Research Reports from the Communications Research Laboratory
at Ilmenau University of Technology

Florian Roemer

Advanced Algebraic Concepts for Efficient Multi-Channel Signal Processing

ILMENAU UNIVERSITY OF TECHNOLOGY

Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität Ilmenau

Advanced Algebraic Concepts for Efficient Multi-Channel Signal Processing

January 11, 2013

Florian Roemer
geboren am 06.04.1981 in Erfurt

Dissertation zur Erlangung des
akademischen Grades Doktor-Ingenieur (Dr.-Ing)

Anfertigung im:	Fachgebiet Nachrichtentechnik Institut für Informationstechnik Fakultät für Elektrotechnik und Informationstechnik
Gutachter:	Univ.-Prof. Dr.-Ing. Martin Haardt
	Univ.-Prof. Dr.-Ing. Eduard Jorswieck
	Dr. Rodrigo C. de Lamare
Vorgelegt am:	12. 07. 2012
Verteidigt am:	04. 10. 2012

urn:nbn:de:gbv:ilm1-2012000421

Acknowledgements

This thesis would not have been possible without the gracious support of many people whom I would like to devote this section to.

Firstly, I would like to thank my supervisor Prof. Martin Haardt for discovering my potential, constantly encouraging me to go further, and providing very valuable lessons on many different levels. In the years we have worked together I have enjoyed a free and open-minded working environment with many sources of input that allowed me to be productive and develop myself further.

I would also like to thank Prof. Eduard Jorswieck and Dr. Rodrigo de Lamare for volunteering to review the thesis. The comments and the mutual discussions were very valuable for the thesis itself and for my work beyond it. Words of encouragement in challenging times provided very valuable motivation.

During the last years, I was lucky enough to be able to work together with a long list of individuals that are all very bright minds and wonderful personalities. I apologize to all I cannot name in person, rest assured that I appreciate every talk we had and the impact it had on my personal and professional views. Special thanks goes to Giovanni Del Galdo for teaching me tensors and the art of visualizing concepts, among many other things. I would also like to thank my former and current colleagues Joao Paulo C. L. Da Costa, Martin Weis, Dominik Schulz, Mike Wolf, Jens Steinwandt, Jianshu Zhang, Martin Fuchs, Bilal Zafar, Marko Hennhöfer, Yao Cheng, Jianhui Li for countless technical discussions and personal talks, for sharing their knowledge and skills, for being open and honest, for helping me out when help was dearly needed, for being good friends, and for providing an overall wonderful working environment. I sincerely hope we can all stay in touch and I wish you all the best for your future.

Yet, none of all this would have been possible without the tremendous support of my family. I would like to thank my parents for providing me safe and sorrow-free childhood, for discovering and supporting my abilities, and for encouraging me to be curious, open-minded, and inquiring. Thank you for supporting my education and for being patient in times where work made me hard to get by. Finally, I would like to express my appreciation for my beloved girlfriend Nadine. Words can hardly describe your impact on my life. Every moment of love and care was a much-needed source to find new energy, every day spent together a new inspiration. Thank you for making my life much more interesting, colorful, and rich in variety.

Abstract

Modern society is undergoing a fundamental change in the way we interact with technology. More and more devices are becoming "smart" by gaining advanced computation capabilities and communication interfaces, from household appliances over transportation systems to large-scale networks like the power grid. Recording, processing, and exchanging digital information is thus becoming increasingly important. As a growing share of devices is nowadays mobile and hence battery-powered, a particular interest in *efficient* digital signal processing techniques emerges.

This thesis contributes to this goal by demonstrating methods for finding efficient algebraic solutions to various applications of multi-channel digital signal processing. These may not always result in the best possible system performance. However, they often come close while being significantly simpler to describe and to implement. The simpler description facilitates a thorough analysis of their performance which is crucial to design robust and reliable systems. The fact that they rely on standard algebraic methods only allows their rapid implementation and test under real-world conditions.

We demonstrate this concept in three different application areas. First, we present a semi-algebraic framework to compute the Canonical Polyadic (CP) decompositions of multidimensional signals, a very fundamental tool in multilinear algebra with applications ranging from chemistry over communications to image compression. Compared to state-of-the art iterative solutions, our framework offers a flexible control of the complexity-accuracy trade-off and is less sensitive to badly conditioned data. The second application area is multidimensional subspace-based high-resolution parameter estimation with applications in RADAR, wave propagation modeling, or biomedical imaging. We demonstrate that multidimensional signals can be represented by tensors, providing a convenient description and allowing to exploit the multidimensional structure in a better way than using matrices only. Based on this idea, we introduce the tensor-based subspace estimate which can be applied to enhance existing matrix-based parameter estimation schemes significantly. We demonstrate the enhancements by choosing the family of ESPRIT-type algorithms as an example and introducing enhanced versions that exploit the multidimensional structure (Tensor-ESPRIT), non-circular source amplitudes (NC ESPRIT), and both jointly (NC Tensor-ESPRIT). To objectively judge the resulting estimation accuracy, we derive a framework for the analytical performance assessment of arbitrary ESPRIT-type algorithms by virtue of an asymptotical first order perturbation expansion. Our results are more general than existing analytical results since we do not need any assumptions about the distribution of the desired signal and the noise and we do not require the number of samples to be large. At the end, we obtain simplified expressions for the mean square estimation error that provide insights into efficiency of the methods under various conditions. The third application area is bidirectional relay-assisted communications. Due to its particularly low complexity and its efficient use of the radio resources we choose two-way relaying with a MIMO amplify and forward relay. We demonstrate that the required channel knowledge can be obtained by a simple algebraic tensor-based channel estimation scheme. We also discuss the design of the relay amplification matrix in such a setting. Existing approaches are either based on complicated numerical optimization procedures or on ad-hoc solutions that to not perform well in terms of the bit error rate or the sum-rate. Therefore, we propose algebraic solutions that are inspired by these performance metrics and therefore perform well while being easy to compute. For the MIMO case, we introduce the algebraic norm maximizing (ANOMAX) scheme, which achieves a very low bit error rate, and its extension Rank-Restored ANOMAX (RR-ANOMAX) that achieves a sum-rate close to an upper bound. Moreover, for the special case of single antenna terminals we derive the semi-algebraic RAGES scheme which finds the sum-rate optimal relay amplification matrix based on generalized eigenvectors. Numerical simulations evaluate the resulting system performance in terms of bit error rate and system sum rate which demonstrates the effectiveness of the proposed algebraic solutions.

Zusammenfassung

Unsere moderne Gesellschaft ist Zeuge eines fundamentalen Wandels in der Art und Weise wie wir mit Technologie interagieren. Geräte werden zunehmend intelligenter – sie verfügen über mehr und mehr Rechenleistung und häufiger über eigene Kommunikationsschnittstellen. Das beginnt bei einfachen Haushaltsgeräten und reicht über Transportmittel bis zu großen überregionalen Systemen wie etwa dem Stromnetz. Die Erfassung, die Verarbeitung und der Austausch digitaler Informationen gewinnt daher immer mehr an Bedeutung. Die Tatsache, dass ein wachsender Anteil der Geräte heutzutage mobil und deshalb batteriebetrieben ist, begründet den Anspruch, digitale Signalverarbeitungsalgorithmen besonders *effizient* zu gestalten. Dies kommt auch dem Wunsch nach einer Echtzeitverarbeitung der großen anfallenden Datenmengen zugute.

Die vorliegende Arbeit demonstriert Methoden zum Finden effizienter algebraischer Lösungen für eine Vielzahl von Anwendungen mehrkanaliger digitaler Signalverarbeitung. Solche Ansätze liefern nicht immer unbedingt die bestmögliche Lösung, kommen dieser jedoch häufig recht nahe und sind gleichzeitig bedeutend einfacher zu beschreiben und umzusetzen. Die einfache Beschreibungsform ermöglicht eine tiefgehende Analyse ihrer Leistungsfähigkeit, was für den Entwurf eines robusten und zuverlässigen Systems unabdingbar ist. Die Tatsache, dass sie nur gebräuchliche algebraische Hilfsmittel benötigen, erlaubt ihre direkte und zügige Umsetzung und den Test unter realen Bedingungen.

Diese Grundidee wird anhand von drei verschiedenen Anwendungsgebieten demonstriert. Zunächst wird ein semi-algebraisches Framework zur Berechnung der kanonisch polyadischen (CP) Zerlegung mehrdimensionaler Signale vorgestellt. Dabei handelt es sich um ein sehr grundlegendes Werkzeug der multilinearen Algebra mit einem breiten Anwendungsspektrum von Mobilkommunikation über Chemie bis zur Bildverarbeitung. Verglichen mit existierenden iterativen Lösungsverfahren bietet das neue Framework die Möglichkeit, den Rechenaufwand und damit die Güte der erzielten Lösung zu steuern. Es ist außerdem weniger anfällig gegen eine schlechte Konditionierung der Ausgangsdaten. Das zweite Gebiet, das in der Arbeit besprochen wird, ist die unterraumbasierte hochauflösende Parameterschätzung für mehrdimensionale Signale, mit Anwendungsgebieten im RADAR, der Modellierung von Wellenausbreitung, oder bildgebenden Verfahren in der Medizin. Es wird gezeigt, dass sich derartige mehrdimensionale Signale mit Tensoren darstellen lassen. Dies erlaubt eine natürlichere Beschreibung und eine bessere Ausnutzung ihrer Struktur als das mit Matrizen möglich ist. Basierend auf dieser Idee

entwickeln wir eine tensor-basierte Schätzung des Signalraums, welche genutzt werden kann um beliebige existierende Matrix-basierte Verfahren zu verbessern. Dies wird im Anschluss exemplarisch am Beispiel der ESPRIT-artigen Verfahren gezeigt, für die verbesserte Versionen vorgeschlagen werden, die die mehrdimensionale Struktur der Daten (Tensor-ESPRIT), nichzirkuläre Quellsymbole (NC ESPRIT), sowie beides gleichzeitig (NC Tensor-ESPRIT) ausnutzen. Um die endgültige Schätzgenauigkeit objektiv einschätzen zu können wird dann ein Framework für die analytische Beschreibung der Leistungsfähigkeit beliebiger ESPRIT-artiger Algorithmen diskutiert. Verglichen mit existierenden analytischen Ausdrücken ist unser Ansatz allgemeiner, da keine Annahmen über die statistische Verteilung von Nutzsignal und Rauschen benötigt werden und die Anzahl der zur Verfügung stehenden Schnappschüsse beliebig klein sein kann. Dies führt auf vereinfachte Ausdrücke für den mittleren quadratischen Schätzfehler, die Schlussfolgerungen über die Effizienz der Verfahren unter verschiedenen Bedingungen zulassen. Das dritte Anwendungsgebiet ist der bidirektionale Datenaustausch mit Hilfe von Relay-Stationen. Insbesondere liegt hier der Fokus auf Zwei-Wege-Relaying mit Hilfe von Amplify-and-Forward-Relays mit mehreren Antennen, da dieser Ansatz ein besonders gutes Kosten-Nutzen-Verhältnis verspricht. Es wird gezeigt, dass sich die nötige Kanalkenntnis mit einem einfachen algebraischen Tensor-basierten Schätzverfahren gewinnen lässt. Außerdem werden Verfahren zum Finden einer günstigen Relay-Verstärkungs-Strategie diskutiert. Bestehende Ansätze basieren entweder auf komplexen numerischen Optimierungsverfahren oder auf Ad-Hoc-Ansätzen die keine zufriedenstellende Bitfehlerrate oder Summenrate liefern. Deshalb schlagen wir algebraische Ansätze zum Finden der Relayverstärkungsmatrix vor, die von relevanten Systemmetriken inspiriert sind und doch einfach zu berechnen sind. Wir zeigen das algebraische ANOMAX-Verfahren zum Erreichen einer niedrigen Bitfehlerrate und seine Modifikation RR-ANOMAX zum Erreichen einer hohen Summenrate. Für den Spezialfall, in dem die Endgeräte nur eine Antenne verwenden, leiten wir eine semi-algebraische Lösung zum Finden der Summenraten-optimalen Strategie (RAGES) her. Anhand von numerischen Simulationen wird die Leistungsfähigkeit dieser Verfahren bezüglich Bitfehlerrate und erreichbarer Datenrate bewertet und ihre Effektivität gezeigt.

Contents

List of Figures

List of Tables

List of Algorithms

1. Introduction and scope of the thesis

Digital Signal Processing has become an indispensable part of the everyday life of our modern society. The ever-increasing growth of transistor density predicted by Moore's law has created a vast amount of devices with significant computing power. These devices range from large stationary devices like industrial control systems over nomadic devices such as laptop computers to battery powered hand-held devices such as tablets, mobile phones, or even small autonomous wireless sensors.

This development is not likely to stop anytime soon. In fact, we are witnessing a fundamental change in the way we interact with technology. More and more devices are becoming "smart" by gaining access to enhanced computing power and communication abilities, such as an Internet access. These range from simple devices such as home appliances to wide-ranged networks such as the power grid or transportation systems. Each device has the ability to collect, process, and exchange digital information, which implies a strong call for novel digital signal processing techniques.

Whenever such a fundamental transformation takes place, it opens up exciting new possibilities. The prospects are seemingly endless: smart devices could optimize the way we interact with them or even interact autonomously, alleviating the need for human interaction. This can for instance lead to a more efficient use of scarce and depleting resources, most prominently energy, which is one of the most urgent challenges of our modern society. Another benefit can be safety, for example by the development of smart cars that can optimize traffic flow and minimize the risk of fatal accidents.

However, the list of opportunities is no shorter than the list of challenges and risks we should be aware of. The more we start relying on such systems, the more crucial become aspects like reliability, robustness, security, privacy, or efficiency. As increased computation power also implies increased energy consumption, the need to find particularly efficient signal processing techniques becomes evident. This is becoming increasingly important also since many of the nodes operate on batteries.

With new application areas of digital signal processing emerging constantly, it is compelling to find very sophisticated numerical procedures for reaching the optimal performance given some system design target. This may be one of the reasons why the area of numerical convex optimization [BV04] has experienced a prosperous decade in which more and more applications have been identified where cost or utility functions are convex or can be well approximated by convex functions [LVBL98, LY06b, LSS06]. While the existing algorithms to solve such problems are guaranteed to have a polynomial-time complexity, they are still iterative in nature. This is practically problematic for a number of reasons. Firstly, we have no deterministic description of the actual complexity in terms of number of multiplications and additions since the number of iterations is in general data-dependent. In fact, in light of the price paid for complexity, namely energy consumption, the ability to control the performance-complexity trade-off would be desirable. Some applications call for on-line implementations which are still difficult to achieve for convex optimization based solutions. A second drawback is that it is hard to analyze such algorithms analytically in order to objectively argue about their performance due to their iterative nature. Finally, it is often not easy to come up with a generic and robust implementation since the methods may contain a number of adjustable parameters one has to choose by hand.

These observations serve as the main motivation for the general train of thought of this thesis, which is to find simple algebraic solutions for various signal processing related problems. They may not always provide the optimal performance. But their simplicity enables us to analyze their performance in-depth and gain insights into the loss of optimality or the behavior under various conditions, e.g., the robustness against violations of the model assumptions. Low-complexity solutions are also attractive in light of energy consumption, which is crucial considering the fact that more and more devices performing signal processing tasks are in fact battery-powered mobile devices. Another advantage is that such algorithms are straightforwardly implemented and can be put to practice rather rapidly as they only rely on standard algebraic methods which are already available on modern DSP hardware. It is important to emphasize that the prime focus of the thesis is less on delivering the ultimate best possible solution for a given problem but more on the methodical side. We aim to provide methods how such algebraic solutions can be found in general or how existing algebraic solutions can be enhanced by simple algebraic modifications.

We demonstrate this idea in three different application areas. To enhance the overall structure of the thesis, each application area is presented in a separate part of the thesis which can be read independently of each other. To outline the common algebraic tools that are used in the three main parts, they are preceded by one part introducing the fundamental algebraic concepts. The following sections provide a brief motivation for the different parts, outlining the possible applications, the open problems of the existing solutions, and summarizing the major contributions. A more detailed introduction for each part is presented at the beginning of each part separately, i.e., in the Chapters 2, 6, 9, and 14, respectively.

1.1. Part I: Advanced Algebraic Concepts

The first part of the thesis is devoted to the mathematical fundamentals that are used to derive efficient algebraic solutions for the application areas discussed in the subsequent three parts (tensor decompositions, subspace-based multi-dimensional parameter estimation, and two-way relaying) of the thesis. The main goal is to align the existing results which appear scattered over many books and papers in a systematic manner. We aim at presenting a compact collection that can directly be used by engineers without having to untangle the very formal language used in mathematical textbooks [Rud76, Mun00, DF03]. There have been previous attempts to creating such collections, e.g., the "matrix cookbook" [PP08]. However, despite being frequently cited, it is not maintained by the authors anymore. Moreover, like other well-cited compendia [MN95, Bre78, GvL80], it is lacking certain concepts that are becoming increasingly important in the signal processing community, for instance, the treatment of multilinear (tensor) algebra for which a compact "cookbook"-style compendium is still missing. Therefore, a tensor extension of matrix-based compendia is another goal of this part of the thesis.

We provide a motivation and an overview in Chapter 2. The subsequent Chapter 3 is then devoted to matrix-based algebraic concepts. In particular, we focus on linear and quadratic forms and discuss how an expression containing linear or quadratic expression can be reformulated in a "canonical" manner. The main advantage of these "canonical" expressions is that they often allow to apply well-known solutions, e.g., a Least-Squares optimal solution for unknown parameters in a linear form (via the method of Least Squares) or the maximization or minimization of a quadratic form (via eigenvectors). As an example, the Least-Squares optimal factorization of a Kronecker product and a column-wise Kronecker (Khatri-Rao) product is shown and its applications are outlined.

In Chapter 4, we discuss multilinear (tensor) algebraic concepts. For signals that are inherently R-dimensional for $R > 2$ (e.g., measured over spatial dimensions, frequency, and time), R-way arrays provide a representation of the sampled signal in its native multidimensional form. Such a representation enables us to exploit the rich multidimensional structure, often resulting in a benefit (e.g., improved estimation accuracy). As we outline in Chapter 4 the applications where such a benefit can be exploited are very diverse, examples include face recognition [VT02a], image compression [SL01], hyperspectral image denoising [LB08], data mining [STF06], social network analysis [ACKY05], pattern recognition [SE07], array signal processing [SBG00], communications [SGB00] biomedical signal processing [Möc88, MHH+06, dVVdL+07], or numerical mathematics [HKT05]. We have successfully applied tensors for prediction of frequency- and time-selective MIMO channels [MGH08], channel estimation for two-way relaying [RH10c] (see also Part IV of this thesis), subspace-based multidimensional parameter estimation (see also Part III of this thesis) and biomedical signal processing [JRW+08, WRH+09, WJG+10b, WJR+10, WJG+10a, BCA+10, BCA+12].

The necessary algebraic concepts to manipulate R-D signals are summarized in Chapter 4, providing many useful properties that are used in the later parts of the thesis and serving as a reference for engineers that want to apply multilinear algebra to signal processing applications. We extend existing tensor surveys like [KB09, CLdA09] in some aspects, for instance, by discussing consistent permutations of tensor unfoldings which are needed for many of the derivations in the performance analysis of subspace-based multidimensional parameter estimation schemes discussed in Part III of the thesis.

The final Chapter 5 of this part of the thesis provides a summary and links the results to the later parts of the thesis. Proofs and derivations of Part I are found in Appendix B.

1.2. Part II: Semi-Algebraic CP decomposition (SECSI)

In this part of the thesis we focus on the decomposition of a given multidimensional signal into a sum of rank-one components, which is referred to as CANDECOMP / PARAFAC [CC70, Har70], or Canonical Polyadic (CP) decomposition. There exist many applications where the underlying signal of interest can be represented by a trilinear or multilinear CP model. These range from psychometrics [CC70, Har70, KdL80] and chemometrics [AB03] over array signal processing [SBG00] and communications [SGB00] to biomedical signal processing [Möc88, MHH+06, dVVdL+07, WRH+09, BCA+12], image compression [SL01] or numerical mathematics [HKT05] (see also [KB09] and references therein). The success of the CP decomposition is mainly due to its inherent essential uniqueness, which allows to decompose a given signal into components without posing additional constraints artificially (as required for matrix decompositions). A detailed motivation, the data model, and a state-of-the-art review is presented in Chapter 6.

The most prominent approach for finding the CP decomposition relies on an Alternating Least Squares (ALS) procedure [KdL80, BSG99]. The original idea goes back to [CC70, Har70] and many improvements have been proposed since then, e.g., enhanced line search (ELS) [RC05, RCH08] or Tikhonov regularization [NdLK08]. The main shortcoming of ALS-based algorithms is that they may require a large number of iterations to converge and are not guaranteed to reach the global optimum. This motivates the need for alternative approaches that are more reliable and have a lower computational complexity. To overcome these issues, we introduce a semi-algebraic framework for approximate CP decompositions based on a set of Simultaneous Matrix Diagonalization (SMD) problems in Chapter 7. The links between the CP decomposition and Simultaneous Matrix Diagonalizations (SMDs) were already pointed out by [vdVP96, AFCC04] for the symmetric case and by [dL04a, dL06] for the non-symmetric case. Recently, [AFCC04] was also generalized to the non-symmetric case [LA11], which differs from [dL04a] in the way the unfoldings are defined. However, these approaches have in common that only a single SMD is solved and all parameters are estimated from the solution of this SMD. We have extended these approaches in [RH08b, RH09a] by showing that due to the structure of the problem, several SMDs can be constructed, each giving rise to one candidate solution for the CP decomposition. The final estimate can then be selected in a subsequent step. This framework opens up many degrees of freedom to flexibly control the complexity-accuracy trade-off. This allows to tune the CP algorithm to the specific needs of the application at hand. We show via simulations that compared to state of the art CP algorithms we obtain enhanced robustness, reduced sensitivity to correlation in the data, and reduced computational complexity.

A summary and an outlook to future work related to CP decompositions is provided in Chapter 8. Moreover, proofs and derivations of Part II are found in Appendix C.

1.3. Part III: Subspace-Based Parameter Estimation

Part III is devoted to subspace-based high-resolution parameter estimation from multidimensional signals. One example for such signals are multi-dimensional harmonics which are sampled on a multi-dimensional lattice. Here, the goal is to estimate the frequencies of these harmonics in all dimensions, which is referred to as *harmonic retrieval* [KAB83, AR88]. The multi-dimensional harmonic retrieval problem appears in many different application areas. For instance, it is required to fit the parameters of a double-directional MIMO channel model [ZFDW00, SMB01, HTR04] from MIMO channel sounder measurements [ZHM+00]. This is a key step to developing more sophisticated and realistic channel models that can then be used to develop and test transmission concepts for next-generation wireless communication systems. It has been shown that under idealized conditions, the channel transfer function corresponding to this model can be described by a R-D harmonic signal [RHST01] where the frequencies in the separate dimensions are linked to physical paramters of the propagation, such as directions of arrival, directions of departure, time delay, or Doppler shift. A related application is MIMO Radar [JLL09, NS10]. As shown in [NS10], single-pulse and multi-pulse bistatic Radar configurations give rise to a 2-D and a 3-D harmonic retrieval problem, respectively. Another application is 2-D nuclear magnetic resonance (NMR) spectroscopy [BL86] where a molecular system is excited with a 2-D Radio Frequency (RF) pulse sequence and the measured signal can be modeled as a sum of 2-D (damped) harmonics.

Chapter 9 provides a more detailed motivation and a state-of-the-art survey. Moreover, we develop the multidimensional data model using two application examples (direction of arrival estimation and geometry-based parametric channel modeling). We discuss separability and centro-symmetry of 2-D antenna arrays and show that arbitrary beam patterns can be allowed as long as all the elements share the same beam pattern.

A major shortcoming of existing multidimensional parameter estimation schemes is that the rich structure of the multidimensional signals is not fully exploited since the signals are represented by matrices that only possess two dimensions (rows and columns). As we discuss in Part III of the thesis, if multidimensional signals are sampled on a separable R-dimensional grid, this multidimensional structure can be exploited in many ways by virtue of multilinear algebra introduced in Part I of the thesis. We develop a tensor-based data model for multidimensional signals in Chapter 9. Based on this model we show in Chapter 10 that already in the subspace estimation step, applying tensor calculus gives rise to an enhanced tensor-based signal subspace estimate [RHD06, HRD08]. This estimate can be used to improve arbitrary subspace-based R-D subspace-based parameter estimation algorithms. We also show that this subspace can be found by applying a structured projection to the (unstructured) matrix-based subspace estimate which filters out all undesired components that do not obey the required structure. This provides an intuitive understanding for the reason why the accuracy is actually improved. The link was first shown by us for the special case $R = 2$ in [RBHW09], the proof for the general R-D case is presented in Appendix D.3 this thesis.

We then apply this enhanced subspace to the family of R-D Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT)-type algorithms in Chapter 11 and demonstrate that they can conveniently be formulated in terms of tensor notation, which in fact simplifies their description [HRD08]. Moreover, we obtain an improved parameter estimation accuracy due to the enhanced tensor-based subspace estimate. We also show that this tensor formulation allows to enhance the algorithms even further. For instance, the overdetermined shift invariance equations in ESPRIT-type algorithms can be solved with the tensor-based TS-SLS algorithm [RH07b] which outperforms the matrix-based LS solution. Another enhancement to ESPRIT we discuss is the exploitation of a specific structure of the source amplitudes. We show that if the sources transmit real-valued symbols only, this gives rise to a specific symmetry in the data model which can be used to virtually double the number of available sensors at the receiver. The resulting "non-circular (NC)" ESPRIT-type algorithms allow to resolve twice as many sources and yield an improved parameter estimation accuracy [HR04]. We also show how to combine both benefits, the multidimensionality and the fact that the sources' symbols are real-valued [RH09b]. It is a non-trivial combination, since the way the sensors are virtually doubled in the matrix case destroys the separable sampling grid required for the tensor-based algorithms. A mode-wise tensor augmentation is proposed and the resulting NC Tensor-ESPRIT-type algorithms are shown to outperform the previously existing algorithms. The discussion in this thesis extends our the previous publications [HR04, RH07b, HRD08, RH09b] in a number of aspects. First of all, it presents proofs for many of the theorems that were stated in the conference papers but not proven due to space limitations (see Appendices D.5 to D.8). Secondly, we show that NC Unitary ESPRIT and NC Unitary Tensor-ESPRIT can be applied to arrays that are not centro-symmetric in spite of the fact that centro-symmetry is required for Unitary ESPRIT. Thirdly, we prove that the "Standard" versions NC Standard ESPRIT and NC Standard Tensor-ESPRIT are actually not needed since their performance is identical to NC Unitary ESPRIT and NC Unitary Tensor-ESPRIT, respectively, while having a higher computational complexity.

A particularly important aspect of this part is Chapter 12 where we provide an analytical performance assessment of the ESPRIT-type algorithms that have been introduced. Our framework is based on a first-order perturbation expansion of the Singular Value Decomposition which was introduced in [LLV93]. This expansion models the estimation error of the signal subspace as an explicit function of the perturbation of the data (i.e., the additive noise), only assuming that the perturbation is small compared to the desired signal. The main advantage of this approach is that no assumptions about the statistics of the desired signal or the perturbation are required. Moreover, it is asymptotic in the effective SNR, i.e., it becomes exact as either the noise variance vanishes or the number of samples approaches infinity.

This is not the case for many existing analytical performance results for high-resolution parameter estimation algorithms. The most frequently cited papers [KB86] for the MUSIC algorithm and [RH89a] for ESPRIT as well as many follow-up papers which extend the original results (e.g., [PK89b], [Fri90], [MZ94], [ZKM92], [MHZ96]) are based on a result on the distribution of the eigenvectors of a sample covariance matrix first published in [Bri75]. However, the result shown in [Bri75] requires the desired signal as well as the noise to be Gaussian distributed which is not required by our framework based on [LLV93]. Moreover, [Bri75] is asymptotic in the number of samples, whereas [LLV93] can be applied even in the single snapshot case (provided the noise variance is sufficiently small). Another drawback of [Bri75] is that the expressions for the covariance matrix of the eigenvectors are rather long and difficult to simplify, whereas [LLV93] provides intuitive and short expressions.

We begin by extending the perturbation expansion for the SVD provided in [LLV93] to the HOSVD [RBHW09]. Based on this result we then find corresponding expansions for the Tensor-ESPRIT-type algorithms. Since they are explicit in the perturbation, they do not require any assumptions about the statistics of the perturbation. This is advantageous since it allows to incorporate many different preprocessing steps such as Forward-Backward Averaging (see Appendix D.14) the transformation onto the real-valued domain used in Unitary ESPRIT [HN95] (see Appendix D.13), the array augmentation of NC ESPRIT-type algorithms [HR04], spatial smoothing [SWK85], and tensor-based spatial smoothing [THRG10, THG09b, THG09a]. Note that we do not provide a performance analysis for NC ESPRIT or for spatial smoothing in this thesis. This is left for future work (see Chapter 13). Note that we also extend our performance algorithms to ESPRIT-type algorithms based on SLS [RH11]. We additionally provide closed-form expressions for the mean square error (MSE) in the special case that the perturbation comprises only of circularly symmetric white noise. For the case of a single source, we simplify the MSE into compact expressions that depend only on the number of sensors and the effective signal to noise ratio (SNR). As we show, all Least-Squares (LS) based ESPRIT-type algorithms yield the same MSE and the asymptotic efficiency decreases with the number of sensors in the array. On the other hand, we also show that a single iteration of the Structured Least Squares (SLS) algorithm changes the situation since in this case, the asymptotic efficiency is very close to one. The discussion in this thesis extends our previous publications [RBHW09, RBH10, RH11] on this subject in a number of aspects. First of all, we provide proofs for all the theorems which could not be included in the conference versions due to space reasons (see Appendices D.12 to D.18). Moreover, we provide the closed-form expression for the MSE of ESPRIT based on SLS in the special case of a single source.

A summary and an outlook to possible future work related to subspace-based multidimensional parameter estimation is given in Chapter 13. An overview of the various ESPRIT-type algorithms and the Least Squares algorithms to solve the overdetermined shift invariance equations is presented in Tables 13.1 and 13.2, respectively. The tables also outline the cases where a performance analysis exists and where it is still open. Proofs and derivations of Part III are found in Appendix D.

1.4. Part IV: Two-Way Relaying

Finally, the third application area is bidirectional relay-assisted communications. The main motivation for considering relaying is that with advanced digital modulation schemes and MIMO techniques we have almost reached the fundamental limits of point-to-point channels predicted a long time ago [Sha48, Tel99]. Multi-User MIMO techniques [SSH04] are already in the process of being implemented into the current standards. To enhance the system performance further we have to seek innovative ways of accessing the wireless medium. Instead of decoupling the networks into parallel point-to-point links by orthogonal access (TDMA, FDMA, SDMA), recent research has taught us that interference can sometimes be used in a constructive manner [PDF+08, SRH10b, VLV07]. It is therefore foreseen that future networks will be built from more sophisticated building blocks such as the interference channel [HK81], the relay-enhanced channel [vdM71, CE79], or the two-way relaying channel [RW05]. However, the performance limits and the optimal access schemes for these building blocks are still unknown [DHL+11].

We contribute to a better understanding of one of these building blocks by investigating a particular example, namely, two-way relaying [RW05, RW07]. Via two-way relaying we can achieve a bidirectional exchange of information between two network nodes while the nodes as well as the relay station can operate in half-duplex mode. This is a desirable feature with regard to the goal to keep the hardware complexity of relay stations as low as possible. For the same reason, we focus on (digital) amplify and forward relays [PWS+04], which are simpler to implement than relay stations that decode the source nodes' messages.

An introduction to relay-assisted communication with a detailed motivation for using two-way relaying and a state-of-the-art review is provided in Chapter 14. We develop the data model and show that a two-way relaying system with a MIMO amplify and forward relay can be converted into two parallel point to point MIMO links if the terminals possess channel knowledge. To obtain the required channel knowledge, we introduce algebraic channel estimation schemes in Chapter 15 [RH09e, RH09d, RH10c]. As we show, the specific structure of the compound channel matrices gives rise to a quadratic Least Squares problem, which can be exploited via the tensor-based channel estimation scheme TENCE. The channel estimation accuracy can be further improved via a Structured Least Squares (SLS) based iterative refinement step, which requires only one to four iterations, depending on the SNR. As a side result, we find design rules for the pilot sequences that should be used during the training phase in order to facilitate the channel estimation procedure.

Based on the acquired channel knowledge we then discuss the design of the relay amplification matrix for the MIMO amplify and forward relays in two-way relaying in Chapter 16. Existing approaches to this task suffer of two major shortcomings: either they rely on very complicated numerical optimization procedures [ZLCC09, LSPL10] that may be difficult to implement in practice or they are based on ad-hoc proposals [UK08, VH11] that do not perform satisfactory with respect to suitable performance metrics such as the bit error rate or the sum-rate. This provides the motivation for finding simple algebraic solutions that are inspired by system performance metrics and hence perform not much worse than the complicated optimal solutions. For the general MIMO case, we develop the Algebraic Norm Maximizing (ANOMAX) transmit strategy [RH09a], which maximizes the Frobenius norms of the effective MIMO channels that convey the desired signals. We demonstrate that ANOMAX concentrates most of the energy on the dominant eigenmodes of these effective channels, yielding a significant SNR improvement for single-stream transmission and thus a low bit error rate providing a reliable transmission link. However, such a low-rank structure is less suitable to spatially multiplex several data streams as it cannot provide the full spatial multiplexing gain. Therefore, we present a simple algebraic modification called Rank-Restored ANOMAX [RH10a], which restores the required rank for high SNRs. We show that a simple heuristic inspired by the water filling (WF) principle yields a sum-rate very close to the optimum sum-rate. Finally, for the special case of single-antenna terminals, we introduce the sum-rate optimal semi-algebraic Rate-Maximization via Generalized Eigenvectors (RAGES) scheme [RH10b]. We verify the optimality of RAGES by comparing it to the sum-rate optimum found via the Polynomial Time DC (POTDC) algorithm [KVRH12, KRVH12]. POTDC solves the non-convex sum-rate maximization problem by a sequence of semi-definite programming problems obtained by locally linearizing the non-convex parts in the cost function. All proposed schemes are evaluate it in numerical simulations and compared to the state-of-the-art schemes. The discussion in this part of the thesis extends previous conference publications in some aspects. Firstly, the discussion of the ANOMAX scheme is significantly extended (see Section 16.3.3). We prove in Appendix E.7 that it gives a rank-one solution if the weighting coefficient is set to zero or to one and in Appendix E.6 that it coincides with the Dual Channel Matching (DCM) strategy from [VH11] for the special case of single-antenna terminals. We also introduce a low-complexity version of RAGES for the special case of white noise at the relay, see Appendix E.9.

Chapter 17 provides a summary and outlines possible future research directions connected to two-way relaying. Proofs and derivations of Part IV are found in Appendix E.

The final Chapters 18 and 19 collect all the contributions from the thesis again and summarize the future research directions related to all four parts of the thesis. There are five appendices to the thesis. Appendix A summarizes the list of acronyms and the mathematical notation used throughout the thesis. Appendices B, C, D, and E contain details, proofs, and derivations of Part I, II, III, and IV, respectively. Note that the bibliography is split into two parts: one part with the own references and a second part with all other references.

Part I.

Advanced Algebraic Concepts

The first part of the thesis is devoted to the mathematical fundamentals that are used to derive efficient algebraic solutions for the application areas discussed in the subsequent three parts (tensor decompositions, subspace-based multi-dimensional parameter estimation, and two-way relaying) of the thesis. The main goal is to align the existing results which appear scattered over many books and papers in a systematic manner. We aim at presenting a compact collection that can directly be used by engineers without having to untangle the very formal language used in mathematical textbooks [Rud76, Mun00, DF03]. There have been previous attempts to creating such collections, e.g., the "matrix cookboox" [PP08]. However, despite being frequently cited, it is not maintained by the authors anymore. Moreover, it is lacking certain concepts that are becoming increasingly important, for instance, the treatment of multilinear (tensor) algebra for which a compact "cookbook"-style compendium is still missing.

We provide a motivation and an overview in Chapter 2. The subsequent Chapter 3 is then devoted to matrix-based algebraic concepts. In particular, we focus on linear and quadratic forms and discuss how an expression containing linear or quadratic expression can be reformulated in a "canonical" manner. The main advantage of these "canonical" expressions is that they often allow to apply well-known solutions, e.g., a Least-Squares optimal solution for unknown parameters in a linear form (via the method of Least Squares) or the maximization or minimization of a quadratic form (via eigenvectors). As an example, the Least-Squares optimal factorization of a Kronecker product and a column-wise Kronecker (Khatri-Rao) product is shown and its applications are outlined.

In the subsequent Chapter 4, we discuss multilinear (tensor) algebraic concepts. For signals that are inherently R-dimensional for $R > 2$ (e.g., measured over spatial dimensions, frequency, and time), R-way arrays provide a representation of the sampled signal in its native multidimensional form. Such a representation enables us to exploit the rich multidimensional structure, often resulting in a benefit (e.g., improved estimation accuracy). The necessary algebraic concepts to manipulate R-D signals are summarized in Chapter 4, providing many useful properties that are used in the later parts of the thesis.

The final Chapter 5 of this part of the thesis provides a summary and links the results to the later parts of the thesis. Proofs and derivations for this part are found in Appendix B.

2. Motivation and Overview

In this first part of the thesis we review the necessary fundamentals which are used for deriving efficient algebraic solutions to various problems discussed in the subsequent three parts (tensor decompositions, subspace-based multi-dimensional parameter estimation, and two-way relaying) of the thesis. We present a set of generic tools that can be applied to reformulate a given problem into a simpler "canonical" form, for which solutions may already exist.

Such results exist in the literature, however, they are spread over many books and papers. Moreover, they are sometimes not easily accessible to the engineer due to the very formal language used by mathematicians [Rud76, Mun00, DF03]. Hence, the main motivation for this part is to present these concepts in a compact and systematic form, from which they follow naturally. There have been attempts for creating comprehensive compendia before, a prominent example is given by the "matrix cookbook" [PP08]. However, despite being frequently cited[1], the matrix cookbook is no longer maintained. Moreover, it does not contain certain concepts we need in this thesis, for instance the vectorization and the LS-optimal factorization of Kronecker and Khatri-Rao products (Section 3.1.2) or concepts of multilinear (tensor) algebra. More examples for well-cited compendia include [MN95] with special focus on matrix differential calculus, [Bre78, Van00] discussing Kronecker products, or [GvL80] with a focus on matrix decompositions. Another shortcoming of these existing references is that none of them presents results on multi-linear algebra, which has become a very important tool in digital signal processing during the last decade. In light of this, a "tensor extension" of the matrix cookbook would be very desirable. This is an additional goal of this part of the thesis.

We start with linear forms and quadratic forms in Chapter 3 which appear very frequently in various fields of signal processing. The main reason for this is that they often have a direct physical interpretation. For instance, linear forms appear when performing transformations into another basis (such as projections), which are very common in digital filter design [OS99] or array signal processing [Van02]. In this thesis, the reformulation of linear forms is extensively used in Part III for the first-order perturbation expansions of the estimation errors of the signal subspace and subspace-based parameter estimation schemes which are formulated in terms of linear expressions. We also apply it in Part IV for rewriting the transmission equations in a two-way relaying system with amplify and forward relays, e.g., in the derivation of the Algebraic Norm Maximizing (ANOMAX) transmit strategy (cf. Section 16.3). Moreover,

[1] In July 2012, Google Scholar had recorded 289 citations for the matrix cookbook.

quadratic forms occur when we compute powers, such as the transmit power or the interference power. We use these results in Part IV, e.g., to show that the sum-rate in the special case of single-antenna terminals can be expressed as a product of two ratios of quadratic forms (cf. Section 16.1.3). Note that even if the given functions are more complicated, we can often approximate them by their Taylor series which, up to order two, consists of a constant, a linear, and a quadratic term. These are, however, not always in their canonical form which is why it is of crucial importance to have the ability to reformulate such expressions.

As an extension, we also discuss ratios of quadratic forms, which are very strongly linked to eigenvectors and often permit closed-form solutions in terms of the eigenvectors of a specific matrix [Str93]. This link is used in Part IV of the thesis to derive the semi-algebraic RAGES (Rate-Maximization via Generalized Eigenvectors for Single Antenna Terminals) strategy (cf. 16.4).

Another important concept we discuss is the vectorization of Kronecker products and column-wise Kronecker (Khatri-Rao) products [LP93], which follow naturally as special cases of linear forms. We show how this vectorization can be used to find (Least-Squares optimal) approximate factorizations of a given matrix into a Kronecker product [Van00] or a Khatri-Rao product of two matrices. This is the key element to the R-D extension ($R > 3$) of the SECSI framework for efficient CP decomposition discussed in Part II of the thesis. It is also one of the two key elements of the Tensor-Based Channel Estimation (TENCE) algorithm for channel estimation in two-way relaying which is shown in Part IV, Chapter 15.

In the next Chapter 4, we shift our attention to multi-linear forms, for which multi-linear (tensor) calculus is needed. The formulation of these forms in terms of unfoldings and n-mode products [dLdMV00b] provides the necessary basis for manipulating multi-dimensional signals in their native, multi-dimensional form. By explicitly exploiting the multi-linear structure inherent in the data, we can tailor signal processing algorithms more specifically to this structure and hence obtain an improvement, for instance, in suppressing the additive noise more efficiently [KB09]. As in the matrix case, this often requires tools to reformulate a given multi-linear form. Moreover, since for R-dimensional signals, the elements are referenced by a set of R indices, we often need to permute the elements of multi-linear forms to make them "compatible" with each other. To this end we introduce a set of permutation matrices which allows the necessary manipulations and discuss their properties. Tensor calculus is used for the tensor-based algebraic manipulations in the entire Part II and III, and even parts of Part IV (Chapter 15). Finally, the Higher-Order SVD (HOSVD) and the Canonical Polyadic (CP) decomposition introduced in Section 4.2 form the basis for Part II of this thesis where their link is exploited to derive an efficient algorithm to compute the CP decomposition. The HOSVD is also the basis for the tensor-based subspace estimation scheme (cf. Section 10.2) used for Tensor-ESPRIT-type parameter estimation algorithms discussed in Part III. Moreover, the CP decomposition is employed in Part IV to facilitate the design of the relaying strategy during the training phase needed for TENCE (cf. Chapter 15).

Finally, Chapter 5 contains a summary of this first part which provides the links to the subsequent parts of the thesis as well as some bibliographical notes and further reading. Moreover, Appendix B contains proofs and derivations for this part.

It should be pointed out that many of the concepts we discuss in this part of the thesis are already known and have been frequently applied. However, we put emphasis on providing a systematic and compact collection of these concepts under a unified framework, as this provides the reader with a reference where also lesser-known tools (e.g., the Least-Squares optimal factorization of Khatri-Rao and Kronecker products used in Part II and Part IV of this thesis or the consistent permutation of tensor unfoldings used in Part III and Part IV of this thesis) are easy to find.

3. Matrix-based algebraic concepts

3.1. Linear forms and quadratic forms

3.1.1. Linear forms

A linear form $\mathcal{L}$ is defined as a mapping from a vector space $\mathbb{V}$ to its field of scalars $\mathbb{F}$ which is linear, i.e., it satisfies the law of superposition $\mathcal{L}(\boldsymbol{x}_1 + \boldsymbol{x}_2) = \mathcal{L}(\boldsymbol{x}_1) + \mathcal{L}(\boldsymbol{x}_2)\ \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{V}$ and it is homogeneous $\mathcal{L}(\alpha \cdot \boldsymbol{x}) = \alpha \cdot \mathcal{L}(\boldsymbol{x})\ \forall \boldsymbol{x} \in \mathbb{V}, \alpha \in \mathbb{F}$. Throughout this thesis, the vector space we consider is $\mathbb{V} = \mathbb{C}^N$ over the field $\mathbb{F} = \mathbb{C}$. In this case, any linear form in a vector $\boldsymbol{x} \in \mathbb{C}^N$ can be expressed as a linear combination of the elements of $\boldsymbol{x}$ with some coefficients, say, α_n, $n = 1, 2, \ldots, N$, i.e.,

$$y = \sum_{n=1}^{N} \alpha_n \cdot x_n = \boldsymbol{a}^{\mathrm{T}} \cdot \boldsymbol{x}. \tag{3.1}$$

We often encounter vectors or matrices where each element is a linear form. A column vector of linear forms can easily be expressed as

$$\boldsymbol{y} = \boldsymbol{A} \cdot \boldsymbol{x} \tag{3.2}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$ contains the coefficients for the M linear forms in $\boldsymbol{x}$ that are contained in the vector $\boldsymbol{y} \in \mathbb{C}^M$. In other words, the m-th component of $\boldsymbol{y}$, which we denote as y_m is given by $y_m = \boldsymbol{a}_m^{\mathrm{T}} \cdot \boldsymbol{x}$, where $\boldsymbol{a}_m^{\mathrm{T}}$ is the m-th row of $\boldsymbol{A}$. We refer to the vector of linear forms in (3.2) as the "canonical" way of expressing a linear form.

An example where we encounter (3.2) in practice is the separation of linear mixtures. Let us assume that the vector $\boldsymbol{x}$ contains parameters of interest and we perform an experiment where we observe a vector $\boldsymbol{y}$ which follows (3.2). In this case, (3.2) represents a set of M linear equations in N unknowns and hence many efficient algorithms for solving it for $\boldsymbol{x}$ are available. Even in the case where no exact solution in $\boldsymbol{x}$ exists since the observed $\boldsymbol{y}$ is not entirely contained in the column space of $\boldsymbol{A}$ (e.g., due to additive noise or imprecise knowledge of $\boldsymbol{A}$), efficient algorithms for finding an approximate solution in $\boldsymbol{x}$ exist, e.g., the method of Least Squares (LS) or the Total Least Squares (TLS) algorithm [GvL80]. Therefore, we show in the sequel how different kinds of linear forms can be reformulated into this "canonical" linear form.

For instance, we may encounter a matrix of linear forms, i.e., a matrix $\boldsymbol{Y} \in \mathbb{C}^{M\times K}$, where each element is a linear form. In the most general case, the (m,k)-element of $\boldsymbol{Y}$ which we denote as $[\boldsymbol{Y}]_{(m,k)}$ can be expressed as

$$[\boldsymbol{Y}]_{(m,k)} = \sum_{n=1}^{N} a_{m,k,n} \cdot x_n \tag{3.3}$$

where $a_{m,k,n}$ represent the corresponding coefficients. It is possible to express this matrix of linear forms directly via a three-mode product between a tensor of coefficients $\boldsymbol{\mathcal{A}}$ and the vector $\boldsymbol{x}$ (which will be covered in Section 4.1). Alternatively, we may reexpress (3.3) into the canonical form (3.2) by applying the vec-operator. This vectorization stacks all columns of a matrix on top of each other to yield a column vector containing all the elements of the matrix. Formally, it is defined as

$$\operatorname{vec}\left\{\begin{bmatrix}\boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_N\end{bmatrix}\right\} = \begin{bmatrix}\boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N\end{bmatrix} \in \mathbb{C}^{M\cdot N\times 1}, \tag{3.4}$$

where $\boldsymbol{x}_n \in \mathbb{C}^{M\times 1}$ are arbitrary vectors for $n = 1, 2, \dots, N$. Therefore, we can write (3.3) as

$$\operatorname{vec}\{\boldsymbol{Y}\} = \bar{\boldsymbol{A}} \cdot \boldsymbol{x}, \tag{3.5}$$

where $\operatorname{vec}\{\boldsymbol{Y}\} \in \mathbb{C}^{M\cdot K}$ and $\bar{\boldsymbol{A}} \in \mathbb{C}^{M\cdot K\times N}$.

There is an important special case where $\boldsymbol{A}$ is in fact more structured. Consider the following "matrix product" linear form

$$\boldsymbol{Y} = \boldsymbol{B} \cdot \boldsymbol{X} \cdot \boldsymbol{C}, \tag{3.6}$$

where $\boldsymbol{Y} \in \mathbb{C}^{M\times K}$, $\boldsymbol{B} \in \mathbb{C}^{M\times N_1}$, $\boldsymbol{X} \in \mathbb{C}^{N_1\times N_2}$, and $\boldsymbol{C} \in \mathbb{C}^{N_2\times K}$. The matrix $\boldsymbol{Y}$ is also a matrix of linear forms but it is less general than (3.3) as there are only $M\cdot N_1 + K\cdot N_2$ coefficients for $N_1\cdot N_2$ variables (compared to $M\cdot K\cdot N$ coefficients for N variables in (3.3)). To reformulate (3.6) into the "canonical" vector linear form shown in (3.3), we can apply the vec-operator to (3.6). Since the resulting vector is a linear form in $\operatorname{vec}\{\boldsymbol{X}\}$ we know there must exist a matrix $\boldsymbol{A}$ such that $\operatorname{vec}\{\boldsymbol{Y}\} = \boldsymbol{A} \cdot \operatorname{vec}\{\boldsymbol{X}\}$. In fact it is easy to see that $\boldsymbol{A}$ must contain all pair-wise products between the elements of $\boldsymbol{B}$ and $\boldsymbol{C}$. More precisely, we have $\boldsymbol{A} = \boldsymbol{C}^{\mathrm{T}} \otimes \boldsymbol{B}$, where $\otimes$ denotes the Kronecker-product (defined in Appendix A.2). The corresponding rule can be written as

$$\text{vec}\{\boldsymbol{B}\cdot\boldsymbol{X}\cdot\boldsymbol{C}\} = (\boldsymbol{C}^{\mathrm{T}}\otimes\boldsymbol{B})\cdot\text{vec}\{\boldsymbol{X}\} \tag{3.7}$$

and dates back to [Neu69]. It became more popular in the signal processing community via [Bre78] which lists lots of properties and applications of the Kronecker product and gives links to the corresponding proofs for them. A selection of properties is discussed in Section 3.1.2. The advantage of (3.7) is that we can directly solve it for $\text{vec}\{\boldsymbol{X}\}$. A drawback compared to the matrix version (3.6) is that the vectorized form does not capture the structure of the model since the information that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are organized in rows and columns is lost.

Yet another popular method of expressing linear forms is given by expressions of the form $\text{trace}\{\boldsymbol{A}\cdot\boldsymbol{X}\}$ for $\boldsymbol{A}\in\mathbb{C}^{M\times N}$ and $\boldsymbol{X}\in\mathbb{C}^{N\times M}$, which is a scalar linear form in the elements of $\boldsymbol{X}$. Therefore, we must be able to write it as a linear combination of the elements of $\text{vec}\{\boldsymbol{X}\}$. This is possible via the following identity

$$\text{trace}\{\boldsymbol{A}\cdot\boldsymbol{X}\} = \text{trace}\{\boldsymbol{X}\cdot\boldsymbol{A}\} = \text{vec}\left\{\boldsymbol{A}^{\mathrm{T}}\right\}^{\mathrm{T}}\cdot\text{vec}\{\boldsymbol{X}\}, \tag{3.8}$$

which is also proven in [Neu69].

3.1.2. Kronecker, Khatri-Rao, and Schur products

The Kronecker product, the column-wise Kronecker (Khatri-Rao) product [KR68], and the element-wise (Schur) product between two matrices are denoted as $\boldsymbol{A}\otimes\boldsymbol{B}$, $\boldsymbol{A}\diamond\boldsymbol{B}$, and $\boldsymbol{A}\odot\boldsymbol{B}$, respectively. These three products are related to each other, since the elements contained in $\boldsymbol{A}\odot\boldsymbol{B}$ are a subset of the elements contained in $\boldsymbol{A}\diamond\boldsymbol{B}$ which are a subset of the elements contained in $\boldsymbol{A}\otimes\boldsymbol{B}$. Therefore, we can compute the Khatri-Rao and the Schur product from the Kronecker product by performing appropriate selections. Formally speaking, this is accomplished via the "reduction matrix" $\boldsymbol{\Xi}_n$ of size $n^2\times n$ which defined as

$$\boldsymbol{\Xi}_n = \boldsymbol{I}_n\diamond\boldsymbol{I}_n \tag{3.9}$$

For example, for $n=4$, the reduction matrix $\boldsymbol{\Xi}_4$ has the following form

$$\boldsymbol{\Xi}_4 = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{bmatrix}^{\mathrm{T}} \tag{3.10}$$

These reduction matrices establish the link between Kronecker, Khatri-Rao, and Schur product given by

$$\boldsymbol{A} \diamond \boldsymbol{B} = (\boldsymbol{A} \otimes \boldsymbol{B}) \cdot \Xi_N \tag{3.11}$$

$$\boldsymbol{A} \odot \boldsymbol{C} = \Xi_M^{\mathrm{T}} \cdot (\boldsymbol{A} \otimes \boldsymbol{C}) \cdot \Xi_N \tag{3.12}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{B} \in \mathbb{C}^{P \times N}$, $\boldsymbol{C} \in \mathbb{C}^{M \times N}$. Note that Ξ_n can also be used to extract the diagonal elements of a matrix from its vectorized version since for $\boldsymbol{X} \in \mathbb{C}^{N \times N}$ we have

$$\Xi_N^{\mathrm{T}} \cdot \mathrm{vec}\{\boldsymbol{X}\} = \mathrm{diag}\{\boldsymbol{X}\}, \tag{3.13}$$

where $\mathrm{diag}\{\boldsymbol{X}\}$ returns a column vector containing the main diagonal elements of $\boldsymbol{X}$.

There are many useful algebraic properties involving these three products, e.g., in [Bre78] or in [PP08]. A few of the properties we need later on are summarized in the following identities

$$(\boldsymbol{A} \otimes \boldsymbol{B})^{\mathrm{H}} = \boldsymbol{A}^{\mathrm{H}} \otimes \boldsymbol{B}^{\mathrm{H}} \tag{3.14}$$

$$(\boldsymbol{A} \otimes \boldsymbol{B})^{+} = \boldsymbol{A}^{+} \otimes \boldsymbol{B}^{+} \tag{3.15}$$

$$\|\boldsymbol{A} \otimes \boldsymbol{B}\|_{\mathrm{F}} = \|\boldsymbol{A}\|_{\mathrm{F}} \cdot \|\boldsymbol{B}\|_{\mathrm{F}} \tag{3.16}$$

$$(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot (\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{A} \cdot \boldsymbol{C}) \otimes (\boldsymbol{B} \cdot \boldsymbol{D}) \tag{3.17}$$

$$(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot (\boldsymbol{C} \diamond \boldsymbol{E}) = (\boldsymbol{A} \cdot \boldsymbol{C}) \diamond (\boldsymbol{B} \cdot \boldsymbol{E}) \tag{3.18}$$

$$(\boldsymbol{C} \diamond \boldsymbol{E})^{\mathrm{H}} \cdot (\boldsymbol{C} \diamond \boldsymbol{E}) = \left(\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{C}\right) \odot \left(\boldsymbol{E}^{\mathrm{H}} \cdot \boldsymbol{E}\right) \tag{3.19}$$

$$(\boldsymbol{C} \diamond \boldsymbol{E}) \cdot \mathrm{diag}\{\boldsymbol{f}\} = ((\boldsymbol{C} \cdot \mathrm{diag}\{\boldsymbol{f}\}) \diamond \boldsymbol{E}) = (\boldsymbol{C} \diamond (\boldsymbol{E} \cdot \mathrm{diag}\{\boldsymbol{f}\})) \tag{3.20}$$

$$\mathrm{diag}\{\boldsymbol{g}\} \cdot \boldsymbol{E} \cdot \mathrm{diag}\{\boldsymbol{f}\} = \boldsymbol{E} \odot \left(\boldsymbol{g} \cdot \boldsymbol{f}^{\mathrm{T}}\right), \tag{3.21}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{B} \in \mathbb{C}^{P \times Q}$, $\boldsymbol{C} \in \mathbb{C}^{N \times R}$, $\boldsymbol{D} \in \mathbb{C}^{Q \times S}$, $\boldsymbol{E} \in \mathbb{C}^{Q \times R}$, $\boldsymbol{f} \in \mathbb{C}^{R \times 1}$, and $\boldsymbol{g} \in \mathbb{C}^{Q \times 1}$. Moreover, $^{+}$ denotes the Moore-Penrose pseudo-inverse [Moo20, Pen55] (cf. its definition in Appendix A.2), and $\|\cdot\|_{\mathrm{F}}$ the Frobenius norm[1]. Note that (3.19) follows by combining (3.12) and (3.18). Via (3.19), the pseudo-inverse of a Khatri-Rao product can be computed in the following manner

$$(\boldsymbol{C} \diamond \boldsymbol{E})^{+} = \left[\left(\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{C}\right) \odot \left(\boldsymbol{E}^{\mathrm{H}} \cdot \boldsymbol{E}\right)\right]^{-1} \cdot (\boldsymbol{C} \diamond \boldsymbol{E})^{\mathrm{H}}, \tag{3.22}$$

if $\left(\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{C}\right) \odot \left(\boldsymbol{E}^{\mathrm{H}} \cdot \boldsymbol{E}\right)$ is full rank (which implies $N \cdot Q \geq R$).

[1]In fact, (3.16) is true for a wider class of norms, including the spectral norm, the 1-norm, or the infinity-norm. However, we only need the Frobenius norm (and the special case of the Euclidean norm for vectors) here.

Another important observation is that $\boldsymbol{A} \otimes \boldsymbol{X}$ and $\boldsymbol{X} \otimes \boldsymbol{A}$ are matrices of linear forms in the elements of $\boldsymbol{X}$ and hence, it must be possible to reformulate them into the canonical vector linear form. This is accomplished via the following proposition.

Proposition 3.1.1. *The vectorization of a Kronecker product in terms of one of its arguments is accomplished via the following identities*

$$\operatorname{vec}\{\boldsymbol{A} \otimes \boldsymbol{X}\} = \left(\begin{bmatrix} \boldsymbol{I}_Q \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_Q \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_Q \otimes \boldsymbol{a}_N \end{bmatrix} \otimes \boldsymbol{I}_P \right) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.23}$$

$$\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\} = \left(\boldsymbol{I}_Q \otimes \begin{bmatrix} \boldsymbol{I}_P \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_P \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_P \otimes \boldsymbol{a}_N \end{bmatrix} \right) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.24}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$ *and* $\boldsymbol{X} \in \mathbb{C}^{P \times Q}$.

Proof: cf. Appendix B.1.

Note that these identities have interesting special cases when $\boldsymbol{A}$ or $\boldsymbol{X}$ are column vectors (i.e., $N = 1$ or $Q = 1$). In this case, the general expressions simplify to

$$\operatorname{vec}\{\boldsymbol{A} \otimes \boldsymbol{x}\} = (\operatorname{vec}\{\boldsymbol{A}\} \otimes \boldsymbol{I}_P) \cdot \boldsymbol{x} \tag{3.25}$$

$$\operatorname{vec}\{\boldsymbol{x} \otimes \boldsymbol{A}\} = \begin{bmatrix} \boldsymbol{I}_P \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_P \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_P \otimes \boldsymbol{a}_N \end{bmatrix} \cdot \boldsymbol{x} \tag{3.26}$$

$$\operatorname{vec}\{\boldsymbol{a} \otimes \boldsymbol{X}\} = (\boldsymbol{I}_Q \otimes \boldsymbol{a} \otimes \boldsymbol{I}_P) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.27}$$

$$\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{a}\} = (\boldsymbol{I}_{P \cdot Q} \otimes \boldsymbol{a}) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.28}$$

$$\operatorname{vec}\{\boldsymbol{a} \otimes \boldsymbol{x}\} = \boldsymbol{a} \otimes \boldsymbol{x} = (\boldsymbol{a} \otimes \boldsymbol{I}_P) \cdot \boldsymbol{x} \tag{3.29}$$

$$\operatorname{vec}\{\boldsymbol{x} \otimes \boldsymbol{a}\} = \boldsymbol{x} \otimes \boldsymbol{a} = (\boldsymbol{I}_P \otimes \boldsymbol{a}) \cdot \boldsymbol{x} \tag{3.30}$$

So far, we have shown how to reformulate a Kronecker product linear form in terms of a canonical vector linear form. However, for the Khatri-Rao product and the Schur product, we can argue in a similar manner: $\boldsymbol{A} \diamond \boldsymbol{X}$, $\boldsymbol{X} \diamond \boldsymbol{A}$, and $\boldsymbol{A} \odot \boldsymbol{X} = \boldsymbol{X} \odot \boldsymbol{A}$ are linear forms in $\boldsymbol{X}$ as well. Their vectorization is achieved via the following proposition.

Proposition 3.1.2. *The vectorization of a Khatri-Rao product and a Schur product in terms of one of its arguments is accomplished via the following identities*

$$\operatorname{vec}\{\boldsymbol{A} \diamond \boldsymbol{X}\} = ([\boldsymbol{I}_N \diamond \boldsymbol{A}] \otimes \boldsymbol{I}_P) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.31}$$

$$\operatorname{vec}\{\boldsymbol{X} \diamond \boldsymbol{A}\} = \left[\boldsymbol{I}_{P \cdot N} \diamond \left(\boldsymbol{A} \cdot \left[\boldsymbol{I}_N \otimes \boldsymbol{1}_{P\times 1}^{\mathrm{T}}\right]\right)\right] \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{3.32}$$

$$\operatorname{vec}\{\boldsymbol{B} \odot \boldsymbol{X}\} = \operatorname{vec}\{\boldsymbol{X} \odot \boldsymbol{B}\} = \operatorname{diag}\{\operatorname{vec}\{\boldsymbol{B}\}\} \cdot \operatorname{vec}\{\boldsymbol{X}\}, \tag{3.33}$$

where $\boldsymbol{A} \in \mathbb{C}^{M\times N}$, $\boldsymbol{X} \in \mathbb{C}^{P\times N}$, $\boldsymbol{B} \in \mathbb{C}^{P\times N}$, *and* $\boldsymbol{1}_{p\times q}$ *represents a* $p \times q$ *matrix filled with ones. Moreover,* $\operatorname{diag}\{\boldsymbol{x}\}$ *yields a square diagonal matrix with the elements of* $\boldsymbol{x}$ *on its main diagonal.*

Proof: Appendix B.2.

Finally, it is sometimes required to permute the order of the matrices in a Kronecker product, i.e., transform $\boldsymbol{A}\otimes\boldsymbol{B}$ into $\boldsymbol{B}\otimes\boldsymbol{A}$. This can be achieved via a special set of permutation matrices, which were introduced under the name "commutation matrices" in [MN95]. The commutation matrix $\boldsymbol{K}_{M,N}$ of size $M \cdot N \times M \cdot N$ is the unique permutation matrix which satisfies

$$\boldsymbol{K}_{M,N} \cdot \operatorname{vec}\{\boldsymbol{X}\} = \operatorname{vec}\left\{\boldsymbol{X}^{\mathrm{T}}\right\} \tag{3.34}$$

for arbitrary matrices $\boldsymbol{X} \in \mathbb{C}^{M\times N}$. In Appendix B.3 it is shown how these commutation matrices can be directly computed from a "permuted column" Kronecker product of two identity matrices. A direct consequence of this identity is that the commutation matrices can be used to permute Kronecker products. In fact, we have [MN79]

$$\boldsymbol{K}_{M,N}^{\mathrm{T}} \cdot (\boldsymbol{A} \otimes \boldsymbol{B}) \cdot \boldsymbol{K}_{P,Q} = \boldsymbol{B} \otimes \boldsymbol{A} \tag{3.35}$$

for $\boldsymbol{A} \in \mathbb{C}^{M\times P}$ and $\boldsymbol{B} \in \mathbb{C}^{N\times Q}$.

As shown in [MN95] with the help of the commutation matrices, the following alternative expressions for the vectorization of Kronecker products introduced in Proposition 3.1.1 can be derived

$$\operatorname{vec}\{\boldsymbol{A} \otimes \boldsymbol{B}\} = \Big(\boldsymbol{I}_N \otimes \Big((\boldsymbol{K}_{Q,M} \otimes \boldsymbol{I}_P) \cdot (\boldsymbol{I}_M \otimes \operatorname{vec}\{\boldsymbol{B}\})\Big)\Big) \cdot \operatorname{vec}\{\boldsymbol{A}\} \tag{3.36}$$

$$= \Big(\Big((\boldsymbol{I}_N \otimes \boldsymbol{K}_{Q,M}) \cdot (\operatorname{vec}\{\boldsymbol{A}\} \otimes \boldsymbol{I}_Q)\Big) \otimes \boldsymbol{I}_P\Big) \cdot \operatorname{vec}\{\boldsymbol{B}\}. \tag{3.37}$$

However, the form shown in Proposition 3.1.1 is better suited for our derivations since it does not involve the commutation matrices and is hence easier to simplify.

3.1.3. Quadratic forms

While for linear forms, only linear combinations of the parameters are allowed, for quadratic forms we can have quadratic and bilinear terms as well. Formally, a quadratic form is defined as a homogeneous polynomial of degree two in N variables [Sch85]. Similar to a linear form, we can see quadratic forms as mappings from $\boldsymbol{x} \in \mathbb{C}^N$ to $\mathbb{C}$. Moreover, any homogeneous polynomial in N variables can be expressed via

$$q(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{R} \cdot \boldsymbol{x} \tag{3.38}$$

where $\boldsymbol{R} \in \mathbb{C}^{N \times N}$. We refer to (3.38) as the "canonical" expression for quadratic forms (in line with the "canonical" linear vector forms from (3.3)). In general, $q(\boldsymbol{x})$ is a complex number. However, if $\boldsymbol{R}$ is a Hermitian symmetric matrix, i.e., $\boldsymbol{R}^{\mathrm{H}} = \boldsymbol{R}$ is is easy to see that $q(\boldsymbol{x})^* = q(\boldsymbol{x})^{\mathrm{H}} = q(\boldsymbol{x})$ and hence $q(\boldsymbol{x}) \in \mathbb{R}$. Moreover, if $\boldsymbol{R}$ is a positive (negative) semi-definite matrix, we have $q(\boldsymbol{x}) \geq 0, \forall \boldsymbol{x}$ $(q(\boldsymbol{x}) \leq 0, \forall \boldsymbol{x})$, with strict inequality for positive (negative) definite matrices [Str93]. Note that the class of positive (semi-)definite quadratic forms plays a major role in many signal processing related applications. This is due to the fact that mean squared quantities (such as errors or powers) can very often be expressed via such a quadratic form.

As for linear forms, we often encounter quadratic forms in a "non-canonical" expression. In this case it is always useful if we are able to reformulate it in terms of a canonical quadratic form. The following proposition (which cannot be found in [PP08]) provides a rather generic expression for quadratic forms from which many other expressions follow as special cases. We use it in Part IV to show that the sum-rate in a two-way relaying system with MIMO Amplify and Forward (AF) relays and single-antenna terminals can be expressed as the product of two Rayleigh quotients. Based on this observation, the semi-algebraic RAGES scheme for sum-rate maximization is derived in Section 16.4.

Proposition 3.1.3. *The expression* $\operatorname{trace}\left\{\boldsymbol{A} \cdot \boldsymbol{X} \cdot \boldsymbol{B} \cdot \boldsymbol{X}^{\mathrm{H}} \cdot \boldsymbol{C}\right\}$ *is a quadratic form in the elements of* $\boldsymbol{X} \in \mathbb{C}^{M \times N}$. *It can be reformulated in terms of a canonical quadratic form in the following manner*

$$\operatorname{trace}\left\{\boldsymbol{A} \cdot \boldsymbol{X} \cdot \boldsymbol{B} \cdot \boldsymbol{X}^{\mathrm{H}} \cdot \boldsymbol{C}^{\mathrm{H}}\right\} = \operatorname{vec}\left\{\boldsymbol{X}\right\}^{\mathrm{H}} \cdot \left(\boldsymbol{B}^{\mathrm{T}} \otimes \left(\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{A}\right)\right) \cdot \operatorname{vec}\left\{\boldsymbol{X}\right\}, \tag{3.39}$$

where $\boldsymbol{B} \in \mathbb{C}^{N \times N}$ *and* $\boldsymbol{A}, \boldsymbol{C} \in \mathbb{C}^{P \times M}$.

Proof: cf. Appendix B.4.

Note that, as a corollary, the following useful identities follow

$$\operatorname{trace}\left\{\boldsymbol{E}\cdot\boldsymbol{X}^{\mathrm{H}}\cdot\boldsymbol{D}\cdot\boldsymbol{X}\cdot\boldsymbol{F}^{\mathrm{H}}\right\} = \operatorname{vec}\{\boldsymbol{X}\}^{\mathrm{H}}\cdot\left(\left(\boldsymbol{E}^{\mathrm{T}}\cdot\boldsymbol{F}^{*}\right)\otimes\boldsymbol{D}\right)\cdot\operatorname{vec}\{\boldsymbol{X}\} \tag{3.40}$$

$$\boldsymbol{a}^{\mathrm{H}}\cdot\boldsymbol{X}\cdot\boldsymbol{B}\cdot\boldsymbol{X}^{\mathrm{H}}\cdot\boldsymbol{a} = \operatorname{vec}\{\boldsymbol{X}\}^{\mathrm{H}}\cdot\left(\boldsymbol{B}^{\mathrm{T}}\otimes\left(\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}\right)\right)\cdot\operatorname{vec}\{\boldsymbol{X}\} \tag{3.41}$$

$$\operatorname{trace}\left\{\boldsymbol{X}\cdot\boldsymbol{B}\cdot\boldsymbol{X}^{\mathrm{H}}\right\} = \operatorname{vec}\{\boldsymbol{X}\}^{\mathrm{H}}\cdot\left(\boldsymbol{B}^{\mathrm{T}}\otimes\boldsymbol{I}_M\right)\cdot\operatorname{vec}\{\boldsymbol{X}\} \tag{3.42}$$

$$\operatorname{trace}\left\{\boldsymbol{B}\cdot\boldsymbol{x}\cdot\boldsymbol{x}^{\mathrm{H}}\right\} = \boldsymbol{x}^{\mathrm{H}}\cdot\boldsymbol{B}\cdot\boldsymbol{x}, \tag{3.43}$$

where $\boldsymbol{D}\in\mathbb{C}^{M\times M}$, $\boldsymbol{E},\boldsymbol{F}\in\mathbb{C}^{P\times N}$, $\boldsymbol{a}\in\mathbb{C}^{M\times 1}$, and $\boldsymbol{x}\in\mathbb{C}^{N\times 1}$.

3.2. Eigenvalues and singular values

In the previous section we have seen how quadratic forms can be reformulated algebraically into their "canonical" form $\boldsymbol{x}^{\mathrm{H}}\cdot\boldsymbol{R}\cdot\boldsymbol{x}$. In this section we provide the algebraic link between this form and eigenvectors of square matrices. Note that this is often the key to finding algebraic solutions to given problems, as the later parts of this thesis exemplify.

For every square diagonalizable[2] matrix $\boldsymbol{R}\in\mathbb{C}^{N\times N}$, there exists a set of N linearly independent vectors $\boldsymbol{q}_n$, such that $\boldsymbol{R}\cdot\boldsymbol{q}_n = \lambda_n\cdot\boldsymbol{q}_n$ for $n = 1,2,\ldots,N$. The vectors $\boldsymbol{q}_n$ are the eigenvectors and the scalars λ_n the corresponding eigenvalues of $\boldsymbol{R}$. Eigenvectors are unique only up to a scaling with an arbitrary complex number[3]: if $\boldsymbol{q}_n$ is an eigenvector then $\alpha\cdot\boldsymbol{q}_n$ is also an eigenvector for any $\alpha\in\mathbb{C}_{\neq 0}$.

There is a strong link between quadratic forms and eigenvalues, which is established by the following theorem:

Theorem 3.2.1. **[Str93]** *Consider the following ratio of quadratic forms*

$$r(\boldsymbol{x}) = \frac{\boldsymbol{x}^{\mathrm{H}}\cdot\boldsymbol{R}\cdot\boldsymbol{x}}{\boldsymbol{x}^{\mathrm{H}}\cdot\boldsymbol{x}} \tag{3.44}$$

for a Hermitian matrix $\boldsymbol{R}\in\mathbb{C}^{N\times N}$. Then we have $\lambda_{\min}(\boldsymbol{R})\leq r(\boldsymbol{x})\leq\lambda_{\max}(\boldsymbol{R})$ with equality for $\boldsymbol{x}=\boldsymbol{q}_{\min}$ and $\boldsymbol{x}=\boldsymbol{q}_{\max}$, respectively. Here, $\lambda_{\min}(\boldsymbol{R})$ and $\lambda_{\max}(\boldsymbol{R})$ denote the largest and the smallest eigenvalue of $\boldsymbol{R}$, and the corresponding eigenvectors are $\boldsymbol{q}_{\min}$ and $\boldsymbol{q}_{\max}$.

The ratio in (3.44) is also called Rayleigh quotient or Rayleigh-Ritz ratio. A proof for this theorem is found in many standard algebra textbooks, for instance, [Str93]. In other words

[2] A matrix is diagonalizable if for each of its eigenvalues the algebraic and the geometric multiplicity agrees [GvL80]. A counter-example is a matrix filled with zeros and a single off-diagonal non-zero entry.

[3] For eigenvectors associated to eigenvalues with multiplicity larger than one, ambiguity is even weaker: any basis for the null space of $\boldsymbol{R}-\lambda_n\cdot\boldsymbol{I}$ can be chosen.

the theorem implies that the unconstrained maximization or minimization of such ratios has an algebraic solution given by one of the eigenvectors of the matrix $\boldsymbol{R}$. In fact, any local extremum of $r(\boldsymbol{x})$ is also an eigenvector of $\boldsymbol{R}$.

A related optimization problem is the maximization or minimization of the quadratic form $\boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{R} \cdot \boldsymbol{x}$ subject to the constraint $\|\boldsymbol{x}\|_2^2 = \boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{x} = 1$. In fact, it can easily be shown via the method of Lagrange multipliers that all local extrema of this constrained optimization problem are eigenvectors of $\boldsymbol{R}$ scaled to unit norm.

A natural extension of the Rayleigh quotient is to consider the ratio of two quadratic forms, i.e.,

$$\bar{r}(\boldsymbol{x}) = \frac{\boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{Q}_1 \cdot \boldsymbol{x}}{\boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{Q}_2 \cdot \boldsymbol{x}} \tag{3.45}$$

for Hermitian matrices $\boldsymbol{Q}_1, \boldsymbol{Q}_2 \in \mathbb{C}^{N \times N}$. The ratio in (3.45) is also referred to as the generalized Rayleigh quotient [Str93] since (3.44) is a special case of (3.45) for $\boldsymbol{Q}_2 = \boldsymbol{I}_N$. Likewise, its extremal values are referred to as the generalized eigenvalues of the matrix pair $(\boldsymbol{Q}_1, \boldsymbol{Q}_2)$ and the vectors yielding these extrema are called generalized eigenvectors. In the special case where $\boldsymbol{Q}_2$ is invertible, the generalized Rayleigh quotient is equal to the Rayleigh quotient of the matrix $\boldsymbol{R} = \boldsymbol{Q}_2^{-1} \cdot \boldsymbol{Q}_1$. Therefore, the maxima and minima are readily found from the eigenvectors of $\boldsymbol{Q}_2^{-1} \cdot \boldsymbol{Q}_1$. However, if $\boldsymbol{Q}_2$ is not invertible, the denominator can become zero and hence $\bar{r}(\boldsymbol{x})$ can become infinitely large. In this case a maximum does not exist, even though we can formally assign generalized eigenvectors to these infinitely large generalized eigenvalues. Note that the finite generalized eigenvalues can be computed as the roots of the determinant of the matrix $\boldsymbol{Q}_1 + \lambda \boldsymbol{Q}_2$. Matrices of this form are also referred to as “matrix pencils” [HS91, Zol88, OWS88].

3.3. Matrix factorization

Another powerful concept in matrix algebra is finding a factorization of a given matrix into factors with specific properties. In general, such a factorization is not unique. For example, consider a square matrix $\boldsymbol{X} \in \mathbb{C}^{M \times M}$ which we want to factorize into two factors $\boldsymbol{A}$ and $\boldsymbol{B}$, such that $\boldsymbol{X} = \boldsymbol{A} \cdot \boldsymbol{B}$. Without additional constraints, $\boldsymbol{A}$ can be chosen as an arbitrary full rank matrix if we set $\boldsymbol{B} = \boldsymbol{A}^{-1} \cdot \boldsymbol{X}$. The situation improves if the given matrix is known to possess a low rank. To this end, consider a matrix $\boldsymbol{X} \in \mathbb{C}^{M \times N}$ with $\operatorname{rank}\{\boldsymbol{X}\} = r < \min\{M, N\}$. Then, we can decompose $\boldsymbol{X}$ into $\boldsymbol{A} \cdot \boldsymbol{B}$ for $\boldsymbol{A} \in \mathbb{C}^{M \times r}$ and $\boldsymbol{B} \in \mathbb{C}^{r \times N}$. This decomposition is unique up to an $r \times r$ full rank matrix $\boldsymbol{T}$, since

$$\boldsymbol{X} = \boldsymbol{A} \cdot \boldsymbol{B} = \underbrace{\boldsymbol{A} \cdot \boldsymbol{T}}_{\bar{\boldsymbol{A}}} \cdot \underbrace{\boldsymbol{T}^{-1} \cdot \boldsymbol{B}}_{\bar{\boldsymbol{B}}} = \bar{\boldsymbol{A}} \cdot \bar{\boldsymbol{B}} \quad \forall \boldsymbol{T} \in \mathbb{C}^{r \times r}. \tag{3.46}$$

In order to reduce the ambiguities further, we can specify additional constraints for the factors $\boldsymbol{A}$ and $\boldsymbol{B}$. Examples include [GvL96]:

- $\boldsymbol{A}$ should be upper triangular, $\boldsymbol{B}$ lower triangular with ones on its main diagonal. This results in an L-U-factorization.
- $\boldsymbol{A}$ should be unitary, $\boldsymbol{B}$ upper triangular. This results in a Q-R-factorization.
- For Hermitian positive semi-definite matrices $\boldsymbol{X}$: $\boldsymbol{A}$ should be lower triangular with non-negative diagonal entries and $\boldsymbol{B} = \boldsymbol{A}^{\mathrm{H}}$. This results in a Cholesky factorization.

Each of this decompositions is *essentially unique* meaning that a unique factorization is provided up to some remaining indeterminacies which are considered to be irrelevant. These relate to the set of all matrices $\boldsymbol{T}$ which, given a valid decomposition $\boldsymbol{A} \cdot \boldsymbol{B}$, yield new factors $\bar{\boldsymbol{A}} = \boldsymbol{A} \cdot \boldsymbol{T}$ and $\bar{\boldsymbol{B}} = \boldsymbol{T}^{-1} \cdot \boldsymbol{B}$ which still satisfy all the constraints that were set to find $\boldsymbol{A}$ and $\boldsymbol{B}$. The fact that matrix decompositions are not unique unless additional constraints are introduced is one of their major drawbacks if we want to apply them to separate a given signal into individual components, because such constraints may lack practical meaning. As we show in Section 4.2 this drawback is alleviated if we consider more than two dimensions and define trilinear (3-D) or in general multilinear (R-D) decompositions. For instance, the Canonical Polyadic (CP) decomposition is essentially unique without additional constraints on its factor matrices.

Another very important decomposition is given by the eigenvalue decomposition (EVD), which is applicable to arbitrary square diagonalizable matrices $\boldsymbol{R}$ (defined in Section 3.2) and can be expressed as

$$\boldsymbol{R} = \boldsymbol{Q} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{-1}, \tag{3.47}$$

where $\boldsymbol{Q} \in \mathbb{C}^{N \times N}$ contains the N eigenvectors of $\boldsymbol{R}$ in its columns and $\boldsymbol{\Lambda} = \operatorname{diag}\{[\lambda_1, \ldots, \lambda_N]\} \in \mathbb{C}^{N \times N}$ is a diagonal matrix of eigenvalues. If all eigenvalues are distinct, this decomposition is unique up to permutation of the columns of $\boldsymbol{Q}$ (and the corresponding elements on the diagonal of $\boldsymbol{\Lambda}$) and scaling of the columns of $\boldsymbol{Q}$ with arbitrary non-zero complex coefficients. The latter ambiguity can be reduced to scaling by arbitrary phase terms if all eigenvectors are normalized to norm one. Note that for Hermitian matrices $\boldsymbol{R}$, the eigenvectors are mutually orthogonal and hence $\boldsymbol{Q}^{-1}$ can be replaced by $\boldsymbol{Q}^{\mathrm{H}}$ if the eigenvectors are scaled to norm one.

For non-square matrices $\boldsymbol{X} \in \mathbb{C}^{M \times N}$, $\operatorname{rank}\{\boldsymbol{X}\} = r$, a generalization of this concept is given by the singular value decomposition (SVD) (see [Ste93] for historical notes) which can be written into

$$\boldsymbol{X} = \boldsymbol{U} \cdot \boldsymbol{\Sigma} \cdot \boldsymbol{V}^{\mathrm{H}}, \tag{3.48}$$

where $\boldsymbol{U} \in \mathbb{C}^{M \times M}$ and $\boldsymbol{V} \in \mathbb{C}^{N \times N}$ are square unitary matrices and $\boldsymbol{\Sigma} \in \mathbb{R}^{M \times N}$ contains the r non-zero singular values $[\sigma_1, \sigma_2, \ldots, \sigma_r]$ on its main diagonal and zeros elsewhere. The singular values are ordered by magnitude, i.e., $\sigma_1 \geq \sigma_2 \ldots \geq \sigma_r$, which provide a unique ordering as long as they are distinct. In this case, the SVD is unique up to one phase term $\mathrm{e}^{\jmath\varphi_n}$ per column of $\boldsymbol{U}$ since the opposite phase term $\mathrm{e}^{-\jmath\varphi_n}$ can be multiplied to the corresponding column of $\boldsymbol{V}$ without changing the result. Since only the first r rows and columns of $\boldsymbol{\Sigma}$ are non-zero, (3.48) can alternatively be expressed in more compact ("economy-size") form as

$$\boldsymbol{X} = \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}} \cdot \boldsymbol{V}_{\mathrm{s}}^{\mathrm{H}}, \tag{3.49}$$

where $\boldsymbol{U}_{\mathrm{s}}$ and $\boldsymbol{V}_{\mathrm{s}}$ contain the first r columns of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively, and $\boldsymbol{\Sigma}_{\mathrm{s}}$ is given be the upper-left $(r \times r)$ block of $\boldsymbol{\Sigma}$. Finally, note that truncating an SVD to $r' < \operatorname{rank}\{\boldsymbol{X}\}$ columns in $\boldsymbol{U}$ and $\boldsymbol{V}$ yields the optimum low-rank approximation of $\boldsymbol{X}$ in the Frobenius norm sense, i.e., the matrix $\hat{\boldsymbol{X}}$ which minimizes

$$\underset{\hat{\boldsymbol{X}} \,|\, \operatorname{rank}\{\hat{\boldsymbol{X}}\} = r'}{\arg\min} \left\| \boldsymbol{X} - \hat{\boldsymbol{X}} \right\|_{\mathrm{F}}, \tag{3.50}$$

shown by the Eckart-Young theorem [EY36]. This fundamental result forms the basis for subspace-based parameter estimation schemes, which are discussed in Part III of this thesis.

For Hermitian symmetric matrices $\boldsymbol{X} = \boldsymbol{X}^{\mathrm{H}} \in \mathbb{C}^{M \times M}$, the EVD and the SVD are highly related. It can be shown that such matrices possess a set of M mutually orthogonal eigenvectors. Hence, if these eigenvectors are scaled to unit norm, the eigenvector matrix $\boldsymbol{Q}$ becomes unitary and we have an EVD of the form $\boldsymbol{X} = \boldsymbol{Q} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{\mathrm{H}}$. Moreover, the eigenvalues λ_i of any Hermitian matrix are real-valued. Therefore, if the eigenvalues are distinct and we order them according to their magnitude in descending order, we have the following: (a) for positive definite matrices $\boldsymbol{X}$ the EVD and the SVD are linked[4] as $\boldsymbol{U} = \boldsymbol{Q} = \boldsymbol{V}$, $\sigma_i = \lambda_i$; (b) for negative definite matrices $\boldsymbol{X}$ we have $\boldsymbol{U} = \boldsymbol{Q} = -\boldsymbol{V}$, $\sigma_i = -\lambda_i$; (c) for indefinite matrices we have $\sigma_i = |\lambda_i|$ and $\boldsymbol{U}$ and $\boldsymbol{Q}$ agree except for permutation of the columns. For non-square matrices

[4]The matrices of eigenvectors and singular vectors are equal up to the scaling ambiguities of one phase term per column of $\boldsymbol{U}$ and $\boldsymbol{Q}$.

$\boldsymbol{X} \in \mathbb{C}^{M \times N}$ we can relate its SVD to the EVD of its Gram matrices. Particularly, if the SVD of $\boldsymbol{X}$ is given by $\boldsymbol{X} = \boldsymbol{U} \cdot \boldsymbol{\Sigma} \cdot \boldsymbol{V}^{\mathrm{H}}$ then the EVD of the Gram matrix of its rows ($\boldsymbol{X} \cdot \boldsymbol{X}^{\mathrm{H}}$) and the Gram matrix of its columns ($\boldsymbol{X}^{\mathrm{H}} \cdot \boldsymbol{X}$) can be written as

$$\boldsymbol{X} \cdot \boldsymbol{X}^{\mathrm{H}} = \boldsymbol{U} \cdot \boldsymbol{\Sigma}^2 \cdot \boldsymbol{U}^{\mathrm{H}} \quad \text{and} \quad \boldsymbol{X}^{\mathrm{H}} \cdot \boldsymbol{X} = \boldsymbol{V} \cdot \boldsymbol{\Sigma}^2 \cdot \boldsymbol{V}^{\mathrm{H}}. \tag{3.51}$$

3.4. Least-Squares Khatri-Rao and Kronecker factorization

As we have argued in the beginning of this chapter, it is quite common in signal processing applications to encounter observations that follow a linear model from which we would like to infer about the parameters of the experiment. This task is straightforward if there is no noise in the system: If we observe

$$\boldsymbol{y}_0 = \boldsymbol{A} \cdot \boldsymbol{x} \tag{3.52}$$

and $\boldsymbol{A} \in \mathbb{C}^{M \times N}$ has full column rank, then $\boldsymbol{A}^+ \cdot \boldsymbol{y}_0$ provides an exact solution for $\boldsymbol{x}$, where $^+$ denotes the Moore-Penrose pseudo inverse. However, we usually have noisy observations so that

$$\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{n} \tag{3.53}$$

and hence $\boldsymbol{y}$ may not be fully in the column space of $\boldsymbol{A}$. In this case, $\boldsymbol{x}_{\mathrm{LS}} = \boldsymbol{A}^+ \cdot \boldsymbol{y}$ provides the Least Squares (LS) solution which minimizes the error (Euclidean norm) between $\boldsymbol{y}$ and $\boldsymbol{A} \cdot \boldsymbol{x}$ [Str93].

As we have seen previously, even for linear models, our parameters may be "hidden" more deeply, e.g., in a matrix product (which can be vectorized via (3.7)) or in a Khatri-Rao or Kronecker product. We may encounter situations where our observations are an estimate of a Khatri-Rao product or a Kronecker product which we would like to factorize (e.g., for the R-D extension of SECSI discussed in Chapter 7 or in the TENCE algorithm shown in Chapter 15). This is the motivation behind considering Least-Squares Khatri-Rao and Kronecker factorizations.

We can distinguish two different cases: one factor being unknown or both factors being unknown. In the first case, where **one factor is unknown** our task is the following:

$$\text{given } \boldsymbol{B} \approx \boldsymbol{A} \diamond \boldsymbol{X}, \text{ find } \boldsymbol{X} \text{ which minimizes } \|\boldsymbol{B} - \boldsymbol{A} \diamond \boldsymbol{X}\|_{\mathrm{F}}^2, \text{ or}$$
$$\text{given } \boldsymbol{D} \approx \boldsymbol{C} \otimes \boldsymbol{Y}, \text{ find } \boldsymbol{Y} \text{ which minimizes } \|\boldsymbol{D} - \boldsymbol{C} \otimes \boldsymbol{Y}\|_{\mathrm{F}}^2,$$

respectively. Here, the dimensions of the matrices are $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{X} \in \mathbb{C}^{P \times N}$, $\boldsymbol{C} \in \mathbb{C}^{M \times N}$ and $\boldsymbol{Y} \in \mathbb{C}^{P \times Q}$. However, since $\boldsymbol{A} \diamond \boldsymbol{X}$ and $\boldsymbol{C} \otimes \boldsymbol{Y}$ are matrices of linear forms in the elements of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively, this is a LS-optimal linear fitting task. Therefore, we can apply Proposition 3.1.2 and Proposition 3.1.1 to reformulate the Kronecker and Khatri-Rao product into the canonical vector linear forms via the vec-operator and then apply the LS solution discussed above. Consequently, we directly obtain the LS-optimal solutions $\boldsymbol{X}_{\mathrm{LS}}$ and $\boldsymbol{Y}_{\mathrm{LS}}$ as

$$\operatorname{vec}\{\boldsymbol{X}_{\mathrm{LS}}\} = \left([\boldsymbol{I}_N \diamond \boldsymbol{A}] \otimes \boldsymbol{I}_P\right)^+ \cdot \operatorname{vec}\{\boldsymbol{B}\} \tag{3.54}$$

$$\operatorname{vec}\{\boldsymbol{Y}_{\mathrm{LS}}\} = \left(\begin{bmatrix} \boldsymbol{I}_Q \otimes \boldsymbol{c}_1 \\ \boldsymbol{I}_Q \otimes \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{I}_Q \otimes \boldsymbol{c}_N \end{bmatrix}^+ \otimes \boldsymbol{I}_P\right) \cdot \operatorname{vec}\{\boldsymbol{D}\} \tag{3.55}$$

Note that since we have always more observations than unknowns ($M \cdot N \cdot P$ observations in $\boldsymbol{B}$ for $N \cdot P$ unknowns in $\boldsymbol{X}$ for the Khatri-Rao factorization and $M \cdot N \cdot P \cdot Q$ observations in $\boldsymbol{D}$ for the $N \cdot Q$ unknowns in $\boldsymbol{Y}$), these LS solutions are almost surely unique. For (3.54) it is sufficient that none of the column vectors of $\boldsymbol{A}$ is equal to the zero vector, whereas (3.55) only requires that the matrix $\boldsymbol{C}$ is not equal to the zero matrix.

The factorization of the permuted quantities $\boldsymbol{B} \approx \boldsymbol{X} \diamond \boldsymbol{A}$ and $\boldsymbol{D} \approx \boldsymbol{Y} \otimes \boldsymbol{C}$ proceeds in a similar fashion. Via the corresponding expressions for the vectorized versions from in Proposition 3.1.2 and Proposition 3.1.1 we find the solutions

$$\operatorname{vec}\{\boldsymbol{X}_{\mathrm{LS}}\} = \left[\boldsymbol{I}_{P \cdot N} \diamond \left(\boldsymbol{A} \cdot \left[\boldsymbol{I}_N \otimes \boldsymbol{1}_{P \times 1}^{\mathrm{T}}\right]\right)\right]^+ \cdot \operatorname{vec}\{\boldsymbol{B}\} \tag{3.56}$$

$$\operatorname{vec}\{\boldsymbol{Y}_{\mathrm{LS}}\} = \left(\boldsymbol{I}_Q \otimes \begin{bmatrix} \boldsymbol{I}_P \otimes \boldsymbol{c}_1 \\ \boldsymbol{I}_P \otimes \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{I}_P \otimes \boldsymbol{c}_N \end{bmatrix}^+\right) \cdot \operatorname{vec}\{\boldsymbol{D}\} \tag{3.57}$$

In Appendix B.6 we show that (3.54)-(3.57) can be further simplified and relate the resulting expressions to the more complicated ones provided in [LP93].

We now turn to the second case where **both factors are unknown**. In this case our task is formulated as:

$$\text{given } \boldsymbol{A} \approx \boldsymbol{W} \diamond \boldsymbol{X} \text{ find } \boldsymbol{X}, \boldsymbol{W} \text{ which minimizes } \|\boldsymbol{A} - \boldsymbol{W} \diamond \boldsymbol{X}\|_{\mathrm{F}}^2, \text{ or}$$
$$\text{given } \boldsymbol{B} \approx \boldsymbol{Y} \otimes \boldsymbol{Z}, \text{ find } \boldsymbol{Y}, \boldsymbol{Z} \text{ which minimizes } \|\boldsymbol{B} - \boldsymbol{Y} \otimes \boldsymbol{Z}\|_{\mathrm{F}}^2,$$

where the dimensions of the matrices are $\boldsymbol{W} \in \mathbb{C}^{M\times P}$, $\boldsymbol{X} \in \mathbb{C}^{N\times P}$, $\boldsymbol{Y} \in \mathbb{C}^{M\times N}$, $\boldsymbol{Z} \in \mathbb{C}^{P\times Q}$. The Kronecker factorization was first shown in [LP93] where it referred to as the "nearest Kronecker product" (NKP) problem. The main idea behind the factorization is the following: Every Kronecker product resembles a collection of all pair-wise products of its elements. This is equivalent to a rank-one matrix constructed from the outer product of two vectors if we arrange the corresponding elements into the matrix in the correct manner. Therefore, in the presence of noise, we have a matrix which is approximately rank-one. However, due to the Eckart-Young theorem we know that the truncated SVD provides the best rank-one approximation in the LS sense. Therefore, we can use it for Kronecker and Khatri-Rao factorization. The difference is that in the latter case, we proceed column by column whereas Kronecker factorization is achieved in one step. The corresponding algorithms are shown in Algorithm 1 and Algorithm 2.

