal, 103, 396
semi-blind, 1, 14
sensor
correlation structure, 89
position, 84–89
spacing, 88
sensor arrays

active, 1
passive, 1
sensor correlation functions
linear dependence of, 144
structure of, 139–140
sensor correlation matrix
lag-zero, 21
structure, 72, 76, 90, 91
sensor correlation row vectors
linear dependence of, 149
sensor cumulant matrix, 114
sensor cumulant row vectors
linear dependence of, 322
sensor cumulants
algebraic structure of, 63–64
arranging in a matrix, 115
fourth order, 63–71
structure of, 114
special structure of, 63–64
sensor noise
annihilation, 139, 227, 300
sensor spacing
normalized, 87
sensor statistics
structure, 139, 227, 300
sensor subspace, 143, 244, 317
Shannon's sampling theorem
spatial equivalent of, 88, 103
shortest-path decomposition, 35
signal
eigenvalue, 89–96, 118–119
eigenvector, 89–96, 118–119
subspace, 89–96, 118–119
signal processing oriented measures, 49
signal projection matrix, 93
signal subspace, 20, 83, 143, 149, 150, 155, 166, 168, 244, 249–251, 255, 262, 265, 317, 322, 328, 335, 337
dimension of, 155, 255, 328
simulated annealing, 96
sine function, 88
periodicity, 88
singular function
right, 455, 456
singular value, 451
singular vector
left, 451, 455, 456
right, 451
SKAI, 15
skewness, 344, 423
smoothing, 105
snapshot, 85
SOS, 83
SOTS, 17, 71–77, 131, 134, 213, 215, 218, 296
joint source-noise, 134
noise, 134
source, 134
source
Region Of Support, 134, 136, 219, 226, 298
Source Localization, 1, 10–16, 27, 83
purpose, 86
source recovery, 34
source roots, 103, 104, 106, 396
source subspace, 143, 156, 244, 256, 317, 328
dimension of, 156, 250, 256, 322, 328
source-noise
Region Of Support, 134, 137, 219, 226, 298
space
curved, 42
of complex-valued functions, 220, 299
of real-valued functions, 137
sparsity, 18, 34
spatial covariance matrix
noise, 113
spatial frequency
normalized, 88
spatial non-stationarity, 10
spatial non-whiteness, 10
spatial sampling distance, 88
Spectral-Min-Norm, 106, 107
algorithm, 108
Spectral-Min-Norm pseudo-spectrum, 106
SPECTRAL-MUSIC, 97–101
example, 98
pseudo-spectrum, 97–101
spherical ellipse, 272, 275, 279, 282
sphering, *see* whitening, 43
spurious noise, 104
spurious peaks, 99, 101, 103, 396

spurious roots, 103, 104, 106, 396
square-root of matrix, 44
standard deviation, 423
start system, 199, 459
stationary, 112
statistical dependence
  minimizing, 48
statistical independence, 4, 29, 432–433
  at order 2, 56
  at order l, 55, 56, 445
  expressed in joint cumulants, 55, 444
  expressed in pdf's, 432
  guiding principle, 82
  information-theoretic measures of, 49–53
  maximizing, 48
  measuring by (nonlinear) expectations, 59
  measuring by cross-cumulants, 55–56
  measuring by nonlinear expectations, 57
  mutual, 47
  relation with non-Gaussianity, **52–53**
  signal processing measures of, 55–62
  spatial, 47
  up to order l, 55, 445
statistically consistent, 96, 121, 122, 126, 372
statistics
  estimated, 96, 121–122
steepest-descent, 42
step size, 42
Stiefel manifold, 80
stochastic gradient
  ascent, 43, 62
  descent, 43, 54
Sub-Gaussian, 125, 423
submarine, 15
subscript position, 86
subspace, 139, 227, 300
  approach, 90, 130
  fitting, 79
  fourth order, 65
  method, 89–96
    based on HOS, 112–128
    based on SOS, 89–96
  noise, 21, 83, 89–96, 118–119
  orthogonal, 447
  perpendicular, 447
  sensor, 244, 317
  signal, 20, 83, 89–96, 118–119, 143, 149, 155, 244, 249, 255, 317, 322, 328
  source, 143, 156, 244, 256, 317, 328
  zero, 118
subspace approach, 14, 25, 66, 83
subspace fitting, 81
subspace matrix, 127, 131, 144, 155, 213, 215, 245, 251, 264, 318, 323
  definition, 151, 252, 255, 324, 327
  reduced SVD of, 266, 338
  SVD of, 264, 336
  uniquiefied, 252
subspace method
  general, 128
subspace methods, 20, 83
subspace techniques, 14, 131, 215, 294
Super-Gaussian, 35, 344, 423
superposition, 12
superscript position, 86
SVD, 139, 144, 227, 245, 300, 318
  functional, 164
  standard matrix, 164
symbolically different
  sensor correlation functions, 142
system
  of homogeneous polyconjugal equations, 215, 294
  of homogeneous polynomial equations, 131, 213
  of nonlinear equations, 95, 121

target system, 199, 459
template theory, 295
temporal
  cross-structure, 221
temporal structure, 2, 7, 17, 27, 29, 33, 37, 38
  AR(1), 6, 135
  conditions, 134, 219, 298
  second order, 71–77
  simpler, 222
tensor, 19, **62–71**, 115

contraction, 64
cumulant, 64
decomposition, 19, 63
definition of, 64
Eigenvalue Decomposition, 66
EVD, 66
four-dimensional, 64
metric, 43
sensor cumulant, 65
symmetric, 64, 65
vectorization, 66
tetrad, **63**, 115
tetradic, 63, 64
D-tetradic decomposition, 63
tetradic form, 349
tetrads
with conjugation tuple, 115
TIME-MUSIC, 369, 370, 373
Tong, 74
total transfer, 41
towed array, 15
trace, 65
transformation
multilinear, 64
tuple-order, 435