Algorithm 1 Least-Squares factorization of a Khatri-Rao product

- Consider a matrix $\boldsymbol{A} \in \mathbb{C}^{M\cdot N\times P}$ which is an approximation of the Khatri-Rao product between a matrix $\boldsymbol{W} \in \mathbb{C}^{M\times P}$ and a matrix $\boldsymbol{X} \in \mathbb{C}^{N\times P}$, i.e., $\boldsymbol{A} \approx \boldsymbol{W} \diamond \boldsymbol{X}$.
- Set $p = 1$.
  1. Let $\boldsymbol{a}_p$, $\boldsymbol{w}_p$, and $\boldsymbol{x}_p$ be the p-th columns of the matrices $\boldsymbol{A}$, $\boldsymbol{W}$, and $\boldsymbol{X}$, respectively. By the definition of the Khatri-Rao product we then have $\boldsymbol{a}_p \approx \boldsymbol{w}_p \otimes \boldsymbol{x}_p$.
  2. Reshape the vector $\boldsymbol{a}_p$ into a matrix $\tilde{\boldsymbol{A}}_p \in \mathbb{C}^{N\times M}$, such that $\operatorname{vec}\{\tilde{\boldsymbol{A}}_p\} = \boldsymbol{a}_p$. It is easy to see that this matrix satisfies $\tilde{\boldsymbol{A}}_p \approx \boldsymbol{x}_p \cdot \boldsymbol{w}_p^{\mathrm{T}}$.
  3. Compute the singular value decomposition of $\tilde{\boldsymbol{A}}_p$ as $\tilde{\boldsymbol{A}}_p = \boldsymbol{U}_p \boldsymbol{\Sigma}_p \boldsymbol{V}_p^{\mathrm{H}}$. Now the best rank-one approximation of $\tilde{\boldsymbol{A}}_p$ is given by truncating the SVD, i.e., $\hat{\boldsymbol{w}}_p = \sqrt{\sigma_1} \cdot \boldsymbol{v}_1^*$ and $\hat{\boldsymbol{x}}_p = \sqrt{\sigma_1} \cdot \boldsymbol{u}_1$, where $\boldsymbol{u}_1$ and $\boldsymbol{v}_1$ represent the first column vectors of $\boldsymbol{U}_p$ and $\boldsymbol{V}_p$, respectively, and σ_1 is the largest singular value.
  4. If $p < P$, set $p = p + 1$ and go to 1).
- The final solution is given by $\hat{\boldsymbol{W}} = [\hat{\boldsymbol{w}}_1, \ldots, \hat{\boldsymbol{w}}_P]$ and $\hat{\boldsymbol{X}} = [\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_P]$.

It is important to note that the "Least Squares Khatri-Rao factorization" presented in Algorithm 1 is not completely unique. There is a scaling ambiguity of one non-zero complex number per column p in every Khatri-Rao product, since $\boldsymbol{x}_p \otimes \boldsymbol{w}_p = (\alpha_p \cdot \boldsymbol{x}_p) \otimes \left(\frac{1}{\alpha_p} \cdot \boldsymbol{w}_p\right)\ \forall \alpha_p \in \mathbb{C}_{\neq 0}$.

The corresponding algorithm for factorizing a Kronecker product is summarized in Algorithm 2. In order to find $\tilde{\boldsymbol{B}}$ from $\boldsymbol{B}$ in Algorithm 2 we remember the fact that $\boldsymbol{Y} \otimes \boldsymbol{Z}$ creates $M \times N$ block matrices, each of these being a scaled version of $\boldsymbol{Z}$. On the other hand,

Algorithm 2 Least-Squares factorization of a Kronecker product

- Consider a matrix $\boldsymbol{B} \in \mathbb{C}^{M \cdot P \times N \cdot Q}$ which is an approximation of the Kronecker product between a matrix $\boldsymbol{Y} \in \mathbb{C}^{M \times N}$ and a matrix $\boldsymbol{Z} \in \mathbb{C}^{P \times Q}$, i.e., $\boldsymbol{B} \approx \boldsymbol{Y} \otimes \boldsymbol{Z}$.
  1. Rearrange the elements of $\boldsymbol{B}$ into a matrix $\tilde{\boldsymbol{B}} \in \mathbb{C}^{P \cdot Q \times M \cdot N}$ such that $\tilde{\boldsymbol{B}} \approx \operatorname{vec}\{\boldsymbol{Z}\} \cdot \operatorname{vec}\{\boldsymbol{Y}\}^{\mathrm{T}}$, cf. Figure 3.1.
  2. Compute the singular value decomposition of $\tilde{\boldsymbol{B}}$ as $\tilde{\boldsymbol{B}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{H}}$. Now the best rank-one approximation of $\tilde{\boldsymbol{B}}$ is given by truncating the SVD, i.e., $\operatorname{vec}\{\hat{\boldsymbol{Y}}\} = \hat{\boldsymbol{y}} = \sqrt{\sigma_1} \cdot \boldsymbol{v}_1^*$ and $\operatorname{vec}\{\hat{\boldsymbol{Z}}\} = \hat{\boldsymbol{z}} = \sqrt{\sigma_1} \cdot \boldsymbol{u}_1$, where $\boldsymbol{u}_1$ and $\boldsymbol{v}_1$ represent the first column vectors of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively, and σ_1 is the largest singular value.
- The final solution is given by $\hat{\boldsymbol{Y}} = \operatorname{unvec}_{M \times N}\{\hat{\boldsymbol{y}}\}$ and $\hat{\boldsymbol{Z}} = \operatorname{unvec}_{P \times Q}\{\hat{\boldsymbol{z}}\}$.

each column of $\tilde{\boldsymbol{B}}$ should be a scaled version of $\operatorname{vec}\{\boldsymbol{Z}\}$, and the scaling coefficients are given by $\operatorname{vec}\{\boldsymbol{Y}\}$. Consequently, we extract the corresponding $M \times N$ blocks of $\boldsymbol{B}$, vectorize them, and collect the resulting vectors as columns of $\tilde{\boldsymbol{B}}$. This process is illustrated for the noise-free case in Figure 3.1. Since the resulting matrix satisfies $\tilde{\boldsymbol{B}} \approx \operatorname{vec}\{\boldsymbol{Z}\} \cdot \operatorname{vec}\{\boldsymbol{Y}\}^{\mathrm{T}}$, we find estimates for $\operatorname{vec}\{\boldsymbol{Z}\}$ and $\operatorname{vec}\{\boldsymbol{Y}\}$ via a rank-one approximation of $\tilde{\boldsymbol{B}}$. Concerning the uniqueness, for the Kronecker factorization there is only one scaling ambiguity, since $\boldsymbol{Y} \otimes \boldsymbol{Z} = (\alpha \cdot \boldsymbol{Y}) \otimes (\frac{1}{\alpha} \cdot \boldsymbol{Z})$, $\forall \alpha \in \mathbb{C}_{\neq 0}$.

Note that both algorithms are easily generalized to the factorization of more than two terms. The factorization of an R-fold Khatri-Rao product $\boldsymbol{F}_1 \diamond \boldsymbol{F}_2 \ldots \diamond \boldsymbol{F}_R$ [RH08a] or an R-fold Kronecker product $\boldsymbol{G}_1 \otimes \boldsymbol{G}_2 \ldots \otimes \boldsymbol{G}_R$ is reduced to the rank-one approximation of an R-dimensional matrix, which is a tensor for $R \geq 2$. As we discuss in Section 4.1, a good rank-one approximation is achieved by computing the truncated Higher-Order SVD. However, this truncated HOSVD is not the LS-optimal solution. For the LS-optimal rank-one approximation of a tensor we need to resort to iterative algorithms, e.g., the HOOI algorithm [dLdMV00b]. However, since the loss in accuracy of the truncated HOSVD compared to the HOOI algorithm is typically very small, we propose to use the truncated HOSVD for R-D Least-Squares Khatri-Rao and Kronecker factorizations.

An extension of the Kronecker factorization was proposed under the name Kronecker-Product SVD in [vL09]. It considers the decomposition of a given matrix into the sum of Kronecker products. This is easily achieved by computing $\tilde{\boldsymbol{B}}$ first, as in Algorithm 2, and then considering all its principle components, instead of only the dominant one. In that respect, we can compute the best approximation of a matrix as the sum of r Kronecker products. Notes on properties and applications of the Kronecker-Product SVD can be found in [vL09].

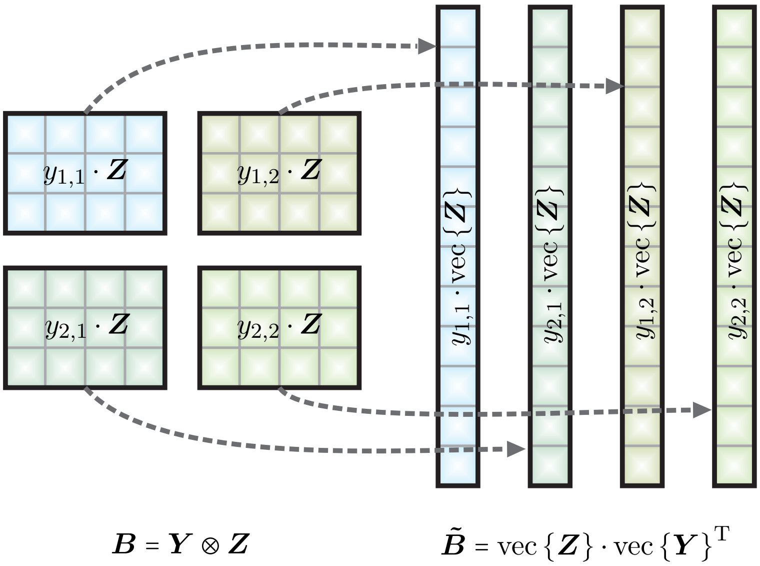


Figure 3.1.: Rearranging a Kronecker product $\boldsymbol{B} = \boldsymbol{Y} \otimes \boldsymbol{Z}$ into a rank one matrix $\tilde{\boldsymbol{B}} = \operatorname{vec}\{\boldsymbol{Z}\} \cdot \operatorname{vec}\{\boldsymbol{Y}\}^{\mathrm{T}}$. The exact factorization (i.e., the noise-free case) is shown for illustration purposes.

4. Multi-linear (tensor) algebraic concepts

In this chapter we review the fundamentals of multilinear (tensor) algebra that are used throughout the thesis in order to store and manipulate multidimensional signals in their native R-dimensional form. We begin by a review of algebraic tools for tensors and state some useful properties. Afterwards, we review the most important tensor decompositions for this thesis, namely, the Higher-Order SVD and the Canonical Polyadic (CP) decomposition and discuss their applications. A major goal of this chapter is to collect results on multilinear algebra that are useful for engineers in a compact and systematic fashion that can serve as a quick reference. In some sense it can be viewed as a "tensor extension" of matrix-based compendia (such as [PP08] or [MN95]). It also extends existing tensor surveys such as [KB09, CLdA09] by some aspects that are not discussed there, e.g., the consistent permutation of the tensor unfoldings or the concatenation of tensors and its algebraic rules.

4.1. Tensor algebra

The term tensor is used in different ways among different scientific communities and therefore, it is used in different meanings throughout different disciplines. Since the focus of our discussion is on the signal processing aspects, we adopt a simple definition by assimilating tensors with their coordinate-system dependent representation as a multi-dimensional array of numbers. In that respect, the term tensor is used synonymously with the term multi-way array and an R-D tensor is defined a collection of numbers referenced by R indices. Consequently, up to $R = 2$ tensors are no different from matrices. However, for $R > 2$ new operations are needed.

Multi-way data analysis has recently become very popular in several scientific fields, ranging from psychometrics and chemometrics over array signal processing and communications to biomedical signal processing, image compression or numerical mathematics (cf. Sections 4.2 as well as [KB09] for a more comprehensive discussion of applications with references). The reason for its popularity is that tensor-based signal processing features some significant fundamental advantages (tensor gains) compared to the matrix-based counterparts. Firstly, we can define multi-linear decompositions that are essentially unique by itself, thus not requiring any additional artificial constraints on the factors to ensure uniqueness (which is required for matrices, cf. Section 3.3). Secondly, identifiability is improved, allowing more components to be present jointly. Thirdly, tensors allow for more efficient denoising since we can exploit the multi-dimensional structure in a more direct way.

In this section we provide a short summary of the notation used throughout this thesis and the main important properties of these operators. Note that our notation is in line with standard notation from other frequently cited papers [dLdMV00b, KB09].

An R-D tensor or R-way array is denoted as $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \ldots M_R}$ with size M_r along mode (or dimension) r for $r = 1, 2, \ldots, R$. As a short-hand notation for the total number of elements we use the convention $M = \prod_{r=1}^{R} M_r$. The r**-mode vectors** of a tensor are the vectors we obtain if the r-th index is varied and all other indices are kept fixed. They represent the generalization of row vectors and column vectors of matrices. We visualize the r-mode vectors of a three-way tensor in Figure 4.1.

The vector space spanned by the r-mode vectors is called r**-mode subspace** or short r**-space** of $\boldsymbol{\mathcal{A}}$. Moreover, an $M_r \times \frac{M}{M_r}$ matrix containing all r-mode vectors as its columns is called r**-mode unfolding** of $\boldsymbol{\mathcal{A}}$. The ordering of the columns in the r-mode unfolding defines how to arrange the remaining $R-1$ indices. Any permutation of these is a valid unfolding and the choice is somewhat arbitrary, as long as it is used consistently. Popular choices include:

- Forward column ordering: Start by varying the first index, then the second, up to the $(r-1)$-th, continue with the $(r+1)$-th all the way up to the R-th index, which is varied last. This is somehow the most natural choice, as it coincides with the standard way to store multi-dimensional data in the memory. It is for instance the column ordering which is returned by MATLAB's `reshape` command[1].

- Reverse column ordering: Like forward but in reverse ordering, starting with the R-th index and proceeding backwards to the first.

- Forward cyclical: Start with the $(r+1)$-th index, then the $(r+2)$-th, all the way to the R-th index, then start from the first and proceed forward up to the $(r-1)$-th index.

- Reverse cyclical: Likewise, the forward cyclical column ordering can be reversed by starting with the $(r-1)$-th index and proceeding backwards, up to the $(r+1)$-th. This convention was proposed by [dLdMV00b] and has since become standard practice in the signal processing community. Therefore, this is the column ordering chosen in this thesis.

An r-mode unfolding in reverse cyclical column ordering is denoted by $[\boldsymbol{\mathcal{A}}]_{(r)}$. We visualize these unfoldings in a 3-D example in Figure 4.1.

[1] In MATLAB, the 1-mode unfolding in forward column ordering is obtained via `reshape(X,M(1),prod(M(2:R)))`, where `X` is an R-way array with `M = size(X)`.

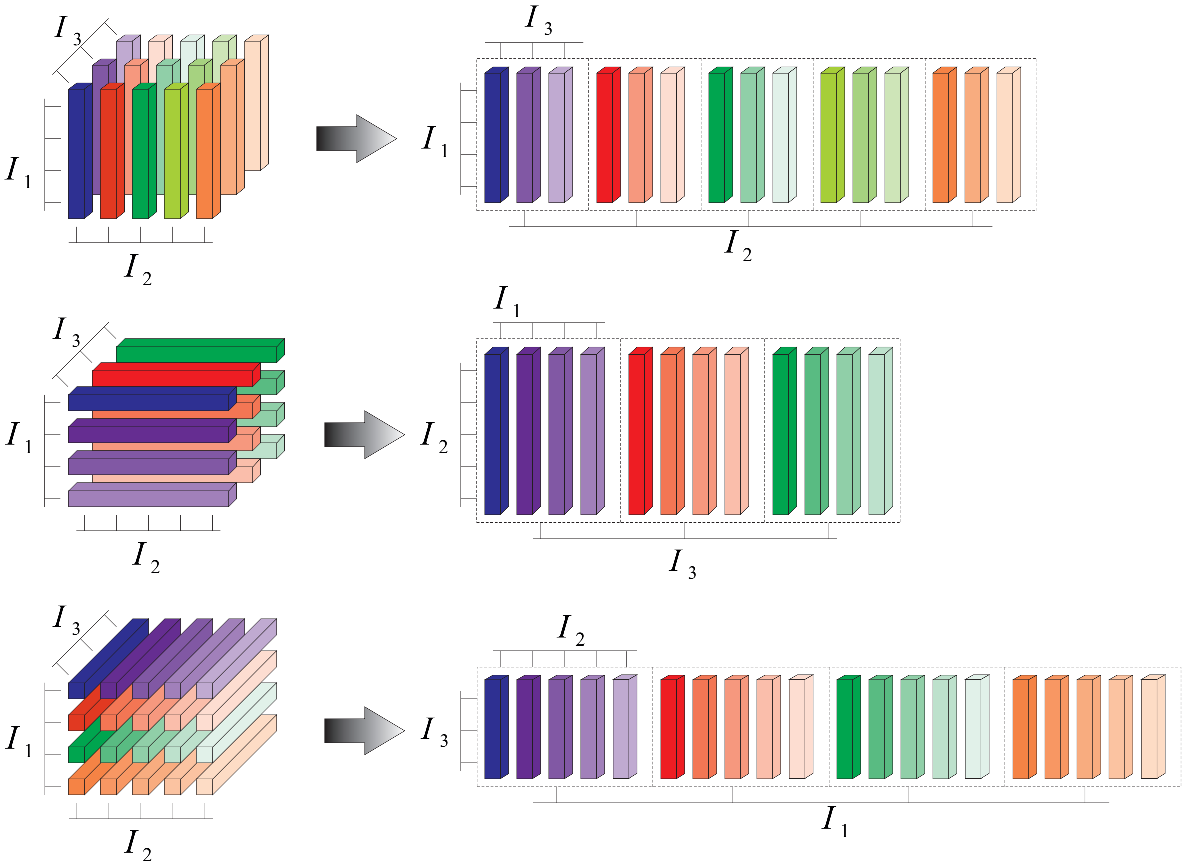


Figure 4.1.: Unfoldings of a $4 \times 5 \times 3$ tensor in reverse cyclical column ordering.

The higher-order extension of the Frobenius norm, also referred to as higher-order norm [dLdMV00b], is symbolized as $\|\boldsymbol{\mathcal{A}}\|_{\mathrm{H}}$. Like the Frobenius norm of a matrix, it is defined as the square-root of the sum of the squared magnitude of all its elements. Therefore, it is obviously equal to the Frobenius norm of an arbitrary unfolding and to the Euclidean norm of an arbitrary vectorization, i.e.,

$$\|\boldsymbol{\mathcal{A}}\|_{\mathrm{H}} = \left\|[\boldsymbol{\mathcal{A}}]_{(r)}\right\|_{\mathrm{F}} = \left\|\mathrm{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\}\right\|_{2}, \; r = 1, 2, \ldots, R, \tag{4.1}$$

for arbitrary R-D tensors $\boldsymbol{\mathcal{A}}$.

The **r-mode product** between a tensor $\boldsymbol{\mathcal{A}}$ and a matrix $\boldsymbol{U}_r$ refers to an r-linear transformation, i.e., a transformation which is linear in the r-th mode. Such a transformation can be expressed via the multiplication of all r-mode vectors by a matrix $\boldsymbol{U}_r$ from the left. We denote it as $\boldsymbol{\mathcal{A}} \times_r \boldsymbol{U}_r$. From its definition it satisfies the following important property

$$[\boldsymbol{\mathcal{A}} \times_r \boldsymbol{U}_r]_{(r)} = \boldsymbol{U}_r \cdot [\boldsymbol{\mathcal{A}}]_{(r)}\,. \tag{4.2}$$

For repeated r-mode products, we introduce the following short-hand notation

$$\boldsymbol{\mathcal{A}} \bigtimes_{r=1}^{R}{}_{r} \boldsymbol{U}_r = \boldsymbol{\mathcal{A}} \times_1 \boldsymbol{U}_1 \ldots \times_R \boldsymbol{U}_R. \tag{4.3}$$

The rank of the r-mode unfolding (i.e., the dimension of the r-space) of a tensor $\boldsymbol{\mathcal{A}}$ is called **r-rank** of $\boldsymbol{\mathcal{A}}$. Note that, unlike for matrices, for tensors all r-ranks (also called multilinear ranks) can be different.

The **outer product** of an R_1-dimensional tensor $\boldsymbol{\mathcal{A}}$ and an R_2-dimensional tensor $\boldsymbol{\mathcal{B}}$ is symbolized by $\boldsymbol{\mathcal{A}} \circ \boldsymbol{\mathcal{B}}$ which yields an $R_1 + R_2$-dimensional tensor containing all pair-wise products of all elements in $\boldsymbol{\mathcal{A}}$ and all elements in $\boldsymbol{\mathcal{B}}$. It is therefore compatible with the outer product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ ($R_1 = R_2 = 1$), which yields a matrix, i.e., $\boldsymbol{a} \circ \boldsymbol{b} = \boldsymbol{a} \cdot \boldsymbol{b}^{\mathrm{T}}$. The outer product is strongly linked to the Kronecker product since the matrix $\boldsymbol{A} \otimes \boldsymbol{B}$ and the 4-D tensor $\boldsymbol{A} \circ \boldsymbol{B}$ contain the same elements. Outer products can also be used in a constructive manner: much like a matrix of rank r can be constructed as a sum of r rank-one matrices, a tensor of rank r is obtained by summing r rank-one tensors. More specifically, we define the **tensor rank** or simply **rank** of $\boldsymbol{\mathcal{A}}$ as the smallest possible integer number r such that $\boldsymbol{\mathcal{A}}$ can be written as a sum of r rank-one tensors. An R-D tensor is rank-one if it can be written as the outer product of R non-zero vectors.

The tensor rank is not directly related to the multilinear ranks, it only provides an upper bound since we have $\mathrm{rank}(\boldsymbol{\mathcal{X}}) \geq r\text{-}\mathrm{rank}(\boldsymbol{\mathcal{X}}), \forall r = 1, 2, \ldots, R$ [KB09]. Determining the rank of a given tensor has been shown to be an NP-hard problem [Has90]. Another interesting property is that the rank of a given tensor can be different over $\mathbb{R}$ and $\mathbb{C}$, a famous example for this is discussed in [Kru89, Ten00]. Also, if we draw the elements of a tensor randomly from a continuous distribution, we do not get only one ("generic") rank with probability one but we may get several ("typical") ranks with non-zero probability. An example is a $2 \times 2 \times 2$ tensor which has generic rank 2 over $\mathbb{C}$ but the typical ranks 2 and 3 over $\mathbb{R}$ (cf. [Kru89] for examples). Finding a low-(tensor-)rank approximation of a given tensor is in general an ill-posed problem since the best rank-k approximation may not even exist [Paa00, dSL08]. Still it is of great practical significance to find good low-rank approximations of tensors, see Chapter 6 for a discussion on approximate CP decompositions. On the other hand, multilinear rank reduction is much better understood, see [dLdMV00b] for details on optimal and suboptimal r-rank reduction techniques.

The **concatenation** of two tensors along the r-th mode [HRD08] is symbolized via $[\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{B}}]$. The r-mode vectors of the resulting tensor are given by the r-mode vectors of $\boldsymbol{\mathcal{A}}$ stacked on top of the r-mode vectors of $\boldsymbol{\mathcal{B}}$. In other words, we have

$$[\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{B}}]_{(r)} = \begin{bmatrix} [\boldsymbol{\mathcal{A}}]_{(r)} \\ [\boldsymbol{\mathcal{B}}]_{(r)} \end{bmatrix}. \tag{4.4}$$

Note that r-mode products, r-mode unfoldings, and r-mode concatenations satisfy the following important properties

$$[\boldsymbol{\mathcal{A}} \times_1 \boldsymbol{U}_1 \ldots \times_R \boldsymbol{U}_R]_{(r)} = \boldsymbol{U}_r \cdot [\boldsymbol{\mathcal{A}}]_{(r)} \cdot (\boldsymbol{U}_{r+1} \otimes \ldots \otimes \boldsymbol{U}_R \otimes \boldsymbol{U}_1 \otimes \ldots \otimes \boldsymbol{U}_{r-1})^{\mathrm{T}} \tag{4.5}$$

$$\boldsymbol{\mathcal{A}} \times_r \boldsymbol{U}_r \times_p \boldsymbol{U}_p = \boldsymbol{\mathcal{A}} \times_p \boldsymbol{U}_p \times_r \boldsymbol{U}_r \quad \text{where } r \neq p \tag{4.6}$$

$$\boldsymbol{\mathcal{A}} \times_r \boldsymbol{U}_r \times_r \boldsymbol{V}_r = \boldsymbol{\mathcal{A}} \times_r (\boldsymbol{V}_r \cdot \boldsymbol{U}_r) \tag{4.7}$$

$$[\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{B}}] \times_p \boldsymbol{U}_p = [\boldsymbol{\mathcal{A}} \times_p \boldsymbol{U}_p \sqcup_r \boldsymbol{\mathcal{B}} \times_p \boldsymbol{U}_p] \quad \text{where } r \neq p \tag{4.8}$$

$$[\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{B}}] \times_r \begin{bmatrix} \boldsymbol{U}_r & \boldsymbol{W}_r \end{bmatrix} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{U}_r + \boldsymbol{\mathcal{B}} \times_r \boldsymbol{W}_r, \tag{4.9}$$

$$\boldsymbol{\mathcal{A}} \times_r \begin{bmatrix} \boldsymbol{X}_r \\ \boldsymbol{Y}_r \end{bmatrix} = [(\boldsymbol{\mathcal{A}} \times_r \boldsymbol{X}_r) \sqcup_r (\boldsymbol{\mathcal{A}} \times_r \boldsymbol{Y}_r)], \tag{4.10}$$

where $r, p \in \{1, 2, \ldots, R\}$ and the dimensions of the tensors and matrices are $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \in \mathbb{C}^{M_1 \times \ldots \times M_R}$, $\boldsymbol{U}_r, \boldsymbol{W}_r \in \mathbb{C}^{N_r \times M_r}$, $\boldsymbol{V}_r \in \mathbb{C}^{P_r \times N_r}$, $\boldsymbol{X}_r \in \mathbb{C}^{N_r \times M_r}$, and $\boldsymbol{Y}_r \in \mathbb{C}^{Q_r \times M_r}$. The order of the matrices in the Kronecker product in (4.5) is a direct consequence of the reverse cyclical column ordering that was chosen for the r-mode unfolding. Property (4.5) is shown in [dLdMV00b] and properties (4.6), (4.7) (also shown in [dLdMV00b]) are a direct consequence. Moreover, (4.8), (4.9), and (4.10) follow straightforwardly by combining (4.4) and (4.5).

As argued in the previous section, we often need to reformulate linear forms into a standard "canonical" linear vector form. Since r-mode products define linear forms in particular modes, they represent a special case. As shown in (4.5), the r-mode multiplication is easily transformed into a matrix multiplication via the r-mode unfolding. However, the remaining difficulty in the tensor case resides in the fact that the vectorized versions of different unfoldings contain the same elements in a different order. Consequently, we need to define permutation matrices which restore the correct ordering. To this end we define the following set of matrices: For every tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R}$, there exists a unique set of permutation matrices $\boldsymbol{P}^{(r)}_{M_1,M_2,\ldots,M_R}$ such that

$$\operatorname{vec}\{\boldsymbol{\mathcal{A}}\} = \boldsymbol{P}^{(r)}_{M_1,M_2,\ldots,M_R} \cdot \operatorname{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\} \tag{4.11}$$

for $r = 1, 2, \ldots, R$. Since $\boldsymbol{P}^{(r)}_{M_1,M_2,\ldots,M_R}$ are permutation matrices, they satisfy $\boldsymbol{P}^{(r)^{-1}}_{M_1,M_2,\ldots,M_R} = \boldsymbol{P}^{(r)^{\mathrm{T}}}_{M_1,M_2,\ldots,M_R}$. The vec-operator applied to a tensor $\operatorname{vec}\{\boldsymbol{\mathcal{A}}\}$ is defined as the natural extension of the vec-operator applied to matrices (cf. (3.4)), i.e., all elements are stacked into a column vector $\in \mathbb{C}^M$ and we stack the indices in ascending order starting with the first, then the second, and so on up to the R-th index[2]. For instance, for a 3-D tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ we can write

$$\operatorname{vec}\{[\boldsymbol{A}_1 \sqcup_3 \boldsymbol{A}_2 \sqcup_3 \ldots \sqcup_3 \boldsymbol{A}_{M_3}]\} = \begin{bmatrix}\operatorname{vec}\{\boldsymbol{A}_1\}^{\mathrm{T}} & \operatorname{vec}\{\boldsymbol{A}_2\}^{\mathrm{T}} & \ldots & \operatorname{vec}\{\boldsymbol{A}_{M_3}\}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}, \tag{4.12}$$

where $\boldsymbol{A}_m \in \mathbb{C}^{M_1 \times M_2}$ for $m = 1, 2, \ldots, M_3$.

A direct manner of computing the permutation matrix defined in (4.11) via regular and "permuted column" Kronecker products is outlined in Appendix B.3.

In a sense, we can view the permutation matrices as a direct extension of the commutation matrices $\boldsymbol{K}_{M,N}$ from Section 3.1.2. In fact, $\boldsymbol{K}_{M,N} = \boldsymbol{P}^{(2)}_{N,M}$ since for a matrix $\boldsymbol{A}$ we have $[\boldsymbol{A}]_{(2)} = \boldsymbol{A}^{\mathrm{T}}$. As shown in Appendix B.7, the permutation matrices defined in (4.11) can reverse the order and perform cyclic shifts in an R-fold Kronecker product.

The following theorem provides useful identities for vectorized unfoldings in the 3-D case, assuming reverse cyclical column ordering.

Theorem 4.1.1. [RH10c] *For a 3-D tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ the following vectorized versions are identical*

$$\operatorname{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(1)}\right\} = \operatorname{vec}\left\{[\boldsymbol{\mathcal{A}}]^{\mathrm{T}}_{(2)}\right\} \tag{4.13}$$

[2]This is the natural order of indices when storing multidimensional variables that is also used by MATLAB. Therefore, this definition is "MATLAB-compatible", i.e., $\operatorname{vec}\{\boldsymbol{\mathcal{A}}\}$ is the same as MATLAB's `A(:)`.

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(2)}\right\} = \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(3)}^{\mathrm{T}}\right\} \tag{4.14}$$

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(3)}\right\} = \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(1)}^{\mathrm{T}}\right\}. \tag{4.15}$$

Proof: cf. Appendix B.8.

This theorem was shown in a different manner in [RH10c] where it appears as Lemma 4. However, the proof in [RH10c] relies on the existence of a unique CP decomposition [Kru77, Kru89] of $\boldsymbol{\mathcal{A}}$ which is not needed for the proof shown in Appendix B.8. Note that a corollary that follows from this theorem is that some unfoldings can be directly permuted into a consistent order only with the help of the commutation matrices $\boldsymbol{K}_{M,N}$. Using (3.34) in (4.13), (4.14), and (4.15), we directly obtain

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(1)}\right\} = \boldsymbol{K}_{M_2,M_1 \cdot M_3} \cdot \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(2)}\right\} \tag{4.16}$$

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(2)}\right\} = \boldsymbol{K}_{M_3,M_1 \cdot M_2} \cdot \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(3)}\right\} \tag{4.17}$$

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(3)}\right\} = \boldsymbol{K}_{M_1,M_2 \cdot M_3} \cdot \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(1)}\right\}. \tag{4.18}$$

In fact, an R-D extension of these identities is possible, as we can show that

$$\text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\} = \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r+1)}^{\mathrm{T}}\right\} = \boldsymbol{K}_{M_{r+1},\frac{M}{M_{r+1}}} \cdot \text{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r+1)}\right\}, \; r = 1,2,\ldots,R-1 \tag{4.19}$$

for any $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R}$. Another useful property related to the vec-operator and n-mode products is the following

$$\text{vec}\left\{\boldsymbol{\mathcal{A}} \times_1 \boldsymbol{U}_1 \ldots \times_R \boldsymbol{U}_R\right\} = (\boldsymbol{U}_R \otimes \ldots \otimes \boldsymbol{U}_1) \cdot \text{vec}\left\{\boldsymbol{\mathcal{A}}\right\} \tag{4.20}$$

since it allows to solve linear forms for the tensor as well. Note that (4.20) can be seen as the R-D extension of (3.7). Also, to solve the repeated n-mode product in (4.20) for one of the matrices $\boldsymbol{U}_n$, we can apply (4.5).

As an extension of the identity matrix $\boldsymbol{I}_N$ of size $N \times N$ we define an R-D identity tensor $\boldsymbol{\mathcal{I}}_{R,N}$ of size $N \times N \ldots \times N$, which is equal to one of all R indices agree ($i_1 = i_2 = \ldots = i_R$) and zero otherwise. Note that there is a strong link between the 3-D identity tensor and the reduction matrices introduced in (3.9) given by the following identity

$$[\boldsymbol{\mathcal{I}}_{3,N}]_{(1)} = [\boldsymbol{\mathcal{I}}_{3,N}]_{(2)} = [\boldsymbol{\mathcal{I}}_{3,N}]_{(3)} = (\boldsymbol{I}_N \diamond \boldsymbol{I}_N)^{\mathrm{T}} = \boldsymbol{\Xi}_N^{\mathrm{T}}, \tag{4.21}$$

i.e., all unfoldings of the $N \times N \times N$ identity tensor are equal to the transpose of the $N^2 \times N$

reduction matrix. A corollary which follows from this is that

$$[\mathcal{I}_{3,N} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}]_{(1)} = \boldsymbol{A} \cdot (\boldsymbol{B} \diamond \boldsymbol{C})^{\mathrm{T}} \tag{4.22}$$

$$[\mathcal{I}_{3,N} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}]_{(2)} = \boldsymbol{B} \cdot (\boldsymbol{C} \diamond \boldsymbol{A})^{\mathrm{T}} \tag{4.23}$$

$$[\mathcal{I}_{3,N} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}]_{(3)} = \boldsymbol{C} \cdot (\boldsymbol{A} \diamond \boldsymbol{B})^{\mathrm{T}}. \tag{4.24}$$

These identities follow by combining (4.5) with (3.11) and (4.21). They are frequently used in the context of the Canonical Polyadic decomposition [KB09], see also Section 4.2.

Finally, note that this is easily extended to the R-D case: all unfoldings of $\mathcal{I}_{R,N}$ are equal to the Khatri-Rao product between $R-1$ identity matrices of size $N \times N$ which reduces an $(R-1)$-fold Kronecker product to an $(R-1)$-fold Khatri-Rao product. Therefore, we can write

$$\left[\mathcal{I}_{R,N} \bigtimes_{r=1}^{R}{}_r \boldsymbol{F}_r\right]_{(p)} = \boldsymbol{F}_p \cdot (\boldsymbol{F}_{p+1} \diamond \ldots \diamond \boldsymbol{F}_R \diamond \boldsymbol{F}_1 \diamond \ldots \diamond \boldsymbol{F}_{p-1})^{\mathrm{T}}. \tag{4.25}$$

4.2. Tensor decompositions

In this section we present a brief overview of the tensor decompositions relevant to this thesis, namely, the Higher-Order Singular Value Decomposition (HOSVD) and the Canonical Polyadic (CP) decomposition. Both can be seen as a possible generalization of the SVD of matrices to the tensor case [KB09].

4.2.1. Tucker3 (Higher-Order SVD) decomposition

The Higher-Order SVD can be viewed as a special case of the Tucker3 decomposition, which has been known since [Tuc66]. An R-D extension was proposed in [KNW86]. Tucker3 is essentially a 3-mode Principle Component Analysis (PCA) and can be expressed as

$$\mathcal{X} = \mathcal{G} \times_1 \boldsymbol{V}_1 \times_2 \boldsymbol{V}_2 \times_3 \boldsymbol{V}_3, \tag{4.26}$$

where $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$, the matrices $\boldsymbol{V}_r \in \mathbb{C}^{M_r \times p_r}$ are the factor matrices of the decomposition, and the tensor $\mathcal{G} \in \mathbb{C}^{p_1 \times p_2 \times p_3}$ is referred to as the core tensor. Here, p_r refers to the r-rank of $\mathcal{X}$. Obviously, the decomposition is exact if the matrices $\boldsymbol{V}_r$ are chosen to contain a basis for the r-space of $\mathcal{X}$. The special case where $\boldsymbol{V}_3 = \boldsymbol{I}_{M_3}$ is also referred to as Tucker2. Moreover, the case where $\boldsymbol{V}_3 = \boldsymbol{I}_{M_3}$ and $\boldsymbol{V}_2 = \boldsymbol{I}_{M_2}$ is called Tucker1 decomposition, see Section 4.3 for details.

The factor matrices $\boldsymbol{V}_r$ are typically chosen to be unitary but this is not required. Therefore, the decomposition is not unique as every factor matrix can be post-multiplied with a square invertible matrix $\boldsymbol{T}_r$ if the inverse matrices $\boldsymbol{T}_r^{-1}$ are absorbed into the new core tensor. These degrees of freedom create the ability to find specific sets of factor matrices which give the core tensor special properties. While it is in general not possible to create a diagonal core tensor (as in the SVD), one could at least try to make as many elements as possible zero, (e.g., [AH99]) or to maximize its diagonal dominance (e.g., [MvL08]).

Due to its simplicity, a more popular approach is given by the HOSVD introduced in [dLdMV00a]. The authors propose to choose $\boldsymbol{V}_r$ as the p_r dominant left singular vectors of the r-mode unfolding $[\boldsymbol{\mathcal{X}}]_{(r)}$. To distinguish this specific Tucker3 decomposition from the general case shown in (4.26) we use the notation $\boldsymbol{U}_r$ for the factor matrices and $\boldsymbol{\mathcal{S}}$ for the core tensor of the HOSVD. In the general R-D case, the HOSVD can be expressed as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \dots \times_R \boldsymbol{U}_R = \boldsymbol{\mathcal{S}} \bigtimes_{r=1}^{R}{}_r \boldsymbol{U}_r \tag{4.27}$$

where $\boldsymbol{U}_r \in \mathbb{C}^{M_r \times p_r}$ and $\boldsymbol{\mathcal{S}} \in \mathbb{C}^{p_1 \times \dots \times p_R}$. The core-tensor $\boldsymbol{\mathcal{S}}$ satisfies the so-called "all-orthogonality" condition [dLdMV00a], which can be written compactly as

$$[\boldsymbol{\mathcal{S}}]_{(r)} \cdot [\boldsymbol{\mathcal{S}}]_{(r)}^{\mathrm{H}} = \operatorname{diag}\left\{ \left[\sigma_1^{(r)^2}, \dots, \sigma_{p_r}^{(r)^2} \right] \right\}, \tag{4.28}$$

i.e., all its unfoldings are row-orthogonal matrices. Moreover, the Euclidean norms of these rows are the r-mode singular values $\sigma_n^{(r)}$. Like for the SVD, they appear ordered by magnitude, i.e., $\sigma_1^{(r)} \geq \sigma_2^{(r)} \geq \dots \geq \sigma_{p_r}^{(r)}$, for $r = 1, 2, \dots, R$. Note that in the tensor case there exist R sets of r-mode singular values which are in general different. As an example, we depict the HOSVD of a 3-way tensor $\boldsymbol{\mathcal{X}}$ of size $6 \times 3 \times 5$ in Figure 4.2.

Another important question is whether the concept of low-rank approximations also carries over from the SVD to the HOSVD. For the SVD, the Eckart-Young theorem shows that the truncated SVD provides the best low-rank approximation in the Frobenius norm sense. Such a theorem does unfortunately not exist for the HOSVD. While the truncated HOSVD does provide a good low-rank approximation, it is in general not optimal. However, it can be used as a starting point for an Alternating Least Squares (ALS)-like algorithm which computes the best approximation. This concept is discussed in [dLdMV00b] and the corresponding algorithm is called Higher-Order Orthogonal Iterations (HOOI).

Note that we use the concept of low-rank approximations to suppress the additive noise of a low-rank desired signal component from a noisy observation (cf. Section 10.2). For this application, the truncated HOSVD is asymptotically optimal in the high Signal-To-Noise Ratio

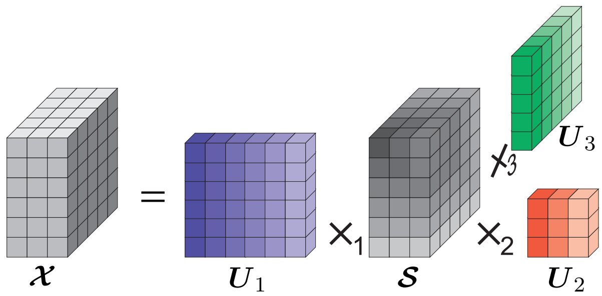


Figure 4.2.: HOSVD of a 3-way tensor $\boldsymbol{\mathcal{X}}$ of size $6 \times 3 \times 5$.

(SNR) regime [dLdMV00b]. An improvement from the HOOI algorithm is only visible for low SNRs, however, even there it is not very significant[3] (see, for instance, the simulation result shown in Figure 7.10). For this reason, we only consider the truncated HOSVD for subspace estimation unless stated otherwise. To make a clear distinction, truncated quantities are denoted by a superscript $(\cdot)^{[\mathrm{s}]}$, i.e., the truncated HOSVD of a tensor $\boldsymbol{\mathcal{X}}$ is expressed as

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}]} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \bigtimes_{r=1}^{R}{}_r \boldsymbol{U}_r^{[\mathrm{s}]}. \tag{4.29}$$

The HOSVD has numerous applications in various areas of signal processing. A famous area is face recognition, which was pioneered by the introduction of *TensorFaces* in [VT02a]. TensorFaces represents a framework for the multilinear analysis of facial expressions from various subjects. Capturing multiple images of each facial expression for each subject gives rise to three tensor dimensions (image, expression, subject). More dimensions arise if various lighting conditions or camera angles are taken into account. The joint processing of all dimensions allows to remove unwanted influences such as lighting conditions and camera angles and to generalize over expressions (for face recognition) or over subjects (for facial expression recognition). As shown in [VT02b], compared to a matrix-based (PCA) analysis of the data, the face recognition accuracy is significantly improved and artifacts can be removed more efficiently. Building on those results, further applications in this area have been identified, e.g., facial expression transfer [VBPP05] or facial expression recognition and compression [WA03]. Other applications in image and video processing include texture rendering [VT04], watermarking in MPEG videos [AHB07], hyperspectral image denoising [LB08], or motion tracking using the tensor null space [CSK09]. Another area where the HOSVD has been successfully ap-

[3] For a discussion on local minima of the best multilinear low-rank approximation of tensors, the reader is referred to [IAvHdL10].

plied is data mining with applications to network modeling [STF06], social network analysis [ACKY05], or pattern recognition [SE07]. These applications have in common that apply the HOSVD to perform rank reduction in individual modes and then use parts of the truncated decomposition, i.e., the core tensor, individual loading matrices, or combinations thereof.

As we demonstrate in Section 10.2, a different application of the truncated HOSVD is signal subspace estimation for multidimensional signals. This signal subspace can be estimated via matrices only if a stacking operation is applied to the data such that each column of the measurement matrix represents a stacked version of one snapshot of the multidimensional signal. However, such a representation does not account for the structure inherent in the data. Therefore, we can improve the subspace estimation accuracy by performing a "structured" subspace estimation based on the HOSVD. The resulting subspace estimate can be combined with any multi-dimensional subspace-based parameter estimation scheme. We demonstrate the tensor gain from this enhanced subspace estimate for the family of ESPRIT-type algorithms numerically in Chapter 11 and analytically in Chapter 12. We discuss two application examples in Section 9.1 in more detail: 2-D direction of arrival (DOA) estimation in Example 9.2.1 and geometry-based channel modeling based on measured MIMO channels in Example 9.2.2. An additional application area is MIMO Radar since it has been shown in [NS10] that single-pulse and multi-pulse bistatic Radar configurations give rise to a 2-D and a 3-D harmonic retrieval problem, respectively. We have also shown that based on the HOSVD, a tensor-based framework for the prediction of frequency-selective time-variant MIMO channels can be found [MGH08].

All the applications of the HOSVD mentioned here have in common that they obtain a tensor gain compared to matrix-based approaches since the multilinear rank reduction obtained from the truncated HOSVD filters out unwanted components in a "structured" manner which yields a more accurate representation of the data.

4.2.2. CANDECOMP/PARAFAC (CP) decomposition

The SVD can also be viewed as a decomposition of a given rank-d matrix into a sum of d rank-one matrices constructed from the outer product of the n-th left and right singular vector and scaled by the n-th singular value for $n = 1, 2, \ldots, d$. In the same manner we can decompose a rank-d tensor into a sum of rank-one tensors which are obtained from the outer product of R non-zero vectors. This gives rise to a decomposition of the form

$$\boldsymbol{\mathcal{X}} = \sum_{n=1}^{d} \boldsymbol{f}_n^{(1)} \circ \boldsymbol{f}_n^{(2)} \circ \ldots \circ \boldsymbol{f}_n^{(R)}, \tag{4.30}$$

where $\boldsymbol{f}_n^{(r)}$ is referred to as the loading vector of the n-th component in the r-th mode. Equation (4.30) describes a very fundamental tensor decomposition which has been known for many decades under different names, e.g., polyadic form [Hit27], parallel proportional profiles [Cat44], Canonical Decomposition (CANDECOMP) [CC70], or Parallel Factor (PARAFAC) analysis [Har70]. We refer to it as CP decomposition which is either seen as an abbreviation for CANDECOMP/PARAFAC [KB09] or Canonical Polyadic decomposition. Note that since the sum in (4.30) is commutative and the products are associative, the loading vectors $\boldsymbol{f}_n^{(r)}$ are determined only up to an inherent permutation ambiguity (between different n) and a scaling ambiguity (between different r). Except for these indeterminacies, the decomposition is unique only due to its structure. This is a fundamental advantage over the bilinear (matrix) decompositions, which require additional assumptions to obtain a comparable uniqueness (cf. Section 3.3). Such assumptions are often unwanted since they may not have any physical meaning.

The CP decomposition in (4.30) may be formulated in a number of different ways. For instance, for $R = 3$ we often find the following equivalent relations

$$[\boldsymbol{\mathcal{X}}]_{(1)} = \boldsymbol{F}^{(1)} \cdot \left(\boldsymbol{F}^{(2)} \diamond \boldsymbol{F}^{(3)}\right)^{\mathrm{T}} \tag{4.31}$$

$$[\boldsymbol{\mathcal{X}}]_{(:,:,k)} = \boldsymbol{F}^{(1)} \cdot \operatorname{diag}\left\{[\boldsymbol{F}^{(3)}]_{(k,:)}\right\} \cdot \boldsymbol{F}^{(2)^{\mathrm{T}}} \tag{4.32}$$

$$[\boldsymbol{\mathcal{X}}]_{(i,j,k)} = \sum_{n=1}^{d} f_{i,n}^{(1)} \cdot f_{j,n}^{(2)} \cdot f_{k,n}^{(3)} \tag{4.33}$$

where $\boldsymbol{F}^{(r)} = [\boldsymbol{f}_1^{(r)}, \ldots, \boldsymbol{f}_d^{(r)}] \in \mathbb{C}^{M_r \times d}$ is the loading matrix in the r-th mode and $f_{i,n}^{(r)}$ represents the i-th element of the vector $\boldsymbol{f}_n^{(r)}$. Moreover, $[\boldsymbol{\mathcal{X}}]_{(:,:,k)}$ represents the k-th three-mode slice, i.e., the matrix of all elements where the third index in the tensor is equal to k. Also, $\operatorname{diag}\left\{[\boldsymbol{F}^{(3)}]_{(k,:)}\right\}$ is a diagonal matrix containing the elements of the k-th row of $\boldsymbol{F}^{(3)}$ on its main diagonal.

A more compact representation is obtained by using the identity tensor $\boldsymbol{\mathcal{I}}_{R,N}$ introduced in Section 4.1 which yields the following representation of the CP decomposition in the general R-D case

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{I}}_{R,d} \times_1 \boldsymbol{F}^{(1)} \ldots \times_R \boldsymbol{F}^{(R)} = \boldsymbol{\mathcal{I}}_{R,d} \bigtimes_{r=1}^{R} {}_r \boldsymbol{F}^{(r)}. \tag{4.34}$$

One of the advantages of (4.34) is that its structure is very similar to the Tucker3 decomposition shown in (4.26). This shows that for the CP, the core tensor has been diagonalized. However, this is only possible as long as the decomposition is exact, i.e., the tensor has rank d (there is

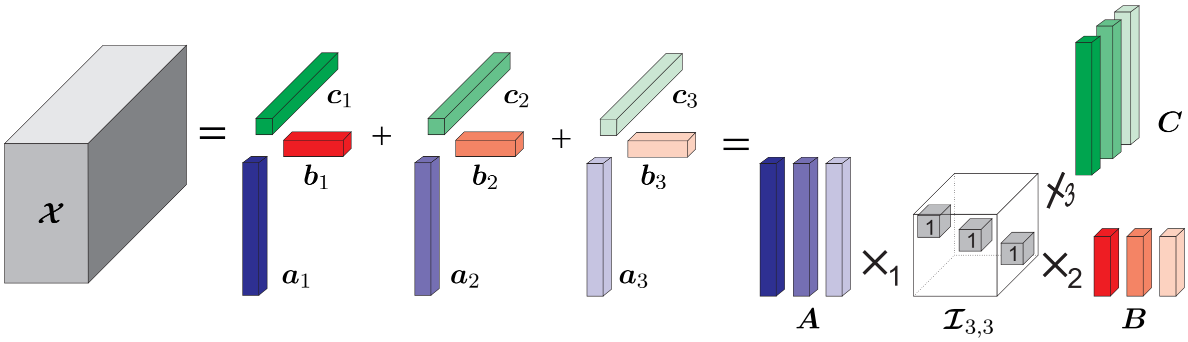


Figure 4.3.: CP decomposition of a 3-way tensor $\mathcal{X}$ of rank $d = 3$ in outer product notation and n-mode product notation.

no additive noise).

A big difference between the HOSVD and the CP is that while the HOSVD is easily computed via SVDs of the unfoldings of the tensor, such a direct computation is in general not possible to find the CP. Instead, we need to resort to iterative techniques, such as Alternating Least Squares (ALS)-like procedures [CC70, Har70, KdL80]. However, these may require many iterations to converge and are not guaranteed to find the global optimum. This is the main motivation for deriving semi-algebraic solutions for approximate CP decompositions, discussed in Part II of this thesis.

For illustration purposes, we visualize the structure of the CP decomposition for a 3-way tensor $\mathcal{X}$ of rank 3 in Figure 4.3. The left-hand side shows the outer product notation $\mathcal{X} = \sum_{n=1}^{3} \boldsymbol{a}_n \circ \boldsymbol{b}_n \circ \boldsymbol{c}_n$ (cf. (4.30)). On the other hand, the right-hand side depicts the n-mode product notation $\mathcal{X} = \mathcal{I}_{3,3} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}$ (cf. (4.34)), where $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \boldsymbol{a}_3 \end{bmatrix}$, $\boldsymbol{B} = \begin{bmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \boldsymbol{b}_3 \end{bmatrix}$, and $\boldsymbol{C} = \begin{bmatrix} \boldsymbol{c}_1 & \boldsymbol{c}_2 & \boldsymbol{c}_3 \end{bmatrix}$.

Note that numerous applications exist, where the signal of interest may be modeled as a linear superposition of rank-one tensors. These range from psychometrics [CC70, Har70] and chemometrics [AB03] over array signal processing [SBG00] and communications [SGB00] to biomedical signal processing [Möc88, MHH⁺06, dVVdL⁺07], image compression [SL01], data mining [KB06], or numerical mathematics [HKT05]. We have successfully applied the CP decomposition (via the SECSI framework introduced in Part II of this thesis), for biomedial signal processing applications [WRH⁺09, BCA⁺12], blind-channel estimation in Space-Time-Coded MIMO systems [RSS⁺11], model order estimation [DRWH10], as well as parameter estimation for multidimensional signals with colored noise [DSR⁺10].

4.3. Other decompositions

Even though the HOSVD and the CP are the only decompositions discussed in this thesis, we would like to provide a brief outlook to different tensor decompositions as well. There have been many specializations as well as generalizations of these ideas which have become viable tools in different signal processing applications.

Firstly, the special case of Tucker3 where the third factor matrix is equal to an identity matrix is referred to as Tucker2 decomposition [Tuc66]. Likewise, if the second and the third factor are identities we call the decomposition Tucker1 [Tuc66] (strictly speaking, this is not a tensor decomposition any more since Tucker1 is exactly the same as a purely matrix-based PCA).

A combination of CP and Tucker is given by block tensor decompositions. Examples include the $(L, L, 1)$, the (L, M, N), and the $(L, M, \cdot)$ decomposition with applications in blind multiuser detection for DS-CDMA systems [LB08], which are discussed in [dLdB08].

A CP decomposition which explicitly models rank deficiencies in the loading matrices has become known as the PARALIND decomposition [BHS05, BHSL09]. Moreover, the Tensor-Train (TT) decomposition [Ose11] represents a numerically stable tensor decomposition (based only on SVDs) with only slightly more parameters than the CP, which is especially attractive for very high-dimensional data.

Another extension of the CP decomposition is known as PARAFAC2 [Har72]. This decomposition allows one of the loading matrices to vary over slices, i.e., instead of modeling the k-th three-mode slice as $\boldsymbol{A} \cdot \operatorname{diag}\left\{[\boldsymbol{C}]_{k,:}\right\} \cdot \boldsymbol{B}$ as for the CP, we have $\boldsymbol{A} \cdot \operatorname{diag}\left\{[\boldsymbol{C}]_{k,:}\right\} \cdot \boldsymbol{B}_k$ so that $\boldsymbol{B}$ does not need to "remain constant" over the 3-mode slices of the tensor. This decomposition is only unique under additional constraints, e.g., $\boldsymbol{B}_k^{\mathrm{H}} \cdot \boldsymbol{B}_k$ should be invariant over k. It has been shown that PARAFAC2 can be used to handle signal delays over space when applied to space-time-frequency tensors [WJG+10b].

5. Summary of the advanced algebraic concepts

In the first part of this thesis we have provided a set of useful tools for algebraic manipulation of certain expressions. In particular, we have investigated various types of linear forms and quadratic forms and their reformulation into simple "canonical" versions. These canonical versions often allow for direct solutions, e.g., a canonical linear vector form can be solved for the unknown parameters in a Least-Squares optimal manner via a pseudo-inverse. Likewise, quadratic forms and ratios of quadratic forms are strongly linked to eigenvectors and generalized eigenvectors of Hermitian matrices, respectively. We have also shown that Kronecker and Khatri-Rao products can be factorized in a Least-Squares optimal manner. The solutions were outlined for the case where one factor is known as well as for the case where both factors are unknown.

Subsequently, we have reviewed elementary concepts of multi-linear (tensor) algebra, which enable storage and manipulation of multi-dimensional signals in their natural multi-dimensional form. After providing definition and important properties of elementary operators such as unfoldings, n-mode products, or n-mode concatenation, we have outlined two of the most frequently used tensor decompositions, the Tucker3 decomposition in form of the Higher-Order SVD and the CANDECOMP/PARAFAC (CP) decomposition.

5.1. Relevance to later parts of the thesis

In the following parts of this thesis we use the results presented in this part extensively. In particular, the vectorization of linear forms is employed in Part III for deriving the first-order perturbation analysis for the estimation error in tensor-based parameter estimation schemes. This first-order expansion requires vectorizing different types of linear forms, including matrix product linear forms and Kronecker products (cf. Theorem 12.4.1). We also use it in Part IV for rewriting the transmission equations in a two-way relaying system, e.g., in the derivation of the ANOMAX transmit strategy (cf. Section 16.3).

The rearrangement of quadratic forms, such as (3.39) is extensively used in Part IV, e.g., for the derivation of the RAGES (Rate-Maximization via Generalized Eigenvectors for Single Antenna Terminals) strategy, where we use it to show that the sum-rate can be expressed as a product of two ratios of quadratic forms. At this point, the link between ratios of quadratic forms and generalized eigenvectors outlined in Section 3.2 is also exploited to derive a semi-algebraic solution.

The EVD and the SVD discussed in Section 3.3 form the basis for low-rank approximations of noisy observations which are used for denoising in Part II and Part III.

Moreover, the Least-Squares Khatri-Rao factorization introduced in Section 3.4 is the key element to the R-D extension ($R > 3$) of the SECSI framework for efficient CP decomposition discussed in Part II of the thesis. It is also one of the two steps of the Tensor-Based Channel Estimation (TENCE) algorithm for channel estimation in two-way relaying which is shown in Part IV, Chapter 15.

The subsequent Section 4.1 is used for the tensor-based algebraic manipulations in the entire Part II and III, and even parts of Part IV (Chapter 15).

Finally, the Higher-Order SVD and the CP decomposition form the basis for Part II of this thesis. Particularly, the fact that they can be represented in a very similar algebraic form via repeated n-mode product has been the motivation to find a CP decomposition by first computing the HOSVD and then applying further processing to it. The HOSVD is also the basis for the tensor-based subspace estimation scheme (cf. Section 10.2) used for Tensor-ESPRIT-type parameter estimation algorithms discussed in Part III. Moreover, the CP decomposition is employed in Part IV to facilitate the design of the relaying strategy during the training phase needed for TENCE (cf. Chapter 15).

5.2. Bibliographical notes and further reading

Many of the matrix-based results described in Chapter 3 can be found in dedicated matrix algebra books. For instance, "The matrix cookbook" [PP08] was long known[1] as a standard compendium on many useful algebraic results, such as linear and quadratic forms, Kronecker and Khatri-Rao products, or derivatives. It comprises a collection of results from many sources. For instance, as mentioned earlier, [Bre78] already lists many properties of the Kronecker and the Khatri-Rao product (which date back to the much earlier [Neu69] and [KR68], respectively). While the fact that Kronecker and Khatri-Rao products can be vectorized into a canonical linear vector form seems obvious at first sight, we could not find a reference for the explicit expressions presented in Proposition 3.1.1 and Proposition 3.1.2, respectively.

The history of the SVD/EVD goes back even further. As shown in [Ste93], early definitions of the SVD date back to the late 19th century and decompositions similar to the EVD have been known since the early 19th century. It is the widespread immersive use of digital signal processing nowadays that has sparked new applications for this long-known mathematical

[1] Unfortunately, as of 2010, the matrix cookbook is no longer maintained by the authors.

theory.

Concerning the Least-Squares Khatri-Rao and Kronecker factorization, it is important to mention [LP93], which discusses the factorization of Kronecker products. In fact, both cases (one factor known, both factors unknown) are discussed, however, the solution is derived in a different manner and stated in a different algebraic form. We relate our expression to [LP93] in Section B.6. An example where Kronecker factorization has been applied is given by [WJS08], where the authors find estimates of Kronecker-structured covariance matrices which appear, for instance, in certain MIMO channel models. The extension to the Khatri-Rao factorization is more or less trivial, as it proceeds by applying the Kronecker factorization column by column. It has been used by other authors without citing any source, e.g., in [LV08] or [CST10].

For the multilinear algebra and the tensor decompositions, a comprehensive survey is given in [KB09]. In particular, [KB09] lists not only the mathematical implications of tensor algebra and the corresponding decompositions but also provides accurate historical context and quotes many practical applications of the theory. However, some aspects presented in this part of the thesis are not discussed, e.g., the permutation matrices introduced in (4.11). Note that several MATLAB toolboxes are available for performing multilinear algebraic operations and computing various tensor decompositions. Popular examples include the "N-way toolbox" [AB00] the "MATLAB Tensor Toolbox" from [BK$^+$12], or the "Tensor Package" from [CLdA11]. In this thesis, parts of the N-way toolbox [AB00] abd the MATLAB Tensor Toolbox [BK$^+$12] are used in Part II as a benchmark for our semi-algebraic solutions, see Section 7.5.

Note that this part of the thesis could still be significantly expanded with many more algebraic concepts that are useful for engineers. For instance, matrix differential calculus (which is very convenient for computing gradients) was not discussed here but many results exist [MN95, PP08]. In light of the increasing importance of multilinear algebra, the differentiation with respect to tensors could be investigated as well.

Part II.

Semi-Algebraic CP Decomposition (SECSI)

This part of the thesis is devoted to the efficient computation of the decomposition of a given multi-dimensional signal into rank-one components known as Canonical Polyadic (CP) decomposition. There is a large number of applications where the signal of interest can be modeled as a linear superposition of multi-dimensional rank-one components that have to be separated. These range from psychometrics and chemometrics over array signal processing and communications to biomedical signal processing, image compression or numerical mathematics.

Existing algorithms to compute the CP decomposition are very often based on iterative techniques such as Alternating Least Squares (ALS). The main drawback of ALS-based schemes is that they may require a very large number of iterations to converge and are not guaranteed to converge to a global optimum. In particular, ALS-based approaches are very sensitive to ill-conditioned data.

Therefore, we introduce a novel semi-algebraic framework for computing an approximate CP decomposition of noisy multi-dimensional data based on Simultaneous Matrix Diagonalization (SECSI). We demonstrate that due to the structure of the problem, multiple Simultaneous Matrix Diagonalizations (SMDs) can be solved which gives rise to multiple estimates of the model parameters. The final estimate can then be chosen in a subsequent step. This approach allows flexible control over the complexity-performance trade-off and can be used to tailor the algorithm to the specific needs of the various applications. We establish the full system of possible SMDs in the 3-D and the general R-D case and propose several heuristic algorithms to select which SMDs to solve and how to obtain the final estimate of the CP model. Numerical simulations demonstrate the reduction on computational complexity and the improvement in robustness against ill-conditioned data compared to state-of-the-art algorithms.

This part of the thesis is organized as follows: Chapter 6 contains an introduction with a motivation, the state of the art review, and the definition of the data model. The derivation of our novel SECSI framework and its evaluation are presented in Chapter 7. Chapter 8 contains a summary and outlines possible future work. Appendix C contains proofs and derivations for this part.

6. Introduction to efficient tensor decompositions

6.1. Motivation, state of the art and own contribution

Separating linear mixtures into components is a fundamental task in signal processing due to its wide range of applications. Tensor-based signal processing schemes represent a promising approach for this task. This is due to the fact that multilinear decompositions offer fundamental advantages over their bilinear (matrix) counterparts. In general, the tensor structure allows more efficient denoising, which can be used to compute a tensor-based signal subspace estimate [HRD08]. Additionally, a trilinear (in general, a multilinear) mixture features inherent essential uniqueness while bilinear mixtures require additional assumptions. Furthermore, identifiability is improved, allowing more components to be present.

In this part of the thesis we focus on the decomposition of a given multidimensional signal into a sum of rank-one components, which is referred to as CANDECOMP / PARAFAC [CC70, Har70], or Canonical Polyadic (CP) decomposition, cf. Section 4.2.2 for a definition and properties of the CP decomposition. There exist many applications where the underlying signal of interest can be represented by a trilinear or multilinear CP model. These range from psychometrics [CC70, Har70] and chemometrics [AB03] over array signal processing [SBG00] and communications [SGB00] to biomedical signal processing [Möc88, MHH+06, dVVdL+07, WRH+09, BCA+12], image compression [SL01] or numerical mathematics [HKT05] (see also [KB09] and references therein).

In practice, the signal of interest is typically contaminated by additive noise. Therefore we need to find an approximate CP decomposition of the noisy data. This is a challenging problem, since finding best low-rank approximations is itself an ill-posed problem [KB09]. Fully algebraic solutions are usually not available, unless strong approximations are made [SK90] (leading to severe fitting errors) or the decomposition possesses a specific structure [CST10] (for instance, a Toeplitz-structured loading matrix).

However, many iterative algorithms for computing approximate CP decompositions exist. A popular class of algorithms is based on the Alternating Least Squares (ALS) procedure [KdL80, BSG99] where linear least squares fits are computed for each dimension separately, alternating through dimensions in a sequential manner. The original idea goes back to [CC70, Har70, KdL80] and many improvements have been proposed since then, e.g., enhanced line search (ELS) [RC05, RCH08] or Tikhonov regularization [NdLK08]. The main shortcoming of ALS-based algorithms is that they may require a large number of iterations to converge and are not guaranteed to reach the global optimum. This motivates searching for alternative approaches that are more reliable and have a lower computational complexity.

There exist also algorithms that are not directly based on ALS. For instance, [ADK11] contains a partial survey about optimization-based approaches, including a non-linear least squares (CPNLS) as well as a gradient-based method (CPOPT). In [CLdA09], quasi-Newton and gradient descent based methods with periodic ELS have been reported. A robust iterative fitting framework which minimizes least absolute errors is presented in [VRSG05] along with two algorithms, one based on interior point methods and a second one based on weighted median filtering.

The links between the CP decomposition and Simultaneous Matrix Diagonalizations (SMDs) were already pointed out by [vdVP96, AFCC04] for the symmetric case and by [dL04a, dL06] for the non-symmetric case. Recently, [AFCC04] was also generalized to the non-symmetric case [LA11], which differs from [dL04a] in the way the unfoldings are defined. However, these approaches have in common that only a single SMD is solved and all parameters are estimated from the solution of this SMD.

In this part of the thesis, we present a **SE**mi-algebraic framework for computing an approximate **C**P decomposition via **SI**multaneous Matrix Diagonalizations (**SECSI**) [RH08b, RH08a][1]. SECSI is applicable if the tensor rank is less than or equal to the size of the tensor in at least two modes. We show that due to the symmetry of the problem instead of only one, many SMDs can be constructed, and we establish the full set of possible SMDs. Therefore, if accuracy is the prime concern, all SMDs can be solved to obtain multiple estimates and the "best" solution can be selected in a subsequent step. On the other hand, if computational complexity is critical we can solve one or a few of the SMDs. Moreover, the selection can be based on an exhaustive best matching scheme which minimizes the reconstruction error or on appropriate heuristics with a lower computational complexity. Consequently, the complexity-accuracy trade-off can be controlled flexibly, allowing the algorithms to adapt to applications with different needs. Such an adaptation is also beneficial when correlations exist in some dimensions, leading to asymmetric conditioning of the decomposition. We investigate the three-way as well as the general R-way case in Sections 7.1 and 7.2, respectively, and propose heuristics to decide which SMDs to solve and how to pick the final estimates in

[1]In the earlier conference versions [RH08b] and [RH08a], the SECSI framework was referred to as "closed-form solution" referring to the fact that albeit SMDs are computed iteratively this is also true for SVDs and matrix inversions, which are usually considered closed form. We have modified the name to *semi-algebraic* solution to account for the iterative nature of SMD algorithms and the fact that they are not guaranteed to find the global optimum.

Section 7.4. Moreover, we investigate the special cases of a two-slab and a two-component CP decomposition in Section 7.3, showing that in both cases, the SECSI framework reduces to known fully algebraic solutions [SK90], alleviating the need for computing any SMD. We compare the achievable accuracy and the complexity in terms of run-times in Section 7.5 via simulations based on synthetical data.

6.2. Notation and data model

6.2.1. R-way CP Decomposition

We model our observed data tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R}$ as a desired signal component $\boldsymbol{\mathcal{X}}_0$ superimposed by additive noise, i.e.,

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}}, \tag{6.1}$$

where the tensor $\boldsymbol{\mathcal{N}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R}$ contains samples of the zero mean additive noise. The rank of our desired signal component $\boldsymbol{\mathcal{X}}_0$ is denoted by d and assumed to be known. In practice d has to be estimated by appropriate model order selection schemes, such as the Core Consistency Diagnostics (CORCONDIA) [BK03] or the R-D Exponential Fitting Test (EFT) [DHRD07]. For a comparison of tensor-based model order selection techniques, the reader is referred to [DRHdS11].

Since $\boldsymbol{\mathcal{X}}_0$ is rank-d, it can be decomposed into a sum of d rank-one tensors. As discussed in Section 4.2.2, we can express the CP decomposition of $\boldsymbol{\mathcal{X}}_0$ via (cf. (4.30) and (4.34))

$$\boldsymbol{\mathcal{X}}_0 = \sum_{n=1}^{d} \boldsymbol{f}_n^{(1)} \circ \boldsymbol{f}_n^{(2)} \circ \ldots \circ \boldsymbol{f}_n^{(R)} \tag{6.2}$$

$$= \boldsymbol{\mathcal{I}}_{R,d} \times_1 \boldsymbol{F}^{(1)} \times_2 \boldsymbol{F}^{(2)} \ldots \times_R \boldsymbol{F}^{(R)} = \boldsymbol{\mathcal{I}}_{R,d} \bigtimes_{r=1}^{R}{}_r \boldsymbol{F}^{(r)}, \tag{6.3}$$

where $\boldsymbol{f}_n^{(r)} \in \mathbb{C}^{M_r \times 1}$ is referred to as the loading vector of the n-th component in the r-th mode and $\boldsymbol{F}^{(r)} = \left[\boldsymbol{f}_1^{(r)}, \ldots, \boldsymbol{f}_d^{(r)}\right] \in \mathbb{C}^{M_r \times d}$ is referred to as the loading matrix in the r-th mode.

Recall that since the sum in (6.2) is commutative and the products are associative, the loading vectors $\boldsymbol{f}_n^{(r)}$ are determined only up to an inherent permutation ambiguity (between different n) and a scaling ambiguity (between different r).

It has been shown that many practical problems can be expressed in the form of (6.3), which explains why the CP decomposition has been successfully applied in a broad field of applications ranging from psychometrics over chemometrics to communications and array signal processing [KB09].

One of the major challenges in data-driven applications is that only a noise-corrupted version of the rank-d tensor $\boldsymbol{\mathcal{X}}_0$ is observed which means that we cannot find an exact but only an *approximate* CP decomposition providing a "best fit" of the observed data according to a previously selected criterion, e.g., the higher order norm.

It is apparent that (6.3) is algebraically very similar to the truncated HOSVD discussed in Section 4.2.1. The truncated HOSVD of $\boldsymbol{\mathcal{X}}_0$ can be expressed as (cf. (4.29))

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \times_2 \boldsymbol{U}_2^{[\mathrm{s}]} \dots \times_R \boldsymbol{U}_R^{[\mathrm{s}]} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \bigtimes_{r=1}^{R}{}_r \boldsymbol{U}_r^{[\mathrm{s}]}, \tag{6.4}$$

where p_r is the r-rank of $\boldsymbol{\mathcal{X}}_0$, $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \in \mathbb{C}^{p_1 \times p_2 \dots \times p_R}$ is the core tensor, and $\boldsymbol{U}_r^{[\mathrm{s}]} \in \mathbb{C}^{M_r \times p_r}$ contains the p_r dominant left singular vectors of $[\boldsymbol{\mathcal{X}}_0]_{(r)}$ for $r = 1, 2, \dots, R$.

Note that (6.4) and (6.3) have a very similar algebraic structure. This similarity can be exploited to compute a CP decomposition efficiently. We start with the HOSVD, which is easy to compute via matrix SVDs, and then convert the result into a CP decomposition. This form of preprocessing is also referred to as Tucker compression [CLdA09], a common approach for dimensionality reduction in tensor-based signal processing. We then compute the CP decomposition by finding the required transform matrices via SMDs constructed from slices of the compressed tensor. This is the underlying idea of the SECSI framework which is derived in the following sections. We start with the 3-way case since it is easier to understand and to visualize and then show the extension to the general R-way case.

Note that in the presence of noise, the observed tensor $\boldsymbol{\mathcal{X}}$ in (6.1) has full n-ranks with probability one if the elements of $\boldsymbol{\mathcal{N}}$ are drawn from a continuous distribution. Therefore, (6.4) holds only approximately. Moreover, as argued in Section 4.2.1, the best low-rank approximation of $\boldsymbol{\mathcal{X}}$ in the Frobenius norm sense requires the iterative HOOI algorithm since the truncated HOSVD is in general suboptimal. However, since the improvement the HOOI algorithm provides compared to the truncated HOSVD is very small, we use the truncated HOSVD to find (6.4), unless stated otherwise. However, we emphasize that the HOOI or any other low-rank approximation can be used as well.

6.2.2. Degeneracies

We refer to the CP decomposition as *degenerate* in the r-th mode if $p_r < d$, i.e., the r-rank of the desired signal component $\boldsymbol{\mathcal{X}}_0$ is less than the tensor rank d. An r-mode degeneracy occurs if the r-mode loading matrix $\boldsymbol{F}^{(r)}$ does not have full column rank. This is always the case for $M_r < d$. For $M_r \geq d$, degeneracies can be caused by linear dependencies among the columns of $\boldsymbol{F}^{(r)}$. Due to the way the SMDs are constructed we need another assumption for non-degeneracy, namely, $\boldsymbol{F}^{(r)}$ needs to have one row where it contains no zero elements.[2]

For our analysis all the causes of degeneracy are treated in the same manner. As we show in the subsequent sections, our SECSI framework is applicable if at least two of the R modes are non-degenerate, allowing for degeneracies in up to $R-2$ modes. This implies the assumption $d \leq M_r$ for at least two values of $r = 1, 2, \ldots, R$.

[2]While this is an additional assumption needed for the SECSI framework, it is not a very critical one. For factors drawn from continuous distributions such elements that are equal to zero occur with probability zero. Loading matrices with many zeros are often the consequence of an underlying structure in which case fully algebraic solutions like [CST10] may be applicable, or they are the consequence of intrinsic sparseness in the data in which case suitable sparse decompositions should be used [BK07].

7. Semi-Algebraic CP Decomposition via Simultaneous Matrix Diagonalization (SECSI)

This chapter is devoted to the Semi-Algebraic CP decomposition via Simultaneous Matrix Diagonalization (SECSI) framework. We start by discussing the 3-D case in Section 7.1 for clarity. The extension to the R-D case is then shown in Section 7.2. Section 7.3 contains the discussion of some special cases where SECSI reduces to known algebraic solutions. Heuristic approaches to decide which Simultaneous Matrix Diagonalizations (SMDs) te solve and how to select the final estimates are introduced in Section 7.4. These give rise to different algorithms inside the SECSI framework which we evaluate in terms of their achieved estimation accuracy and computational complexity in Section 7.5. Finally, a summary and an outlook to future work is provided in Section 7.6.

7.1. Three-way CP

In this section we focus our attention on the three-way CP decomposition for clarity. The extension to the general R-way CP decomposition is provided in Section 7.2.

7.1.1. Construction of Simultaneous Matrix Diagonalizations (SMDs)

Let us first consider the case that the CP decomposition is non-degenerate in all three modes, i.e., $p_1 = p_2 = p_3 = d$. For clarity of presentation, the following derivations relate to the noise-free case, i.e., $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0$, since this assumption allows us to write equalities. In the presence of noise, the following relations hold only approximately. Consequently, the noise-free tensor $\boldsymbol{\mathcal{X}}$ can be expressed as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{I}}_{3,d} \times_1 \boldsymbol{F}^{(1)} \times_2 \boldsymbol{F}^{(2)} \times_3 \boldsymbol{F}^{(3)} \tag{7.1}$$

$$= \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \times_2 \boldsymbol{U}_2^{[\mathrm{s}]} \times_3 \boldsymbol{U}_3^{[\mathrm{s}]}, \tag{7.2}$$

where the dimension of the truncated HOSVD are now $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \in \mathbb{C}^{d \times d \times d}$ and $\boldsymbol{U}_r^{[\mathrm{s}]} \in \mathbb{C}^{M_r \times d}$ (for a discussion on the use of the truncated HOSVD for noisy data cf. discussion at the end of Section 4.2.1).

Let us consider the one-mode unfolding of $\boldsymbol{\mathcal{X}}$. Using the representations (7.1) and (7.2) we find that

$$\boldsymbol{U}_1^{[\mathrm{s}]} \cdot \left[\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}\right]_{(1)} \cdot \left(\boldsymbol{U}_2^{[\mathrm{s}]} \otimes \boldsymbol{U}_3^{[\mathrm{s}]}\right)^{\mathrm{T}} = \boldsymbol{F}^{(1)} \cdot \left[\boldsymbol{\mathcal{I}}_{3,d}\right]_{(1)} \cdot \left(\boldsymbol{F}^{(2)} \otimes \boldsymbol{F}^{(3)}\right)^{\mathrm{T}}.$$

Consequently, the subspace spanned by the columns of $\boldsymbol{U}_1^{[\mathrm{s}]}$ and $\boldsymbol{F}^{(1)}$ are identical. This implies that there exist a non-singular transform matrix $\boldsymbol{T}_1 \in \mathbb{C}^{d \times d}$, such that $\boldsymbol{F}^{(1)} = \boldsymbol{U}_1^{[\mathrm{s}]} \cdot \boldsymbol{T}_1$. By applying the same reasoning to the two-mode unfolding and the three-mode unfolding of $\boldsymbol{\mathcal{X}}$ we find that $\boldsymbol{F}^{(2)} = \boldsymbol{U}_2^{[\mathrm{s}]} \cdot \boldsymbol{T}_2$ and $\boldsymbol{F}^{(3)} = \boldsymbol{U}_3^{[\mathrm{s}]} \cdot \boldsymbol{T}_3$ for non-singular matrices $\boldsymbol{T}_2, \boldsymbol{T}_3 \in \mathbb{C}^{d \times d}$.

Inserting these relations into (7.1) we obtain

$$\begin{aligned}
\boldsymbol{\mathcal{X}} &= \boldsymbol{\mathcal{I}}_{3,d} \times_1 \left(\boldsymbol{U}_1^{[\mathrm{s}]} \cdot \boldsymbol{T}_1\right) \times_2 \left(\boldsymbol{U}_2^{[\mathrm{s}]} \cdot \boldsymbol{T}_2\right) \times_3 \left(\boldsymbol{U}_3^{[\mathrm{s}]} \cdot \boldsymbol{T}_3\right) \\
&= \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \times_2 \boldsymbol{U}_2^{[\mathrm{s}]} \times_3 \boldsymbol{U}_3^{[\mathrm{s}]} \\
\Rightarrow \boldsymbol{\mathcal{I}}_{3,d} &= \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{T}_1^{-1} \times_2 \boldsymbol{T}_2^{-1} \times_3 \boldsymbol{T}_3^{-1} && (7.3) \\
\Rightarrow \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} &= \boldsymbol{\mathcal{I}}_{3,d} \times_1 \boldsymbol{T}_1 \times_2 \boldsymbol{T}_2 \times_3 \boldsymbol{T}_3. && (7.4)
\end{aligned}$$

This establishes the relation between the core tensor of the HOSVD and the transform matrices $\boldsymbol{T}_r$, which can be interpreted in two ways: (a) the inverse transform matrices $\boldsymbol{T}_1^{-1}$, $\boldsymbol{T}_2^{-1}$, and $\boldsymbol{T}_3^{-1}$ diagonalize the core tensor, as shown in equation (7.3) or (b) the transform matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ represent the loading matrices of the CP decomposition of the $d \times d \times d$ core tensor of the HOSVD, as shown in equation (7.4).

Consequently, it is sufficient to estimate the transform matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ from the HOSVD of the data. This can be achieved via Simultaneous Matrix Diagonalizations (SMDs). To this end, let $\boldsymbol{\mathcal{S}}_3^{[\mathrm{s}]} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_3 \boldsymbol{U}_3^{[\mathrm{s}]} \in \mathbb{C}^{d \times d \times M_3}$ and $\boldsymbol{\mathcal{D}}_3 = \boldsymbol{\mathcal{I}}_{3,d} \times_3 \boldsymbol{F}^{(3)}$. The tensor $\boldsymbol{\mathcal{D}}_3$ is visualized in Figure 7.1. Its three-mode slices are diagonal matrices, given by $[\boldsymbol{\mathcal{D}}_3]_{(:,:,n)} = \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(n,:)}\right\}$, $n = 1, 2, \ldots, M_3$. In other words, the n-th three-mode slice of $\boldsymbol{\mathcal{D}}_3$ contains the n-th row of $\boldsymbol{F}^{(3)}$ on its main diagonal.

With the help of (7.1) and (7.2) it is easily verified that

$$\boldsymbol{\mathcal{S}}_3^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \times_2 \boldsymbol{U}_2^{[\mathrm{s}]} = \boldsymbol{\mathcal{D}}_3 \times_1 \boldsymbol{F}^{(1)} \times_2 \boldsymbol{F}^{(2)}. \tag{7.5}$$

Inserting $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{T}_r$ for $r = 1, 2$ into (7.5) and then multiplying with $\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}}$ in the r-th mode for $r = 1, 2$, we obtain

$$\begin{aligned}
\boldsymbol{\mathcal{S}}_3^{[\mathrm{s}]} &= \boldsymbol{\mathcal{D}}_3 \times_1 \boldsymbol{T}_1 \times_2 \boldsymbol{T}_2 && (7.6) \\
\Rightarrow \boldsymbol{\mathcal{D}}_3 &= \boldsymbol{\mathcal{S}}_3^{[\mathrm{s}]} \times_1 \boldsymbol{T}_1^{-1} \times_2 \boldsymbol{T}_2^{-1}. && (7.7)
\end{aligned}$$

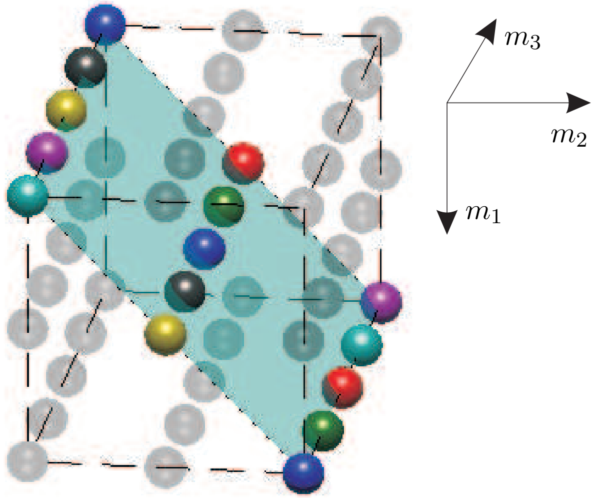


Figure 7.1.: Structure of the tensor $\mathcal{D}_3 = \mathcal{I}_{3,d} \times_3 \boldsymbol{F}^{(3)} \in \mathbb{C}^{d \times d \times M_3}$ for $M_3 = 5$ and $d = 3$. The first index m_1 is varied along the rows, the second index m_2 along the columns, and the third index m_3 along the lateral direction, starting with the top-left-front element. Gray circles represent zero elements.

Equation (7.7) shows the link to the joint diagonalization of matrices, since the fixed transform matrices $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ transform the slices of the full tensor $\mathcal{S}_3^{[\mathrm{s}]}$ into the diagonal matrix slices of the tensor $\mathcal{D}_3$ depicted in Figure 7.1.

Formally, the k-th three-mode slice of a tensor can be extracted by multiplying with $\boldsymbol{e}_{M_3,k}^{\mathrm{T}}$ in the third mode, where $\boldsymbol{e}_{M_3,k}^{\mathrm{T}}$ is the k-th row of an $M_3 \times M_3$ identity matrix. Applying this slicing operator to (7.7) yields

$$\underbrace{\mathcal{D}_3 \times_3 \boldsymbol{e}_{M_3,k}^{\mathrm{T}}}_{\mathrm{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(k,:)}\right\}} = \underbrace{\mathcal{S}_3^{[\mathrm{s}]} \times_3 \boldsymbol{e}_{M_3,k}^{\mathrm{T}}}_{\boldsymbol{S}_{3,k}} \times_1 \boldsymbol{T}_1^{-1} \times_2 \boldsymbol{T}_2^{-1} \tag{7.8}$$

$$\mathrm{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(k,:)}\right\} = \boldsymbol{T}_1^{-1} \cdot \boldsymbol{S}_{3,k} \cdot \boldsymbol{T}_2^{-\mathrm{T}}, \tag{7.9}$$

where $k = 1, 2, \ldots, M_3$. The SMD in (7.9) is “asymmetric” in the sense that the matrices $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ are completely different and possess no specific relation. Typically, SMDs are cast as finding a square invertible matrix $\boldsymbol{M}$ such that $\boldsymbol{M} \cdot \boldsymbol{X}_n \cdot \boldsymbol{M}^{-1}$ becomes as diagonal as possible for a set of N given matrices $\boldsymbol{X}_n, n = 1, 2, \ldots, N$. In order to rewrite (7.9) into such a “symmetric” form, we define two new sets of matrices by multiplying all slices by the inverse of one particular slice from the right hand side (rhs) and the left hand side (lhs), respectively[1]. Thereby, we obtain

$$\boldsymbol{S}_{3,k}^{\mathrm{rhs}} = \boldsymbol{S}_{3,k} \cdot \boldsymbol{S}_{3,s_3}^{-1} \quad \text{and} \quad \boldsymbol{S}_{3,k}^{\mathrm{lhs}} = \left(\boldsymbol{S}_{3,s_3}^{-1} \cdot \boldsymbol{S}_{3,k}\right)^{\mathrm{T}}, \tag{7.10}$$

where the additional transpose is introduced for symmetry reasons. Here s_3 refers to one arbitrary fixed integer number between 1 and M_3; its choice is discussed below. Inserting (7.9) into (7.10) we obtain

$$\begin{aligned} \boldsymbol{S}_{3,k}^{\mathrm{rhs}} &= \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_2^{\mathrm{T}} \cdot \boldsymbol{T}_2^{-\mathrm{T}} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(s_3,:)}\right\}^{-1} \cdot \boldsymbol{T}_1^{-1} \\ &= \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(k,:)}\right\} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(s_3,:)}\right\}^{-1} \cdot \boldsymbol{T}_1^{-1} \\ \Rightarrow \boldsymbol{S}_{3,k}^{\mathrm{rhs}} &= \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(3)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_1^{-1}, \end{aligned} \tag{7.11}$$

where $\tilde{\boldsymbol{F}}^{(3)} = \boldsymbol{F}^{(3)} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(s_3,:)}\right\}^{-1}$. Likewise, for $\boldsymbol{S}_{3,k}^{\mathrm{lhs}}$ we get

$$\begin{aligned} \boldsymbol{S}_{3,k}^{\mathrm{lhs}} &= \boldsymbol{T}_2 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(k,:)}\right\} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(s_3,:)}\right\}^{-1} \cdot \boldsymbol{T}_2^{-1} \\ &= \boldsymbol{T}_2 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(3)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_2^{-1}. \end{aligned} \tag{7.12}$$

Equations (7.11) and (7.12) show that $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ can be found by simultaneous diagonalization of the matrix slices $\boldsymbol{S}_{3,k}^{\mathrm{rhs}}$ and $\boldsymbol{S}_{3,k}^{\mathrm{lhs}}$, respectively. This provides us with $\boldsymbol{F}^{(1)}$ and $\boldsymbol{F}^{(2)}$ via the relation $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[s]} \cdot \boldsymbol{T}_r$. The simultaneous diagonalization of (7.11) and (7.12) can, for instance, be achieved via the joint diagonalization algorithm proposed in [FG06].

Note that the diagonal elements of the matrices $\boldsymbol{T}_1^{-1} \cdot \boldsymbol{S}_{3,k}^{\mathrm{rhs}} \cdot \boldsymbol{T}_1$ as well as $\boldsymbol{T}_2^{-1} \cdot \boldsymbol{S}_{3,k}^{\mathrm{rhs}} \cdot \boldsymbol{T}_2$ provide an estimate for the matrix $\tilde{\boldsymbol{F}}^{(3)}$, which identifies $\boldsymbol{F}^{(3)}$ up to one scaling ambiguity per column. This ambiguity is irrelevant since the CP decomposition itself is only unique up to column permutation and scaling (cf. discussion in Section 6.2).

Consequently, we have one estimate for $\boldsymbol{F}^{(1)}$ and $\boldsymbol{F}^{(3)}$ from the SMD of $\boldsymbol{S}_{3,k}^{\mathrm{rhs}}$. The missing $\boldsymbol{F}^{(2)}$ could easily be found via a pseudo inverse[2], i.e., $\boldsymbol{F}^{(2)} = [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\boldsymbol{F}^{(3)} \diamond \boldsymbol{F}^{(1)}\right)^{+}\right]^{\mathrm{T}}$. Note that while in the absence of noise this solution is exact, in the presence of noise it represents a least squares (LS) estimate. Likewise, a second estimate for all three loading matrices is

[1] A similar technique was used in [AFCC04] for the CP decomposition of a pair-wise symmetric $2q$-th order cumulant tensor.

[2] For those heuristics that discard SMDs before reconstructing the tensor (e.g., the RES heuristic), we can skip this step for all discarded SMDs.

obtained by augmenting the estimate for $\boldsymbol{F}^{(2)}$ and $\boldsymbol{F}^{(3)}$ obtained from $\boldsymbol{S}_{3,k}^{\text{lhs}}$ by a LS fit for $\boldsymbol{F}^{(1)}$, i.e., $\boldsymbol{F}^{(1)} = [\boldsymbol{\mathcal{X}}]_{(1)} \cdot \left[\left(\boldsymbol{F}^{(2)} \diamond \boldsymbol{F}^{(3)}\right)^{+}\right]^{\text{T}}$.

It is possible to construct even more SMDs by considering other mode-combinations. Up to here we have estimated $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ by diagonalizing three-mode slices. Similarly, we can estimate $\boldsymbol{T}_1$ and $\boldsymbol{T}_3$ by diagonalizing two-mode slices and $\boldsymbol{T}_2$ and $\boldsymbol{T}_3$ by diagonalizing one-mode slices. The corresponding SMDs are given by

$$\boldsymbol{S}_{2,k}^{\text{rhs}} = \boldsymbol{S}_{2,k} \cdot \boldsymbol{S}_{2,s_2}^{-1} = \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(2)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_1^{-1} \tag{7.13}$$

$$\boldsymbol{S}_{2,k}^{\text{lhs}} = \left(\boldsymbol{S}_{2,s_2}^{-1} \cdot \boldsymbol{S}_{2,k}\right)^{\text{T}} = \boldsymbol{T}_3 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(2)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_3^{-1} \tag{7.14}$$

$$\tilde{\boldsymbol{F}}^{(2)} = \boldsymbol{F}^{(2)} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(2)}\right]_{(s_2,:)}\right\}^{-1} \tag{7.15}$$

$$\boldsymbol{S}_{2,k} = \left[\left(\boldsymbol{\mathcal{S}}^{[\text{s}]} \times_2 \boldsymbol{U}_2^{[\text{s}]}\right) \times_2 \boldsymbol{e}_{M_2,k}^{\text{T}}\right]_{(1)} \tag{7.16}$$

for $\boldsymbol{T}_1$ and $\boldsymbol{T}_3$ and similarly,

$$\boldsymbol{S}_{1,k}^{\text{rhs}} = \boldsymbol{S}_{1,k} \cdot \boldsymbol{S}_{1,s_1}^{-1} = \boldsymbol{T}_2 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(1)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_2^{-1} \tag{7.17}$$

$$\boldsymbol{S}_{1,k}^{\text{lhs}} = \left(\boldsymbol{S}_{1,s_1}^{-1} \cdot \boldsymbol{S}_{1,k}\right)^{\text{T}} = \boldsymbol{T}_3 \cdot \operatorname{diag}\left\{\left[\tilde{\boldsymbol{F}}^{(1)}\right]_{(k,:)}\right\} \cdot \boldsymbol{T}_3^{-1} \tag{7.18}$$

$$\tilde{\boldsymbol{F}}^{(1)} = \boldsymbol{F}^{(1)} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(1)}\right]_{(s_1,:)}\right\}^{-1} \tag{7.19}$$

$$\boldsymbol{S}_{1,k} = \left[\left(\boldsymbol{\mathcal{S}}^{[\text{s}]} \times_1 \boldsymbol{U}_1^{[\text{s}]}\right) \times_1 \boldsymbol{e}_{M_1,k}^{\text{T}}\right]_{(2)}. \tag{7.20}$$

for $\boldsymbol{T}_2$ and $\boldsymbol{T}_3$, respectively. Note that the unfolding used in the definition of $\boldsymbol{S}_{1,k}$ and $\boldsymbol{S}_{2,k}$ is introduced to convert the $d \times 1 \times d$ tensor and the $1 \times d \times d$ tensor into $d \times d$ matrices. Moreover, s_1 (s_2) represents an arbitrary integer number between 1 and M_1 (M_2).

Overall this shows that we can construct $N_{\text{SMD}} = 6$ SMDs and obtain an estimate of all three loading matrices from each of them. This provides us with up to six estimates for each factor matrix $\boldsymbol{F}^{(r)}$. We will refer to the m-th estimate of the r-th factor matrix as $\hat{\boldsymbol{F}}_m^{(r)}$.

Note that we have transformed the "asymmetric" SMDs into "symmetric" ones by multiplying with the inverse of one particular slice $s_r \in \{1, 2, \ldots, M_r\}$. While for an exact CP decomposition the choice of s_r is indeed irrelevant, for an approximate CP decomposition a meaningful choice is given by

$$s_r = \underset{s=1,2,\dots,M_r}{\arg\min} \ \mathrm{cond}\left(\boldsymbol{S}_{r,s}\right), \ r = 1,2,3, \tag{7.21}$$

i.e., picking the slice with the best (smallest) conditioning number. This process is similar to the selection of pivots when solving sets of linear equations.

7.1.2. Degeneracy

Up to this point we have assumed that the CP decomposition is non-degenerate in all three modes, i.e., $p_1 = p_2 = p_3 = d$ so that we can express the r-th loading matrix as $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{T}_r$. We now consider the case that the CP decomposition is degenerate in one mode and non-degenerate in the remaining two modes.

Without loss of generality let us assume that the first mode is degenerate, i.e., $p_1 < d$, $p_2 = d$, and $p_3 = d$. Therefore, $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{T}_r$ remains valid for $r = 2,3$. Consequently, we can still establish the SMDs for $\boldsymbol{S}_{1,k}^{\mathrm{rhs}}$ and $\boldsymbol{S}_{1,k}^{\mathrm{lhs}}$ from which we obtain two estimates for all three loading matrices, as discussed above. However, since a transform matrix $\boldsymbol{T}_1$ does not exist anymore there are no corresponding SMDs for $\boldsymbol{S}_{2,k}$ and for $\boldsymbol{S}_{3,k}$. This shows that the SECSI framework still applies with the main difference that the number of SMDs we can establish is reduced from six to two.

7.1.3. Uniqueness

It is easy to show that the SMD-based approach to compute an approximate CP decomposition does not introduce any additional ambiguities. Therefore, as long as the tensor has two non-degenerate modes, SECSI yields estimates for the loading matrices that are unique up to a scaling and a (consistent) permutation of the columns. This is summarized in the following Theorem:

Theorem 7.1.1. *In the absence of noise, the estimates of the loading matrices from the n-th SMD satisfy*

$$\hat{\boldsymbol{F}}_n^{(r)} = \boldsymbol{F}^{(r)} \cdot \boldsymbol{P}_n \cdot \boldsymbol{D}_{r,n}, \tag{7.22}$$

where $\boldsymbol{P}_n$ are a permutation matrices and $\boldsymbol{D}_{r,n} = \mathrm{diag}\left\{\boldsymbol{d}_{r,n}\right\} \in \mathbb{C}^{d \times d}$ are diagonal matrices with

$$\prod_{r=1}^{R} \boldsymbol{D}_{r,n} = \boldsymbol{I}_d. \tag{7.23}$$

Proof: cf. Appendix C.1.

Note that this theorem implies that the ambiguities are consistent within one SMD (n) but they may differ between SMDs. Therefore, when we combine estimates from different SMDs in order to improve the overall performance, these ambiguities need to be resolved. This is discussed in Section 7.4.

7.1.4. Symmetry

The model is said to possess a symmetry if two loading matrices are equal, i.e., $\boldsymbol{F}^{(r_1)} = \boldsymbol{F}^{(r_2)}$ for $r_1 \neq r_2 \in \{1, 2, 3\}$. Similarly, we refer to a Hermitian symmetry if one loading matrix is the complex conjugate of another loading matrix, i.e., $\boldsymbol{F}^{(r_1)} = \boldsymbol{F}^{(r_2)^*}$. In the special case where all factor matrices are equal, the model is referred to as pair-wise symmetric [dLdMV00b] or super-symmetric [dL06]. These symmetries are found in different applications. For instance, a CP decomposition with a Hermitian symmetry between the first and the second mode is obtained in [vdVP96, RVGS05][3]. Moreover, it can be shown that all real-valued cumulant tensors obey a pair-wise symmetric CP decomposition [Car91]. For complex-valued cumulant tensors, the loading matrices are equal or conjugates of each other, depending on how we choose the conjugation operations in the definition of the complex cumulants. Such a tensor still falls under the definition of pair-wise symmetric tensor given in [dLdMV00b], where arbitrary conjugations are allowed.

Note that the noise contribution typically obeys the same symmetry due to the processing applied to the original data. For instance, in [RVGS05], the Hermitian symmetry appears due to the computation of covariance matrices from the raw data. If this computation is performed on the noisy observation it induces the same Hermitian symmetry also in the additive noise component. Similarly, the above-mentioned cumulant tensors are pair-wise symmetric since the corresponding cumulants themselves possess a permutation symmetry in the indices due to their construction. This property remains valid also for cumulant tensors estimated from a finite data set.

If both, the desired signal component as well as the noise contribution possess the same symmetry, our SECSI framework readily allows to exploit this fact. First of all, we note that the HOSVD obeys the same symmetry[4], i.e., $\boldsymbol{U}_{r_1}^{[s]} = \boldsymbol{U}_{r_2}^{[s]}$ for symmetric modes (r_1, r_2) and $\boldsymbol{U}_{r_1}^{[s]} =$

[3] Although [vdVP96] is not using tensor notation and hence does not refer to the problem as a CP decomposition, it is recast as an SMD of the form $\boldsymbol{Y}_n = \boldsymbol{W}^{\mathrm{H}} \cdot \boldsymbol{\Lambda}_n \cdot \boldsymbol{W}$, where $\boldsymbol{\Lambda}_n$ is diagonal, $n = 1, 2, \ldots, N$. As evident from the discussion in this section, this is in fact in the same algebraic form as a CP model with a Hermitian symmetry. Note that [vdVP96] proposes a generalized simultaneous Schur decomposision to solve the approximate SMDs in the presence of noise.

[4] The HOSVD obeys the same symmetry provided that we choose the inherent scaling ambiguity of one unknown

$\boldsymbol{U}_{r_2}^{[\mathrm{s}]^*}$ for Hermitian symmetric modes (r_1, r_2). Consequently, the corresponding transform matrices $\boldsymbol{T}_{r_1}$ and $\boldsymbol{T}_{r_2}$ are identical for symmetric modes and complex conjugates of each other for Hermitian symmetric modes. Therefore, we can solve the "asymmetric" SMD in (7.9) directly via diagonalization by congruence. If the first two modes are symmetric, we obtain an SMD in the form of $\boldsymbol{T}_1 \cdot \boldsymbol{S}_{3,k} \cdot \boldsymbol{T}_1^{\mathrm{T}}$ and in case of a Hermitian symmetry we have $\boldsymbol{T}_1 \cdot \boldsymbol{S}_{3,k} \cdot \boldsymbol{T}_1^{\mathrm{H}}$. Both forms of the SMD can for instance be solved by the ACDC algorithm [Yer02] in its non-Hermitian and its Hermitian version, respectively. For the latter form, many alternative algorithms exist, e.g., the JTJD algorithm [GZMB10]. Note that in the pair-wise symmetric case we have $\boldsymbol{S}_{1,k} = \boldsymbol{S}_{2,k} = \boldsymbol{S}_{3,k}$ and hence it is sufficient to solve one of the three possible SMDs.

Alternatively, we can stick to the diagonalization by similarity transformation obtained by multiplying with the inverse of a pivot slice, as before. Even in this case, the symmetries can be exploited in the following ways: (a) For (Hermitian) symmetric modes (r_1, r_2) the slices $\boldsymbol{S}_{r_3,k}$ are (Hermitian) symmetric matrices and hence $\boldsymbol{S}_{r_3,k}^{\mathrm{rhs}} = \left(\boldsymbol{S}_{r_3,k}^{\mathrm{lhs}}\right)^{\mathrm{T}} \left(\boldsymbol{S}_{r_3,k}^{\mathrm{rhs}} = \left(\boldsymbol{S}_{r_3,k}^{\mathrm{lhs}}\right)^{\mathrm{H}}\right)$, it is therefore sufficient to solve one of these two SMDs; (b) For a pair-wise symmetric tensor where all loading matrices are equal we have $\boldsymbol{S}_{1,k}^{\mathrm{rhs}} = \left(\boldsymbol{S}_{1,k}^{\mathrm{lhs}}\right)^{\mathrm{T}} = \boldsymbol{S}_{2,k}^{\mathrm{rhs}} = \left(\boldsymbol{S}_{2,k}^{\mathrm{lhs}}\right)^{\mathrm{T}} = \boldsymbol{S}_{3,k}^{\mathrm{rhs}} = \left(\boldsymbol{S}_{3,k}^{\mathrm{lhs}}\right)^{\mathrm{T}}$, it is therefore enough to solve any of the six SMDs, since all six are exactly identical.

7.2. R-way CP

7.2.1. R-way extension

In this section we demonstrate how to extend the SECSI framework to the general R-way case for $R \geq 3$. As in the three-way case, we begin with the assumption that the CP decomposition is non-degenerate in all R modes and discuss the allowed degeneracies at the end of the section. Moreover, for clarity we assume the noise-free case $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0$ again. The influence of noise is discussed at the end of Section 6.2.1 and in Section 7.4. The "economy-size" HOSVD of the noise-free tensor $\boldsymbol{\mathcal{X}}$ satisfies $p_1 = p_2 = \ldots = p_R = d$ (which is replaced by the truncated HOSVD in the noisy case, cf. end of Section 6.2) and reads as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}]} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \bigtimes_{r=1}^{R} {}_r \boldsymbol{U}_r^{[\mathrm{s}]}, \tag{7.24}$$

phase term $\mathrm{e}^{\jmath\varphi}$ for each n-mode singular vector such that the symmetries hold.

where $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \in \mathbb{C}^{d \times d \ldots \times d}$ and $\boldsymbol{U}_r^{[\mathrm{s}]} \in \mathbb{C}^{M_r \times d}$. Again, we notice that the decomposition has a very similar structure to the R-way CP decomposition which is given by

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{I}}_{R,d} \times_1 \boldsymbol{F}^{(1)} \ldots \times_R \boldsymbol{F}^{(R)} = \boldsymbol{\mathcal{I}}_{R,d} \bigtimes_{r=1}^{R} {}_r \boldsymbol{F}^{(r)}. \tag{7.25}$$

By expressing the r-mode unfolding of $\boldsymbol{\mathcal{X}}$ in terms of (7.24) and (7.25) we find that $\boldsymbol{F}^{(r)}$ and $\boldsymbol{U}_r^{[\mathrm{s}]}$ span the same column space. Thus, there exists a non-singular transform matrix $\boldsymbol{T}_r \in \mathbb{C}^{d \times d}$ such that $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{T}_r$. Inserting this relation into (7.24) and (7.25) we find that the fundamental link between the R-way HOSVD and the R-way CP decomposition is given by

$$\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} = \boldsymbol{\mathcal{I}}_{R,d} \times_1 \boldsymbol{T}_1 \ldots \times_R \boldsymbol{T}_R \tag{7.26}$$

$$\boldsymbol{\mathcal{I}}_{R,d} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{T}_1^{-1} \ldots \times_R \boldsymbol{T}_R^{-1}. \tag{7.27}$$

Therefore, we can interpret the transform matrices $\boldsymbol{T}_r$ as the loading matrices of the CP decomposition of the core tensor (7.26) or equivalently as the matrices $\boldsymbol{T}_r^{-1}$ that diagonalize the core tensor (7.27).

As in the three-way case, we estimate the transform matrices via joint diagonalizations of a set of matrices. Again, instead of solving only one SMD we use the R-D structure to construct multiple SMDs and solve all of them, thus obtaining multiple estimates of all factors. We then pick the final solution in a subsequent step.

In [RH08a], we have presented this extension by operating on the slices of suitably constructed R-way tensors. Here we show a simpler, albeit algebraically equivalent, explanation of the R-way extension. To this end, define the tensors $\boldsymbol{\mathcal{X}}^{(k,\ell)}$ for $k, \ell = 1, 2, \ldots, R$, $k < \ell$ of size $M_k \times M_\ell \times M/(M_k \cdot M_\ell)$, where $M = \prod_{r=1}^{R} M_r$. The tensor $\boldsymbol{\mathcal{X}}^{(k,\ell)}$ contains the same elements as $\boldsymbol{\mathcal{X}}$ rearranged such that the k-th index appears in the first dimension, the ℓ-th index in the second dimension, and all other indices are stacked along the third dimension with indices arranged in increasing order. Consequently, if n is the third index of $\boldsymbol{\mathcal{X}}^{(k,\ell)}$, it is related to the indices $i_r = 1, 2, \ldots M_r$, $r = 1, 2, \ldots, R, r \neq k, \ell$ connected to $\boldsymbol{\mathcal{X}}$ in the following manner

$$\begin{aligned} n = {} & i_1 + (i_2 - 1) \cdot M_1 + \ldots + (i_{k-1} - 1) \cdot \prod_{r=1}^{k-1} M_r + \ldots + (i_{k+1} - 1) \cdot \prod_{\substack{r=1 \\ r \neq k}}^{k+1} M_r + \ldots \\ & + (i_{\ell-1} - 1) \cdot \prod_{\substack{r=1 \\ r \neq k}}^{\ell-1} M_r + \ldots + (i_{\ell+1} - 1) \cdot \prod_{\substack{r=1 \\ r \neq k,\ell}}^{\ell+1} M_r + (i_R - 1) \cdot \prod_{\substack{r=1 \\ r \neq k,\ell}}^{R} M_r \end{aligned} \tag{7.28}$$

for $n = 1, 2, \ldots, M/(M_k \cdot M_\ell)$. Since $\boldsymbol{\mathcal{X}}^{(k,\ell)}$ is a rearranged version of $\boldsymbol{\mathcal{X}}$ it can still be expressed as a sum of d rank-one tensors. Therefore, it obeys a rank-d CP decomposition given by

$$\boldsymbol{\mathcal{X}}^{(k,\ell)} = \boldsymbol{\mathcal{I}}_{R,d} \times_1 \boldsymbol{F}^{(k)} \times_2 \boldsymbol{F}^{(\ell)} \times_3 \boldsymbol{F}^{(\overline{k,\ell})}, \tag{7.29}$$

where $\boldsymbol{F}^{(\overline{k,\ell})} \in \mathbb{C}^{M/(M_k \cdot M_\ell) \times d}$ is given by

$$\boldsymbol{F}^{(\overline{k,\ell})} = \boldsymbol{F}^{(R)} \diamond \boldsymbol{F}^{(R-1)} \diamond \ldots \diamond \boldsymbol{F}^{(\ell+1)} \diamond \boldsymbol{F}^{(\ell-1)} \diamond \ldots \diamond \boldsymbol{F}^{(k+1)} \diamond \boldsymbol{F}^{(k-1)} \diamond \ldots \diamond \boldsymbol{F}^{(1)}. \tag{7.30}$$

Since $\boldsymbol{\mathcal{X}}^{(k,\ell)}$ obeys a three-way CP decomposition, the transform matrices $\boldsymbol{T}_k$ and $\boldsymbol{T}_\ell$ can be estimated by diagonalization of the three-mode slices of $\boldsymbol{\mathcal{X}}^{(k,\ell)}$, as described in the previous section. As before, the "asymmetric" SMDs are converted into diagonalization by similarity transformation via the multiplication with the inverse of a pivot slice from the right hand side (rhs) or the left hand side (lhs), giving rise to two SMDs. We obtain one estimate for $\boldsymbol{T}_k$ from the SMD rhs, one estimate for $\boldsymbol{T}_\ell$ from the SMD lhs and one estimate for $\tilde{\boldsymbol{F}}^{(\overline{k,\ell})}$ from both rhs and lhs. In order to obtain estimates for all loading matrices we need to "invert" (7.30), i.e., find the loading matrices $\boldsymbol{F}^{(r)}$, $r = 1, 2, \ldots, R, r \neq k, \ell$ that satisfy (7.30). In the presence of noise, where (7.30) holds only approximately, this is achieved approximately via what we refer to as Least Squares Khatri-Rao Factorization. Such a factorization can be achieved algebraically, i.e., only with the help of (Higher Order) SVDs, as we show in the following.

To this end, let $\boldsymbol{f}_n^{(\overline{k,\ell})}$ be the n-th column of $\boldsymbol{F}^{(\overline{k,\ell})}$ for $n = 1, 2, \ldots, d$. Since $\boldsymbol{F}^{(\overline{k,\ell})}$ contains a column-wise Kronecker product, $\boldsymbol{f}_n^{(\overline{k,\ell})}$ is the Kronecker product between the n-th columns of the loading matrices $\boldsymbol{F}^{(r)}$, $r = 1, 2, \ldots, R, r \neq k, \ell$. Rearranging this vector into an $(R-2)$-dimensional tensor $\boldsymbol{\mathcal{F}}_n^{(\overline{k,\ell})}$ yields the rank-one tensor

$$\boldsymbol{\mathcal{F}}_n^{(\overline{k,\ell})} = \boldsymbol{f}_n^{(R)} \circ \boldsymbol{f}_n^{(R-1)} \ldots \circ \boldsymbol{f}_n^{(\ell+1)} \circ \boldsymbol{f}_n^{(\ell-1)} \boldsymbol{f}_n^{(k+1)} \circ \boldsymbol{f}_n^{(k-1)} \ldots \circ \boldsymbol{f}_n^{(2)} \circ \boldsymbol{f}_n^{(1)}. \tag{7.31}$$

In the presence of noise, this relation holds only approximately. Consequently, estimating the n-th column of the loading matrices is achieved by determining a rank-one approximation of the tensor $\boldsymbol{\mathcal{F}}_n^{(\overline{k,\ell})}$. While the optimal rank-one approximation requires an iterative algorithm [dLdMV00b], the truncated HOSVD already provides an estimate that is sufficiently accurate for all practical purposes[5]. By repeating this process for all columns $n = 1, 2, \ldots, d$, we can estimate the loading matrices $\boldsymbol{F}^{(r)}$, $r = 1, 2, \ldots, R, r \neq k, \ell$ up to one unknown scaling ambiguity per column. These are irrelevant since they are already inherent in the CP

[5] For $R = 4$, the tensor $\boldsymbol{\mathcal{F}}_n^{(\overline{k,\ell})}$ has only two dimensions. Therefore, a truncated SVD yields the optimal rank-one approximation in the Frobenius-norm sense.

decomposition itself (cf. discussion in Section 6.2).

As in the three-way case, from both SMDs (rhs, lhs) we have obtained all loading matrices except for one ($\boldsymbol{F}^{(k)}$ for rhs and $\boldsymbol{F}^{(\ell)}$ for lhs), which may be estimated via a Least Squares fit in the very last step[6]. Consequently, for each (k, ℓ) we obtain two estimates for all loading matrices $\boldsymbol{F}^{(r)}, r = 1, 2, \ldots, R$. In the non-degenerate case, the number of possible combinations for k and ℓ is equal to $R \cdot (R-1)/2$ which provides us with up to $R \cdot (R-1)$ estimates for all loading matrices in the general R-way case. This shows that with larger R our SECSI framework generates even more candidate solutions and thus allows even more flexibility in designing efficient algorithms. Table 7.1 exemplifies the possible SMDs and the corresponding estimates for the loading matrices in a non-degenerate $R = 5$-way CP decomposition.

7.2.2. Degeneracies

If the CP decomposition is degenerate in mode r, i.e., $p_r < d$, then the transform matrix $\boldsymbol{T}_r$ does not exist anymore. Consequently, all SMDs (k, ℓ) where either k or ℓ is equal to r cannot be solved anymore. However, as long as there is at least one combination (k, ℓ) where the CP decomposition is non-degenerate in both k and ℓ, we can still obtain two estimates for all loading matrices by solving the two SMDs (rhs, lhs) connected to these modes (k, ℓ). Therefore, the SECSI framework is applicable if the problem is degenerate in up to $R-2$ modes, it breaks down if degeneracies in $R-1$ or in all R modes occur.

7.2.3. Symmetries

The comments about symmetric CP decompositions from Section 7.1.4 also generalize to the R-way case. In particular, in the case of a pair-wise symmetric tensor where all R loading matrices are equal, all $R \cdot (R-1)$ SMDs are identical and therefore it is sufficient to solve exactly one of them.

7.3. Special cases

In this section we discuss the two special cases of a “two-slab” and a two-component CP. Here, “two-slab” refers to the case that the tensor consists of only two slices, i.e., its size is equal to two in one of its modes. It is known that in these cases a direct algebraic solution is possible. As we show, our semi-algebraic framework naturally reduces to these known algebraic

[6]As in the three-way case, this step can be skipped for SMDs that are discarded by a an appropriate heuristic, e.g., RES.

(k, ℓ)	r/l	$\boldsymbol{F}^{(1)}$	$\boldsymbol{F}^{(2)}$	$\boldsymbol{F}^{(3)}$	$\boldsymbol{F}^{(4)}$	$\boldsymbol{F}^{(5)}$
$(1, 2)$	rhs	T	L	D	D	D
$(1, 2)$	lhs	L	T	D	D	D
$(1, 3)$	rhs	T	D	L	D	D
$(1, 3)$	lhs	L	D	T	D	D
$(1, 4)$	rhs	T	D	D	L	D
$(1, 4)$	lhs	L	D	D	T	D
$(1, 5)$	rhs	T	D	D	D	L
$(1, 5)$	lhs	L	D	D	D	T
$(2, 3)$	rhs	D	T	L	D	D
$(2, 3)$	lhs	D	L	T	D	D
$(2, 4)$	rhs	D	T	D	L	D
$(2, 4)$	lhs	D	L	D	T	D
$(2, 5)$	rhs	D	T	D	D	L
$(2, 5)$	lhs	D	L	D	D	T
$(3, 4)$	rhs	D	D	T	L	D
$(3, 4)$	lhs	D	D	L	T	D
$(3, 5)$	rhs	D	D	T	D	L
$(3, 5)$	lhs	D	D	L	D	T
$(4, 5)$	rhs	D	D	D	T	L
$(4, 5)$	lhs	D	D	D	L	T

Table 7.1.: Possible SMDs for a non-degenerate $R = 5$-way CP showing which factors are estimates from the transform matrices $\boldsymbol{T}_r$ ("T"), from the diagonalized matrices via $\boldsymbol{F}^{(\overline{k,\ell})}$ ("D"), and via the LS fit ("L"). The column "r/l" refers to the two SMDs rhs, lhs that can be constructed for each combination of modes, cf. (7.10).

solutions, i.e., there is no need to solve any SMD. Since the general R-way case can always be reduced to a 3-way tensor (as shown in Section 7.2), we only consider the 3-way case.

7.3.1. Two slabs

First of all, consider the case where the tensor has size two in one mode, i.e., $\boldsymbol{\mathcal{X}}$ is of size $M_1 \times M_2 \times 2$, without loss of generality. In other words, $\boldsymbol{\mathcal{X}}$ consists of two three-mode matrices $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathbb{C}^{M_1 \times M_2}$, such that $\boldsymbol{\mathcal{X}} = [\boldsymbol{X}_1 \sqcup_3 \boldsymbol{X}_2]$. If we estimate the CP via joint diagonalization of both three-mode slices we need $d \leq \min(M_1, M_2)$ but d may be larger than two. The two three-mode slices of $\boldsymbol{\mathcal{X}}$ satisfy

$$\boldsymbol{X}_1 = \boldsymbol{F}^{(1)} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(1,:)}\right\} \cdot \boldsymbol{F}^{(2)^{\mathrm{T}}} \tag{7.32}$$

$$\boldsymbol{X}_2 = \boldsymbol{F}^{(1)} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(2,:)}\right\} \cdot \boldsymbol{F}^{(2)^{\mathrm{T}}}. \tag{7.33}$$

The three-mode slices $\boldsymbol{S}_{3,k}$ are formed by multiplying $\boldsymbol{\mathcal{X}}$ with $\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}}$ from the truncated HOSVD in the first $(r = 1)$ and the second mode $(r = 2)$ which gives

$$\boldsymbol{S}_{3,1} = \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{X}_1 \cdot \boldsymbol{U}_2^{[\mathrm{s}]^*} = \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(1,:)}\right\} \cdot \boldsymbol{T}_2^{\mathrm{T}} \tag{7.34}$$

$$\boldsymbol{S}_{3,2} = \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{X}_2 \cdot \boldsymbol{U}_2^{[\mathrm{s}]^*} = \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(2,:)}\right\} \cdot \boldsymbol{T}_2^{\mathrm{T}}. \tag{7.35}$$

Therefore, for $\boldsymbol{S}_{3,k}^{\mathrm{rhs}}$ and $\boldsymbol{S}_{3,k}^{\mathrm{lhs}}$ we only have one slice which yields

$$\boldsymbol{S}_{3,1}^{\mathrm{rhs}} = \boldsymbol{T}_1 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(1,:)}\right\} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(2,:)}\right\}^{-1} \cdot \boldsymbol{T}_1^{-1}$$

$$\boldsymbol{S}_{3,1}^{\mathrm{lhs}} = \boldsymbol{T}_2 \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(2,:)}\right\} \cdot \operatorname{diag}\left\{\left[\boldsymbol{F}^{(3)}\right]_{(1,:)}\right\}^{-1} \cdot \boldsymbol{T}_2^{-1}.$$

This shows that $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ can be directly estimated as the matrix of eigenvectors of $\boldsymbol{S}_{3,k}^{\mathrm{rhs}}$ and $\boldsymbol{S}_{3,k}^{\mathrm{lhs}}$, respectively. Albeit computed in a slightly different manner, this procedure is algebraically equivalent to the Direct Trilinear Decomposition (DTLD) which was proposed in [SK90]. The main difference is that DTLD was devised for three-way tensors of an arbitrary size. The reduction to two slabs is achieved by performing a three-mode projection onto the dominant two three-mode singular vectors. Since for two slabs $\boldsymbol{U}_3^{[\mathrm{s}]}$ is of size 2×2, this projection has no impact on the estimation accuracy of the CP model.

Overall, this shows that the SECSI framework presented in this chapter and reduces to the known algebraic DTLD approach for the special case of two slabs. Hence, there is no need to solve any SMDs in this case.

7.3.2. Two components

In this subsection we discuss the special case where the rank of the tensor $\boldsymbol{\mathcal{X}}_0$ is equal to two, i.e., the CP model contains the sum of $d = 2$ rank-one components. Moreover, the size of the tensor is arbitrary, i.e., $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$. Note that in this case, the CP can never be degenerate, since rank-one loading matrices are not allowed in Kruskal's identifiability condition [Kru77]. Consider the noise-free case first. We have

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{I}}_{3,2} \times_1 \boldsymbol{F}^{(1)} \times_2 \boldsymbol{F}^{(2)} \times_3 \boldsymbol{F}^{(3)} \tag{7.36}$$

$$= \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \times_2 \boldsymbol{U}_2^{[\mathrm{s}]} \times_3 \boldsymbol{U}_3^{[\mathrm{s}]}, \tag{7.37}$$

where $\boldsymbol{F}^{(r)}$ and $\boldsymbol{U}_r^{[\mathrm{s}]}$ are of size $M_r \times 2$ and we have $\boldsymbol{F}^{(r)} = \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{T}_r$ for $r = 1, 2, 3$. Consequently, the CP of $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \in \mathbb{C}^{2 \times 2 \times 2}$ can be expressed as

$$\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} = \boldsymbol{\mathcal{I}}_{3,2} \times_1 \boldsymbol{T}_1 \times_2 \boldsymbol{T}_2 \times_3 \boldsymbol{T}_3, \tag{7.38}$$

where $\boldsymbol{T}_r \in \mathbb{C}^{2 \times 2}$ for $r = 1, 2, 3$. However, this is a CP with two slabs so the direct solution discussed in Section 7.3.1 applies. Since $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$ is also of size two in the first and the second mode it is even simpler: another Tucker compression step in the first two modes is not needed. Instead we can directly compute the eigenvectors of $\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}$ which yields $\boldsymbol{T}_1$ and the eigenvectors of $\left(\boldsymbol{S}_1^{-1} \cdot \boldsymbol{S}_2\right)^{\mathrm{T}}$ which yields $\boldsymbol{T}_2$, if $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are the two three-mode slabs of $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$.

However, there is a catch when the process is applied to noisy data. When the tensor is real-valued, $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$ is also real-valued, but $\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}$ may have complex eigenvectors. This happens when $\operatorname{trace}\left\{\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}\right\}^2 < 4 \cdot \det\left\{\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}\right\}$. The interpretation of this scenario is that the core tensor $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$ has rank three over $\mathbb{R}$, which may happen due to the noise influence. Note that a $2 \times 2 \times 2$ tensor has generic rank[7] two over $\mathbb{C}$ but no generic rank over $\mathbb{R}$, since it possesses the two typical ranks two and three. This was already observed in [Ten91]. Additionally, the proof presented in [Ten91] closely resembles the method for computing the CP of the core tensor that is discussed in this section.

For the estimation of the CP model this implies that in the real-valued case, there are

[7]The *typical* rank is defined as any tensor rank that occurs with non-zero probability for tensors drawn from a continuous distribution. For tensors which have only one typical rank, this rank is called *generic* [CtBdLC09].

scenarios where this direct fit fails to provide a real-valued model. However, this only happens when the noise contribution is so strong that it dominates at least one of the two components. Consequently, this can be seen as a reliability test. If the two slabs of the core tensor satisfy the condition $\operatorname{trace}\left\{\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}\right\}^2 < 4 \cdot \det\left\{\boldsymbol{S}_1 \cdot \boldsymbol{S}_2^{-1}\right\}$ the reliability test has failed and we know that a rank-two model cannot be fitted reliably. As we show in the simulations, this direct fit performs just equally well as an ALS-based PARAFAC while being completely non-iterative.

What does this mean for the SMDs constructed via our SECSI framework? It is easy to show that as long as the core tensor is rank-two, the direct fit presented above represents an exact solution to all SMDs. Therefore, there is no need to compute any of the SMDs, since they converge to the same solution that can be directly obtained via the direct fit above.

In summary, this shows that for two components, iterative procedures are never required and our SECSI framework reduces to the direct fitting algorithm that was explained as part of the constructive proof in [Ten91].

7.4. Heuristics

In the previous sections we have demonstrated how the CP model can be reformulated into Simultaneous Matrix Decompositions. We have exploited the structure to obtain multiple estimates for the loading matrices $\boldsymbol{F}^{(r)}, r = 1, 2, \ldots, R$. While in the noise-free case, each SMD yields the exact loading matrices (up to permutation and scaling), in the presence of noise, the estimates can be different. This allows us to select one of these estimates for each loading matrix that we consider to be the best choice. We therefore need to devise methods to select which of the candidate solutions to output as the final estimate. With the help of these selection criteria, different algorithms to solve the approximate CP decomposition can be defined within the SECSI framework presented in this chapter. If a very fast algorithm is needed we can choose to solve only one SMD which provides already a full estimate of the model. However, if the accuracy of the solution is critical, all SMDs can be solved and an exhaustive best matching scheme can be used to minimize the reconstruction error.

To this end, let $\hat{\boldsymbol{F}}_n^{(r)}$ be the estimate of $\boldsymbol{F}^{(r)}$ from the n-th SMD, where $n = 1, 2, \ldots, N_{\mathrm{SMD}}$. The overall number of SMDs that are solved is denoted by N_{SMD}. If the CP decomposition is completely non-degenerate, we can choose to solve between 1 and $R \cdot (R-1)$ SMDs, i.e., $1 \leq N_{\mathrm{SMD}} \leq R(R-1)$. If degeneracies occur, the maximum is lower, for a maximally degenerate problem we have $1 \leq N_{\mathrm{SMD}} \leq 2$ (cf. discussions in Sections 7.1.2 and 7.2.2).

In principle, it is possible to combine estimates from different SMDs, say, pick $\hat{\boldsymbol{F}}^{(1)}$ and $\hat{\boldsymbol{F}}^{(3)}$ from the first SMD and $\hat{\boldsymbol{F}}^{(2)}$ from the second. However, in this case, special attention must be paid to the scaling and permutation ambiguities mentioned in Section 6.2. Within one SMD these indeterminacies are "consistent", meaning they have no impact on the reconstructed tensor $\hat{\boldsymbol{\mathcal{X}}}$. However, combining estimates from different SMDs can lead to inconsistent permutation and scaling, which results in an invalid model. Therefore, these ambiguities must be resolved first. A simple approach is to normalize all estimates of all factors such that each column has norm one and to estimate the corresponding amplitudes via another Least Squares fit, i.e.,[8]

$$\boldsymbol{\gamma} = \left(\bar{\boldsymbol{F}}^{(R)} \diamond \bar{\boldsymbol{F}}^{(R-1)} \ldots \diamond \bar{\boldsymbol{F}}^{(1)}\right)^{+} \cdot \operatorname{vec}\left\{\boldsymbol{X}\right\}, \tag{7.39}$$

where $\bar{\boldsymbol{F}}^{(r)}$ represents the normalized loading matrix in the r-th mode. These amplitudes $\gamma_i, i = 1, 2, \ldots, d$ have been introduced under the name "Least Squares PARAFAC amplitudes" in [WRH⁺09]. The permutation ambiguity is then solved by permuting the columns of every estimate $\hat{\boldsymbol{F}}_n^{(r)}$ such that the magnitudes of the amplitudes γ_i appear in descending order. [9] When reconstructing a tensor by combining estimates from different SMDs we first compute an LS fit of the corresponding amplitudes $\boldsymbol{\gamma}$ and then rescale the components accordingly, i.e., by placing γ_i on the main diagonal of the identity tensor or equivalently by scaling the columns of the loading matrix in one of the R modes by γ_i.

We use the following measures for building selection criteria:

- Reconstruction error (REC): The reconstruction error is defined as

$$\operatorname{REC}(n_1, n_2, \ldots, n_R) = \frac{\left\| \boldsymbol{\mathcal{I}}_{R,d} \bigtimes_{r=1}^{R}{}_r \hat{\boldsymbol{F}}_{n_r}^{(r)} - \boldsymbol{\mathcal{X}} \right\|_{\mathrm{H}}^2}{\left\|\boldsymbol{\mathcal{X}}\right\|_{\mathrm{H}}^2},$$

  for $n_r = 1, 2, \ldots, N_{\mathrm{SMD}}$, i.e., the relative error between the tensor reconstructed from the estimated loading matrices and the observed (noisy) data tensor.

- SMD residuals (RES): The residual after the completion of the n-th SMD measures the "diagonality" of the slices after joint diagonalization. Let $\boldsymbol{S}_{n,k}$ be the input slices to the n-th SMD, where $k = 1, 2, \ldots, N_{\text{slices}}$ and let $\boldsymbol{T}$ be the resulting diagonalizer, such that

[8] Obviously, this step is not needed for estimates originating from the *same* SMD since in this case, the last loading matrix is estimated via LS, which already includes this LS fit for the amplitudes. Therefore, we only use it when combining estimates from *different* SMDs.

[9] Alternatively, the permutation ambiguity can be resolved by comparing the angles between the columns of $\hat{\boldsymbol{F}}_n^{(r)}$ and $\hat{\boldsymbol{F}}_1^{(r)}$. At each step the two columns with the closest match are associated and then removed from the set of remaining columns. However, we found the approach based on (7.39) to be more reliable in our numerical experiments.

$\boldsymbol{T}^{-1} \cdot \boldsymbol{S}_{n,k} \cdot \boldsymbol{T}$ is approximately diagonal for all k. Then, the residual is defined as

$$\mathrm{RES}(n) = \frac{1}{N_{\mathrm{slices}}} \sum_{k=1}^{N_{\mathrm{slices}}} \left\| \mathrm{off}\left\{ \boldsymbol{T}^{-1} \cdot \boldsymbol{S}_{n,k} \cdot \boldsymbol{T} \right\} \right\|_{\mathrm{F}}^{2}, \tag{7.40}$$

where off $\{\cdot\}$ is a function that extracts all off-diagonal elements, i.e., off $\{\boldsymbol{X}\} = \boldsymbol{X} - (\boldsymbol{X} \odot \boldsymbol{I}_d)$ for $\boldsymbol{X} \in \mathbb{C}^{d \times d}$.

- Condition (CON): This measure evaluates how well-conditioned the SMDs connected to the modes (k, ℓ) are by storing the best conditioning number among the slices that was found when selecting the pivot slice. Consequently, if $\boldsymbol{S}_i^{(k,\ell)}$ represent the slices from which the SMDs connected to modes k and ℓ are constructed, the corresponding conditioning measure is defined as

  $$\mathrm{CON}(k, \ell) = \min_{i=1,2,\dots,N_{\mathrm{slices}}} \mathrm{cond}\{\boldsymbol{S}_i^{(k,\ell)}\}, \ 1 \le k < \ell \le R. \tag{7.41}$$

  It is expected that an SMD where all slices are badly conditioned (and hence $\mathrm{CON}(k, \ell)$ is large) is more difficult to solve and hence more sensitive to additive noise compared to an SMD with well conditioned slices. Therefore, while RES and REC only allow to assess the quality of the estimates after the SMD was computed, CON can be used to select/discard SMDs prior to computing them.

  In fact, it can be shown that a bad conditioning of the r-th loading matrix $\boldsymbol{F}^{(r)}$ leads to badly conditioned matrix slices for all (k, ℓ) where either $k = r$ or $\ell = r$. Consequently, if the model has up to $R - 2$ badly conditioned and two well-conditioned loading matrices, choosing the latter two modes as k and ℓ yields well-conditioned matrix slices for the SMD which promises faster convergence and hence a better estimate.

Various combinations of the above criteria as well as other measures are possible to devise specific algorithms for the computation of an approximate CP decomposition within our SECSI framework. Depending on the allowed computational complexity we can choose to solve only one or many SMDs and then select the final estimate for the loading matrices according to different criteria. The specific algorithms we propose are the following:

- **BM** (Best Matching): Solve all possible SMDs (up to $R \cdot (R - 1)$ if no degeneracies occur) and select the final solution following the REC-criterion, testing all possible combinations in an exhaustive search. In other words, the final loading matrices are given by $\hat{\boldsymbol{F}}_{i_r}^{(r)}$,

$r = 1, 2, \ldots, R$, where

$$(i_1, i_2, \ldots, i_R) = \underset{n_1, n_2, \ldots, n_R}{\arg\min} \ \mathrm{REC}(n_1, n_2, \ldots, n_R). \tag{7.42}$$

This algorithm provides the minimum possible reconstruction error in our SECSI framework. However, its complexity is comparably high. Since different SMDs are combined, we have to resolve scaling and permutation ambiguities (as discussed above). Moreover, the exhaustive search requires to reconstruct $(R \cdot (R-1))^R$ combinations of loading matrices, which grows rapidly with R (729 for $R = 3$, 20 736 for $R = 4$, and already 3.2 million for $R = 5$). In any case, BM can serve as a reference to determine the achievable reconstruction error in the SECSI framework in order to benchmark different heuristics.

- **REC PS**: Similarly to **BM**, the REC criterion is used with the difference that only solutions originating from the same SMD are combined, hence **PS** ("paired solutions"). The final estimates are given by $\hat{\boldsymbol{F}}_i^{(r)}$, $r = 1, 2, \ldots, R$, where

$$i = \underset{n}{\arg\min} \, \mathrm{REC}(n, n, \ldots, n). \tag{7.43}$$

  This is a much more practical algorithm since its complexity is significantly lower. There is no need to resolve permutation and scaling ambiguities between estimates and we only have to reconstruct $N_{\mathrm{SMD}} \leq R \cdot (R-1)$ tensors.

- **RES**: This algorithm also solves all SMDs and then selects the final estimates from the SMD which minimizes the RES criterion, i.e., the smallest residuals after the diagonalization of the corresponding matrix slices are chosen. In this method there is no need to compute a reconstructed tensor, which is advantageous if the dimensions of the tensor are very large.

- **CON PS**: In this algorithm we solve only the two SMDs which correspond to the lowest value in the CON-criterion. Afterwards, the two solutions are compared in terms of their reconstruction error and the one which yields the lower reconstruction error is returned as the final solution. No mutual combinations between the estimates from the two SMDs are considered (PS = "paired solutions").

  This algorithm has a very low complexity since only two SMDs are solved and only two tensor reconstructions are computed.

It should be emphasized again that these choices only represent examples of algorithms that can be devised within the SECSI framework presented in this chapter. The generic way in which it is formulated allows to control the trade-off between performance and complexity. Furthermore, for a specific application at hand, an algorithm can be customized to the specific requirements of the given scenario.

7.5. Simulation results

In this section we present numerical simulation results to demonstrate the accuracy and the computational complexity of our SECSI framework. If not stated otherwise, the loading matrices for the CP decomposition are drawn from a $\mathcal{N}(0,1)$ standard Gaussian distribution for the real-valued case and a $\mathcal{CN}(0,1)$ complex Gaussian distribution for the complex-valued case. The observed tensor $\boldsymbol{\mathcal{X}}$ is generated according to (6.1), i.e., by adding a noise tensor $\boldsymbol{\mathcal{N}}$ which contains zero mean i.i.d. circularly symmetric complex Gaussian random variables with identical variance σ_n^2. The SNR is then defined as $\text{SNR} = \sigma_n^{-2}$.

We evaluate the accuracy of an estimated CP model $\hat{\boldsymbol{F}}_1, \hat{\boldsymbol{F}}_2, \ldots, \hat{\boldsymbol{F}}_R$ with two different measures, the relative Mean Square Reconstruction Error (MSRE) and the relative Total Mean Squared Factor Error (TMSFE). The MSRE is defined as

$$\text{MSRE} = \mathbb{E}\left\{\frac{\left\|\hat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}\right\|_{\text{H}}^2}{\left\|\boldsymbol{\mathcal{X}}\right\|_{\text{H}}^2}\right\}, \tag{7.44}$$

where the reconstructed tensor $\hat{\boldsymbol{\mathcal{X}}}$ is given by $\hat{\boldsymbol{\mathcal{X}}} = \boldsymbol{\mathcal{I}}_{R,d} \mathop{\times}_{r=1}^{R} {}_r \hat{\boldsymbol{F}}^{(r)}$. It models how well the estimated CP model can *approximate* the observed noisy tensor. On the other hand, the TMSFE is defined via

$$\text{TMSFE} = \mathbb{E}\left\{\sum_{r=1}^{R} \min_{\boldsymbol{P}_r \in \mathcal{M}_{\text{PD}}(d)} \frac{\left\|\hat{\boldsymbol{F}}^{(r)} \cdot \boldsymbol{P}_r - \boldsymbol{F}^{(r)}\right\|_{\text{F}}^2}{\left\|\boldsymbol{F}^{(r)}\right\|_{\text{F}}^2}\right\}, \tag{7.45}$$

where $\mathcal{M}_{\text{PD}}(d)$ is the set of $d \times d$ permuted diagonal matrices (also called monomial matrices), i.e., the matrices $\boldsymbol{P}_r$ correct the permutation and scaling ambiguity that is inherent in the estimation of the loading matrices. Consequently, the TMSFE measures how accurately the *actual* CP model can be *estimated* from the noisy observations. Depending on the application either MSRE or TMSFE may be the more appropriate figure of merit and hence it is instructive to look at both.

If not stated otherwise we use the algorithm proposed in [FG06] for the diagonalization by

Scen.	Dimension , Rank	$\mathbb{R}/\mathbb{C}$	Correlation	SNR	Trials (approx.)
I	$4 \times 7 \times 6$, $d = 3$	$\mathbb{R}$	(none)	20 dB	1 500 000
II	$4 \times 8 \times 7$, $d = 3$	$\mathbb{R}$	mode 1	40 dB	850 000
III	$4 \times 8 \times 7$, $d = 3$	$\mathbb{C}$	mode 1	40 dB	550 000
IV	$4 \times 9 \times 6 \times 4$, $d = 5$	$\mathbb{C}$	(none)	30 dB	500 000
V	$4 \times 7 \times 15 \times 6$, $d = 3$	$\mathbb{C}$	mode 1, 2	35 dB	10 000
VI	$80 \times 80 \times 80$, $d = 4$	$\mathbb{R}$	mode 1	20 dB	5 000

Table 7.2.: Simulation settings for scenarios investigated in Section 7.5.1.

Scen.	ALS	CPOPT	BM	REC PS	RES	CON PS
I	113 ms	217 ms	175 ms	27 ms	25 ms	10 ms
II	284 ms	457 ms	199 ms	40 ms	38 ms	12 ms
III	992 ms	N/A	287 ms	89 ms	86 ms	21 ms
IV	338 ms	N/A	243 ms	200 ms	198 ms	200 ms
V	33.39 s	N/A	52.43 s	1.48 s	1.47 s	88 ms
VI	0.51 s	5.02 s	29.50 s	0.59 s	0.52 s	0.29 s

Table 7.3.: Mean run time for different algorithms in all scenarios.

similarity transform.

7.5.1. Three-way and Four-way CP decomposition

In this section we present some results on three- and four-way tensors without any symmetries. Our goal is to evaluate the achievable accuracy within our SECSI framework and to investigate the accuracy-complexity trade-off of different heuristics, i.e., how much faster are they compared to the benchmark given by the exhaustive best matching scheme and how much accuracy is lost. In addition to the heuristics introduced in Section 7.4 we depict a “dummy” heuristic where only a single SMD is solved (always selecting the first). The comparison to this dummy heuristic enables us to see how much we gain by a smart selection of appropriate SMDs. As an additional benchmark we also depict a “genie” heuristic where for each loading matrix the best estimate is selected from all candidate solutions. This guarantees the lowest possible TMSFE inside the SECSI framework (but not necessarily the lowest possible MSRE).

To get further insight into these aspects under various settings we consider six different scenarios. The corresponding simulation settings are listed in Table 7.2. Figures 7.2-7.8 depict the complementary cumulative density function (CCDF) of either MSRE or TMSFE for each

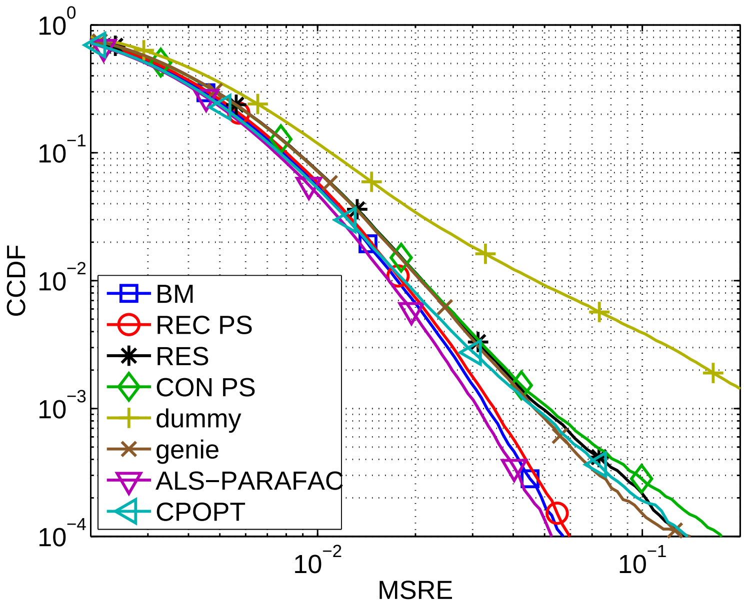


Figure 7.2.: CCDF of the MSRE for Scenario I.

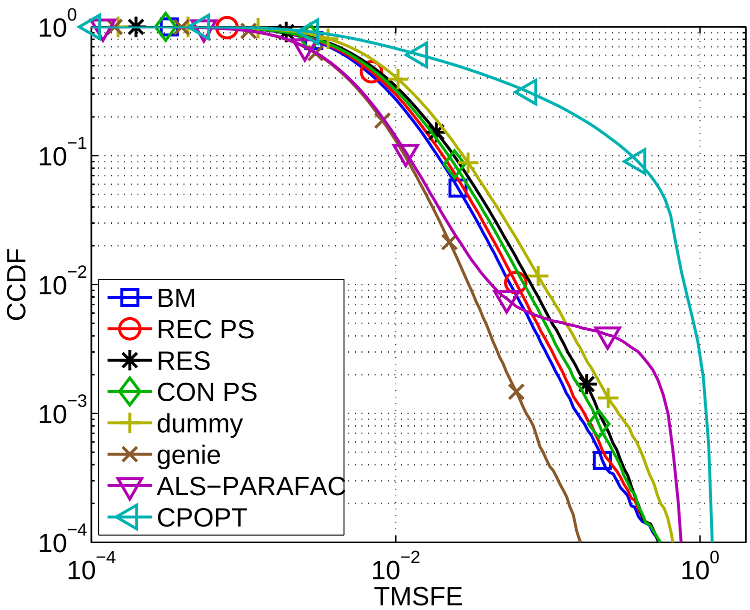


Figure 7.3.: CCDF of the TMSFE for Scenario II.

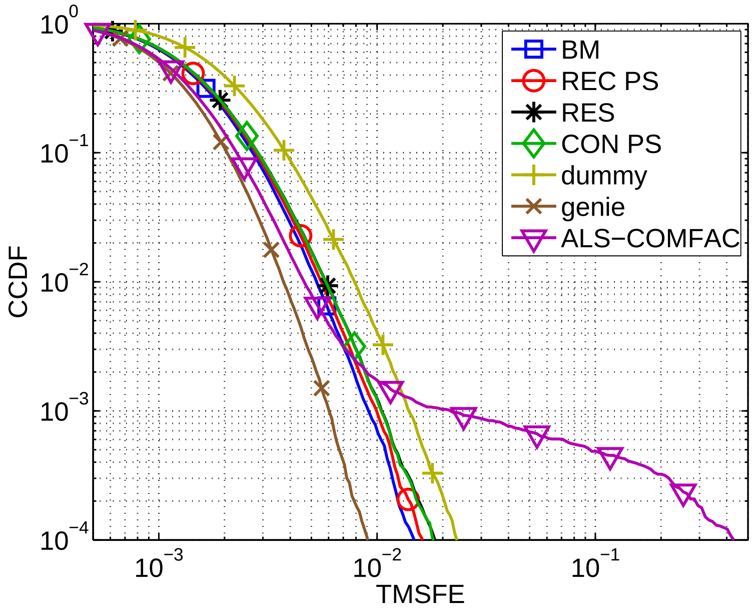


Figure 7.4.: CCDF of the TMSFE for Scenario III.

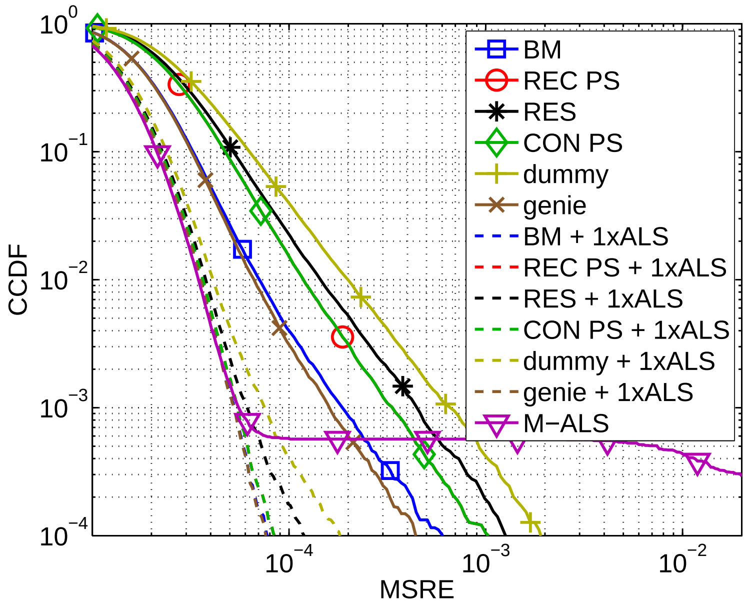


Figure 7.5.: CCDF of the MSRE for Scenario IV. The dashed curves result from running a single ALS iteration starting with the SECSI solutions.

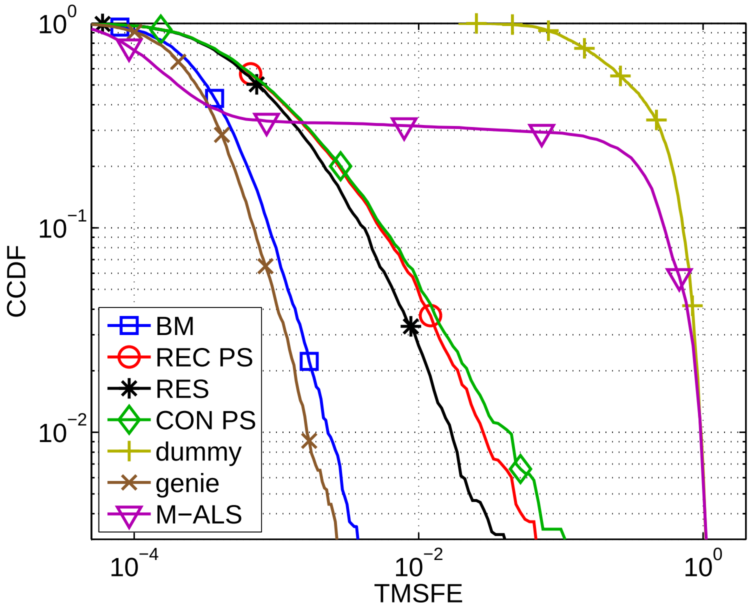


Figure 7.6.: CCDF of the TMSFE for Scenario V.

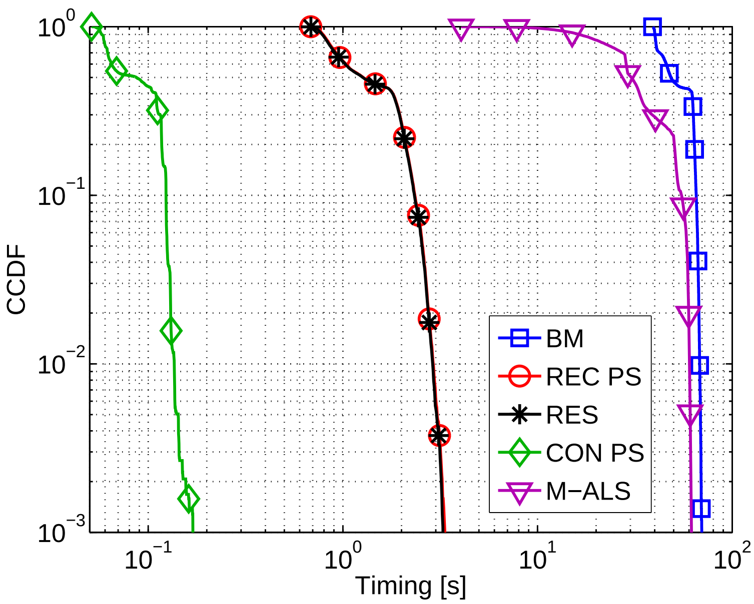


Figure 7.7.: CCDF of the run times for Scenario V.

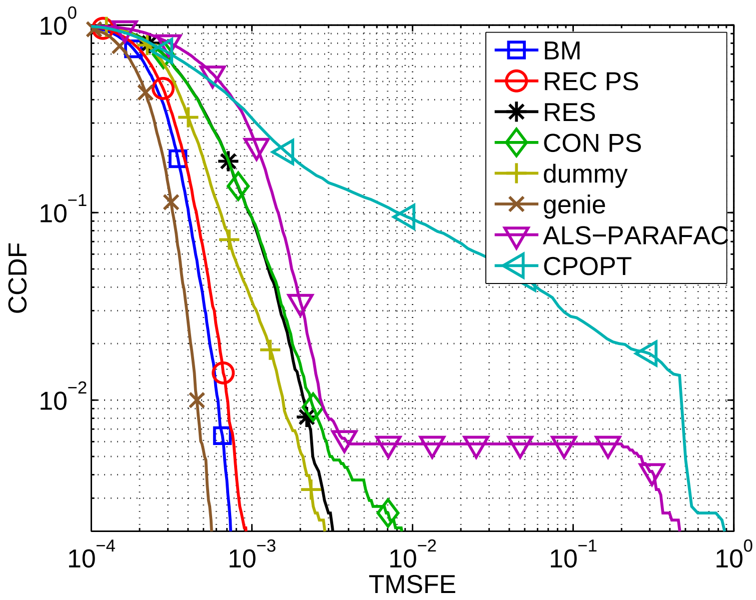


Figure 7.8.: CCDF of the TMSFE for Scenario VI.

scenario and Table 7.3 shows the average run time of the algorithms[10].

In the first scenario we investigate a real-valued CP decomposition with randomly drawn loading matrices. The corresponding CCDF of the MSRE is shown in Figure 7.2. We compare the four algorithms introduced in Section 7.4 with the ALS-PARAFAC algorithm from the N-way toolbox [AB00] (which contains the enhanced line search procedure)[11] and the CPOPT algorithm from the MATLAB Tensor Toolbox [ADK11]. We observe that in such a case, ALS-PARAFAC performs best and the SECSI algorithms BM and REC PS are very close. Moreover, while BM is slower, REC PS is already faster than ALS-PARAFAC (cf. Table 7.3). The even faster algorithms RES and CON PS perform a bit worse, albeit still comparable to the CPOPT algorithm. Furthermore, all heuristics outperform the "dummy" heuristic where only a single SMD is solved quite significantly. Essentially this shows the "diversity advantage" which is obtained if we select the final estimate among several candidate solutions. Note that the genie heuristic does not provide the lowest MSRE since it selects the estimates according to the estimation errors for the individual loading matrices which does not guarantee a low MSRE.

[10] Run times have been evaluated on an Intel X 5550 2.67 GHz machine running Linux (Red Hat 3.4.6-9, Kernel version 2.6.9-67.0.4.ELsmp) and MATLAB R2012a (7.14.0.739) 64-bit. While absolute numbers will vary greatly from machine to machine, the relative comparison should remain about the same.

[11] Concerning the stopping criteria and the initializations, the default setting of the public version of the N-way toolbox are used.

For the second scenario we modify the previous experiment by choosing the loading matrix in the first mode according to

$$\boldsymbol{F}^{(1)} = \begin{bmatrix} 1.00 & 1.00 & 1.00 \\ 1.00 & 0.95 & 0.95 \\ 1.00 & 0.95 & 1.00 \\ 1.00 & 1.00 & 0.95 \end{bmatrix}, \tag{7.46}$$

whereas $\boldsymbol{F}^{(2)}$ and $\boldsymbol{F}^{(3)}$ are chosen randomly. Since $\mathrm{cond}(\boldsymbol{F}^{(1)}) \approx 116$, the tensor is badly conditioned in the first mode but well-conditioned in the second and the third mode. Such scenarios, often referred to as bottlenecks [CLdA09], are a severe challenge to ALS-based algorithms since the correlation may stall the convergence of the ALS iterations. Moreover, as highlighted earlier, in such a case the SMDs that involve $\boldsymbol{T}_1$ (i.e., the joint diagonalization of two- and three-mode slices) are badly conditioned, however, the SMD obtained from the diagonalization of one-mode slices is unaffected. This situation is automatically detected by the CON heuristic so that we expect it to perform well in this scenario. Figure 7.3 shows the TMSFE for this scenario. We observe that while the ALS-PARAFAC algorithm yields a lower TMSFE in some cases, it has outliers with a much higher TMSFE. The algorithms within our SECSI framework are more robust compared to ALS in the sense that the variation of the TMSFE is much lower and no outliers are observed. They are rather close to each other: BM performs best, closely followed by the heuristics REC PS, then CON PS, then RES, and finally the dummy heuristic. However, the difference between the TMSFEs is only about 0.5 dB with an improvement compared to the dummy heuristic of about 1 dB. The TMSFE of the genie heuristic is almost 3 dB lower than BM (and always better than ALS-PARAFAC), suggesting that there is still room for improvement for finding better heuristics. Considering the corresponding run time of the algorithms we observe that CON PS is around 25 times faster than the ALS-based PARAFAC. Consequently, the CON PS heuristic yields an algorithm that is very fast, reliable, and yet quite accurate in this scenario.

Scenario III is similar to II except that we now consider a complex-valued CP decomposition where $\boldsymbol{F}^{(2)}$, $\boldsymbol{F}^{(3)}$, and the additive noise are drawn from a complex Gaussian distribution. For $\boldsymbol{F}^{(1)}$ we still use (7.46). The CCDF of the TMSFE is depicted in Figure 7.4. This time, we compare our SECSI solutions with the complex-valued ALS-based COMFAC algorithm [BSG99] (note that CPOPT is limited to real-valued data only). As before we find that the ALS-based algorithm is often slightly more accurate, however, it shows outliers with a very large error, which does not occur for our SECSI algorithms. The four algorithms perform almost equally well, in the same order as before (BM, REC PS, CON PS, RES) and with only

0.2 dB difference, outperforming the dummy heuristic by about 1.5 dB. The gap to the genie heuristic is around 2 dB in this case. Also, it always outperforms ALS-PARAFAC. Considering the run-time we find again a very significant improvement by a factor of almost 50 comparing COMFAC to the fastest SECSI algorithm CON PS.

In Scenario IV we consider a four-way tensor that is degenerate in two modes. This implies that only the two SMDs can be solved which stem from the diagonalization of the three-mode slices of $\boldsymbol{\mathcal{X}}^{(2,3)}$ and all four SECSI algorithms solve both of them. Therefore, two estimates for each loading matrix are available. In this case, for the selection of the final estimates, the BM algorithm tests all $2^4 = 16$ combinations, the REC PS as well as the CON PS algorithm consider only two, and the RES criterion considers only one. This explains the difference in MSRE between the four algorithms, which is depicted in Figure 7.5. In this case, we use a simple multi-linear ALS implementation (M-ALS) without line search since the existing schemes are not applicable to this scenario (ALS-COMFAC is limited to $R = 3$ whereas the ALS-PARAFAC is limited to real-valued data). We initialize M-ALS randomly and allow for a maximum of 10 000 iterations. The algorithm is terminated when the innovation in one iteration drops below $\delta = 10^{-8}$, where the innovation is defined as $\sum_{r=1}^{R} \|\hat{\boldsymbol{F}}_k^{(r)} - \hat{\boldsymbol{F}}_{k-1}^{(r)}\|_{\mathrm{F}}^2 / \|\hat{\boldsymbol{F}}_k^{(r)}\|_{\mathrm{F}}^2$. Here $\hat{\boldsymbol{F}}_k^{(r)}$ refers to the estimate of the r-th loading matrix in the k-th iteration.

As before, we find infrequent outliers for M-ALS and a more accurate estimate in all other cases, however, the difference is more severe. Therefore, we depict another set of dashed curves where we use our SECSI algorithms as a starting point and then run a single ALS sweep for refinement. The resulting TMSFE is very close to the performance of the M-ALS scheme and yet does not show any outliers. In terms of the run-time, Table 7.3 shows that the SECSI solutions are faster than M-ALS by 30-40 %. This is still true if a single ALS refinement step is carried out since this step adds only around 4 ms of run time.

In Scenario V we consider a non-degenerate complex-valued four-way tensor with badly conditioned loading matrices in the first and the second mode. Random loading matrices with bad conditioning are generated according to $\boldsymbol{F}_{\mathrm{cor}} = \boldsymbol{F}_{\mathrm{white}} \cdot \boldsymbol{R}(\rho)$, where $\boldsymbol{F}_{\mathrm{white}}$ contains i.i.d. elements and $\boldsymbol{R}(\rho)$ is a fixed matrix given by

$$\boldsymbol{R}(\rho) = (1-\rho) \cdot \boldsymbol{I}_d + \frac{\rho}{d} \cdot \mathbf{1}_{d \times d}. \tag{7.47}$$

For Scenario V we choose $\rho = 0.98$ so that $\mathrm{cond}(\boldsymbol{R}) = 50$. Figure 7.6 depicts the CCDF of the TMSFE for this scenario and Figure 7.7 shows the CCDF of the run-times. We observe that the outlier probability for M-ALS is very high and the SECSI algorithms do not produce any outliers. In terms of estimation accuracy, the BM algorithm performs best, almost achieving the optimum genie heuristic. However, since 12 SMDs have to be solved and $12^4 = 20\ 736$ combinations of estimates have to be computed for the exhaustive search, the computational complexity is very high, even higher than the ALS procedure. On the other hand, the RES and the REC PS heuristics are significantly faster than ALS but the TMSFE is higher than for BM. As before, the CON PS heuristics is the fastest (beating ALS by a factor of 380 in terms of computation time), however, it is also the least accurate. Yet, CON PS still outperforms the dummy heuristic quite significantly. This simulation shows that our SECSI framework allows to adjust the trade-off between accuracy and computational complexity.

Finally, for Scenario VI, we have a large real-valued tensor of size $80 \times 80 \times 80$ with a badly conditioned loading matrix in the first mode, which is randomly drawn like in Scenario V. Figure 7.8 shows the CCDF of the TMSFE. We observe that all SECSI heuristics outperform the ALS-based PARAFAC from the N-way toolbox. As before, BM performs best, however, its complexity is high since the large tensor has to be reconstructed many times. The REC PS heuristic performs almost as well at a significantly reduced computation time, comparable to ALS-PARAFAC. The fastest heuristic CON PS performs slightly worse than the others. However, its TMSFE is still better than ALS-PARAFAC, in particular with respect to the outliers. Note that compared to the optimal choice of factors given by the genie heuristic, the TMSFE of BM and REC PS are only 1 dB and 2 dB higher, respectively.

The following statements summarize our observations from simulations conducted under various conditions:

- For well-conditioned factor matrices, all SMDs perform equally well, especially if $M_r \gg d$. Therefore, it is typically sufficient to solve one SMD without incurring a significant loss compared to ALS.

- In a strongly correlated case, heuristics help to detect the "correct" SMD to solve and CON PS provides a fast and yet reliable algorithm. Moreover, in such a case ALS tends to produce severe outliers which does not happen for the SECSI schemes.

- In an underdetermined case, the number of SMDs that can be solved is lower and hence it becomes more attractive to solve all of them. REC PS or even BM are attractive solutions here. Also, for large values of d and in an underdetermined case, a single ALS iteration helps to improve the accuracy to the level achieved by ALS (in the cases where ALS converges). For small values of d (i.e., for $M_r \gg d$), this single ALS step has not shown any significant improvements.

- In $R \geq 4$-way problems, BM becomes computationally very complex and therefore, the

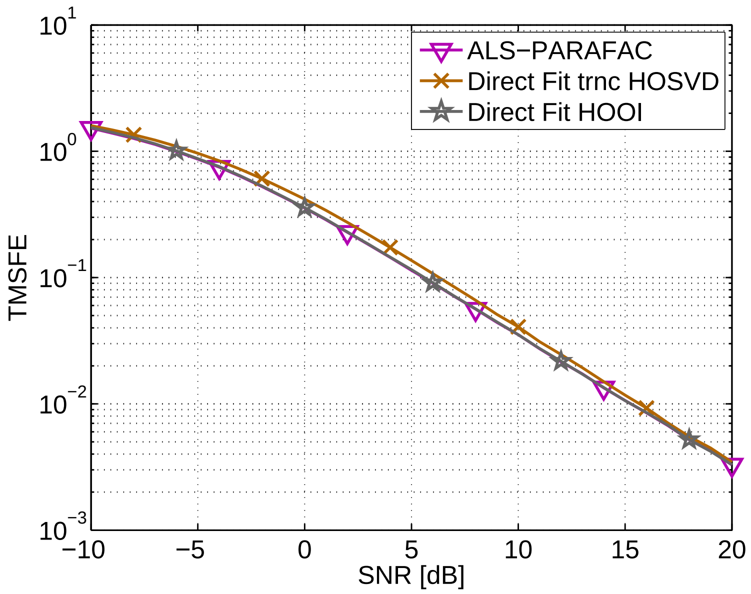


Figure 7.9.: TMSFE vs. SNR for a real-valued $4 \times 7 \times 6$ tensor and $d = 2$.

gain in terms of computation time that is achieved via heuristics is very attractive. Here, RES and CON PS have proven to provide a good complexity/performance trade-off.

7.5.2. Two-component CP decomposition

This section demonstrates numerical results on the direct fitting algorithm for the special case of two components ($d = 2$) which is discussed in Section 7.3.2. We consider a real-valued $4{\times}7{\times}6$ tensor with randomly drawn loading matrices and estimate the CP model from the noisy observations via three different methods: (a) the direct fitting algorithm from Section 7.3.2, based on the truncated HOSVD of $\boldsymbol{\mathcal{X}}$; (b) the same algorithm based on the optimal rank-(2,2,2) approximation found via the HOOI algorithm[12] applied to $\boldsymbol{\mathcal{X}}$; (c) the ALS-PARAFAC algorithm from the N-way toolbox applied to $\boldsymbol{\mathcal{X}}$.

Figure 7.9 shows the TMSFE vs. the SNR. We observe that the direct fitting based on the HOOI algorithm yields the same TMSFE as the ALS-PARAFAC procedure. Moreover, the direct fitting based on the truncated HOSVD results in a TMSFE that is only insignificantly higher. The corresponding MSRE is depicted in Figure 7.10 where the difference between truncated HOSVD and HOOI is even smaller. Considering the average run-time of the algorithms we find 31 ms for ALS-PARAFAC, 22 ms for the direct fitting based on the HOOI and

[12] A MATLAB implementation of the HOOI algorithm was kindly provided to us by L. de Lathauwer.

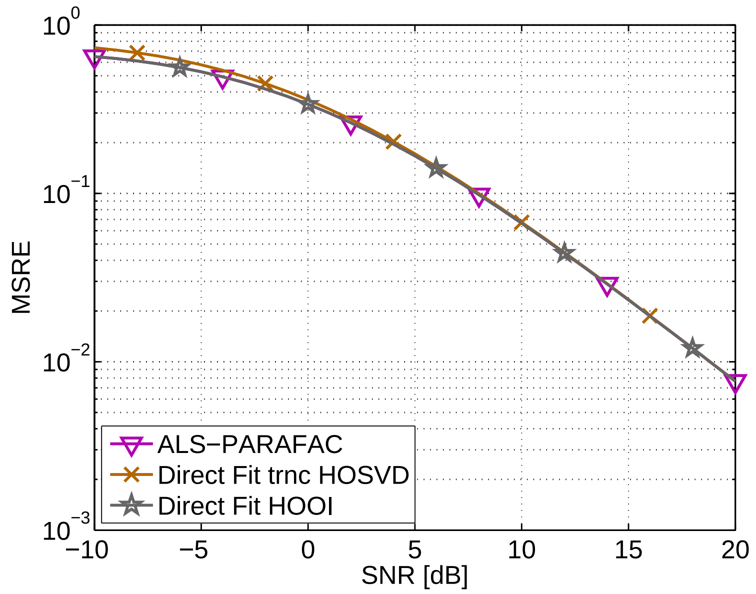


Figure 7.10.: MSRE vs. SNR for a real-valued $4 \times 7 \times 6$ tensor and $d = 2$.

only 1.5 ms for the direct fitting based on the truncated HOSVD, which is more than 20 times faster than ALS-PARAFAC.

7.6. Summary and Conclusions

In this chapter, we have discussed a framework for the **SE**mi-algebraic computation of approximate **C**P decompositions via **SI**multaneous Matrix Diagonalizations (SECSI).

In contrast to previous approaches that have already pointed out the link between CP decompositions and Simultaneous Matrix Decompositions (SMDs), we have shown that the symmetry of the problem allows to construct several SMDs instead of only one. This enables us to obtain multiple estimates for the CP model, and, therefore, we can select a final solution in a subsequent step. The selection of the final estimates as well as the choice which SMDs to solve are design parameters that allow to create different algorithms within the SECSI framework. Overall this creates the flexibility to adapt to specific scenarios and to different demands in terms of accuracy and computational complexity. We have proposed a number of heuristics and have evaluated them via numerical computer simulations demonstrating the enhanced reliability as well as the flexibility in controlling the performance-complexity trade-off.

Moreover, we have extended the SECSI framework to the general multi-way case and have demonstrated that symmetries in the CP can readily be exploited. Finally, in the special cases of a two-slab or a two-component CP decomposition, the SECSI framework reduces to known fully algebraic solution without the necessity to compute any SMD.

8. Summary of efficient tensor decompositions

8.1. Summary of own contributions

In this part of the thesis, we have discussed a framework for the **se**mi-algebraic computation of approximate **C**P decompositions via **Si**multaneous Matrix Diagonalizations (SECSI) first proposed by us for the three-way case in [RH08b] and for the general R-way case in [RH08a]. While the fundamental idea to compute a CP decomposition via SMDs has been proposed before [vdVP96, dL04a], we have extended the concept in many ways.

The main novel contributions are:

- Demonstrating that not only one, but many SMDs can be constructed due to the symmetry of the problem. In fact, we have established the full set of possible SMDs for the 3-D and the general R-D cases. In the non-degenerate case, up to $(R-1) \cdot R$ SMDs can be constructed.
- Showing that from each SMD, estimates for all loading matrices can be obtained. This provides us with several candidate solutions from which we can select one final estimate in a subsequent step.
- Depending on how many SMDs we choose to solve, how we select them, and how we find the final estimate for the loading matrices, many different algorithms inside the SECSI framework can be defined, which achieve different complexity accuracy trade-off points. A major contribution here is to propose several selection criteria for which SMDs to solve and for finding the final model estimate. As a result, four different algorithms have been proposed to demonstrate this flexibility with examples. The algorithms are compared in terms of their accuracy and their complexity (via run times).
- Discussing the special cases of two-slab and two-component CP where SECSI reduces naturally to known algebraic CP solutions without the need to compute any SMD.

It should be noted that the SECSI framework has already successfully been applied in several practical applications related to biomedical signal processing. In [JRW+08, WRH+09], it is used to identify space-time-frequency components in event-related Electroencephalography (EEG) data. This idea is extended to a temporally resolved component analysis of dynamic EEG sources in [WJG+10b, WJR+10, WJG+10a]. The analysis is based on PARAFAC2 which can be computed via a sequence of CP decompositions, for which the SECSI framework is used. Finally, in [BCA+10, BCA+12], the SECSI framework is applied to a space–time–wave-vector tensor obtained by a local spatial 3-D Fourier transform of the EEG measurement data.

The SECSI framework was also applied to model order selection for multidimensional data [DRWH10] where the fact that multiple estimates of the same model are found in SECSI was used to build a test for the plausibility of a candidate model order. Moreover, [DSR+10] discusses SECSI for parameter estimation from multidimensional signals with colored noise and imperfect antenna arrays. Another application is blind channel estimation for Space-Time-Coded MIMO systems [RSS+11].

8.2. Outlook and future work

While the SECSI framework offers a lot of flexibility in designing algorithms tailored to the specific needs of certain applications, it still has a number of shortcomings.

Firstly, it would be desirable to find analytical results for the achievable accuracy of the SMDs. Ideally, such results could enable us to define the different algorithms in the SECSI framework based on analytically more profound arguments than the heuristic solutions we have defined so far. They could even guide us to devise mechanisms for the automatic selection of a suitable heuristic based on certain properties of the observed data (e.g., size, conditioning). However, this is a difficult task due to the iterative nature of SMD algorithms [FG06, LA11]. Up to now, no usable performance analysis for SMDs exists (in fact, the same is true for the iterative ALS-based schemes).

Secondly, the identifiability constraint of requiring two non-degenerate modes seems to be too harsh. Especially in R-D scenarios for $R > 3$ the tensor rank can be significantly higher than the size of the tensor in all modes [KB09]. This constraint could be relaxed by considering "generalized" unfoldings where several tensor indices are aligned in the rows and the remaining indices in the columns of the unfolding matrix (as in [LA11]). We have developed an extension of the SECSI framework to take these generalized unfoldings into account and initial results on its performance are reported in [RSH12]. As expected, this extended framework outperforms SECSI in particular for $R > 3$. Since the generalized unfoldings offer a vast amount of options how to arrange the data, the number of potential model estimates becomes very large. Therefore, designing and evaluating appropriate heuristics for the extended framework is a challenging task with great potential for additional improvements.

Thirdly, in terms of an efficient practical implementation, the potential to parallelize SECSI should be investigated. It seems that the idea to compute several candidate solutions for the

CP model is already tailored to this goal. However, a further parallelization might be needed, i.e., a parallelized implementation of the SMD algorithms that build the core of the framework. For many applications, another desirable aspect is an on-line implementation where the current solution is updated by a new observation without having to recompute the entire model. Note that adaptive PARAFAC decompositions (not based on SMDs) exist, e.g., [NS09].

Finally, as highlighted in Section 4.3, many other explanatory tensor decomposition have been found, such as the Block Tensor Decompositions [dL08], the PARALIND decomposition [BHS05, BHSL09] or the PARAFAC2 decomposition [Har72]. The potential of applying the spirit of the SECSI framework to any of these tensor decompositions should be evaluated.

Part III.

Subspace-Based Parameter Estimation

This part of the thesis is devoted to multi-dimensional subspace-based high-resolution parameter estimation. Its main focus is on demonstrating that multi-dimensional signals can be stored and manipulated in their native multi-dimensional form by virtue of tensor calculus. Using the concepts of multi-linear algebra and tensor decompositions discussed in Part I of the thesis, we arrive at a data model that reflects the multi-dimensional structure of the signals in a natural form. As we show, based on such models, many existing multi-dimensional subspace-based high-resolution parameter estimation techniques can be significantly improved.

In Chapter 9 we provide a motivation, the tensor-based multi-dimensional data model, and a review of the state of the art. Chapter 10 is then devoted to the multi-dimensional subspace estimation, where an enhanced tensor-based subspace estimate is introduced which can be applied to improve arbitrary multi-dimensional subspace-based parameter estimation schemes. The subsequent Chapter 11 discusses one specific class of parameter estimation schemes, namely the family of Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT)-type algorithms. A tensor-based formulation of ESPRIT is shown which forms the basis for several enhancements of known ESPRIT-type algorithms that are achieved by exploiting certain patterns in the received data. In Chapter 12 we introduce a framework for the analytical performance assessment of arbitrary ESPRIT-type algorithms and use it to assess the enhancements obtained via the modified ESPRIT-type algorithms proposed in Chapter 11. Finally, Chapter 13 contains a summary of the major achievements and an outlook to possible future work. To enhance the readability, proofs and derivations have been moved to Appendix D.

9. Introduction to subspace-based parameter estimation

9.1. Motivation

Multi-dimensional subspace-based high-resolution parameter estimation is a generic task required for a variety of applications. The general idea is to estimate parameters from a linear superposition of signals that are separable across R dimensions. A mathematical definition of this model is given in Section 9.2.

One example for such signals are multi-dimensional harmonics which are sampled on a multi-dimensional lattice. Here, the goal is to estimate the frequencies of these harmonics in all dimensions, which is referred to as *harmonic retrieval* [KAB83, AR88]. The multi-dimensional harmonic retrieval problem appears in many different application areas. For instance, it is required to fit the parameters of a double-directional MIMO channel model [ZFDW00, SMB01] from MIMO channel sounder measurements [ZHM+00]. It has been shown that under idealized conditions, the channel transfer function corresponding to this model can be described by a R-D harmonic signal ($R = 3$ using the spatial dimension at the transmitter, the receiver, and the frequency dimension [ZHR+04], or even $R = 6$ using the 2-D aperture domains at the transmitter and the receiver, the time, and the frequency dimension [RHST01]). A related application is MIMO Radar [JLL09, NS10]. As shown in [NS10], single-pulse and multi-pulse bistatic Radar configurations give rise to a 2-D and a 3-D harmonic retrieval problem, respectively. Another application is 2-D nuclear magnetic resonance (NMR) spectroscopy [BL86] where a molecular system is excited with a 2-D RF pulse sequence and the measured signal can be modeled as a sum of 2-D (damped) harmonics.

A related example where such signals are present is direction of arrival (DOA) estimation in antenna array processing applications. It can be shown that sources transmitting to a two-dimensional antenna array configuration can be resolved in azimuth and elevation if they are in the far-field of a receiving antenna array [SK93]. Note that the 2-D DOA estimation task is a special case of 2-D harmonic retrieval only under certain conditions on the antenna array (see Appendix D.2). Firstly, each element in the array must possess the same complex beam pattern and secondly, the array manifold must be separable in two spatial dimensions (e.g., using a uniform rectangular array). Many 2-D DOA estimation algorithms do not rely on these assumptions. As an example, we have shown how to perform 2-D DOA estimation for hexagonally shaped arrays via 3-D Unitary ESPRIT in [RH05, RH06]. For non-uniform arrays, array interpolation [FW92], the manifold separation technique [DD94, BRK07], or Fourier-domain techniques [RG09] have been investigated.

While harmonic retrieval could be achieved via Discrete Fourier Transform (DFT)-based spectral estimators (namely, the periodogram), the main disadvantage of such an approach is that the resolution is fundamentally limited by the sample support via the "Rayleigh" resolution limit: If M uniformly spaced samples of the harmonics are available, two frequencies that are closer than $2\pi/M$ cannot be resolved [Sch91].

High-resolution spectral estimators are able to break the limit by performing a *parametric* spectral estimation. Here, parametric refers to assuming a specific model of our data (e.g., a superposition of harmonics) and only estimating the model parameters (namely, frequencies and amplitudes) instead of the entire spectrum. Therefore, while the DFT can be applied to arbitrary signals, parametric spectral estimators can only be applied if we know that our data follows a specific model. Exploiting this knowledge typically allows us to increase the resolution tremendously.

High-resolution parameter estimation has been a field of very active research for several decades and a large family of methods exists. In the light of the scope of this thesis we are particularly interested in low-complexity algorithms. Therefore, we focus on the class of *subspace-based* multi-dimensional parameter estimation schemes schemes, since these are known for their simplicity.

Existing subspace-based high-resolution parameter estimation schemes can be classified into three categories according to their numerical procedure:

- *extrema-searching techniques*, e.g., spectral MUSIC [Sch79], the Rank-Reduction Estimator (RARE) [PGW02] or the weighted subspace fitting approach [VOK91]
- *polynomial rooting techniques*, e.g., Pisarenko's harmonic decomposition [Pis73], the Min-Norm algorithm [KT83], Root-MUSIC [Bar83], Unitary Root-MUSIC [PGH00], root-RARE [PGW02], the Method of Direction of Arrival Estimation (MODE) [Van02], the manifold separation technique [DD94, BRK07] or the Fourier domain root-MUSIC [RG09]
- *matrix-shifting techniques*: State-Space methods [KAB83, RA92], matrix pencil methods [HS91], Standard ESPRIT [RPK86], optimally weighted ESPRIT [ES94], Multiple Invariance ESPRIT [SORK92], or Unitary ESPRIT [HN95].

Multi-dimensional parameter estimation problems can be solved by applying the above-mentioned estimators to each dimension separately. However, the major drawback of this approach is that the correct pairing of the parameter estimates across the dimensions has to be found in a subsequent step (see, for instance, [RZZ93]). Moreover, by exploiting the R-D structure via a joint processing over all dimensions, we can typically benefit in terms of an enhanced estimation accuracy. Therefore, multidimensional versions of some of these techniques have also been introduced. For instance, [HN98] discusses a Simultaneous Schur Decomposition (SSD) to achieve the automatic pairing in an R-D version of Unitary ESPRIT. A 3-D version of the Multidimensional Folding (MDF) scheme is introduced in [MSPM04], the R-D improved MDF (IMDF) algorithm is shown in [LL06]. Moreover, a 2-D version of RARE is discussed in [PMB04]. In [HF94], the Polynomial Root Intersection for Multi-dimensional Estimation (PRIME) algorithm is introduced that allows to extend 1-D polynomial rooting techniques to the 2-D case. An overview of search-free 1-D and 2-D DOA estimation methods is also presented in [GRP10]. The application of 2-D and 3-D harmonic retrieval for MIMO radar is discussed in [NS10].

These approaches to R-D subspace based parameter estimation have in common that the R-D signals are stored in matrices by means of a stacking operation. Obviously this representation does not account for the R-D grid structure inherent in the data. This means that the structure is not fully exploited which implies a potential loss in estimation accuracy. In fact, comparing different unbiased subspace-based high-resolution parameter estimation schemes to the ultimate lower bound on their performance given by the Cramér-Rao Bound [SN89] we find that for many schemes there exist scenarios where the lower bound is not reached.

Therefore, we introduce a more natural approach to store and manipulate the R-D data in its "native" multidimensional form, which is accomplished by using tensors. Tensor decompositions have been known for many decades. As discussed in Section 4.2, the prominent Canonical Polyadic (CP) decomposition, also known as Canonical Decomposition (CANDECOMP) or Parallel Factor (PARAFAC) analysis, goes back to [CC70, Har70]. PARAFAC has been linked to multidimensional harmonic retrieval in [SBG00]. The main drawback of PARAFAC-based harmonic retrieval is that the computation of the PARAFAC decomposition requires iterative algorithms with a high computational complexity. A thorough discussion of this point is provided in Part II of this thesis.

A computationally much more simple tensor decomposition is given by the Higher Order Singular Value Decomposition (HOSVD) [dLdMV00a], which is a special case of the Tucker3 decomposition [Tuc66]. Like the CP decomposition, the HOSVD can be seen as a multilinear extension of the Singular Value Decomposition (SVD). Since it is very easy to compute via a sequence of SVDs, the HOSVD is preferable over the CP decomposition in the light of the scope of this thesis as we aim at finding algebraically simple and yet efficient solutions.

There have been other attempts to combine tensor decompositions with subspace-based high-resolution parameter estimation schemes before. For instance, a tensor-based spectral MUSIC algorithm for polarized seismic source localization is discussed in [MLM05]. The authors construct a covariance tensor from a sequence of 2-D observation matrices (antennas and polarization), which they decompose into a set of mutually orthogonal eigentensors with a Higher-Order Eigenvalue Decomposition (HOEVD) [dLdMV00a]. The null-space projection for spectral MUSIC is then found from the $M - d$ weakest eigentensors where M is the total number of sensors and d the number of sources. A tensor gain of the resulting method compared to previous matrix-based MUSIC is reported in [MLM05]. Single-channel and multi-channel harmonic retrieval via the HOSVD and the Hankel total least squares method is discussed in [PdLvH05]. A 3-D "Hankel tensor" is constructed and decomposed into a diagonal tensor and three Vandermonde matrices in the separate modes that share common generators. Subspaces for these 1-D Vandermonde matrices are estimated separately in the three dimensions based on a truncated HOSVD, whereas we propose to use all dimensions jointly, see Section 10.2. A decoupled Root-MUSIC algorithm for R-D harmonic retrieval is discussed in [Boy08], where a HOSVD is used to find subspaces in the separate 1-D modes for a sequence of 1-D Root-MUSIC algorithms. The correct pairing across dimensions has to be found in a subsequent step via an iterative procedure. Finally, R-D harmonic retrieval based on the CP decomposition of a fourth-order cumulant tensor is for instance discussed in [Boy06]. However, all these approaches have in common that they are tailored to one specific algorithm instead of providing a generic approach to using tensors in multi-dimensional parameter estimation schemes. Moreover, the HOSVD is used only to find subspaces in separate modes but not to estimate one combined R-D tensor-based subspace estimate.

To this end, we have defined the HOSVD-based signal subspace estimate for multi-dimensional signals which takes advantage of the multidimensional structure of the data in [RHD06]. This subspace replaces the unstructured matrix-based subspace estimate and can hence be used to improve arbitrary multi-dimensional subspace-based parameter estimation schemes. Based on this result, completely tensor-based formulations of R-D Standard ESPRIT and R-D Unitary ESPRIT have been provided in [HRD08]. These have enabled us to find further improvements to R-D ESPRIT-type algorithms that take advantage of the tensor structure, e.g., the Tensor-Structure SLS (TS-SLS) algorithm [RH07b], which provides a tensor gain even in cases where the HOSVD-based signal subspace estimate is identical to the SVD-based subspace estimate (cf. Section 10.2 and the simulation result shown in Figure 11.6). R-D Standard Tensor-ESPRIT and R-D Unitary Tensor-ESPRIT are discussed in Section 11.4, TS-SLS is shown in Section 11.7 (see also Table 13.1 and Table 13.2 for an overview of the various ESPRIT-type algorithms discussed in this part of the thesis).

In DOA applications where the sources are restricted to transmit symbols from real-valued constellations (such as Binary Phase Shift Keying (BPSK) or Amplitude Shift Keying (ASK)) or they use an Offset Quadrature Phase Shift Keying (OQPSK) or a Minimum Shift Keying (MSK) modulation scheme, the receiver obtains a superposition of multi-dimensional signals with complex amplitudes that are no longer circularly symmetric [Pic94]. In this case, the covariance matrix does not fully describe the second-order behavior and a second quantity known as relation matrix [Pic96], pseudo covariance matrix [NM93], or complementary covariance matrix [SS03] needs to be taken into account. It has been shown that this matrix provides additional structural information that can be exploited, e.g., via widely linear signal processing. Fundamental results on the modeling and the prediction of non-circular random processes are discussed in [PB97], widely linear MMSE estimation is shown in [PC95], and a rank-reduced Wiener filter is provided in [SS03]. Moreover, applications to detection and estimation, widely linear beamforming, equalization, interference-suppression, and multi-user detection in DS-CDMA systems are, for instance, presented in [SS05], [CB07, SSW+11, SdLW+11], [GSL03], [SDHW12], and [BL03], respectively. For a survey on non-circular signals and their applications in signal processing, the interested reader is referred to [SS10].

For the subsequent chapters, we are interested in the application of widely linear signal processing for non-circular random variables to subspace-based high-resolution parameter estimations schemes. It has been shown that existing algorithms can be improved significantly if the non-circularity of the source symbols is exploited. Corresponding versions of Root-MUSIC and Standard ESPRIT have been introduced in [CWS01], [ZCW03], respectively. Moreover, MUSIC-like procedures for the case where circular and strict-sense non-circular signals coexist are discussed in [GNW08, LLXZ12]. A spectral MUSIC-like estimation algorithm that takes advantage of weak-sense non-circular sources is shown in [AD06]. However, the improvement compared to the circular case is particularly strong only for the limiting case of strict-sense non-circularity. Due to the fact that non-circular sources can be separated based on their phase offsets, the improvement in terms of resolution capabilities can be tremendous (cf. Section 11.8 where we present numerical results).

We have extended these proposals by a corresponding version of Unitary ESPRIT, first introduced in [HR04]. As we show in Section 11.5.2, the performance of the resulting "NC Unitary ESPRIT" algorithm is identical to the Standard ESPRIT version from [ZCW03] while being significantly less complex since all computations are carried out in the real-valued domain. Moreover, due to the row-wise augmentation of the mesurement matrix, the antenna array is virtually doubled, which doubles the number of resolvable wavefronts and provides a significantly lower estimation error compared to Unitary ESPRIT.

Exploiting the non-circularity and the multidimensional structure of the signals jointly has first been proposed in [RH09b], where R-D NC Tensor-ESPRIT-type algorithms are introduced. As we discuss in Section 11.6, combining the two approaches is not a trivial task, since the row-wise augmentation performed for NC Unitary ESPRIT destroys the separable array structure required for tensor-based ESPRIT-type algorithms. As a remedy, a mode-wise tensor augmentation is proposed and Tensor-ESPRIT-type algorithms for non-circular sources are derived.

An additional major contribution in this part of the thesis is the introduction of a generic framework for the *analytical* performance assessment of multi-dimensional subspace-based parameter estimation schemes. This framework is used to derive a first-order perturbation expansion of the tensor-based subspace estimate as well as the ESPRIT-based estimates of the desired frequencies. The motivation behind such an analytical framework is to gain further insights into when the tensor-based approach provides a significant improvement in terms of the performance and when this is not the case.

Our framework is based on a first-order perturbation expansion of the Singular Value Decomposition which was introduced in [LLV93]. This expansion models the estimation error of the signal subspace as an explicit function of the perturbation of the data (i.e., the additive noise), only assuming that the perturbation is small compared to the desired signal. The main advantage of this approach is that no assumptions about the statistics of the desired signal or the perturbation are required. Moreover, it is asymptotic in the effective SNR, i.e., it becomes exact as either the noise variance vanishes or the number of samples approaches infinity.

This is not the case for many existing analytical performance results for high-resolution parameter estimation algorithms. The most frequently cited papers [KB86] for the MUSIC algorithm and [RH89a] for ESPRIT as well as many follow-up papers which extend the original results (e.g., [PK89b], [Fri90], [MZ94], [ZKM92], [MHZ96]) are based on a result on the distribution of the eigenvectors of a sample covariance matrix first published in [Bri75]. However, the result shown in [Bri75] requires the desired signal as well as the noise to be Gaussian distributed which is not required by our framework based on [LLV93]. Moreover, [Bri75] is asymptotic in the number of samples, whereas [LLV93] can be applied even in the single snapshot case (provided the noise variance is sufficiently small). Another drawback of [Bri75] is that the expressions for the covariance matrix of the eigenvectors are rather long and difficult to simplify, whereas [LLV93] provides intuitive and short expressions.

We develop explicit expansions of the estimation errors of the subspaces and the spatial frequencies in terms of the additive noise as well as generic Mean Squared Error (MSE) expressions. Furthermore, we show that in special cases, the MSE expressions can be simplified into very compact formulas that only depend on the physical parameters such as the dimensions of the array or the Signal to Noise Ratio (SNR). Such expressions provide deep insights into when the corresponding algorithms are particularly efficient. One important conclusion we can draw based on these analytical results is that for a single source, all Least Squares (LS)-based ESPRIT-type algorithms yield the same MSE and that for larger arrays, the MSE can be significantly improved via a single iteration of the Structured Least Squares (SLS) algorithm.

9.2. Notation and data model

In order to motivate the tensor-based data model for multi-dimensional signals we discuss two examples for applications where such signals appear. The examples are:

Example 9.2.1. 2-D direction of arrival (DOA) estimation: Consider d terminals that transmit signals wirelessly to a central receiver, e.g., a base station, which is equipped with a two-dimensional antenna array. It is our goal to estimate the azimuth and elevation angles (directions of arrival, defined in Figure 9.1) under which the base station receives the users' transmissions.

Example 9.2.2. Geometry-based parametric channel modeling: Here the goal is to describe the propagation of a signal between a transmitter and a receiver using a geometry-based channel model and fitting the model parameters to realistic MIMO channels measured in a channel sounding experiment [ZHM+00]. A popular geometry-based channel model is the double-directional MIMO channel model [ZFDW00, SMB01], which describes the propagation by a superposition of rays that are reflected by point-like scatterers. To perform the measurement, both transmitter and receiver are equipped with multiple antennas and one link end (typically the receiver) is moved along predefined measurement tracks. For each position, the channel between all pairs of transmit and receive antennas is measured, for instance, by employing an antenna switching pattern. More precisely, for each antenna pair the channel transfer function is measured in a certain bandwidth, which can be achieved by transmitting a periodic multi-sinusoidal signal to obtain the complex channel gain at specific frequency bins [THR+00]. This measurement setup allows to resolve the dominant individual propagation paths in multiple dimensions:

(a) the spatial dimensions at the receiver, from which directions of arrival can be estimated;

(b) the spatial dimensions at the transmitter, allowing us to estimate directions of departure;

(c) the frequency dimension and the time dimension, which allow to access the delay and the Doppler shift of the individual paths;

(d) polarization, providing additional insight to how the spatial orientation of the wave is altered during the propagation.

As we show in the sequel, under certain assumptions both examples give rise to a superposition of multi-dimensional signals which can efficiently be modeled via tensors. Let us begin with a generic mathematical formulation of the data model and then provide the link to the two examples given above.

9.2.1. Scalar data model

Definition 9.2.1. A signal $x^{(0)}(t) \in \mathbb{C}$ is called R**-dimensional** ($R > 1$) if it can be expressed as a separable function in R variables $p^{(1)}, p^{(2)}, \ldots, p^{(R)} \in \mathbb{R}$ in the following manner

$$x^{(0)}\left(p^{(1)}, p^{(2)}, \ldots, p^{(R)}, t\right) = s(t) \cdot \prod_{r=1}^{R} a^{(r)}\left(p^{(r)}\right), \tag{9.1}$$

where $s(t) \in \mathbb{C}$ is the amplitude of the signal and the function $a^{(r)}\left(p^{(r)}\right) : \mathbb{R} \to \mathbb{C}$ describes the variation of the signal along variable $p^{(r)}$ (in dimension r).

Note that the superscript $^{(0)}$ is used to indicate the noise-free signal.

Definition 9.2.2. A **snapshot** of an R-dimensional signal is defined as an observation of an R-dimensional signal according to Definition 9.2.1 sampled with M_r sampling points $p_1^{(r)} < p_2^{(r)} \ldots < p_{M_r}^{(r)} \in \mathbb{R}$ in the variable $p^{(r)}$ for $r = 1, 2, \ldots, R$, such that we can write

$$x^{(0)}_{m_1, m_2, \ldots, m_R}[n] = x^{(0)}\left(p^{(1)}_{m_1}, p^{(2)}_{m_2}, \ldots, p^{(R)}_{m_R}, t_n\right) = s[n] \cdot \prod_{r=1}^{R} a^{(r)}\left(p^{(r)}_{m_r}\right), \tag{9.2}$$

where n is an integer index enumerating the N snapshots, i.e., $n = 1, 2, \ldots, N$ and $s[n] = s(t_n) \in \mathbb{C}$ is the amplitude of the signal x at the snapshot n.

Based on Definition 9.2.2 the generic formulation of our R-D data model is the following: Our observed signal is modeled as a sequence of N snapshots of a linear superposition of d R-dimensional signals under additive noise. Mathematically, we can write

$$x_{m_1, m_2, \ldots, m_R}[n] = \left(\sum_{i=1}^{d} s_i[n] \cdot \prod_{r=1}^{R} a_i^{(r)}\left(p^{(r)}_{m_r}\right)\right) + n_{m_1, m_2, \ldots, m_R}[n], \tag{9.3}$$

where $i = 1, 2, \ldots, d$ enumerates the signals and $n_{m_1, m_2, \ldots, m_R}[n]$ represents the additive noise.

To underpin the practical significance of the mathematical model shown in (9.3), we return to the examples from above and show under which conditions they give rise to signals that obey (9.3).