ULA, 12, 13, 24, 87–88, 106
ULA-TIME-ROOT-MUSIC, 395
ULA-TIME-ROOT-MUSIC polynomial, 396
unambiguous, 86, 89
array, 86, 88
array response matrix, 89
Unbehauen, 60
uncorrelated, 74
uncorrelatedness, 71–77
spatial, 16, 39
temporal, 16, 39
underdetermined, *see* blind problems
system of equations, 72
underdetermined blind problems, **34**
Uniform Linear Array, 12, 86–88
unifying, 17
unique, 234
sensor correlation functions, 142, 143, 245
sensor correlation row vectors, 250
sensor cumulant functions, 318
sensor cumulant vectors, 322
uniquiefication, 142, 234
uniquiefied, 231, 233, 303
unit vector, 85, 371
unitary, 264, 337
matrix, 68, 74, 451
maximizer, 68
UTRM, 395

variance, 423
vector space
affine, 19, 35
of discrete-time functions, 24
complex-valued, 24
real-valued, 24
Very Large Array telescope, 15
VLA, 15

wave
plane, 12
spherical, 12
wave vector, 372
wavefield, 88
waveform preserving solution, 41
wavenumber, 372
whitening, 30, 43–47
complexity reduction, 45
matrix, 44
non-uniqueness of, 45
transform, 44

Yeredor, 81

zero
eigenvalue, 118
subspace, 118
zero subspace vector, 171, 269, 340
zero-mean, 132, 216
z-transform, 102

Acknowledgements

The research work presented in this dissertation has been conducted in the Signal Processing Systems (SPS) group at the Technische Universiteit Eindhoven (TU/e) in the Netherlands. After a delay, the last part of the thesis has been finished at the Philips Research Laboratories and in my own time. First of all I would like to thank my co-promoter dr. ir. Piet Sommen for his unshakeable belief in me and for his support in difficult times; I will never forget. I would also like to thank my first promoter prof. dr. ir. Jan Bergmans, as well as Piet Sommen, for giving me the opportunity to do a Ph. D. research project in the interesting area of Blind Identification and Separation, and for providing me with the working environment and facilities. I am very grateful to the other members of by Ph. D. committee, viz. prof. dr. ir. M. Moonen from the Katholieke Universiteit Leuven (KUL), prof. dr. ir. A.-J. van der Veen and prof. dr. ir. P. Urbach from the Technische Universiteit Delft (TUD), prof. dr. A. Tijhuis from the TU/e, and ing. K. Janse from the Philips Research Laboratories, for participating in the committee and wrestling themselves through my thick thesis. Their comments and suggestions have improved the quality of this work. Special thanks to Marc Moonen for being my second promotor and for meticulously proofreading my thesis. I also thank my former roommates, Emanuël, Jeroen, Alfonso, Yulia, and Nicolae for the 'social life' in our room. Special thanks to Emanuël, who has become one of my best friends and always made time for proofreading my articles, discussions, and walks around the campus in which we continuously had to try to walk slower in a serious attempt to relax. Thanks also to the other people from the SPS group for providing me with the working environment and facilities. During the last years time was precious because the writing process consumed much of my time. I want to thank the people who had to 'suffer' from this for their support and understanding. I would also like to thank my girl friend Ingrid for her love and patience during the final part of the Ph. D. road. Last, but certainly not least, I would like to express my gratitude and appreciation to my other friends Hanneke & Wouter, Heidi & Gerard, Johan, Judith & Remko, Matthijs, Michelle, Prarthana, Sebastiaan, and Wim & Arina for their support, love and encouragement.

Curriculum Vitae

Jakob van de Laar was born in 's-Hertogenbosch, The Netherlands, on June 12, 1977. He spent his childhood in Nederhemert, a small village between the dikes of the rivers the Maas and the Waal. He received the M.Sc. degree in Electrical Engineering in 2000 from the Technische Universiteit Eindhoven (TU/e), The Netherlands, where he graduated cum laude on his work for the TNO Physics and Electronics Laboratory concerning data acquisition and signal processing for SHIRA (Ship's Radar), a digital X-band radar system used for monitoring and detecting sea surface features. From 2000 till 2004 he worked as a Ph.D. student in the Signal Processing Systems (SPS) group of the TU/e, where he carried out the work reported in this dissertation. His main research activities concern various Digital Statistical Signal Processing areas, in particular Adaptive Filters, Array Signal Processing, and Blind Identification and Source Separation. Since the beginning of 2005 he is working as a Research Scientist in the Digital Signal Processing group of the Philips Research Laboratories in Eindhoven, The Netherlands.