Example 9.2.1 (continued): The 2-D DOA Example 9.2.1 obeys (9.3) under the following assumptions:

(A1) Each terminal transmits a narrow-band bandpass signal modulated at the carrier frequency f_{c} and described by a digital modulation scheme with symbol rate T_{s} where $T_{\mathrm{s}} \cdot f_{\mathrm{c}} \ll 1$. Therefore, at time instant $n \cdot T_{\mathrm{s}}$, the complex symbol $s_{0,i}[n]$ is transmitted by the i-th terminal for $i = 1, 2, \dots, d$. The received symbol $s_i[n]$ is then given by $s_i[n] = g \cdot s_{0,i}[n] \mathrm{e}^{-\jmath \varphi_i}$, where $g \in \mathbb{R}$ and $\varphi_i \in [-\pi, \pi]$ represent the magnitude and the phase of the path between the i-th user and the receiver, respectively.

(A2) The terminals are in the far-field of the antenna array, i.e., their distance to the base station is significantly larger than the wavelength $\lambda_{\mathrm{c}} = \mathrm{c}/f_{\mathrm{c}}$, where c denotes the speed of light.

(A3) The array manifold can be described by a separable function in two spatial variables. An example for such an array configuration is an $M_1 \times M_2$ Uniform Rectangular Array (URA) with isotropic antenna elements and inter-element spacing $\Delta^{(1)}$ and $\Delta^{(2)}$, respectively. To avoid spatial aliasing, $\Delta^{(1)}$ and $\Delta^{(2)}$ should be less than or equal to $\lambda_{\mathrm{c}}/2$.

Note that from (A2) it follows that the base station observes a planar wavefront in two dimensions which is sampled uniformly due to (A3), i.e., we observe a 2-dimensional harmonic wave. The resulting observed sample at antenna m_1, m_2 and time snapshot n can be written as [HN98]

$$x_{m_1,m_2}[n] = \sum_{i=1}^{d} s_i[n] \prod_{r=1}^{2} \mathrm{e}^{\jmath \mu_i^{(r)} (m_r - 1)} + n_{m_1,m_2}[n], \tag{9.4}$$

for $m_r = 1, 2, \dots, M_r$, $r = 1, 2$ and $n = 1, 2, \dots, N$. Moreover, $\mu_i^{(1)}$ and $\mu_i^{(2)}$ are the spatial frequencies of the i-th wavefront (also referred to as the *direction cosines*) which are given by

$$\mu_i^{(1)} = 2\pi \frac{\Delta^{(1)}}{\lambda_{\mathrm{c}}} \cos(\theta_i) \sin(\alpha_i) \tag{9.5}$$

$$\mu_i^{(2)} = 2\pi \frac{\Delta^{(2)}}{\lambda_{\mathrm{c}}} \sin(\theta_i) \sin(\alpha_i), \tag{9.6}$$

where θ_i and α_i represent the azimuth and the co-elevation angle of the i-th planar wavefront

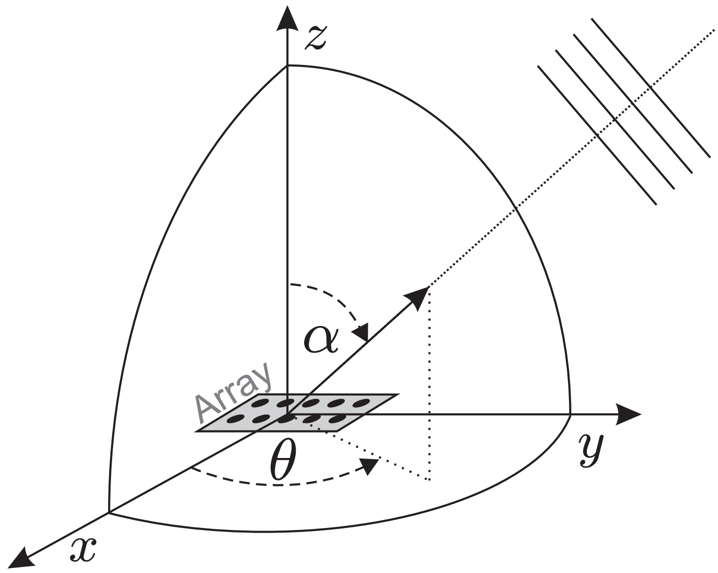


Figure 9.1.: Definition of azimuth (θ) and co-elevation (α) angles of an impinging planar wavefront for the 2-D DOA Example 9.2.1.

relative to the uniform rectangular array. The definition of the angles is visualized in Figure 9.1. Finally $n_{m_1,m_2}[n]$ represents the additive measurement noise. Note that α_i cannot be uniquely recovered from $\mu_i^{(1)}$ and $\mu_i^{(2)}$ since $\sin(\alpha) = \sin(\pi - \alpha), \forall\alpha$. Geometrically, this ambiguity refers to the half space above and below the antenna array. However, this can often be resolved via plausibility in practice. Obviously, (9.4) is a special case of (9.3) for $R = 2$, where the dimensions $p^{(1)}$ and $p^{(2)}$ correspond to the two spatial dimensions of the URA and the functions $a_i^{(r)}(p^{(r)})$ are harmonic waves with frequencies $\mu_i^{(r)}$, i.e., $a_i^{(r)}(p^{(r)}) = \mathrm{e}^{\jmath \cdot \mu_i^{(r)} \cdot p^{(r)}}$.

Example 9.2.2 (continued): It has been shown that under idealized conditions, the channel transfer function corresponding to the double-directional ray-based channel model can be described by a R-D harmonic signal (e.g., $R = 6$ using the 2-D aperture domains at the transmitter and the receiver, the time, and the frequency dimension [RHST01, HTR04]). For illustration purposes, let us consider a strongly simplified example where we assume the following:

(A1) Transmitter and receiver are equipped with uniform linear arrays. Only one polarization is excited at the transmitter and only one polarization is measured at the receiver.

(A2) Only specular components are considered, i.e., we assume that the received signal can be modeled as the superposition of d planar wavefronts as a result of d reflections in the far-field of the transmitting and the receiving antenna arrays.

(A3) We transmit a band-limited signal comprising of N_F sinusoidals spaced by a frequency offset of f_0 apart and centered around a carrier frequency f_c. The complex amplitude of the sinusoidal transmitted at frequency bin n_F is $c[n_\mathrm{F}]$. Note that the antenna characteristics have to be constant over the entire frequency range. This implies that the bandwidth should not be chosen too large since otherwise the variation of the wavelength over the frequency becomes significant. In other words, we need the narrow-band assumption $N_\mathrm{F} \cdot f_0 \ll f_\mathrm{c}$.

(A4) The scenario is stationary during the entire switching between the antennas at the transmitter, i.e., we can model one snapshot of the set of channel gains between the M_T transmit antennas and the M_R receive antennas as if they were measured simultaneously.

In this case, the set of measured channel gains corresponding to samples of the time-varying transfer function between transmit antenna m_T and receive antenna m_R at frequency bin n_F is then estimated by dividing the measured complex amplitudes by the amplitudes $c[n_\mathrm{F}]$ used at the transmitter for $n_\mathrm{F} = 1, 2, \ldots, N_\mathrm{F}$. The resulting channel gains can be expressed in the complex low-pass domain as

$$x_{m_\mathrm{T},m_\mathrm{R},n_\mathrm{F}}[n] = \sum_{i=1}^{d} s_i[n] \cdot g_{m_\mathrm{T}}(\theta_{\mathrm{T},i}) \cdot \mathrm{e}^{\jmath \cdot m_\mathrm{T} \cdot \mu_\mathrm{T}(\theta_{\mathrm{T},i})} \cdot g_{m_\mathrm{R}}(\theta_{\mathrm{R},i}) \cdot \mathrm{e}^{-\jmath \cdot m_\mathrm{R} \cdot \mu_\mathrm{R}(\theta_{\mathrm{R},i})} \cdot \mathrm{e}^{-\jmath \cdot n_\mathrm{F} \cdot \mu_{\mathrm{F},i}} + n_{m_\mathrm{T},m_\mathrm{R},n_\mathrm{F}}[n], \tag{9.7}$$

for $m_\mathrm{R} = 1, 2, \ldots, M_\mathrm{R}$, $m_\mathrm{T} = 1, 2, \ldots M_\mathrm{T}$, and $n_\mathrm{F} = 1, 2, \ldots, N_\mathrm{F}$. Here, $s_i[n] \in \mathbb{C}$ is the time-varying amplitude of the i-th path and $\mu_\mathrm{T}(\theta_{\mathrm{T},i})$ as well as $\mu_\mathrm{R}(\theta_{\mathrm{R},i})$ represent the spatial frequencies of the i-th path at the transmitter and the receiver given by

$$\mu_\mathrm{T}(\theta_{\mathrm{T},i}) = 2\pi \frac{\Delta_\mathrm{T}}{\lambda_\mathrm{c}} \cos(\theta_{\mathrm{T},i}) \tag{9.8}$$

$$\mu_\mathrm{R}(\theta_{\mathrm{R},i}) = 2\pi \frac{\Delta_\mathrm{R}}{\lambda_\mathrm{c}} \cos(\theta_{\mathrm{R},i}), \tag{9.9}$$

where Δ_T and Δ_R correspond to the inter-element spacing of the transmitter's and the receiver's ULA, respectively. Moreover, $\theta_{\mathrm{T},i}$ and $\theta_{\mathrm{R},i}$ are the direction of departure and the direction of arrival angle, as shown in Figure 9.2. The terms $g_{m_\mathrm{T}}(\theta) \in \mathbb{C}$ and $g_{m_\mathrm{R}}(\theta) \in \mathbb{C}$ denote the complex beam patterns of the m_T-th transmit and the m_R-th receive antenna as a function of the azimuth angle θ, respectively. Finally, we have

$$\mu_{\mathrm{F},i} = 2\pi(\tau_i - \tau_\mathrm{ref}) f_0 \tag{9.10}$$

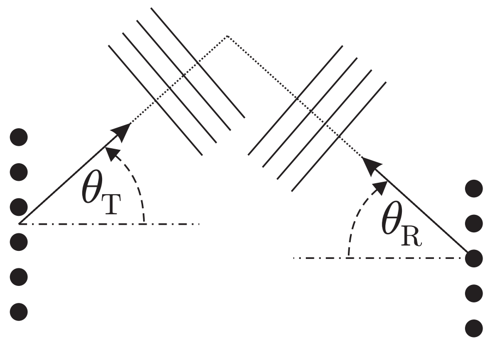


Figure 9.2.: Definition of direction of arrival and direction of departure angles θ_{T} and θ_{R} for the channel sounding Example 9.2.2.

which models the linear phase term in the frequency domain which arises due to the fact that the i-th wavefront arrives with a delay τ_i relative to some reference delay τ_{ref} (e.g., the delay of the first path). Note that due to the sampling of the frequency axis with a subcarrier spacing of f_0, delays are only uniquely recovered in the range $[0, 1/f_0]$.

We conclude that (9.7) is a special case of (9.3) for $R = 3$, where the dimensions $p^{(1)}$ and $p^{(2)}$ correspond to the angular dimensions at the transmitter and the receiver, and $p^{(3)}$ corresponds to the frequency domain, respectively.

However, one measurement of the channel gains provides us with a single snapshot $N = 1$ only. In the next snapshot, the channel conditions have already changed due to the time-varying nature of the propagation conditions. Hence, to obtain multiple snapshots which are needed for subspace-based parameter estimation schemes, spatial smoothing [SWK85] or tensor-based spatial smoothing [THG09a] must be applied (cf. Section 10.3).

9.2.2. Tensor and matrix formulation

In order to arrive at a more compressed formulation of the generic data model in (9.3) we collect the samples of the R-D signal $x_{m_1,m_2,\ldots,m_R}[n]$ at N subsequent snapshots into one array. As our signal is referenced by $R+1$ indices, the most natural way of formulating the model is to employ an $(R+1)$-way array $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R \times N}$ which contains $x_{m_1,m_2,\ldots,m_R}[n]$ for $m_r = 1, 2, \ldots, M_r$, $r = 1, 2, \ldots, R$, and $n = 1, 2, \ldots, N$. We can then conveniently express the model from (9.3) via the n-mode product introduced in Section 4.1 and obtain

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{A}} \times_{R+1} \boldsymbol{S}^{\mathrm{T}} + \boldsymbol{\mathcal{N}}, \tag{9.11}$$

where $\boldsymbol{S} \in \mathbb{C}^{d \times N}$ contains the amplitudes $s_i[n]$, $i = 1, 2, \ldots, d$, $n = 1, 2, \ldots, N$ (corresponding to the symbols in Example 9.2.1 and to the path amplitudes in Example 9.2.2). Note that subspace-based parameter estimation schemes require the rank of the matrix to be equal to d. If the $\operatorname{rank}\{\boldsymbol{S}\} < d$, preprocessing has to be applied, cf. Section 10.3. Moreover, $\boldsymbol{\mathcal{N}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R \times N}$ collects all the noise samples $n_{m_1, m_2, \ldots, m_R}[n]$ in the same manner as $\boldsymbol{\mathcal{X}}$. Finally, $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R \times d}$ is referred to as the "array steering tensor" [HRD08]. It can be expressed by virtue of the concatenation operator defined in (4.4) via

$$\boldsymbol{\mathcal{A}} = [\boldsymbol{\mathcal{A}}_1 \sqcup_{R+1} \boldsymbol{\mathcal{A}}_2 \sqcup_{R+1} \ldots \sqcup_{R+1} \boldsymbol{\mathcal{A}}_d] \tag{9.12}$$

$$\boldsymbol{\mathcal{A}}_i = \boldsymbol{a}_i^{(1)} \circ \boldsymbol{a}_i^{(2)} \circ \ldots \circ \boldsymbol{a}_i^{(R)} \in \mathbb{C}^{M_1 \times M_2 \times \ldots \times M_R} \tag{9.13}$$

$$\boldsymbol{a}_i^{(r)} = \begin{bmatrix} a_{i,1}^{(r)}\left(p_1^{(r)}\right) \\ a_{i,2}^{(r)}\left(p_2^{(r)}\right) \\ \vdots \\ a_{i,M_r}^{(r)}\left(p_{M_r}^{(r)}\right) \end{bmatrix} \in \mathbb{C}^{M_r \times 1}. \tag{9.14}$$

Here, $\circ$ represents the outer product operator introduced in Section 4.1. Equation (9.13) shows how the necessary assumption of separability in defining R-D signals translates to the outer product structure across dimensions in the array steering tensors.

As shown in Appendix D.1, an alternative expression for the array steering tensor is given by

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{I}}_{R+1,d} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \ldots \times_R \boldsymbol{A}^{(R)} = \boldsymbol{\mathcal{I}}_{R+1,d} \mathop{\times}_{r=1}^{R}{}_r \boldsymbol{A}^{(r)}, \tag{9.15}$$

where $\boldsymbol{A}^{(r)} = \left[\boldsymbol{a}_1^{(r)}, \ldots, \boldsymbol{a}_d^{(r)}\right] \in \mathbb{C}^{M_r \times d}$ is referred to as the array steering matrix in the r-th mode.

The strength of the data model in (9.11) is that it represents the signal in its natural R-dimensional structure by virtue of the measurement tensor $\boldsymbol{\mathcal{X}}$. Before tensor calculus was used in this area, a matrix-based formulation of (9.11) was needed. This requires stacking some of the dimensions into rows or columns. A meaningful definition of an R-D measurement matrix $\boldsymbol{X}$ is to apply stacking to all "spatial" dimensions $1, 2, \ldots, R$ along the rows and align the snapshots $n = 1, 2, \ldots, N$ along the columns. Mathematically, we can write $\boldsymbol{X} = [\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times N}$, where $M = \prod_{r=1}^{R} M_r$. Applying this stacking operation to (9.11), we arrive at the matrix-based data model

$$\boldsymbol{X} = \boldsymbol{A} \cdot \boldsymbol{S} + \boldsymbol{N}. \tag{9.16}$$

Here, $\boldsymbol{A} = [\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times d}$ and $\boldsymbol{N} = [\boldsymbol{\mathcal{N}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times N}$. Note that since $\boldsymbol{\mathcal{A}}$ obeys (9.15) we immediately find

$$\boldsymbol{A} = [\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}} = \boldsymbol{A}^{(1)} \diamond \boldsymbol{A}^{(2)} \diamond \ldots \diamond \boldsymbol{A}^{(R)} \tag{9.17}$$

by applying identity (4.25). Therefore, whenever we encounter a linear mixture model like (9.16) where the "mixing matrix" $\boldsymbol{A}$ can be decomposed into a Khatri-Rao product of smaller mixture matrices according to (9.17), an alternative representation of the matrix-valued data model is given by the tensor-valued data model (9.11). We can typically benefit from the more explicit representation of the R-dimensional structure in (9.11) via a tensor, e.g., filter out the unstructured noise more efficiently, as we show in Section 10.2.

9.2.3. 1-D and 2-D antenna arrays

We have seen in Example 9.2.1 that a Uniform Rectangular Array (URA) is an example for a 2-D antenna array that obeys our R-D model for $R = 2$. This sparks the question whether there are other types of 2-D arrays that can be used in this context.

In general, to fully describe a 2-D antenna we have to measure the response of each element as a function of azimuth and elevation, taking into account the polarization of the wave, i.e., excite and measure with both vertical and horizontal polarizations. Since anything in the near-field of an antenna element influences the beam patterns, it is best to measure the elements within the array to capture the effects of mutual coupling. For simplicity, let us focus on a single polarization only. Then, the result of this measurement is a complex beam pattern $g_m(\theta, \alpha) \in \mathbb{C}$ which contains the response of the m-th antenna to a wave impinging from direction θ, α. The 2-D antenna array obeys a 2-D model according to (9.11) only under the following two conditions:

1. The array elements are placed in a 2-D grid that is separable, i.e., it can be constructed as the outer product of two 1-D grids. Examples of not separable and separable 2-D sampling grids are shown in Figure 9.3a and Figure 9.3b, respectively. Figure 9.3c shows the special case of a 2-D uniform sampling grid[1].

2. The complex beam patterns $g_m(\theta, \alpha)$ are **either**
   a) expressed via a separable function over the direction cosines $\mu^{(r)}$ corresponding to the two array dimensions, cf. (9.5) and (9.6) **or**

[1] Hexagonal arrays are another example of non-separable 2-D arrays. Therefore, we cannot apply tensor calculus there, even though the application of 3-D Unitary ESPRIT is possible, as we have shown in [RH05, RH06].

b) equal for all antenna elements, i.e., $g_m(\theta, \alpha) = g(\theta, \alpha)$ for $m = 1, 2, \ldots, M$ (including their spatial orientation in the array). In the latter case, separability is not needed, as shown in Appendix D.2.

Note that the if the 2-D array does not obey the two assumptions, its two dimensions cannot be separated and we have to stack the elements in one mode of our measurement tensor.

As explained in Example 9.2.2, another way to obtain two spatial dimensions is to consider the transmitter and the receiver jointly, measuring the channel gain between all pairs of transmit and receive antennas. In this case, the vectors $\boldsymbol{a}^{(r)}$ correspond directly to the array steering vectors of the transmit and the receive array (cf. (9.13)), respectively. To obtain $\boldsymbol{a}^{(r)}$ as a function of the desired parameters, e.g., the azimuth angle θ, measured beam patterns can be interpolated. For a discussion on efficient storage and interpolation of measured beam patterns, the reader is referred to [TLRT05].

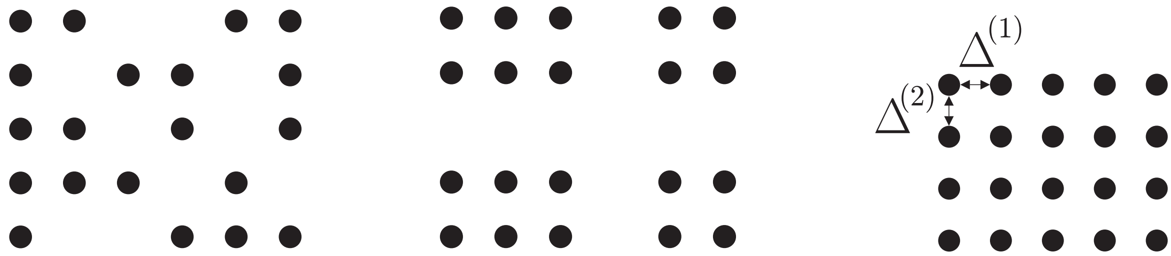
A simpler approach to compute $\boldsymbol{a}^{(r)}$ is to impose a model. For instance, if the antenna elements are assumed to be aligned uniformly, we can write

$$\boldsymbol{a}^{(r)}(\theta) = \begin{bmatrix} g_1(\theta) \\ g_2(\theta) \cdot \mathrm{e}^{\jmath \cdot \mu^{(r)}(\theta)} \\ g_3(\theta) \cdot \mathrm{e}^{\jmath \cdot 2 \cdot \mu^{(r)}(\theta)} \\ \vdots \\ g_{M_r}(\theta) \cdot \mathrm{e}^{\jmath \cdot (M_r - 1) \cdot \mu^{(r)}(\theta)} \end{bmatrix} \in \mathbb{C}^{M_r \times 1}, \tag{9.18}$$

where $g_{m_r}(\theta) \in \mathbb{C}$ represents the complex beam pattern of the m_r-th element as a function of the azimuth angle θ and the first element is chosen as a phase reference. Moreover, $\mu^{(r)}(\theta) = 2\pi \frac{\Delta}{\lambda_c} \cos(\theta)$, where Δ is the spacing between elements. Note that if all elements have the same beam pattern, i.e., $g_1(\theta) = \ldots = g_{M_r}(\theta)$, the beam pattern can be moved in front of the array steering vector as a scalar quantity, i.e.,

$$\boldsymbol{a}^{(r)}(\theta) = g(\theta) \cdot \bar{\boldsymbol{a}}^{(r)}(\theta) = g(\theta) \cdot \begin{bmatrix} 1 \\ \mathrm{e}^{\jmath \cdot \mu^{(r)}(\theta)} \\ \mathrm{e}^{\jmath \cdot 2 \cdot \mu^{(r)}(\theta)} \\ \vdots \\ \mathrm{e}^{\jmath \cdot (M_r - 1) \cdot \mu^{(r)}(\theta)} \end{bmatrix} \in \mathbb{C}^{M_r \times 1}. \tag{9.19}$$

In this case, the "remaining" array steering vector $\bar{\boldsymbol{a}}^{(r)}(\theta)$ has a Vandermonde structure. Arrays that obey (9.19) are referred to as Uniform Linear Arrays (ULAs). In the special case where the antennas are assumed to be omnidirectional we have $g(\theta) = 1 \ \forall \theta$. Note that in this



(a) Not a separable 2-D grid (b) Separable 2-D grid (c) Uniform 2-D grid

Figure 9.3.: Examples of 2-D sampling grids: (a) not a separable 2-D sampling grid; (b) separable 2-D sampling grid composed of the outer product of two (non-uniform) linear arrays; (c) uniform separable 2-D grid (URA).

case the azimuth angle cannot be uniquely recovered since $\mu(\theta) = \mu(-\theta)$ and hence the array steering vector for an impinging wavefront from the azimuth angle θ is indistinguishable from the array steering vector for an impinging wavefront from the azimuth angle $-\theta$.

Another important special case are antenna arrays that are invariant under mirroring around their centroids. Such arrays are called centro-symmetric [XRK94]. For instance, ULAs and URAs are centro-symmetric arrays[2]. Moreover, the 2-D antenna configuration depicted in Figure 9.3b is centro-symmetric in vertical direction but not centro-symmetric in horizontal direction (and hence not 2-D centro-symmetric). Mathematically, the arrays steering matrix $\boldsymbol{A} \in \mathbb{C}^{M \times d}$ for an R-D array that is R-D centro-symmetric satisfies

$$\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* = \boldsymbol{A} \cdot \boldsymbol{\Delta}, \tag{9.20}$$

where $\boldsymbol{\Delta} \in \mathbb{C}^{d \times d}$ is a unitary diagonal matrix which depends on the choice of the phase center of the array.[3]

9.2.4. Strict-sense non-circular sources

Up to here we have not made any further assumptions about the amplitudes of the multi-dimensional signals, which we collect in the matrix $\boldsymbol{S}$, except for the fact that the rank of $\boldsymbol{S}$ should be equal to d. However, as we discuss in Section 11.5 and 11.6, via simple modi-

[2]If the elements are not omnidirectional this requires that their complex beam patterns also satisfy $g_{M-m+1}(\theta,\alpha) = g_m^*(\theta,\alpha)$. As, shown in Appendix D.2 this restriction does not apply if all elements have the same beampattern, i.e., $g_m(\theta,\alpha) = g(\theta,\alpha)$ for $m = 1, 2, \ldots, M$, where $g(\theta,\alpha)$ can be arbitrary.

[3]If the phase center coincides with the array's centroid we have $\boldsymbol{\Delta} = \boldsymbol{I}_d$.

fications of ESPRIT-type algorithms[4] [HR04, RH09b] we can take advantage of a particular structure in these amplitudes referred to as second-order non-circularity. This occurs for instance in communication-type scenarios (cf. Example 9.2.1) where the transmitters employ specific modulation schemes, such as BPSK, ASK, MSK, or OQPSK.

A full statistical description of complex random variables includes not only the individual distribution of their real and imaginary parts but also the joint distribution since they might be correlated. A simplifying assumption that is often made is to consider second-order *circularly symmetric* complex random variables. A zero mean complex random variable $Z = X + \jmath Y$ is said to be second-order circularly symmetric if it satisfies $\mathbb{E}\left\{Z^2\right\} = 0$, which implies that real part and imaginary part are uncorrelated and have the same variance. Consequently, if $\mathbb{E}\left\{Z^2\right\} \neq 0$, the random variable Z is (second-order) non-circular. We can measure the degree of non-circularity via a scalar parameter ζ referred to as the "non-circularity rate" [DA04], "circularity coefficient" [EK06], or "circularity quotient" [Oll08], which is defined as

$$\zeta = \frac{\mathbb{E}\left\{Z^2\right\}}{\mathbb{E}\left\{|Z|^2\right\}}. \tag{9.21}$$

It can be shown that $|\zeta| \leq 1$. A random variable with $0 < |\zeta| < 1$ is called (second-order) *weak-sense* non-circular, for $|\zeta| = 1$ we speak of (second-order) *strict-sense* non-circularity [RH07a]. Strict-sense non-circular random variables are sometimes also referred to as rectilinear [CP06]. Note that strict-sense non-circularity implies a linear dependence between real and imaginary part of Z. We can think of Z as a real-valued random variable which is rotated by a complex phase term, i.e., $Z = W \cdot e^{\jmath\varphi}$, where $W \in \mathbb{R}$ is a random variable and φ is deterministic (fixed). In a communication system, the amplitudes $s_i[n]$ are non-circular random variables if the symbols are drawn from constellations which are not circularly symmetric. We obtain strict-sense non-circular (rectilinear) amplitudes if the transmitters use real-valued constellations (such as BPSK or ASK), which appear rotated by complex phase terms at the receiver since each transmitter may have a different transmission delay. Note that OQPSK and MSK symbols can be transformed into rectilinear amplitudes by applying an appropriate derotation at the receiver [CP06]. Figure 9.4 shows an example of an I/Q diagram displaying Inphase vs. Quadrature (I/Q) components of the received symbols for two users that transmit using a real-valued constellation. Since each user's transmission undergoes a different phase rotation (φ_i), the receiver observes rotated real-valued random variables that satisfy the strict-sense non-circularity property. For the symbol matrix $\boldsymbol{S} \in \mathbb{C}^{d \times N}$ this implies that it can be written

[4] As discussed in Section 9.1, many other subspace-based parameter estimation schemes can benefit from non-circular sources as well [CWS01, ZCW03, AD06, LLXZ12].

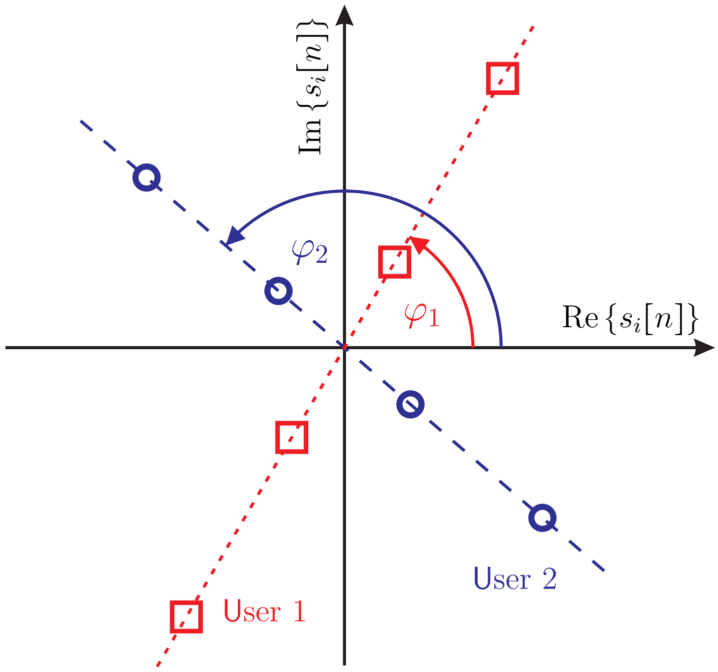


Figure 9.4.: Example for strict-sense non-circular amplitudes: Two users (red, blue) transmit symbols drawn from real-valued constellations. Since they undergo different phase rotations, the I/Q diagram at the receiver consists of differently rotated real-valued random variables, i.e., the complex symbols $s_i[n]$ can be described as strict-sense non-circular random variables.

as [ZCW03]

$$\boldsymbol{S} = \boldsymbol{\Psi} \cdot \boldsymbol{S}_0, \tag{9.22}$$

where $\boldsymbol{S}_0 \in \mathbb{R}^{d \times N}$ and $\boldsymbol{\Psi} = \mathrm{diag}\{[\mathrm{e}^{\jmath\varphi_1}, \ldots, \mathrm{e}^{\jmath\varphi_d}]\}$.

Non-circular random variables can be exploited in signal processing applications since they carry a specific structure. If $\boldsymbol{s}[n]$ is a non-circular random variable, in addition to the covariance matrix $\boldsymbol{R}_{\mathrm{ss}} = \mathbb{E}\left\{\boldsymbol{s}[n] \cdot \boldsymbol{s}[n]^{\mathrm{H}}\right\}$, the pseudo-covariance matrix $\tilde{\boldsymbol{R}}_{\mathrm{ss}} = \mathbb{E}\left\{\boldsymbol{s}[n] \cdot \boldsymbol{s}[n]^{\mathrm{T}}\right\}$ contains statistical information about $\boldsymbol{s}[n]$ we can take advantage of. For circular random variables, the pseudo-covariance matrix is equal to the zero matrix.

10. Subspace estimation

The first step of subspace-based parameter estimation is to find an estimate of the signal subspace. The **signal subspace** is defined as the span (i.e., the set of all linear combinations) of the columns of the array steering matrix $\boldsymbol{A}$ (cf. Section 9.2). Its orthogonal complement in $\mathbb{C}^M$ is referred to as the **noise subspace**.

In this section we show how the multidimensional structure of R-dimensional signals can be exploited to enhance the accuracy of the signal subspace estimate. This enhancement is achieved by enforcing the specific structure of the multidimensional signals onto the unstructured subspace estimate. Note that this enhanced signal subspace estimate can be combined with arbitrary multi-dimensional subspace-based parameter estimation schemes, e.g., R-D MODE, R-D RARE, R-D MUSIC[1], or R-D ESPRIT. We also provide a strong link between the matrix-based and the tensor-based subspace estimate that provides a more clear intuition where the improvement comes from and in which scenarios it is absent.

We begin by reviewing matrix-based subspace estimation in Section 10.1. In Section 10.2 we introduce the enhanced tensor-based subspace estimate and prove its link to the matrix-based subspace estimate via a structured projection. Section 10.3 discusses forward-backward averaging and real-valued subspace estimation in the matrix and the tensor case. Finally, a summary is provided in Section 10.4.

10.1. Matrix-based subspace estimation

As explained in Section 9.2, the N snapshots from the R-D signal can be represented via an $M \times N$ matrix $\boldsymbol{X}$ where each column represents one snapshot and the R dimensions are stacked along its rows. This matrix then obeys the data model (9.16) that we restate here for convenience

$$\boldsymbol{X} = \underbrace{\boldsymbol{A} \cdot \boldsymbol{S}}_{\boldsymbol{X}_0} + \boldsymbol{N}, \tag{10.1}$$

[1]This R-D extension differs from the existing R-D MUSIC [MLM05] which, as explained in Section 9.1, is based on eigentensors obtained from an HOEVD of the covariance tensor estimated from the vector-sensor observation matrices.

where $\boldsymbol{X} = [\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times N}$, $\boldsymbol{A} = [\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times d}$, and $\boldsymbol{N} = [\boldsymbol{\mathcal{N}}]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times N}$. Obviously, the noise-free part of the data $\boldsymbol{X}_0$ in (10.1) is at most rank-d. Therefore, we perform a rank-d-approximation of $\boldsymbol{X}$ via the truncated SVD, which is optimal in the Frobenius norm sense [EY36], cf. (3.50). The column space of this matrix is spanned by the d dominant left singular vectors of $\boldsymbol{X}$ and it is an estimate of the true signal subspace which is spanned by the columns $\boldsymbol{A}$. Note that since $\boldsymbol{X}_0 = \boldsymbol{A} \cdot \boldsymbol{S}$, the true signal subspace is identical to the span of the d dominant left singular vectors of $\boldsymbol{X}_0$, i.e., all columns of $\boldsymbol{X}_0$ lie in the signal subspace. The SVDs of $\boldsymbol{X}_0$ and $\boldsymbol{X}$ are expressed as

$$\boldsymbol{X}_0 = \begin{bmatrix} \boldsymbol{U}_{\mathrm{s}}, & \boldsymbol{U}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_{\mathrm{s}} & \boldsymbol{0}_{d \times (N-d)} \\ \boldsymbol{0}_{(M-d) \times d} & \boldsymbol{0}_{(M-d) \times (N-d)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{V}_{\mathrm{s}}, & \boldsymbol{V}_{\mathrm{n}} \end{bmatrix}^{\mathrm{H}} \tag{10.2}$$

$$\boldsymbol{X} = \begin{bmatrix} \hat{\boldsymbol{U}}_{\mathrm{s}}, & \hat{\boldsymbol{U}}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{\mathrm{s}} & \boldsymbol{0}_{d \times (N-d)} \\ \boldsymbol{0}_{(M-d) \times d} & \hat{\boldsymbol{\Sigma}}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{V}}_{\mathrm{s}}, & \hat{\boldsymbol{V}}_{\mathrm{n}} \end{bmatrix}^{\mathrm{H}}. \tag{10.3}$$

Here, $\boldsymbol{U}_{\mathrm{s}} \in \mathbb{C}^{M \times d}$ and $\boldsymbol{U}_{\mathrm{n}} \in \mathbb{C}^{M \times (M-d)}$ are orthonormal basis for the signal subspace and the noise subspace, and $\hat{\boldsymbol{U}}_{\mathrm{s}} \in \mathbb{C}^{M \times d}$ and $\hat{\boldsymbol{U}}_{\mathrm{n}} \in \mathbb{C}^{M \times (M-d)}$ are their estimates obtained from $\boldsymbol{X}$. Moreover, $\boldsymbol{\Sigma}_{\mathrm{s}} = \mathrm{diag}\left\{\left[\sigma_1, \sigma_2, \dots, \sigma_d\right]\right\} \in \mathbb{R}^{d \times d}$ contains the d non-zero singular values on its main diagonal.

10.2. Tensor-based subspace estimation

To find a subspace estimate that takes the natural tensor structure into account (as discussed in Section 9.1) we employ a multi-dimensional extension of the SVD in form of a suitable tensor decomposition. As discussed in Section 4.2, depending on whether diagonality or unitarity should be preserved, different multilinear extensions of the SVD can be defined. We choose the Higher-Order SVD (HOSVD) since it is easily computed via SVDs of the unfoldings of the tensor. Moreover, the truncated HOSVD shown in (4.29) allows for multilinear low-rank approximation in a manner similar to the truncated SVD (for a discussion on the optimal multilinear rank-reduction via the HOOI algorithm vs. the truncated HOSVD see Section 4.2.1).

Let $\boldsymbol{\mathcal{X}}_0$ be the noise-free observation, such that $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}}$. Then, the SVD of the r-th unfolding of $\boldsymbol{\mathcal{X}}_0$ and $\boldsymbol{\mathcal{X}}$ can be expressed as

$$[\boldsymbol{\mathcal{X}}_0]_{(r)} = \begin{bmatrix} \boldsymbol{U}_r^{[\mathrm{s}]}, & \boldsymbol{U}_r^{[\mathrm{n}]} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_r^{[\mathrm{s}]} & \boldsymbol{0}_{d \times (N-d)} \\ \boldsymbol{0}_{(M-d) \times d} & \boldsymbol{0}_{(M-d) \times (N-d)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{V}_r^{[\mathrm{s}]}, & \boldsymbol{V}_r^{[\mathrm{n}]} \end{bmatrix}^{\mathrm{H}} \tag{10.4}$$

$$[\boldsymbol{\mathcal{X}}]_{(r)} = \left[\hat{\boldsymbol{U}}_r^{[\mathrm{s}]}, \quad \hat{\boldsymbol{U}}_r^{[\mathrm{n}]}\right] \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_r^{[\mathrm{s}]} & \boldsymbol{0}_{d\times(N-d)} \\ \boldsymbol{0}_{(M-d)\times d} & \hat{\boldsymbol{\Sigma}}_r^{[\mathrm{n}]} \end{bmatrix} \cdot \left[\hat{\boldsymbol{V}}_r^{[\mathrm{s}]}, \quad \hat{\boldsymbol{V}}_r^{[\mathrm{n}]}\right]^{\mathrm{H}}, \tag{10.5}$$

where $\boldsymbol{U}_r^{[\mathrm{s}]} \in \mathbb{C}^{M_r \times p_r}$ and $\boldsymbol{U}_r^{[\mathrm{n}]} \in \mathbb{C}^{M_r \times (M_r - p_r)}$ denote the basis for the r-space and its orthogonal complement, respectively. Here, p_r denotes the r-rank[2] of $\boldsymbol{\mathcal{X}}_0$. Based on the r-spaces $\boldsymbol{U}_r^{[\mathrm{s}]}$ we can estimate the core tensor $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \in \mathbb{C}^{p_1 \times \ldots \times p_R \times p_{R+1}}$ via

$$\boldsymbol{\mathcal{S}}^{[\mathrm{s}]} = \boldsymbol{\mathcal{X}}_0 \times_1 \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}]^{\mathrm{H}}} \times_{R+1} \boldsymbol{U}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}} \tag{10.6}$$

$$\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} = \boldsymbol{\mathcal{X}} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]^{\mathrm{H}}} \ldots \times_R \hat{\boldsymbol{U}}_R^{[\mathrm{s}]^{\mathrm{H}}} \times_{R+1} \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}}. \tag{10.7}$$

The truncated HOSVD then reads as

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{S}}^{[\mathrm{s}]} \times_1 \boldsymbol{U}_1^{[\mathrm{s}]} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}]} \times_{R+1} \boldsymbol{U}_{R+1}^{[\mathrm{s}]} \tag{10.8}$$

$$\boldsymbol{\mathcal{X}} \approx \hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \ldots \times_R \hat{\boldsymbol{U}}_R^{[\mathrm{s}]} \times_{R+1} \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{X}}}. \tag{10.9}$$

If we compare the truncated HOSVD of $\boldsymbol{\mathcal{X}}$ in (10.9) with the truncated SVD in (10.3), we observe that a unique feature of the HOSVD is that it performs low-rank approximations in all $R+1$ modes. Hence, the multilinear structure is exploited to perform more efficient denoising.

As a multilinear extension of the subspace estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ we introduce the following *signal subspace tensor*[3] $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \in \mathbb{C}^{M_1 \times \ldots \times M_R \times d}$

$$\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \ldots \times_R \hat{\boldsymbol{U}}_R^{[\mathrm{s}]} \times_{R+1} \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}}. \tag{10.10}$$

A formal link between $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}$ and $\hat{\boldsymbol{U}}_{\mathrm{s}}^{[\mathrm{s}]}$ is given by the following theorem:

Theorem 10.2.1. *The HOSVD-based signal subspace estimate* $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ *can be computed from the SVD-based subspace estimate* $\hat{\boldsymbol{U}}_{\mathrm{s}}$ *via the following relation*

$$\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = \left(\hat{\boldsymbol{T}}_1 \otimes \hat{\boldsymbol{T}}_2 \otimes \ldots \hat{\boldsymbol{T}}_R\right) \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}, \tag{10.11}$$

[2]In practice, we can estimate the r-ranks individually via a model order selection scheme operating on all unfoldings individually. Alternatively, we can apply tensor-based model order selection schemes [DHRD07, DRHdS11] to estimate d and then use $p_r = \min(M_r, d)$.

[3]Note that in [RHD06] and [HRD08], $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}$ was defined without the multiplication by $\hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}}$ in the $(R+1)$-th mode. While this has no impact on the subspace of interest, we include it here since it simplifies the notation in this section and in Chapters 11 and 12.

where $\hat{\boldsymbol{T}}_r \in \mathbb{C}^{M_r \times M_r}$ *represent estimates of the projection matrices onto the r-spaces of* $\boldsymbol{\mathcal{X}}_0$*, which are computed via* $\hat{\boldsymbol{T}}_r = \hat{\boldsymbol{U}}_r^{[\mathrm{s}]} \hat{\boldsymbol{U}}_r^{[\mathrm{s}]^{\mathrm{H}}}$.

Proof: cf. Appendix D.3. Note that a special case of this theorem for $R = 2$ was first shown by us in [RBHW09].

It is worth pointing out that (10.11) provides some rather interesting insights. Firstly, it shows that the matrix $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} \in \mathbb{C}^{M \times d}$ yields an estimate for the signal subspace, which can be used to replace the matrix $\hat{\boldsymbol{U}}_{\mathrm{s}}$. Secondly, it demonstrates that an explicit computation of the core tensor of $\boldsymbol{\mathcal{X}}$ is actually not necessary if only the HOSVD-based subspace estimate is needed. Thirdly, it shows that the HOSVD-based subspace estimate can be seen as the projection of the (unstructured) matrix-based subspace estimate onto the Kronecker structure inherent in the data and that this projection is achieved by virtue of the Kronecker product of r-space projection matrices. Since this projection leaves the desired signal unaltered (cf. Corollary D.3.2) it affects only the noise, filtering out the part of the noise which does not obey the required Kronecker structure. This observation provides a different way of understanding the denoising obtained via multilinear rank reduction. The relation (10.11) also shows that for any mode r where $d \geq M_r$ we have $\hat{\boldsymbol{T}}_r = \boldsymbol{I}_r$ and hence no performance improvement can be obtained in this particular mode r. As a corollary from this we have $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = \hat{\boldsymbol{U}}_{\mathrm{s}}$ if $d \geq \max_{r=1,2,\ldots,R}(M_r)$, i.e., there is no improvement in terms of the subspace estimation accuracy from the HOSVD-based subspace estimate if the number of wavefront d is greater than or equal to the number of sensors in all R modes.

Note that we can even use (10.11) as a basis for establishing algorithms that track an HOSVD-based subspace estimate adaptively: all we need to do is to track the subspaces of all $(R+1)$ unfoldings of the tensor in parallel and use (10.11) to construct an overall subspace estimate. Since all unfoldings are matrices, arbitrary matrix-based subspace tracking schemes can be used for this task.

10.3. Forward-Backward Averaging and Real-Valued Subspace Estimation

If the array is centro-symmetric, i.e., $\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* = \boldsymbol{A} \cdot \boldsymbol{\Delta}$ (cf. Section 9.2.3), we can apply Forward-Backward Averaging (FBA) to the data [EJS82, PK89a]. FBA uses a symmetry in the data to improve the estimation accuracy of the signal subspace. Moreover, two coherent sources can be decorrelated. A "direct data" approach that operates on the matrix $\boldsymbol{X}$ to create an additional set of N "virtual" snapshots was proposed in [HN95, Haa97a]. The augmented measurement matrix $\boldsymbol{X}^{(\mathrm{fba})}$ can be written as

$$\boldsymbol{X}^{(\mathrm{fba})} = \begin{bmatrix} \boldsymbol{X} & \boldsymbol{\Pi}_M \cdot \boldsymbol{X}^* \cdot \boldsymbol{\Pi}_N \end{bmatrix} \in \mathbb{C}^{M \times 2N}. \tag{10.12}$$

Note that $\boldsymbol{X}^{(\mathrm{fba})}$ has $2N$ columns, i.e., the number of snapshots has been virtually doubled. The main advantage of the direct data formulation in (10.12) is that $\boldsymbol{X}^{(\mathrm{fba})}$ is a centro-Hermitian matrix[4] and hence we can apply the one-to-one mapping between the set of centro-Hermitian matrices and the set of real-valued matrices from [Lee80]. In other words, the matrix

$$\varphi\left(\boldsymbol{X}^{(\mathrm{fba})}\right) = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{X}^{(\mathrm{fba})} \cdot \boldsymbol{Q}_{2N} = \mathcal{T}\left(\boldsymbol{X}\right) \tag{10.13}$$

is real-valued for unitary matrices $\boldsymbol{Q}_M$ that are left-$\boldsymbol{\Pi}$-real, i.e., they satisfy $\boldsymbol{Q}_M^* \cdot \boldsymbol{\Pi}_M = \boldsymbol{Q}_M$. Note that the transformation (10.13) can be efficiently implemented by considering sparse unitary left-$\boldsymbol{\Pi}$-real matrices [HN95]. These are denoted by $\boldsymbol{Q}_p^{(\mathrm{s})} \in \mathbb{C}^{d \times d}$ and shown in Appendix A.2. The notation $\mathcal{T}\left(\boldsymbol{X}\right)$ in (10.13) is introduced to simplify the application of both, FBA and the real-valued transformation. The advantage of (10.13) is that since the matrix is real-valued, a subspace estimate is obtained by a real-valued SVD, which has a lower computational complexity compared to the complex-valued counterpart. A real-valued signal subspace estimate $\hat{\boldsymbol{E}}_{\mathrm{s}} \in \mathbb{R}^{M \times d}$ is then obtained by collecting the d dominant left singular vectors of $\varphi\left(\boldsymbol{X}^{(\mathrm{fba})}\right)$ into a matrix. Based on $\hat{\boldsymbol{E}}_{\mathrm{s}}$, "unitary" versions of many DOA estimation algorithms can be defined, e.g., the Unitary ESPRIT algorithm discussed in Section 11.3.2.

Forward-Backward Averaging and the real-valued transformation can also be defined for the measurement tensor $\boldsymbol{\mathcal{X}}$. As we show in [HRD08] we can define

$$\boldsymbol{\mathcal{X}}^{(\mathrm{fba})} = \left[\boldsymbol{\mathcal{X}} \sqcup_{R+1} \boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N\right] \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R \times N} \tag{10.14}$$

$$\varphi\left(\boldsymbol{\mathcal{X}}^{(\mathrm{fba})}\right) = \boldsymbol{\mathcal{X}}^{(\mathrm{fba})} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} \times_{R+1} \boldsymbol{Q}_{2N}^{\mathrm{H}} = \mathcal{T}\left(\boldsymbol{\mathcal{X}}^{(\mathrm{fba})}\right) \in \mathbb{R}^{M_1 \times M_2 \ldots \times M_R \times N}, \tag{10.15}$$

similar to (10.12) and (10.13).

A preprocessing step that is not discussed in this thesis is spatial smoothing. Spatial smoothing can be applied to decorrelate coherent wavefronts [SWK85] by dividing the array into L identical displaced subarrays and averaging their spatial covariance matrices. As a result, L coherent wavefronts are decorrelated. However, the number of antenna elements is also reduced to $M - L + 1$. Spatial smoothing can be combined with forward-backward-averaging [PK89b], in

[4] A matrix $\boldsymbol{X} \in \mathbb{C}^{p \times q}$ is called centro-Hermitian if it satisfies $\boldsymbol{\Pi}_p \cdot \boldsymbol{X}^* \cdot \boldsymbol{\Pi}_q = \boldsymbol{X}$.

which case $2L$ coherent wavefronts are decorrelated when using L subarrays. Spatial smoothing is readily formulated in terms of tensors via the concatenation operator, as shown in [HRD08]. However, as shown in [THRG10, THG09b, THG09a], the tensor structure can be exploited further to derive tensor-based spatial smoothing techniques that outperform matrix-based approaches significantly.

10.4. Summary

In this chapter we have derived the tensor-based signal subspace estimate for R-dimensional signals sampled on an R-D grid. Extending the matrix-based subspace estimate obtained from an SVD of the measurement matrix, we have applied this idea to define a tensor-based subspace estimate computed from the HOSVD of the measurement tensor. This subspace estimate takes advantage of the R-D structure in the data to perform more efficient denoising and thus results in an improved subspace estimation accuracy. We have provided the link between the matrix-based and the tensor-based subspace estimates via a structured projection in Theorem 10.2.1. Note that the subspace can be combined with arbitrary multi-dimensional subspace-based parameter estimation schemes, e.g., R-D ESPRIT, R-D MUSIC, R-D MODE, or R-D RARE. It represents a generic approach to extend multi-dimensional subspace-based parameter algorithms to tensors and hence differs from existing approaches (e.g., the tensor-based MUSIC algorithm for vector-sensor arrays from [MLM05], as explained in Section 9.1). We have also briefly discussed forward-backward averaging and the real-valued subspace estimation both in the matrix as well as in the tensor case.

In the next section we introduce enhancements of R-D ESPRIT-type algorithms that are achieved by exploiting the multidimensional structure of the data further. We begin with Tensor-ESPRIT-type algorithms based on Least Squares which achieve an improvement in estimation accuracy from the enhanced subspace estimate. Subsequently, we identify further sources of improvement, e.g., a tensor-based scheme to solve the overdetermined shift invariance equations. The achieved improvements in terms of estimation accuracy are shown in numerical simulations.

11. ESPRIT-type parameter estimation schemes

11.1. Overview

The purpose of this chapter is to show how subspace-based parameter estimation schemes can be improved by exploiting the specific structure of the signal of interest. As an example, we choose the ESPRIT algorithm to demonstrate the enhancements, however, they can be applied to different multi-dimensional subspace-based parameter estimation schemes as well, e.g., MODE, RARE, or MUSIC (cf. Section 9.1 for a literature overview).

We first discuss how to exploit the structure of an R-dimensional signal sampled on a separable R-D grid as defined in Section 9.2. In this case, the HOSVD-based subspace estimate [RHD06, HRD08] shown in Section 10.2 can be used to improve the accuracy of subspace-based parameter estimation schemes. In Section 11.2, we argue that the shift invariance structure used for ESPRIT allows for a natural tensor formulation of the entire algorithm. We then demonstrate that these "Tensor-ESPRIT"-type algorithms (introduced in Section 11.4) are algebraically equivalent to the existing matrix-based R-D ESPRIT algorithms (reviewed in Section 11.3) if the SVD-based subspace estimate is replaced by the HOSVD-based subspace estimate [HRD08].

Secondly, we show that if the amplitudes of the signals have a special structure which we refer to as strict-sense non-circularity (cf. Section 9.2.4), the array can be virtually doubled by augmenting the rows of the measurement matrix by their conjugates [ZCW03]. The resulting "NC ESPRIT"-type algorithms enjoy an enhanced estimation accuracy. Moreover, the maximum number of wavefronts which can be resolved jointly is doubled as well [HR04]. Compared to our original publication [HR04], the discussion of NC ESPRIT-type algorithms shown in Section 11.5 is more general since we show that we do not need to assume centro-symmetric antenna arrays. Moreover, we show that NC Standard ESPRIT and NC Unitary ESPRIT provide an identical performance.

Combining the ideas of Tensor-ESPRIT and NC ESPRIT is not a trivial task since the augmentation performed for NC ESPRIT destroys the separable R-D sampling grid required for Tensor-ESPRIT. We therefore consider a different form of augmentation in each of the R dimensions of the tensor [RH09b] and show that they can be combined. Based on these ideas we introduce NC Tensor-ESPRIT-type algorithms in Section 11.6 and demonstrate their superior estimation accuracy numerically. Compared to [RH09b], Section 11.6 extends the available results by performing rigorous proofs and discussing non-centro-symmetric arrays as well. Moreover, we also show that NC Standard Tensor-ESPRIT and NC Unitary Tensor-ESPRIT provide an identical performance.

A final step to exploit more of the rich structure in the multi-dimensional data is the solution of the shift invariance equations which are the basis for all ESPRIT-type methods. Since they represent an overdetermined set of equations, we seek for an approximate solution that minimizes a suitably defined error function. By modifying the unstructured ("Least Squares (LS)") error to take into account the shift invariance structure, we can define a "Structured Least Squares (SLS)"-type cost function [Haa97b] which we show in Section 11.7. Since it resembles a quadratic LS problem it can be solved iteratively via a sequence of linear LS problems. We demonstrate that the tensor structure can be incorporated to enhance SLS even further in the R-D case, resulting in a "Tensor-Structure Structured Least Squares (TS-SLS)" algorithm [RH07b].

Section 11.8 provides numerical simulation results demonstrating the performance of the various algorithms discussed in this Chapter. Finally, conclusions are drawn in Section 11.9.

11.2. R-D shift invariance

ESPRIT is based on a "shift invariance" in the array, which means that the array can be divided into two subarrays that are identical except for a displacement, as illustrated in Figure 11.1. Such a structure can be used to estimate the frequencies of R-dimensional harmonic signals efficiently [HN98]. Since for a harmonic wave, a displacement incurs a phase offset proportional to the frequency of the wave, frequency estimates are obtained by estimating the phase offsets for all waves.

To estimate the frequencies of an R-D harmonic wave in all dimensions[1] in this manner, we require a shift invariance of the sampling grid in all R dimensions. This can be expressed via tensor calculus in a natural way. Let $\boldsymbol{\mathcal{A}}_i \in \mathbb{C}^{M_1 \times M_2 \ldots \times M_R}$ be the "array steering tensor" of the i-th wavefront as defined in (9.13). Then, the shift invariance of $\boldsymbol{\mathcal{A}}_i$ in the r-th mode can be expressed as

$$\left(\boldsymbol{\mathcal{A}}_i \times_r \boldsymbol{J}_1^{(r)}\right) \cdot \mathrm{e}^{\jmath \mu_i^{(r)}} = \boldsymbol{\mathcal{A}}_i \times_r \boldsymbol{J}_2^{(r)}, \quad , r = 1, 2, \ldots, R, \tag{11.1}$$

where $\boldsymbol{J}_1^{(r)}$ and $\boldsymbol{J}_2^{(r)} \in \mathbb{R}^{M_r^{(\mathrm{sel})} \times M_r}$ are the selection matrices which select the $M_r^{(\mathrm{sel})}$ out of

[1]If the signal is harmonic only in R' of the R dimensions ($R' < R$), we can apply R'-D ESPRIT in these modes and leave the others modes untouched. In this case the $R - R'$ "non-harmonic" dimensions simply provide additional snapshots (which we collect in mode $R + 1$ in our data model for simplicity).

M_r indices belonging to the first and the second subarray in the r-th mode, respectively. Moreover, $\mu_i^{(r)}$ is the spatial frequency of the i-th wavefront in the r-th mode, cf. Example 9.2.1 or Example 9.2.2 introduced in Section 9.2.

For the special case of a uniform sampling grid introduced in Section 9.2, we choose $\boldsymbol{J}_1^{(r)}$ and $\boldsymbol{J}_2^{(r)}$ to

$$\boldsymbol{J}_1^{(r)} = \left[\boldsymbol{I}_{M_r-1} \quad \boldsymbol{0}_{(M_r-1)\times 1}\right] \quad \boldsymbol{J}_2^{(r)} = \left[\boldsymbol{0}_{(M_r-1)\times 1} \quad \boldsymbol{I}_{M_r-1}\right] \tag{11.2}$$

such that $M_r^{(\mathrm{sel})} = M_r - 1$, which corresponds to maximally overlapping subarrays.

The shift invariance relation for a single source from (11.1) can be extended to consider all d sources jointly. We obtain [HRD08]

$$\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)}, \tag{11.3}$$

where $\boldsymbol{\mathcal{A}} = [\boldsymbol{\mathcal{A}}_1 \sqcup_{R+1} \boldsymbol{\mathcal{A}}_2 \sqcup_{R+1} \dots \sqcup_{R+1} \boldsymbol{\mathcal{A}}_d] = \boldsymbol{\mathcal{I}}_{R+1,d} \times_1 \boldsymbol{A}^{(1)} \dots \times_R \boldsymbol{A}^{(R)} \in \mathbb{C}^{M_1 \times M_2 \dots \times M_R \times d}$ is the array steering tensor (cf. (9.12) and (9.15)) and $\boldsymbol{\Phi}^{(r)} = \mathrm{diag}\left\{\left[\mathrm{e}^{\jmath \mu_1^{(r)}}, \dots, \mathrm{e}^{\jmath \mu_d^{(r)}}\right]\right\} \in \mathbb{C}^{d \times d}$.

Note that a matrix-based equivalent of (11.3) in terms of the array steering matrix $\boldsymbol{A} = [\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}}$ is found by considering the transpose of the $(R+1)$-mode unfolding of (11.3). Using (4.5) we obtain

$$\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{A} \cdot \boldsymbol{\Phi}^{(r)} = \tilde{\boldsymbol{J}}_2^{(r)} \cdot \boldsymbol{A} \quad \text{where} \tag{11.4}$$

$$\tilde{\boldsymbol{J}}_n^{(r)} = (\boldsymbol{I}_{M_1} \otimes \dots \otimes \boldsymbol{I}_{M_{r-1}}) \otimes \boldsymbol{J}_n^{(r)} \otimes (\boldsymbol{I}_{M_{r+1}} \otimes \dots \otimes \boldsymbol{I}_{M_R}), \quad n = 1, 2, \tag{11.5}$$

which coincides with the matrix-based shift invariance equations derived in [HN98].

In the 1-D case, the shift invariance equations simplify into

$$\boldsymbol{J}_1 \cdot \boldsymbol{A} \cdot \boldsymbol{\Phi} = \boldsymbol{J}_2 \cdot \boldsymbol{A}. \tag{11.6}$$

Figure 11.1 shows the 2-D shift invariance of the 5×4 separable 2-D sampling grid that was already introduced in Figure 9.3b. The left-hand side shows how to choose the selection matrices $\boldsymbol{J}_1^{(1)}$ and $\boldsymbol{J}_2^{(1)}$ for the first dimension (horizontal) and the right-hand side shows how to choose the selection matrices $\boldsymbol{J}_1^{(2)}$ and $\boldsymbol{J}_2^{(2)}$ for the second dimension (vertical), i.e.,

$$\boldsymbol{J}_1^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \boldsymbol{J}_2^{(1)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \boldsymbol{J}_1^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \boldsymbol{J}_2^{(2)} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

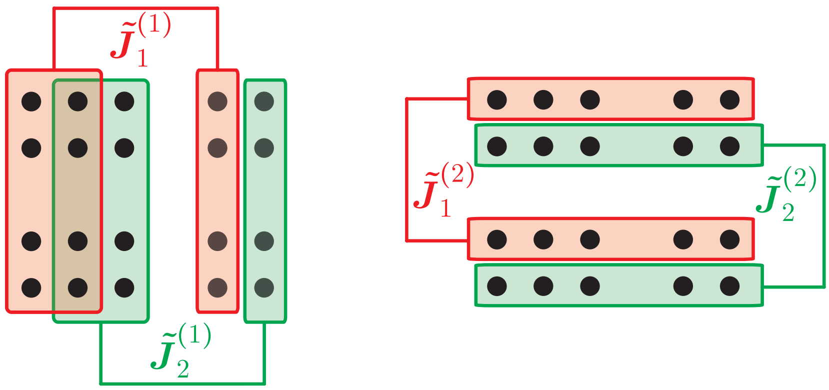


Figure 11.1.: 2-D shift invariance for the 5×4 separable 2-D sampling grid from Figure 9.3b. Left: subarrays for the first (horizontal) dimension, right: subarrays for the second (vertical) dimension.

Moreover, $\tilde{\boldsymbol{J}}_n^{(1)} = \boldsymbol{J}_n^{(1)} \otimes \boldsymbol{I}_4$ and $\tilde{\boldsymbol{J}}_n^{(2)} = \boldsymbol{I}_5 \otimes \boldsymbol{J}_n^{(2)}$ for $n = 1, 2$.

11.3. R-D matrix-based ESPRIT

11.3.1. R-D Standard ESPRIT

To solve the shift invariance equations (11.4) for the matrices $\boldsymbol{\Phi}^{(r)}$, we need to eliminate the unknown array steering matrix $\boldsymbol{A}$. This is achieved by observing that the columns of $\boldsymbol{A}$ represent a basis for the column space of $\boldsymbol{X}_0$ and hence each column of $\boldsymbol{A}$ can be expressed as a linear combination of the columns of the matrix of the d dominant left singular vectors of $\boldsymbol{X}_0$, which we denote as $\boldsymbol{U}_{\mathrm{s}}$ (cf. Section 10.1). Mathematically speaking, we can write $\boldsymbol{A} = \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{T}$ where $\boldsymbol{T} \in \mathbb{C}^{d \times d}$ is a full-rank matrix which contains the coefficients of the expansion of the columns of $\boldsymbol{A}$ with respect to the basis $\boldsymbol{U}_{\mathrm{s}}$. In practice, we estimate $\boldsymbol{U}_{\mathrm{s}}$ via an SVD of the noisy measurements $\boldsymbol{X}$. The estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ satisfies $\boldsymbol{A} \approx \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{T}$. Inserting this relation into (11.4), we have

$$\begin{aligned} \tilde{\boldsymbol{J}}_1^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{T} \cdot \boldsymbol{\Phi}^{(r)} &\approx \tilde{\boldsymbol{J}}_2^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{T} \\ \tilde{\boldsymbol{J}}_1^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \underbrace{\boldsymbol{T} \cdot \boldsymbol{\Phi}^{(r)} \cdot \boldsymbol{T}^{-1}}_{\boldsymbol{\Psi}^{(r)}} &\approx \tilde{\boldsymbol{J}}_2^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}, \end{aligned} \tag{11.7}$$

which is an overdetermined set of equations for $\boldsymbol{\Psi}^{(r)}$. An unstructured "Least Squares" solution of (11.7) for $\boldsymbol{\Psi}^{(r)}$ is obtained by (for a structured solution, cf. Section 11.7)

$$\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)} = \arg\min_{\boldsymbol{\Psi}} \left\| \tilde{\boldsymbol{J}}_1^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi} - \tilde{\boldsymbol{J}}_2^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \right\|_{\mathrm{F}}^2 = \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \right)^{+} \cdot \tilde{\boldsymbol{J}}_2^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}. \tag{11.8}$$

Since $\boldsymbol{\Psi}^{(r)} = \boldsymbol{T} \cdot \boldsymbol{\Phi}^{(r)} \cdot \boldsymbol{T}^{-1}$ represents an EigenValue Decomposition (EVD), we obtain an estimate of $\boldsymbol{\Phi}^{(r)}$ via an EVD of $\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)}$. To ensure the correct pairing across the dimensions, the matrices $\hat{\boldsymbol{\Phi}}^{(r)}$ should be estimated via a *joint* EVD of $\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)}$ (e.g., via [FG06]). Algorithm 3 summarizes the R-D Standard ESPRIT procedure.

Algorithm 3 Summary of R-D Standard ESPRIT using Least Squares.

1. Estimate the signal subspace $\hat{\boldsymbol{U}}_{\mathrm{s}}$ via the truncated SVD of the observation matrix $\boldsymbol{X} \in \mathbb{C}^{M \times N}$.

2. Solve the overdetermined shift invariance equations

$$\tilde{\boldsymbol{J}}_1^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi}^{(r)} \approx \tilde{\boldsymbol{J}}_2^{(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}$$

   for the matrices $\boldsymbol{\Psi}^{(r)}$ for $r = 1, 2, \ldots, R$ via the method of Least Squares (LS).

3. Compute the eigenvalues $\hat{\lambda}_i^{(r)}$ for $i = 1, 2, \ldots, d$ of $\hat{\boldsymbol{\Psi}}^{(r)}$ jointly for all $r = 1, 2, \ldots, R$, e.g., via the joint diagonalization scheme proposed in [FG06]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = \arg\left\{\hat{\lambda}_i^{(r)}\right\}$.

11.3.2. R-D Unitary ESPRIT

As discussed in Section 10.3, if the array is centro-symmetric (cf. examples given in Section 9.2.3) we can exploit the fact that $\boldsymbol{A}$ and $\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^*$ span the same column space. Therefore, the measurements $\boldsymbol{X} \in \mathbb{C}^{M \times N}$ can be augmented by $\boldsymbol{\Pi}_M \cdot \boldsymbol{X}^*$ along the columns without changing the column space, as shown in (10.12). This creates another set of N "virtual snapshots". Moreover, this step allows to decorrelate two coherent sources[2] . Finally, the redundancies in the resulting augmented measurement matrix can be used to transform the complex-valued measurement in the real-valued domain and perform the entire processing using real-valued

[2] The decorrelation relies on phase offsets between the sources and hence there are pathological cases were it fails, e.g., sources arriving in-phase (which means that their complex correlation coefficient is equal to 1 or -1) at an array where the phase reference is chosen in the center.

additions and multiplications only. The details of the derivation are found in [HN95, HN98]. Here we only provide a summary in Algorithm 4.

Algorithm 4 [HN98] Summary of R-D Unitary ESPRIT using Least Squares.

1. Estimate the real-valued signal subspace $\hat{\boldsymbol{E}}_{\mathrm{s}}$ via the truncated SVD of the transformed real-valued observation matrix $\mathcal{T}(\boldsymbol{X}) = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \begin{bmatrix} \boldsymbol{X} & \boldsymbol{\Pi}_M \boldsymbol{X}^* \boldsymbol{\Pi}_N \end{bmatrix} \cdot \boldsymbol{Q}_{2N} \in \mathbb{R}^{M \times 2N}$, where $\boldsymbol{Q}_p$ is a unitary $p \times p$ left-$\boldsymbol{\Pi}$-real matrix (i.e., $\boldsymbol{\Pi}_p \cdot \boldsymbol{Q}_p^* = \boldsymbol{Q}_p$), cf. Section 10.3.

2. Solve the overdetermined shift invariance equations
$$\tilde{\boldsymbol{K}}_1^{(r)} \cdot \hat{\boldsymbol{E}}_{\mathrm{s}} \cdot \boldsymbol{\Upsilon}^{(r)} \approx \tilde{\boldsymbol{K}}_2^{(r)} \cdot \hat{\boldsymbol{E}}_{\mathrm{s}}$$
for the matrices $\boldsymbol{\Upsilon}^{(r)}$ for $r = 1, 2, \ldots, R$ via the method of Least Squares (LS), where $\tilde{\boldsymbol{K}}_1^{(r)}$ and $\tilde{\boldsymbol{K}}_2^{(r)}$ are the transformed selection matrices given by
$$\tilde{\boldsymbol{K}}_1^{(r)} = 2 \cdot \mathrm{Re}\left\{ \boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})} \cdot M / M_r} \cdot \tilde{\boldsymbol{J}}_2^{(r)} \cdot \boldsymbol{Q}_M \right\} \tag{11.9}$$
$$\tilde{\boldsymbol{K}}_2^{(r)} = 2 \cdot \mathrm{Im}\left\{ \boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})} \cdot M / M_r} \cdot \tilde{\boldsymbol{J}}_2^{(r)} \cdot \boldsymbol{Q}_M \right\}. \tag{11.10}$$

3. Compute the eigenvalues $\hat{\omega}_i^{(r)}$ for $i = 1, 2, \ldots, d$ of $\hat{\boldsymbol{\Upsilon}}^{(r)}$ jointly for all $r = 1, 2, \ldots, R$, e.g., via the joint diagonalization scheme proposed in [FG06] or via the Simultaneous Schur Decomposition proposed in [HN98]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = 2 \cdot \arctan(\hat{\omega}_i^{(r)})$.

11.4. R-D Tensor-ESPRIT-type algorithms

11.4.1. R-D Standard Tensor-ESPRIT

As we have demonstrated in Section 11.2, the use of tensor algebra leads to a simplified and more natural formulation of the R-D shift invariance equations, since the artificial stacking operation and its consequences (such as the introduction of many Kronecker products) are avoided. Based on this idea, an R-D Standard ESPRIT algorithm can be formulated entirely in terms of tensors [HRD08]. As we show in the sequel, this enhances the estimation accuracy due to the improved tensor-based subspace estimate shown in Section 10.2. Moreover, it enables us to find tensor-based solutions to the overdetermined shift invariance equations [RH07b] which we discuss in Section 11.7.

We first eliminate the unknown array steering tensor from the shift invariance equations (11.3) by virtue of the signal subspace tensor (10.10). This step is facilitated by the following relation between $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$.

Theorem 11.4.1. **[HRD08]** *The array steering tensor* $\boldsymbol{\mathcal{A}}$ *and the signal subspace tensor* $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ *are related via*

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}}^{[\mathrm{s}]} \times_{R+1} \bar{\boldsymbol{T}}, \tag{11.11}$$

where $\bar{\boldsymbol{T}} \in \mathbb{C}^{d \times d}$ *is a non-singular transform matrix.*

Proof. Firstly, note that the true signal subspace is defined as the space spanned by the columns of $\boldsymbol{A}$. Moreover, we have $\boldsymbol{A} = [\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}}$ from (9.17). Let us compute the transpose of the $(R+1)$-mode unfolding of (11.11). Since $[\boldsymbol{\mathcal{A}}]_{(R+1)}^{\mathrm{T}} = \boldsymbol{A}$, (11.11) is equivalent to $\boldsymbol{A} = \left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} \cdot \bar{\boldsymbol{T}}^{\mathrm{T}}$. We also know that $\left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = \boldsymbol{U}_{\mathrm{s}} \cdot \tilde{\boldsymbol{T}}$, since $\left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ and $\boldsymbol{U}_{\mathrm{s}}$ span the same column space (cf. Corollary D.3.2). Likewise, we have $\boldsymbol{A} = \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{T}$ (cf. Section 11.3.1). Combining the last two relations we finally obtain $\boldsymbol{A} = \left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} \cdot \tilde{\boldsymbol{T}}^{-1} \cdot \boldsymbol{T}$ which coincides with the transpose of the $(R+1)$-mode unfolding of (11.11) for $\bar{\boldsymbol{T}}^{\mathrm{T}} = \tilde{\boldsymbol{T}}^{-1} \cdot \boldsymbol{T}$. □

Note that (11.11) was first stated by us in [HRD08], however, there it was not proven. Essentially, Theorem 11.4.1 shows that the row spaces of the $(R+1)$-mode unfoldings of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ agree. At the same time, if we compute any r-mode unfolding of (11.11) it becomes apparent that all the r-spaces of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ coincide for $r = 1, 2, \ldots, R$. Equation (11.11) allows to eliminate the unknown array steering tensor from the shift invariance equations, replacing it by the estimated signal subspace tensor $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}$ via $\boldsymbol{\mathcal{A}} \approx \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_{R+1} \bar{\boldsymbol{T}}$. We then obtain

$$\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Psi}^{(r)} \approx \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_2^{(r)}, \tag{11.12}$$

where[3] $\boldsymbol{\Psi}^{(r)} = \bar{\boldsymbol{T}}^{-1} \cdot \boldsymbol{\Phi}^{(r)} \cdot \bar{\boldsymbol{T}}, \quad r = 1, 2, \ldots, R$ follows by applying identity (4.7) for repeated n-mode products. Note that due to (4.7), the order of the matrices in the definition of $\boldsymbol{\Psi}^{(r)}$ is reversed compared to the matrix case shown in (11.7).

The next step is the solution of the overdetermined sets of equations (11.12) to yield the estimates $\hat{\boldsymbol{\Psi}}^{(r)}$. The following theorem shows how a Least Squares solution to (11.12) can be obtained in closed-form.

[3] Note that this is not exactly the same as the $\boldsymbol{\Psi}^{(r)}$ defined in Section 11.3.1, since the matrix of eigenvectors is different. However, since we are only interested in the eigenvalues, this difference is irrelevant, and hence we use the same variable for brevity.

Theorem 11.4.2. **[HRD08]** *The Least Squares solution of the tensor-valued shift invariance equation* (11.12) *has the following closed-form solution*

$$\hat{\mathbf{\Psi}}_{\mathrm{LS}}^{(r)} = \arg\min_{\mathbf{\Psi}} \left\| \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \mathbf{\Psi} - \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_2^{(r)} \right\|_{\mathrm{H}}^2 \tag{11.13}$$

$$\Rightarrow \hat{\mathbf{\Psi}}_{\mathrm{LS}}^{(r)^{\mathrm{T}}} = \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \right]_{(R+1)}^{\mathrm{T}} \right)^{+} \cdot \tilde{\boldsymbol{J}}_2^{(r)} \cdot \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \right]_{(R+1)}^{\mathrm{T}}. \tag{11.14}$$

Proof: cf. Appendix D.4.

Comparing (11.14) with (11.8) we see that the Least Squares solution of the matrix-based shift invariance equations for R-D Standard ESPRIT and the tensor-based shift invariance equations for R-D Standard Tensor-ESPRIT differ only in the choice of the subspace. Contemplating that the remaining steps (joint eigendecomposition of $\hat{\mathbf{\Psi}}_{\mathrm{LS}}^{(r)}$ to recover the frequencies $\mu_i^{(r)}$) are also the same, we can conclude that R-D Standard Tensor-ESPRIT is algebraically equivalent to R-D Standard ESPRIT if we replace the SVD-based subspace estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ by the HOSVD-based subspace estimate $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \right]_{(R+1)}^{\mathrm{T}}$.

11.4.2. R-D Unitary Tensor-ESPRIT

In the previous section we have seen that tensor calculus allows to derive a tensor-valued version of R-D Standard ESPRIT and that it is algebraically equivalent to matrix-based R-D Standard ESPRIT except for using the enhanced HOSVD-based subspace estimate.

We can proceed in a similar manner for R-D Unitary Tensor-ESPRIT. As discussed in Section 10.3, if the array is centro-symmetric, we can apply forward-backward averaging to the measurement tensor $\boldsymbol{\mathcal{X}}$ and then transform the resulting tensor onto the real-valued domain [HRD08] to lower the computational complexity. We can then estimate the signal subspace tensor via a truncated HOSVD of the transformed tensor $\mathcal{T}(\boldsymbol{\mathcal{X}}) \in \mathbb{R}^{M_1 \times \ldots \times M_R \times 2N}$ shown in (10.15), i.e.,

$$\begin{aligned} \mathcal{T}(\boldsymbol{\mathcal{X}}) &\approx \hat{\boldsymbol{\mathcal{S}}}_{\mathcal{T}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{E}}_1^{[\mathrm{s}]} \ldots \times_R \hat{\boldsymbol{E}}_R^{[\mathrm{s}]} \times_{R+1} \hat{\boldsymbol{E}}_{R+1}^{[\mathrm{s}]} \\ \Rightarrow \hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]} &= \hat{\boldsymbol{\mathcal{S}}}_{\mathcal{T}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{E}}_1^{[\mathrm{s}]} \ldots \times_R \hat{\boldsymbol{E}}_R^{[\mathrm{s}]} \times_{R+1} \mathbf{\Sigma}_{R+1}^{[\mathrm{s}]^{-1}} \end{aligned} \tag{11.15}$$

where $\hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]} \in \mathbb{R}^{M_1 \times \ldots \times M_R \times d}$. Applying the real-valued transformation in (10.15) to the shift invariance equations (11.12) yields the following transformed equations

$$\hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]} \times_r \boldsymbol{K}_1^{(r)} \times_{R+1} \mathbf{\Upsilon}^{(r)} \approx \hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]} \times_r \boldsymbol{K}_2^{(r)}, \tag{11.16}$$

where $\boldsymbol{K}_1^{(r)} = 2 \cdot \mathrm{Re}\left\{\boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2^{(r)} \cdot \boldsymbol{Q}_{M_r}\right\}$ and $\boldsymbol{K}_2^{(r)} = 2 \cdot \mathrm{Im}\left\{\boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2^{(r)} \cdot \boldsymbol{Q}_{M_r}\right\}$. The real-valued shift invariance equations in (11.16) have the same algebraic form as the shift invariance equations for R-D Standard Tensor-ESPRIT shown in (11.12). Consequently, using similar arguments as in Theorem 11.4.2 we find the closed-form Least Squares solution to (11.16) via

$$\hat{\boldsymbol{\Upsilon}}_{\mathrm{LS}}^{(r)^{\mathrm{T}}} = \left(\tilde{\boldsymbol{K}}_1^{(r)} \cdot \left[\hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)}\right)^{+} \cdot \tilde{\boldsymbol{K}}_2^{(r)} \cdot \left[\hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)}. \tag{11.17}$$

Note that as for R-D Standard Tensor-ESPRIT, for R-D Unitary Tensor-ESPRIT we again obtain a solution which is algebraically equivalent to the matrix-based R-D Unitary ESPRIT algorithm [HN98]. A slight difference is that the transformed selection matrices $\tilde{\boldsymbol{K}}_1^{(r)}$ and $\tilde{\boldsymbol{K}}_2^{(r)}$ used in (11.17) are given by [HRD08]

$$\tilde{\boldsymbol{K}}_n^{(r)} = (\boldsymbol{I}_{M_1} \otimes \ldots \otimes \boldsymbol{I}_{M_{r-1}}) \otimes \boldsymbol{K}_n^{(r)} \otimes (\boldsymbol{I}_{M_{r+1}} \otimes \ldots \otimes \boldsymbol{I}_{M_R}), \quad n = 1, 2, \tag{11.18}$$

and hence, they coincide with the matrices $\tilde{\boldsymbol{K}}_n^{(r)}$ defined in Algorithm 4 only if we choose the unitary left-$\boldsymbol{\Pi}$-real matrices $\boldsymbol{Q}_M$ and $\boldsymbol{Q}_{M_r^{(\mathrm{sel})} \cdot M/M_r}$ according to

$$\boldsymbol{Q}_M = \boldsymbol{Q}_{M_1} \otimes \boldsymbol{Q}_{M_2} \otimes \ldots \otimes \boldsymbol{Q}_{M_r} \otimes \ldots \otimes \boldsymbol{Q}_{M_R} \tag{11.19}$$

$$\boldsymbol{Q}_{M_r^{(\mathrm{sel})} \cdot M/M_r} = \boldsymbol{Q}_{M_1} \otimes \boldsymbol{Q}_{M_2} \otimes \ldots \otimes \boldsymbol{Q}_{M_r^{(\mathrm{sel})}} \otimes \ldots \otimes \boldsymbol{Q}_{M_R}, \tag{11.20}$$

where the smaller $\boldsymbol{Q}_{M_r}$ are arbitrary unitary left-$\boldsymbol{\Pi}$-real matrices. Moreover, the matrix $\boldsymbol{Q}_M$ used in the transformation $\mathcal{T}(\boldsymbol{X})$ from (10.13) should also be chosen as in (11.19). However, since the particular choice of the left-$\boldsymbol{\Pi}$-real matrices is irrelevant for the performance of R-D Unitary ESPRIT we again conclude that R-D Unitary ESPRIT and R-D Unitary Tensor-ESPRIT are algebraically equivalent except for the fact that the SVD-based subspace estimate $\boldsymbol{E}_{\mathrm{s}}$ is replaced by the HOSVD-based subspace estimate $\hat{\boldsymbol{\mathcal{E}}}^{[\mathrm{s}]}$.

11.5. R-D NC ESPRIT-type algorithms

11.5.1. R-D NC Standard ESPRIT

R-D NC Standard ESPRIT is applicable if the amplitudes $s_i[n]$ for $i = 1, 2, \ldots, d$ represent samples from a strict-sense non-circular distribution, as described in Section 9.2.4. This implies that they can be expressed as $s_i[n] = \mathrm{e}^{\jmath \varphi_i} \cdot s_{0,i}[n]$, where $s_{0,i}[n] \in \mathbb{R}$ and φ_i does not change with time (n). For the matrix of amplitudes $\boldsymbol{S} \in \mathbb{C}^{d \times N}$ we can then write

$$\boldsymbol{S} = \boldsymbol{\Psi} \cdot \boldsymbol{S}_0, \quad \text{where} \quad \boldsymbol{\Psi} = \operatorname{diag}\left\{\left[\mathrm{e}^{\jmath\varphi_1}, \ldots, \mathrm{e}^{\jmath\varphi_d}\right]\right\} \tag{11.21}$$

and $\boldsymbol{S}_0 \in \mathbb{R}^{d \times N}$, cf. (9.22).

Based on this assumption we can define an augmented measurement matrix $\boldsymbol{X}^{(\mathrm{nc})}$ as [CWS01, HR04][4]

$$\boldsymbol{X}^{(\mathrm{nc})} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{X}^* \end{bmatrix}. \tag{11.22}$$

Inserting $\boldsymbol{X} = \boldsymbol{A} \cdot \boldsymbol{S} + \boldsymbol{N}$ and $\boldsymbol{S} = \boldsymbol{\Psi} \cdot \boldsymbol{S}_0$, we can rewrite (11.22) into

$$\boldsymbol{X}^{(\mathrm{nc})} = \begin{bmatrix} \boldsymbol{A} \cdot \boldsymbol{S} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{S}^* \end{bmatrix} + \begin{bmatrix} \boldsymbol{N} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{N}^* \end{bmatrix} \tag{11.23}$$

$$= \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^* \end{bmatrix} \cdot \boldsymbol{S} + \begin{bmatrix} \boldsymbol{N} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{N}^* \end{bmatrix} \tag{11.24}$$

$$= \boldsymbol{A}^{(\mathrm{nc})} \cdot \boldsymbol{S} + \boldsymbol{N}^{(\mathrm{nc})}, \tag{11.25}$$

since $\boldsymbol{S}^* = \boldsymbol{\Psi}^* \cdot \boldsymbol{S}_0$ and $\boldsymbol{\Psi}^* \cdot \boldsymbol{S} = \boldsymbol{S}_0$. Equation (11.25) shows that the desired signal component of the augmented $\boldsymbol{X}^{(\mathrm{nc})}$ can be factorized into an extended array steering matrix $\boldsymbol{A}^{(\mathrm{nc})} \in \mathbb{C}^{2M \times d}$ and the original matrix of amplitudes $\boldsymbol{S} \in \mathbb{C}^{d \times N}$. A remarkable property of $\boldsymbol{A}^{(\mathrm{nc})}$ is summarized in the following theorem:

Theorem 11.5.1. *If the array steering matrix $\boldsymbol{A}$ is shift-invariant, i.e., $\boldsymbol{J}_1 \cdot \boldsymbol{A} \cdot \boldsymbol{\Phi} = \boldsymbol{J}_2 \cdot \boldsymbol{A}$, where $\boldsymbol{J}_1$ and $\boldsymbol{J}_2 \in \mathbb{R}^{M^{(\mathrm{sel})} \times M}$ are the selection matrices for the first and the second subarray, then $\boldsymbol{A}^{(\mathrm{nc})}$ satisfies*

$$\boldsymbol{J}_1^{(\mathrm{nc})} \cdot \boldsymbol{A}^{(\mathrm{nc})} \cdot \boldsymbol{\Phi} = \boldsymbol{J}_2^{(\mathrm{nc})} \cdot \boldsymbol{A}^{(\mathrm{nc})} \quad \textit{where} \tag{11.26}$$

$$\boldsymbol{J}_1^{(\mathrm{nc})} = \begin{bmatrix} \boldsymbol{J}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{\Pi}_M \end{bmatrix} \quad \textit{and} \quad \boldsymbol{J}_2^{(\mathrm{nc})} = \begin{bmatrix} \boldsymbol{J}_2 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{\Pi}_M \end{bmatrix} \in \mathbb{R}^{2M^{(\mathrm{sel})} \times 2M}. \tag{11.27}$$

Proof: cf. Appendix D.5.

The shift invariance in (11.26) was already used in [ZCW03] and by us in [HR04] for the

[4][CWS01] defines $\boldsymbol{X}^{(\mathrm{nc})}$ for Root-MUSIC without the matrix $\boldsymbol{\Pi}_M$. The formulation in (11.22) we use here was first proposed by us in [HR04] to facilitate the real-valued implementation for Unitary ESPRIT.

special case of a ULA and the special case of a centro-symmetric array, respectively. Theorem 11.5.1 is more general since it does need further assumptions about the array except for the shift invariance. Note that Theorem 11.5.1 implies that via the augmentation we have created a virtual array of $2M$ sensors with two shift invariant subarrays containing $2M^{(\mathrm{sel})}$ sensors, as visualized in Figure 11.2. Consequently, this step doubles the number of sources that can be resolved simultaneously as well. Based on the shift invariance equation shown in (11.26) we can define an R-D Standard ESPRIT-type algorithm following the same steps as before. The resulting R-D NC Standard ESPRIT algorithm is summarized in Algorithm 5. A 1-D version of it was first proposed in [ZCW03].

Algorithm 5 Summary of R-D NC Standard ESPRIT using Least Squares.

1. Estimate the augmented signal subspace $\hat{\boldsymbol{U}}_{\mathrm{s}}^{(\mathrm{nc})} \in \mathbb{R}^{2M\times d}$ via the truncated SVD of the augmented observation matrix $\boldsymbol{X}^{(\mathrm{nc})} \in \mathbb{C}^{2M\times N}$.

2. Solve the overdetermined shift invariance equations

$$\tilde{\boldsymbol{J}}_1^{(\mathrm{nc})(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}^{(\mathrm{nc})} \cdot \boldsymbol{\Psi}^{(r)} \approx \tilde{\boldsymbol{J}}_2^{(\mathrm{nc})(r)} \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}^{(\mathrm{nc})}$$

for the matrices $\boldsymbol{\Psi}^{(r)}$ for $r = 1,2,\ldots,R$ via the method of Least Squares (LS), where $\tilde{\boldsymbol{J}}_1^{(\mathrm{nc})(r)}$ and $\tilde{\boldsymbol{J}}_2^{(\mathrm{nc})(r)}$ are defined as (cf. (11.27))

$$\tilde{\boldsymbol{J}}_n^{(\mathrm{nc})(r)} = \boldsymbol{I}_{M_1\cdot\ldots\cdot M_{r-1}} \otimes \boldsymbol{J}_n^{(\mathrm{nc})(r)} \otimes \boldsymbol{I}_{M_{r+1}\cdot\ldots\cdot M_R} \tag{11.28}$$

$$\boldsymbol{J}_1^{(\mathrm{nc})(r)} = \begin{bmatrix} \boldsymbol{J}_1^{(r)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2^{(r)} \cdot \boldsymbol{\Pi}_{M_r} \end{bmatrix} \tag{11.29}$$

$$\boldsymbol{J}_2^{(\mathrm{nc})(r)} = \begin{bmatrix} \boldsymbol{J}_2^{(r)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \cdot \boldsymbol{J}_1^{(r)} \cdot \boldsymbol{\Pi}_{M_r} \end{bmatrix}. \tag{11.30}$$

3. Compute the eigenvalues $\hat{\lambda}_i^{(r)}$ for $i = 1,2,\ldots,d$ of $\hat{\boldsymbol{\Psi}}^{(r)}$ jointly for all $r = 1,2,\ldots,R$, e.g., via the joint diagonalization scheme proposed in [FG06]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = \arg\left\{\hat{\lambda}_i^{(r)}\right\}$.

11.5.2. R-D NC Unitary ESPRIT

The extension of R-D NC Standard ESPRIT to R-D NC Unitary ESPRIT [HR04] is again quite straightforward. There are three remarkable things to note here though. Firstly, while

R-D Unitary ESPRIT requires the original array to be centro-symmetric, this is not required for R-D NC Unitary ESPRIT. The reason is that even if $\boldsymbol{A}$ is not centro-symmetric, the augmented array steering matrix $\boldsymbol{A}^{(\mathrm{nc})}$ is always centro-symmetric[5].

The second surprising result is that forward-backward averaging has no effect on the performance. That means if we apply FBA to $\boldsymbol{X}^{(\mathrm{nc})}$ the subspace estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ remains unaltered since

$$\boldsymbol{X}^{(\mathrm{nc})^{(\mathrm{fba})}} \cdot \left(\boldsymbol{X}^{(\mathrm{nc})^{(\mathrm{fba})}}\right)^{\mathrm{H}} = 2 \cdot \boldsymbol{X}^{(\mathrm{nc})} \cdot \left(\boldsymbol{X}^{(\mathrm{nc})}\right)^{\mathrm{H}} \tag{11.31}$$

where $\boldsymbol{X}^{(\mathrm{nc})^{(\mathrm{fba})}} = \left[\boldsymbol{X}^{(\mathrm{nc})} \quad \boldsymbol{\Pi}_{2M} \cdot \boldsymbol{X}^{(\mathrm{nc})^*} \cdot \boldsymbol{\Pi}_N\right]$. Note that (11.31) has two important consequences. Firstly, it shows that the performance of R-D NC Standard ESPRIT and R-D NC Unitary ESPRIT is asymptotically identical (cf. Section 12.4.4 where it is shown that the real-valued transformation has no effect on the asymptotical performance)[6]. Secondly, it shows that unlike Unitary ESPRIT, NC Unitary ESPRIT cannot handle two coherent sources: FBA has no decorrelation effect as shown in (11.31) and the row-wise augmentation applied for NC ESPRIT has no decorrelation effect either (as evident from (11.25)). We demonstrate this effect numerically in the simulation results in Section 11.8.3 (cf. Figure 11.10).

The third surprising result is that applying forward-backward averaging and the real-valued transformation, the resulting transformed measurement matrix takes the following simple form

$$\mathcal{T}(\boldsymbol{X}^{(\mathrm{nc})}) = 2 \cdot \begin{bmatrix} \mathrm{Re}\left\{\boldsymbol{X}\right\} & \boldsymbol{0}_{M \times N} \\ \mathrm{Im}\left\{\boldsymbol{X}\right\} & \boldsymbol{0}_{M \times N} \end{bmatrix}, \tag{11.32}$$

if the sparse left-$\boldsymbol{\Pi}$-real matrices $\boldsymbol{Q}_p^{(\mathrm{s})}$ proposed in [HN95] (cf. Appendix A.2) are used for the real-valued transformation. Since the zero block matrices and the factor 2 in front can be skipped, we conclude that the signal subspace can be estimated directly from the matrix where the real part of $\boldsymbol{X}$ and the imaginary part of $\boldsymbol{X}$ are stacked on top of each other. Based on this observation, an R-D NC Unitary ESPRIT algorithm can be derived, which is summarized in Algorithm 6.

[5] In the special case where the array is centro-symmetric, we have $\boldsymbol{J}_2 = \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{\Pi}_M$ and hence the augmented selection matrices simplify into $\boldsymbol{J}_n^{(\mathrm{nc})} = \boldsymbol{I}_2 \otimes \boldsymbol{J}_n$, $n = 1, 2$.

[6] Combining the first two observations, it becomes clear that there is actually no need for a "Standard" version of R-D NC Unitary ESPRIT. It is included in this thesis for the sake of completeness only.

Algorithm 6 [HR04] Summary of R-D NC Unitary ESPRIT using Least Squares.

1. Estimate the augmented real-valued signal subspace $\hat{\boldsymbol{E}}_{\mathrm{s}}^{(\mathrm{nc})} \in \mathbb{C}^{2M \times d}$ via the truncated SVD of the stacked observation $[\mathrm{Re}\{\boldsymbol{X}\}^{\mathrm{T}}, \mathrm{Im}\{\boldsymbol{X}\}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{2M \times N}$.

2. Solve the overdetermined shift invariance equations

$$\tilde{\boldsymbol{K}}_1^{(\mathrm{nc})(r)} \cdot \hat{\boldsymbol{E}}_{\mathrm{s}}^{(\mathrm{nc})} \cdot \boldsymbol{\Upsilon}^{(r)} \approx \tilde{\boldsymbol{K}}_2^{(\mathrm{nc})(r)} \cdot \hat{\boldsymbol{E}}_{\mathrm{s}}^{(\mathrm{nc})}$$

for the matrices $\boldsymbol{\Upsilon}^{(r)}$ for $r = 1, 2, \ldots, R$ via the method of Least Squares (LS), where

$$\tilde{\boldsymbol{K}}_1^{(\mathrm{nc})(r)} = 2 \cdot \mathrm{Re}\left\{\boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})} \cdot M/M_r} \cdot \tilde{\boldsymbol{J}}_2^{(\mathrm{nc})(r)} \cdot \boldsymbol{Q}_M\right\} \tag{11.33}$$

$$\tilde{\boldsymbol{K}}_2^{(\mathrm{nc})(r)} = 2 \cdot \mathrm{Im}\left\{\boldsymbol{Q}^{\mathrm{H}}_{M_r^{(\mathrm{sel})} \cdot M/M_r} \cdot \tilde{\boldsymbol{J}}_2^{(\mathrm{nc})(r)} \cdot \boldsymbol{Q}_M\right\} \tag{11.34}$$

and $\tilde{\boldsymbol{J}}_n^{(\mathrm{nc})(r)}$ are defined in (11.28).

3. Compute the eigenvalues $\hat{\omega}_i^{(r)}$ for $i = 1, 2, \ldots, d$ of $\hat{\boldsymbol{\Upsilon}}^{(r)}$ jointly for all $r = 1, 2, \ldots, R$, e.g., via the joint diagonalization scheme proposed in [FG06] or via the Simultaneous Schur Decomposition proposed in [HN98]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = 2 \cdot \arctan(\hat{\omega}_i^{(r)})$.

11.6. R-D NC Standard Tensor-ESPRIT and R-D NC Unitary Tensor-ESPRIT

In Section 11.4 we have shown how we can exploit the multidimensional structure of the R-D harmonic retrieval problem by virtue of tensor algebra, giving rise to the R-D Tensor-ESPRIT-type algorithms. On the other hand, in Section 11.5 we have shown how strict-sense non-circularity of the amplitudes (source symbols) can be exploited by virtue of widely linear signal processing, giving rise to NC ESPRIT-type algorithms. This sparks the question whether both approaches can be combined for the case of R-D harmonic retrieval with strict-sense non-circular source signals.

However, combining the two approaches is not a trivial task. In fact, the augmentation that was applied for R-D NC ESPRIT-type algorithms destroys the R-D separable sampling grid structure required for R-D Tensor-ESPRIT-type algorithms. This is exemplified in Figure 11.2, where we show the virtual 18-sensor array, which results from performing the augmentation for matrix-based NC ESPRIT-type algorithms to a 3×3 URA. The additional virtual URA is flipped in both dimensions but neither augmented vertically nor horizontally. Hence, the resulting array is not a separable 2-D sampling grid as defined in Section 9.2.3, since we cannot express it as the outer product of 1-D sampling grids.

Consequently, in order to exploit both, the R-D structure and the strict-sense non-circularity at the same time, a tensor-compliant way of exploiting non-circularity is required. As shown in [RH09b], this is accomplished by performing the augmentation along the individual modes separately (in the 2-D example along the rows and along the columns) and exploiting all these augmentations jointly.

To this end, let the r-mode augmented measurement tensor be given by

$$\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = [\boldsymbol{\mathcal{X}} \sqcup_r \boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R}] \in \mathbb{C}^{M_1 \times \ldots \times M_{r-1} \times 2M_r \times M_{r+1} \times \ldots \times M_R \times N}. \tag{11.35}$$

This tensor admits a factorization similar to (11.25), i.e.,

$$\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_{R+1} \boldsymbol{S}^{\mathrm{T}} + \boldsymbol{\mathcal{N}}^{(\mathrm{nc},r)}, \tag{11.36}$$

where the r-mode augmented array steering tensor $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ is given by

$$\begin{aligned}\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} &= [\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{A}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} (\boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*)] \\ &\in \mathbb{C}^{M_1 \times \ldots \times M_{r-1} \times 2M_r \times M_{r+1} \times \ldots \times M_R \times d}.\end{aligned} \tag{11.37}$$

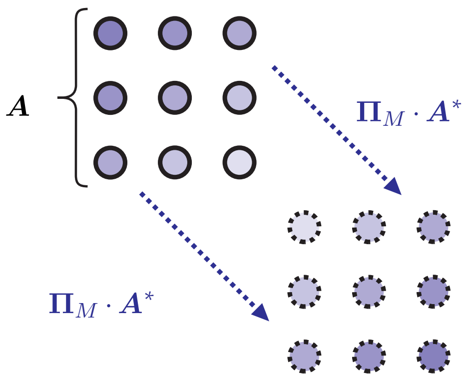


Figure 11.2.: Virtually doubled 2-D array after matrix-based augmentation of the measurements. The virtually doubled 3×3 URA is augmented by a second URA flipped in both dimensions. The resulting array is not a separable 2-D sampling grid.

The R-D NC Tensor-ESPRIT-type algorithms are based on the shift invariance of $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$, which is established in the following theorem:

Theorem 11.6.1. *The r-mode augmented array steering tensor $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ defined in* (11.37) *obeys the following shift invariance equation*

$$\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)} \tag{11.38}$$

for $r = 1, 2, \ldots R$, where $\boldsymbol{J}_1^{(\mathrm{nc})(r)}$ and $\boldsymbol{J}_2^{(\mathrm{nc})(r)}$ are defined in (11.29) *and* (11.30)*, respectively.*

Proof: cf. Appendix D.6.

Note that the shift invariance relation in (11.38) was used by us in [RH09b] to derive the R-D NC Unitary Tensor-ESPRIT algorithms. However, due to space limitations, a proof for it was not included in [RH09b]. In other words, Theorem 11.6.1 shows that the r-mode augmented array steering tensor is shift invariant with a "doubled" number of elements in the r-th mode[7]. Therefore, the idea to exploit non-circularity and the R-D tensor structure jointly is to use all r-mode augmentations together, i.e., to extract estimates for $\boldsymbol{\Phi}^{(r)}$ only from $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$ for $r = 1, 2, \ldots, R$.

In order accomplish this goal, the unknown array steering tensors need to be replaced by estimates of appropriate signal subspace tensors. This is accomplished by virtue of the following theorem:

[7]Note that $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ is shift invariant in the other modes $q = 1, 2, \ldots, R$, $q \neq r$ only if the array is centro-symmetric in the q-th mode, i.e., $\boldsymbol{\Pi}_{M_q} \cdot \boldsymbol{A}^{(q)^*}$ and $\boldsymbol{A}^{(q)}$ span the same column space. However, this additional shift invariance is not needed for R-D NC Tensor-ESPRIT type algorithms.

Theorem 11.6.2. *Let the truncated HOSVD of the noise-free r-mode augmented measurement tensor* $\boldsymbol{\mathcal{X}}_0^{(\mathrm{nc},r)}$ *be given by* $\boldsymbol{\mathcal{X}}_0^{(\mathrm{nc},r)} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}](r)} \times_1 \boldsymbol{U}_1^{[\mathrm{s}](r)} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}](r)} \times_{R+1} \boldsymbol{U}_{R+1}^{[\mathrm{s}](r)}$*. Define the r-mode augmented signal subspace tensor* $\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)}$ *via*

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}](r)} \times_1 \boldsymbol{U}_1^{[\mathrm{s}](r)} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}](r)} \times_{R+1} \boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)^{-1}} \tag{11.39}$$

be the signal subspace tensor originating from the r-mode augmented measurement tensor $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$*. Then, the following set of shift invariance equations is satisfied*

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \boldsymbol{\Psi}^{(r)} = \boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)} \quad r = 1, 2, \ldots, R \tag{11.40}$$

where $\boldsymbol{\Psi}^{(r)} = \boldsymbol{T} \cdot \boldsymbol{\Phi}^{(r)} \cdot \boldsymbol{T}^{-1}$*, i.e.,* $\boldsymbol{T}$ *is not a function of* r*.*

Proof: cf. Appendix D.7.

Note that the shift invariance relation in (11.40) was used by us in [RH09b] to derive the R-D NC Unitary Tensor-ESPRIT algorithms. However, due to space limitations, a proof for it was not included in [RH09b]. Moreover, we were assuming centro-symmetric arrays in [RH09b].

It is important to note that, as highlighted in Appendix D.7, if the array is not centro-symmetric, the n-ranks of $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ can exceed d, which must be taken into account when computing the truncated HOSVD for $\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)}$. Since they are equal to $2d$ in the worst case, it is safe to truncate the HOSVD to $2d$ in the first R modes (of course, we still truncate to d in mode $R+1$).

The most important part of this theorem is that $\boldsymbol{T}$ is not a function of r, i.e., all $\boldsymbol{\Psi}^{(r)}$ still have a common set eigenvectors. This is important since the automatic pairing in R-D ESPRIT-type algorithms is based on this fact.

The R-D NC Standard Tensor-ESPRIT follows naturally from (11.40). It is summarized in Algorithm 7.

The extension of R-D NC Standard Tensor-ESPRIT to R-D NC Unitary Tensor-ESPRIT is again quite straightforward. In fact, many of the results from the matrix case (cf. Section 11.5.2) carry over to the tensor case. Firstly, the augmented array steering tensor $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ is centro-symmetric even if the original array steering tensor $\boldsymbol{\mathcal{A}}$ is not centro-symmetric[8]. Secondly, Forward-Backward Averaging has no effect on the augmented tensor $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$, i.e.,

$$\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)^{(\mathrm{fba})}}\right]_{(R+1)}^{\mathrm{T}} \cdot \left(\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)^{(\mathrm{fba})}}\right]_{(R+1)}^{\mathrm{T}}\right)^{\mathrm{H}} = 2 \cdot \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}^{\mathrm{T}} \left(\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}^{\mathrm{T}}\right)^{\mathrm{H}}, \tag{11.41}$$

[8]The condition on centro-symmetry $\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* = \boldsymbol{A} \cdot \boldsymbol{\Delta}$ for the matrix case is expressed in tensor notation as $\boldsymbol{\mathcal{A}}^* \bigtimes_{r=1}^{R} {}_r \boldsymbol{\Pi}_{M_r} = \boldsymbol{\mathcal{A}} \times_{R+1} \boldsymbol{\Delta}$, where $\boldsymbol{\Delta}$ is a unitary diagonal matrix.

Algorithm 7 Summary of R-D NC Standard Tensor-ESPRIT using Least Squares.

1. Estimate the augmented signal subspace tensors $\hat{\mathcal{U}}^{[\mathrm{s}](r)} \in \mathbb{C}^{M_1 \times \ldots \times 2M_r \times \ldots \times M_R \times d}$ via the truncated HOSVD of the r-mode augmented observation tensors $\mathcal{X}^{(\mathrm{nc},r)} \in \mathbb{C}^{M_1 \times \ldots \times M_{r-1} \times 2M_r \times M_{r+1} \times \ldots \times M_R \times N}$ following (11.39) for $r = 1, 2, \ldots, R$.

2. Solve the overdetermined shift invariance equations

$$\hat{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \hat{\boldsymbol{\Psi}}^{(r)} \approx \hat{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)}$$

   for the matrices $\hat{\boldsymbol{\Psi}}^{(r)}$ for $r = 1, 2, \ldots, R$ via the method of Least Squares (LS).

3. Compute the eigenvalues $\hat{\lambda}_i^{(r)}$ for $i = 1, 2, \ldots, d$ of $\hat{\boldsymbol{\Psi}}^{(r)}$ jointly for all $r = 1, 2, \ldots, R$, e.g., via the joint diagonalization scheme proposed in [FG06]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = \arg\left\{\hat{\lambda}_i^{(r)}\right\}$.

where $\mathcal{X}^{(\mathrm{nc},r)^{(\mathrm{fba})}} = \left[\mathcal{X}^{(\mathrm{nc},r)} \sqcup_{R+1} \mathcal{X}^{(\mathrm{nc},r)^*} \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N\right]$. As in the matrix case, this shows that the performance of R-D NC Standard Tensor-ESPRIT is identical to R-D NC Unitary Tensor-ESPRIT and hence the latter is clearly preferable due to the lower computational complexity. Thirdly, in the matrix case, we had the result that the transformed real-valued measurement matrix has a very simple form (cf. (11.32)). Applying the tensor-based forward-backward averaging and the corresponding real-valued transformation which was introduced in (10.15), we arrive at a simple direct form of the transformed measurement tensor as well. It is summarized in the following theorem.

Theorem 11.6.3. **[RH09b]** *Applying forward-backward averaging and the real-valued transformation to the r-mode augmented measurement tensor $\mathcal{X}^{(\mathrm{nc},r)}$, we obtain*

$$\begin{aligned}\mathcal{T}(\mathcal{X}^{(\mathrm{nc},r)}) &= \left[\mathcal{X}^{(\mathrm{nc},r)} \sqcup_{R+1} (\mathcal{X}^{(\mathrm{nc},r)^*} \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N)\right] \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} \times_{R+1} \boldsymbol{Q}_{2N}^{\mathrm{H}} \\ &= \left[\left[2 \cdot \mathrm{Re}\left\{\bar{\mathcal{X}}^{(r)}\right\} \sqcup_r 2 \cdot \mathrm{Im}\left\{\bar{\mathcal{X}}^{(r)}\right\}\right] \sqcup_{R+1} \left[\mathcal{O}_{M_1 \times \ldots \times M_R \times N} \sqcup_r \mathcal{O}_{M_1 \times \ldots \times M_R \times N}\right]\right]\end{aligned}$$

where $\bar{\mathcal{X}}^{(r)} = \mathcal{X} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}}$ and $\boldsymbol{Q}_p$ are unitary left-$\boldsymbol{\Pi}$-real matrices.

Proof: cf. Appendix D.8.

This simplified form of the real-valued transformed measurement tensor was already shown in [RH09b]. However, due to space limitations, a proof for it was not included there.

Note that the zero entries in $\mathcal{T}(\boldsymbol{\mathcal{X}}^{(\text{nc},r)})$ and the factor 2 can be skipped as they have no influence on the signal subspace estimate. Therefore, we can replace $\mathcal{T}(\boldsymbol{\mathcal{X}}^{(\text{nc},r)})$ by the following simplified version

$$\bar{\mathcal{T}}(\boldsymbol{\mathcal{X}}^{(\text{nc},r)}) = \left[\text{Re}\left\{\bar{\boldsymbol{\mathcal{X}}}^{(r)}\right\} \sqcup_r \text{Im}\left\{\bar{\boldsymbol{\mathcal{X}}}^{(r)}\right\}\right] \in \mathbb{R}^{M_1\times\ldots\times M_{r-1}\times 2M_r\times M_{r+1}\times\ldots\times M_R\times N}. \tag{11.42}$$

Based on this result, the R-D NC Unitary Tensor-ESPRIT algorithm follows straightforwardly. It is summarized in Algorithm 8.

Algorithm 8 [RH09b] Summary of R-D NC Unitary Tensor-ESPRIT using Least Squares.

1. Estimate the real-valued augmented signal subspace tensors $\hat{\boldsymbol{\mathcal{E}}}^{[\text{s}](r)} \in \mathbb{R}^{M_1\times\ldots\times 2M_r\times\ldots\times M_R\times d}$ via the truncated HOSVD of the transformed r-mode augmented observation tensors $\bar{\mathcal{T}}(\boldsymbol{\mathcal{X}}^{(\text{nc},r)}) \in \mathbb{R}^{M_1\times\ldots\times M_{r-1}\times 2M_r\times M_{r+1}\times\ldots\times M_R\times N}$ shown in (11.42) for $r = 1,2,\ldots,R$.

2. Solve the overdetermined shift invariance equations
$$\hat{\boldsymbol{\mathcal{E}}}^{[\text{s}](r)} \times_r \boldsymbol{K}_1^{(\text{nc})(r)} \times_{R+1} \hat{\boldsymbol{\Upsilon}}^{(r)} \approx \hat{\boldsymbol{\mathcal{E}}}^{[\text{s}](r)} \times_r \boldsymbol{K}_2^{(\text{nc})(r)}$$
for the matrices $\hat{\boldsymbol{\Upsilon}}^{(r)}$ for $r = 1,2,\ldots,R$ via the method of Least Squares (LS), where
$$\boldsymbol{K}_1^{(\text{nc})(r)} = 2\cdot\text{Re}\left\{\boldsymbol{Q}^{\text{H}}_{M_r^{(\text{sel})}} \cdot \boldsymbol{J}_2^{(\text{nc})(r)} \cdot \boldsymbol{Q}_{M_r}\right\} \tag{11.43}$$
$$\boldsymbol{K}_2^{(\text{nc})(r)} = 2\cdot\text{Im}\left\{\boldsymbol{Q}^{\text{H}}_{M_r^{(\text{sel})}} \cdot \boldsymbol{J}_2^{(\text{nc})(r)} \cdot \boldsymbol{Q}_{M_r}\right\} \tag{11.44}$$
and $\boldsymbol{J}_n^{(\text{nc})(r)}$ are defined in (11.29) and (11.30).

3. Compute the eigenvalues $\hat{\omega}_i^{(r)}$ for $i = 1,2,\ldots,d$ of $\hat{\boldsymbol{\Upsilon}}^{(r)}$ jointly for all $r = 1,2,\ldots,R$, e.g., via the joint diagonalization scheme proposed in [FG06] or via the Simultaneous Schur Decomposition proposed in [HN98]. Recover the correctly paired frequencies $\hat{\mu}_i^{(r)}$ via $\hat{\mu}_i^{(r)} = 2\cdot\arctan(\hat{\omega}_i^{(r)})$.

11.7. Structured Least Squares

So far, the various ESPRIT-type algorithms we have discussed are all based on LS, i.e., the overdetermined shift invariance equations are solved using LS only. While the LS solution is closed-form and simple to implement, it is in general suboptimal. The reason for this is that an LS solution to an overdetermined set of equations, say, $\boldsymbol{A}\cdot\boldsymbol{x} \approx \boldsymbol{b}$ can always be interpreted as finding a projection of the vector $\boldsymbol{b}$ onto the subspace spanned by the columns of $\boldsymbol{A}$. In that respect, one inherently assumes that $\boldsymbol{A}$ is perfectly known and the only error lies on the right-hand side of the equation, i.e., we find an error term $\Delta\boldsymbol{b}$ such that $\boldsymbol{A}\cdot\boldsymbol{x} = \boldsymbol{b}+\Delta\boldsymbol{b}$ and $\|\Delta\boldsymbol{b}\|_2$ is minimized. However, in the case of a shift invariance equation we have $\boldsymbol{J}_1\hat{\boldsymbol{U}}_{\rm s}\boldsymbol{\Psi} \approx \boldsymbol{J}_2\hat{\boldsymbol{U}}_{\rm s}$, which we solve for $\boldsymbol{\Psi}$. Consequently, the error clearly lies on both sides of the equation as neither "$\boldsymbol{A}$" ($\boldsymbol{J}_1\hat{\boldsymbol{U}}_{\rm s}$) nor "$\boldsymbol{b}$" ($\boldsymbol{J}_2\hat{\boldsymbol{U}}_{\rm s}$) are perfectly known.

This observation has inspired the use of the Total Least Squares (TLS) procedure for solving the invariance equation [RK87]. TLS allows for errors in all variables, hence one error term for $\boldsymbol{J}_1\hat{\boldsymbol{U}}_{\rm s}$ and another error term for $\boldsymbol{J}_2\hat{\boldsymbol{U}}_{\rm s}$ is explicitly computed with the goal to align their subspaces until an exact solution for $\boldsymbol{\Psi}$ exists.

The drawback of TLS is that the error terms for $\boldsymbol{J}_1\hat{\boldsymbol{U}}_{\rm s}$ and $\boldsymbol{J}_2\hat{\boldsymbol{U}}_{\rm s}$ are found independently of each other. However, as long as the two subarrays used for ESPRIT overlap, they have common elements. This is additional information coming from the particular structure of the array which is ignored by TLS. In order to take this structure into account, SLS was proposed in [Haa97b]. In SLS we model an explicit error term for $\hat{\boldsymbol{U}}_{\rm s}$, accounting for the fact that the true source of error in the shift invariance equation is the subspace estimation error. Since the resulting cost function represents a quadratic least squares problem, an exact closed-form solution does not exist anymore. However, it is shown in [Haa97b] that the cost function can be solved iteratively by local linearization and that one iteration is typically sufficient.

To this end, the SLS cost function for a 1-D shift invariance equation[9] $\boldsymbol{J}_1\cdot\hat{\boldsymbol{U}}_{\rm s}\cdot\boldsymbol{\Psi} \approx \boldsymbol{J}_2\cdot\hat{\boldsymbol{U}}_{\rm s}$ can be expressed as

$$\begin{aligned}\hat{\boldsymbol{\Psi}}_{\rm SLS} &= \hat{\boldsymbol{\Psi}}_{\rm LS} + \Delta\boldsymbol{\Psi}_{\rm SLS} \quad \text{where}\\ \Delta\boldsymbol{\Psi}_{\rm SLS} &= \underset{\Delta\boldsymbol{\Psi},\Delta\boldsymbol{U}_{\rm s}}{\arg\min} \left\|\boldsymbol{J}_1\cdot\left(\hat{\boldsymbol{U}}_{\rm s}+\Delta\boldsymbol{U}_{\rm s}\right)\cdot\left(\hat{\boldsymbol{\Psi}}_{\rm LS}+\Delta\boldsymbol{\Psi}\right) - \boldsymbol{J}_2\cdot\left(\hat{\boldsymbol{U}}_{\rm s}+\Delta\boldsymbol{U}_{\rm s}\right)\right\|_{\rm F}^2 + \kappa^2\left\|\Delta\boldsymbol{U}_{\rm s}\right\|_{\rm F}^2. \quad (11.45)\end{aligned}$$

Here, $\hat{\boldsymbol{\Psi}}_{\rm LS}$ refers to the LS solution given by $\hat{\boldsymbol{\Psi}}_{\rm LS} = \left(\boldsymbol{J}_1\cdot\hat{\boldsymbol{U}}_{\rm s}\right)^+\cdot\boldsymbol{J}_2\cdot\hat{\boldsymbol{U}}_{\rm s}$. Moreover, κ is a regularization constant controlling the influence of the regularization term that penalizes too large updates in $\Delta\boldsymbol{U}_{\rm s}$. It is given by $\kappa^2 = \frac{M^{(\rm sel)}}{M\cdot\alpha}$, where $\alpha \in (0,\infty)$ controls the amount of regularization: large values of α refer to using less regularization. Since (11.45) is a quadratic least squares problem, it is solved iteratively by local linearization. In the k-th iteration, the

[9]The same algorithm applies to R-D shift invariance equations (where $\boldsymbol{J}_n$ is replaced by $\boldsymbol{J}_n^{(r)}$ for $n = 1,2$ and $r = 1,2,\ldots,R$) and to the transformed real-valued invariance equations (where $\boldsymbol{J}_n^{(r)}$, $\boldsymbol{U}_{\rm s}$, and $\boldsymbol{\Psi}^{(r)}$ are replaced by $\boldsymbol{K}_n^{(r)}$, $\boldsymbol{E}_{\rm s}$ and $\boldsymbol{\Upsilon}^{(r)}$, respectively).

updates to $\Delta \boldsymbol{U}_{\mathrm{s}}$ and $\Delta \boldsymbol{\Psi}$ are calculated via [Haa97b]

$$\begin{aligned}
\Delta \boldsymbol{U}_{\mathrm{s},k+1} &= \Delta \boldsymbol{U}_{\mathrm{s},k} + \Delta\Delta \boldsymbol{U}_{\mathrm{s},k} \quad \text{and} \quad \Delta \boldsymbol{\Psi}_{k+1} = \Delta \boldsymbol{\Psi}_k + \Delta\Delta \boldsymbol{\Psi}_k \quad \text{where} \\
\begin{bmatrix} \operatorname{vec}\{\Delta\Delta\boldsymbol{\Psi}_k\} \\ \operatorname{vec}\{\Delta\Delta\boldsymbol{U}_{\mathrm{s},k}\} \end{bmatrix} &= -\boldsymbol{F}^+ \cdot \begin{bmatrix} \operatorname{vec}\{\boldsymbol{R}_k\} \\ \kappa \cdot \operatorname{vec}\{\Delta \boldsymbol{U}_{\mathrm{s},k}\} \end{bmatrix} \quad \text{with} \\
\boldsymbol{R}_k &= \boldsymbol{J}_1 \cdot \left(\hat{\boldsymbol{U}}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s},k}\right) \cdot \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}_k\right) - \boldsymbol{J}_2 \cdot \left(\hat{\boldsymbol{U}}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s},k}\right) \quad \text{and} \\
\boldsymbol{F} &= \begin{bmatrix} \boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \left(\hat{\boldsymbol{U}}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s},k}\right)\right) & \left[\left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}_k\right) \otimes \boldsymbol{J}_1\right] - \left[\boldsymbol{I}_d \otimes \boldsymbol{J}_2\right] \\ \boldsymbol{0} & \kappa \cdot \boldsymbol{I}_{M \cdot d} \end{bmatrix},
\end{aligned} \tag{11.46}$$

where the initial values are given by $\Delta \boldsymbol{U}_{\mathrm{s},0} = \boldsymbol{0}_{M \times d}$ and $\Delta \boldsymbol{\Psi}_0 = \boldsymbol{0}_{d \times d}$. Even though SLS is derived as an iterative procedure, [Haa97b] argues that only one iteration is required to achieve a considerable improvement in estimation accuracy and therefore only a single iteration is needed.

It is important to note that TLS and SLS can be used to replace LS for the solution of the shift invariance equations in all the ESPRIT-type algorithms that were introduced up to here. Since they all follow the same three steps (signal subspace estimation, solution of the invariance equations, extraction of the spatial frequencies), we simply exchange the second step, using SLS instead of LS to solve the invariance equations.

SLS-based ESPRIT outperforms LS-based ESPRIT (both in the matrix as well as the tensor case) since the structure of the shift invariance equations is explicitly exploited to find an approximate solution for it. On the other hand, we have also seen that Tensor-ESPRIT-type algorithms outperform matrix-based ESPRIT since the multidimensional structure is exploited already in the signal subspace estimation step. This sparks the natural question whether SLS can be further improved if the multidimensional structure is also taken into account in the solution of the shift invariance equations.

This question was answered in [RH07b], where the TS-SLS scheme was introduced. TS-SLS is based on the underlying idea in SLS to model an explicit perturbation for the signal subspace. However, the structure of the subspace tensor is exploited explicitly by modeling individual perturbation terms for the components it is constructed from. Consider $R = 2$ as an example. In this case, the signal subspace tensor $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ can be estimated via $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \times_2 \hat{\boldsymbol{U}}_2^{[\mathrm{s}]}$ (cf. (10.10))[10]. Therefore, instead of modeling one unstructured perturbation term $\Delta\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ for the entire tensor $\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}$ we can exploit its structure to define three perturbation terms: one term $\Delta\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$ for the truncated core tensor $\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]}$ and two terms $\Delta\boldsymbol{U}_r^{[\mathrm{s}]}$ for the r-mode singular vectors

[10]Note that the multiplication with $\hat{\boldsymbol{\Sigma}}_{\mathrm{s}}^{-1}$ along the third mode is skipped at this point since it has no impact on the subspace estimate and skipping it simplifies the notation we need here.

$\hat{\boldsymbol{U}}_r^{[\mathrm{s}]}$, $r = 1, 2$.

The cost function for TS-SLS can then be expressed as [RH07b]

$$\Delta\boldsymbol{\Psi}^{(r)}_{\text{TS-SLS}} = \underset{\Delta\boldsymbol{\Psi}^{(r)},\Delta\boldsymbol{U}_1^{[\mathrm{s}]},\Delta\boldsymbol{U}_2^{[\mathrm{s}]},\Delta\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}}{\arg\min} \left\|\boldsymbol{\mathcal{R}}^{(r)}\right\|_{\mathrm{H}}^2 + \kappa_1^{(r)^2}\cdot\left\|\Delta\boldsymbol{U}_1^{[\mathrm{s}]}\right\|_{\mathrm{F}}^2 + \kappa_2^{(r)^2}\cdot\left\|\Delta\boldsymbol{U}_2^{[\mathrm{s}]}\right\|_{\mathrm{F}}^2 + \kappa_3^{(r)^2}\cdot\left\|\Delta\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}\right\|_{\mathrm{H}}^2$$

$$\boldsymbol{\mathcal{R}}^{(r)} = \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)} + \Delta\boldsymbol{\Psi}^{(r)}\right) - \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \times_r \boldsymbol{J}_2^{(r)} \tag{11.47}$$

$$\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} = \left(\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} + \Delta\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}\right) \times_1 \left(\hat{\boldsymbol{U}}_1^{[\mathrm{s}]} + \Delta\boldsymbol{U}_1^{[\mathrm{s}]}\right) \times_2 \left(\hat{\boldsymbol{U}}_2^{[\mathrm{s}]} + \Delta\boldsymbol{U}_2^{[\mathrm{s}]}\right)$$

$$\kappa_1^{(r)^2} = \frac{M_2 \cdot M_r^{(\mathrm{sel})}\cdot d}{\alpha\cdot M_r\cdot p_1}, \quad \kappa_2^{(r)^2} = \frac{M_1 \cdot M_r^{(\mathrm{sel})}\cdot d}{\alpha\cdot M_r\cdot p_2}, \quad \kappa_3^{(r)^2} = \frac{M \cdot M_r^{(\mathrm{sel})}}{\alpha\cdot M_r\cdot p_1\cdot p_2}, \quad p_r = \min\left\{M_r, d\right\}.$$

A solution based on iterative linearization can be developed in a similar manner as for the SLS scheme. To simplify the notation, we drop the $(\cdot)^{[\mathrm{s}]}$ superscript. Then, the k-th iteration comprises of computing the following update term

$$\begin{aligned}
\Delta\boldsymbol{\Psi}^{(r)}_{k+1} &= \Delta\boldsymbol{\Psi}^{(r)}_k + \Delta\Delta\boldsymbol{\Psi}^{(r)}_k \\
\Delta\boldsymbol{U}_{1,k+1} &= \Delta\boldsymbol{U}_{1,k} + \Delta\Delta\boldsymbol{U}_{1,k} \\
\Delta\boldsymbol{U}_{2,k+1} &= \Delta\boldsymbol{U}_{2,k} + \Delta\Delta\boldsymbol{U}_{2,k} \\
\Delta\boldsymbol{\mathcal{S}}_{k+1} &= \Delta\boldsymbol{\mathcal{S}}_k + \Delta\Delta\boldsymbol{\mathcal{S}}_k
\end{aligned}$$

$$\begin{bmatrix} \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\Psi}^{(r)}_k\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix} = -\begin{bmatrix} \boldsymbol{F}_1^{(r)} & \boldsymbol{F}_2^{(r)} & \boldsymbol{F}_3^{(r)} & \boldsymbol{F}_4^{(r)} \\ \boldsymbol{0} & \kappa_1^{(r)}\boldsymbol{I}_{M_1\cdot p_1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \kappa_2^{(r)}\boldsymbol{I}_{M_2\cdot p_2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \kappa_3^{(r)}\boldsymbol{I}_{p_1\cdot p_2\cdot d} \end{bmatrix}^+ \cdot \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}^{(r)}_k\right\} \\ \kappa_1^{(r)}\,\mathrm{vec}\left\{\Delta\boldsymbol{U}_{1,k}\right\} \\ \kappa_2^{(r)}\,\mathrm{vec}\left\{\Delta\boldsymbol{U}_{2,k}\right\} \\ \kappa_3^{(r)}\,\mathrm{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix}$$

$$\begin{aligned}
\boldsymbol{\mathcal{R}}^{(r)}_k &= \hat{\boldsymbol{\mathcal{U}}}_k^{[\mathrm{s}]} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \left(\hat{\boldsymbol{\Psi}}^{(r)}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}^{(r)}_k\right) - \hat{\boldsymbol{\mathcal{U}}}_k^{[\mathrm{s}]} \times_r \boldsymbol{J}_2^{(r)} \\
\hat{\boldsymbol{\mathcal{U}}}_k^{[\mathrm{s}]} &= \left(\hat{\boldsymbol{\mathcal{S}}} + \Delta\hat{\boldsymbol{\mathcal{S}}}_k\right) \times_1 \left(\hat{\boldsymbol{U}}_1 + \Delta\hat{\boldsymbol{U}}_{1,k}\right) \times_2 \left(\hat{\boldsymbol{U}}_2 + \Delta\hat{\boldsymbol{U}}_{2,k}\right)
\end{aligned}$$

A sketch of the derivation of this update rule together with the (rather lengthy) expressions for $\boldsymbol{F}_n^{(r)}$, $n = 1, 2, 3, 4$, $r = 1, 2$ is shown in Appendix D.9.

Instead of solving the TS-SLS problem for both dimensions $r = 1, 2$ separately it is also possible to solve one joint 2-D TS-SLS problem. This extension is similar to the R-D extension for SLS proposed in [Haa97b]. Considering the sum of norms of the residual tensors in the cost function and proceeding with the iterative linearization as before, we arrive at the following update rule for 2-D TS-SLS

$$\begin{bmatrix} \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(2)}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{F}_1^{(1)} & \boldsymbol{0} & \boldsymbol{F}_2^{(1)} & \boldsymbol{F}_3^{(1)} & \boldsymbol{F}_4^{(1)} \\ \boldsymbol{0} & \boldsymbol{F}_1^{(2)} & \boldsymbol{F}_2^{(2)} & \boldsymbol{F}_3^{(2)} & \boldsymbol{F}_4^{(2)} \\ \boldsymbol{0} & \boldsymbol{0} & \kappa_1^{(2\mathrm{D})}\boldsymbol{I}_{M_1 \cdot p_1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \kappa_2^{(2\mathrm{D})}\boldsymbol{I}_{M_2 \cdot p_2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \kappa_3^{(2\mathrm{D})}\boldsymbol{I}_{p_1 \cdot p_2 \cdot d} \end{bmatrix}^+ \cdot \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k^{(1)}\right\} \\ \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k^{(2)}\right\} \\ \kappa_1^{(2\mathrm{D})}\,\mathrm{vec}\left\{\Delta\boldsymbol{U}_{1,k}\right\} \\ \kappa_2^{(2\mathrm{D})}\,\mathrm{vec}\left\{\Delta\boldsymbol{U}_{2,k}\right\} \\ \kappa_3^{(2\mathrm{D})}\,\mathrm{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix}$$

$$\begin{aligned} \kappa_1^{(2\mathrm{D})^2} &= \frac{\left(M_1^{(\mathrm{sel})} \cdot M_2 + M_1 \cdot M_2^{(\mathrm{sel})}\right) \cdot d}{\alpha \cdot M_1 \cdot p_1} \\ \kappa_2^{(2\mathrm{D})^2} &= \frac{\left(M_1^{(\mathrm{sel})} \cdot M_2 + M_1 \cdot M_2^{(\mathrm{sel})}\right) \cdot d}{\alpha \cdot M_2 \cdot p_2} \\ \kappa_3^{(2\mathrm{D})^2} &= \frac{M_1^{(\mathrm{sel})} \cdot M_2 + M_1 \cdot M_2^{(\mathrm{sel})}}{\alpha \cdot p_1 \cdot p_2}. \end{aligned}$$

Concerning the iterations of TS-SLS, one can either fix the number of iterations a priori (in simulations we found between 2 and 4 iterations to be sufficient) or keep iterating until a suitable stopping criterion is fulfilled. For instance, TS-SLS can be terminated if the relative change in the norms of the residual tensors $\boldsymbol{\mathcal{R}}_k^{(r)}$ between iterations drops below a threshold ε, i.e., $\sum_{r=1}^{R}\left(\left\|\boldsymbol{\mathcal{R}}_{k-1}^{(r)}\right\|_{\mathrm{H}} - \left\|\boldsymbol{\mathcal{R}}_k^{(r)}\right\|_{\mathrm{H}}\right) / \left\|\boldsymbol{\mathcal{R}}_k^{(r)}\right\|_{\mathrm{H}} < \varepsilon$.

11.8. Simulation Results

In this section we compare the various ESPRIT-type algorithms in terms of their estimation accuracy via Monte-Carlo simulations. We generate data according to the model (9.11). The noise samples $\boldsymbol{\mathcal{N}}$ are drawn according to a Zero Mean Circularly Symmetric Complex Gaussian (ZMCSCG) distribution with variance σ_{n}^2 and assumed to be mutually uncorrelated. The amplitudes $\boldsymbol{S}$ are also assumed to be ZMCSCG distributed with unit variance. In the special case where strict-sense non-circular sources are considered, $\boldsymbol{S}$ is generated according to (11.21) where the samples in $\boldsymbol{S}_0$ are drawn from a real-valued zero mean Gaussian distribution with unit variance and the phases φ_i are fixed. For the case where source correlation is considered we generate $\boldsymbol{S}$ such that $\boldsymbol{R}_{\mathrm{ss}} = \mathbb{E}\left\{\boldsymbol{S} \cdot \boldsymbol{S}^{\mathrm{H}}\right\}$ takes the form

$$\boldsymbol{R}_{\mathrm{ss}} = \begin{bmatrix} 1 & \rho_{1,2} & \cdots & \rho_{1,d} \\ \rho_{2,1} & 1 & \cdots & \rho_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{d,1} & \rho_{d,2} & \cdots & 1 \end{bmatrix} \quad \text{with } |\rho_{i,j}| = \rho \ \forall i,j = 1,2,\ldots,d \tag{11.48}$$

and $\arg\{\rho_{i,j}\}$ drawn from a uniform distribution in $[0, 2\pi]$. Consequently, the scalar parameter $\rho \in [0, 1]$ controls the level of correlation between each pair of sources. For the array we assume uniform sampling in R dimensions with $M_1 \times M_2 \ldots \times M_R$ sampling points, i.e., for $R = 1$ we have an M-element ULA and for $R = 2$ an $M_1 \times M_2$ URA.

The estimation accuracy of the algorithms is measured with respect to a root mean square estimation error (RMSE) defined as

$$\mathrm{RMSE} = \sqrt{\mathbb{E}\left\{\frac{1}{d}\frac{1}{R}\sum_{i=1}^{d}\sum_{r=1}^{R}\left(\mu_i^{(r)} - \hat{\mu}_i^{(r)}\right)^2\right\}}. \tag{11.49}$$

To estimate the RMSE, the ensemble average is replaced by an arithmetic average over Monte-Carlo trials. In order to demonstrate the enhanced HOSVD-based subspace estimate we also display the Largest Principle Angle (LPA) between the true subspace (given by the columns of the array steering matrix $\boldsymbol{A}$) and the estimated signal subspaces. The LPA between the column spaces of two matrices $\boldsymbol{U}_1, \boldsymbol{U}_2 \in \mathbb{C}^{M\times d}$ can be computed through [GvL96]

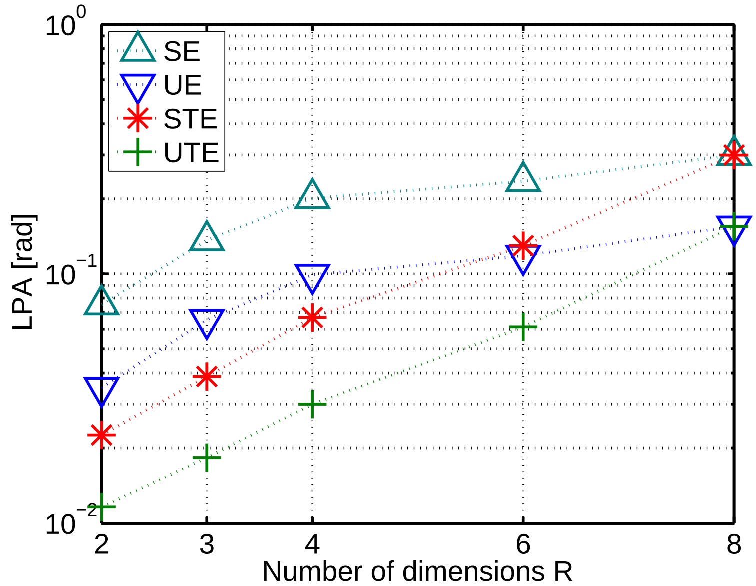
$$\mathrm{LPA} = \cos^{-1}\left(\sigma_{\min}\left\{\mathrm{orth}\left\{\boldsymbol{U}_1\right\}^{\mathrm{H}} \cdot \mathrm{orth}\left\{\boldsymbol{U}_2\right\}\right\}\right), \tag{11.50}$$

where $\mathrm{orth}\{\boldsymbol{U}_i\}$, $i = 1, 2$ is an orthonormal basis for the column space of $\boldsymbol{U}_i$ and $\sigma_{\min}\{\boldsymbol{Z}\}$ denotes the smallest singular values of the matrix $\boldsymbol{Z}$.

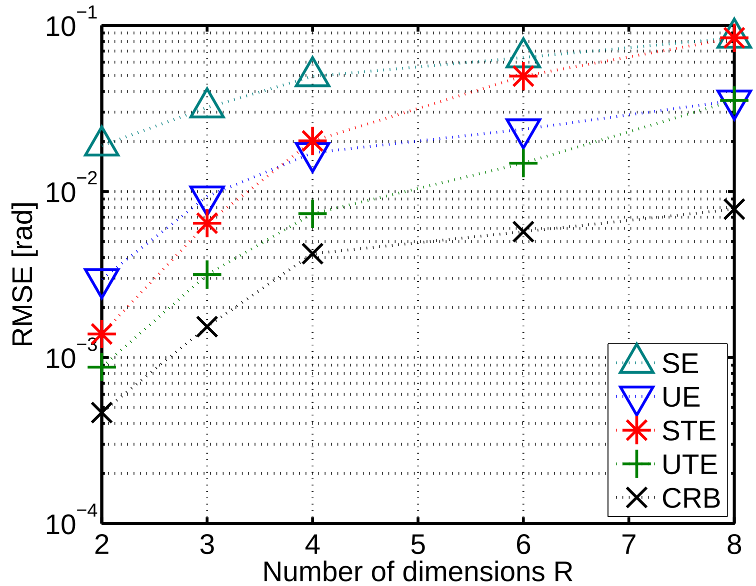
In Section 11.8.1 we compare 1-D and R-D versions of Standard ESPRIT (SE), Unitary ESPRIT (UE), Standard Tensor-ESPRIT (STE), and Unitary Tensor-ESPRIT (UTE). The Least-Squares algorithms to solve the invariance equations are compared in Section 11.8.2. 1-D and R-D ESPRIT-type algorithms for strict-sence non-circular sources are shown in Sections 11.8.3 and 11.8.4, respectively. As a reference, we also display the corresponding deterministic Cramér-Rao Bound (CRB) [SN89] and its corresponding version for strict-sense non-circular sources (CRBnc) derived in [RH07a]. Table 11.1 summarizes the abbreviations used in the subsequent figure captions.

11.8.1. R-D Standard and Unitary Tensor-ESPRIT

The simulations results shown in Figure 11.3a and Figure 11.3b provide a comparison of the performance as a function of the number of dimensions R in terms of the LPA and the RMSE, respectively. We distribute a total of $M = 256$ sensors among a varying number of dimensions R. The point $R = 2$ corresponds to a 16×16 URA, for $R = 3$, a $8 \times 8 \times 4$ dimensional harmonic retrieval problem is simulated, and $R = 4, 6$, and 8 correspond to R-dimensional harmonic retrieval problems of size $4\times4\times4\times4$, $4\times4\times2\times2\times2\times2$, and $2\times2\times2\times2\times2\times2\times2\times2$, respectively.



(a) Largest Principle Angle (LPA)



(b) Root Mean Square Error (RMSE)

Figure 11.3.: Performance of the algorithms versus the number of dimensions R: The number of sensors is fixed to 256 and distributed among R dimensions with uniform spacing in all modes ($R = 2$: 16×16, $R = 3$: $8 \times 8 \times 4$, $R = 4$: $4 \times 4 \times 4 \times 4$, $R = 6$: $4 \times 4 \times 2 \times 2 \times 2 \times 2$, and $R = 8$: $2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2$). Two sources are positioned at $\mu_1^{(r)} = 1$, $\mu_2^{(r)} = 0.95$, $r = 1, 2, \ldots, R$. The SNR is fixed to 40 dB, $N = 2$ snapshots are used.

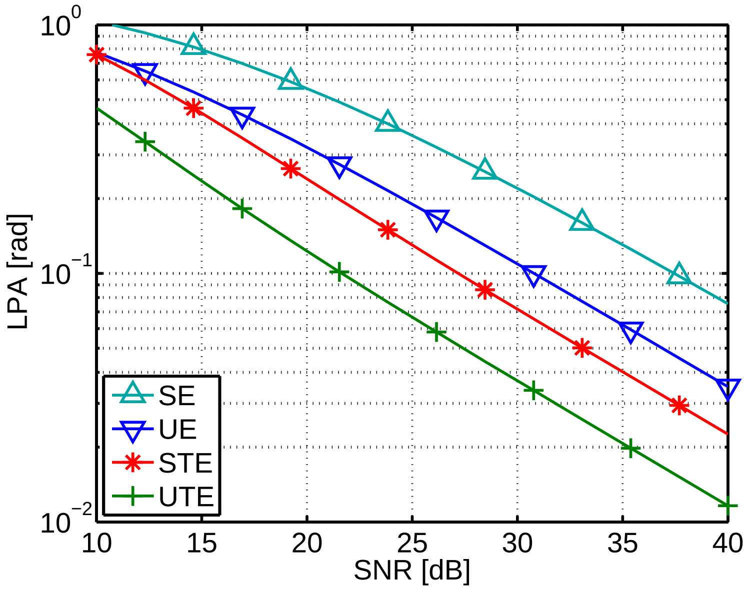
Abbreviation	Meaning
LPA	Largest principal angle
RMSE	Root mean square error
SNR	Signal to noise ratio
SE	R-D Standard ESPRIT
UE	R-D Unitary ESPRIT
STE	R-D Standard Tensor-ESPRIT
UTE	R-D Unitary Tensor-ESPRIT
NC UE	R-D NC Unitary ESPRIT
NC UTE	R-D NC Unitary Tensor-ESPRIT
CRB	Deterministic Cramér-Rao Bound
CRBnc	det. Cramér-Rao Bound for strict-sense non-circular sources [RH07a]
LS	Least Squares
SLS	Structured Least Squares
TS-SLS	Tensor-Structure Structured Least Squares
TS-RD-SLS	Tensor-Structure R-D Structured Least Squares

Table 11.1.: Abbreviations used for R-D the simulation results

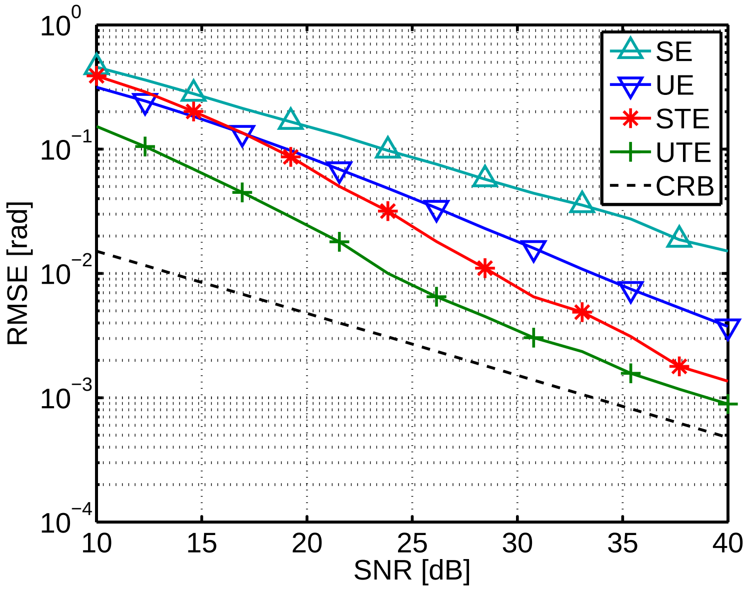
The SNR is fixed to 40 dB and $N = 2$ snapshots are used. The $d = 2$ sources are relatively close to each other since $\mu_1^{(r)} = 1$ and $\mu_2^{(r)} = 0.95$ for $r = 1, 2, \dots, R$. The results show that there is a significant improvement both in terms of the LPA as well as the RMSE. We observe the largest improvement in the two-dimensional case ($R = 2$). Note that the improvement of the tensor approach diminishes as R increases. This gain vanishes completely if $R = 8$. In this case, we have $M_r = 2$ sensors in each of the dimensions and, therefore, the condition $d \geq \max_{r=1,2,\dots,R}(M_r)$ is fulfilled. Consequently, as shown in Section 10.2, the HOSVD-based signal subspace estimate is equal to the one obtained through the matrix approach.

In Figure 11.4a and Figure 11.4b we vary the SNR for closely spaced sources at $\mu_1^{(1)} = \mu_1^{(2)} = 1$ and $\mu_2^{(1)} = \mu_2^{(2)} = 0.95$. Again, only $N = 2$ snapshots are used, and a URA of size 16×16 is employed. As before, the improvements in terms of the LPA and the RMSE are clearly visible.

The effects of varying the source separation are studied in the simulation results shown in Figure 11.5a and Figure 11.5b. The simulation setup is similar to the previous case. However, now the SNR is fixed to 40 dB and the source positions are varied in the following fashion: The first source is fixed at $\mu_1^{(1)} = \mu_1^{(2)} = 1$, while the second source is moved to the positions $\mu_2^{(1)} = \mu_2^{(2)} = 1 - \Delta$ for various values of the separation parameter $\Delta\mu$, which is also shown on the horizontal axis. We can see that increasing the separation above 0.5 does not alter the performance any further. The improvements of Standard Tensor-ESPRIT and Unitary

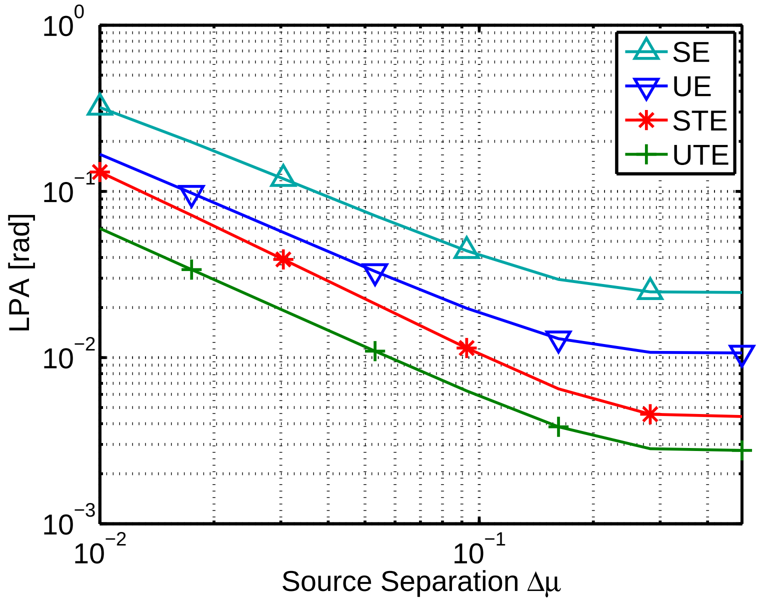


(a) Largest Principle Angle (LPA)

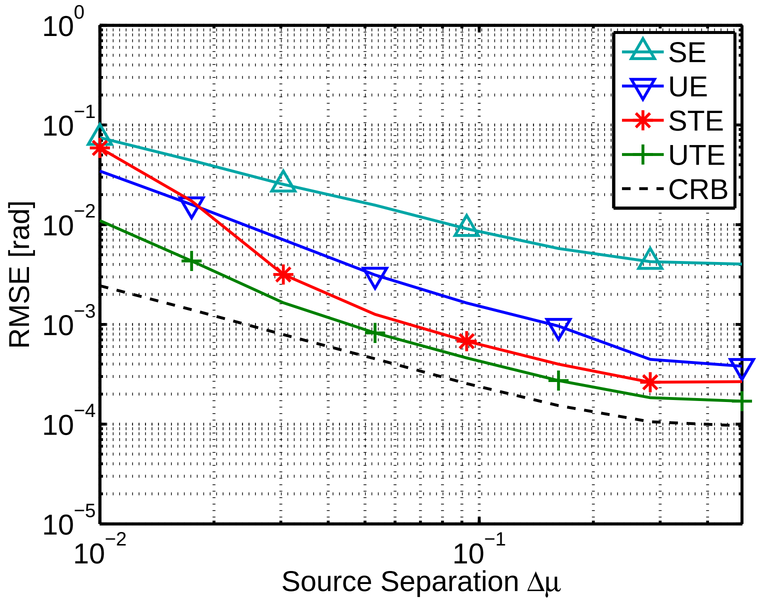


(b) Root Mean Square Error (RMSE)

Figure 11.4.: Performance of the algorithms versus SNR in dB. Two sources are positioned at $\mu_1^{(r)} = 1$, $\mu_2^{(r)} = 0.95$, $r = 1, 2$, a 16×16 URA and $N = 2$ snapshots.



(a) Largest Principle Angle (LPA)



(b) Root Mean Square Error (RMSE)

Figure 11.5.: Performance of the algorithms versus the separation of the two sources. The first source is fixed at $\mu_1^{(r)} = 1$ and the second source is moved to the positions $\mu_2^{(r)} = 1 - \Delta\mu$ for $r = 1, 2$, where $\Delta\mu$ is the separation shown on the horizontal axis. The SNR is fixed to 40 dB, a 16×16 URA, and $N = 2$ snapshots are used.

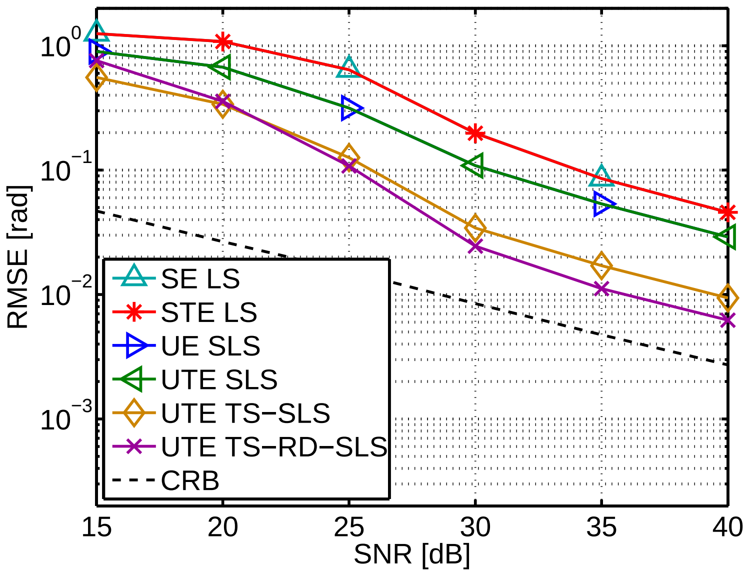


Figure 11.6.: RMSE vs. SNR for $d = 3$ correlated sources ($\rho = 0.999$) on a 3×3 URA, $N = 10$, $\mu_1^{(1)} = 1$, $\mu_1^{(2)} = -1$, $\mu_2^{(1)} = 0$, $\mu_2^{(2)} = 1$, $\mu_3^{(1)} = -1$, $\mu_3^{(2)} = 0$.

Tensor-ESPRIT in terms of the LPA and the RMSE are significant.

11.8.2. Tensor-Structure Structured Least Squares

The next set of simulation result compares the different Least Squares approaches to solve the invariance equations, namely the LS approach, the matrix-based SLS technique and its tensor extensions TS-SLS and TS-RD-SLS.

In Figure 11.6, $d = 3$ sources with a strong correlation of $\rho = 0.9999$ are captured by a 3×3 uniform rectangular array collecting $N = 10$ subsequent snapshots for various values of the SNR at the receiver. The true spatial frequencies are set to $\mu_1^{(1)} = 1$, $\mu_1^{(2)} = -1$, $\mu_2^{(1)} = 0$, $\mu_2^{(2)} = 1$, $\mu_3^{(1)} = -1$, $\mu_3^{(2)} = 0$. Since we have $d \geq \max(M_1, M_2)$, the HOSVD-based subspace estimate coincides with the matrix-based subspace estimate. Therefore, there is no improvement of the LS-based STE over the LS-based SE. Likewise, the SLS-based UE algorithm performs identical to the SLS-based UTE algorithm. However, the RMSE of UTE combined with TS-SLS is significantly lower and its R-D extension TS-RD-SLS provides an additional gain. This shows that even in scenarios where the condition $d \geq \max_{r=1,2,\ldots,R}(M_r)$ is fulfilled such that the HOSVD-based subspace estimate provides no improvement, we can benefit from the multidimensional structure via the TS-SLS algorithm.

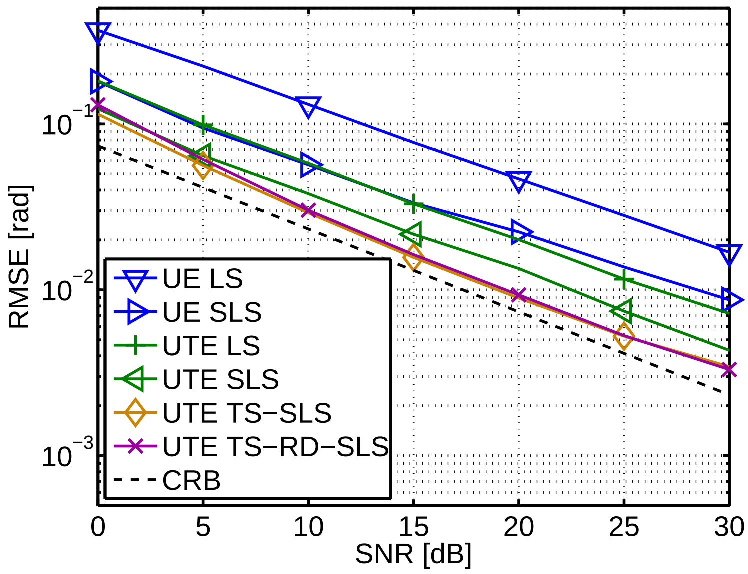


Figure 11.7.: RMSE vs. SNR for $d = 2$ correlated sources on a 3×3 URA, single snapshot ($N = 1$), $\mu_1^{(1)} = 1$, $\mu_1^{(2)} = -1$, $\mu_2^{(1)} = 0$, $\mu_2^{(2)} = 1$.

For the simulation shown in Figure 11.7, a 5×7 URA with $N = 1$ snapshot was used for $d = 2$ harmonics with true spatial frequencies $\mu_1^{(1)} = 1$, $\mu_1^{(2)} = -1$, $\mu_2^{(1)} = 0$, $\mu_2^{(2)} = 1$. Note that since $N = 1$, Standard ESPRIT cannot be applied. Here we can observe the effect of the improved signal subspace estimate in going from UE to UTE and the additional gain in using TS-SLS or TS-RD-SLS. In general, we observe that TS-SLS and TS-RD-SLS are beneficial in critical scenarios, e.g., if the number of snapshots is small and/or if the sources are highly correlated.

11.8.3. 1-D NC Unitary ESPRIT

In order to show the improvement obtained by exploiting strict-sense non-circular sources we show four simulation results comparing the 1-D Standard and Unitary ESPRIT with the 1-D NC Unitary ESPRIT technique. We do not show 1-D NC Standard ESPRIT since its performance is identical to 1-D NC Unitary ESPRIT, as discussed in Section 11.5.2. The strict-sense non-circular source amplitudes are generated according to (9.22), generating $\boldsymbol{S}_0$ from a standard normal distribution with source correlation $|\mathbb{E}\{s_{0,i}[n] \cdot s_{0,j}[n]\}| = \rho$.

Since the results presented in this section refer to the 1-D case only, we compute the spatial frequency of the i-th source as a function of the azimuth angle $\theta \in [0°, 180°]$ via $\mu_i = \pi \cdot \cos(\theta_i)$ which corresponds to a ULA with $\lambda/2$ inter-element spacing and a broadside of $\theta = 90°$. The

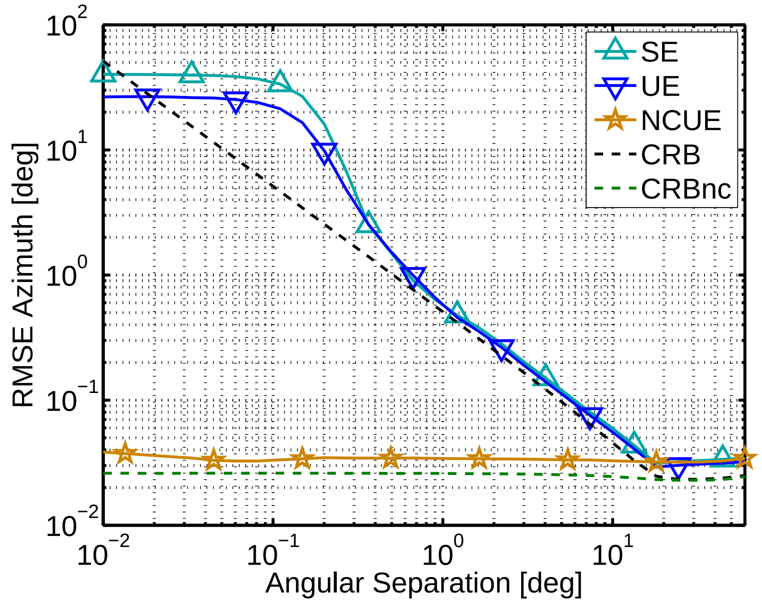


Figure 11.8.: RMSE vs. the angular separation $\Delta\theta$ for a $M = 8$ ULA, $N = 10$ snapshots, an SNR of 30 dB, $d = 2$ uncorrelated sources with a phase separation of $\Delta\varphi = \pi/2$.

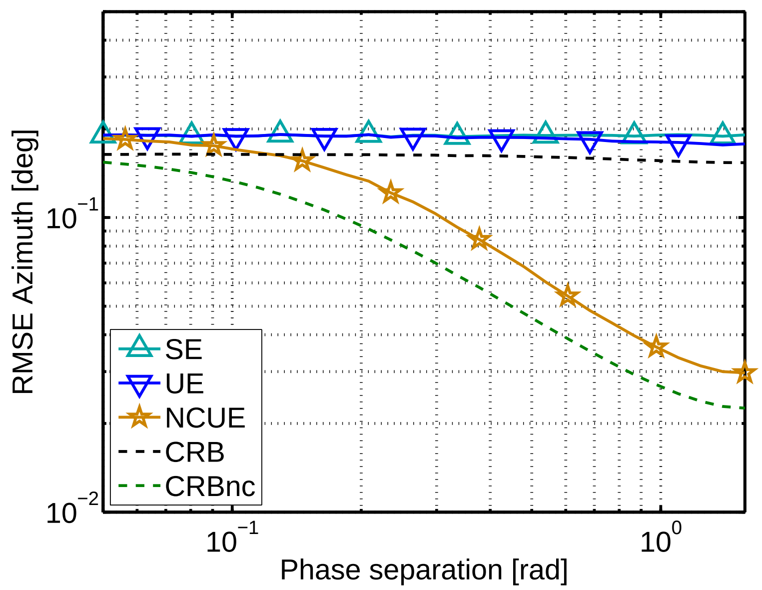


Figure 11.9.: RMSE vs. the non-circularity phase separation $\Delta\varphi$ for a $M = 8$ ULA, $N = 10$ snapshots, an SNR of 30 dB, and $d = 2$ uncorrelated sources at $\theta_1 = 88.5°$, $\theta_2 = 90°$.

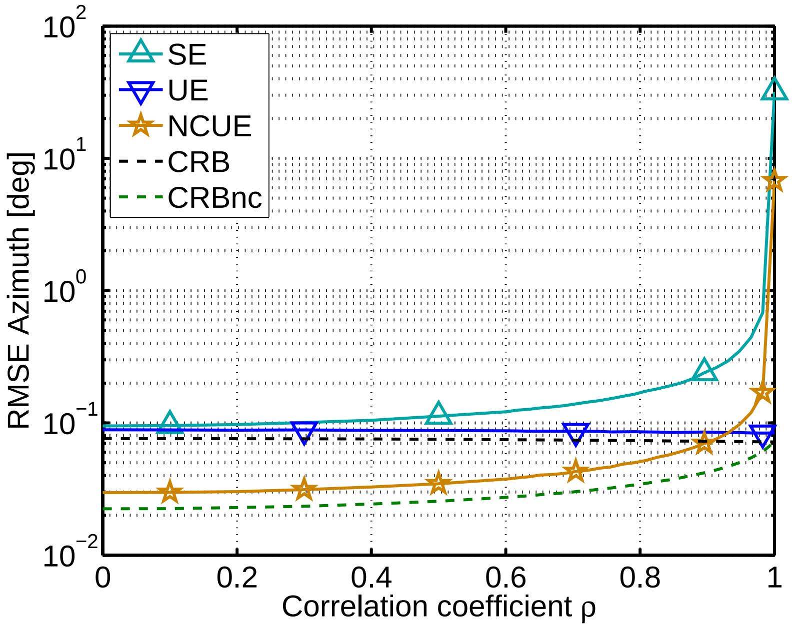


Figure 11.10.: RMSE vs. the correlation coefficient ρ for $d = 2$ with a non-circularity phase separation of $\Delta\varphi = \pi/2$ and azimuth angles $\theta_1 = 88°$, $\theta_2 = 93°$. An $M = 8$ ULA, $N = 10$ snapshots, and an SNR of 30 dB is assumed.

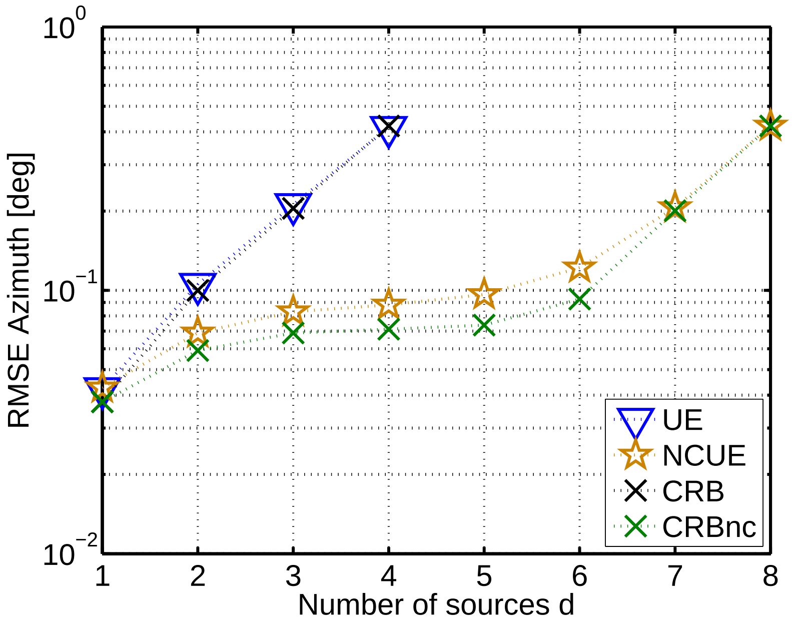


Figure 11.11.: RMSE vs. the number of sources d for a 5-ULA, $N = 10$ snapshots and an SNR of 40 dB. The positions of the sources are chosen equi-spaced in the interval $[20°, 160°]$, the non-circularity phases φ_i are chosen randomly.

RMSE of the azimuth angle is then defined as

$$\mathrm{RMSE}_{\mathrm{Azimuth}} = \sqrt{\mathbb{E}\left\{\frac{1}{d}\sum_{i=1}^{d}\left(\theta_i - \hat{\theta}_i\right)^2\right\}} \tag{11.51}$$

and measured in degrees.

In Figure 11.8 we consider an $M = 8$ element ULA, $N = 10$ snapshots, an SNR of 30 dB, and $d = 2$ uncorrelated sources with a phase separation of $\Delta\varphi = \pi/2$. We vary the spatial separation by setting $\theta_1 = 60°$ and $\theta_2 = \theta_1 + \Delta\theta$. We observe that for NCUE the RMSE is independent from the angular separation, which is due to the fact that the phase angles are orthogonal ($\Delta\varphi = \pi/2$).

To study the effect of the phase separation we vary $\Delta\varphi$ for the simulation shown in Figure 11.9. The simulation setup is similar to the previous experiment except that the azimuth angles are fixed to $\theta_1 = 88.5°$, $\theta_2 = 90°$ and $\Delta\varphi$ is varied. We observe that for SE and UE, $\Delta\varphi$ has no impact. On the other hand, NCUE converges to UE for $\Delta\varphi \to 0$ and has the largest gain compared to UE for $\Delta\varphi = \pi/2$.

The impact of source correlation is investigated in the simulation result shown in Figure 11.10. Again, the settings are similar as before, except that the azimuth angles are changed to $\theta_1 = 88°$ and $\theta_2 = 90°$, the phase separation is fixed to $\Delta\varphi = \pi/2$. We observe that Unitary ESPRIT is immune to the source correlation which is due to the inherent Forward-Backward Averaging step that decorrelates the sources. While NCUE always outperforms NCSE, it is outperformed by UE for high values of the correlation. This is due to the fact that FBA has no docorrelation effect for NCUE, as shown in Section 11.5.2.

To demonstrate the effect of increasing the number of sources d we show Figure 11.11 where d is varied from 1 to its maximum possible value. For a ULA of $M = 5$ sensors, SE and UE are limited to $d_{\max} = 4$ wavefronts. However, since the augmentation performed for NC UE virtually doubles the number of possible sources, the largest possible d for NC UE is also doubled to $d_{\max} = 8$. For the simulation we fix the SNR to 40 dB, consider $N = 10$ snapshots, and draw the phase angles φ_i randomly according to a uniform distribution in $[-\pi, \pi]$. The azimuth angles are chosen with a uniform spacing in the interval $[20°, 160°]$, i.e., for $d > 1$ we choose $\theta_i = 20° + 140° \cdot (i-1)/(d-1)$ for $i = 1, 2, \ldots, d$ and for $d = 1$ we choose $\theta_1 = 90°$.

11.8.4. R-D NC Unitary Tensor-ESPRIT

The final set of simulations demonstrates the performance of the R-D ESPRIT-type algorithms for non-circular sources. As in the 1-D case we only consider Unitary ESPRIT-type algorithms

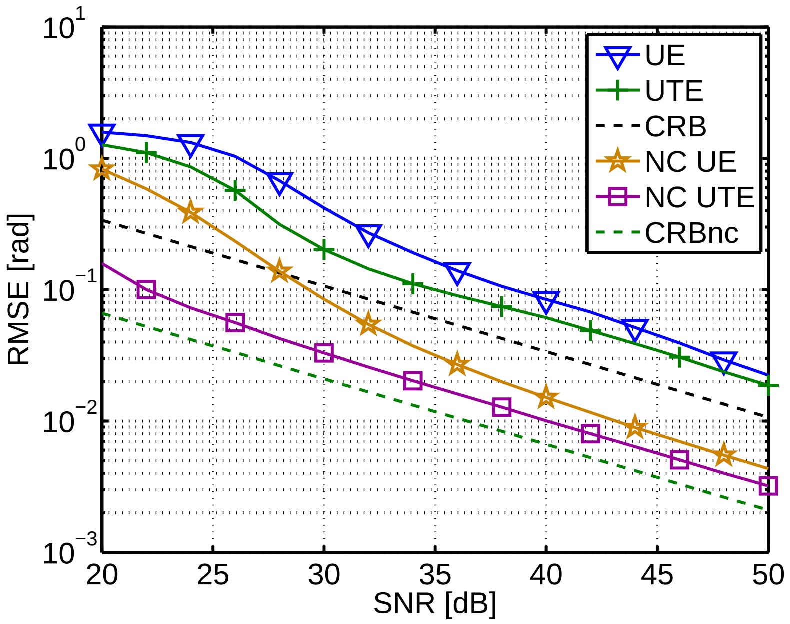


Figure 11.12.: RMSE vs. SNR for $d = 3$ correlated sources ($\rho = 0.99$) at fixed positions $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.85, \mu_3^{(1)} = \mu_3^{(2)} = 1.15$ with phase angles $\varphi_1 = 0, \varphi_2 = \pi/2, \varphi_3 = \pi/4$. A 5×7 URA and $N = 10$ snapshots.

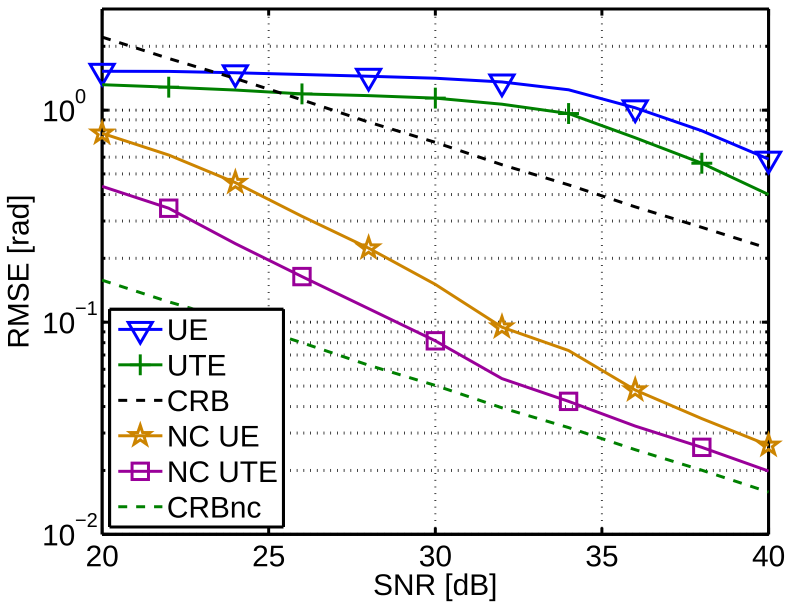


Figure 11.13.: RMSE vs. SNR for a 6×6 URA, $N = 10$ snapshots, $d = 4$ uncorrelated sources at fixed positions $\mu_1^{(1)} = \mu_1^{(2)} = 1, \mu_2^{(1)} = \mu_2^{(2)} = 0.9, \mu_3^{(1)} = \mu_3^{(2)} = 0.8, \mu_4^{(1)} = \mu_4^{(2)} = 0.7$ with phase angles $\varphi_1 = 0, \varphi_2 = \pi/6, \varphi_3 = \pi/3, \varphi_4 = \pi/2$.

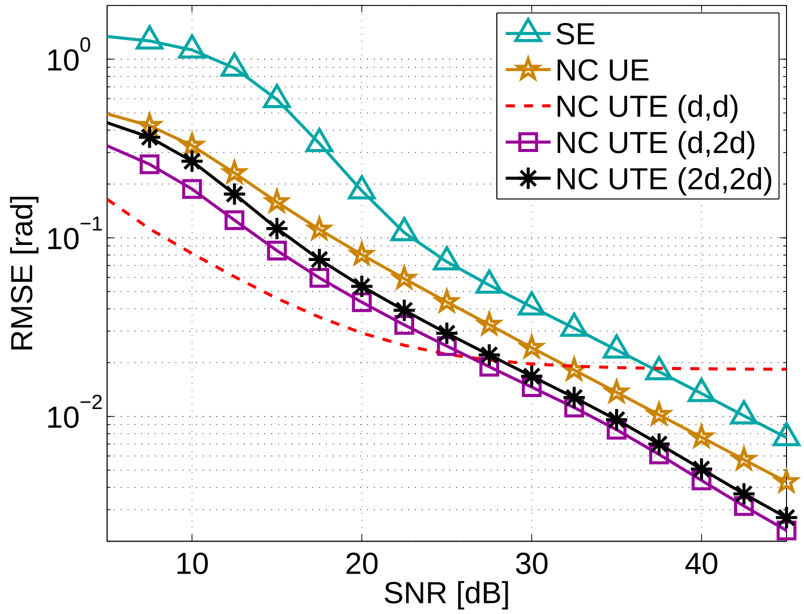


Figure 11.14.: RMSE vs. SNR for the 20-element 2-D array from Figure 9.3b, $d = 2$ correlated sources ($\rho = 0.99$) at $\mu_1^{(1)} = 0.7$, $\mu_2^{(1)} = 1$, $\mu_1^{(2)} = -0.3$, $\mu_2^{(2)} = -0.1$ with phase angles $\varphi_1 = 0$, $\varphi_2 = \pi/4$, and $N = 10$ snapshots.

and compare R-D Unitary ESPRIT and R-D NC Unitary ESPRIT with their tensor versions R-D Unitary Tensor-ESPRIT and R-D NC Unitary Tensor-ESPRIT.

For the simulation result shown in Figure 11.12 we consider a 5×7 URA, $N = 10$ snapshots and $d = 3$ correlated sources ($\rho = 0.99$). The sources' phase angles are fixed to $\varphi_1 = 0$, $\varphi_2 = \pi/2$, $\varphi_3 = \pi/4$ and the true spatial frequencies are given by $\mu_1^{(1)} = \mu_1^{(2)} = 1$, $\mu_2^{(1)} = \mu_2^{(2)} = 0.85$, $\mu_3^{(1)} = \mu_3^{(2)} = 1.15$. On the other hand, for Figure 11.13 we use a 6×6 URA and $d = 4$ uncorrelated sources with phase angles given by $\varphi_1 = 0$, $\varphi_2 = \pi/6$, $\varphi_3 = \pi/3$, and $\varphi_4 = \pi/2$. Moreover, the true spatial frequencies in this scenario are $\mu_1^{(1)} = \mu_1^{(2)} = 1$, $\mu_2^{(1)} = \mu_2^{(2)} = 0.9$, $\mu_3^{(1)} = \mu_3^{(2)} = 0.8$, and $\mu_4^{(1)} = \mu_4^{(2)} = 0.7$. Both simulation results show that NC UE outperforms UE (due to exploiting the non-circularity), UTE outperforms UE (due to exploiting the R-D structure), and NC UTE outperforms both (by combining both benefits).

Finally, Figure 11.14 shows the result of using a 20-element array that is not uniformly spaced and not centro-symmetric, which is depicted in Figure 9.3b. We choose $d = 2$ correlated sources ($\rho = 0.99$) with true spatial frequencies given by $\mu_1^{(1)} = 0.7$, $\mu_2^{(1)} = 1$, $\mu_1^{(2)} = -0.3$, $\mu_2^{(2)} = -0.1$ and phase angles set to $\varphi_1 = 0$, $\varphi_2 = \pi/4$. The number of snapshots is set to $N = 10$ Since the array is not centro-symmetric, R-D Unitary ESPRIT and R-D Unitary Tensor-ESPRIT cannot be applied. However, R-D NC Unitary ESPRIT and R-D NC Unitary Tensor-ESPRIT can be used as shown in Section 11.6. However, for non-centro-symmetric arrays the mode-wise augmentation can lead to an increase in the n-ranks of the desired signal component exceeding d in the first $R = 2$ modes. This must be taken into account when computing the truncated HOSVD for the tensor-based signal subspace estimate. We therefore depict three curves for NCUTE labeled by (p_1, p_2), which refers to truncating the HOSVD to rank p_1 and p_2 in the first and the second mode (we always set $p_3 = d$). We observe that choosing $p_1 = p_2 = d$ leads to an error floor since the tensor is truncated to an n-rank that is too low. The "safe" choice $p_1 = p_2 = 2d$ performs well. The optimal choice of n-ranks in this scenario is $p_1 = d$ and $p_2 = 2d$ (since this choice coincides with the n-ranks of the noise-free tensor), which performs slightly better. Note that the n-ranks depend only on the antenna configuration and not on the actual spatial frequencies. Hence, they can be determined beforehand from the n-ranks of the noise-free array steering tensor.

11.9. Summary

In this chapter we have shown how subspace-based parameter estimation schemes can be improved by exploiting the specific structure of the signal of interest. We have chosen the ESPRIT algorithm to demonstrate the enhancements, however, many of them can be applied to many different subspace-based parameter estimation schemes as well.

For multidimensional signals sampled on separable R-D grids we have seen that tensor calculus allows to improve the signal subspace estimate as long as the number of wavefronts is not too large. In this case, the multidimensional structure can be used for more efficient denoising, effectively filtering out all parts of the noise that do not obey the required structure. This enhanced signal subspace estimate can be combined with arbitrary subspace-based multidimenaional parameter estimation schemes.

Based on tensor calculus, a natural formulation of R-D shift invariance and the resulting R-D versions of Standard ESPRIT and Unitary ESPRIT can be found. We have also shown that if the tensor-valued shift invariance equations are solved via LS, the resulting Tensor-ESPRIT-type algorithms are identical to their matrix-based counterparts except for the signal subspace estimate which is replaced by the enhanced HOSVD-based signal subspace estimate.

We have also discussed more structured solutions of the invariance equations, namely the Structured Least Squares (SLS) algorithm and its tensor extension Tensor-Structure SLS (TS-SLS). As we have shown via simulations, even in the cases where the HOSVD-based subspace estimate provides no improvement, we can benefit from tensors via the TS-SLS solution to the invariance equations. This is particularly relevant in critical scenarios where the sources are highly correlated or the number of available snapshots is small.

For the case where the source signals can be modeled as strict-sense non-circular random variables we have introduced a class of ESPRIT-type algorithms that takes advantage of this fact. In the matrix-case, we have shown that by augmenting the rows of the measurement matrix by the conjugates of the original measurements, a virtual array steering matrix is created that has twice the number of elements and can hence resolve more sources. Based on this idea, the NC Standard ESPRIT and the NC Unitary ESPRIT algorithm can be defined. As we have seen numerically, NC ESPRIT-type algorithms outperform ESPRIT-type algorithms in terms of the estimation accuracy. The improvement is particularly pronounced for closely spaced sources that arrive at different phase angles φ_i.

To combine the benefits of multidimensional signals and strict-sense non-circular sources, a different manner of performing the augmentation is needed, since the matrix-based approach destroys the separable R-D sampling grid required for tensor-based algorithms. Based on a sequence of r-mode augmentations for $r = 1, 2, \ldots, R$ we have defined the R-D NC Standard Tensor-ESPRIT and the R-D NC Unitary Tensor-ESPRIT algorithm. Simulations have shown that they allow to combine both benefits, outperforming the matrix-based R-D NC ESPRIT-type algorithms as well as the R-D Tensor-ESPRIT-type algorithms which do not take advantage of the strict-sense non-circularity. A summary of the different ESPRIT-type algorithms and the different solutions to the overdetermined shift invariance equations is also shown in Chapter 13, in Tables 13.1 and 13.2, respectively.

It is important to point out that the modifications of the ESPRIT algorithm that are proposed here represent simple algebraic steps which do not significantly increase the computational complexity. This underlines the original idea to exploit given structures in the data in order to enhance the performance with simple algebraic modifications. Some of these are in fact quite generic. For instance, the HOSVD-based subspace estimate can be combined with arbitrary subspace-based algorithms in order to enhance their performance. Likewise, the widely linear processing performed for NC ESPRIT-type algorithms can be applied to many other schemes (e.g., the Root-MUSIC algorithm [CWS01] or Spectral MUSIC [AD06]).

12. Performance analysis

12.1. Overview

It is quite common in many engineering disciplines to compare the performance of different algorithms via Monte-Carlo simulations. That means that for a specific setting of the system parameters, the corresponding input data to the algorithm is generated in a random manner several times and the performance is evaluated by estimating some statistical measure (often the mean or the squared mean) via an average over many random trials. The main motivation for this procedure is its simplicity. Often it is too difficult to infer from the distribution of the input data directly on the distribution of the parameters of interest, due to the involved non-linear calculations in between.

Yet, this simplicity comes at a price. Monte-Carlo simulation results suffer from some major drawbacks one has to be aware of. Firstly, they are never truly generic since for the simulation specific assumptions about parameters or at least about their distributions have to be made. Secondly, the outcome is an estimate of the true parameter and it is often difficult to predict how accurate this estimate is. Thirdly, since they are based on randomly generated data, for which, typically, system-specific pseudo-random number generators are used, the results are never fully reproducible and hence always somewhat subjective.

So even if from a Monte-Carlo simulation result it looks like one algorithm outperforms the other this is nowhere close to a rigorous proof. This behavior is specific to the system setting considered in the simulation and it is non-objective as it may as well be the result of a random fluctuation in the data set that was considered. This is the main motivation for considering analytical performance assessments. These allow to compare different algorithms on an objective and generic basis without the need of performing Monte-Carlo simulations.

In this chapter we discuss an asymptotic performance analysis for subspace-based parameter estimation schemes which is based on an existing first order expansion of the subspace estimation error in a singular value decomposition. The underlying result is very generic since it represents an explicit first-order expansion in the perturbation term, which means that we do not need to make any assumptions about the statistics of this perturbation. The only assumption we need is that the perturbation is "small", that is to say, the result becomes exact only as the norm of the perturbation goes to zero compared to the norm of the desired signal component.

We demonstrate how to extend these existing results to the HOSVD-based subspace estimate and how they can be applied to compute the analytical performance of arbitrary ESPRIT-type algorithms, including Tensor-ESPRIT-type or NC ESPRIT-type algorithms. On top of the explicit expansion in terms of the perturbation term we derive the mean square estimation error for ESPRIT-type algorithms if the perturbation is circularly symmetric white noise noise. For the special case of a single source and uniform arrays we show how the expressions can be simplified to yield MSE expressions which depend only on the system parameters such as the number of antennas or the SNR. Such expressions allow interesting and practically relevant insights, for instance, how far the estimators are away from the ultimate lower bound on their variance given by the Cramér-Rao Lower Bound.

12.2. State of the art

Subspace-based parameter estimation schemes have been known for more than two decades. Especially the publication of the Multiple Signal Classification (MUSIC) algorithm [Sch86] and the ESPRIT algorithm [RPK86] have sparked a great interest in finding a suitable analytical performance assessment.

Such analytical results have been published very shortly afterwards, most frequently cited are [KB86] for the MUSIC algorithm and [RH89a] for ESPRIT. Many follow-up papers exist which extend the original results, e.g., [PK89b], [Fri90], [MZ94], [ZKM92], [MHZ96], and many others. However, these results have something in common: they all go more or less directly back to a result on the distribution of the eigenvectors of a sample covariance matrix first published in [Bri75].

In contrast to these results, in [LLV93] an entirely different approach was proposed, which provides an explicit first-order expansion of the subspace of a desired signal component if it is observed superimposed by a small additive perturbation. This approach has a number of advantages compared to [Bri75]. Firstly, [Bri75] is asymptotic in the sample size N, i.e., the result becomes only accurate as the number of snapshots N is very large, whereas [LLV93] is asymptotic in the effective SNR, i.e., it can be used even for $N = 1$ as long as the noise variance is sufficiently small. Secondly, [Bri75] requires strong Gaussianity assumptions, not only on the perturbation (i.e., the noise), but also on the source symbols. Since [LLV93] is explicit, no assumptions about the statistics of either the desired signal or the perturbation are needed. Thirdly, the covariance expressions from [Bri75] are much less intuitive than the expansion from [LLV93] which shows directly how much of the noise subspace “leaks into” the signal subspace due to the erroneous estimate. Finally, the expressions involved in [Bri75] are quite complex and tough to handle, whereas [LLV93] requires only a few terms which appear directly as block matrices of the SVD of the noise-free observation matrix.

Due to these advantages we clearly favor [LLV93] as a starting point. The authors in [LLV93] have already shown that their results on the perturbation of the subspace can be used to find a first order expansion for the MUSIC, the Root-MUSIC, the Min-Norm, the State-Space-Realization, and even the ESPRIT algorithm. However, they only considered 1-D Standard ESPRIT. We extend their work by considering multiple dimensions (R-D ESPRIT), by incorporating forward-backward-averaging (for Unitary ESPRIT), by considering the tensor-based subspace estimate (for Standard and Unitary Tensor-ESPRIT), by investigating the effect of using SLS to solve the invariance equation instead of the LS solution used in [LLV93], and by providing mean square error expressions of the resulting estimation errors in these cases.

There have been other follow-up papers based on [LLV93]. For instance, [Xu02] provides a first-order and second-order perturbation expansion which can be seen as a generalization of [LLV93]. In [LLM08] the authors show that there is also a first-order contribution of the perturbation of the signal subspace which lies in the signal subspace (which [Xu02] and [LLV93] have argued to be of second order and hence negligible). Based on [Xu02], subspace-based blind multiuser detection algorithms have been analyzed, e.g., in [ZLW04]. A mean square error expression for Standard ESPRIT is provided by the same authors in [LV92], however, it does not generalize easily to the tensor case. Moreover, other authors have studied the asymptotical performance of ESPRIT, e.g., [SS91, ESS93] where harmonic retrieval from time series is investigated and MSE expressions for a large number of snapshots as well as MSE expressions for a high SNR are derived. Note that we find MSE expressions compatible to [ESS93] by only assuming a high effective SNR, i.e., either the number of snapshots or the SNR can tend to infinity. Interestingly, [SS91, ESS93] also consider the special case for a single source. However, the expressions provided there are specific to harmonic retrieval from time series and they are not compared to the corresponding Cramér-Rao Bound. Some analytical results on the asymptotic efficiency of MUSIC, Root-MUSIC, ESPRIT, and TLS-ESPRIT are, among others, presented in [PF88], [RH89a], [RH89b], and [OVK91], respectively. However, these results are asymptotic in the number of snapshots N and sometimes even in the number of sensors M. The asymptotic equivalence of LS-ESPRIT, TLS-ESPRIT, Pro-ESPRIT, and the Matrix Pencil method has also been shown, see for instance [HS91]. Overall, in the matrix case, the number of existing results is quite large, since the underlying methods have been known for more than two decades. However, we emphasize again that the main novelty of the work presented here does not reside in the performance analysis for the matrix-based Standard ESPRIT algorithm but in its extensions to the tensor case and to the incorporation of Structured Least Squares.

In the tensor case a first-order expansion for the HOSVD has been proposed in [dLdMV00a]. However, it is not suitable for our application since it does not consider the HOSVD-based subspace estimate but the subspaces of the separate n-mode unfoldings and their singular values. Moreover, the perturbation is modeled via a single scalar real-valued parameter ϵ. A first-order expansion for the best rank-(R_1, R_2, R_3)-expansion is provided in [dL04b]. However, again, it is not directly applicable for analyzing the HOSVD-based subspace estimate as it investigates the approximation error of the entire tensor. Consequently, our approach to analyze the HOSVD-based subspace estimate based on the link to the SVD-based subspace estimate via a structured projection is entirely novel. Moreover, the application of these results to find the analytical performance of Tensor-ESPRIT-type algorithms is novel as well. It is a particular strength of the framework we use that many extensions and modifications of ESPRIT are easily incorporated, e.g., Forward-Backward-Averaging (cf. Section 12.4.4), ESPRIT for strict-sence non-circular sources (cf. Section 12.4.5), or Structured Least Squares (cf. Section 12.4.6).

12.3. Performance of tensor-based subspace estimation

12.3.1. Review of perturbation results for the SVD

Let us first review the results from [LLV93] which are relevant to the discussion in this section. Following the model introduced in Chapter 10 we let $\boldsymbol{X} = \boldsymbol{X}_0 + \boldsymbol{N} \in \mathbb{C}^{M \times N}$ be the measurement matrix comprising N subsequent observations from M channels ($\boldsymbol{X}_0$) under additive noise ($\boldsymbol{N}$). Following (10.2) and (10.3), the SVD of $\boldsymbol{X}$ and $\boldsymbol{X}_0$ can be expressed as

$$\boldsymbol{X}_0 = \begin{bmatrix} \boldsymbol{U}_{\mathrm{s}} & \boldsymbol{U}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_{\mathrm{s}} & \boldsymbol{0}_{d \times (N-d)} \\ \boldsymbol{0}_{(M-d) \times d} & \boldsymbol{0}_{(M-d) \times (N-d)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{V}_{\mathrm{s}} & \boldsymbol{V}_{\mathrm{n}} \end{bmatrix}^{\mathrm{H}} \tag{12.1}$$

$$\boldsymbol{X} = \begin{bmatrix} \hat{\boldsymbol{U}}_{\mathrm{s}} & \hat{\boldsymbol{U}}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{\mathrm{s}} & \boldsymbol{0}_{d \times (N-d)} \\ \boldsymbol{0}_{(M-d) \times d} & \hat{\boldsymbol{\Sigma}}_{\mathrm{n}} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{V}}_{\mathrm{s}} & \hat{\boldsymbol{V}}_{\mathrm{n}} \end{bmatrix}^{\mathrm{H}}, \tag{12.2}$$

where the columns of $\boldsymbol{U}_{\mathrm{s}} \in \mathbb{C}^{M \times d}$ provide an orthonormal basis for the signal subspace which we want to estimate and the "hat" denotes the estimated quantities. Moreover $\boldsymbol{\Sigma}_{\mathrm{s}} = \operatorname{diag}\left\{\left[\sigma_1, \sigma_2, \ldots, \sigma_d\right]\right\} \in \mathbb{R}^{d \times d}$ contains the d non-zero singular values on its main diagonal. We can write $\hat{\boldsymbol{U}}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s}}$, where $\Delta \boldsymbol{U}_{\mathrm{s}}$ represents the estimation error. At this point, we are ready to state the main result on the first order perturbation expansion of $\Delta \boldsymbol{U}_{\mathrm{s}}$ from [LLV93]

$$\Delta \boldsymbol{U}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{\Gamma}_{\mathrm{n}} + \mathcal{O}\left\{\Delta^2\right\}, \text{ where } \Delta = \|\boldsymbol{N}\| \text{ and} \tag{12.3}$$

$$\boldsymbol{\Gamma}_{\mathrm{n}} = \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}} \cdot \boldsymbol{N} \cdot \boldsymbol{V}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}}^{-1} \in \mathbb{C}^{(M-d)\times d} \tag{12.4}$$

Here $\|.\|$ represents an arbitrary sub-multiplicative[1] norm, e.g., the Frobenius norm. Equation (12.4) shows the first order expansion of the signal subspace estimation error $\Delta \boldsymbol{U}_{\mathrm{s}}$ in terms of the noise subspace $\boldsymbol{U}_{\mathrm{n}}$, i.e., how much of the noise subspace "leaks into" the signal subspace due to the estimation errors from the perturbation $\boldsymbol{N}$. Since it is explicit in $\boldsymbol{N}$ it makes no assumptions about the *statistics* of $\boldsymbol{N}$, in fact, it is purely deterministic.

The expansion (12.4) only models the leakage of the noise subspace into the signal subspace. That is to say, the perturbation of the particular basis (the columns of $\boldsymbol{U}_{\mathrm{s}}$) is ignored. While for subspace-based parameter estimation schemes this is indeed sufficient since the particular choice of the basis is irrelevant, there are other applications where this term matters. For instance, in a communication system where the channel is decomposed into its individual eigenmodes and one or several of these eigenmodes are used for transmission, such errors have a major impact. Therefore, other authors have extended (12.4) to take this error term into account. For instance, in [LLM08] the authors provide the following expansion

$$\Delta \boldsymbol{U}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{\Gamma}_{\mathrm{n}} + \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Gamma}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\}, \text{ where} \tag{12.5}$$

$$\boldsymbol{\Gamma}_{\mathrm{s}} = \boldsymbol{D} \odot \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{H}} \cdot \boldsymbol{N} \cdot \boldsymbol{V}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}} + \boldsymbol{\Sigma}_{\mathrm{s}} \cdot \boldsymbol{V}_{\mathrm{s}}^{\mathrm{H}} \cdot \boldsymbol{N}^{\mathrm{H}} \cdot \boldsymbol{U}_{\mathrm{s}}\right) \in \mathbb{C}^{p_r \times p_r}. \tag{12.6}$$

Here, the matrix $\boldsymbol{D}$ is defined as

$$[\boldsymbol{D}]_{(k,\ell)} = \begin{cases} \frac{1}{\sigma_\ell^2 - \sigma_k^2} & k \neq \ell \\ 0 & k = \ell \end{cases} \quad \text{for } k, \ell = 1, 2, \dots, d. \tag{12.7}$$

Equation (12.5) additionally shows the perturbation of the individual singular vectors via the term $\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Gamma}_{\mathrm{s}}$. This term can be dropped for the evaluation of subspace-based parameter estimation schemes since for these, the particular choice of the basis is irrelevant. Therefore we do not consider it in Section 12.4 where ESPRIT-type algorithms are investigated. However, we show its impact in the simulation results in Section 12.5.1 where the subspace estimation accuracy is evaluated.

[1] A matrix norm is called submultiplicative if $\|\boldsymbol{A} \cdot \boldsymbol{B}\| \leq \|\boldsymbol{A}\| \cdot \|\boldsymbol{B}\|$ for arbitrary matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

12.3.2. Extension to the HOSVD-based subspace estimate

As we have shown in Section 10.2, in the multidimensional case, an improved signal subspace estimate can be computed via the HOSVD of the measurement tensor $\boldsymbol{\mathcal{X}}$. Since the HOSVD is obtained from SVDs of the unfoldings, we can apply the same framework to find a perturbation expansion of the HOSVD-based subspace estimate. Following (10.4) and (10.5), the SVDs of the r-mode unfoldings of $\boldsymbol{\mathcal{X}}$ and the noise-free tensor $\boldsymbol{\mathcal{X}}_0$ are given by

$$[\boldsymbol{\mathcal{X}}_0]_{(r)} = \begin{bmatrix} \boldsymbol{U}_r^{[\mathrm{s}]} & \boldsymbol{U}_r^{[\mathrm{n}]} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_r^{[\mathrm{s}]} & \mathbf{0}_{d\times(M\cdot N/M_r - d)} \\ \mathbf{0}_{(M_r-d)\times d} & \mathbf{0}_{(M_r-d)\times(M\cdot N/M_r-d)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{V}_r^{[\mathrm{s}]} & \boldsymbol{V}_r^{[\mathrm{n}]} \end{bmatrix}^{\mathrm{H}} \tag{12.8}$$

$$[\boldsymbol{\mathcal{X}}]_{(r)} = \begin{bmatrix} \hat{\boldsymbol{U}}_r^{[\mathrm{s}]} & \hat{\boldsymbol{U}}_r^{[\mathrm{n}]} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_r^{[\mathrm{s}]} & \mathbf{0}_{d\times(M\cdot N/M_r - d)} \\ \mathbf{0}_{(M_r-d)\times d} & \hat{\boldsymbol{\Sigma}}_r^{[\mathrm{n}]} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{V}}_r^{[\mathrm{s}]} & \hat{\boldsymbol{V}}_r^{[\mathrm{n}]} \end{bmatrix}^{\mathrm{H}}. \tag{12.9}$$

where $\boldsymbol{\Sigma}_r^{[\mathrm{s}]} = \operatorname{diag}\left\{[\sigma_1^{(r)}, \sigma_2^{(r)}, \dots, \sigma_d^{(r)}]\right\}$ and $r = 1, 2, \dots, R$. Note that since (12.8) and (12.9) are in fact SVDs, we can apply (12.5) and find $\hat{\boldsymbol{U}}_r^{[\mathrm{s}]} = \boldsymbol{U}_r^{[\mathrm{s}]} + \Delta\boldsymbol{U}_r^{[\mathrm{s}]}$ where

$$\Delta\boldsymbol{U}_r^{[\mathrm{s}]} = \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]} + \mathcal{O}\left\{\Delta^2\right\}, \tag{12.10}$$

$$\boldsymbol{\Gamma}_r^{[\mathrm{n}]} = \boldsymbol{U}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)} \cdot \boldsymbol{V}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_r^{[\mathrm{s}]^{-1}}, \tag{12.11}$$

$$\boldsymbol{\Gamma}_r^{[\mathrm{s}]} = \boldsymbol{D}_r \odot \left(\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)} \cdot \boldsymbol{V}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_r^{[\mathrm{s}]} + \boldsymbol{\Sigma}_r^{[\mathrm{s}]} \cdot \boldsymbol{V}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)}^{\mathrm{H}} \cdot \boldsymbol{U}_r^{[\mathrm{s}]}\right) \quad \text{and} \tag{12.12}$$

$$[\boldsymbol{D}_r]_{(k,\ell)} = \begin{cases} \frac{1}{\sigma_\ell^{(r)^2} - \sigma_k^{(r)^2}} & k \neq \ell \\ 0 & k = \ell \end{cases} \quad \text{for } k, \ell = 1, 2, \dots, d. \tag{12.13}$$

Our goal is to use the perturbation of the r-mode unfoldings to find a corresponding expansion for the HOSVD-based subspace estimate introduced in Section 10.2. As we have seen in (10.10), the subspace estimate can be computed via $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$, where

$$\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \dots \times_R \hat{\boldsymbol{U}}_R^{[\mathrm{s}]} \times_{R+1} \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}}. \tag{12.14}$$

The crucial point to develop a perturbation expansion for the HOSVD-based subspace estimate $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ is an algebraic relation to the SVD-based subspace estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ which was shown in Theorem 10.2.1 in Section 10.2. We restate the relation here for convenience

$$\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = \left(\hat{\boldsymbol{T}}_1 \otimes \hat{\boldsymbol{T}}_2 \otimes \dots \hat{\boldsymbol{T}}_R\right) \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}, \tag{12.15}$$

where $\hat{\boldsymbol{T}}_r \in \mathbb{C}^{M_r \times M_r}$ represent estimates of the projection matrices onto the r-spaces of $\boldsymbol{\mathcal{X}}_0$, which are computed via $\hat{\boldsymbol{T}}_r = \hat{\boldsymbol{U}}_r^{[\mathrm{s}]} \hat{\boldsymbol{U}}_r^{[\mathrm{s}]^{\mathrm{H}}}$.

Equation (12.15) shows that a perturbation expansion for $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ can be developed based on the subspaces of all $R+1$ unfoldings, as the core tensor is not needed for its computation. A first order perturbation expansion for the HOSVD-based subspace estimate based on this idea is summarized in the following proposition.

Proposition 12.3.1. **[RBHW09]** *A first order expansion for* $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ *can be expressed as* $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = \boldsymbol{U}_{\mathrm{s}} + \left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ *where*

$$
\begin{aligned}
\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} = (\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \ldots \otimes \boldsymbol{T}_R) \cdot \Delta\boldsymbol{U}_{\mathrm{s}} &+ \left(\left[\Delta\boldsymbol{U}_1^{[\mathrm{s}]} \cdot \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}}\right] \otimes \boldsymbol{T}_2 \otimes \ldots \otimes \boldsymbol{T}_R\right) \cdot \boldsymbol{U}_{\mathrm{s}} \\
&+ \left(\boldsymbol{T}_1 \otimes \left[\Delta\boldsymbol{U}_2^{[\mathrm{s}]} \cdot \boldsymbol{U}_2^{[\mathrm{s}]^{\mathrm{H}}}\right] \otimes \ldots \otimes \boldsymbol{T}_R\right) \cdot \boldsymbol{U}_{\mathrm{s}} \\
&+ \ldots \\
&+ \left(\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \ldots \otimes \left[\Delta\boldsymbol{U}_R^{[\mathrm{s}]} \cdot \boldsymbol{U}_R^{[\mathrm{s}]^{\mathrm{H}}}\right]\right) \cdot \boldsymbol{U}_{\mathrm{s}} \\
&+ \mathcal{O}\left\{\Delta^2\right\},
\end{aligned}
\tag{12.16}
$$

where the SVD-based signal subspace perturbation $\Delta\boldsymbol{U}_{\mathrm{s}}$ *is given by* (12.5) *and the perturbation of the* r*-space can be computed via*

$$
\Delta\boldsymbol{U}_r^{[\mathrm{s}]} = \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} = \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{U}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)} \cdot \boldsymbol{V}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_r^{[\mathrm{s}]^{-1}}. \tag{12.17}
$$

Proof: cf. Appendix D.10.

Note that while $\Delta\boldsymbol{U}_{\mathrm{s}}$ in general contains both perturbation terms $\boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{\Gamma}_{\mathrm{n}}$ and $\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Gamma}_{\mathrm{s}}$, for $\Delta\boldsymbol{U}_r^{[\mathrm{s}]}$ the term $\boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]}$ cancels. This is not surprising since the r-mode subspaces enter (12.15) only via projection matrices for which the choice of the particular basis is irrelevant. A special case of this theorem for $R = 2$ was first shown by us in [RBHW09].

12.4. Performance of ESPRIT-type algorithms

12.4.1. Review of perturbation results for the 1-D Standard ESPRIT

In [LLV93] the authors point out that once a first order expansion of the subspace estimation error is available it can be used to find a corresponding first order expansion of the estimation error of a suitable parameter estimation scheme. One of the examples the authors show is the 1-D Standard ESPRIT algorithm based on LS, which we use as a starting point to discuss various ESPRIT-type algorithms in this section. In the noise-free case, the shift invariance equation for 1-D standard ESPRIT can be expressed as (cf. (11.6))

$$\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} = \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}, \tag{12.18}$$

where $\boldsymbol{J}_1, \boldsymbol{J}_2 \in \mathbb{R}^{M^{(\mathrm{sel})} \times M}$ are the selection matrices that select the $M^{(\mathrm{sel})}$ elements from the M antenna elements which correspond to the first and the second subarray, respectively. Moreover, $\boldsymbol{\Psi} = \boldsymbol{Q} \cdot \boldsymbol{\Phi} \cdot \boldsymbol{Q}^{-1}$, where $\boldsymbol{\Phi} = \mathrm{diag}\left\{\left[\mathrm{e}^{\jmath\mu_1}, \quad \ldots, \quad \mathrm{e}^{\jmath\mu_d}\right]\right\} \in \mathbb{C}^{d \times d}$ contains the spatial frequencies μ_k, $k = 1, 2, \ldots, d$ that we want to estimate. Therefore, $\mu_k = \arg\left\{\mathrm{EV}_k\left\{\boldsymbol{\Psi}\right\}\right\}$, i.e., the k-th spatial frequency is obtained from the phase of the k-th eigenvalue ($\mathrm{EV}_k\left\{\cdot\right\}$) of $\boldsymbol{\Psi}$.

In presence of noise, we only have an estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ of the signal subspace $\boldsymbol{U}_{\mathrm{s}}$. Consequently, (12.18) does in general not have an exact solution anymore. A simple way of finding an approximate $\hat{\boldsymbol{\Psi}}$ is given by the LS solution which can be expressed as

$$\hat{\boldsymbol{\Psi}}_{\mathrm{LS}} = \left(\boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}\right)^{+} \cdot \boldsymbol{J}_2 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}. \tag{12.19}$$

To simplify the notation we skip the index "LS" for the remainder of this section (since only LS is considered) and pick it up again in the next section where we expand the discussion to SLS.

For the estimation error of the k-th spatial frequency corresponding to the LS solution from (12.19), [LLV93] provides the following expansion

$$\Delta\mu_k = \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot [\boldsymbol{J}_2/\lambda_k - \boldsymbol{J}_1] \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{12.20}$$

where $\lambda_k = \mathrm{e}^{\jmath\mu_k}$ and $\boldsymbol{q}_k$ is the k-th column of $\boldsymbol{Q}$. Moreover, $\boldsymbol{p}_k^{\mathrm{T}}$ represents the k-th row vector of the matrix $\boldsymbol{P} = \boldsymbol{Q}^{-1}$. A sketch of the derivation of (12.20) from [LLV93] is shown in Appendix D.11. Note that $\Delta\boldsymbol{U}_{\mathrm{s}}$ can be expanded in terms of the perturbation term $\boldsymbol{N}$ by directly using the expansion (12.4). The additional term from (12.5) is not needed as it is irrelevant for the performance of ESPRIT.

12.4.2. Extension to R-D Standard (Tensor-)ESPRIT

The previous result from [LLV93] on the first order perturbation expansion of 1-D Standard ESPRIT using LS is easily generalized to the R-D case. The reason is that for R-D LS-based

ESPRIT, the R shift invariance equations are solved independently from each other[2]. Hence, the arguments from [LLV93] are readily applied to all modes individually and we directly obtain a first order expansion for the estimation error of the k-th spatial frequency in the r-th mode

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}\right)^+ \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)} / \lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right] \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{12.21}$$

where $\tilde{\boldsymbol{J}}_1^{(r)}, \tilde{\boldsymbol{J}}_2^{(r)} \in \mathbb{R}^{\frac{M}{M_r} \cdot M_r^{(\mathrm{sel})} \times M}$ are the effective R-D selection matrix for the first and the second subarray in the r-th mode, respectively. They can be expressed as (cf. (11.5))

$$\tilde{\boldsymbol{J}}_\ell^{(r)} = \boldsymbol{I}_{\prod_{n=1}^{r-1} M_n} \otimes \boldsymbol{J}_\ell^{(r)} \otimes \boldsymbol{I}_{\prod_{n=r+1}^{R} M_n} \tag{12.22}$$

for $\ell = 1, 2$ and $r = 1, 2, \ldots, R$, where $\boldsymbol{J}_\ell^{(r)} \in \mathbb{R}^{M_r^{(\mathrm{sel})} \times M_r}$ are the selection matrices which select the $M_r^{(\mathrm{sel})}$ elements belonging to the first and the second subarray in the r-th mode, respectively, [HN98].

Since this expansion for R-D Standard ESPRIT is explicit in the perturbation of the subspace estimate and R-D Standard Tensor-ESPRIT only differs in the fact that it uses the enhanced HOSVD-based subspace estimate, we immediately conclude that a first order perturbation expansion for R-D Standard Tensor-ESPRIT is given by

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}\right)^+ \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)} / \lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right] \cdot \left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} \cdot \boldsymbol{q}_k\right\} + \mathcal{O}\left\{\Delta^2\right\}. \tag{12.23}$$

An explicit expansion of $\Delta\mu_k^{(r)}$ in terms of the noise tensor $\boldsymbol{\mathcal{N}}$ is given by inserting the previous result (12.16).

12.4.3. Mean Square Errors

As explained above, the advantage of the first order perturbation expansion we have discussed so far is that it is explicit in the perturbation term $\boldsymbol{N}$ (or $\boldsymbol{\mathcal{N}}$). Hence, it requires no assumptions about its distribution. However, it is often also desirable to know the mean square error if a

[2] If the shift invariance equations are solved completely independently, the correct pairing of the parameters across dimensions has to be found in a subsequent step. This is often avoided by computing the LS solutions for $\boldsymbol{\Psi}^{(r)}$ independently but then performing a joint eigendecomposition of all R dimensions to yield $\boldsymbol{\Phi}^{(r)}$. This step is not included in the performance analysis presented in this section, since no performance results on joint eigendecompositions are available and it appears to be a very difficult task. Moreover, this step has indeed no impact on the asymptotic estimation error of the spatial frequencies for high SNRs since the eigenvectors become asymptotically equal. As shown in [LT78] (cf. Appendix D.11), the impact of the perturbation of the eigenvectors is of second-order and can hence be ignored in a first-order perturbation analysis.

specific distribution is assumed and the ensemble average over all possible noise realizations is computed.

The following theorem summarizes the result of performing this task for the special case of zero mean, circularly symmetric, i.i.d. noise. For simplicity we consider the special case $R = 2$ for Standard Tensor-ESPRIT, however, a generalization to a larger number of dimensions is quite straightforward.

Theorem 12.4.1. **[RBH10]** *Assume that the entries of the perturbation term $\boldsymbol{N}$ or $\boldsymbol{\mathcal{N}}$ are mutually uncorrelated, zero mean, circularly symmetric random variables with identical variance σ_{n}^2. Then, the mean square estimation error for the k-th spatial frequency in the r-th mode is given by*

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\| \boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)} \right\|_2^2 \tag{12.24}$$

for R-D Standard ESPRIT and

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\| \boldsymbol{W}_{\mathrm{ten}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)} \right\|_2^2 \tag{12.25}$$

for 2-D Standard Tensor-ESPRIT[3] *where $k = 1, 2, \ldots, d$ and $r = 1, 2, \ldots, R$. The vector $\boldsymbol{r}_k^{(r)}$ and the matrices $\boldsymbol{W}_{\mathrm{mat}}$ and $\boldsymbol{W}_{\mathrm{ten}}$ are given by*

$$\boldsymbol{r}_k^{(r)} = \boldsymbol{q}_k \otimes \left(\left[\left(\tilde{\boldsymbol{J}}_1^{(r)} \boldsymbol{U}_{\mathrm{s}} \right)^+ \left(\tilde{\boldsymbol{J}}_2^{(r)} / \mathrm{e}^{\jmath \cdot \mu_k^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)} \right) \right]^{\mathrm{T}} \cdot \boldsymbol{p}_k \right) \tag{12.26}$$

$$\boldsymbol{W}_{\mathrm{mat}} = \left(\boldsymbol{\Sigma}_{\mathrm{s}}^{-1} \cdot \boldsymbol{V}_{\mathrm{s}}^{\mathrm{T}} \right) \otimes \left(\boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}} \right) \tag{12.27}$$

$$\begin{aligned} \boldsymbol{W}_{\mathrm{ten}} = {} & \left(\boldsymbol{\Sigma}_3^{[\mathrm{s}]^{-1}} \boldsymbol{U}_3^{[\mathrm{s}]^{\mathrm{H}}} \right) \otimes \left([\boldsymbol{T}_1 \otimes \boldsymbol{T}_2] \, \boldsymbol{V}_3^{[\mathrm{n}]^*} \boldsymbol{V}_3^{[\mathrm{n}]^{\mathrm{T}}} \right) \\ & + \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M \right) \bar{\boldsymbol{T}}_2 \left(\boldsymbol{U}_1^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \boldsymbol{V}_1^{[\mathrm{s}]^{\mathrm{T}}} \otimes \boldsymbol{U}_1^{[\mathrm{n}]} \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}} \right) \cdot \boldsymbol{K}_{M_2 \times (M_1 \cdot N)} \\ & + \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M \right) \bar{\boldsymbol{T}}_1 \left(\boldsymbol{U}_2^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_2^{[\mathrm{s}]^{-1}} \boldsymbol{V}_2^{[\mathrm{s}]^{\mathrm{T}}} \otimes \boldsymbol{U}_2^{[\mathrm{n}]} \boldsymbol{U}_2^{[\mathrm{n}]^{\mathrm{H}}} \right) \quad \textit{where} \end{aligned} \tag{12.28}$$

$$\bar{\boldsymbol{T}}_1 = \begin{bmatrix} \boldsymbol{I}_{M_2} \otimes \boldsymbol{t}_{1,1} \\ \vdots \\ \boldsymbol{I}_{M_2} \otimes \boldsymbol{t}_{1,M_1} \end{bmatrix} \otimes \boldsymbol{I}_{M_2}, \quad \bar{\boldsymbol{T}}_2 = \boldsymbol{I}_{M_1} \otimes \begin{bmatrix} \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,1} \\ \vdots \\ \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,M_2} \end{bmatrix}, \tag{12.29}$$

and $\boldsymbol{t}_{r,m}$ is the m-th column of $\boldsymbol{T}_r$. Finally, $\boldsymbol{K}_{p,q}$ is the commutation matrix (cf. eqn. (3.34)).

Proof: cf. Appendix D.12.

[3] For $R > 2$, (12.25) has the same form, however, $\boldsymbol{W}_{\mathrm{ten}}$ shown in (12.28) is only developed for $R = 2$ here.

Note that for 1-D Standard ESPRIT, this MSE expression agrees with the one shown in [LV92]. However, [LV92] does not directly generalize to the tensor case. This is the advantage of the MSE expressions in Theorem 12.4.1 where we only need to replace $\boldsymbol{W}_{\mathrm{mat}}$ by $\boldsymbol{W}_{\mathrm{ten}}$ to account for the enhanced signal subspace estimate. The MSE expressions in Theorem 12.4.1 were already shown by us in [RBH10]. However, due to space limitations, a proof for them was not included there.

12.4.4. Incorporation of Forward-Backward-Averaging

So far we have shown the explicit expansion and the MSE expressions for R-D Standard ESPRIT and R-D Standard Tensor-ESPRIT. In order to extend these results to Unitary-ESPRIT-type algorithms we need to incorporate the mandatory preprocessing for Unitary ESPRIT which is given by Forward-Backward-Averaging.

As discussed in Section 10.3, Forward-Backward-Averaging augments the N observations of the sampled R-D signal by N new "virtual" observations which are a conjugated and row- as well as column-flipped version of the original ones [HN95]. This can be expressed in matrix form as shown in (10.12), which we restate here for convenience

$$\boldsymbol{X}^{(\mathrm{fba})} = \begin{bmatrix} \boldsymbol{X} & \boldsymbol{\Pi}_M \cdot \boldsymbol{X}^* \cdot \boldsymbol{\Pi}_N \end{bmatrix} \in \mathbb{C}^{M \times 2N}. \tag{12.30}$$

Inserting $\boldsymbol{X} = \boldsymbol{X}_0 + \boldsymbol{N}$ we find

$$\boldsymbol{X}^{(\mathrm{fba})} = \begin{bmatrix} \boldsymbol{X}_0 & \boldsymbol{\Pi}_M \cdot \boldsymbol{X}_0^* \cdot \boldsymbol{\Pi}_N \end{bmatrix} + \begin{bmatrix} \boldsymbol{N} & \boldsymbol{\Pi}_M \cdot \boldsymbol{N}^* \cdot \boldsymbol{\Pi}_N \end{bmatrix} = \boldsymbol{X}_0^{(\mathrm{fba})} + \boldsymbol{N}^{(\mathrm{fba})}. \tag{12.31}$$

However, the latter relation shows that we are interested in the perturbation of the subspace of a matrix $\boldsymbol{X}_0^{(\mathrm{fba})}$ superimposed by an additive perturbation $\boldsymbol{N}^{(\mathrm{fba})}$, which is small. Since the explicit perturbation expansion we have used up to this point requires no additional assumptions, the surprisingly simple answer is that we do not need to change anything but we can apply the previous results directly. All we need to do is to replace all exact (noise-free) subspaces of $\boldsymbol{X}_0$ by the corresponding subspaces of $\boldsymbol{X}_0^{(\mathrm{fba})}$. From (12.21), we immediately obtain the following explicit first-order expansion which is valid for R-D Standard ESPRIT with Forward-Backward Averaging

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}\right)^+ \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)} / \lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right] \cdot \Delta\boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{12.32}$$

where $\Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$ is given by

$$\Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} = \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})} \cdot \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})^{\mathrm{H}}} \cdot \boldsymbol{N}^{(\mathrm{fba})} \cdot \boldsymbol{V}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{\Sigma}_{\mathrm{s}}^{(\mathrm{fba})^{-1}} \tag{12.33}$$

and $\boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$, $\boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})}$, $\boldsymbol{V}_{\mathrm{s}}^{(\mathrm{fba})}$, $\boldsymbol{\Sigma}_{\mathrm{s}}^{(\mathrm{fba})}$ correspond to the signal subspace, the noise subspace, the row space, and the singular values of $\boldsymbol{X}_0^{(\mathrm{fba})}$, respectively. Likewise, $\boldsymbol{q}_k^{(\mathrm{fba})}$ and $\boldsymbol{p}_k^{(\mathrm{fba})}$ represent the corresponding versions of $\boldsymbol{q}_k$ and $\boldsymbol{p}_k$ if $\boldsymbol{U}_{\mathrm{s}}$ is replaced by $\boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$ in the shift invariance equations.

The second step in Unitary ESPRIT is the transformation into the real-valued domain. However, as the following theorem shows, this step has no impact on the performance for high SNRs.

Theorem 12.4.2. *The first-order perturbation for the estimation error in the spatial frequencies $\Delta\mu_k$ for Unitary ESPRIT is equal to the corresponding error for Standard ESPRIT including forward-backward averaging.*

Proof: cf. Appendix D.13.

Therefore, the asymptotic performance of Unitary-ESPRIT-type algorithms is found once Forward-Backward-Averaging is taken into account and the real-valued transformation can be ignored. Consequently, the expansion shown in (12.32) is valid for R-D Unitary ESPRIT.

With the same reasoning, an explicit expansion for R-D Unitary Tensor-ESPRIT is obtained by consistently replacing $\boldsymbol{\mathcal{X}}_0$ by $\boldsymbol{\mathcal{X}}_0^{(\mathrm{fba})}$ in (12.23), i.e.,

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}\right)^{+} \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)} / \lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right] \cdot \left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}](\mathrm{fba})}\right]_{(R+1)}^{\mathrm{T}} \cdot \boldsymbol{q}_k^{(\mathrm{fba})}\right\} + \mathcal{O}\left\{\Delta^2\right\}. \tag{12.34}$$

where $\boldsymbol{\mathcal{X}}_0^{(\mathrm{fba})}$ can be expressed in tensor notation as

$$\boldsymbol{\mathcal{X}}_0^{(\mathrm{fba})} = \left[\boldsymbol{\mathcal{X}}_0 \sqcup_{R+1} \boldsymbol{\mathcal{X}}_0^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N\right] \tag{12.35}$$

The situation changes slightly if mean square error expressions are considered since the Forward-Backward-Averaging does have an impact on the statistics of the perturbation $\boldsymbol{N}$. The following theorem summarizes the corresponding MSE results:

Theorem 12.4.3. *Assume that the entries of the perturbation term $\boldsymbol{N}$ or $\boldsymbol{\mathcal{N}}$ are mutually uncorrelated, zero mean, circularly symmetric random variables with identical variance σ_{n}^2. Then, the mean square estimation error for the k-th spatial frequency in the r-th mode is given* *by*

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \frac{\sigma_{\rm n}^2}{2} \cdot \left(\left\| \boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}} \right\|_2^2 - {\rm Re}\left\{ \boldsymbol{r}_k^{(r)^{({\rm fba})^{\rm T}}} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})} \cdot \boldsymbol{\Pi}_{2MN} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}} \right\} \right) \tag{12.36}$$

for R-D Unitary ESPRIT and

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \frac{\sigma_{\rm n}^2}{2} \cdot \left(\left\| \boldsymbol{W}_{\rm ten}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}} \right\|_2^2 - {\rm Re}\left\{ \boldsymbol{r}_k^{(r)^{({\rm fba})^{\rm T}}} \cdot \boldsymbol{W}_{\rm ten}^{({\rm fba})} \cdot \boldsymbol{\Pi}_{2MN} \cdot \boldsymbol{W}_{\rm ten}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}} \right\} \right) \tag{12.37}$$

for 2-D Unitary Tensor-ESPRIT, where $\boldsymbol{r}_k^{(r)^{({\rm fba})}}$*,* $\boldsymbol{W}_{\rm mat}^{({\rm fba})}$*, and* $\boldsymbol{W}_{\rm ten}^{({\rm fba})}$ *are computed as in Theorem 12.4.1 by consistently replacing all quantities by their forward-backward-averaged equivalents.*

Proof: cf. Appendix D.14.

We observe that due to the Forward-Backward-Averaging, an additional term must be added to the MSE expressions. However, the first term still has a similar form than in the case of R-D Standard (Tensor-)ESPRIT. The MSE expressions in Theorem 12.4.3 were already shown by us in [RBH10]. However, due to space limitations, a proof for them was not included there.

It is important to note that Theorem 12.4.2 and 12.4.3 relate to the LS solution only. If TLS is used instead, the equivalence of Standard ESPRIT with Forward-Backward-Averaging and Unitary ESPRIT is shown in [HN95].

12.4.5. Extension to other ESPRIT-type algorithms

In a similar manner as in the previous section, other ESPRIT-type algorithms can be analyzed. For instance, the NC Standard ESPRIT and NC Unitary ESPRIT algorithm for strict-sense non-circular sources (cf. Section 11.5) are based on a different kind of preprocessing where instead of augmenting the columns we augment the rows of the measurement matrix. Yet, the explicit first order perturbation expansion still applies since the result can be written as a noise-free (augmented) measurement matrix superimposed by a small (augmented) perturbation matrix. Consequently, for the explicit expansion we only need to consistently replace the quantities originating from the SVD of $\boldsymbol{X}_0$ by the corresponding quantities from the appropriately preprocessed measurement matrix $\boldsymbol{X}_0^{({\rm nc})}$. However, for the evaluation of the mean square error, the altered noise statistics have to be taken into account.

Another possible extension is to incorporate spatial smoothing. As discussed in Section 10.3, if sources are mutually coherent, preprocessing must be applied to the data to decorrelate the sources prior to any subspace-based parameter estimation scheme. Via Forward-Backward-Averaging, two sources can be decorrelated. However, if more than two sources are coherent (or if FBA cannot be applied), additional preprocessing is needed. For spatial smoothing we divide the array into a number of identical displaced subarrays and average the spatial covariance matrix over these subarrays. Since the number of subarrays we choose is a design parameter influencing the performance, investigating its effect by virtue of an analytical performance assessment would be desirable. Note that the spatial averaging introduces a correlation into the noise. Therefore, the presented framework is particularly attractive since for the explicit expansion, no assumptions about the noise statistics are needed. A further extension is the performance assessment of tensor-based schemes for spatial smoothing. We have introduced a tensor-based formulation of spatial smoothing for R-D signals in [HRD08]. Moreover, a tensor-based spatial smoothing technique for 1-D damped and undamped harmonic retrieval with a single snapshot is shown in [THG09b]. The extension to multiple snapshots is introduced in [THRG10] and an R-D extension is shown in [THG09a]. A major advantage of [THRG10, THG09a] is that the performance of the ESPRIT-type parameter estimates is almost independent of the choice of the subarray size. This could be verified by analytical results if the performance analysis is extended accordingly.

12.4.6. Incorporation of Structured Least Squares

So far, all performance results are based on ESPRIT using LS, i.e., the overdetermined shift invariance equations are solved using LS only. As discussed in Section 11.7, the LS solution to the shift invariance equation is in general suboptimal as errors on both sides of the equations need to be taken into account. Even more so, since for overlapping subarrays, the shift invariance equation has a specific structure resulting in common error terms on both sides of the equations, this structure should be taken into account when solving them. The corresponding SLS procedure is discussed in Section 11.7.

Since it has been shown that the resulting ESPRIT algorithm using SLS outperforms ESPRIT using LS and TLS for overlapping subarrays [Haa97b], it is desirable to extend our performance analysis results to SLS-based ESPRIT as well.

Due to the fact that our analysis is asymptotic in the SNR we can make the following simplifying assumptions for SLS. Firstly, we consider only a single iteration. Not only does this follow the suggestion made in [Haa97b], it is also optimal for high SNRs, since the underlying cost function is quadratic but actually asymptotically linear (the quadratic term vanishes against the linear terms for high SNRs). Secondly, we do not consider the optional regularization term in SLS (i.e., we set the corresponding regularization parameter α to infinity) as regularization is typically not needed for high SNRs.

Under these conditions we derive the following results on the estimation error of ESPRIT using SLS.

Theorem 12.4.4. *A first order expansion of the estimation error of 1-D Standard ESPRIT using SLS is given by*

$$\Delta\mu_{k,\mathrm{SLS}} = \mathrm{Im}\left\{\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{12.38}$$

$$= \mathrm{Im}\left\{\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} \cdot \boldsymbol{W}_{\mathrm{mat}} \cdot \mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{12.39}$$

where $\boldsymbol{W}_{\mathrm{mat}}$ *is defined in* (12.27) *and* $\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}}$ *is given by*

$$\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} = \boldsymbol{q}_k^{\mathrm{T}} \otimes \left[\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_s)^+ \cdot \left(\frac{\boldsymbol{J}_2}{\mathrm{e}^{\jmath\mu_k}} - \boldsymbol{J}_1\right)\right] - \left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \left[\boldsymbol{p}_k^{\mathrm{T}} \cdot \frac{(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}}{\mathrm{e}^{\jmath\mu_k}}\right]\right) \cdot \left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}}$$

$$\boldsymbol{W}_{\mathrm{R,U}} = \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) + \boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ \cdot \boldsymbol{J}_2\right) - \boldsymbol{\Psi}^{\mathrm{T}} \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ \cdot \boldsymbol{J}_1\right) - (\boldsymbol{I}_d \otimes \boldsymbol{J}_2)$$

$$\boldsymbol{F}_{\mathrm{SLS}} = \left[\boldsymbol{I}_d \otimes (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}),\ \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) - (\boldsymbol{I}_d \otimes \boldsymbol{J}_2)\right]$$

for $k = 1, 2, \ldots, d$. *The MSE for zero mean, circularly symmetric i.i.d. noise samples with variance* σ_{n}^2 *can then be computed via*

$$\mathbb{E}\left\{\left(\Delta\mu_{k,\mathrm{SLS}}\right)^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\|\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_{k,\mathrm{SLS}}\right\|_2^2. \tag{12.40}$$

Proof: Equation (12.39) is shown in Appendix D.15. Since the explicit expansion of (12.39) has the same form as the explicit expansion in (12.20), the MSE expression (12.40) is shown analogously to Theorem 12.4.1 as presented in Appendix D.12. Note that the first-order expansion and the corresponding MSE expressions in Theorem 12.4.4 were already shown by us in [RH11]. However, due to space limitations, a proof for them was not included there.

12.4.7. Special case: Single source

Even though we have already found comparably compact MSE expressions for different kinds of ESPRIT-type algorithms, they are still too complicated to provide actual insights into the differences of various ESPRIT-type algorithms. As they are deterministic, they can be plotted for varying system parameters without performing Monte-Carlo simulations and one can learn from these plots under which conditions the performance changes how much.

However, it would be desirable to find expressions that are even more insightful. The biggest disadvantage of the MSE expressions in their current form is that they are formulated in terms of the subspaces of the noise-free observation matrix and not in terms of the actual parameters with a physical significance, such as, the number of sensors or the positions of the sources.

Finding such a formulation in the very general case seems to be impossible given the complicated algebraic nature in which the MSE expressions depend on the physical parameters. However, it becomes much easier if some special cases are considered. Therefore we present one example of such a special case in this section, namely, the case of a single source captured by a uniform linear array (ULA) and a uniform rectangular array (URA). Although this is a very trivial case, it serves as an example which types of impact such an analytical performance assessment can provide. The following theorem summarizes our results for the 1-D case (ULA).

Theorem 12.4.5. *For a single source ($d = 1$) and a uniform linear array of M sensors, the mean square estimation error of the spatial frequency for Standard ESPRIT and for Unitary ESPRIT is given by*

$$\mathbb{E}\left\{(\Delta\mu)^2\right\} = \frac{1}{\hat{\rho}} \cdot \frac{1}{(M-1)^2} + \mathcal{O}\left\{\frac{1}{\hat{\rho}^2}\right\} \tag{12.41}$$

Moreover, the deterministic Cramér-Rao Bound can be simplified into

$$\mathrm{CRB} = \frac{1}{\hat{\rho}} \cdot \frac{6}{M \cdot (M^2-1)} \tag{12.42}$$

Consequently, the asymptotic efficiency is given by

$$\eta = \lim_{\hat{\rho}\to\infty} \frac{\mathrm{CRB}}{\mathbb{E}\left\{(\Delta\mu)^2\right\}} = \frac{6(M-1)}{M(M+1)}. \tag{12.43}$$

Here, $\hat{\rho}$ represents the effective SNR given by $\hat{\rho} = \frac{\hat{P}_{\mathrm{T}} \cdot N}{\sigma_{\mathrm{n}}^2}$, where $\hat{P}_{\mathrm{T}}$ is the empirical transmit power given by $\hat{P}_{\mathrm{T}} = \|\boldsymbol{S}\|_{\mathrm{F}}^2 / N$ if $\boldsymbol{S}$ is the matrix of source symbols.

Proof: cf. Appendix D.16.

Note that [RH89a] provide an MSE expression for ESPRIT for the case of a single source which scales with $1/M^2$ and is derived under the assumption of high "array SNR" $P \cdot M/\sigma_{\mathrm{n}}^2$, i.e., it is asymptotic also in M. The result presented here is accurate for small values of M as well and only asymptotic in the effective SNR $N \cdot P_{\mathrm{T}}/\sigma_{\mathrm{n}}^2$. Also note that analytical expression for the *stochastic* Cramér-Rao Bound for one and two sources are available in [Smi05].

We can simplify the MSE expression for ESPRIT using SLS shown in Section 12.4.6 in a similar manner. The following theorem summarizes the result.

Theorem 12.4.6. *For a single source ($d = 1$) and a uniform linear array of M sensors, the mean square estimation error of the spatial frequency for Standard ESPRIT using SLS is given by*

$$\mathbb{E}\left\{(\Delta\mu)^2\right\} = \frac{1}{\hat{\rho}} \cdot 6 \cdot \frac{M^4 - 2M^3 + 24M^2 - 22M + 23}{M(M^2+11)^2(M-1)^2} + \mathcal{O}\left\{\frac{1}{\hat{\rho}^2}\right\} \tag{12.44}$$

Consequently, the asymptotic efficiency is given by

$$\begin{aligned}\eta = \lim_{\hat{\rho}\to\infty} \frac{\mathrm{CRB}}{\mathbb{E}\left\{(\Delta\mu)^2\right\}} &= \frac{(M^2+11)^2(M-1)}{(M+1)(M^4 - 2M^3 + 24M^2 - 22M + 23)} \\ &= \frac{M^5 - M^4 + 22M^3 - 22M^2 + 121M - 121}{M^5 - M^4 + 22M^3 + 2M^2 + M + 23}\end{aligned} \tag{12.45}$$

Proof: cf. Appendix D.17.

Moreover, for the 2-D case we have the following theorem for ESPRIT using LS.

Theorem 12.4.7. [RBH10] *For a single source ($d = 1$) and a uniform rectangular array of $M_1 \times M_2$ sensors, the mean square estimation error of the spatial frequency for 2-D Standard ESPRIT, 2-D Standard Tensor-ESPRIT, 2-D Unitary ESPRIT, and 2-D Unitary Tensor-ESPRIT is given by*

$$\mathbb{E}\left\{(\Delta\mu^{(1)})^2 + (\Delta\mu^{(2)})^2\right\} = \frac{1}{\hat{\rho}} \cdot \left(\frac{1}{(M_1-1)^2 \cdot M_2} + \frac{1}{M_1 \cdot (M_2-1)^2}\right) + \mathcal{O}\left\{\frac{1}{\hat{\rho}^2}\right\} \tag{12.46}$$

Moreover, the deterministic Cramér-Rao Bound can be simplified into

$$\mathrm{CRB} = \mathrm{trace}\left\{\boldsymbol{C}\right\} = \frac{1}{\hat{\rho}} \cdot \left(\frac{6}{M \cdot (M_1^2 - 1)} + \frac{6}{M \cdot (M_2^2-1)}\right). \tag{12.47}$$

Proof: in Appendix D.18 we derive MSE expressions for R-D Standard ESPRIT, R-D Unitary ESPRIT, and the Cramér-Rao Bound for the more general R-D case. From these, this theorem follows by setting $R = 2$. Moreover, we show the identity of R-D Standard Tensor-ESPRIT and R-D Unitary Tensor-ESPRIT with R-D Standard ESPRIT for $R = 2$. Note that the simplified MSE expressions in Theorem 12.4.7 were already shown by us in [RBH10]. However, due to space limitations, a proof could not be included there.

These three theorems provide some very interesting insights. Firstly, they show that for a single source there is neither an improvement in terms of the estimation accuracy from applying Forward-Backward-Averaging nor from the HOSVD-based subspace estimate. This is surprising at first sight since the HOSVD-based subspace estimate itself is more accurate for a single source.

Moreover, they show that the asymptotic efficiency can be explicitly computed and that it is only a function of the array geometry, i.e., the number of sensors in the array. The outcome of this analysis is that ESPRIT-type algorithms using LS are asymptotically efficient for $M = 2, 3$ in the 1-D case and $M_1 \in [2, 3]$, $M_2 \in [2, 3]$ in the 2-D case. However, they become less and less efficient when the number of sensors grows, in fact, for $M \to \infty$ we even have $\eta \to 0$. A possible explanation for this phenomenon could be that an M-sensor ULA offers not only the one shift invariance used in LS (the first and last $M - 1$ sensors) but multiple invariances [SORK92], which are not fully exploited by LS.

On the other hand, for ESPRIT using SLS, the asymptotic efficiency is in general higher, in fact, for a ULA we have $\eta \to 1$ as $M \to \infty$ for a single source. Moreover, even for limited M, η is never far away from 1. As we show in Section 12.5.4, we have $\eta = 1$ for $M = 2, 3$ and the smallest value of η is obtained for $M = 5$ where $\eta = 36/37 \approx 0.973$ for $d = 1$.

12.5. Numerical results

In this section we show numerical results to demonstrate the asymptotic behavior of the analytical performance assessment presented in this chapter. We first investigate the subspace estimation accuracy in order to verify Proposition 12.3.1. Note that the analytical results for the subspace estimates are explicit expansions in terms of the perturbation (i.e., the additive noise). Therefore, we repeat the experiment with a number of randomly generated realizations of the noise and perform Monte-Carlo averaging over the analytical expansions. These "**semi-analytical**" results are then compared with purely **empirical** results where we estimate the subspace via an SVD or a HOSVD and compute the estimation error compared to the true signal subspace.

The subsequent numerical results demonstrate the performance of ESPRIT-type parameter estimation schemes. Here, we compute the mean square estimation error in three different ways. Firstly, **analytically**, via the MSE expressions provided in Theorem 12.4.1, Theorem 12.4.3, Theorem 12.4.4, Theorem 12.4.5, Theorem 12.4.6, and Theorem 12.4.7, repectively. Secondly, **semi-analytically**, by performing Monte-Carlo averaging over the explicit first-order expansions provided in equation (12.23), (12.32), and (12.39), respectively. Thirdly, **empirically**, by estimating the spatial frequencies via the corresponding ESPRIT-type algorithms and comparing the estimates to the true spatial frequencies.

For all the simulations we assume a known number of planar wavefronts impinging on an antenna array of M istrotropic sensor elements. We assume uniform $\lambda/2$ spacing in all dimensions, i.e., an M-element uniform linear array (ULA) in the 1-D case and an $M_1 \times M_2$ uniform rectangular array (URA) in the 2-D case. The sources emit narrow-band waveforms $s_i(t)$ modeled as complex Gaussian distributed symbols $s_i[n]$ and we observe N subsequent snapshots $n = 1, 2, \ldots, N$. All sources are assumed to have unit power, i.e., $\mathbb{E}\left\{|s_i[n]|^2\right\} = 1$. In the case where source correlation is investigated we generate the symbols $s_i[n]$ such that $\mathbb{E}\left\{s_i[n] \cdot s_j[n]^*\right\} = \rho \cdot \mathrm{e}^{\jmath\varphi_{i,j}}$ for $i \neq j = 1, 2, \ldots, d$, where ρ is the correlation coefficient between each pair of sources and $\varphi_{i,j}$ is a uniformly distributed correlation phase. The additive noise is generated according to a circularly symmetric complex Gaussian distribution with zero mean and variance σ_{n}^2 and noise samples are assumed to be mutually independent. Therefore, the Signal to Noise Ratio (SNR) is defined as $1/\sigma_{\mathrm{n}}^2$.

12.5.1. Subspace estimation accuracy

We evaluate the subspace estimation accuarcy by computing the Frobenius norm of the subspace estimation error, i.e., $\|\Delta \boldsymbol{U}_{\mathrm{s}}\|_{\mathrm{F}}^2$ in the matrix case and $\left\|\left[\Delta \boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right\|_{\mathrm{H}}^2$ in the tensor case.

In order to find the estimation error empirically, we obtain a subspace estimate $\hat{\boldsymbol{U}}_{\mathrm{s}}$ via an SVD of the noisy observation and then compare it to the true subspace $\boldsymbol{U}_{\mathrm{s}}$ column by column. The estimation error of the n-th column is computed via

$$\Delta \boldsymbol{u}_n = \hat{\boldsymbol{u}}_n \cdot \frac{\hat{\boldsymbol{u}}_n^{\mathrm{H}} \boldsymbol{u}_n}{\left|\hat{\boldsymbol{u}}_n^{\mathrm{H}} \boldsymbol{u}_n\right|} - \boldsymbol{u}_n, \quad n = 1, 2, \ldots, d \tag{12.48}$$

to account for the inherent phase ambiguity in each column of the SVD, cf. [LLM08].

For the analytical estimation error we calculate $\Delta \boldsymbol{U}_{\mathrm{s}}$ via the first-order expansion $\Delta \boldsymbol{U}_{\mathrm{s}} \approx \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{\Gamma}^{[\mathrm{n}]}$ provided in (12.4) and the expansion $\Delta \boldsymbol{U}_{\mathrm{s}} \approx \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{\Gamma}^{[\mathrm{n}]} + \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Gamma}^{[\mathrm{s}]}$ provided in (12.5), respectively. Note that the latter is more accurate since it additionally considers the perturbation of the individual singular vectors, i.e., the particular choice of the basis for the signal subspace. However, this contribution is irrelevant for the performance of ESPRIT-type algorithms.

Figure 12.1 shows the results for a scenario where we consider two uncorrelated and closely spaced sources. Their spatial frequencies are given by $\mu_1^{(1)} = \mu_1^{(2)} = 1$ and $\mu_2^{(1)} = \mu_2^{(2)} = 0.95$. A 5×5 URA is considered and $N = 20$ snapshots are taken. In Figure 12.2 we have $d = 3$ sources positioned at $\mu_1^{(1)} = 0.7, \mu_2^{(1)} = 0.9, \mu_3^{(1)} = 1.1, \mu_1^{(2)} = -0.1, \mu_2^{(2)} = -0.3, \mu_3^{(2)} = -0.5$ and mutually correlated with a correlation coefficient of $\rho = 0.97$. Moreover, the array size is increased to an 8×8 URA. For the third simulation result shown in Figure 12.3 we consider $d = 4$ uncorrelated sources located at $\mu_1^{(1)} = -1.5$, $\mu_2^{(1)} = 0.5$, $\mu_3^{(1)} = 1.0$, $\mu_4^{(1)} = -0.3$, $\mu_1^{(2)} = 1.3$, $\mu_2^{(2)} = -0.2$,

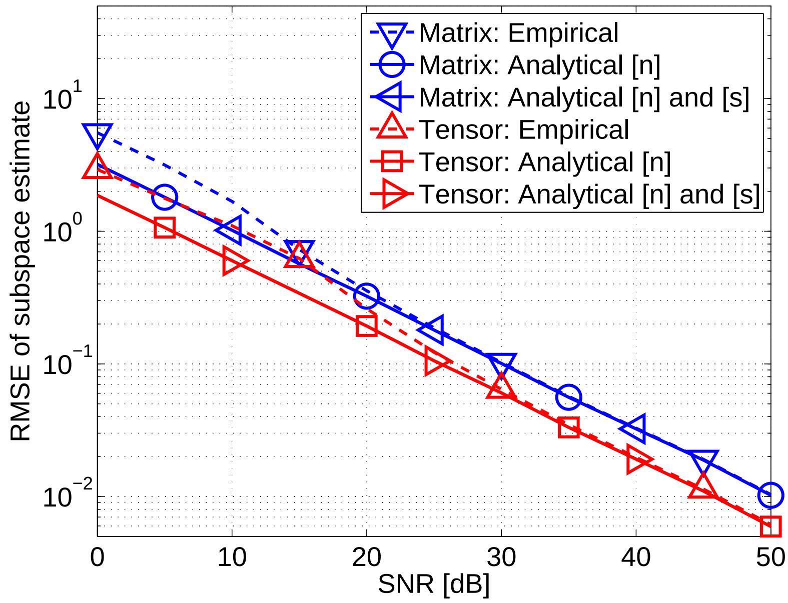


Figure 12.1.: Subspace estimation accuracy using $\mathbf{\Gamma}^{[\mathrm{n}]}$ only vs. using $\mathbf{\Gamma}^{[\mathrm{n}]}$ and $\mathbf{\Gamma}^{[\mathrm{s}]}$. Scenario: two closely spaced sources ($\mu_1^{(1)} = \mu_1^{(2)} = 1$ and $\mu_2^{(1)} = \mu_2^{(2)} = 0.95$), a 5×5 URA, and $N = 20$ snapshots.

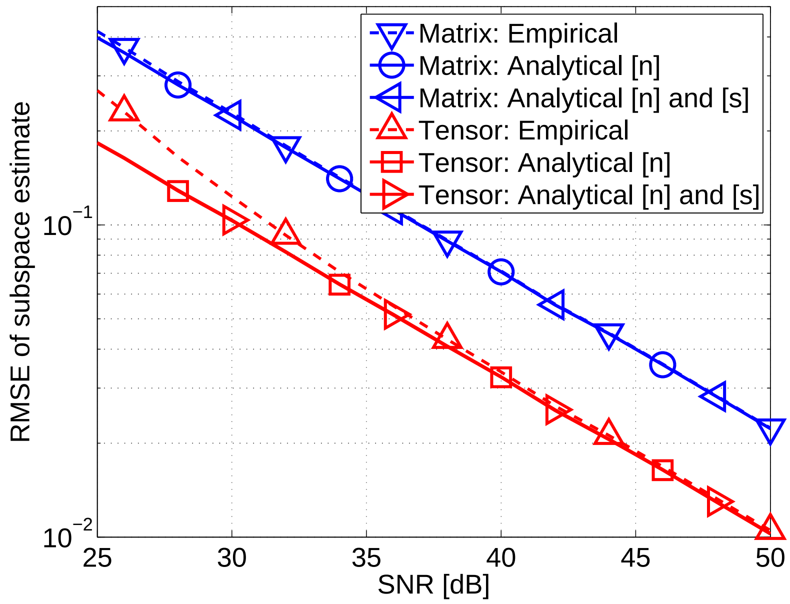


Figure 12.2.: Subspace estimation accuracy using $\mathbf{\Gamma}^{[\mathrm{n}]}$ only vs. using $\mathbf{\Gamma}^{[\mathrm{n}]}$ and $\mathbf{\Gamma}^{[\mathrm{s}]}$. Scenario: $d = 3$ correlated sources ($\rho = 0.97$) at $\mu_1^{(1)} = 0.7, \mu_2^{(1)} = 0.9, \mu_3^{(1)} = 1.1, \mu_1^{(2)} = -0.1, \mu_2^{(2)} = -0.3, \mu_3^{(2)} = -0.5$, an 8×8 URA, and $N = 20$ snapshots.

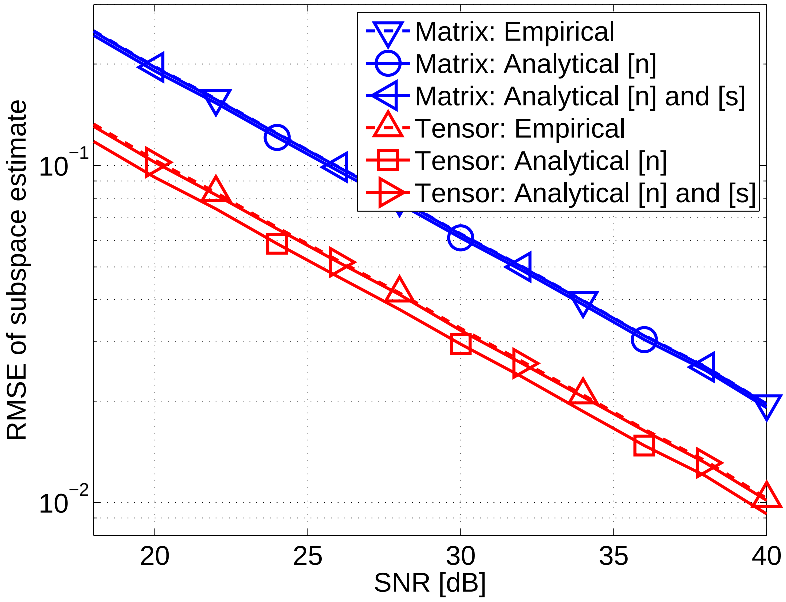


Figure 12.3.: Subspace estimation accuracy using $\boldsymbol{\Gamma}^{[\mathrm{n}]}$ only vs. using $\boldsymbol{\Gamma}^{[\mathrm{n}]}$ and $\boldsymbol{\Gamma}^{[\mathrm{s}]}$. Scenario: $d = 4$ uncorrelated sources at $\mu_1^{(1)} = -1.5$, $\mu_2^{(1)} = 0.5$, $\mu_3^{(1)} = 1.0$, $\mu_4^{(1)} = -0.3$, $\mu_1^{(2)} = 1.3$, $\mu_2^{(2)} = -0.2$, $\mu_3^{(2)} = 0.7$, $\mu_4^{(2)} = -1.5$, an 8×8 URA, and $N = 5$ snapshots.

$\mu_3^{(2)} = 0.7$, $\mu_4^{(2)} = -1.5$ and $N = 5$ snapshots.

All three simulations show that the empirical estimation errors agree with the analytical results as the SNR tends to infinity. Therefore, the improvement obtained by the HOSVD-based subspace estimate can be reliably predicted via the analytical expressions. In general, it is particularly pronounced for correlated sources and for a small number of snapshots. Moreover, while for two closely-spaced sources and for three correlated sources, the impact of the additional term $\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Gamma}^{[\mathrm{s}]}$ is negligibly small, it is clearly visible for the four uncorrelated sources shown in Figure 12.3.

12.5.2. R-D Tensor-ESPRIT

The following set of simulation results demonstrates the performance of R-D matrix-based and tensor-based ESPRIT. As explained in the beginning of this section, for the analytical results we use Theorem 12.4.1 for R-D standard ESPRIT and R-D standard Tensor-ESPRIT and Theorem 12.4.3 for R-D Unitary ESPRIT and R-D Unitary Tensor-ESPRIT, respectively. Likewise, the semi-analytical results are obtained by Monte-Carlo averaging of the explicit first-order expansion provided in (12.23) and (12.32), respectively.

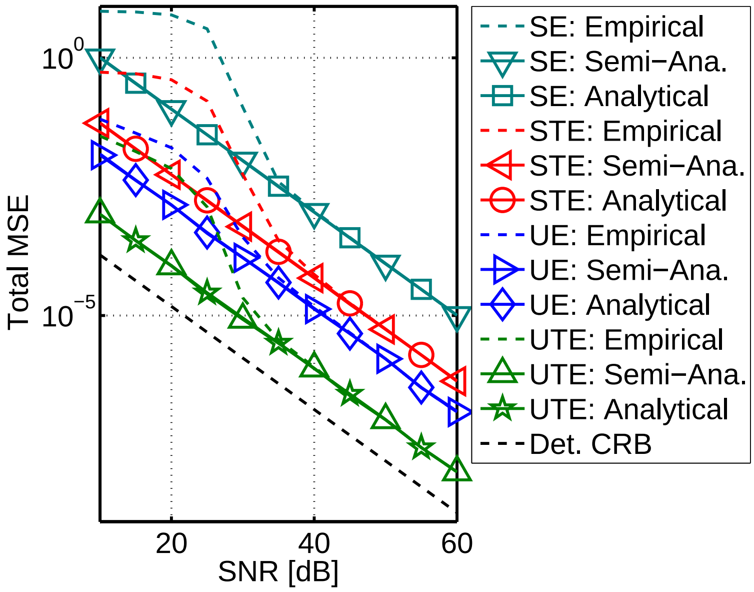


Figure 12.4.: Performance of 2-D SE, STE, UE, UTE for $d = 2$ highly correlated sources ($\rho = 0.9999$) located at $\mu_1^{(1)} = 1$, $\mu_2^{(1)} = -0.5$, $\mu_1^{(2)} = -0.5$, and $\mu_2^{(2)} = 1$, a 5×6 URA, and $N = 20$ snapshots.

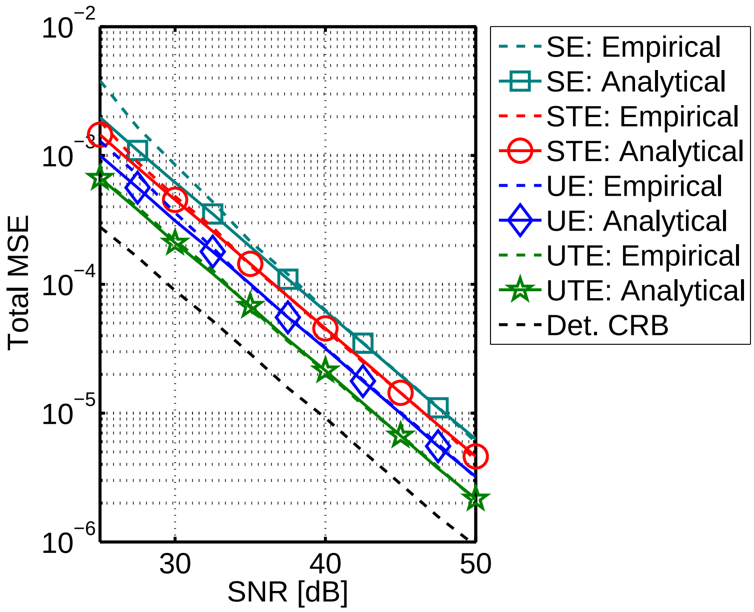


Figure 12.5.: Performance of 2-D SE, STE, UE, UTE for $d = 3$ correlated sources ($\rho = 0.97$) positioned at $\mu_1^{(1)} = 0.7, \mu_2^{(1)} = 0.9, \mu_3^{(1)} = 1.1, \mu_1^{(2)} = -0.1, \mu_2^{(2)} = -0.3, \mu_3^{(2)} = -0.5$, an 8×8 URA, and $N = 20$ snapshots.

For Figure 12.4 we employ a 5×6 URA and collect $N = 20$ snapshots from two sources located at $\mu_1^{(1)} = 1$, $\mu_2^{(1)} = -0.5$, $\mu_1^{(2)} = -0.5$, and $\mu_2^{(2)} = 1$. The sources are highly correlated with a correlation of $\rho = 0.9999$. On the other hand, for Figure 12.5 we incease the number of sources to $d = 3$ and the correlation coefficient to $\rho = 0.97$. Moreover, the spatial frequencies of the sources are given by $\mu_1^{(1)} = 0.7, \mu_2^{(1)} = 0.9, \mu_3^{(1)} = 1.1, \mu_1^{(2)} = -0.1, \mu_2^{(2)} = -0.3, \mu_3^{(2)} = -0.5$ and we use an 8×8 URA.

To enhance the legibility, we show the semi-analytical estimation errors only in Figure 12.4 since they always agree with the analytical results, as expected. Moreover, the empirical estimation errors agree with the analytical ones for high SNRs. This is also expected as the performance analysis framework presented here is asymptotically accurate for high effective SNRs. We conclude that the improvement in terms of estimation accuracy for Tensor-ESPRIT-type parameter estimation schemes can be reliably predicted via the analytical expressions we have derived.

12.5.3. Structured Least Squares

The next set of simulation results illustrates the analytical expressions for ESPRIT using SLS. The semi-analytical MSE is obtained by Monte-Carlo averaging over the explicit expansion provided in (12.39) and the analytical MSE is computed via Theorem 12.4.4. For the empirical estimation errors we perform a single iteration of the Structured Least Squares algorithm and do not use regularization (i.e., the regularization parameter α is set to ∞).

The first simulation result is shown in Figure 12.6. Here we consider $N = 3$ snapshots from $d = 4$ uncorrelated sources captured by an $M = 8$ element uniform linear array. The sources' spatial frequencies are given by $\mu_1 = 1.0, \mu_2 = 0.7, \mu_3 = -0.6, \mu_4 = -0.3$. Note that since $N < d$, we cannot apply standard ESPRIT, therefore, only Unitary ESPRIT is used. On the other hand, in the second scenario we consider $N = 10$ snapshots from $d = 3$ sources that are mutually correlated with a correlation coefficient of $\rho = 0.99$. The sources are located at $\mu_1 = 1$, $\mu_2 = 0$, $\mu_3 = -1$ and a $M = 12$ element ULA is used. The corresponding estimation errors are shown in Figure 12.7.

As before, the empirical results agree with the analytical results for high SNRs. Moreover, the improvement in MSE obtained via SLS is particularly pronounced for the correlated sources. However, even the very slight improvement which is present for four uncorrelated sources can be reliably predicted via the analytical MSE expressions we have derived.

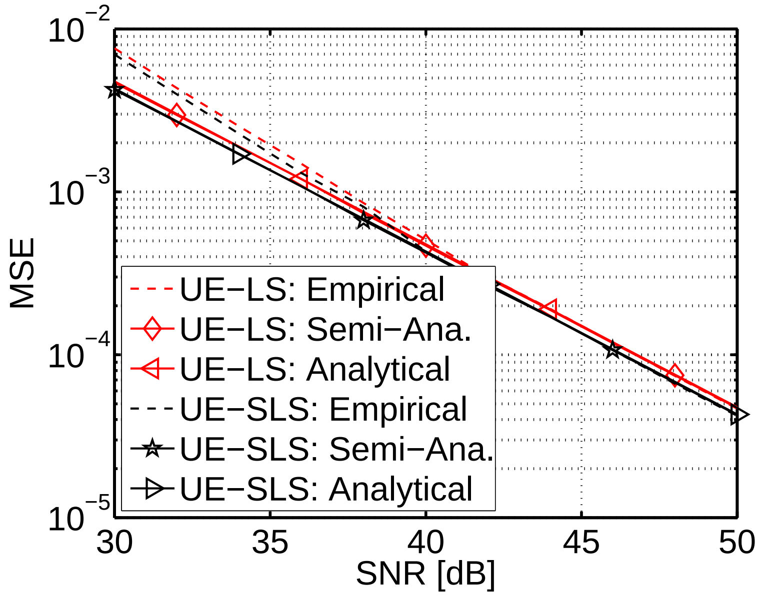


Figure 12.6.: Performance of LS-ESPRIT vs. SLS-ESPRIT for 4 sources at $\mu_1 = 1.0, \mu_2 = 0.7, \mu_3 = -0.6, \mu_4 = -0.3$, an $M = 8$ ULA, $N = 3$ shapshots.

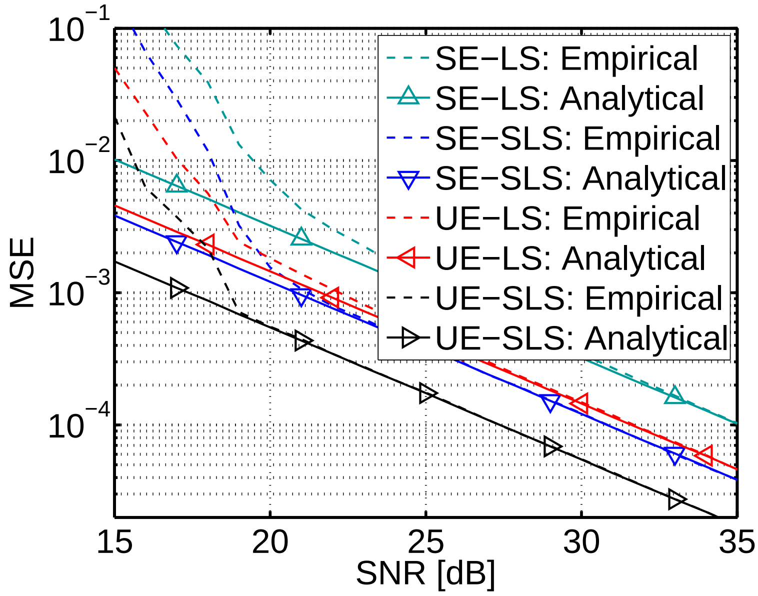


Figure 12.7.: Performance of LS-ESPRIT vs. SLS-ESPRIT for $d = 3$ correlated sources ($\rho = 0.99$) at $\mu_1 = 1$, $\mu_2 = 0$, $\mu_3 = -1$, a $M = 12$ ULA and $N = 10$ shapshots.

12.5.4. Asymptotic efficiency for a single source

The final set of simulation results demonstrates the special case of a single source, in which case the MSE expressions can be simplified to very compact closed-form expressions which only depend on the physical parameters, i.e., the array size and the SNR.

Figures 12.8 and 12.9 show the MSE vs. the number of sensors M for a (1-D) Uniform Linear Array and vs. M_1 for a (2-D) $M_1 \times M_1$ Uniform Rectangular Array, respectively. For both scenarios, the spatial frequencies of the single source were drawn randomly (note that they have no impact on the MSE). The effective SNR was set to 25 dB for Figure 12.8 ($P_{\rm T} = 1, \sigma_{\rm n}^2 = 0.032, N = 10$) and to 46 dB for Figure 12.9 ($P_{\rm T} = 1, \sigma_{\rm n}^2 = 10^{-4}, N = 4$), respectively.

For both plots we observe that LS-ESPRIT is asymptotically efficient for $M = 2$ and $M = 3$ (which, in the 2-D case means, a 3×3 URA) and then becomes increasingly inefficient as the array size grows. Moreover, for the 1-D case we see that SLS-based ESPRIT is in fact very close to the Cramér-Rao Bound, which may, at first sight, lead to believe that the asymptotic efficiency is in fact 1 for all M. However, as we have shown it is in fact slightly lower than one. Therefore, we provide two additional figures where we depict the "asymptotic efficiency", i.e., we divide the CRB by the corresponding value of the MSE. The resulting efficiency plot is shown on the left-hand side of Figure 12.10. Moreover, a zoom on the same figure is provided on the right-hand side. This plot shows more clearly that LS-ESPRIT becomes increasingly inefficient for $M > 3$, whereas SLS-ESPRIT approaches $\eta = 1$ for large M. The worst efficiency is found for $M = 5$ where we have $\eta = 36/37 \approx 0.973$.

12.6. Summary

In this chapter we have discussed a framework for analytical performance assessment of subspace-based parameter estimation schemes. It is based on earlier results on an explicit first-order expansion of the SVD and its application to 1-D versions of subspace-based parameter estimation schemes, e.g., ESPRIT. We have extended this framework in a number of ways. Firstly, we have derived an explicit first-order expansion of the HOSVD-based subspace estimate which is the basis for Tensor-ESPRIT-type algorithms. However, it can be applied to enhance other subspace-based parameter estimation schemes as well. As an important side-result we have established a fundamental link between the SVD-based subspace estimate and the HOSVD-based subspace estimate via a Kronecker-structured projection matrix. In addition to its insightful interpretation, this link can be used for other purposes, e.g., tensor-based subspace tracking.

Secondly, we have shown that the first-order expansion for 1-D Standard ESPRIT can be

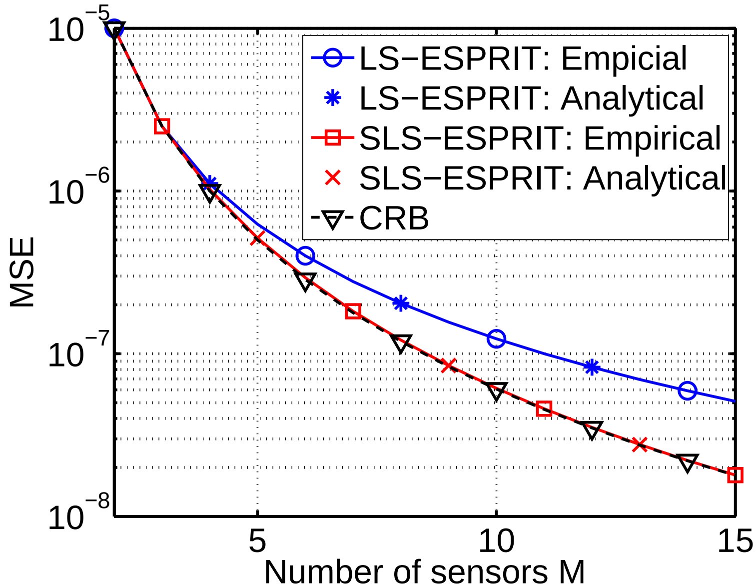


Figure 12.8.: Performance of LS-ESPRIT and SLS-ESPRIT for a single source vs. the number of sensors M (M-ULA) at an effective SNR of 25 dB ($P_{\mathrm{T}} = 1, \sigma_{\mathrm{n}}^2 = 0.032, N = 10$).

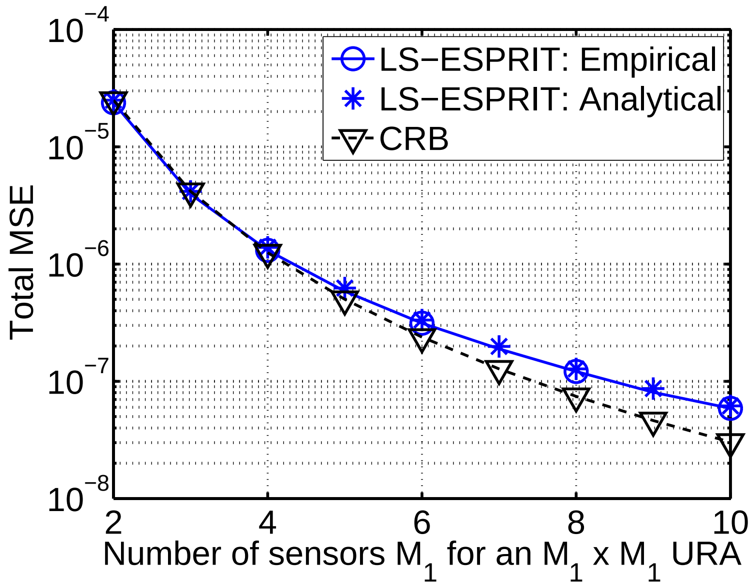


Figure 12.9.: Performance of LS-ESPRIT for a single source vs. M_1 using an $M_1 \times M_1$ URA at an effective SNR of 46 dB ($P_{\mathrm{T}} = 1, \sigma_{\mathrm{n}}^2 = 10^{-4}, N = 4$).

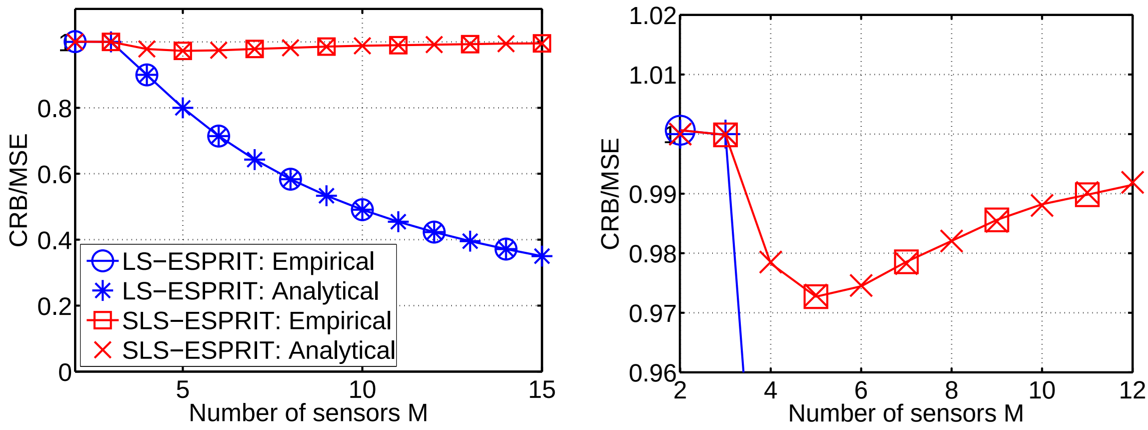


Figure 12.10.: Asymptotic efficiency of LS-ESPRIT and SLS-ESPRIT vs. M. Same scenario as in Figure 12.8. The right-hand side shows a zoom on the region where η is close to one.

extended to other ESPRIT-type algorithms, e.g., R-D Standard ESPRIT, R-D Unitary ESPRIT, R-D Standard Tensor-ESPRIT, R-D Unitary Tensor-ESPRIT, NC Standard ESPRIT, or NC Unitary ESPRIT.

Thirdly, we have derived a corresponding first-order expansion for Structured Least Squared (SLS)-based ESPRIT-type algorithms.

All these expansions have in common that they are explicit, i.e., no assumption about the statistics of either desired signal or additive perturbation need to be made. We only require the perturbation to be small compared to the desired signal.

Our fourth contribution is to show that the mean square error can readily be computed in closed-form if we assume zero mean circularly symmetric white noise. Note that we do not need the noise to be Gaussian distributed. The corresponding MSE expressions are asymptotic in the effective SNR, i.e., they become accurate as either the noise variance goes to zero or the number of observations goes to infinity. Therefore, in contrast to existing results based on [Bri75] which require N to be large, they even apply to the single snapshot ($N = 1$) case.

As a final contribution we have investigated the special case of a single source and uniform (1-D) or rectangular (2-D) arrays. In this case we have been able to show analytically, that R-D standard ESPRIT, R-D Unitary ESPRIT, and (for $R = 2$) R-D standard Tensor-ESPRIT as well as R-D Unitary Tensor-ESPRIT yield the same MSE, which only depends on the effective SNR and the number of antenna elements. We have also shown that SLS-based 1-D standard ESPRIT has a lower MSE which is also expressed explicitly as a function of the effective SNR and the number of antenna elements. Concerning the asymptotic efficiency, this study has shown that LS-based ESPRIT is efficient only for two and three sensor elements and then becomes increasingly inefficient as the number of antennas grows. In contrast to this, SLS-based ESPRIT yields an asymptotic efficiency very close to one for all array sizes.

13. Summary of subspace-based parameter estimation

13.1. Summary of contributions

In this part of the thesis, multi-dimensional subspace-based parameter estimation was discussed. We have shown how subspace-based parameter estimation schemes can be improved by exploiting the specific structure of the signal of interest. We have chosen the family of ESPRIT-type algorithms to demonstrate the enhancements, however, they can be applied to many different subspace-based parameter estimation schemes as well. A summary of the various ESPRIT-type methods is shown in Table 13.1. The column "proposed" shows the reference where the algorithm was originally proposed, the column "performance analysis" lists the reference where an analytical performance assessment was first discussed. Here "(open)" means that up to now no performance analysis has been published. Table 13.2 provides the same overview for the various approaches to solve the overdetermined shift invariance equations of ESPRIT-type algorithms.

Based on these tables it is easy to identify the main contributions as well as the open aspects. The main contributions are:

- The HOSVD-based subspace estimate ([RHD06], [HRD08]), which provides an enhanced version of the signal subspace by exploiting the multilinear structure of the data via a more efficient denoising. It can be applied to enhance arbitrary existing multidimensional subspace-based parameter estimation schemes.

Algorithm	Proposed	Performance analysis based on	
		[Bri75]	[LLV93]
1-D Standard ESPRIT	[RPK86]	[RH89b]	[LLV93], ...
1-D Unitary ESPRIT	[HN95]	[MHZ96]	[RBH10]
R-D Unitary ESPRIT	[HN98]	[MHZ96] (2-D)	[RBH10]
R-D Standard/Unitary Tensor-ESPRIT	[HRD08]		[RBHW09, RBH10]
1-D NC Standard ESPRIT	[ZCW03]	= 1-D NC Unitary ESPRIT, cf. Sec. 11.5.2	
R-D NC Unitary ESPRIT	[HR04]	(open)	
R-D NC Standard/Unitary Tensor-ESPRIT	[RH09b]	(open)	

Table 13.1.: Overview of ESPRIT-type algorithms and their performance analysis

Algorithm	Proposed	Performance analysis based on	
		[Bri75]	[LLV93]
Least Squares (LS)	[RPK86]	[RH89b]	[LLV93]
Total Least Squares (TLS)	[RK87]	[RH89b, OVK91]	
1-D Structured Least Squares (SLS)	[Haa97b]		[RH11]
R-D Structured Least Squares (SLS)	[Haa97b]	(open)	
Tensor-Structure SLS	[RH07b]	(open)	

Table 13.2.: Overview of Least-Squares algorithms to solve the invariance equations of ESPRIT-type algorithms and their performance analysis

- The link between the HOSVD-based subspace estimate and the SVD-based subspace estimate ([RBHW09]) which is introduced in Theorem 10.2.1 which provides a solid analytical explanation why and under which conditions the HOSVD-based subspace estimate yields an enhanced signal subspace estimate. Moreover, it shows that the core tensor in the HOSVD is not needed which greatly simplifies the performance analysis.
- The R-D Tensor-ESPRIT-type algorithms R-D Standard Tensor-ESPRIT and R-D Unitary Tensor-ESPRIT ([HRD08]) which are summarized in Section 11.4. The tensor formulation of the algorithms is more natural and hence simpler than in the matrix case. It leads to similar results as for matrix-based ESPRIT except for the fact that the SVD-based subspace estimate is replaced by the improved HOSVD-based subspace estimate (cf. Theorem 11.4.2). Since the latter is easily computed via the structured projection shown in Theorem 10.2.1, existing implementations of R-D ESPRIT-type algorithms can be upgraded without large effort.
- The Tensor-Structure SLS algorithm [RH07b] which is shown in Section 11.7. It allows to benefit from the tensor structure even further in the step where the invariance equations are solved. TS-SLS can be combined with arbitrary R-D ESPRIT-type algorithms and provides an additional improvement in terms of the estimation accuracy, even in the cases where the HOSVD-based subspace estimate coincides with the SVD-based subspace estimate, i.e., if $d \geq \max_{r=1,2,\ldots,R}(M_r)$.
- The R-D ESPRIT-algorithms for strict-sense non-circular sources R-D NC Standard ESPRIT and R-D NC Unitary ESPRIT which are discussed in Section 11.5. As an interesting observation we have seen that R-D NC Standard ESPRIT is not needed since the preprocessing creates a virtual array steering matrix which is always centro-symmetric so that R-D NC Unitary ESPRIT is applicable even if the original array is

not centro-symmetric. We have also shown that R-D NC Standard ESPRIT and R-D NC Unitary ESPRIT have the same asymptotical peformance (cf. (11.31)).

- The R-D Tensor-ESPRIT-type algorithms for strict-sense non-circular sources which are discussed in Section 11.6. The main idea was finding a tensor-compliant way of exploiting strict-sense non-circularity which was found by a sequence of r-mode augmentations of the measurement tensor.
- The first-order analytical performance assessment of the HOSVD, based on first-order perturbation results of the SVD.
- The application of these results to find an explicit first-order perturbation expansion for R-D ESPRIT-type algorithms based on Least Squares, in particular R-D Standard ESPRIT, R-D Unitary ESPRIT, R-D Standard-Tensor-ESPRIT, and R-D Unitary Tensor-ESPRIT. The strength of the explicit expansion is that it does not require any assumptions about the statistics of the perturbation (i.e., the additive noise).
- The extension of this result to ESPRIT based on Structured Least Squares (SLS), for simplicity only considering 1-D SLS and a single SLS iteration.
- The derivation of closed-form MSE expressions for all previously mentioned algorithms for the case of circularly symmetric white noise. Note that Gaussianity is not needed here.
- The simplification of the MSE expression in terms of parameters with a physical significance such as the number of antenna elements, or the SNR, for the special case of a single source. This led to the conclusion that all ESPRIT-type algorithms based on LS perform identical[1] and that their asymptotical efficiency decreases with an increasing number of sensor elements. Moreover, SLS-based ESPRIT was shown to have an asymptotic efficiency significantly better than LS-based ESPRIT, in fact, very close to one.

13.2. Future work

The HOSVD-based subspace estimate is a convenient and algebraically simple way to benefit from the multidimensional structure in R-D parameter estimation problems. While its

[1] We have not drawn this conclusion for NC ESPRIT-type algorithms here since no performance analysis is available yet. However, numerical results suggest that this is indeed the case for NC ESPRIT-type algorithms as well.

combination with ESPRIT-type algorithms was successfully demonstrated, it opens up many options for future research.

Firstly, its combination with different subspace-based algorithms is still to be investigated. For instance, the R-D RARE or the R-D MODE algorithm are expected to benefit from the enhanced subspace as well. In fact, as for ESPRIT, a tensor-valued formulation of the algorithms could be beneficial in order to find further algorithm-specific improvements using tensors in addition to the enhanced subspace estimate (like the TS-SLS algorithm [RH07b] for Tensor-ESPRIT-type algorithms).

Secondly, the explicit link to the matrix-based subspace estimate via the structured projection shown in Theorem 11.4.2 opens up exciting new possibilities. It shows that the HOSVD-based subspace estimate can be computed based on the subspaces of all the unfoldings of the tensor. Consequently, matrix-based subspace tracking schemes can be applied to all unfoldings in order to adaptively track the HOSVD-based subspaces in time-varying scenarios. Initial results on this idea suggest that this is a promising research direction.

Concerning the exploitation of the non-circularity of the source signals, further extensions are possible as well. For instance, it would be interesting to exploit the coexistence of circular and non-circular sources. Also, it would be desirable to find a procedure by which the less restrictive weak-sense non-circularity can be used in a beneficial way.

In terms of the analytical performance assessment, the open aspects can be categorized into four directions. Firstly, some of the proposed methods have not yet been described analytically. For R-D NC Unitary ESPRIT and R-D NC Unitary Tensor-ESPRIT this is expected to be straightforward (cf. Section 12.4.5). Also, incorporating spatial smoothing and the tensor-based spatial smoothing (TBSS) technique from [THRG10, THG09b, THG09a] seems to be an easy task. They could lead to a better understanding of the impact of the design parameters (i.e., the number of subarrays in the separate dimensions) in spatial smoothing and prove that the methods of [THRG10, THG09a] are insensitive to this parameter. For the TS-SLS-based Tensor-ESPRIT-type algorithms it is not as simple since one iteration may not be sufficient due to the many cross terms that are occur in the TS-SLS cost function [RH07b]. This complicates the derivation of a first-order perturbation expansion since an iterative expansion of the pseudo-inverse involved in the update steps of TS-SLS becomes necessary.

Secondly, the analysis of the single source case has revealed some interesting aspects. However, there is no gain from using tensors in parameter estimation accuracy (although the signal subspace estimate has been improved). This gain is present as soon as more than one source is estimated. Therefore, extending this study to the case of two sources and computing the estimation error as an explicit function of the source correlation or the source separation promises to be even more insightful when the tensor gain is particularly pronounced. However, it is expected that the necessary calculations for this become quite tedious as many of the simplifications for the single source case cannot be applied anymore. Even if simplified MSE expressions cannot be found we could analyze the performance based on the MSE expressions we have. Performing such an analysis under various conditions in a systematic manner may reveal some general conclusions in which cases the tensor gain is large or small.

A third direction is to investigate other types of R-D subspace-based parameter estimation schemes such as R-D MUSIC, R-D MODE or R-D RARE and investigate the tensor gain and the asymptotic efficiency for these cases. Note that this extension uses the HOSVD-based subspace estimate and hence differs from existing tensor-based R-D estimation schemes (such as [MLM05] which operators on eigentensors of the covariance tensor of the vector-sensor array observations, as explained in Section 9.1).

Finally, a fourth direction is to extend the first-order perturbation analysis which is only valid for high effective SNRs to a second-order expansion, which then also captures the threshold effect for lower SNRs. The second-order expansion of the SVD is already available [Xu02] and could serve as a starting point. Note that this is also of particular interest with respect to the tensor gain since from numerical simulations it seems that even for cases where there is no tensor gain for high SNRs (e.g., a single source), there can still be a gain for low SNRs.

Part IV.

Two-Way Relaying

The following chapters of this thesis are devoted to applications of advanced algebraic concepts in the area of two-way relaying. We begin in Chapter 14 by motivating why relaying is an increasingly important aspect of wireless communication systems and why efficient use of relays may play a key role to cope with the challenges we face in the design of next-generation mobile networks. We also elaborate on various relaying protocols and relay operation modes to motivate why we study one particular relaying scheme, namely, two-way relaying with (digital) amplify and forward (AF) relays.

We then discuss two crucial aspects of two-way relaying systems with MIMO AF relays. The first aspect discussed in Chapter 15 is the acquisition of channel state information which is crucial in order to cancel the self-interference at the terminals which is caused by the two-way relaying protocol. Since no suitable channel estimation scheme has existed, we propose a novel tensor-based algebraic channel estimation (TENCE) procedure which takes advantage of the specific structure of the channel matrices in this system. To enhance the estimation accuracy further, a Structured Least Squares based iterative refinement for TENCE is introduced.

The effectiveness of the channel estimation procedures is demonstrated in numerical simulations.

The subsequent Chapter 16 is devoted to the design of the relay amplification matrix. As we discuss, existing approaches either involve very complicated numerical optimization procedures or they are based on ad-hoc choices which are not motivated by an appropriate system design goal and hence demonstrate an unsatisfactory system performance. Therefore, we introduce the algebraic norm maximizing (ANOMAX) transmit strategy which achieves a very good bit error rate performance while being very simple to compute via a dominant eigenvector. We also introduce an algebraic modification of ANOMAX for rate maximization which adapts the singular value profile of the ANOMAX solution as a function of the SNR in order to achieve the full spatial multiplexing gain for high SNRs. Finally, we introduce the semi-algebraic rate maximization via generalized eigenvectors for single-antenna terminals (RAGES) which provides the optimal sum-rate in a two-way relaying system for the special case that the terminals have a single antenna.

A summary of our achievements and an outlook to possible future work is finally provided in Chapter 17. Finally, some proofs and derivations have been moved to Appendix E to enhance the readability.

14. Introduction to two-way relaying

14.1. Motivation and state of the art

Communication among individuals is without much doubt of the most fundamental endeavors of mankind. Throughout history our society has constantly looked for ways to improve our abilities to exchange messages over longer distances. In many ways, these achievements had major impacts on society itself. Nowadays it is considered most natural to have conversations with individuals on different continents, several thousand miles away. This ability allows us to share knowledge, beliefs, and emotions on a global scale which would have been inconceivable a few decades ago.

It is this ever-increasing deep desire for faster, more reliable, and completely ubiquitous access to communication that has created more and more sophisticated global communication networks. In the past decades society has witnessed the creation of the first analog wireless transmission systems with the first handheld devices available in the 70's and 80's, the transition to digital wireless telephony in the 90's, and new improved generations (3G and 4G) in the last decade. With every new standard created we have found increased data rates, enhanced coverage, new services, and more sophisticated devices. These improvements were achieved by exploiting novel technologies to push use of the available scarce resources (spectrum, power) closer to the known limits.

With enhanced digital modulation schemes as well as flexible spectrum access and reuse methods we have reached a point where the data rate per link is already very close to the ultimate limit predicted by Shannon's channel coding theorem [Sha48]. A further improvement in data rate was achieved with using multiple antennas at both link ends. Multiple Input Multiple Output (MIMO) technologies enable us to serve multiple users on the same bandwidth simultaneously via Space-Division Multiple Access (SDMA). Many efficient Multi-User MIMO transmit strategies have been proposed, e.g., the Block Diagonalization (BD) algorithm [SSH04] or its regularized version RBD [SH08]. These are often close to the optimal strategies given by non-linear precoding schemes [Cos83, CS03]. Note that MIMO technologies are already a part of the 4G standards [WIN08] that are now in the process of being implemented. As with the capacity of a single link, the capacity of MIMO links [Tel99] is already closely within reach.

We are standing at a point where we have to ask ourselves what the next fundamental aspect will be to improve communication systems further. A key indicator for the answer to this question is that while the achievable spectral efficiency of a single source-destination link (Point to point (P2P)) has almost been reached, in terms of other measures there is still a far way to go. If we consider the normalized data rate per area (say, Bits per second per Hertz per square kilometer) or the energy efficiency (say, Bits per second per Hertz per Joule), the limits are either unknown or far from being reached [DHL+11]. Therefore, in addition to making our communication systems more energy efficient, one of the prime challenges will be to improve *coverage* in an information theoretical sense. That means it will not be enough (albeit required) to make cells ever so small and increase network node density further and further. Additionally, we need to use the available area more efficiently. This leads to an inevitable and very essential change in how we treat interference.

Ever since the first cellular systems were established, interference was regarded as a purely negative aspect which needs to be avoided by orthogonal access schemes. This means that neighboring cells would use different frequency bands to avoid interfering each other and multiple users would be assigned to different frequency bins (Frequency Division Multiple Access (FDMA)), time slots (Time Division Multiple Access (TDMA)), and/or they would be spatially separated (SDMA). This leads to a hierarchical decomposition of the network (network $\rightarrow$ cell $\rightarrow$ link) into the basic building block P2P link between a transmitter and a receiver. While this approach simplifies the design significantly, we have now reached a point where it may simply not be sufficient any more to serve the growing traffic demands with the available limited resources.

Instead of regarding interference as a purely negative aspect we have started to understand that there are cases where it can be used much more flexibly, sometimes even in a constructive manner. A prominent example for such a technology is inter-cell interference coordination, i.e., managing interference across multiple cells by allowing limited cooperation between base stations of adjacent cells. Such coordination mechanisms are partly already included in present standards, e.g., [3GP06, WIN08].

Research continues towards Coordinated Multipoint (CoMP) [PDF+08, SRH10b] and network MIMO [VLV07] which try to use several base stations jointly for downlink beamforming across multiple cells. Moreover, interference cancellation and interference alignment techniques are currently being discussed. More details on current and prospective interference coordination techniques are provided in [ACH07] and [BPG+09].

Understanding interference in a much broader context is the key to building a more efficient generation of wireless networks. This implies that we need to move away from using P2P links as the essential building blocks to more sophisticated structures such as the interference chan-

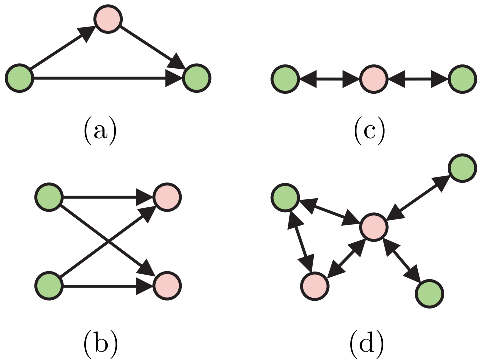


Figure 14.1.: Elementary building blocks of next-generation wireless systems: (a) relay-enhanced channel; (b) interference channel; (c) two-way relaying channel; (d) ad-hoc communication system

nel, the relay-enhanced channel, the two-way relaying channel, or the ad-hoc communication scheme [DHL+11], as exemplified in Figure 14.1. These building blocks themselves have been known for a long time, see for instance the seminal papers by Shannon [Sha61] on two-way communication channels or the well cited papers on the interference channel [HK81] and the relay channel [vdM71, CE79]. Yet, some of their properties are still not fully understood and more research is needed until we can construct larger networks from these building blocks without wasting any of their potential.

In this thesis we contribute to this mission by investigating one particular aspect of such building blocks further, namely relay-assisted communications in a two-way relaying fashion. In general, the deployment of relays is a very attractive approach for improving wireless networks in the near-future, see for instance [PWS+04] for a comprehensive survey of various relaying techniques and their impact on wireless communication systems. In fact, even current candidates for 4G standards like WiMAX or LTE-Advanced already include initial support for relays [YHXM09, 3GP10]. The main advantage of this approach is the great flexibility we gain. Small simple low-cost relay devices can likely be deployed in larger quantities, contributing to the goal of an increase in network node density. The more network nodes we can control, the more flexibility we gain in how to use them. A significant part of the existing literature on relaying is dedicated to one-way relaying. Here one-way means that the transmission is directed in one direction, i.e., from a specific source node via one or several relays to a specific destination node. For instance, one-way relaying can be used for coverage extension, acting as a wirelessly connected base station for users far away from the cell center. Alternatively, relays can be employed in a cooperative manner improving data rate or diversity [PWS+04]. The one-way relaying channel is comparably well understood. Performance limits, achievable rates, and efficient signaling schemes in the single hop case are, for example, examined in [NBK04], a treatment of the multi-hop case is found in [BFY04]. However, the main drawback of one-way relaying is that for a bidirectional exchange of information, at least four time slots are needed if the relay operates in half duplex mode. This incurs a fundamental loss in spectral efficiency.

To overcome this drawback, we focus on two-way relaying [RW05] where two communication partners exchange data bidirectionally with the assistance of one relay node. Each of these communication partners could be a mobile user as well as a fixed base station. Therefore, two-way relaying can be applied to user/network connections (as in a "classical" downlink/uplink) as well as to user/user connections (as in ad-hoc networks). In the latter case, traffic exchange can be handled by a relay station without the interaction of a base station, which allows to distribute the traffic load more evenly. Moreover, as we discuss in the following sections, via two-way relaying we can achieve a bidirectional exchange of information in only two time slots while all nodes operate in half-duplex mode. Consequently, two-way relaying allows a very efficient use of the scarce available radio resources without posing large demands on the hardware complexity of the relay nodes. For this reason we consider two-way relaying as a strong candidate for a relaying protocol that can be implemented in the near future.

The two-way relaying protocol was popularized by [RW05, RW07] as a means for compensating the spectral efficiency loss in one-way relaying due to the half duplex constraint of the relay. Two-Way Relaying can be combined with both operation modes discussed in Section 14.2, i.e., AF and DF. We focus on the (digital) AF scheme due to the advantages mentioned at the end of Section 14.2. For a thorough treatment of two-way relaying with DF relays, the reader is referred to [OSBB08], [OB07], [LJS06], [SAKH11], or [YCP11]. Note that besides AF and DF other types of two-way relaying schemes exist, e.g., Space-Time Coding is discussed in [CGHN09], XOR and superposition coding are investigated in [HKE+07], and Compute-and-Forward based two-way relaying is shown in [KKE11].

In order to combine two-way relaying with AF relays, we need channel state information at both terminals. Therefore, we discuss algebraic channel estimation schemes for two-way relaying with MIMO AF relays in Chapter 15. These provide both terminals with all relevant channel parameters. Moreover, the channel can also be estimated at the relay station.

As we discuss in Chapter 16, if the relay station possesses channel state information, the amplification matrix can be adjusted in order to improve the transmissions between the two nodes. Existing literature on the design of the relay amplification matrix suffers from major drawbacks: The proposed algorithms are either designed based on complicated convex optimization procedures [ZLCC09] with a high computational complexity or in an ad-hoc manner without a particular cost function in mind [UK08, VH11]. This is the main motivation for proposing simple algebraic approaches that outperform existing ad-hoc proposals with respect to the bit error rate or the sum rate (cf. Section 16.5) while being significantly less complex than convex optimization based algorithms.

The following chapters are devoted to our contributions in the field of two-way relaying. The required prerequisites are provided in this chapter with a discussion on various relay operation modes in Section 14.2 and the notation and the data model in Section 14.3. Chapter 15 is focused on channel estimation in two-way relaying. The design of the relay amplification matrix is discussed in Chapter 16. Finally, Chapter 17 provides a summary of our contributions and an outlook to possible future work.

14.2. Relay operation modes

In general, a relay node can be viewed as a device which receives a signal of interest, applies some form of processing to it, and then retransmits the processed version of this signal to one or multiple destinations. Therefore, relays can be classified according to nature of the processing they apply.

A first group of relays are so-called *Decode and Forward (DF)* relays. Their task is to decode the received signal until the actual transmitted information sequence is retrieved, applying similar processing as the destination of the transmission. In a second step, the information is re-encoded, either using the same Modulation and Coding Scheme (MCS) or possibly even modifying it to adapt to the conditions of the next transmission link. The advantage of DF relays is that if no decoding errors occur, a “clean” (noise-free) version of the signal is retransmitted. Moreover, the design of receivers and transmitters for DF relays follows more closely the design of existing wireless network nodes since they behave much like receivers and transmitters in P2P links. However, the disadvantage is that DF relays require the full base-band processing chain, including decoders for the MCSs that are applied in the network. Since the retransmission cannot be initiated before the entire information sequence has been decoded, this may cause significant additional latency, for instance, if Turbo codes [BGT93] are used that require long interleavers. In conjunction with two-way relaying, DF relays have another disadvantage. If two users transmit to the relay at the same time and the relay has to decode their messages, the achievable rate region is limited by the multiple access (MAC) channel. Likewise, if the relay transmits to both users in the second phase, there is a limitation of the broadcast (BC) channel (characterized by the coupling due to the common input, cf. [OSBB08]).

This is the main motivation for a second group of relays with a lower complexity, namely, the so-called *Amplify and Forward (AF)* relays. Their task is to retransmit an amplified version of the received signal without decoding it. This means that the receive noise is also amplified and retransmitted, which is the main drawback of AF relays. Yet, their advantage is that they do not need to decode the information which means that they cause less latency and that they do not need to know the MCSs applied between source and destination to encode the information[1] Moreover, the MAC or BC limitation does not apply, as we discuss in Section 14.3[2].

However, close inspection reveals that the group of AF relays must indeed be divided into two subgroups. It is often stated that the main advantage of AF relays is that they require no base-band processing since the amplification can be performed directly in the Radio Frequency (RF) band. However, this "pure" form of AF, which we will call *"analog AF"*, implies that the signal cannot be stored and hence the relay must operate in full-duplex mode. This poses a severe problem since the device receives a signal several orders of magnitude weaker than the signal it transmits itself. There are two different forms of such devices (also called "repeaters" [3GP11]): in-band and cross-band (out-of-band) relays [GR11]. Cross-band repeaters receive the signal in one frequency band then shift it to another frequency and retransmit it there. The advantage is that duplex interference can be avoided by using duplex filters which block the transmitted signal from entering the receiver chain by placing it into their stop band. If the two bands are sufficiently far from each other, a filter with a sufficiently large stop band attenuation can easily be designed. However, this implies that less than half of the available system bandwidth can be used for the transmission from source to destination. Moreover, due to the frequency translation it seems difficult to apply phase modulation since phase information is lost if the frequency is altered. On the other hand, in-band repeaters retransmit on the same frequency. This is problematic since their own transmitted signal can reenter their receive chain and it is typically several orders of magnitudes stronger. Consequently, if this self-interference cannot be prevented (e.g., by physically separating transmit and receive side), very sophisticated loop-interference cancellation techniques are required [RWW09]. Note that this requires an extremely large dynamic range of the receivers due to the large power differences.

This shows that the complexity argument for analog AF relays has to be treated with care. Saving the cost of a base-band processing chain comes at the price of increased complexity due to the full-duplex operation mode. There is a second sub-class of AF relays, which we call *"digital AF"*. These devices are equipped with a base-band processing chain, i.e., the received

[1] This is also advantageous if multiple operators share one relay in the context of voluntary spectrum and infrastructure sharing, see Section 17.1.

[2] Note that this does not mean that the rate region for AF relays is larger than for DF relays since AF relays waste a part of their transmit power for amplifying the noise and for transmitting the self-interference which is later cancelled at the terminals.

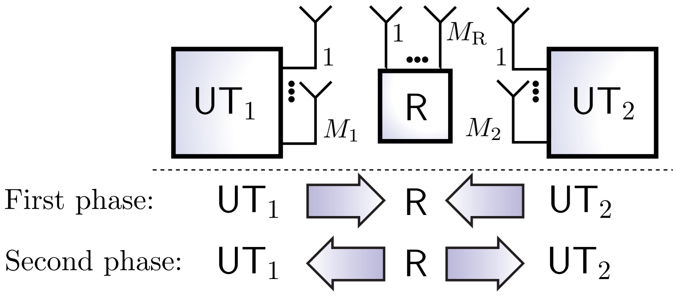


Figure 14.2.: Two-way relaying system model: Two terminals equipped with M_1 and M_2 antennas communicate with a relay station that has M_R antennas. There are two transmission phases: First both terminals transmit to the relay then the relay sends the amplified signal back to both terminals.

signal is first converted to the base-band. However, it is then amplified in the "digital" domain without decoding it, e.g., by applying a linear transformation to the received signal vector. The amplified signal is then transformed back into the RF band and retransmitted in a second time slot in a Time Division Duplexing (TDD) fashion. The obvious advantage of digital AF over analog AF is that digital AF relays can operate in half-duplex mode. Consequently, digital AF relays represent a good trade-off, combining the advantages of analog AF and DF relays: We neither need to decode the transmitted information, nor do we require sophisticated full-duplex hardware. Moreover, combined with two-way relaying we achieve a bidirectional exchange of information using only two time slots. Another advantage is an inherent near-far robustness since both signals travel through both channels and are hence affected by the stronger and by the weaker path loss [RH09c].

Note that besides AF and DF, other relay operation modes exist. For instance, XOR and superposition coding in the context of two-way relaying are discussed in [HKE+07]. Moreover, estimate-and-forward (EF) as well as compress-and-forward (CF) in the context of one-way relaying are found in [LY06a].

14.3. Notation and data model

As we have motivated in the previous sections we consider two-way relaying with a digital AF relay for the bidirectional exchange of information between two nodes UT_1 and UT_2. The corresponding system model is shown in Figure 14.2. We consider the general MIMO case where the nodes are equipped with M_1 and M_2 antennas, and the relay possesses M_R antennas, respectively.

We assume quasi-static block fading channels, i.e., the channel does not change significantly within the coherence time, which is long enough to acquire CSI at an adequate quality. We also consider frequency-flat fading, since a frequency-selective channel can be converted into a set of parallel frequency-flat fading channels by employing Orthogonal Frequency Division Multiplexing (OFDM) and chunk-wise processing. Consequently, the received signal at the relay during the first time slot can be expressed as

$$\boldsymbol{r} = \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{x}_1 + \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{x}_2 + \boldsymbol{n}_\mathrm{R} \in \mathbb{C}^{M_\mathrm{R} \times 1}, \tag{14.1}$$

where $\boldsymbol{H}_1^{(\mathrm{f})} \in \mathbb{C}^{M_\mathrm{R} \times M_1}$ and $\boldsymbol{H}_2^{(\mathrm{f})} \in \mathbb{C}^{M_\mathrm{R} \times M_2}$ denote the forward MIMO channel matrices between the terminals and the relay, $\boldsymbol{x}_1 \in \mathbb{C}^{M_1 \times 1}$ and $\boldsymbol{x}_2 \in \mathbb{C}^{M_2 \times 1}$ are the transmitted signals by the terminals, and $\boldsymbol{n}_\mathrm{R}$ represents the additive noise component at the relay which is assumed to be independent of the transmitted signals. The covariance matrices of the transmit signals[3] $\boldsymbol{x}_i$ and the noise component at the relay $\boldsymbol{n}_\mathrm{R}$ are given by $\boldsymbol{R}_i = \mathbb{E}\left\{\boldsymbol{x}_i \cdot \boldsymbol{x}_i^\mathrm{H}\right\} \in \mathbb{C}^{M_i \times M_i}$ and $\boldsymbol{R}_{\mathrm{N,R}} = \mathbb{E}\left\{\boldsymbol{n}_\mathrm{R} \cdot \boldsymbol{n}_\mathrm{R}^\mathrm{H}\right\} \in \mathbb{C}^{M_\mathrm{R} \times M_\mathrm{R}}$, respectively. The transmit power of UT_i is then defined as $P_{\mathrm{T},i} = \mathrm{trace}\left\{\boldsymbol{R}_i\right\}$.

The relay amplifies the received signal by applying linear processing to the received vector $\boldsymbol{r}$. This can be expressed in terms of a relay amplification matrix $\boldsymbol{G} \in \mathbb{C}^{M_\mathrm{R} \times M_\mathrm{R}}$ in the following manner

$$\bar{\boldsymbol{r}} = \gamma \cdot \boldsymbol{G} \cdot \boldsymbol{r} = \boldsymbol{G}_\gamma \cdot \boldsymbol{r}, \tag{14.2}$$

where $\gamma \in \mathbb{R}$ is a scalar relay amplification factor which scales the transmitted signal to an appropriate power level and $\boldsymbol{G}_\gamma \doteq \gamma \cdot \boldsymbol{G}$. Typically we assume that there is an average power constraint at the relay, i.e.,

$$P_{\mathrm{T,R}} \doteq \mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} \leq P_{\mathrm{T,R}}^{\max}, \tag{14.3}$$

where the expectation is with respect to the random source symbols and the noise realizations.

The processed signal $\bar{\boldsymbol{r}}$ is then transmitted by the relay and received by the nodes in the subsequent second time slot. The corresponding received signals can be expressed as

$$\boldsymbol{y}_1 = \boldsymbol{H}_1^{(\mathrm{b})} \cdot \bar{\boldsymbol{r}} + \boldsymbol{n}_1 \in \mathbb{C}^{M_1 \times 1} \tag{14.4}$$

$$\boldsymbol{y}_2 = \boldsymbol{H}_2^{(\mathrm{b})} \cdot \bar{\boldsymbol{r}} + \boldsymbol{n}_2 \in \mathbb{C}^{M_2 \times 1}, \tag{14.5}$$

[3]Note that we assume the transmit signals to be zero mean, which is very reasonable in practice since a mean different from zero implies a waste of power on a DC component (which carries no information). If $\boldsymbol{x}_i$ has a mean different from zero, $\boldsymbol{R}_i$ should be defined as the correlation matrix $\boldsymbol{R}_i = \mathbb{E}\left\{\boldsymbol{x}_i \cdot \boldsymbol{x}_i^\mathrm{H}\right\}$, not as the covariance matrix $\mathbb{E}\left\{(\boldsymbol{x}_i - \mathbb{E}\{\boldsymbol{x}_i\}) \cdot (\boldsymbol{x}_i - \mathbb{E}\{\boldsymbol{x}_i\})^\mathrm{H}\right\}$.

where $\boldsymbol{H}_1^{(\mathrm{b})} \in \mathbb{C}^{M_1 \times M_\mathrm{R}}$ and $\boldsymbol{H}_2^{(\mathrm{b})} \in \mathbb{C}^{M_2 \times M_\mathrm{R}}$ are the corresponding backward MIMO channels from the relay to the nodes and $\boldsymbol{n}_1 \in \mathbb{C}^{M_1 \times 1}$, $\boldsymbol{n}_2 \in \mathbb{C}^{M_2 \times 1}$ represent the terminals' own noise contributions which are assumed to be independent of the noise at the relay station and the transmitted signals. Inserting (14.1) and (14.2) into (14.4) and (14.5) we obtain

$$\boldsymbol{y}_1 = \boldsymbol{H}_{1,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 + \boldsymbol{H}_{1,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 + \tilde{\boldsymbol{n}}_1 \in \mathbb{C}^{M_1 \times 1} \tag{14.6}$$

$$\boldsymbol{y}_2 = \boldsymbol{H}_{2,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 + \boldsymbol{H}_{2,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 + \tilde{\boldsymbol{n}}_2 \in \mathbb{C}^{M_2 \times 1}, \tag{14.7}$$

where $\boldsymbol{H}_{i,j}^{(\mathrm{e})} \doteq \gamma \cdot \boldsymbol{H}_i^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_j^{(\mathrm{f})} \in \mathbb{C}^{M_i \times M_j}$ represents the "effective" channel (also called "compound" channel) between terminal j (transmitting) and terminal i (receiving). Moreover, the vector $\tilde{\boldsymbol{n}}_i = \boldsymbol{H}_i^{(\mathrm{b})} \cdot \gamma \cdot \boldsymbol{G} \cdot \boldsymbol{n}_\mathrm{R} + \boldsymbol{n}_i$ symbolizes the effective noise contribution at terminal $i = 1, 2$ which consists of the terminal's own noise as well as the forwarded relay noise.

The first term in the nodes' received signals in (14.6) and (14.7) represents their own transmitted signals conveyed through the effective channels $\boldsymbol{H}_{1,1}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,2}^{(\mathrm{e})}$, respectively. Consequently, these quantities are referred to as *self-interference* since they are created by each nodes' own transmitted signal $\boldsymbol{x}_i$. An important aspect to note is that since UT_i knows its own transmitted signal $\boldsymbol{x}_i$, the self-interference can simply be subtracted from the received signal, provided that the effective channel matrix $\boldsymbol{H}_{i,i}^{(\mathrm{e})}$ is known. This step is also referred to as Analog Network Coding (ANC) [KGK07].

If the terminals possess perfect channel knowledge, the self-interference can be perfectly canceled. We then have

$$\boldsymbol{z}_1 = \boldsymbol{y}_1 - \boldsymbol{H}_{1,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 = \boldsymbol{H}_{1,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 + \tilde{\boldsymbol{n}}_1 \tag{14.8}$$

$$\boldsymbol{z}_2 = \boldsymbol{y}_2 - \boldsymbol{H}_{2,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 = \boldsymbol{H}_{2,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 + \tilde{\boldsymbol{n}}_2. \tag{14.9}$$

This shows that ANC basically decomposes the two-way relaying system into two parallel single-user MIMO systems. The difference to a "regular" P2P MIMO system is that the additive noise cannot be considered white since $\tilde{\boldsymbol{n}}_i$ contains the forwarded relay noise. More specifically, the noise covariance matrix is given by

$$\mathbb{E}\left\{\tilde{\boldsymbol{n}}_i \cdot \tilde{\boldsymbol{n}}_i^\mathrm{H}\right\} = \gamma^2 \cdot \boldsymbol{H}_i^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^\mathrm{H} \cdot \boldsymbol{H}_i^{(\mathrm{b})^\mathrm{H}} + \boldsymbol{R}_{\mathrm{N},i}, \tag{14.10}$$

where $\boldsymbol{R}_{\mathrm{N},i} = \mathbb{E}\left\{\boldsymbol{n}_i \cdot \boldsymbol{n}_i^\mathrm{H}\right\}$ represent the nodes' noise covariance matrices for $i = 1, 2$, respectively. Another important aspect to note is that even though (14.8) and (14.9) suggest that the two P2P MIMO links are fully independent that is not true. There is a coupling between them since both $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ contain the relay amplification matrix $\gamma \cdot \boldsymbol{G}$ which depends on both transmit signals at least weakly via the power constraint (14.3): If one terminal uses more power, the relay has to lower its relay amplification factor γ which lowers the norm of both effective channels $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$.

Still, both terminals can apply single-user MIMO transmission techniques over the effective channels, e.g., Spatial Multiplexing (SMUX) with Water-Filling or Dominant Eigenmode Transmission (DET) which are known to be the sum-rate maximizing and the Signal to Noise Ratio (SNR)-maximizing strategies in P2P MIMO systems [PNG03].

Note that in general, the forward and backward channels are not assumed to be reciprocal. However, since we operate in TDD mode, the reciprocity assumption may actually be fulfilled. In this special case we can simplify our notation to $\boldsymbol{H}_i^{(\mathrm{f})} = \boldsymbol{H}_i = \boldsymbol{H}_i^{(\mathrm{b})^{\mathrm{T}}}$ for $i = 1, 2$ and hence

$$\boldsymbol{H}_{i,j}^{(\mathrm{e})} = \gamma \cdot \boldsymbol{H}_i^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_j. \tag{14.11}$$

We assume reciprocity for the derivation of the TENCE algorithm and its SLS-based iterative refinement in Chapter 15. For the design of the relay amplification matrix in Chapter 16, reciprocity is not assumed, unless stated otherwise.

15. Channel estimation

In this section we discuss channel estimation for two-way relaying with a MIMO AF relay. We assume reciprocity, i.e., $\boldsymbol{H}_i^{(\mathrm{f})} = \boldsymbol{H}_i = \boldsymbol{H}_i^{(\mathrm{b})^{\mathrm{T}}}$ for $i = 1, 2$ and hence $\boldsymbol{H}_{i,j}^{(\mathrm{e})} = \gamma \cdot \boldsymbol{H}_i^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_j$, cf. (14.11). We first introduce a simple closed-form Least Squares (LS)-based compound channel estimator [RH10c] in Section 15.2, which only estimates the compound channels $\boldsymbol{H}_{i,j}^{(\mathrm{e})}$ at UT_i. Then we show the algebraic Tensor-Based Channel Estimation (TENCE) scheme [RH10c, RH09e] in Section 15.3, which estimates the channels $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ at both terminals directly, taking advantage of the special structure of $\boldsymbol{H}_{i,j}^{(\mathrm{e})}$. Since TENCE is not MMSE-optimal, we provide a simple iterative refinement based on Structured Least Squares (SLS) [RH10c, RH09d] in Section 15.4. Note that SLS has been discussed in conjunction with ESPRIT in Chapter 11 of this thesis. The algorithms are discussed and compared based on simulations in Sections 15.5 and 15.6, respectively.

15.1. Problem statement and state of the art

As shown in (14.8) and (14.9), applying ANC in a two-way relaying system with an AF relay decouples the bidirectional transmission into two parallel P2P MIMO links. Therefore, in order to enable the terminals to transmit information, each terminal should know the following quantities:

- Its own "self-interference" channel, i.e., $\boldsymbol{H}_{1,1}^{(\mathrm{e})}$ for UT_1 and $\boldsymbol{H}_{2,2}^{(\mathrm{e})}$ for UT_2. This information is critical since it is the basis for the ANC step. Note that if the relay cancels the self-interference (as, for instance, in the ZF transceiver proposed in [UK08]), these channels are not needed (in fact, they are equal to zero).
- Its own effective receive channel, i.e., $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ for UT_1 and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ for UT_2. This channel is needed to decode the transmission from the other terminal.
- There might also be some benefit in knowning its own "forward" channel, i.e., $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ for UT_1 and $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ for UT_2. These are the channels into which the terminals transmit their own signals. Therefore, if the channels are known, the terminals can apply transmit precoding to match their signals to the channels. In the special case where the relay chooses a relay amplification matrix which is symmetric, i.e., $\boldsymbol{G} = \boldsymbol{G}^{\mathrm{T}}$, we have $\boldsymbol{H}_{1,2}^{(\mathrm{e})} = \boldsymbol{H}_{2,1}^{(\mathrm{e})^{\mathrm{T}}}$ and, therefore, the terminals can obtain transmit CSI from their receive CSI.

Most previous publications on two-way AF relaying have assumed that channel knowledge is available at the terminals. For one-way AF-based relaying, a large number of results on channel estimation is available. Prominent examples include [AGU10, LZZL09, KYKI10] for one single-antenna AF relay, [GCN08] for a network of single-antenna AF relays, [KH11, LV08, MOZL09, RKX12] for one MIMO AF relay, or [WdLM11] for a multi-hop network of AF relay nodes. Maximum likelihood channel estimation schemes for two-way relaying with AF relays are proposed in [GZL09b] for the flat fading case and in [GZL09a] for frequency-selective fading, however these techniques are limited to the single-antenna case and a MIMO extension is not straightforward. Further approaches that are limited to the single-antenna case are [JGGN10] (which considers power allocation), [AP10] (which studies semi-blind channel estimation using constant modulus modulation), [YCHY10] (which is based on OFDM), [WGZT10] (which considers joint channel and carrier frequency offset estimation), or [CGR+12] (which applies compressed sensing concepts). Channel estimation in two-way relaying systems with multiple antennas is limited to relays employing DF [ZKWB08] or space-time coding [TSQ09]. The authors of [PLNG10] consider channel estimation in MIMO two-way relaying systems with frequency-selective fading based on OFDM and relays using a scalar real-valued amplification α per antenna only. Note that [PLNG10] cannot be compared to the channel estimation schemes proposed in this chapter since (a) we consider another form of AF where the relay may multiply the received signal vector with one complex relay amplification matrix and (b) in [PLNG10] the frequency-selective channels and the resulting circulant structure of the channels in the time domain due to the cyclic prefix is explicitly exploited whereas our focus is on flat fading only. We introduce two different approaches to the channel estimation problem. First, we discuss the compound LS-based channel estimator which provides UT_1 with estimates for $\boldsymbol{H}_{1,1}^{(\mathrm{e})}$ and $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and UT_2 with estimates for $\boldsymbol{H}_{2,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$, i.e., the terminals do not have transmit CSI unless $\boldsymbol{G} = \boldsymbol{G}^{\mathrm{T}}$.

Then, we introduce the TENCE scheme and its SLS-based iterative refinement, which provide both terminals with knowledge of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$. Based on these channels, all compound channels can be constructed, provided that the terminals know the relay's transmit strategy. Note that there may be an additional benefit in knowing the channels $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$. In fact, it is often proposed to choose the terminals' transmit strategies based on $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ instead of $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$, cf. [XH10, LCSK10, JS10b, WT12]. In light of this, it is an attractive feature of the proposed TENCE scheme that transmit CSI ($\boldsymbol{H}_i$), receive CSI ($\boldsymbol{H}_{i,j}^{(\mathrm{e})}$), and the required self-interference channel ($\boldsymbol{H}_{i,i}^{(\mathrm{e})}$) are obtained at UT_i, $i = 1, 2$ at the same time without the need for explicit feedback.

15.2. Compound channel estimator

In this section we show a LS-based scheme for estimating the compound channel matrices $\boldsymbol{H}_{i,j}^{(\mathrm{e})}$ at UT_i for $i, j = 1, 2$. While this scheme is simple and robust, it is not necessarily optimal, since it ignores the special structure of the compound channel matrices. It also fails to provide UT_i with an estimate of $\boldsymbol{H}_{j,i}^{(\mathrm{e})}$ which it needs to compute a proper precoding matrix. Note that $\boldsymbol{H}_{j,i}^{(\mathrm{e})} = \boldsymbol{H}_{i,j}^{(\mathrm{e})^{\mathrm{T}}}$ only if $\boldsymbol{G} = \boldsymbol{G}^{\mathrm{T}}$. We have shown in [RH09c] that ANOMAX (cf. Section 16.3.1) with unequal weighting should be chosen in near-far scenarios or if the UTs have different target SINRs. In this case, $\boldsymbol{G} \neq \boldsymbol{G}^{\mathrm{T}}$.

In order to estimate the channels, both terminals transmit a sequence of N_{P} pilot symbols $\boldsymbol{x}_{1,j}$, $\boldsymbol{x}_{2,j}$ for $j = 1, 2, \ldots, N_{\mathrm{P}}$. The overall training data received by the relay can be expressed as

$$\boldsymbol{Y}_{\mathrm{R}} = \boldsymbol{H}_1 \cdot \boldsymbol{X}_1 + \boldsymbol{H}_2 \cdot \boldsymbol{X}_2 + \boldsymbol{N}_{\mathrm{R}} \in \mathbb{C}^{M_{\mathrm{R}} \times N_{\mathrm{P}}}, \tag{15.1}$$

where the pilot symbol matrices $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are defined as

$$\boldsymbol{X}_i = \left[\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, \ldots, \boldsymbol{x}_{i,N_{\mathrm{P}}}\right] \in \mathbb{C}^{M_i \times N_{\mathrm{P}}}. \tag{15.2}$$

Let $\boldsymbol{X} = \left[\boldsymbol{X}_1^{\mathrm{T}}, \boldsymbol{X}_2^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{C}^{(M_1+M_2) \times N_{\mathrm{P}}}$. Then, a least-squares estimate of the channel matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ at the relay station is obtained via

$$\left[\hat{\boldsymbol{H}}_1, \hat{\boldsymbol{H}}_2\right] = \boldsymbol{Y}_{\mathrm{R}} \cdot \boldsymbol{X}^{+}. \tag{15.3}$$

Note that (15.3) requires $N_{\mathrm{P}} \geq M_1 + M_2$. Based on these estimates, the relay can compute a suitable relay amplification matrix $\boldsymbol{G}$, e.g., via the Algebraic Norm-Maximizing (ANOMAX) transmit strategy [RH09a]. The received training data $\boldsymbol{R}$ is then multiplied with $\boldsymbol{G}$ and transmitted back to the terminals. The signal received at UT_i, $i = 1, 2$ can be expressed as

$$\boldsymbol{Y}_i = \boldsymbol{H}_{i,i}^{(\mathrm{e})} \cdot \boldsymbol{X}_i + \boldsymbol{H}_{i,j}^{(\mathrm{e})} \cdot \boldsymbol{X}_j + \tilde{\boldsymbol{N}}_i \tag{15.4}$$

Consequently, the LS estimates of the effective channels are given by

$$\left[\hat{\boldsymbol{H}}_{1,1}^{(\mathrm{e})}, \hat{\boldsymbol{H}}_{1,2}^{(\mathrm{e})}\right] = \boldsymbol{Y}_1 \cdot \boldsymbol{X}^{+} \quad \text{for } \mathsf{UT}_1 \text{ and } \quad \left[\hat{\boldsymbol{H}}_{2,1}^{(\mathrm{e})}, \hat{\boldsymbol{H}}_{2,2}^{(\mathrm{e})}\right] = \boldsymbol{Y}_2 \cdot \boldsymbol{X}^{+} \quad \text{for } \mathsf{UT}_2, \tag{15.5}$$

where we again require that $N_{\mathrm{P}} \geq M_1 + M_2$. Consequently, with $M_1 + M_2$ pilots we have estimated the channel matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ at the relay, the effective channel matrices $\boldsymbol{H}_{1,1}^{(\mathrm{e})}$

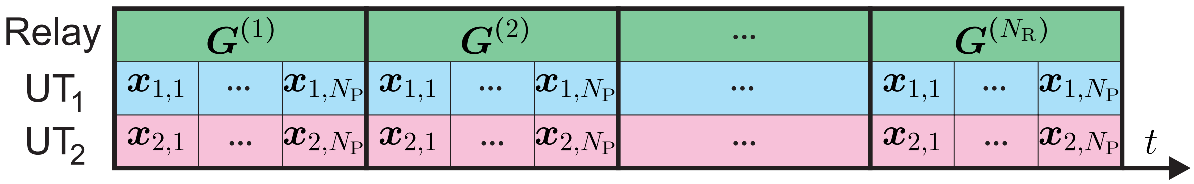


Figure 15.1.: Visualization of the pilot sequences in the training phase for TENCE: N_P pilots are transmitted from both terminals in N_R consecutive frames. In each frame, the relay forwards the pilots using a different amplification matrix $\boldsymbol{G}^{(i)}$, $i = 1, 2, \dots, N_\mathrm{R}$.

and $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ at UT_1, and the effective channel matrices $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,2}^{(\mathrm{e})}$ at UT_2. However, to compute proper precoding matrices, UT_1 requires an estimate of $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ and UT_2 needs an estimate of $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$. In the case where the relay chooses its amplification matrix $\boldsymbol{G}$ such that $\boldsymbol{G} = \boldsymbol{G}^\mathrm{T}$, UT_1 can obtain an estimate of $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ via $\hat{\boldsymbol{H}}_{2,1}^{(\mathrm{e})} = \hat{\boldsymbol{H}}_{1,2}^{(\mathrm{e})^\mathrm{T}}$. Otherwise, additional pilots are needed to estimate $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ at UT_1 and $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ at UT_2. Alternatively, open loop techniques such as Orthogonal Space-Time Block Codes [TSC88] can be used to convey the desired information without transmit channel state information. Another drawback of the simple LS-based channel estimation procedure is that the structure of the compound channels is completely ignored. We show in the next section how the estimation accuracy can be improved by exploiting this special structure and estimating the channel matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ directly.

15.3. Tensor-based Channel Estimation (TENCE)

The LS-based scheme for the estimation of the effective (compound) channel ignores their structure completely. For instance, $\boldsymbol{H}_{i,i}^{(\mathrm{e})} = \boldsymbol{H}_i^\mathrm{T} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_i$, i.e., the M_i^2 elements of $\boldsymbol{H}_{i,i}^{(\mathrm{e})}$ are second-order polynomials in the $M_i \cdot M_\mathrm{R}$ coefficients in $\boldsymbol{H}_i$. Consequently, if $M_\mathrm{R} < M_i$ it may be more efficient to estimate $\boldsymbol{H}_i$ by solving a quadratic LS problem and exploiting the special structure of $\boldsymbol{H}_{i,i}^{(\mathrm{e})}$. This is the motivation behind the tensor-based channel estimation (TENCE) scheme presented in this section. TENCE itself is an algebraic (i.e., non-iterative) solution to the non-linear least squares problem, which is very simple to compute. If a more accurate solution is required, TENCE can be refined by a few iterations of an iterative channel estimation scheme described in Section 15.4.

15.3.1. Training

In order to acquire channel knowledge of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ at the user terminals we require a special training phase in which known pilot symbols are transmitted for known relay amplification matrices. We therefore divide the training phase into N_{R} frames. For each frame, we choose a different relay amplification matrix $\boldsymbol{G}^{(i)} \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}}}, i = 1, 2, \ldots, N_{\mathrm{R}}$. For every $\boldsymbol{G}^{(i)}$, the same pilot sequences $\boldsymbol{x}_{1,j} \in \mathbb{C}^{M_1}$ and $\boldsymbol{x}_{2,j} \in \mathbb{C}^{M_2}$ for $j = 1, 2, \ldots, N_{\mathrm{P}}$ are transmitted from UT_1 and UT_2, respectively. The number of pilot symbols N_{P} that are transmitted for each $\boldsymbol{G}^{(i)}$ and the number of frames N_{R} will be specified later. Note that the total number of training time slots is given by $N_{\mathrm{R}} \cdot N_{\mathrm{P}}$. We depict this process in Figure 15.1 where we show the pilot vectors transmitted by the terminals as well as the relay amplification matrix used by the relay station during the entire training phase. The received signal from the j-th pilot symbol within the i-th training block is given by

$$\begin{aligned}
\boldsymbol{y}_{1,j,i} &= \boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}^{(i)} \cdot \boldsymbol{H}_1 \cdot \boldsymbol{x}_{1,j} + \boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}^{(i)} \cdot \boldsymbol{H}_2 \cdot \boldsymbol{x}_{2,j} + \tilde{\boldsymbol{n}}_{1,j,i} \\
\boldsymbol{y}_{2,j,i} &= \boldsymbol{H}_2^{\mathrm{T}} \cdot \boldsymbol{G}^{(i)} \cdot \boldsymbol{H}_1 \cdot \boldsymbol{x}_{1,j} + \boldsymbol{H}_2^{\mathrm{T}} \cdot \boldsymbol{G}^{(i)} \cdot \boldsymbol{H}_2 \cdot \boldsymbol{x}_{2,j} + \tilde{\boldsymbol{n}}_{2,j,i}.
\end{aligned} \tag{15.6}$$

The data model in (15.6) can be expressed in a more compact form using tensor notation, cf. Chapter 4. To this end, let us introduce the following definitions

$$\boldsymbol{H} \doteq \begin{bmatrix} \boldsymbol{H}_1 & \boldsymbol{H}_2 \end{bmatrix} \in \mathbb{C}^{M_{\mathrm{R}} \times (M_1 + M_2)} \tag{15.7}$$

$$\boldsymbol{\mathcal{G}} \doteq \begin{bmatrix} \boldsymbol{G}^{(1)} \sqcup_3 \boldsymbol{G}^{(2)} \ldots \sqcup_3 \boldsymbol{G}^{(N_{\mathrm{R}})} \end{bmatrix} \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}} \times N_{\mathrm{R}}}. \tag{15.8}$$

Using these definitions, the received training data can be rewritten as

$$\boldsymbol{\mathcal{Y}}_k = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{H}_k^{\mathrm{T}} \times_2 (\boldsymbol{H} \cdot \boldsymbol{X})^{\mathrm{T}} + \boldsymbol{\mathcal{N}}_k \in \mathbb{C}^{M_k \times N_{\mathrm{P}} \times N_{\mathrm{R}}},\ k = 1, 2 \tag{15.9}$$

where $\boldsymbol{X}$ is defined in (15.2) and the tensors $\boldsymbol{\mathcal{Y}}_1$ and $\boldsymbol{\mathcal{Y}}_2$ contain the vectors $\boldsymbol{y}_{1,j,i}$ and $\boldsymbol{y}_{2,j,i}$ in such a way that the second index in the tensor represents $j = 1, 2, \ldots, N_{\mathrm{P}}$ and the third index represents $i = 1, 2, \ldots, N_{\mathrm{R}}$. The tensors $\boldsymbol{\mathcal{N}}_1$ and $\boldsymbol{\mathcal{N}}_2$ collect of the noise vectors $\tilde{\boldsymbol{n}}_{1,j,i}$ and $\tilde{\boldsymbol{n}}_{2,j,i}$ in a similar fashion.

We depict the structure of (15.9) in Figure 15.2. It should be noted that the structure of (15.9) is similar to a Tucker-2 decomposition (cf. Section 4.3). However, the difference to Tucker-2 is that the core tensor $\boldsymbol{\mathcal{G}}$ is known (and can even be designed). Also, a certain symmetry in the factors is present since the 2-mode factor includes $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ which are also present in the 1-mode factor. Finally, the decomposition involves the pilot matrix $\boldsymbol{X}$ which is also known and can be designed. These particular properties can be exploited to derive efficient solutions to the channel estimation problem. Moreover, we obtain design rules and recommendations on how to choose the pilot matrix $\boldsymbol{X}$ and the training tensor $\boldsymbol{\mathcal{G}}$ in order to facilitate the implementation of these channel estimation algorithms[1].

[1]We use the term "design rules" for properties that $\boldsymbol{X}$ and $\boldsymbol{\mathcal{G}}$ *must* fulfill for TENCE to be applicable and

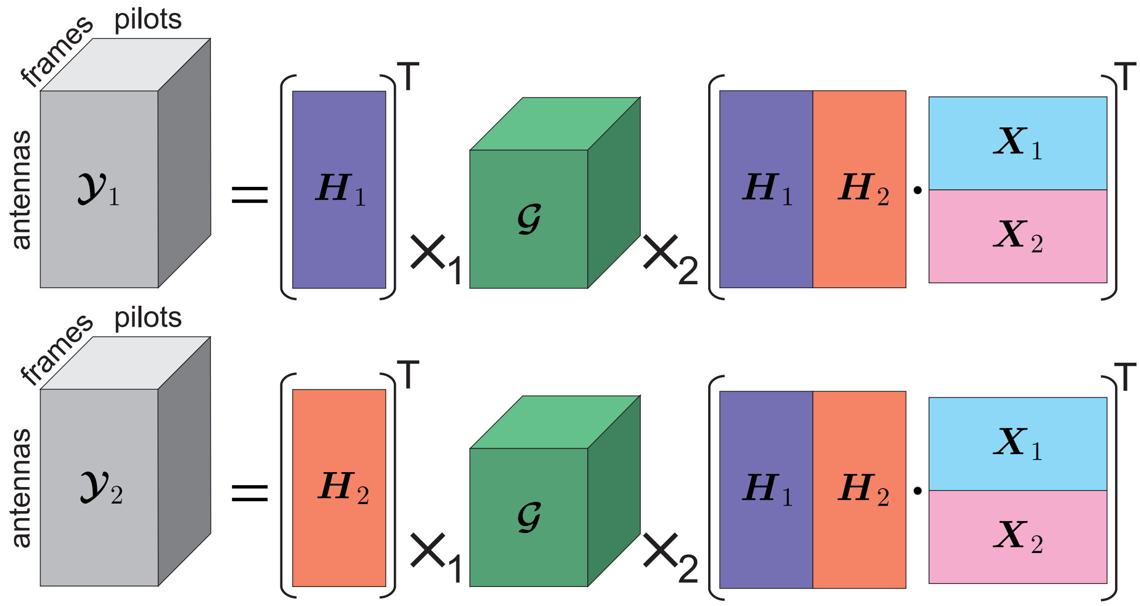


Figure 15.2.: Visualization of the structure of the received training data $\mathcal{Y}_1$ and $\mathcal{Y}_2$. We observe $\mathcal{Y}_1$ and $\mathcal{Y}_2$ and we want to estimate $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$. The tensor $\mathcal{G}$ and the matrices $\boldsymbol{X}_1, \boldsymbol{X}_2$ can be designed.

15.3.2. Derivation of TENCE

Based on this training data we show the derivation of TENCE in this section. For notational convenience, we ignore the contribution of the noise and write equalities. In the presence of (colored) noise (cf. (14.10)), the following identities will only hold approximately. Also, we derive the solution for UT_1 only. Due to the symmetry of the problem, the solution for UT_2 is very similar.

First of all, consider the training tensor $\mathcal{G}$. Let r_G be the rank of the tensor $\mathcal{G}$. Then $\mathcal{G}$ can be expressed in terms of its CP decomposition (cf. (4.34))

$$\mathcal{G} = \mathcal{I}_{3,r_G} \times_1 \boldsymbol{G}_1 \times_2 \boldsymbol{G}_2 \times_3 \boldsymbol{G}_3, \tag{15.10}$$

where $\mathcal{I}_{3,r_G}$ is the identity tensor of size $r_G \times r_G \times r_G$ and the matrices $\boldsymbol{G}_1 \in \mathbb{C}^{M_\mathrm{R} \times r_G}$, $\boldsymbol{G}_2 \in \mathbb{C}^{M_\mathrm{R} \times r_G}$, and $\boldsymbol{G}_3 \in \mathbb{C}^{N_\mathrm{R} \times r_G}$ represent the factor matrices of the decomposition. Instead of designing the tensor $\mathcal{G}$ directly, we propose design rules for the matrices $\boldsymbol{G}_1$, $\boldsymbol{G}_2$, and $\boldsymbol{G}_3$ individually from the steps in the derivation where they appear.

"design recommendations" for additional properties that $\boldsymbol{X}$ and $\mathcal{G}$ *may* satisfy to improve the estimation accuracy.

Inserting (15.10) into (15.9) yields

$$\boldsymbol{\mathcal{Y}}_1 = \boldsymbol{\mathcal{I}}_{3,r_G} \times_1 \left(\boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_1\right) \times_2 \left(\boldsymbol{X}^{\mathrm{T}} \cdot \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{G}_2\right) \times_3 \boldsymbol{G}_3. \tag{15.11}$$

Using the elementary properties of n-mode products shown in (4.25) in Section 4.1, it is easy to verify that the 3-mode unfolding of (15.11) satisfies

$$[\boldsymbol{\mathcal{Y}}_1]_{(3)} = \boldsymbol{G}_3 \cdot \left[\left(\boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_1\right) \diamond \left(\boldsymbol{X}^{\mathrm{T}} \cdot \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{G}_2\right)\right]^{\mathrm{T}}. \tag{15.12}$$

In order to isolate the Khatri-Rao product, the multiplication by $\boldsymbol{G}_3$ must be inverted. To guarantee that this inversion is unique, we require that $N_{\mathrm{R}} \geq r_G$ and $\boldsymbol{G}_3$ to be a full rank matrix. This leads to the first design rule for $\boldsymbol{\mathcal{G}}$:

Design Rule 1. *The number of training blocks N_{R} must satisfy $N_{\mathrm{R}} \geq r_G$ and $\boldsymbol{G}_3$ must have full column rank (r_G).*

Since we can design $\boldsymbol{G}_3$ we can choose this matrix such that it has orthogonal columns, i.e., $\boldsymbol{G}_3^{\mathrm{H}}\boldsymbol{G}_3$ is a diagonal matrix. This makes it easy to guarantee that the inversion step is well conditioned, which is favorable from a numerical standpoint (since inverting a non-orthogonal matrix inevitably leads to noise amplification) and avoids explicit matrix inversion.

Design Recommendation 1. *The 3-mode factor matrix $\boldsymbol{G}_3$ should have orthogonal columns.*

We can now isolate the Khatri-Rao product in (15.12) in the following way

$$\left(\boldsymbol{G}_3^{+} \cdot [\boldsymbol{\mathcal{Y}}_1]_{(3)}\right)^{\mathrm{T}} = \left(\boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_1\right) \diamond \left(\boldsymbol{X}^{\mathrm{T}} \cdot \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{G}_2\right), \tag{15.13}$$

where $\boldsymbol{G}_3^{+}$ is the pseudo-inverse of $\boldsymbol{G}_3$ (which is a scaled version of $\boldsymbol{G}_3^{\mathrm{H}}$ if $\boldsymbol{G}_3$ is chosen to have orthogonal columns).

The Khatri-Rao product in (15.13) can be inverted up to one scaling ambiguity per column. That means we can find matrices $\boldsymbol{F}_1 \in \mathbb{C}^{M_1 \times r_G}$ and $\boldsymbol{F}_2 \in \mathbb{C}^{N_{\mathrm{P}} \times r_G}$ such that

$$\boldsymbol{F}_1 = \boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_1 \cdot \boldsymbol{\Lambda} \tag{15.14}$$

$$\boldsymbol{F}_2 = \boldsymbol{X}^{\mathrm{T}} \cdot \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{G}_2 \cdot \boldsymbol{\Lambda}^{-1}, \tag{15.15}$$

where $\boldsymbol{\Lambda} = \operatorname{diag}\{[\lambda_1, \lambda_2, \ldots, \lambda_{r_G}]\}$ and λ_n represent arbitrary complex numbers. Since in the presence of noise (15.13) is only approximately a Khatri-Rao product, the factors represent an estimate. As discussed in Section 3.4, a Least-Squares optimal factorization of a Khatri-Rao product can be found by a sequence of SVDs[2]. The corresponding Least-Squares Khatri-Rao factorization algorithm is shown in Section 3.4, Algorithm 1.

Note that for column p there is one scaling ambiguity in inverting the Khatri-Rao product since $\boldsymbol{a}_p \otimes \boldsymbol{b}_p = (\lambda_p \cdot \boldsymbol{a}_p) \otimes (\boldsymbol{b}_p/\lambda_p)\,,\ \forall \lambda_p \in \mathbb{C} \setminus \{0\}$. In order to resolve the unknown parameters λ_p we need to eliminate the unknown channels in (15.14) and (15.15). First of all, $\boldsymbol{H}_2$ can easily be eliminated in (15.15) if we restrict the pilot matrix $\boldsymbol{X}$ to have orthogonal rows. Again, this choice is also desirable from a numerical point of view because then the pilot matrix does not affect the conditioning of the problem. Note that the rows can only be orthogonal if the matrix is square or "flat" which yields the necessary condition $N_{\mathrm{P}} \geq M_1 + M_2$.

Design Rule 2. *The number of pilot symbols per training block N_{P} must satisfy $N_{\mathrm{P}} \geq M_1 + M_2$.*

Design Rule 3. *The pilot symbol matrix $\boldsymbol{X} \in \mathbb{C}^{(M_1+M_2)\times N_{\mathrm{P}}}$ must have orthogonal rows.*

Note that we formulate the last property as a design rule instead of a recommendation since the subsequent derivations assume that $\boldsymbol{X}$ has orthogonal rows. It is possible to generalize the derivation to non-orthogonal pilots, this is discussed in Section 15.5.2.

From these design rules it also follows that the pilot transmissions of the two users are mutually orthogonal. Therefore,

$$\left(\boldsymbol{X}_1^{\mathrm{T}}\right)^+ \cdot \boldsymbol{X}^{\mathrm{T}} = [\boldsymbol{I}_{M_1},\, \boldsymbol{0}_{M_1\times M_2}] \quad \text{and} \quad \left(\boldsymbol{X}_2^{\mathrm{T}}\right)^+ \cdot \boldsymbol{X}^{\mathrm{T}} = [\boldsymbol{0}_{M_2\times M_1},\, \boldsymbol{I}_{M_2}]\,. \tag{15.16}$$

Due to the orthogonality constraint, $\left(\boldsymbol{X}_1^{\mathrm{T}}\right)^+$ and $\left(\boldsymbol{X}_2^{\mathrm{T}}\right)^+$ are scaled versions of $\boldsymbol{X}_1^*$ and $\boldsymbol{X}_2^*$, respectively. Using equation (15.16) in (15.15) we can eliminate $\boldsymbol{H}_2$ in the following fashion

$$\begin{aligned} \tilde{\boldsymbol{F}}_2 \doteq \left(\boldsymbol{X}_1^{\mathrm{T}}\right)^+ \cdot \boldsymbol{F}_2 = \left(\boldsymbol{X}_1^{\mathrm{T}}\right)^+ \cdot \boldsymbol{X}^{\mathrm{T}} \cdot \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{G}_2 \cdot \boldsymbol{\Lambda}^{-1} = \boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_2 \cdot \boldsymbol{\Lambda}^{-1} \\ \Rightarrow \tilde{\boldsymbol{F}}_2 \cdot \boldsymbol{\Lambda} = \boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G}_2. \end{aligned} \tag{15.17}$$

In order to remove the unknown $\boldsymbol{H}_1^{\mathrm{T}}$ we need to solve (15.17) for $\boldsymbol{H}_1^{\mathrm{T}}$. This solution is only unique if $\boldsymbol{G}_2$ is a square or a flat matrix, i.e., $r_G \geq M_{\mathrm{R}}$. Also, to render this inversion numerically stable, $\boldsymbol{G}_2$ should have orthogonal rows.

Design Rule 4. *The rank of the tensor $\boldsymbol{\mathcal{G}}$ must satisfy $r_G \geq M_{\mathrm{R}}$. Also, from design rule 1, the number of training blocks N_{R} must be greater or equal to r_G. Therefore, to reduce the pilot overhead, r_G should be as small as possible. Consequently, we choose $r_G = M_{\mathrm{R}}$. Note that it follows that $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are square matrices.*

[2] A similar idea was used to solve a channel estimation problem for a one-way relaying scenario in [LV08].

Design Rule 5. *The two-mode factor matrix $\boldsymbol{G}_2$ must have full rank.*

Design Recommendation 2. *The two-mode factor matrix $\boldsymbol{G}_2$ should be an orthogonal matrix.*

Now we can solve (15.17) for $\boldsymbol{H}_1^{\mathrm{T}}$ and insert this solution into (15.14). We obtain

$$\boldsymbol{H}_1^{\mathrm{T}} = \tilde{\boldsymbol{F}}_2 \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{G}_2^{+} \Rightarrow \boldsymbol{F}_1 = \tilde{\boldsymbol{F}}_2 \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{G}_2^{+} \cdot \boldsymbol{G}_1 \cdot \boldsymbol{\Lambda} \tag{15.18}$$

$$\boldsymbol{F}_1 = \tilde{\boldsymbol{F}}_2 \cdot \left[(\boldsymbol{G}_2^{+} \cdot \boldsymbol{G}_1) \odot \left(\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}} \right) \right], \tag{15.19}$$

where in the last step we have used the fact that $\boldsymbol{\Lambda} = \operatorname{diag}\{\boldsymbol{\lambda}\}$ and property (3.21). In order to solve (15.19) for the unknown vector $\boldsymbol{\lambda}$, we have to isolate $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ on one side of the equation. However, to achieve this, we need to move $\tilde{\boldsymbol{F}}_2$ to the other side. Since $\tilde{\boldsymbol{F}}_2$ is of size $M_1 \times r_G$ this step requires $M_1 \geq r_G$. For the smallest possible r_G, which was chosen in design rule 4, this condition reduces to $M_1 \geq M_{\mathrm{R}}$. From the equivalent equation at the other user terminal, we also get the condition $M_2 \geq M_{\mathrm{R}}$. As a consequence, we now consider two cases separately. First of all, we solve the case where both conditions are met, i.e., $\min\{M_1, M_2\} \geq M_{\mathrm{R}}$. Then we consider the case where this condition is not true. Note that TENCE is only expected to outperform the LS-based compound channel estimator in case 1, as pointed out in the beginning of this section. The second case is only shown for completeness to demonstrate that the tensor-based approach can be used for arbitrary antenna configurations.

Case 1: $\min\{M_1, M_2\} \geq M_{\mathrm{R}}$

In this case, we can solve (15.19) directly for $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ in the following fashion

$$\begin{aligned} \tilde{\boldsymbol{F}}_2^{+} \cdot \boldsymbol{F}_1 &= \left(\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1 \right) \odot \left(\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}} \right) \\ \left(\tilde{\boldsymbol{F}}_2^{+} \cdot \boldsymbol{F}_1 \right) \oslash \left(\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1 \right) &= \boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}. \end{aligned} \tag{15.20}$$

Note that since we assume $r_G = M_{\mathrm{R}}$, the matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are square and hence the pseudo-inverse is replaced by the matrix inverse. Here we apply the inverse Schur product $\oslash$ (i.e., element-wise division), which requires that the matrix $\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1$ does not contain any zero entries. This leads to another design rule

Design Rule 6. *The factor matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ must be chosen such that the matrix $\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1$ does not contain any entries that are equal to zero or very close to zero.*

In the presence of noise, (15.20) holds only approximately. Therefore, the matrix estimated from (15.20) does not necessarily have rank one. In order to find the best approximation of $\boldsymbol{\lambda}$

we can proceed in a manner similar to the inversion of the Khatri-Rao product and additionally exploit the symmetry of the matrix. The algorithm to estimate $\boldsymbol{\lambda}$ is summarized in Alorithm 9.

Algorithm 9 [RH10c] Estimation of $\boldsymbol{\lambda}$ for TENCE

- Compute the matrix $\boldsymbol{L} = \left(\tilde{\boldsymbol{F}}_2^+ \cdot \boldsymbol{F}_1\right) \oslash \left(\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1\right)$.
- Force the matrix to be symmetric by computing $\tilde{\boldsymbol{L}} = \frac{1}{2}\left(\boldsymbol{L} + \boldsymbol{L}^{\mathrm{T}}\right)$.
- Since $\tilde{\boldsymbol{L}}$ is symmetric, an SVD of this matrix is given by $\tilde{\boldsymbol{L}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}$. An SVD of this form can for instance be computed via the Takagi factorization [Tak24].
- Then, the least squares estimate for $\boldsymbol{\lambda}$ is given by $\hat{\boldsymbol{\lambda}} = \sqrt{\sigma_1} \cdot \boldsymbol{u}_1$, where $\boldsymbol{u}_1$ represents the first column of $\boldsymbol{U}$ and σ_1 is the largest singular value of $\tilde{\boldsymbol{L}}$.

Note that the estimation of $\boldsymbol{\lambda}$ involves one sign ambiguity since $(-\boldsymbol{\lambda}) \cdot (-\boldsymbol{\lambda})^{\mathrm{T}} = \boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$.

From the estimate of $\boldsymbol{\lambda}$ we finally obtain estimates for the channel matrices with the help of (15.14) and (15.15)

$$\hat{\boldsymbol{H}}_1 = \left(\boldsymbol{F}_1 \cdot \operatorname{diag}\left\{\hat{\boldsymbol{\lambda}}\right\}^{-1} \cdot \boldsymbol{G}_1^{-1}\right)^{\mathrm{T}} \tag{15.21}$$

$$\hat{\boldsymbol{H}}_2 = \left(\left(\boldsymbol{X}_2^{\mathrm{T}}\right)^+ \boldsymbol{F}_2 \cdot \operatorname{diag}\left\{\hat{\boldsymbol{\lambda}}\right\} \cdot \boldsymbol{G}_2^{-1}\right)^{\mathrm{T}}. \tag{15.22}$$

It is also possible to obtain a second estimate for $\boldsymbol{H}_1$ from $\boldsymbol{F}_2$ by replacing $\boldsymbol{X}_2$ by $\boldsymbol{X}_1$ in (15.22). However, since the estimate found from (15.21) is always more accurate, this additional estimate for $\boldsymbol{H}_1$ will not be used in the simulations. Note that (15.21) involves the inverse of $\boldsymbol{G}_1$. With the same reasoning as before, we therefore propose the corresponding design rule for $\boldsymbol{G}_1$:

Design Rule 7. *The 1-mode factor matrix $\boldsymbol{G}_1$ must have full rank.*

Design Recommendation 3. *The 1-mode factor matrix $\boldsymbol{G}_1$ should be an orthogonal matrix. Note that from design rule 4 it follows that $\boldsymbol{G}_1$ is a square matrix.*

Note that the sign ambiguity in $\boldsymbol{\lambda}$ leads to one sign ambiguity in the channel estimates: instead of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ we may estimate $-\boldsymbol{H}_1$ and $-\boldsymbol{H}_2$. However, since this sign cancels in the transmission equations (14.6), this scaling ambiguity is irrelevant. This concludes the channel estimation algorithm for Case 1.

Case 2: $1 < \min\{M_1, M_2\} < M_{\mathrm{R}}$

Without loss of generality, we consider the case where $M_1 \leq M_2$. Since $\tilde{\boldsymbol{F}}_2$ in (15.19) is a "flat" matrix, we cannot solve (15.19) for the unknown matrix $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ directly. Essentially, there are only $M_1 \cdot M_{\mathrm{R}}$ equations for M_{R}^2 unknowns. However, it is actually not required to estimate all elements in $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$, because this matrix has rank one and hence does not have M_{R}^2 degrees of freedom. It is not difficult to see that already $2M_{\mathrm{R}} - 1$ elements from $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ are enough to reconstruct the entire matrix via the following naive approach: the M_{R} main diagonal elements of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ are equal to λ_m^2 from which we can obtain all λ_m up to one $\pm$ ambiguity per coefficient. These unknown signs can be estimated from the $M_{\mathrm{R}} - 1$ elements on the first off-diagonal of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$.

The approach we take to solve this case is to reduce the number of variables we estimate from M_{R}^2 to $M_1 \cdot M_{\mathrm{R}}$ via a suitable design of the tensor $\boldsymbol{\mathcal{G}}$ which then facilitates a well-defined inversion. From the $M_1 \cdot M_{\mathrm{R}}$ estimated elements in $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ we can reconstruct the missing elements using the rank-1 structure (cf., Algorithm 10) and then proceed in the same manner as in the previous case.

To simplify the notation, we introduce the following definitions

$$\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1 = \tilde{\boldsymbol{G}} = \begin{bmatrix} \tilde{\boldsymbol{g}}_1 & \tilde{\boldsymbol{g}}_2 & \dots & \tilde{\boldsymbol{g}}_{M_{\mathrm{R}}} \end{bmatrix}, \tag{15.23}$$

$$\boldsymbol{F}_1 = \begin{bmatrix} \boldsymbol{f}_{1,1} & \boldsymbol{f}_{1,2} & \dots & \boldsymbol{f}_{1,M_{\mathrm{R}}} \end{bmatrix}, \tag{15.24}$$

i.e., $\tilde{\boldsymbol{g}}_m$ and $\boldsymbol{f}_{1,m}$ represent the m-th columns of $\tilde{\boldsymbol{G}}$ and $\boldsymbol{F}_1$, respectively. Note that we have again used the assumption $r_G = M_{\mathrm{R}}$. In order to proceed, we need the following lemma:

Lemma 15.3.1. [RH10c] *For arbitrary matrices $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{B} \in \mathbb{C}^{N \times P}$, and $\boldsymbol{C} \in \mathbb{C}^{N \times P}$ we can define a matrix $\boldsymbol{D} \in \mathbb{C}^{M \times P}$ as $\boldsymbol{D} = \boldsymbol{A} \cdot (\boldsymbol{B} \odot \boldsymbol{C})$. Then, the p-th column of $\boldsymbol{D}$ can be expressed as*

$$\boldsymbol{d}_p = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{b}_p\} \cdot \boldsymbol{c}_p = \boldsymbol{A} \cdot \operatorname{diag}\{\boldsymbol{c}_p\} \cdot \boldsymbol{b}_p, \tag{15.25}$$

where $\boldsymbol{b}_p$ and $\boldsymbol{c}_p$ represent the p-th column vectors of $\boldsymbol{B}$ and $\boldsymbol{C}$, respectively and $p = 1, 2, \dots, P$.

Proof. Obviously, for arbitrary vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{C}^N$ we have that $\boldsymbol{x} \odot \boldsymbol{y} = \operatorname{diag}\{\boldsymbol{x}\} \cdot \boldsymbol{y} = \operatorname{diag}\{\boldsymbol{y}\} \cdot \boldsymbol{x}$. Moreover, the p-th column of $\boldsymbol{D}$ is given by $\boldsymbol{d}_p = \boldsymbol{A} \cdot (\boldsymbol{b}_p \odot \boldsymbol{c}_p)$. Combining these two identities for $\boldsymbol{x} = \boldsymbol{b}_p$ and $\boldsymbol{y} = \boldsymbol{c}_p$ proves the lemma. □

Using this lemma and the definitions (15.23) and (15.24), we rewrite the matrix equation

(15.19) into a system of matrix-vector equations

$$\boldsymbol{f}_{1,m} = \tilde{\boldsymbol{F}}_2 \cdot \operatorname{diag}\{\tilde{\boldsymbol{g}}_m\} \cdot \boldsymbol{\lambda} \cdot \lambda_m, \; m = 1, 2, \ldots, M_{\mathrm{R}} \tag{15.26}$$

Note that if we set the k-th element of the vector $\tilde{\boldsymbol{g}}_m$ to zero, the k-th column of the matrix $\tilde{\boldsymbol{F}}_2 \cdot \operatorname{diag}\{\tilde{\boldsymbol{g}}_m\}$ becomes zero. This is equivalent to removing the k-th column of $\tilde{\boldsymbol{F}}_2$ and the k-th row of the parameter vector $\boldsymbol{\lambda} \cdot \lambda_m$ in the m-th matrix vector equation of (15.26). Consequently, we can reduce the number of variables in each of the matrix-vector equations from M_{R} to M_1 if we place $M_{\mathrm{R}} - M_1$ zeros in each of the vectors $\tilde{\boldsymbol{g}}_m$. This leads to the crucial design rule for the second case:

Design Rule 8. *The 1-mode and 2-mode factor matrices of the tensor $\boldsymbol{\mathcal{G}}$ must be designed in such a way that each column of the matrix $\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1 \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}}}$ contains at most* $\min\{M_1, M_2\}$ *non-zero entries.*

Note that design rule 8 does not contradict design rule 6 since for the first case we have $\min\{M_1, M_2\} \geq M_{\mathrm{R}}$ and hence all elements are allowed to be non-zero by rule 8 (and are forced to be nonzero by rule 6).

Using this design we can solve all matrix-vector equations in (15.26) and hence obtain M_1 entries of each column of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$. The elements we obtain are exactly the non-zero positions in the matrix $\tilde{\boldsymbol{G}}$. From these elements we can reconstruct an estimate of the full matrix $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$, provided that $M_1 > 1$.[3] This reconstruction algorithm is summarized in Algorithm 10.

At the end of this algorithm we have an estimate of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$. Depending on the pattern of the unknown elements, this estimate may not be exactly symmetric and it may also not be exactly rank one. We therefore proceed in the same manner as in Case 1 to estimate the vector $\boldsymbol{\lambda}$ from this matrix: First the matrix is forced to be symmetric. After that, a best rank-one approximation is computed with the help of a singular value decomposition (cf. Algorithm 9). The estimated vector $\hat{\boldsymbol{\lambda}}$ is then used to compute estimates for the channel matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ (cf. equations (15.21) and (15.22)).

15.3.3. Summary

The TENCE algorithm is summarized in Algorithm 11. Concerning the design rules for the matrix $\boldsymbol{X}$ and the tensor $\boldsymbol{\mathcal{G}}$ we have the following:

[3] Following the proposed design of $\boldsymbol{\mathcal{G}}$, for $M_1 = 1$ we only obtain the main diagonal of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$, i.e., $\lambda_i^2, \forall i$. Therefore, we cannot determine the sign of the individual λ_i in this case. However, $M_1 > 1$ has been explicitly assumed, and the case $M_1 = 1$ is further discussed in Section 15.5.

Algorithm 10 [RH10c] Rank-one matrix reconstruction

- The input to the algorithm is a matrix $\boldsymbol{L}$ which contains the estimates of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ we have and the pattern of non-zero elements in the matrix $\tilde{\boldsymbol{G}}$. The non-zero positions in $\tilde{\boldsymbol{G}}$ are the known elements in the estimate of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$.
- First of all, we can use the symmetry of $\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}^{\mathrm{T}}$ by filling each unknown element $l_{i,j}$ with $l_{j,i}$ if the latter is known.
- If after this step there are unknown elements left. we continue by estimating the ratios $\rho_m \doteq \lambda_m / \lambda_{m-1}$ for $m = 2, 3, \ldots, M_{\mathrm{R}}$ in the following fashion:
  1. Set $m = 2$.
  2. Obtain the set of column indices $i \in \mathcal{I}$ for which the elements (m, i) and $(m-1, i)$ are known.
  3. Obtain the set of row indices $j \in \mathcal{J}$ for which the elements (j, m) and $(j, m-1)$ are known.
  4. Estimate ρ_m as the arithmetic average of the ratios $l_{m,i}/l_{m-1,i}$ and the ratios $l_{j,m}/l_{j,m-1}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}$.
  5. If $m < M_{\mathrm{R}}$ set $m = m + 1$ and go to 2).
- Now we can apply these ratios to fill the rest of the matrix. For every unknown element (i, j) in the matrix $\boldsymbol{L}$, we check:
  1. If the element $(i, j-1)$ is known, an estimate of $l_{i,j}$ is given by $l_{i,j-1} \cdot \rho_m$.
  2. If the element $(i-1, j)$ is known, an estimate of $l_{i,j}$ is given by $l_{i-1,j} \cdot \rho_m$.
  3. If the element $(i, j+1)$ is known, an estimate of $l_{i,j}$ is given by $l_{i,j+1}/\rho_m$.
  4. If the element $(i+1, j)$ is known, an estimate of $l_{i,j}$ is given by $l_{i+1,j}/\rho_m$.
- Again, if more than one estimate for $l_{i,j}$ is available, an arithmetic average is computed.

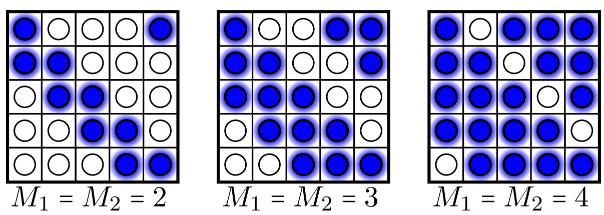


Figure 15.3.: Structure of the matrix $\boldsymbol{S}$ for $M_{\mathrm{R}} = 5$ and different values for $\min\{M_1, M_2\}$. Empty circles represent zeros, filled circles represent ones.

Algorithm 11 [RH10c] Summary of the TENCE algorithm at UT_1. For UT_2 we replace $\boldsymbol{\mathcal{Y}}_1$ by $\boldsymbol{\mathcal{Y}}_2$ in the first step and $\boldsymbol{X}_1$ by $\boldsymbol{X}_2$ in the third step. Moreover, in the final result (15.21) and (15.22) we exchange $\hat{\boldsymbol{H}}_1$ and $\hat{\boldsymbol{H}}_2$ and replace $\boldsymbol{X}_1$ by $\boldsymbol{X}_2$.

- Compute the matrix $\left(\boldsymbol{G}_3^+ \cdot [\boldsymbol{\mathcal{Y}}_1]_{(3)}\right)^{\mathrm{T}}$, cf. (15.12).
- Factorize it into the two Khatri-Rao factor matrices $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ using Algorithm 1.
- Compute $\tilde{\boldsymbol{F}}_2 = \left(\boldsymbol{X}_1^{\mathrm{T}}\right)^+ \cdot \boldsymbol{F}_2$.
- If $\min\{M_1, M_2\} \geq M_{\mathrm{R}}$: compute $\boldsymbol{L} = \left(\tilde{\boldsymbol{F}}_2^+ \cdot \boldsymbol{F}_1\right) \oslash \left(\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1\right)$.
- If $1 < \min\{M_1, M_2\} < M_{\mathrm{R}}$:
  - Let $\boldsymbol{f}_{1,m}$ be the m-th column of $\boldsymbol{F}_1$ and $\tilde{\boldsymbol{g}}_m$ the m-th column of $\tilde{\boldsymbol{G}} = \boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1$.
  - Compute $\boldsymbol{l}_m = \left[\tilde{\boldsymbol{F}}_2 \cdot \operatorname{diag}\{\tilde{\boldsymbol{g}}_m\}\right]^+ \cdot \boldsymbol{f}_{1,m}$ for $m = 1, 2, \ldots, M_{\mathrm{R}}$.
  - Collect the vectors $\boldsymbol{l}_m$ column-wise into the matrix $\boldsymbol{L}$.
  - Use Algorithm 10 to fill the elements in $\boldsymbol{L}$ which have not been estimated.
- Estimate $\boldsymbol{\lambda}$ as the best symmetric rank-one approximation of $\boldsymbol{L}$ using Algorithm 9.
- Compute the final channel estimates using equations (15.21) and (15.22).

- The pilot matrix $\boldsymbol{X} \in \mathbb{C}^{M_1+M_2 \times N_\mathrm{P}}$: The number of pilots N_P must satisfy $N_\mathrm{P} \geq M_1 + M_2$ and $\boldsymbol{X}$ must have orthogonal rows (cf. design rules 2 and 3). A reasonable choice is given by constructing a DFT matrix (cf. Appendix A.2) of size $N_\mathrm{P} \times N_\mathrm{P}$ and then using the first M_1 rows for $\boldsymbol{X}_1$ and the next M_2 rows for $\boldsymbol{X}_2$. To ensure that the transmit power is limited to $P_{\mathrm{T},i}$ for each user terminal $i = 1, 2$, $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ can be scaled individually, such that the norm of each column is equal to $P_{\mathrm{T},i}$. Note that $N_\mathrm{P} = M_1 + M_2$ is sufficient for the training, higher values can be used to increase the estimation accuracy in the presence of noise. Another possible choice is given by Zadoff-Chu sequences [Chu72] since these fulfill the required orthogonality conditions as well.

- The relay amplification tensor $\boldsymbol{\mathcal{G}}$:
  - The rank r_G must satisfy $r_G \geq M_\mathrm{R}$ according to design rule 4. A larger rank leads to higher pilot overhead according to design rule 1. Therefore we choose $r_G = M_\mathrm{R}$.
  - The factor matrices $\boldsymbol{G}_1 \in \mathbb{C}^{M_\mathrm{R} \times M_\mathrm{R}}$, $\boldsymbol{G}_2 \in \mathbb{C}^{M_\mathrm{R} \times M_\mathrm{R}}$, and $\boldsymbol{G}_3 \in \mathbb{C}^{N_\mathrm{R} \times M_\mathrm{R}}$ must have full rank (M_R) according to design rules 1, 5, and 7. Moreover, N_R must satisfy $N_\mathrm{R} \geq M_\mathrm{R}$ according to the design rules 1 and 4. Note that $N_\mathrm{R} = M_\mathrm{R}$ is sufficient for the training, higher values can be used to increase the estimation accuracy in the presence of noise.
  - The matrix $\boldsymbol{G}_2^{-1} \cdot \boldsymbol{G}_1$ must have $\min\{M_1, M_2\}$ nonzero elements per column according to rules 6 and 8. Note that this implies that this matrix should not have any zero entries if $\min\{M_1, M_2\} \geq M_\mathrm{R}$.
  - The factor matrix $\boldsymbol{G}_3 \in \mathbb{C}^{N_\mathrm{R} \times M_\mathrm{R}}$ should have orthogonal columns and the factor matrices $\boldsymbol{G}_1, \boldsymbol{G}_2 \in \mathbb{C}^{M_\mathrm{R} \times M_\mathrm{R}}$ should be orthogonal according to recommendations 1, 2, and 3.

The total number of pilots is equal to $N_\mathrm{P} \cdot N_\mathrm{R}$. Following the design rules we conclude that at least $(M_1 + M_2) \cdot M_\mathrm{R}$ pilots are needed. Note that the total number of parameters that must be identified is equal to $M_1 \cdot M_\mathrm{R}$ in $\boldsymbol{H}_1$ and $M_2 \cdot M_\mathrm{R}$ in $\boldsymbol{H}_2$. Therefore, the total number of required pilots is equal to the total number of parameters that are identified. Note that this does not correspond to the minimum possible pilot overhead since the number of observations is indeed larger (by a factor of M_i at terminal i). To conclude this chapter we give an example how a tensor $\boldsymbol{\mathcal{G}}$ can easily be constructed that follows all the design rules.

- Choose $\boldsymbol{G}_2 = \boldsymbol{I}_{M_\mathrm{R}}$.

- Set $\boldsymbol{G}_3$ to a $M_\mathrm{R} \times M_\mathrm{R}$ DFT matrix (cf. Appendix A.2). If a larger number of training blocks (frames) is desired, use a $N_\mathrm{R} \times N_\mathrm{R}$ DFT matrix and truncate it to M_R columns.

- Then, compute $\boldsymbol{G}_1$ in the following way: If $\min\{M_1, M_2\} \geq M_\mathrm{R}$: Set $\boldsymbol{G}_1 = \boldsymbol{D}_{M_\mathrm{R}}$, where $\boldsymbol{D}_{M_\mathrm{R}}$ is an $M_\mathrm{R} \times M_\mathrm{R}$ DFT matrix. Otherwise set $\boldsymbol{G}_1 = \boldsymbol{D}_{M_\mathrm{R}} \odot \boldsymbol{S}$, where $\boldsymbol{S}$ is a circulant matrix computed from the vector $\boldsymbol{v} = \left[\boldsymbol{1}^\mathrm{T}_{\min\{M_1,M_2\}\times 1}, \boldsymbol{0}^\mathrm{T}_{M_\mathrm{R}-\min\{M_1,M_2\}\times 1}\right]^\mathrm{T}$. That means that the n-th column of $\boldsymbol{S}$ is equal to $\boldsymbol{v}$ shifted by $n-1$ elements in a cyclic manner. To illustrate the structure of $\boldsymbol{S}$, Figure 15.3 displays $\boldsymbol{S}$ for $M_\mathrm{R} = 5$ and three different values for $\min\{M_1, M_2\}$. We have verified numerically that this design provides a full rank matrix $\boldsymbol{G}_1$ for all combinations of M_R, M_1, and M_2 up to $M_\mathrm{R} = 50$.

Note that this design of $\boldsymbol{\mathcal{G}}$ also fulfills all design recommendations if $\min\{M_1, M_2\} \geq M_\mathrm{R}$. Otherwise, $\boldsymbol{G}_1$ is not necessarily orthogonal which violates the design recommendation 3.

The amplification matrix $\boldsymbol{G}^{(i)}$ which the relay uses in the i-th frame can be computed from the matrices $\boldsymbol{G}_1$, $\boldsymbol{G}_2$, and $\boldsymbol{G}_3$ in the following fashion

$$\boldsymbol{G}^{(i)} = c_i \cdot \boldsymbol{G}_1 \cdot \mathrm{diag}\left\{[\boldsymbol{G}_3]_{i,:}\right\} \cdot \boldsymbol{G}_2^\mathrm{T},\ i = 1, 2, \ldots, N_\mathrm{R},$$

where $[\boldsymbol{G}_3]_{i,:}$ represents the i-th row of $\boldsymbol{G}_3$ and c_i is chosen such that $\left\|\boldsymbol{G}^{(i)}\right\|_\mathrm{F} = 1$. Therefore, if $\min\{M_1, M_2\} \geq M_\mathrm{R}$, the relay uses shifted DFT matrices during the training phase.

15.4. Iterative refinement for TENCE

The TENCE algorithm which we have derived in the previous section is a purely algebraic closed-form solution [RH09e, RH10c]. Therefore, it is very fast, since it does not require any iterative procedures. However it does not provide the MMSE solution. In this section we show that the MSE can be further reduced by an iterative procedure [RH09d, RH10c]. The mathematical manipulations that are used for this derivation are similar to Structured Least Squares (SLS) [Haa97b] even though the underlying problem that is solved in [Haa97b] is different (cf. Section 11.7).

As in the previous section we derive the solution for UT_1. Due to the strong symmetries in the data model, the solution for UT_2 is very similar.

Let the initial estimates for the channel matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ be given by $\hat{\boldsymbol{H}}_1$ and $\hat{\boldsymbol{H}}_2$ and define $\hat{\boldsymbol{H}} = \begin{bmatrix}\hat{\boldsymbol{H}}_1 & \hat{\boldsymbol{H}}_2\end{bmatrix}$. Our goal is to improve the estimates $\hat{\boldsymbol{H}}_1$ and $\hat{\boldsymbol{H}}_2$ based on the received training data. Therefore we need to define a measure for the quality of the channel estimates. To this end, introduce the following definition

$$\tilde{\boldsymbol{\mathcal{Y}}}_1 = \boldsymbol{\mathcal{Y}}_1 \times_2 \left(\boldsymbol{X}^\mathrm{T}\right)^+. \tag{15.27}$$

Note that if $\boldsymbol{X}$ is chosen to have orthogonal rows as proposed in the previous section, $\left(\boldsymbol{X}^\mathrm{T}\right)^+$

is a scaled version of $\boldsymbol{X}^*$. Inserting (15.27) into (15.9) we find that in the absence of noise $\tilde{\boldsymbol{\mathcal{Y}}}_1$ has the following structure

$$\tilde{\boldsymbol{\mathcal{Y}}}_1 = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{H}_1^{\mathrm{T}} \times_2 \boldsymbol{H}^{\mathrm{T}}. \tag{15.28}$$

As we can see, the channel matrix $\boldsymbol{H}_1$ is present in the first and in the second factor. For TENCE, we exploit this symmetry only in the second step, i.e., to estimate $\boldsymbol{\Lambda}$. In the first step of TENCE this is not considered since for the inversion of the Khatri-Rao product, $\boldsymbol{H}_1$ is eliminated in the second factor. This is the reason that the estimate obtained by TENCE can still be improved by exploiting the structure of $\tilde{\boldsymbol{\mathcal{Y}}}_1$.

In the presence of noise, (15.28) holds only approximately. We can therefore judge the quality of the channel estimate via the norm of the residual tensor $\tilde{\boldsymbol{\mathcal{Y}}}_1 - \boldsymbol{\mathcal{G}} \times_1 \hat{\boldsymbol{H}}_1^{\mathrm{T}} \times_2 \hat{\boldsymbol{H}}^{\mathrm{T}}$. In order to minimize this norm we introduce update terms $\Delta\boldsymbol{H}_1$ and $\Delta\boldsymbol{H}_2$ for the channel estimates $\hat{\boldsymbol{H}}_1$ and $\hat{\boldsymbol{H}}_2$, respectively. Since we already have an initial estimate we additionally apply regularization to enhance the numerical stability. This ensures that the update terms are small compared to the initial solution. Similar to the approach taken for SLS in [Haa97b], the overall cost function we minimize can be written in the following way[4]

$$J(\Delta\boldsymbol{H}_k) = \|\boldsymbol{\mathcal{R}}_k\|_{\mathrm{H}}^2 + \kappa_1^2 \left\|\Delta\boldsymbol{H}_{1,k}\right\|_{\mathrm{F}}^2 + \kappa_2^2 \left\|\Delta\boldsymbol{H}_{2,k}\right\|_{\mathrm{F}}^2, \tag{15.29}$$

where $\boldsymbol{\mathcal{R}}_k$ is the residual tensor after the k-th iteration which is given by

$$\boldsymbol{\mathcal{R}}_k = \tilde{\boldsymbol{\mathcal{Y}}}_1 - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta\boldsymbol{H}_{1,k}\right)^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta\boldsymbol{H}_k\right)^{\mathrm{T}}. \tag{15.30}$$

Here, $\Delta\boldsymbol{H}_{1,k}$ and $\Delta\boldsymbol{H}_{2,k}$ represent the updates after the k-th iteration and $\Delta\boldsymbol{H}_k = [\Delta\boldsymbol{H}_{1,k},\ \Delta\boldsymbol{H}_{2,k}]$. Moreover, the terms κ_1 and κ_2 in (15.29) are given by $\kappa_1 = \sqrt{M_1/\alpha}$ and $\kappa_2 = \sqrt{M_2/\alpha}$ where $\alpha \in \mathbb{R}, \alpha > 0$ controls the amount of regularization used (the larger α, the less regularization)[5].

The cost function (15.29) represents a quadratic Least Squares problem that does not have a

[4] This cost function ignores the fact that the noise is not white due to the forwarded relay noise. Since an initial estimate of the channel matrices is already available via TENCE, the cost function can be extended to take the noise correlation into account. This is achieved by replacing $\|\boldsymbol{\mathcal{R}}_k\|_{\mathrm{H}}$ in the cost function by $\mathrm{vec}\{\boldsymbol{\mathcal{R}}_k\}^{\mathrm{H}} \cdot \hat{\boldsymbol{\Gamma}}^{-1} \cdot \mathrm{vec}\{\boldsymbol{\mathcal{R}}_k\}$, where $\hat{\boldsymbol{\Gamma}}$ is an estimate of the noise covariance matrix. However, in simulations we have found no significant improvement of the modified iterative scheme in terms of the channel estimation accuracy. Since this modification significantly complicates the presentation of the algorithm, it is omitted here for clarity.

[5] Our simulations have shown that the performance is not very sensitive to the choice of the regularization parameter α. For a low SNR, a moderate amount of regularization ($\alpha \approx 100$) enhances the numerical stability, but α should not be chosen too small. Moreover, for a high SNR, regularization is not needed and we can choose $\alpha = \infty$. If not stated otherwise, we use $\alpha = 100$ for all the simulations.

closed-form solution. However, it can be solved iteratively by linearizing it locally and solving a sequence of linear Least Squares problems [Haa97b]. In each iteration, the terms $\Delta\boldsymbol{H}_{1,k}$ and $\Delta\boldsymbol{H}_{2,k}$ are updated according to the following rules

$$\Delta\boldsymbol{H}_{1,k+1} = \Delta\boldsymbol{H}_{1,k} + \Delta\Delta\boldsymbol{H}_{1,k} \tag{15.31}$$

$$\Delta\boldsymbol{H}_{2,k+1} = \Delta\boldsymbol{H}_{2,k} + \Delta\Delta\boldsymbol{H}_{2,k}, \tag{15.32}$$

where the initial values are given by

$$\Delta\boldsymbol{H}_{1,k=0} = \boldsymbol{0}_{M_\mathrm{R}\times M_1} \quad \text{and} \quad \Delta\boldsymbol{H}_{2,k=0} = \boldsymbol{0}_{M_\mathrm{R}\times M_2}. \tag{15.33}$$

Our goal is to find $\Delta\Delta\boldsymbol{H}_{1,k}$ and $\Delta\Delta\boldsymbol{H}_{2,k}$ that minimize the cost function in the k-th iteration. As shown in Appendix E.1, these updates can be found via

$$\begin{bmatrix} \mathrm{vec}\left\{\Delta\Delta\boldsymbol{H}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix} = -\begin{bmatrix} -\boldsymbol{F}_k^{(1)} & -\boldsymbol{F}_k^{(2)} \\ \kappa_1 \cdot \boldsymbol{I}_{M_1\cdot M_\mathrm{R}} & \boldsymbol{0}_{M_1\cdot M_\mathrm{R}\times M_2\cdot M_\mathrm{R}} \\ \boldsymbol{0}_{M_2\cdot M_\mathrm{R}\times M_1\cdot M_\mathrm{R}} & \kappa_2\cdot\boldsymbol{I}_{M_2\cdot M_\mathrm{R}} \end{bmatrix}^+ \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\} \\ \kappa_1\cdot\mathrm{vec}\left\{\Delta\boldsymbol{H}_{1,k}\right\} \\ \kappa_2\cdot\mathrm{vec}\left\{\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix}, \tag{15.34}$$

where the matrices $\boldsymbol{F}_k^{(1)}$ and $\boldsymbol{F}_k^{(2)}$ are given in (E.10) and (E.11), respectively.

The SLS-based iterative refinement proceeds by computing the updates according to (15.34) and applying these updates as shown in (15.31) and (15.32). Different criteria can be used to check whether the iterative procedure has converged. For example, we can compute the norm of the update terms $\Delta\Delta\boldsymbol{H}_{1,k}$ and $\Delta\Delta\boldsymbol{H}_{2,k}$ and terminate the algorithm when this norm drops below a predefined threshold. Alternatively, define the quantity $r_k = \|\boldsymbol{\mathcal{R}}_k\|_\mathrm{H}$, which is a measure of the fit of the current channel estimates to the data received during the training phase. Then we can terminate the iteration if $r_k - r_{k+1} < \delta$ for a predefined threshold[6] $\delta > 0$. Moreover, if $r_{k+1} > r_k$, the $(k+1)$-th iteration is ignored and the k-th iteration is used as a final solution. The SLS-based refinement of TENCE is summarized in Algorithm 12.

[6] The threshold parameter δ represents a trade-off between computational complexity and estimation accuracy. We observed that $\delta = 10^{-3}$ is a reasonable value. Smaller values lead to more iterations, however these do not result in a significant improvement in accuracy. Larger values of δ terminate the algorithm too early. As we show in the simulations, for this choice of δ the number of iterations is between one and four, even in critical scenarios.

Algorithm 12 [RH10c] Summary of the SLS-based iterative refinement for TENCE at UT_1. For UT_2 we consistently exchange $\boldsymbol{H}_1$ with $\boldsymbol{H}_2$ and replace $\boldsymbol{\mathcal{Y}}_1$ by $\boldsymbol{\mathcal{Y}}_2$ in equations (15.30), (E.12), and (15.34).

1. Initialize $\hat{\boldsymbol{H}}_1$ and $\hat{\boldsymbol{H}}_2$ with the estimates obtained via TENCE.
2. Set $\Delta\boldsymbol{H}_{1,k=0} = \mathbf{0}_{M_{\mathrm{R}}\times M_1}$, $\Delta\boldsymbol{H}_{2,k=0} = \mathbf{0}_{M_{\mathrm{R}}\times M_2}$, and $k = 0$.
3. Compute $\tilde{\boldsymbol{\mathcal{Y}}}_1$ from the received data during the transmission phase according to $\tilde{\boldsymbol{\mathcal{Y}}}_1 = \boldsymbol{\mathcal{Y}}_1 \times_2 \left(\boldsymbol{X}^{\mathrm{T}}\right)^+$.
4. Calculate the residual tensor $\boldsymbol{\mathcal{R}}_k$ as shown in (15.30) and the matrices $\boldsymbol{F}_k^{(1)}$ and $\boldsymbol{F}_k^{(2)}$ from (E.10) and (E.11), respetively.
5. Solve the least squares problem in $\Delta\Delta\boldsymbol{H}_{1,k}$ and $\Delta\Delta\boldsymbol{H}_{2,k}$ according to (15.34).
6. Apply the updates to obtain $\Delta\boldsymbol{H}_{1,k+1} = \Delta\boldsymbol{H}_{1,k} + \Delta\Delta\boldsymbol{H}_{1,k}$ and $\Delta\boldsymbol{H}_{2,k+1} = \Delta\boldsymbol{H}_{2,k} + \Delta\Delta\boldsymbol{H}_{2,k}$.
7. Compute $r_k = \|\boldsymbol{\mathcal{R}}_k\|_{\mathrm{H}}$. If $k > 1$ test whether the last iteration has resulted in a significant innovation by comparing $r_k - r_{k+1}$ with the threshold δ. If the innovation is greater than δ set $k = k + 1$ and go to step 4. If the innovation is negative, ignore the update from the k-th iteration.
8. The improved channel estimates are given by $\hat{\boldsymbol{H}}_1 + \Delta\boldsymbol{H}_{1,k+1}$ and $\hat{\boldsymbol{H}}_2 + \Delta\boldsymbol{H}_{2,k+1}$.

15.5. Discussion

15.5.1. Computational complexity

The LS-based channel estimation scheme presented in Section 15.2 requires solving an overdetermined set of $M_i \cdot N_\mathrm{P}$ equations for $M_i \cdot (M_1 + M_2)$ unknowns, where $N_\mathrm{P} \geq M_1 + M_2$.

In TENCE, since most matrices that have to be inverted are chosen orthogonal, the only explicit matrix inversion we require is the pseudo-inverse of $\tilde{\boldsymbol{F}}_2$ which is of size $M_i \times M_\mathrm{R}$ for $i = 1, 2$. Therefore, the complexity is dominated by the Least-Squares Khatri-Rao factorization of a matrix of size $M_i \cdot N_\mathrm{P} \times M_\mathrm{R}$ for which M_R SVDs of size $N_\mathrm{P} \times M_i$ are required (NB: for each SVD, only the dominant singular vectors are needed). On the other hand, for the SLS-based refinement, an overdetermined set of $M_\mathrm{R} \cdot (M_i + 1) \cdot (M_1 + M_2)$ linear equations needs to be solved for $M_\mathrm{R} \cdot (M_1 + M_2)$ variables in each iteration. For $\alpha = \infty$ the number of equations reduces to $M_\mathrm{R} \cdot M_i \cdot (M_1 + M_2)$.

15.5.2. Non-orthogonal pilots

The way the derivation of TENCE is presented, we rely on the fact that the pilot matrix $\boldsymbol{X}$ has orthogonal rows (cf. design rule 3). This condition can be relaxed to allow a non-orthogonal $\boldsymbol{X}$ by replacing the pseudo-inverse of $\boldsymbol{X}_i^\mathrm{T}$ used at various steps of the derivation by a block of the pseudo-inverse of $\boldsymbol{X}$. However, such a choice for $\boldsymbol{X}$ is detrimental in terms of the channel estimation accuracy, as the simulation results in [GZL09b] have also verified. For a more profound discussion of the suboptimality of non-orthogonal pilots, see also [BG06].

15.5.3. Single-antenna case

Since previous channel estimation scheme for two-way relaying with AF relays focus on the single-antenna case [GZL09b], we briefly discuss this special case here. For $M_1 = M_2 = M_\mathrm{R} = 1$ the smallest pilot overhead is achieved by choosing $N_\mathrm{R} = 1$ and $N_\mathrm{P} = 2$. The relay amplification tensor $\boldsymbol{\mathcal{G}}$ becomes a scalar g and therefore the factor matrices are trivially $\boldsymbol{G}_1 = \boldsymbol{G}_2 = 1$ and $\boldsymbol{G}_3 = g$. Then, TENCE simplifies into the following algebraic equations for $\hat{h}_1$ and $\hat{h}_2$ estimated at UT_1

$$\hat{h}_1 = \sqrt{\frac{\boldsymbol{x}_1^* \cdot \boldsymbol{y}_1^\mathrm{T}}{\|\boldsymbol{x}_1\|_2 \cdot g}}, \quad \hat{h}_2 = \frac{\boldsymbol{x}_2^* \cdot \boldsymbol{y}_1^\mathrm{T}}{\|\boldsymbol{x}_2\|_2 \cdot g \cdot \hat{h}_1}, \tag{15.35}$$

where $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{C}^{1 \times N_\mathrm{P}}$ are the pilot sequences and $\boldsymbol{y}_1 \in \mathbb{C}^{1 \times N_\mathrm{P}}$ is the received training data. Here, $\sqrt{x}$ denotes the principle square-root, i.e., for $x = |x| \cdot \mathrm{e}^{\jmath \arg\{x\}} \in \mathbb{C}$ we have $\sqrt{x} = \sqrt{|x|} \cdot \mathrm{e}^{\jmath \frac{\arg\{x\}}{2}}$.

We compare the channel estimation accuracy of TENCE in this special case with the ML and LMMSE estimators from [GZL09b] in the simulations section. Note that the SLS-based refinement does not provide any improvement in the single-antenna case. Also note that we cannot replace the TENCE algorithm in the general MIMO case by a sequential application of the SISO case presented here. The reason is that each estimate is only unique up to one sign ambiguity which would leave the estimates of the channel matrices with one sign ambiguity per element. These ambiguities alter the subspace which renders SVD-based pre-/postprocessing infeasible.

For the case $\min\{M_1, M_2\} = 1$ and $M_{\mathrm{R}} > 1$, TENCE is not applicable. However, this case is actually not very relevant for the channel estimation schemes discussed in this chapter. To see why, let us first of all consider the case $M_1 = M_2 = 1$. In this setup, the effective channel matrices that are used to convey the transmitted signals $h_{i,j}^{(\mathrm{e})} = \boldsymbol{h}_i^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{h}_j$ are scalars for all $i = 1, 2$ and $j = 1, 2$. Consequently, providing both terminals with channel knowledge of $\boldsymbol{h}_1 \in \mathbb{C}^{M_{\mathrm{R}} \times 1}$ and $\boldsymbol{h}_2 \in \mathbb{C}^{M_{\mathrm{R}} \times 1}$ would be an unnecessary overhead. Only the relay requires channel knowledge to compute a suitable matrix $\boldsymbol{G}$ which is obtained from one set of $N_{\mathrm{P}} \geq 2$ pilots as shown in 15.3. Next, the scalar channel taps $h_{i,j}^{(\mathrm{e})}$ are estimated directly by transmitting a set of $N_{\mathrm{P,D}} \geq 2$ "dedicated" pilots (i.e., keeping $\boldsymbol{G}$ fixed, cf. Section 15.5.4).

What remains is the case $M_1 = 1 < M_2$. Since $M_1 = 1$, only one data stream can be transmitted. Therefore, UT_2 uses its M_2 antennas for beamforming which reduces the size of the effective channels again to scalars. As in the previous case, the relay estimates the channels $\boldsymbol{h}_1$ and $\boldsymbol{H}_2$ directly from $N_{\mathrm{P}} \geq M_2 + 1$ pilots to determine a suitable matrix $\boldsymbol{G}$. Then, UT_1 transmits $N_{\mathrm{P,S}} \geq 1$ pilot symbols and UT_2 remains silent. This enables UT_2 to estimate $\boldsymbol{h}_1^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2 \in \mathbb{C}^{1 \times M_2}$. Based on this channel UT_2 computes suitable beamforming weights. Applying this beamforming, the effective channels are reduced to scalars, which, as before are estimated directly by transmitting a set of $N_{\mathrm{P,D}} \geq 2$ dedicated pilots.

15.5.4. Dedicated pilots

Once both terminals and the relay have an estimate of both channels $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ they can use this information to compute a suitable relay amplification matrix $\boldsymbol{G}$ for the data transmission (e.g., via ANOMAX [RH09a] or using the ZF/MMSE transceivers from [UK08]). Since the data is transmitted over the effective channels $\boldsymbol{H}_1^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2$ and $\boldsymbol{H}_2^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1$, the precoding and decoding vectors can be chosen from the left and right singular vectors of the effective channel matrices. In the case of single-stream transmission this transforms the data model into

$$y_1 = h_{1,1}^{(\mathrm{eq})} \cdot s_1 + h_{1,2}^{(\mathrm{eq})} \cdot s_2 + n_1^{(\mathrm{eq})} \tag{15.36}$$

$$y_2 = h^{(\mathrm{eq})}_{2,2} \cdot s_2 + h^{(\mathrm{eq})}_{2,1} \cdot s_1 + n^{(\mathrm{eq})}_2, \tag{15.37}$$

where the "equivalent" channels $h^{(\mathrm{eq})}_{i,j}$ are given by $h^{(\mathrm{eq})}_{i,j} = \gamma \cdot \boldsymbol{d}_i^{\mathrm{T}} \cdot \boldsymbol{H}_i^{\mathrm{T}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_j \cdot \boldsymbol{p}_j$. Here $\boldsymbol{d}_i^{\mathrm{T}} \in \mathbb{C}^{1\times M_i}$ is the decoding vector used at terminal i and $\boldsymbol{p}_j \in \mathbb{C}^{M_j\times 1}$ represents the precoding vector used at terminal j. Then the crucial ANC step to subtract the self-interference requires precise knowledge of $h^{(\mathrm{eq})}_{i,i}$ at terminal i. Obtaining this channel from the estimates of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ is possible, however, due to the multiple estimated quantities that are involved ($\boldsymbol{H}_i$, $\boldsymbol{G}$, $\boldsymbol{d}_i$, $\boldsymbol{p}_i$) the errors may accumulate and the estimate can be unreliable. Therefore, an alternative is given by estimating the equivalent channels via another set of so-called dedicated pilots. Dedicated pilots are transmitted via the equivalent channels for fixed precoding and decoding vectors in order to obtain a new estimate the equivalent channels $h^{(\mathrm{eq})}_{i,j}$. Since in this case our transmission model is dual input single output, already two dedicated pilots are sufficient to estimate $h^{(\mathrm{eq})}_{i,i}$ and $h^{(\mathrm{eq})}_{i,j}$ at terminal i. If instead of one stream, r streams are used simultaneously, $2r$ dedicated pilots are required.

15.6. Simulation results

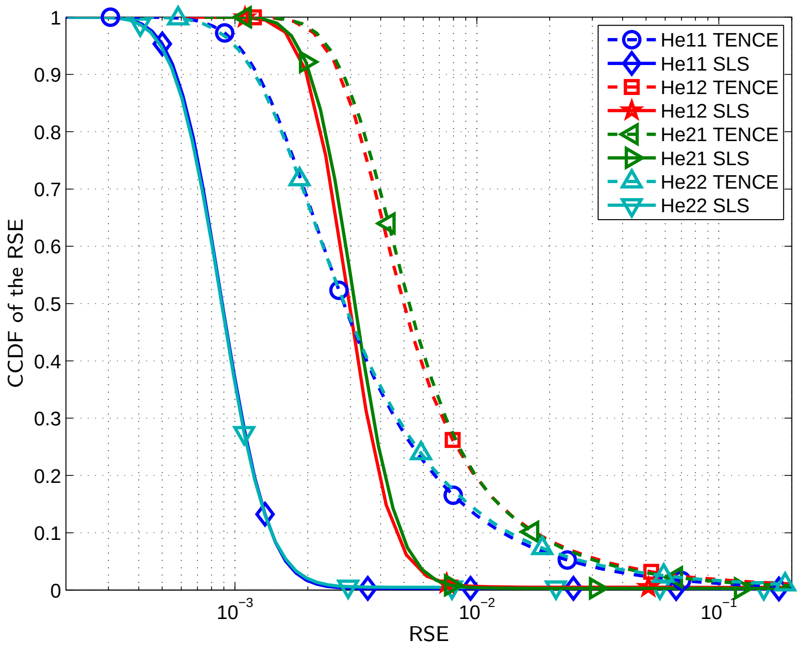


Figure 15.4.: CCDF of the RSE for TENCE and the SLS-based iterative refinement. Scenario: $M_1 = M_2 = M_{\mathrm{R}} = 5$, SNR = 20 dB, $\rho_{\mathrm{R}} = \rho_1 = \rho_2 = 0$ (uncorrelated Rayleigh fading).

In this section, simulation results are shown to compare the different channel estimation

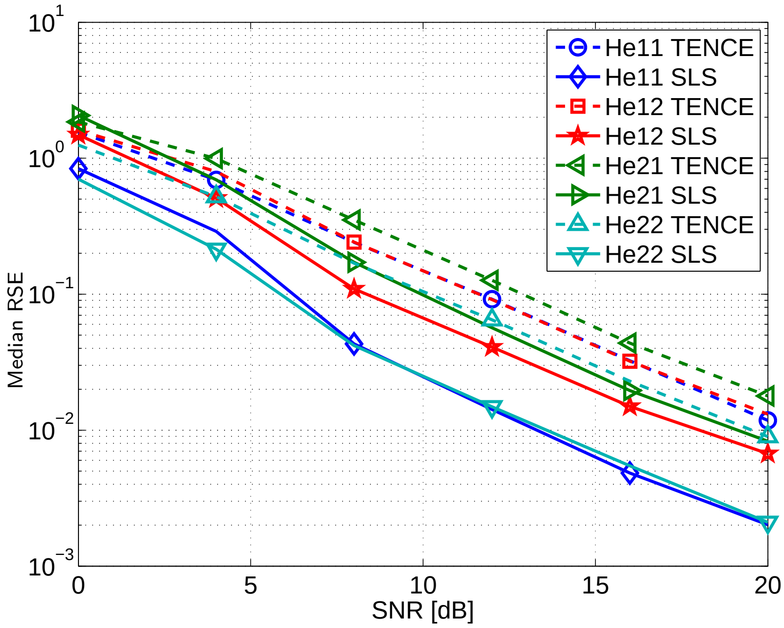


Figure 15.5.: Median of the RSE vs. the SNR for TENCE and the SLS-based iterative refinement. Scenario: $M_1 = 4$, $M_2 = 5$, $M_{\mathrm{R}} = 3$, $\rho_{\mathrm{R}} = 0.9$, $\rho_1 = \rho_2 = 0$ (correlated Rayleigh fading).

approaches and demonstrate the corresponding achievable channel estimation accuracies. We first show the achievable channel estimation accuracy of the separate channels $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ with TENCE and its SLS-based iterative refinement. Then, we compare the LS-based compound channel estimator with the tensor-based channel estimation approach in terms of the estimation error of the compound channels.

For all simulations, the channel matrices are generated according to a correlated Rayleigh fading distribution. The spatial correlation follows a Kronecker model, i.e.,

$$\mathrm{E}\{\boldsymbol{H}_{\mathrm{NLOS},i} \cdot \boldsymbol{H}_{\mathrm{NLOS},i}^{\mathrm{H}}\} = \boldsymbol{R}_{\mathrm{R}} \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}}} \tag{15.38}$$

$$\mathrm{E}\{\boldsymbol{H}_{\mathrm{NLOS},i}^{\mathrm{H}} \cdot \boldsymbol{H}_{\mathrm{NLOS},i}\} = \boldsymbol{R}_i \in \mathbb{C}^{M_i \times M_i}, \; i = 1, 2, \tag{15.39}$$

where $\boldsymbol{R}_{\mathrm{R}} \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}}}$ and $\boldsymbol{R}_i \in \mathbb{C}^{M_i \times M_i}$ model the spatial correlation matrices at the relay and and at user terminal i, respectively. For simplicity, the matrices $\boldsymbol{R}_{\mathrm{R}}$ and $\boldsymbol{R}_i$ are chosen such that their main diagonal elements are equal to one and the magnitude of all off-diagonal elements is equal to ρ_{R} and ρ_i, respectively. The channels are assumed to be constant during the training phase.

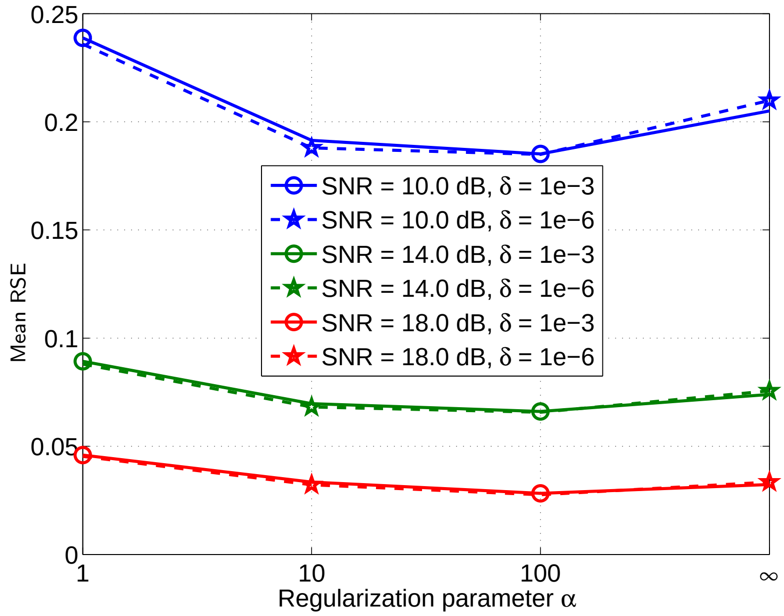


Figure 15.6.: Mean RSE vs. regularization parameter α for different SNRs. Scenario: $M_1 = M_2 = 2$, $M_R = 4$, $\rho_R = \rho_1 = \rho_2 = 0$ (uncorrelated Rayleigh fading).

15.6.1. Performance of TENCE and its SLS-based refinement

In this section we present a selection of simulation results demonstrating the accuracy achievable with TENCE and its SLS-based refinement.

As a measure of the accuracy, we compute the relative squared estimation error (RSE) defined as

$$\mathrm{RSE}(\boldsymbol{H}, \hat{\boldsymbol{H}}) = \min_{p \in \{1,-1\}} \frac{\left\| \boldsymbol{H} - p \cdot \hat{\boldsymbol{H}} \right\|_F^2}{\|\boldsymbol{H}\|_F^2}, \tag{15.40}$$

where p accounts for the sign ambiguity in the estimation of the channels. The estimation error curves are labeled as `He11`, `He12`, `He21`, and `He22`, where the first number indicates the terminal which estimates the channel referenced by the second number. For instance, `He12` represents the estimate of $\boldsymbol{H}_2$ at UT_1.

If not stated otherwise, the design of the training data follows the rules derived in Section 15.3 and we choose $N_R = M_R$ and $N_P = M_1 + M_2$ to minimize the pilot overhead. Moreover, the default values for α and δ are $\alpha = 100$, $\delta = 10^{-3}$. We use a fixed transmit power of $P_T = 1$ for both terminals and the relay and vary the noise power P_N at the terminals and at the relay as a function of the SNR $= 1/P_N$.

The first result shown in Figure 15.4 corresponds to an uncorrelated Rayleigh fading scenario

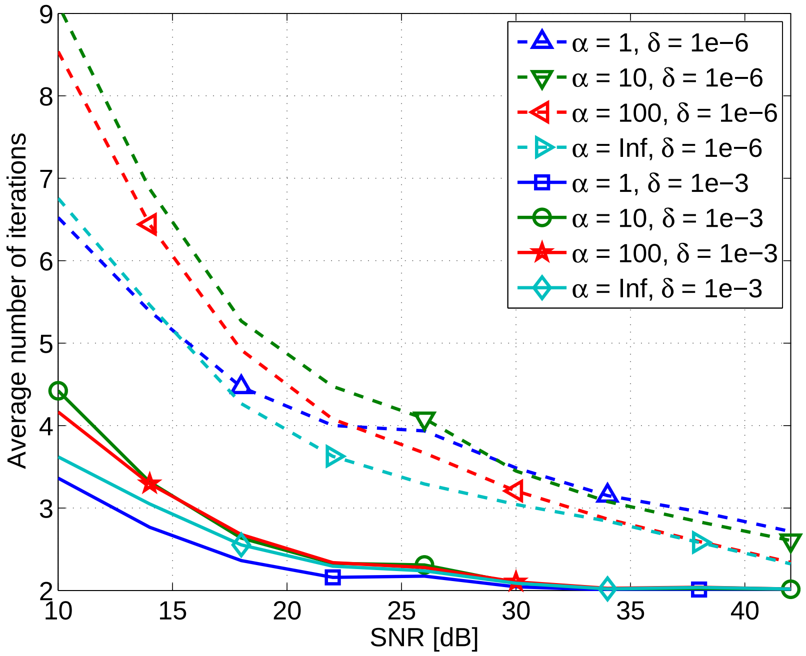


Figure 15.7.: Number of iterations for the SLS-based refinement vs. the SNR for different choices of α and δ. Scenario: $M_1 = M_2 = 2$, $M_{\rm R} = 4$, $\rho_{\rm R} = \rho_1 = \rho_2 = 0$ (uncorrelated Rayleigh fading).

where each terminal is equipped with five antennas. We show the complementary cumulative distribution function (CCDF) of the RSE (i.e., the probability that the RSE exceeds its abscissa) for a fixed SNR of 20 dB and randomly drawn channel realizations. Dashed lines represent the initial estimate obtained via TENCE and solid lines are used for the SLS-based iterative refinement. We observe significant improvements via the iterative scheme in the terminals' own channels to the relay and mild improvements in the channels between the other terminal and the relay. Moreover, the slope of the CCDF is steeper for the SLS-based iterative refinement which means that their estimates are numerically more stable than the initial TENCE estimates.

A correlated Rayleigh fading scenario is investigated in Figure 15.5 where we choose $M_1 = 4, M_2 = 5, M_{\rm R} = 3, K_1 = K_2 = 0, \rho_1 = \rho_2 = 0$, and $\rho_{\rm R} = 0.9$. Therefore, a strong spatial correlation at the relay is present which impacts both $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$. We observe significant improvements obtained by the SLS-based iterative refinement for the estimates of each terminal's own channel to the relay since the iterative channel estimate exploits the fact that each terminal's own channel is present in the first as well as the second mode of the training tensor.

The impact of the design parameters α and δ on the performance of the SLS-based iterative refinement is shown in Figures 15.6 and 15.7. Here, we consider a scenario with uncorrelated

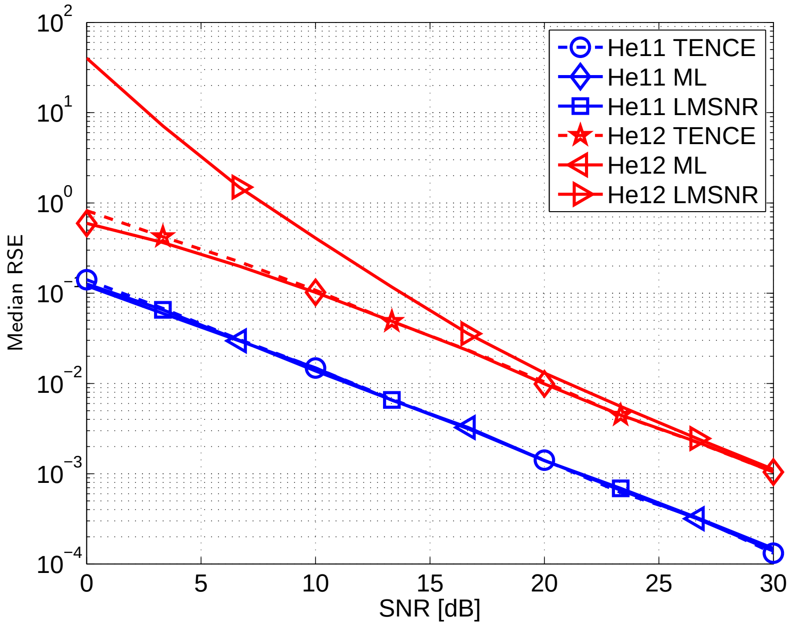


Figure 15.8.: Median of the RSE vs. the SNR comparing TENCE with the ML and the LMSNR estimates from [GZL09b]. Scenario: $M_1 = M_2 = M_{\mathrm{R}} = 1$, (Rayleigh fading).

Rayleigh fading ($K_1 = K_2 = 0$, $\rho_{\mathrm{R}} = \rho_1 = \rho_2 = 0$) for $M_1 = M_2 = 2$ and $M_{\mathrm{R}} = 4$ antennas. In Figure 15.6 we depict the mean RSE for different choices of the regularization parameter α and the SNR. Note that the last point $\alpha = \infty$ corresponds to the case where no regularization is used at all. We observe that for a low SNR a mild amount of regularization ($\alpha \approx 100$) helps to lower the mean RSE and that this effect diminishes for higher SNRs. For a very high SNR, we can skip the regularization completely by setting $\alpha = \infty$. For the same scenario, the average number of iterations of the SLS-based refinement is depicted in Figure 15.7. We observe a slight increase in the number of iterations for the cases where a mild amount of regularization is used. Moreover, we compare two different choices of the threshold parameter δ. Obviously, for $\delta = 10^{-6}$, significantly more iterations are required. However, as evident from Figure 15.6, these additional iterations do not lead to a visible improvement in the RSE. Consequently, $\delta = 10^{-3}$ is a reasonable choice. For a high SNR, the SLS-based iterative refinement always terminates after two iterations. This means that the second iteration does not improve the norm of the residual tensor anymore. Consequently, one could even limit the number of iterations to one without losing any performance in the high SNR regime.

Finally, Figure 15.8 shows the comparison of TENCE with the ML and LMSNR channel estimators proposed in [GZL09b]. Since the latter are only applicable to the SISO case, we set $M_1 = M_2 = M_{\mathrm{R}} = 1$. Note that in this case, TENCE simplifies to the equations shown in Section 15.5.3. Also, we consider a NLOS scenario, i.e., $K_1 = K_2 = 0$. We observe that in terms

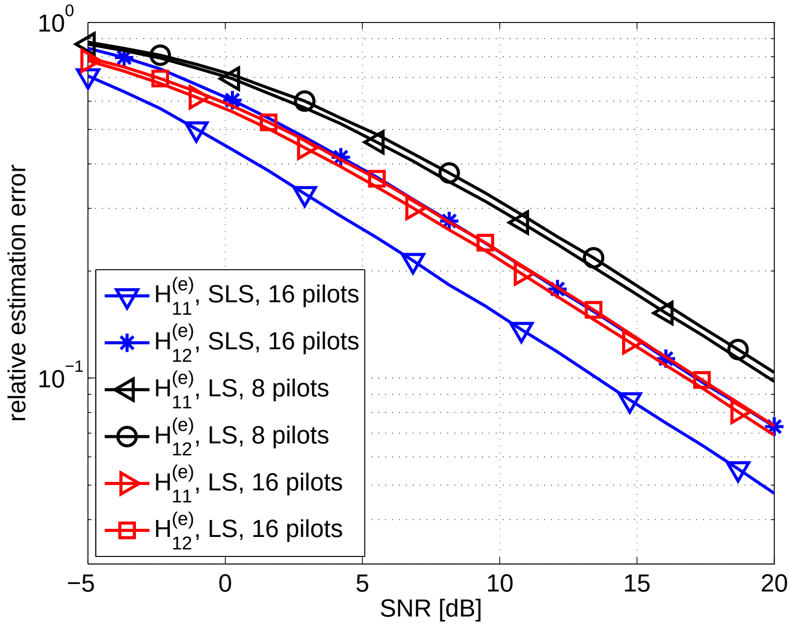


Figure 15.9.: Median rCEE vs. the SNR. Scenario: $M_1 = M_2 = 4$, $M_\mathrm{R} = 2$, $\rho_\mathrm{R} = \rho_1 = \rho_2 = 0$ (uncorrelated Rayleigh fading).

of the Median RSE, TENCE and ML perform almost equally and outperform the suboptimal LMSNR scheme. It should be noted that the complexity of the closed-form TENCE algorithm is lower than the complexity of ML or LMSNR.

15.6.2. Comparison between compound and tensor-based estimator

In order to compare the LS-based compound channel estimator proposed in Section 15.2 with the tensor-based approach presented in Sections 15.3 and 15.4 we consider the relative estimation error (rCEE) of the compound channels defined via

$$\mathrm{rCEE}_{i,j} = \frac{\left\| \boldsymbol{H}^{(\mathrm{e})}_{i,j} - \hat{\boldsymbol{H}}^{(\mathrm{e})}_{i,j} \right\|^2_\mathrm{F}}{\left\| \boldsymbol{H}^{(\mathrm{e})}_{i,j} \right\|^2_\mathrm{F}}. \tag{15.41}$$

Figures 15.9 and 15.10 depict the $\mathrm{rCEE}_{1,1}$ and $\mathrm{rCEE}_{1,2}$ achieved via different approaches. The curves for UT_2 (i.e., $\mathrm{rCEE}_{2,1}$ and $\mathrm{rCEE}_{2,2}$) are omitted since they coincide with the ones for UT_1 due to the symmetry of the problem. The curves labeled "SLS" depict the tensor-based approach using TENCE and the SLS-based iterative refinement with $M_\mathrm{R} \cdot (M_1 + M_2)$ pilots. The curves labeled "LS" show the LS-based approach for the estimation of the compound channel. Since LS requires only $M_1 + M_2$ pilots, two sets of curves are shown: One set that

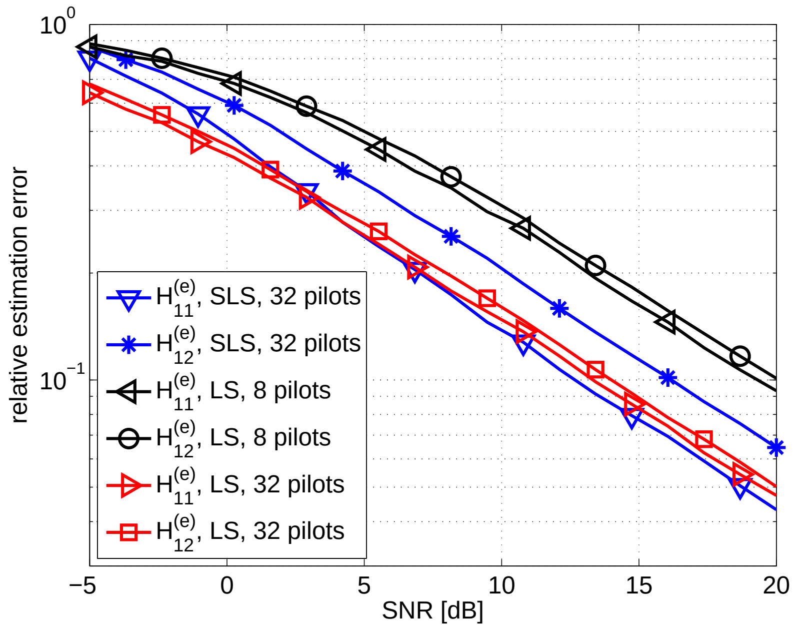


Figure 15.10.: Median rCEE vs. the SNR. Scenario: $M_1 = M_2 = 4$, $M_{\mathrm{R}} = 4$, $\rho_{\mathrm{R}} = \rho_1 = \rho_2 = 0$ (uncorrelated Rayleigh fading).

corresponds to the minimum number of pilots and another set where the number of pilots has been chosen to $M_{\mathrm{R}} \cdot (M_1 + M_2)$ for a fair comparison to the tensor-based approach. Both simulations assume $M_1 = M_2 = 4$ antennas at the user terminals. The number of antennas at the relay is set to $M_{\mathrm{R}} = 2$ for Figure 15.9 and to $M_{\mathrm{R}} = 4$ for Figure 15.10. The relay amplification matrix $\boldsymbol{G}$ is chosen as a DFT matrix (cf. Appendix A.2). We observe that in both cases, the channel $\boldsymbol{H}_{1,1}^{(\mathrm{e})}$, which conveys the self-interference, is estimated more accurate by the tensor-based approach. The estimation accuracies for the channel matrix $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ achieved by LS and SLS are equal for $M_{\mathrm{R}} = 2$ and SLS is slightly worse for $M_{\mathrm{R}} = 4$ (comparing SLS and LS for the same number of pilots).

15.7. Summary

In this chapter we discuss channel estimation schemes for two-way relaying with AF MIMO relays first proposed by us in [RH09e, RH09d, RH10c]. We investigate two channel estimation approaches. First, the LS-based estimator for the compound channels is introduced. It represents a simple and robust scheme with a small pilot overhead. However, it fails to provide the terminals with transmit CSI for non-symmetric relay amplification matrices. Moreover, it ignores the structure of the compound channel matrices which provides room for improvements in the channel estimation accuracy. Then, we introduce a tensor-based approach for estimating the separate channel matrices between the terminals and the relay. We first derive the closed-form TENCE algorithm using the tensor-based algebraic concepts introduced in Chapter 4 and the Least-Squares Khatri-Rao factorization shown in Chapter 3 of this thesis. Furthermore, we propose design rules for the training symbols and the relay amplification matrices that are required for the implementation of TENCE as well as recommendations that improve its estimation accuracy. In a subsequent step we demonstrate that the estimates obtained via TENCE can be further improved by an iterative algorithm based on Structured Least Squares (see Section 11.7 for a discussion of SLS in the context of ESPRIT-type algorithms). We show via simulations that significant improvements are achievable and, depending on the scenario, between one and four iterations are sufficient.

Comparing the two approaches we find that the tensor-based approach yields more accurate estimates of the compound channel matrices that convey the self-interference if the number of antennas at the relay is smaller than the number of antennas at the terminals. Moreover, it always provides the user terminals with transmit CSI, even for non-symmetric relay amplification matrices.

16. Relay amplification matrix design

We now shift our focus to the design of the relay amplification matrix for two-way relaying with a MIMO AF relay. Based on the CSI acquisition concepts discussed in the previous section, the relay can adapt its amplification matrix to the current channel state. We first show in Section 16.1 that this matrix has an impact on the effective channel matrices as well as the noise covariance matrices that each user experiences. Therefore, finding a design of this matrix that provides the optimal system sum-rate is a mathematically challenging problem.

We summarize the state of the art in Section 16.2. As we show there, the major shortcoming of the existing approaches for the relay amplification matrix design is that they are either based on heuristic solutions [UK08, VH11] that do not show a satisfactory system performance or on complicated numerical optimization procedures that are difficult to implement in practice [ZLCC09, LSPL10].

To this end, we first present the Algebraic Norm Maximizing (ANOMAX) transmit strategy, first proposed by us in [RH09a], in Section 16.3 which is a closed-form solution that maximizes the norms of the effective channel matrices. This approach tends to concentrate the energy onto the dominant eigenmodes of the effective channels which yields a very good bit error rate for single-stream transmission schemes. However, it is less suitable for spatially multiplexing several streams as the low-rank nature of the effective channels fails to provide the full spatial multiplexing gain for high SNRs. In order to address this issue, we present the "Rank-Restored" ANOMAX (RR-ANOMAX) scheme [RH10a] in Section 16.3.4. RR-ANOMAX restores the required rank of the channel matrices in order to obtain the full spatial multiplexing gain for high SNRs.

We then turn to the special case of single-antenna terminals in Section 16.4. We show that in this case, the sum-rate can be expressed as the product of two Rayleigh quotients (which has been shown before only in the high SNR regime by [LYC08]). We also demonstrate that the optimal solution to this problem can be found via a semi-algebraic scheme based on generalized eigenvectors (RAGES) [RH10b]. RAGES is significantly less complex than existing optimal strategies based on numerical optimization procedures [ZLCC09]. We verify that RAGES provides the optimal system sum-rate by comparing it numerically to the sum-rate optimal Polynomial Time DC (POTDC) algorithm, proposed by us in [KVRH12, KRVH12].

Numerical results are shown in Section 16.5 before drawing the conclusions and outlining possible directions of future work in Section 16.6.

16.1. Problem description

16.1.1. Impact of the relay amplification matrix

Recall that the users' received signals after the cancellation of the self-interference can be expressed as given in (14.8) and (14.9) which is restated here for convenience

$$\boldsymbol{z}_1 = \boldsymbol{H}_{1,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 + \tilde{\boldsymbol{n}}_1 \tag{16.1}$$

$$\boldsymbol{z}_2 = \boldsymbol{H}_{2,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 + \tilde{\boldsymbol{n}}_2. \tag{16.2}$$

where $\boldsymbol{H}_{i,j}^{(\mathrm{e})} = \gamma \cdot \boldsymbol{H}_i^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_j^{(\mathrm{f})}$ and $\tilde{\boldsymbol{R}}_{\mathrm{N},i} = \mathbb{E}\left\{\tilde{\boldsymbol{n}}_i \cdot \tilde{\boldsymbol{n}}_i^{\mathrm{H}}\right\} = \gamma^2 \cdot \boldsymbol{H}_i^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}} \cdot \boldsymbol{H}_i^{(\mathrm{b})^{\mathrm{H}}} + \boldsymbol{R}_{\mathrm{N},i}$. Consequently, the choice of the relay amplification matrix $\boldsymbol{G}$ influences the effective MIMO channel matrices which convey our signal of interest as well as the covariance matrix of the effective noise contributions. In this chapter we investigate suboptimal choices of $\boldsymbol{G}$ motivated by different goals, e.g., sum-rate maximization or Bit Error Rate (BER) minimization. In the general case, the capacity of the MIMO two-way relaying channel is still not known since it involves the joint optimization over the terminals' transmit covariance matrices[1] $\boldsymbol{R}_i = \mathbb{E}\left\{\boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{H}}\right\}$, $i = 1, 2$, and the relay amplification matrix $\boldsymbol{G}$. This optimization problem is a rather complicated coupled non-convex problem [LSPL10]. This motivates simple algebraic approximations to arrive at tractable solutions.

A strong simplification that is often made is to decouple the users' transmit strategies from the relay amplification strategy [VH11]. This step simplifies the design significantly, albeit optimality is in general lost. Yet, it is a practical solution that fits well into the algebraic approach to system design we emphasize in this thesis.

Also, for the system under investigation the loss incurred due to this assumption is not very severe. Once the relay amplification matrix is fixed, for the users the effective input-output relation takes the form of (16.1) and (16.2), i.e., P2P MIMO links with colored noise. If the channel matrix and the noise covariance matrix are known, the users can use any P2P MIMO strategy to transmit data over the channels. For instance, SMUX and DET can be used to maximize the data rate or to minimize the BER [PNG03] for the transmission over the compound channels. For a fixed $\boldsymbol{G}$ these strategies are indeed "optimal". The only reason they are not optimal in general is that the relay could adopt its choice of $\boldsymbol{G}$ to the users' transmit strategies if it knew them. However, this requires a joint design of both strategies

[1] Note that as highlighted in Section 14.3 we assume the transmit signals to be zero mean. If $\boldsymbol{x}_i$ has a mean different from zero, $\boldsymbol{R}_i$ should be defined as the correlation matrix $\boldsymbol{R}_i = \mathbb{E}\left\{\boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{H}}\right\}$, not as the covariance matrix $\mathbb{E}\left\{(\boldsymbol{x}_i - \mathbb{E}\{\boldsymbol{x}_i\}) \cdot (\boldsymbol{x}_i - \mathbb{E}\{\boldsymbol{x}_i\})^{\mathrm{H}}\right\}$.

which is in general intractable.

We therefore design $\boldsymbol{G}$ independently and then treat (16.1) and (16.2) as two parallel P2P MIMO links. Since the capacity of P2P MIMO is known, we can define a "capacity" of the two-way relaying system as a function of $\boldsymbol{G}$ by summing the capacities of (16.1) and (16.2) and dividing the result by two to account for the fact that we require two time slots in two-way relaying. However, since the resulting number still depends on $\boldsymbol{G}$, it is not a capacity in the information theoretic sense. Hence, we refer to it as maximum mutual information (MMI). Formally, we can express it as

$$\mathrm{MMI}(\boldsymbol{G}) = \frac{1}{2}\left(\underset{P_{\mathrm{T},2}}{\mathrm{Capacity}}\left\{\boldsymbol{H}_{1,2}^{(\mathrm{e})}|\tilde{\boldsymbol{R}}_{\mathrm{N},1}\right\} + \underset{P_{\mathrm{T},1}}{\mathrm{Capacity}}\left\{\boldsymbol{H}_{2,1}^{(\mathrm{e})}|\tilde{\boldsymbol{R}}_{\mathrm{N},2}\right\}\right) \tag{16.3}$$

where $P_{\mathrm{T},i} = \mathrm{trace}\{\boldsymbol{R}_i\}$ is the i-th user's transmit power and $\mathrm{Capacity}_P\{\boldsymbol{H}|\boldsymbol{R}_{\mathrm{N}}\}$ denotes the capacity of the P2P MIMO link with noise covariance matrix $\boldsymbol{R}_{\mathrm{N}} \in \mathbb{C}^{M_{\mathrm{R}}\times M_{\mathrm{R}}}$ and transmit power P. This capacity can be computed as [Tel99]

$$\underset{P}{\mathrm{Capacity}}\{\boldsymbol{H}|\boldsymbol{R}_{\mathrm{N}}\} = \underset{\boldsymbol{Q}\succeq\boldsymbol{0}|\,\mathrm{trace}\{\boldsymbol{Q}\}=P}{\arg\max} \log_2\det\left(\boldsymbol{I}_{M_{\mathrm{R}}} + \boldsymbol{R}_{\mathrm{N}}^{-1}\cdot\boldsymbol{H}\cdot\boldsymbol{Q}\cdot\boldsymbol{H}^{\mathrm{H}}\right) \tag{16.4}$$

where the maximization over the transmit covariance matrix $\boldsymbol{Q}$ leads to the water-filling algorithm. Note that in (16.3), both effective channels $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ as well as both noise covariance matrices $\tilde{\boldsymbol{R}}_{\mathrm{N},i}$, $i = 1, 2$ depend on $\boldsymbol{G}$.

16.1.2. Power constraint at the relay

Like every physical transmission device, the relay station's transmit power is limited. Therefore, when we compare transmission schemes we need to satisfy a power constraint for all of them to render the comparison fair. The average relay's transmit power (averaged over symbols and noise realizations) can be computed via

$$\begin{aligned}\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} &= \mathrm{trace}\left\{\mathbb{E}\left\{\bar{\boldsymbol{r}}\cdot\bar{\boldsymbol{r}}^{\mathrm{H}}\right\}\right\} \qquad (16.5)\\ &= \gamma^2\cdot\mathrm{trace}\left\{\boldsymbol{G}\cdot\boldsymbol{H}_1^{(\mathrm{f})}\cdot\boldsymbol{R}_1\cdot\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}}\cdot\boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G}\cdot\boldsymbol{H}_2^{(\mathrm{f})}\cdot\boldsymbol{R}_2\cdot\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}}\cdot\boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G}\cdot\boldsymbol{R}_{\mathrm{N,R}}\cdot\boldsymbol{G}^{\mathrm{H}}\right\}\end{aligned}$$

Consequently, if we want to compute γ such that $\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = P_{\mathrm{T,R}}$ we need to know the users' transmit strategies. On the other hand, these are a function of γ as well since γ scales the users' effective channels and hence the effective SNR. Therefore, such a choice of γ requires a joint (or at least iterative) design of the relay strategy and the users' transmit covariance matrices. Hence, to arrive at a practical, low-complexity design we need to perform approximations.

Proposition 16.1.1. *Choosing γ as*

$$\gamma^2 = \frac{P_{\mathrm{T,R}}}{\mathrm{trace}\left\{P_{\mathrm{T},1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + P_{\mathrm{T},2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\}} \tag{16.6}$$

guarantees that $\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} \leq P_{\mathrm{T,R}}$ regardless of the users' transmit covariance matrices.

Proof: cf. Appendix E.2.

The choice of γ in Proposition 16.1.1 corresponds to assuming that $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ are scaled identity matrices. Note that the inequality $\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} \leq P_{\mathrm{T,R}}$ is in general not tight, i.e., the power used by the relay is strictly lower than the upper limit $P_{\mathrm{T,R}}$. In fact, as the users distribute their power on $r > 1$ spatial streams it tends to be by a factor of r lower. This is shown for a special case in the following proposition.

Proposition 16.1.2. *Consider the special case where $M_1 = M_2 = M_{\mathrm{U}}$ and the users perform spatial multiplexing over the effective channels in order to maximize their data rate. Then, choosing γ according to* (16.6) *we have in the high SNR regime $\lim_{P_{\mathrm{N}} \to 0} \mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \frac{P_{\mathrm{T,R}}}{\min\{M_{\mathrm{U}}, M_{\mathrm{R}}\}}$, where $P_{\mathrm{N}} = \max\{P_{\mathrm{N},1}, P_{\mathrm{N},2}, P_{\mathrm{N,R}}\}$.*

Proof: cf. Appendix E.3.

However, there are special cases where the power can be adjusted exactly to the desired $P_{\mathrm{T,R}}$. For instance, for the single-antenna case, the transmit covariance matrices are scalars equal to the transmit powers $P_{\mathrm{T},1}$ and $P_{\mathrm{T},2}$. Consequently we have the following corollary from Proposition 16.1.1:

Corollary 16.1.3. *For $M_1 = M_2 = 1$ (single-antenna terminals), the inequality in Proposition 16.1.1 is tight, i.e., $\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = P_{\mathrm{T,R}}$.*

Proof. By inserting $\boldsymbol{R}_1 = P_{\mathrm{T},1}$ and $\boldsymbol{R}_2 = P_{\mathrm{T},2}$ into equation (16.5), we immediately obtain (16.6). □

Note that if we replace the instantaneous power constraint for the current channel realization by an "ergodic" power constraint which is satisfied on average over all channel realizations given a specific distribution, we may be able to compute the used power in closed form. Such an assumption is reasonable if only long-term CSI is available at the relay. In this case, closed-form power allocation strategies, for the Dual Channel Matching (DCM) scheme and the fixed DFT matrix are shown in [JS10a].

16.1.3. Single antenna terminals

In the special case where the user terminals have a single antenna ($M_1 = M_2 = 1$), the data model can be significantly simplified. Since many publications on this special case exist, we present the simplified model here.

After canceling the self-interference, the received signals can be expressed via (14.8) and (14.9), which for a single antenna simplifies into

$$z_1 = h_{1,2}^{(e)} \cdot x_2 + \tilde{n}_1 \tag{16.7}$$

$$z_2 = h_{2,1}^{(e)} \cdot x_1 + \tilde{n}_2, \tag{16.8}$$

where $h_{i,j}^{(e)} = \boldsymbol{h}_i^{(b)^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{h}_j^{(f)}$ and $\tilde{n}_i = \boldsymbol{h}_i^{(b)^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{n}_{\mathrm{R}} + n_i$, $i = 1, 2$. Since (16.7) comprises of one scalar desired signal component and noise, it is easy to express the SNR in closed-form. Let $P_{\mathrm{R},i}$ and $\tilde{P}_{\mathrm{N},i}$ denote the power of the desired signal component and the effective noise component at UT_i, i.e.,

$$P_{\mathrm{R},1} = \mathbb{E}\left\{\left|h_{1,2}^{(e)} \cdot x_2\right|^2\right\}, \quad P_{\mathrm{R},2} = \mathbb{E}\left\{\left|h_{2,1}^{(e)} \cdot x_1\right|^2\right\}, \quad \text{and} \quad \tilde{P}_{\mathrm{N},i} = \mathbb{E}\left\{|\tilde{n}_i|^2\right\}, \tag{16.9}$$

so that the SNR at UT_i is given by $\eta_i = \frac{P_{\mathrm{R},i}}{\tilde{P}_{\mathrm{N},i}}$. Since both, desired signal and noise component are linear functions in $\boldsymbol{G}_\gamma$, their powers become quadratic functions in the elements of $\boldsymbol{G}_\gamma$. To facilitate the further algebraic manipulation of measures derived from these powers (such as SNRs and sum-rates) it is desirable to express the quadratic expressions into their canonical form. Following the argumentation in Section 3.1.3, the following "canonical" expressions can be derived:

Lemma 16.1.4. [RH10b] *The powers of the desired signal component and the effective noise component $P_{\mathrm{R},i}$ and $\tilde{P}_{\mathrm{N},i}$ can be written as quadratic forms in the vector $\boldsymbol{g}_\gamma \doteq \mathrm{vec}\left\{\boldsymbol{G}_\gamma\right\} \in \mathbb{C}^{M_{\mathrm{R}}^2 \times 1}$ via*

$$P_{\mathrm{R},1} = \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \boldsymbol{K}_{2,1} \cdot \boldsymbol{g}_\gamma \cdot P_{\mathrm{T},2} \tag{16.10}$$

$$P_{\mathrm{R},2} = \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \boldsymbol{K}_{1,2} \cdot \boldsymbol{g}_\gamma \cdot P_{\mathrm{T},1} \tag{16.11}$$

$$\tilde{P}_{\mathrm{N},i} = \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \boldsymbol{J}_i \cdot \boldsymbol{g}_\gamma + P_{\mathrm{N},i}, \; i = 1, 2, \tag{16.12}$$

where $P_{\mathrm{N},i} = \mathbb{E}\left\{|n_i|^2\right\}$ and the matrices $\boldsymbol{K}_{i,j} \in \mathbb{C}^{M_{\mathrm{R}}^2 \times M_{\mathrm{R}}^2}$ and $\boldsymbol{J}_i \in \mathbb{C}^{M_{\mathrm{R}}^2 \times M_{\mathrm{R}}^2}$ are given by

$$\boldsymbol{K}_{i,j} = \left[\left(\boldsymbol{h}_i^{(f)} \cdot \boldsymbol{h}_i^{(f)^{\mathrm{H}}}\right) \otimes \left(\boldsymbol{h}_j^{(b)} \cdot \boldsymbol{h}_j^{(b)^{\mathrm{H}}}\right)\right]^{\mathrm{T}}$$

$$\boldsymbol{J}_i = \left[\boldsymbol{R}_{\mathrm{N,R}} \otimes \left(\boldsymbol{h}_i^{(\mathrm{b})} \cdot \boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{H}}}\right)\right]^{\mathrm{T}}.$$

Moreover, the power consumed by the relay can be written as

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_\gamma, \quad \textit{where} \quad \boldsymbol{Q} = \boldsymbol{R}_{\mathrm{R}}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_{\mathrm{R}}} \tag{16.13}$$

and $\boldsymbol{R}_{\mathrm{R}} = \mathbb{E}\left\{\boldsymbol{r} \cdot \boldsymbol{r}^{\mathrm{H}}\right\} = \boldsymbol{h}_1^{(\mathrm{f})} \cdot \boldsymbol{h}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot P_{\mathrm{T},1} + \boldsymbol{h}_2^{(\mathrm{f})} \cdot \boldsymbol{h}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot P_{\mathrm{T},2} + \boldsymbol{R}_{\mathrm{N,R}}$ *is the relay's receive correlation matrix.*

Proof: cf. Appendix E.4.

Based on this lemma, the semi-algebraic RAGES scheme (proposed by us in [RH10b]) for finding the sum-rate optimal relay amplification matrix is derived in Section 16.4 and compared to the optimal solution found via the Polynomial Time DC (POTDC) algorithm (proposed by us in [KVRH12, KRVH12]) in Section 16.5. In extension to [KRVH12], we present a low-complexity version of RAGES in this thesis, see Appendix E.9.

16.2. State of the art

As the general state of the art in (two-way) relaying was already discussed in Section 14.1, we focus on existing literature related to the design of the relay amplification matrix in two-way relaying with MIMO AF relays here.

For the general MIMO case, the sum-rate optimal relay amplification strategy is still not known. However, many sub-optimal algebraic solutions have been proposed. For instance, [UK08] proposes the ZF and MMSE transceivers, which cancel the self-interference of the terminals as well as their inter-stream interference at the relay station. A multi-user extension of this idea is studied in [JS10b] where the relay is designed based on ZF and MMSE approaches and the users' beamforming strategies are optimized. The Dual Channel Matching (DCM) approach is introduced in [VH11], along with an upper and a lower bound on its achievable sum-rate. The main drawback of [UK08] and [VH11] is that in terms of the relaying strategy, they are proposed in an ad-hoc manner, i.e., not based on a cost function that should be optimized. Hence their performance is sub-optimal with respect to sum-rate or bit error rate, as we show in the numerical simulations in Section 16.5. Optimization-based strategies for finding the users' precoding and decoding matrices for the case where the relay station uses only a scaled identity matrix are discussed in [LPC08]. The extension to the joint optimization of the relay's and the users' pre- and decoders is shown in [LSPL10] and a multi-pair extension is shown in [CLSO12]. However, [LSPL10, CLSO12] rely on gradient descent methods that have a high computational complexity and no guarantee to find the global optimum. Note that the optimal beamforming design for a two-way relaying network of single-antenna relay nodes has been well investigated too, e.g., [HNSG10, ZZC11, SD12, TW12] and many others. However, the results are not applicable to MIMO relays since these have more degrees of freedom in terms of spatial adaptation. These shortcomings of existing approaches provide the motivation for finding algebraic solutions for the design of the relay amplification matrix that are inspired by optimizing a system parameter (e.g., bit error rate or sum-rate, as shown in Section 16.3.1) and hence outperform existing algebraic approaches.

For the special case of single-antenna terminals, the optimal beamforming strategy was first derived in [ZLCC09]. However, [ZLCC09] requires a numerically quite involved procedure. First, the problem is recast in terms of finding an unknown complex 2×2 matrix, then rate profiles are introduced, an equivalent power minimization problem is cast, and then solved via a semi-definite relaxation step. Note that [ZLCC09] also describes the capacity region as well as the achievable rate region. In [LYC08], the approximate sum-rate for high SNR is formulated in terms of a product of Rayleigh quotients and maximized by a General Power Iterative algorithm. Both methods require complicated iterative optimization-based algorithms to find the solution. This motivates why we aim at finding algebraic approaches for computing the sum-rate optimal relay amplification matrix. As we show in Section 16.4, a semi-algebraic algorithm based on generalized eigenvectors provides the optimal sum-rate while being significantly less complex compared to [ZLCC09].

Note that in [KVRH12, KRVH12] we demonstrate that the maximization of the product of Rayleigh quotients is a special case of a class of problems referred to as DC (Difference of Convex functions). While DC problems are in general NP-hard, a subset of these can be solved efficiently in polynomial time via a sequence of semi-definite problems obtained by local linearizations of the non-convex parts of the cost function. The underlying Polynomial-Time DC (POTDC) algorithm and its application to finding the sum-rate optimal strategy for two-way relaying with single-antenna terminals are discussed in [KVRH12, KRVH12].

16.3. Algebraic Norm Maximizing (ANOMAX) transmit strategy

In this section we discuss the Algebraic Norm Maximizing (ANOMAX) transmit strategy first introduced in [RH09a]. ANOMAX represents a very simple and fully algebraic design of $\boldsymbol{G}$ obtained via an intuitively designed cost function which is introduced in Section 16.3.1. ANOMAX tends to concentrate the energy onto the dominant eigenmodes of the effective channels, which yields a very good BER performance and renders it an attractive approach to systems where we target diversity optimization. However, in terms of system sum-rate, a more even distribution of power across eigenmodes is required, especially in the high SNR regime. A modified version of ANOMAX that restores the rank of the effective channels when needed is discussed in Section 16.3.4.

16.3.1. Derivation of ANOMAX

Recall that the users' received signals after the cancellation of the self-interference can be expressed as given in (14.8) and (14.9) which is restated here for convenience

$$\boldsymbol{z}_1 = \gamma \cdot \boldsymbol{H}_{1,2}^{(\mathrm{e})} \cdot \boldsymbol{x}_2 + \tilde{\boldsymbol{n}}_1 \tag{16.14}$$

$$\boldsymbol{z}_2 = \gamma \cdot \boldsymbol{H}_{2,1}^{(\mathrm{e})} \cdot \boldsymbol{x}_1 + \tilde{\boldsymbol{n}}_2. \tag{16.15}$$

We observe that the desired signals $\boldsymbol{x}_i$, $i = 1, 2$ are conveyed through the effective channels $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$, respectively, and hence we should design $\boldsymbol{G}$ such that the transmission over these channels is improved. The difficult task is to design an appropriate measure to evaluate how the channel quality changes with $\boldsymbol{G}$. The approach taken in ANOMAX is to simply consider the Frobenius norms of these channels, ignoring the impact of $\boldsymbol{G}$ on the covariance matrix of $\tilde{\boldsymbol{n}}_i$. This cost function is motivated by the fact that the maximization of the Frobenius norm in P2P MIMO systems yields the SNR-optimal transmission strategy DET as shown in [PNG03]. This motivates the cost function for ANOMAX which is given by

$$\begin{aligned}\boldsymbol{G}_{\mathrm{opt}} &= \underset{\boldsymbol{G}|\|\boldsymbol{G}\|_{\mathrm{F}}=1}{\arg\max} \left(\left\| \boldsymbol{H}_{1,2}^{(\mathrm{e})} \right\|_{\mathrm{F}}^2 + \left\| \boldsymbol{H}_{2,1}^{(\mathrm{e})} \right\|_{\mathrm{F}}^2 \right) &(16.16)\\ &= \underset{\boldsymbol{G}|\|\boldsymbol{G}\|_{\mathrm{F}}=1}{\arg\max} \left(\left\| \boldsymbol{H}_1^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \right\|_{\mathrm{F}}^2 + \left\| \boldsymbol{H}_2^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \right\|_{\mathrm{F}}^2 \right). &(16.17)\end{aligned}$$

Note that since the cost function scales linearly with the squared Frobenius norm of $\boldsymbol{G}$ we have to confine this norm to an arbitrary constant since otherwise no maximum exists. We have chosen norm one for simplicity[2]. The main advantage of this simple cost function is that it admits an algebraic solution. Let $\boldsymbol{g} = \mathrm{vec}\{\boldsymbol{G}\}$. Then, the vector $\boldsymbol{g}$ which maximizes (16.16) can be computed via

[2]Remember that, in order to satisfy the power constraint at the relay, $\boldsymbol{G}$ is scaled by γ, cf. Section 16.1.2.

- The dominant eigenvector of the matrix

$$\boldsymbol{R}_{\text{ANO}} = \left(\left(\boldsymbol{H}_2^{(\text{f})} \cdot \boldsymbol{H}_2^{(\text{f})^{\text{H}}} \right)^{\text{T}} \otimes \left(\boldsymbol{H}_1^{(\text{b})^{\text{H}}} \cdot \boldsymbol{H}_1^{(\text{b})} \right) \right) + \left(\left(\boldsymbol{H}_1^{(\text{f})} \cdot \boldsymbol{H}_1^{(\text{f})^{\text{H}}} \right)^{\text{T}} \otimes \left(\boldsymbol{H}_2^{(\text{b})^{\text{H}}} \cdot \boldsymbol{H}_2^{(\text{b})} \right) \right) \tag{16.18}$$

- (*or equivalently*) The complex conjugate of the dominant left singular vector ($\boldsymbol{u}_1^*$) of the matrix

$$\boldsymbol{K}_{\text{ANO}} = \left[\left(\boldsymbol{H}_2^{(\text{f})} \otimes \boldsymbol{H}_1^{(\text{b})^{\text{T}}} \right) \quad \left(\boldsymbol{H}_1^{(\text{f})} \otimes \boldsymbol{H}_2^{(\text{b})^{\text{T}}} \right) \right] \tag{16.19}$$

- (*or equivalently*) The dominant right singular vector ($\boldsymbol{v}_1$) of the matrix

$$\boldsymbol{K}_{\text{ANO}}^{\text{T}} = \begin{bmatrix} \boldsymbol{H}_2^{(\text{f})^{\text{T}}} \otimes \boldsymbol{H}_1^{(\text{b})} \\ \boldsymbol{H}_1^{(\text{f})^{\text{T}}} \otimes \boldsymbol{H}_2^{(\text{b})} \end{bmatrix}. \tag{16.20}$$

Proof: cf. Section E.5.
Note that $\boldsymbol{R}_{\text{ANO}} = \boldsymbol{K}_{\text{ANO}}^* \cdot \boldsymbol{R}_{\text{ANO}}^{\text{T}}$.

16.3.2. Weighted ANOMAX

The ANOMAX solution can easily be generalized into a "weighted" version [RH09a] by introducing a weighting coefficient into (16.16). This yields the modified cost function

$$\boldsymbol{G}_{\text{opt}}^{(\beta)} = \underset{\boldsymbol{G} | \|\boldsymbol{G}\|_{\text{F}}=1}{\arg\max} \left(\beta^2 \cdot \left\| \boldsymbol{H}_{1,2}^{(\text{e})} \right\|_{\text{F}}^2 + (1-\beta)^2 \cdot \left\| \boldsymbol{H}_{2,1}^{(\text{e})} \right\|_{\text{F}}^2 \right) \tag{16.21}$$

where $\beta \in [0,1]$ allows to distribute the weight unequally between the link from UT_1 to UT_2 and the link from UT_2 to UT_1. For $\beta = 0.5$ we have equal weighting between both links as before. Using similar steps as above, the solution to this cost function becomes the dominant eigenvector of the modified matrix $\boldsymbol{R}_{\text{ANO}}^{(\beta)}$

$$\boldsymbol{R}_{\text{ANO}}^{(\beta)} = \beta^2 \cdot \left(\left(\boldsymbol{H}_2^{(\text{f})} \cdot \boldsymbol{H}_2^{(\text{f})^{\text{H}}} \right)^{\text{T}} \otimes \left(\boldsymbol{H}_1^{(\text{b})^{\text{H}}} \cdot \boldsymbol{H}_1^{(\text{b})} \right) \right) + (1-\beta)^2 \cdot \left(\left(\boldsymbol{H}_1^{(\text{f})} \cdot \boldsymbol{H}_1^{(\text{f})^{\text{H}}} \right)^{\text{T}} \otimes \left(\boldsymbol{H}_2^{(\text{b})^{\text{H}}} \cdot \boldsymbol{H}_2^{(\text{b})} \right) \right)$$

or equivalently, the conjugate of the dominant left singular vector of the modified matrix $\boldsymbol{K}_{\text{ANO}}^{(\beta)}$

$$\boldsymbol{K}_{\text{ANO}}^{(\beta)} = \left[\beta \cdot \left(\boldsymbol{H}_2^{(\text{f})} \otimes \boldsymbol{H}_1^{(\text{b})^{\text{T}}} \right) \quad (1-\beta) \cdot \left(\boldsymbol{H}_1^{(\text{f})} \otimes \boldsymbol{H}_2^{(\text{b})^{\text{T}}} \right) \right], \tag{16.22}$$

which satisfies $\boldsymbol{R}_{\mathrm{ANO}}^{(\beta)} = \boldsymbol{R}_{\mathrm{ANO}}^{(\beta)^*} \cdot \boldsymbol{K}_{\mathrm{ANO}}^{(\beta)^{\mathrm{T}}}$. This weighting adds more flexibility to the system design, since it allows to distribute the power unevenly between the two links. The norm of one of the channels can be increased further at the cost of a reduction in the norm of the other channel. The ratio by which the power can be shifted depends on the channel realizations and on the antenna configuration. The more relay antennas we have, the larger this ratio. For instance, for $M_{\mathrm{R}} = 4$, a typical value for this ratio would be 10. We have shown that this weighting can be used to lower the required transmit power at the relay in order to achieve certain SNR targets in near/far scenarios [RH09c]. For instance, if UT_1 has a higher SNR target than UT_2 we can set $\beta > 0.5$ to shift more power onto $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$.

16.3.3. Discussion

To gain further insights into the ANOMAX solution, we first investigate some special cases. The following proposition establishes a link between ANOMAX and DCM[3].

Proposition 16.3.1. *For the special case $M_1 = M_2 = 1$ (single-antenna terminals), ANOMAX and DCM are identical.*

Proof: cf. Appendix E.6.

As highlighted in Appendix E.6 it follows from the proof of Proposition 16.3.1 that ANOMAX is most efficient for the case of colinear channels and least efficient for the case of orthogonal channels where $\boldsymbol{h}_1^{\mathrm{H}} \cdot \boldsymbol{h}_2 = 0$. Note that it also follows from this proposition that the rank of $\boldsymbol{G}$ is less than or equal to two in this special case. For the MIMO case we have the following result for the limiting values of β:

Proposition 16.3.2. *In the special cases $\beta = 0$ and $\beta = 1$, we have the following rank-one solutions*

$$\boldsymbol{G}^{(\beta=1)} = \boldsymbol{v}_{1,1}^{(\mathrm{b})} \cdot \boldsymbol{u}_{2,1}^{(\mathrm{f})^{\mathrm{H}}} \quad \text{and} \quad \boldsymbol{G}^{(\beta=0)} = \boldsymbol{v}_{2,1}^{(\mathrm{b})} \cdot \boldsymbol{u}_{1,1}^{(\mathrm{f})^{\mathrm{H}}}, \tag{16.23}$$

where $\boldsymbol{u}_{n,1}^{(\mathrm{f})}$ is the dominant left singular vector of $\boldsymbol{H}_n^{(\mathrm{f})}$ and $\boldsymbol{v}_{n,1}^{(\mathrm{b})}$ is the dominant right singular vector of $\boldsymbol{H}_n^{(\mathrm{b})}$, $n = 1, 2$, respectively.

Proof: cf. Appendix E.7.

Note that for reciprocal channels $\boldsymbol{H}_n^{(\mathrm{f})} = \boldsymbol{H}_n = \boldsymbol{H}_n^{(\mathrm{b})^{\mathrm{T}}}$, equation (16.23) simplifies to $\boldsymbol{G}^{(\beta=1)} = \boldsymbol{u}_{1,1}^* \cdot \boldsymbol{u}_{2,1}^{\mathrm{H}}$ and $\boldsymbol{G}^{(\beta=0)} = \boldsymbol{u}_{2,1}^* \cdot \boldsymbol{u}_{1,1}^{\mathrm{H}}$, where $\boldsymbol{u}_{n,1}$ is the dominant left singular vector of $\boldsymbol{H}_n$ for $n = 1, 2$.

[3] DCM chooses $\boldsymbol{G}$ according to $\boldsymbol{G} = \boldsymbol{H}_2^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} + \boldsymbol{H}_1^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}}$ [VH11].

Consequently, in the limiting cases $\beta = 0$ and $\beta = 1$, ANOMAX directs all the energy onto the dominant eigenmode of one of the links, yielding effective channel matrices $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ that are rank one. For other values of β, both links are taken into account and the matrix $\boldsymbol{G}$ has a higher rank. However, for maximizing the norms of the effective channel matrices it is still best to focus the energy on the dominant eigenmode of the effective channel matrices as much as possible. Consequently, we typically have the situation that the effective channel matrices $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ are almost rank-one.

Therefore, ANOMAX should be combined with a single-stream transmission technique over the effective channels, e.g., DET. DET results in an SNR gain of $\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue of $\boldsymbol{H}_{i,j}^{(\mathrm{e})} \cdot \boldsymbol{H}_{i,j}^{(\mathrm{e})^{\mathrm{H}}}$ [PNG03]. Since the effective channels are almost rank-one we have $\left\|\boldsymbol{H}_{i,j}^{(\mathrm{e})}\right\|_{\mathrm{F}}^{2} \approx \lambda_{\max}$. We therefore conclude that ANOMAX results in an SNR improvement for DET, since the norm is maximized. However, since $\boldsymbol{G}$ also influences the noise power, we cannot formally state that ANOMAX *maximizes* the SNR. Yet, as the simulations demonstrate, ANOMAX yields a very good BER performance for single-stream transmission, see Section 16.5.

16.3.4. Rank-Restored (RR)-ANOMAX

As highlighted in the previous section, the norm-maximization that was used as a cost function for ANOMAX tends to concentrate the energy on the dominant eigenmodes of the effective channels, yielding channel matrices that are almost rank-one. This is desirable if the system design target is to maximize the diversity order for robust transmission with a low BER.

However, in terms of the system sum-rate, such a feature may be undesirable. If the terminals have multiple antennas, these can also be used to spatially multiplex data streams, which results in a significantly increased sum-rate. If the ANOMAX transmit strategy is used, the second eigenmode is very weak which results in an unsatisfactory sum-rate performance as we show in Section 16.5. In fact, to achieve the full multiplexing gain, the rank effective channel matrices should have $\min\{M_1, M_2, M_{\mathrm{R}}\}$ non-zero eigenmodes.

As we show in the sequel it is easy to devise a scheme that adapts the rank of the channels in order to inherit the good system performance of ANOMAX for low SNRs and restore the required rank for high SNRs. The corresponding approach is called “Rank-Restored (RR)-ANOMAX” [RH10a, RJH10].

The main idea is to start with the ANOMAX solution for $\boldsymbol{G}$, leave its singular vectors intact, and adjust its singular values depending on the required rank. To this end, let the SVD of

$\boldsymbol{G}_{\mathrm{ANOMAX}}$ be given by

$$\boldsymbol{G}_{\mathrm{ANOMAX}} = \boldsymbol{U}_{\mathrm{A}} \cdot \boldsymbol{\Sigma}_{\mathrm{A}} \cdot \boldsymbol{V}_{\mathrm{A}}^{\mathrm{H}} \tag{16.24}$$

Then the singular value profile can be adjusted via the vector $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \dots, \sigma_{M_{\mathrm{R}}}]^{\mathrm{T}}$ by defining

$$\boldsymbol{G}(\boldsymbol{\sigma}) = \boldsymbol{U}_{\mathrm{A}} \cdot \mathrm{diag}\{\boldsymbol{\sigma}\} \cdot \boldsymbol{V}_{\mathrm{A}}^{\mathrm{H}}, \tag{16.25}$$

where $\mathrm{diag}\{\boldsymbol{\sigma}\}$ is a diagonal matrix containing the elements of the vector $\boldsymbol{\sigma}$ on its main diagonal. Our new optimization problem therefore takes the following form

$$\max_{\boldsymbol{\sigma}} \ \mathrm{MMI}(\boldsymbol{G}(\boldsymbol{\sigma})), \text{ s.t. } \|\boldsymbol{\sigma}\|_2 = 1 \text{ and } \sigma_1 \geq \sigma_2 \geq \ldots \geq 0 \tag{16.26}$$

where $\mathrm{MMI}(\boldsymbol{G}(\boldsymbol{\sigma}))$ denotes the maximum mutual information achievable with the relay amplification matrix $\boldsymbol{G}(\boldsymbol{\sigma})$ which is introduced in (16.3). The constraints stem from the fact that $\boldsymbol{G}(\boldsymbol{\sigma})$ is normalized to Frobenius norm one and the singular values are ordered and non-negative.

Note that in the special case $\sigma_i = 1/\sqrt{M_{\mathrm{R}}}$, $\forall i$, the matrix $\sqrt{M_{\mathrm{R}}} \cdot \boldsymbol{G}(\boldsymbol{\sigma})$ becomes unitary. This represents the best approximation of $\sqrt{M_{\mathrm{R}}} \cdot \boldsymbol{G}_{\mathrm{ANOMAX}}$ by a unitary matrix in the Frobenius norm sense, commonly referred to as the Procrustes approximation (see [GvL80], chapter 12). The Procrustes approximation has been applied in array processing applications before, e.g., ESPRIT-based DOA estimation [ZS89].

The optimization problem in (16.26) can be simplified by taking into account the nature of our parameters. First of all, the norm constraint on $\boldsymbol{\sigma}$ can be used to reduce the search space to $M_{\mathrm{R}} - 1$ dimensions by optimizing over $\bar{\boldsymbol{\sigma}} = [\sigma_2/\sigma_1, \dots, \sigma_{M_{\mathrm{R}}}/\sigma_1]^{\mathrm{T}} \in \mathbb{R}^{M_{\mathrm{R}}-1}$, where each element of $\bar{\boldsymbol{\sigma}}$ is in $[0, 1]$. Secondly, the search space for $\bar{\boldsymbol{\sigma}}$ can be further reduced by exploiting the fact that the singular values are ordered, i.e., the i-th element of $\bar{\boldsymbol{\sigma}}$ is optimized in the interval between 0 and the current value of the $(i-1)$-th element of $\bar{\boldsymbol{\sigma}}$.

The optimization over $M_{\mathrm{R}} - 1$ real-valued parameters quickly becomes cumbersome for larger values of M_{R}. A typical result of RR-ANOMAX is depicted in Figure 16.1. Here, we consider uncorrelated Rayleigh fading channels with $M_1 = M_2 = 6$, $M_{\mathrm{R}} = 3$ and find the optimal singular value profile by solving (16.26) via an exhaustive search. The red lines depict the resulting profile of the squared singular values. We observe that for low SNRs, a low-rank solution is obtained and for increasing SNRs, eigenmodes are activated sequentially until eventually all singular values become equal for high SNRs. This is a very typical trend that we have found also in all other scenarios that have been investigated.

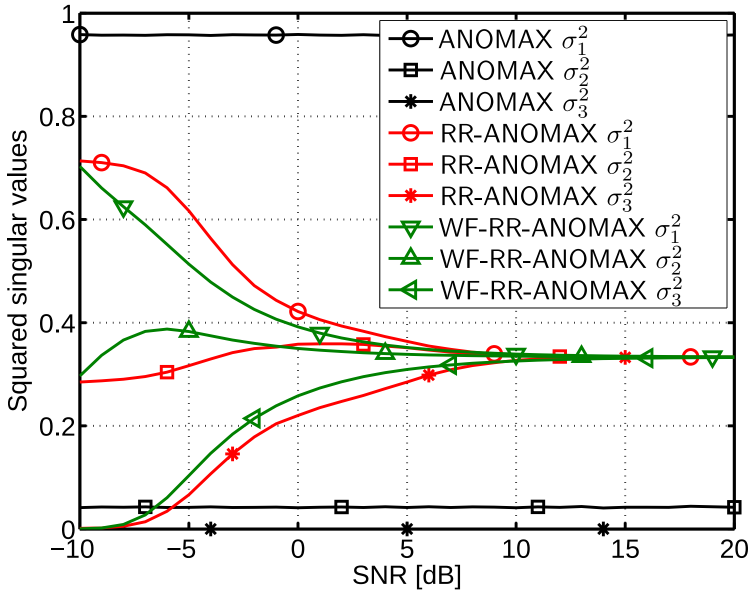


Figure 16.1.: Profile of squared singular values optimized via RR-ANOMAX and WF-RR-ANOMAX for $M_1 = M_2 = 6$, $M_{\mathrm{R}} = 3$. The original singular values of the ANOMAX solution are shown as a reference.

Note that such a behavior is very similar to the water-filling (WF) principle that is used to allocate power to eigenmodes in MIMO systems based on spatial multiplexing [Tel99]. It seems therefore possible to replace the optimization procedure by a closed-form WF-based heuristic. The resulting singular values do not perfectly match the ones obtained via RR-ANOMAX. However, according to out experience, the cost function (16.26) is not very sensitive to small changes in the singular values and hence the sum-rate achieved via this heuristic is always close to RR-ANOMAX based on the exhaustive search, which we demonstrate numerically in Section 16.5.

We provide an example for a very simple heuristic below. It has been developed by performing simulations and manually imitating the trend of the optimal singular value profile found via an exhaustive search. Consequently, it is specific to the assumptions that have been made for the simulations. In particular, we have assumed Rayleigh fading, reciprocity, and $P_{\mathrm{T},1} = P_{\mathrm{T},2} = 1$, for simplicity (see also Section 16.5).

The resulting WF-based heuristic rule to choose σ_k in (16.25) is given by

$$\sigma_k^2 = \left(\mu - \frac{P_{\mathrm{N,R}}}{\lambda_k}\right)_+, \; k = 1, 2, \dots, r, \tag{16.27}$$

where $P_{\mathrm{N,R}}$ represents the noise power at the relay, $(x)_+ = \max\{0, x\}$, and $\mu > 0$ is the water level chosen such that $\sum_{k=1}^{r} \sigma_k^2 = 1$. The coefficients λ_k represents the "virtual eigenvalue" profile which we can compute via

$$\lambda_k = (\sigma_{1,k} + \delta) \cdot (\sigma_{2,k} + \delta), \; k = 1, 2, \dots, r, \tag{16.28}$$

where δ is a positive constant to assure that we obtain r non-zero eigenvalues for high SNRs ($\delta = 1$ is used in the simulations) and $\sigma_{n,k}$ represents the k-th singular value of $\boldsymbol{H}_n$. The rank r is chosen heuristically as $r = \min\{M_{\mathrm{R}}, \min\{M_1, M_2\} + 1\}$. We display the resulting singular value profile using this heuristic in Figure 16.1 with the curves labeled "WF-RR-ANOMAX". Comparing the profile of the singular values obtained via WF-RR-ANOMAX and RR-ANOMAX in we conclude that they follow a similar trend. Again, note that WF-RR-ANOMAX is just one example for a heuristic. Depending on the specific system at hand, more sophisticated heuristics can be devised.

16.4. Rate-Maximization via Generalized Eigenvectors for Single-Antenna Terminals (RAGES)

In this section we present the RAGES scheme which is a semi-algebraic approach to find the sum-rate optimal relay amplification matrix for the special case of single-antenna terminals, first proposed in [RH10b]. We show that in this case the sum-rate can be expressed as a product of two Rayleigh quotients. This suggests that the optimal strategy is linked to generalized eigenvectors, which correspond to the local and global optima of Rayleigh quotients (cf. Section 3.2). Based on this idea, the rate-optimal relay amplification strategy can be found via the search over two real-valued scalar parameters. Since they both have a physical interpretation, the search can be efficiently implemented and a simplified 1-D search can be found which performs very close to optimal. We verify the optimality of RAGES by comparing it to the sum-rate optimal strategy found via the Polynomial Time DC (POTDC) algorithm which has been proposed by us in [KRVH12, KVRH12].

16.4.1. Problem statement

Let us first find a canonical expression for the sum-rate maximization problem in the special case of single-antenna terminals. As shown in Section 16.1.3, the SNR of each terminal can be written as $\eta_i = P_{\mathrm{R},i}/\tilde{P}_{\mathrm{N},i}$, where $P_{\mathrm{R},i}$ and $\tilde{P}_{\mathrm{N},i}$ are quadratic forms in $\boldsymbol{g}_\gamma \doteq \mathrm{vec}\{\boldsymbol{G}_\gamma\}$, cf. Lemma 16.1.4. Since the transmitted and received signals are scalars, the maximum mutual information for UT_i simplifies into

$$r_i = \frac{1}{2}\log_2(1+\eta_i), \tag{16.29}$$

where the factor 1/2 stems from the two time slots needed for the bidirectional transmission of information. Inserting $\eta_i = P_{\mathrm{R},i}/\tilde{P}_{\mathrm{N},i}$, we can write

$$r_i = \frac{1}{2}\log_2\left(1+\frac{P_{\mathrm{R},i}}{\tilde{P}_{\mathrm{N},i}}\right) = \frac{1}{2}\log_2\left(\frac{\tilde{P}_{\mathrm{N},i}+P_{\mathrm{R},i}}{\tilde{P}_{\mathrm{N},i}}\right) = \frac{1}{2}\log_2\left(\frac{\tilde{P}_{\mathrm{R},i}}{\tilde{P}_{\mathrm{N},i}}\right), \tag{16.30}$$

where we have defined $\tilde{P}_{\mathrm{R},i} = P_{\mathrm{R},i} + \tilde{P}_{\mathrm{N},i}$ for $i = 1,2$. Therefore, the sum of the maximum mutual information of both UTs $r_1 + r_2$ (hereafter simply referred to as the sum-rate for brevity) becomes

$$r_1 + r_2 = \frac{1}{2}\log_2\left(\frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}}\right) + \frac{1}{2}\log_2\left(\frac{\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},2}}\right) = \frac{1}{2}\log_2\left(\frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}}\cdot\frac{\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},2}}\right). \tag{16.31}$$

Inserting the results from Lemma 16.1.4 and performing some algebraic manipulations, we can show the following proposition:

Proposition 16.4.1. [RH10b] *The relay amplification matrix $\boldsymbol{G}_{\gamma,\mathrm{opt}}$ which maximizes the sum-rate shown in* (16.31) *subject to the power constraint $P_{\mathrm{T,R}} \leq P_{\mathrm{T,R}}^{\max}$ can be found by solving the following equivalent optimization problem*

$$\mathrm{vec}\{\boldsymbol{G}_{\gamma,\mathrm{opt}}\} = \sqrt{\frac{P_{\mathrm{T,R}}^{\max}}{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}}\cdot\boldsymbol{Q}\cdot\boldsymbol{g}_{\mathrm{opt}}}}\cdot\boldsymbol{g}_{\mathrm{opt}} \quad \textit{where}$$
$$\boldsymbol{g}_{\mathrm{opt}} = \arg\max_{\boldsymbol{g}} \frac{\boldsymbol{g}^{\mathrm{H}}\cdot\tilde{\boldsymbol{K}}_1\cdot\boldsymbol{g}}{\boldsymbol{g}^{\mathrm{H}}\cdot\tilde{\boldsymbol{J}}_1\cdot\boldsymbol{g}}\cdot\frac{\boldsymbol{g}^{\mathrm{H}}\cdot\tilde{\boldsymbol{K}}_2\cdot\boldsymbol{g}}{\boldsymbol{g}^{\mathrm{H}}\cdot\tilde{\boldsymbol{J}}_2\cdot\boldsymbol{g}}, \tag{16.32}$$

and the positive definite[4] *matrices* $\tilde{\boldsymbol{K}}_1, \tilde{\boldsymbol{K}}_2, \tilde{\boldsymbol{J}}_1, \tilde{\boldsymbol{J}}_2$ *are given by*

$$\tilde{\boldsymbol{K}}_1 = \boldsymbol{K}_{2,1} \cdot P_{\mathrm{T},2} + \tilde{\boldsymbol{J}}_1, \quad \tilde{\boldsymbol{K}}_2 = \boldsymbol{K}_{1,2} \cdot P_{\mathrm{T},1} + \tilde{\boldsymbol{J}}_2,$$
$$\tilde{\boldsymbol{J}}_1 = \boldsymbol{J}_1 + \frac{P_{\mathrm{N},1}}{P_{\mathrm{T,R}}^{\max}} \cdot \boldsymbol{Q}, \quad \tilde{\boldsymbol{J}}_2 = \boldsymbol{J}_2 + \frac{P_{\mathrm{N},2}}{P_{\mathrm{T,R}}^{\max}} \cdot \boldsymbol{Q}.$$

Note that $\boldsymbol{Q}$*,* $\boldsymbol{J}_i$*, and* $\boldsymbol{K}_{i,j}$ *have been defined in Lemma 16.1.4.*

Proof: cf. Appendix E.8.

Proposition 16.4.1 shows that finding the sum-rate optimal relay amplification matrix is equivalent to the (unconstrained) maximization of the product of two Rayleigh quotients.

Note that [LYC08] formulates the sum-rate maximization problem in two-way relaying with a MIMO AF relay and single-antenna terminals also in terms of a product of Rayleigh quotients. However, the expression that [LYC08] derives is only valid for high SNRs, whereas (16.32) is exact for arbitrary SNRs.

16.4.2. RAGES approach: Generalized Eigenvectors

To derive the link between (16.32) and generalized eigenvectors we start with the necessary condition for optimality that the gradient of (16.32) vanishes. Therefore, if we find all vectors $\boldsymbol{g}$ where the gradient of the cost function is zero, the global optimum must be one of them. By using the product rule and the chain rule of differentiation we find that this condition can be expressed as

$$\frac{\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},1}\tilde{P}_{\mathrm{N},2}} \cdot \tilde{\boldsymbol{K}}_1 \cdot \boldsymbol{g} + \frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}\tilde{P}_{\mathrm{N},2}} \cdot \tilde{\boldsymbol{K}}_2 \cdot \boldsymbol{g} = \frac{\tilde{P}_{\mathrm{R},1}\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},1}^2\tilde{P}_{\mathrm{N},2}} \cdot \tilde{\boldsymbol{J}}_1 \cdot \boldsymbol{g} + \frac{\tilde{P}_{\mathrm{R},1}\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},1}\tilde{P}_{\mathrm{N},2}^2} \cdot \tilde{\boldsymbol{J}}_2 \cdot \boldsymbol{g} \tag{16.33}$$

Rearranging (16.33) we obtain

$$\left(\tilde{\boldsymbol{K}}_1 + \rho_{\mathrm{sig}} \cdot \tilde{\boldsymbol{K}}_2\right) \cdot \boldsymbol{g} = \frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}} \cdot \left(\tilde{\boldsymbol{J}}_1 + \rho_{\mathrm{noi}} \cdot \tilde{\boldsymbol{J}}_2\right) \cdot \boldsymbol{g} \tag{16.34}$$

where ρ_{sig} and ρ_{noi} are defined via

$$\rho_{\mathrm{sig}} = \frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{R},2}} \text{ and } \rho_{\mathrm{noi}} = \frac{\tilde{P}_{\mathrm{N},1}}{\tilde{P}_{\mathrm{N},2}}. \tag{16.35}$$

[4]While positive semi-definiteness follows directly from their definition, to be positive definite we require the noise covariance matrix at the relay $\boldsymbol{R}_{\mathrm{N,R}}$ to be full rank.

From (16.34) we conclude that the optimal $\boldsymbol{g}$ must be a generalized eigenvector of the matrices $\left(\tilde{\boldsymbol{K}}_1 + \rho_{\rm sig} \cdot \tilde{\boldsymbol{K}}_2\right)$ and $\left(\tilde{\boldsymbol{J}}_1 + \rho_{\rm noi} \cdot \tilde{\boldsymbol{J}}_2\right)$ and that the corresponding generalized eigenvalue is given by $\frac{\tilde{P}_{\rm R,1}}{\tilde{P}_{\rm N,1}}$ which is logarithmically proportional to r_1, i.e., $r_1 = \frac{1}{2}\log_2 \frac{\tilde{P}_{\rm R,1}}{\tilde{P}_{\rm N,1}}$. However, the matrices contain the parameters $\rho_{\rm sig}$ and $\rho_{\rm noi}$ which also depend on $\boldsymbol{g}$ and are hence not known in advance. Therefore, we still need to optimize over these two parameters. Compared to the original problem of finding a complex-valued $M_{\rm R} \times M_{\rm R}$ matrix, optimizing over two real-valued scalar parameters is already significantly simpler. The following subsections show how to simplify this 2-D search even further.

Note that [LYC08] also mentions the link to generalized eigenvectors for their approximate high-SNR expression for the sum-rate. However, [LYC08] proposes to find the generalized eigenvector by a General Power Iterative (GPI) algorithm, which may require many iterations to converge and does not give any guarantee for optimality. In contrast, the 2-D RAGES solution is sum-rate optimal and can be efficiently implemented, as discussed below.

16.4.3. Bounds on the parameters

Since both parameters $\rho_{\rm sig}$ and $\rho_{\rm noi}$ have a physical interpretation we can easily find lower and upper bounds for them in order to limit the search space that has to be tested. For instance, $\rho_{\rm noi}$ can be expanded into

$$\rho_{\rm noi} = \frac{\tilde{P}_{\rm N,1}}{\tilde{P}_{\rm N,2}} = \frac{\boldsymbol{g}^{\rm H}\boldsymbol{J}_1\boldsymbol{g} + P_{\rm N,1}}{\boldsymbol{g}^{\rm H}\boldsymbol{J}_2\boldsymbol{g} + P_{\rm N,2}}. \tag{16.36}$$

The quadratic forms are bounded by using the fact that for any Hermitian matrix $\boldsymbol{R}$ we have [Str93]

$$\lambda_{\min}(\boldsymbol{R}) \cdot \|\boldsymbol{g}\|_2^2 \leq \boldsymbol{g}^{\rm H} \cdot \boldsymbol{R} \cdot \boldsymbol{g} \leq \lambda_{\max}(\boldsymbol{R}) \cdot \|\boldsymbol{g}\|_2^2 \tag{16.37}$$

where $\lambda_{\min}$ and $\lambda_{\max}$ are the smallest and the largest eigenvalues of $\boldsymbol{R}$, respectively. From the definition of $\boldsymbol{J}_i$ in Lemma 16.1.4 it follows that $\lambda_{\min}(\boldsymbol{J}_1) = 0$ and $\lambda_{\max}(\boldsymbol{J}_1) = \lambda_{\max}(\boldsymbol{R}_{\rm N,R}) \cdot {\alpha_i^{(\rm b)}}^2$, where $\alpha_i^{(\rm b)}$ is a short hand notation for $\left\|\boldsymbol{h}_i^{(\rm b)}\right\|_2$. Furthermore, we have $\lambda_{\max}(\boldsymbol{R}_{\rm N,R}) \leq P_{\rm N,R} \cdot M_{\rm R}$ in general and for white noise at the relay the tighter condition $\lambda_{\max}(\boldsymbol{R}_{\rm N,R}) = P_{\rm N,R}$. These relations can be used to bound (16.36): We find an upper bound by upper-bounding the enumerator and lower-bounding the denominator. The lower bound is found by lower-bounding the enumerator and upper-bounding the denominator. This yields

$$\rho_{\rm noi} \leq \frac{P_{\rm N,R}}{P_{\rm N,2}} \cdot M_{\rm R} \cdot {\alpha_1^{(\rm b)}}^2 \cdot \xi_g^2 + \frac{P_{\rm N,1}}{P_{\rm N,2}} \tag{16.38}$$

$$\rho_{\rm noi} \geq \left(\frac{P_{\rm N,R}}{P_{\rm N,1}} \cdot M_{\rm R} \cdot \alpha_2^{(\rm b)^2} \cdot \xi_g^2 + \frac{P_{\rm N,2}}{P_{\rm N,1}} \right)^{-1} \tag{16.39}$$

where $\xi_g^2 = \|\boldsymbol{g}\|_2^2$. Moreover, $M_{\rm R}$ can be dropped if the noise at the relay is white, i.e., in this case the bounds tighten to $P_{\rm N,1}/(P_{\rm N,R} \cdot \alpha_2^{(\rm b)^2} \cdot \xi_g^2 + P_{\rm N,2}) \leq \rho_{\rm noi} \leq (P_{\rm N,R} \cdot \alpha_1^{(\rm b)^2} \cdot \xi_g^2 + P_{\rm N,1})/P_{\rm N,2}$. However, we still need an upper bound for ξ_g^2. Due to the relay power constraint we have $\boldsymbol{g}^{\rm H} \cdot \boldsymbol{Q} \cdot \boldsymbol{g} = P_{\rm T,R}^{\rm max}$. From this condition we can derive the bound $\xi_g^2 \leq P_{\rm T,R}^{\rm max}/\lambda_{\rm min}(\boldsymbol{Q})$. However, this is a very loose bound since for white noise at the relay we have $\lambda_{\rm min}(\boldsymbol{Q}) = P_{\rm N,R}$ and for arbitrary relay noise covariance matrices no lower bound exists (the infimum over $\lambda_{\rm min}$ is zero). This bound is so loose because it is extremely pessimistic: it measures the norm of $\boldsymbol{g}$ in the case where only noise is amplified and no power is put on the eigenvalues related to the signals of interest. However, such a case is practically irrelevant since it corresponds to a sum-rate equal to zero. The optimal strategy at the relay is much more likely to focus on the desired signal component. We therefore propose to replace ξ_g^2 in (16.38) and (16.39) by[5] $P_{\rm T,R}^{\rm max}/\lambda_2(\boldsymbol{R}_{\rm R})$, where $\lambda_2(\boldsymbol{R}_{\rm R})$ is the second-largest eigenvalue of $\boldsymbol{R}_{\rm R}$, which is chosen since the desired signal component in $\boldsymbol{R}_{\rm R}$ is of rank two.

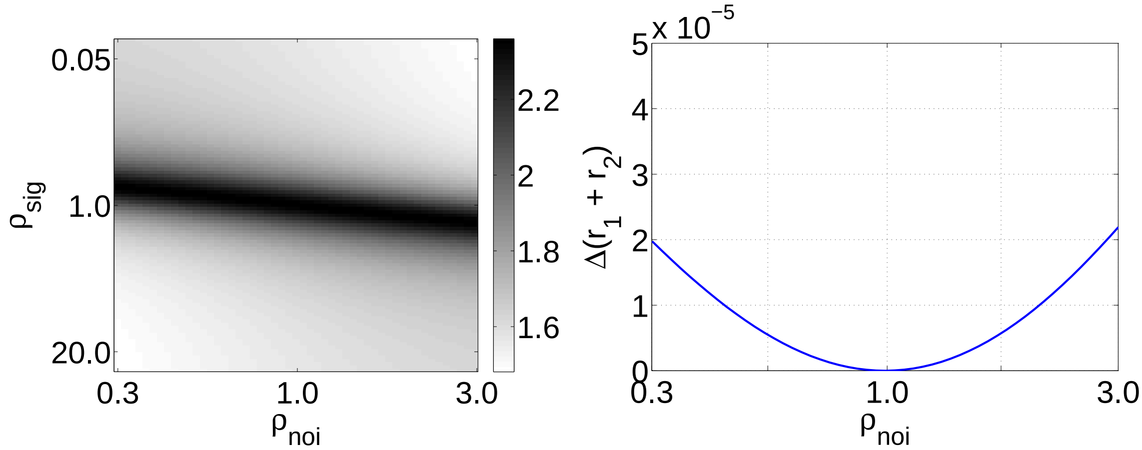
In a similar manner, $\rho_{\rm sig}$ can be bounded. In this case, enumerator and denominator have the additional terms $P_{\rm T,2} \cdot \boldsymbol{g}^{\rm H}\boldsymbol{K}_{2,1}\boldsymbol{g}$ and $P_{\rm T,1} \cdot \boldsymbol{g}^{\rm H}\boldsymbol{K}_{1,2}\boldsymbol{g}$, respectively. A pessimistic (loose) bound is obtained by bounding these two terms independently, i.e., $0 \leq \boldsymbol{g}^{\rm H}\boldsymbol{K}_{2,1}\boldsymbol{g} \leq \xi_g^2 \alpha_2^{(\rm f)^2} \alpha_1^{(\rm b)^2}$ and $0 \leq \boldsymbol{g}^{\rm H}\boldsymbol{K}_{1,2}\boldsymbol{g} \leq \xi_g^2 \alpha_1^{(\rm f)^2} \alpha_2^{(\rm b)^2}$. This yields

$$\rho_{\rm sig} \leq \frac{P_{\rm T,2}}{P_{\rm N,2}} \alpha_2^{(\rm f)^2} \alpha_1^{(\rm b)^2} \xi_g^2 + \frac{P_{\rm N,R}}{P_{\rm N,2}} M_{\rm R} \alpha_1^{(\rm b)^2} \xi_g^2 + \frac{P_{\rm N,1}}{P_{\rm N,2}} \tag{16.40}$$

$$\rho_{\rm sig} \geq \left(\frac{P_{\rm T,1}}{P_{\rm N,1}} \alpha_1^{(\rm f)^2} \alpha_2^{(\rm b)^2} \xi_g^2 + \frac{P_{\rm N,R}}{P_{\rm N,1}} M_{\rm R} \alpha_2^{(\rm b)^2} \xi_g^2 + \frac{P_{\rm N,2}}{P_{\rm N,1}} \right)^{-1}. \tag{16.41}$$

Again, these bounds are pessimistic since they assume that there exists an optimal relay strategy for which $P_{\rm R,1} = P_{\rm T,2} \cdot \alpha_2^{(\rm f)^2} \cdot \alpha_1^{(\rm b)^2} \cdot \xi_g^2$ but $P_{\rm R,2} = 0$, i.e., the rate of the second terminal is equal to zero. However, it is typically sum-rate optimal to have significantly more balanced rates between the two users. In fact, for a "symmetric" scenario where $P_{\rm T,1} = P_{\rm T,2}$, $\boldsymbol{h}_i^{(\rm f)} = \boldsymbol{h}_i^{(\rm b)}$ for $i = 1, 2$, and $\alpha_1^{(\rm f)} = \alpha_2^{(\rm f)}$ we always have $P_{\rm R,1} = P_{\rm R,2}$ at the optimal point. Therefore, these bounds can be further tightened if a priori knowledge about the scenario is available. For instance, if we know that the received powers differ by no more than a factor of r, i.e., $P_{\rm R,1}/r \leq P_{\rm R,2} \leq P_{\rm R,1} \cdot r$, then the lower bounds for $\boldsymbol{g}^{\rm H}\boldsymbol{K}_{i,j}\boldsymbol{g}$ can be improved according to

[5] We have observed in all our simulations that this value poses indeed an upper bound on the norm of the optimal solution $\boldsymbol{g}_{\rm opt}$, however, no analytical proof of this fact could be found.



(a) Sum-rate $r_1 + r_2$ vs. ρ_{sig} and ρ_{noi}

(b) Relative change in sum-rate $r_1 + r_2$ vs. ρ_{noi}, optimizing over ρ_{sig} for every choice of ρ_{noi}

Figure 16.2.: Impact of ρ_{sig} and ρ_{noi} on the sum-rate for $M_{\text{R}} = 6$, $P_{\text{T},1} = P_{\text{T},2} = P_{\text{T,R}}^{\max} = 1$, $P_{\text{N},1} = P_{\text{N},2} = P_{\text{N,R}} = 0.1$.

$P_{\text{T},2}\xi_g^2\alpha_1^{(\text{f})^2}\alpha_2^{(\text{b})^2}/r \le P_{\text{T},1}\boldsymbol{g}^{\text{H}}\boldsymbol{K}_{2,1}\boldsymbol{g} \le P_{\text{T},1}\xi_g^2\alpha_2^{(\text{f})^2}\alpha_1^{(\text{b})^2}$ and $P_{\text{T},1}\xi_g^2\alpha_2^{(\text{f})^2}\alpha_1^{(\text{b})^2}/r \le P_{\text{T},2}\boldsymbol{g}^{\text{H}}\boldsymbol{K}_{1,2}\boldsymbol{g} \le P_{\text{T},2}\xi_g^2\alpha_1^{(\text{f})^2}\alpha_2^{(\text{b})^2}$.

16.4.4. Efficient 2-D and 1-D search

Once the search space for ρ_{sig} and ρ_{noi} has been fixed we need to find the maximum via a 2-D search. In general, a 2-D exhaustive search can be computationally demanding. However, as we show in the sequel, for the problem at hand this search can be implemented efficiently. These efficient implementations are heuristic since they rely on properties of the cost function that are apparent by visual inspection but due to the complex dependency of the sum-rate on the scalar parameters they cannot be proven analytically. However, as we show in simulations the resulting algorithm performs well in practice.

To this end, Figure 16.2a demonstrates a typical example of the sum-rate $r_1 + r_2$ as a function of ρ_{sig} and ρ_{noi}. For this example we have chosen $M_{\text{R}} = 6$, $P_{\text{T},1} = P_{\text{T},2} = P_{\text{T,R}}^{\max} = 1$, $P_{\text{N},1} = P_{\text{N},2} = P_{\text{N,R}} = 0.1$ and we have drawn the channel vectors from an uncorrelated Rayleigh fading distribution assuming reciprocity. By visual inspection this sample cost function shows two interesting properties. Firstly, it is a quasi-convex function in ρ_{noi} and ρ_{sig} (i.e., it does not have any local maxima [BV04]) which allows for efficient (quasi-convex) optimization tools for finding its maximum. Albeit this property is only demonstrated for one example it was always present in our numerical evaluations even when largely varying all system parameters.

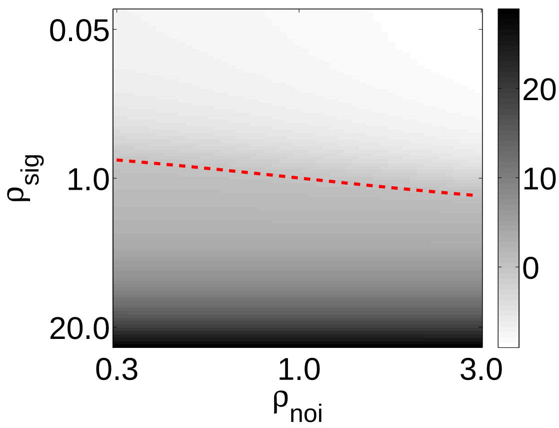


Figure 16.3.: Cost function $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}})$ using the same dataset as in Figure 16.2a. The red line indicates the points where $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}}) = 0$.

Secondly, for every value of ρ_{noi} the corresponding maximization over ρ_{sig} yields one maximal value which depends on ρ_{noi} only very weakly. This is illustrated by Figure 16.2b where we display the relative change of our cost function $r_1 + r_2$ for different choices of ρ_{noi}, each time optimizing it over ρ_{sig}. The displayed values represent the relative decrease of the cost function compared to the global optimum, i.e., for the worst choice of ρ_{noi}, the achieved sum-rate is about $2 \cdot 10^{-5} = 0.002$ % lower than for the best choice of ρ_{noi}. Consequently, we propose to replace the 2-D search over ρ_{sig} and ρ_{noi} by a 1-D search over ρ_{sig} for one fixed value of ρ_{noi} (e.g., the geometric mean of the upper and the lower bound).

Instead of performing the search directly over the original cost function $r_1 + r_2$ we can find an even simpler cost function by using the physical meaning of our two search parameters. To this end, let

$$\hat{\rho}_{\mathrm{sig}}(\boldsymbol{g}) \doteq \frac{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_1 \cdot \boldsymbol{g}}{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_2 \cdot \boldsymbol{g}} \tag{16.42}$$

where $\boldsymbol{g}$ is the relay weight vector at the current search point $(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}})$. Then we know that in the optimal point $\boldsymbol{g}_{\mathrm{opt}}$ we have $\hat{\rho}_{\mathrm{sig}}(\boldsymbol{g}_{\mathrm{opt}}) = \rho_{\mathrm{sig}}$. This can be used to construct a new cost function $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}}) \doteq \hat{\rho}_{\mathrm{sig}}(\boldsymbol{g}) - \rho_{\mathrm{sig}}$, where $\boldsymbol{g}$ is the dominant eigenvector of (16.34) at the current search point $(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}})$. Using the same data set as before we display the corresponding shape of $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}})$ in Figure 16.3. The red line indicates the set of points where $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}}) = 0$. We observe that for every value of ρ_{noi}, $A_{\mathrm{sig}}(\rho_{\mathrm{sig}}, \rho_{\mathrm{noi}})$ is a monotonic function in ρ_{sig}. Therefore, we can use the bisection method to find the zero crossing in ρ_{sig}

which coincides with the sum-rate-optimal ρ_{sig} for the given ρ_{noi}.

In the special case where the noise at the relay is white, i.e., $\boldsymbol{R}_{\mathrm{N,R}} = P_{\mathrm{N,R}} \cdot \boldsymbol{I}_{M_{\mathrm{R}}}$, and the channels are reciprocal, the complexity can be further reduced. In this case, it is easy to show that the optimal matrix $\boldsymbol{G}$ has the form [ZLCC09]

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{h}_1 & \boldsymbol{h}_2 \end{bmatrix}^* \cdot \boldsymbol{B} \cdot \begin{bmatrix} \boldsymbol{h}_1 & \boldsymbol{h}_2 \end{bmatrix}^{\mathrm{H}}, \tag{16.43}$$

where $\boldsymbol{B} \in \mathbb{C}^{2\times 2}$. Translated to the RAGES approach it means that instead of searching over M_{R}^2-dimensional vectors $\boldsymbol{g}$, we can reduce the search to the set of 4-dimensional vectors $\boldsymbol{b} = \operatorname{vec}\{\boldsymbol{B}\}$. The details of this approach are shown in Appendix E.9.

16.4.5. Summary

We conclude that the RAGES approach simplifies the optimization over a complex-valued $M_{\mathrm{R}} \times M_{\mathrm{R}}$ matrix into the optimization over two real-valued parameters which both have a physical interpretation. Even more so, the 2-D search can be simplified into a 1-D search by fixing one of the parameters. The loss incurred to this step is typically small, in the example provided above it was only 0.002 % but even varying the system parameters largely and using many random trials we never found a relative difference higher than a few percent.

Moreover, the 1-D search can be efficiently implemented by exploiting the quasi-convexity of $r_1 + r_2$ (e.g., using a branch and bound algorithm) or the monotonicity of A_{sig} (e.g., using the bisection method). Again, these properties are only demonstrated by examples but we found in all our simulations that the resulting algorithm yields a sum-rate very close to the optimum found via exhaustive search. This comparison is further illustrated in Section 16.5 with numerical simulations.

16.5. Simulation Results

In this section we present numerical results obtained via Monte Carlo simulations to evaluate the performance of the proposed schemes. For all simulation results we assume that reciprocity is valid, i.e., $\boldsymbol{H}_i^{(\mathrm{f})} = \boldsymbol{H}_i^{(\mathrm{b})^{\mathrm{T}}} = \boldsymbol{H}_i$ for $i = 1, 2$. The channel matrices $\boldsymbol{H}_i$ are drawn according to a correlated Rayleigh fading distribution with Kronecker correlation. Therefore, the elements of $\boldsymbol{H}_i$ are zero mean circularly symmetric complex Gaussian random variables with variance denoted by α_i^2 for $i = 1, 2$. Moreover, the one-sided covariance matrices of $\boldsymbol{H}_i$ are given by

$$\frac{1}{M_i} \cdot \mathbb{E}\left\{\boldsymbol{H}_i \cdot \boldsymbol{H}_i^{\mathrm{H}}\right\} = \alpha_i^2 \cdot \boldsymbol{R}_{\mathrm{R}} \in \mathbb{C}^{M_{\mathrm{R}} \times M_{\mathrm{R}}},\ i = 1, 2 \tag{16.44}$$

$$\frac{1}{M_{\mathrm{R}}} \cdot \mathbb{E}\left\{\boldsymbol{H}_i^{\mathrm{H}} \cdot \boldsymbol{H}_i\right\} = \alpha_i^2 \cdot \boldsymbol{R}_i \in \mathbb{C}^{M_i \times M_i}, \; i = 1, 2, \tag{16.45}$$

where $\boldsymbol{R}_i$ and $\boldsymbol{R}_{\mathrm{R}}$ represent the normalized spatial covariance matrices at terminal i and at the relay, respectively. They are constructed such that their main diagonal elements are equal to one and all off-diagonal elements have magnitude ρ_i and ρ_{R}, respectively. If not stated otherwise, the variances are set to $\alpha_1^2 = \alpha_2^2 = 1$ and the transmit powers are set to $P_{\mathrm{T},1} = P_{\mathrm{T},2} = P_{\mathrm{T,R}} = 1$. Moreover, the noise powers are assumed to be equal, i.e., $P_{\mathrm{N},1} = P_{\mathrm{N},2} = P_{\mathrm{N,R}} = \sigma_{\mathrm{n}}^2$. The SNR is then defined as $\mathrm{SNR} = \sigma_{\mathrm{n}}^{-2}$.

Figures 16.4a and 16.4b compare the link performance of the ANOMAX scheme introduced in Section 16.3 with the Dual Channel Matching (DCM) [VH11] as well as the ZF and MMSE schemes from [UK08]. Moreover, the curve labeled "DFT" corresponds to the case where the relay uses a scaled $M_{\mathrm{R}} \times M_{\mathrm{R}}$ DFT matrix (cf. Appendix A.2)[6]. The terminals transmit a single data stream over the effective channel matrices $\boldsymbol{H}_{i,j}^{(\mathrm{e})}$ via dominant eigenmode transmission (DET). As a modulation scheme, uncoded QPSK is used. For Figure 16.4a, the terminals are equipped with $M_1 = M_2 = 4$ antennas and the relay used $M_{\mathrm{R}} = 2$ antennas. On the other hand, for Figure 16.4b, the antenna configuration is changed to $M_1 = M_2 = 2$ and $M_{\mathrm{R}} = 5$, respectively. The solid curves represent the case where the channels are perfectly known, whereas for the dashed curves, the channels are estimated using the SLS-based iterative refinement for TENCE (introduced in Section 15.4) and $N_{\mathrm{P,D}} = 2$ dedicated pilots (cf. Section 15.5.4). We clearly see that ANOMAX outperforms all other schemes for perfect as well as for imperfect CSI.

Figures 16.5a and 16.5b depict the maximum mutual information defined in (16.3) for $M_1 = M_2 = 2$ and $M_{\mathrm{R}} = 4$. No spatial correlation at the UTs is assumed, i.e., the correlation coefficients at the UTs are set to $\rho_1 = \rho_2 = 0$. For Figure 16.5a, we also set $\rho_{\mathrm{R}} = 0$, whereas for Figure 16.5b we introduce spatial correlation at the relay by setting $\rho_{\mathrm{R}} = 0.9$. As an upper bound for the maximum mutual information we depict the optimal relay amplification matrix obtained via a gradient search, as described in [ZRH12a]. This iterative algorithm finds the optimal relay strategy, however, it is impractical since the number of required gradient steps may be very large (in the order of millions), leading to a very high computational complexity. We notice that while ANOMAX performs well for low SNRs, it fails to provide the full spatial multiplexing gain for high SNRs, since it favors low-rank solutions. This effect is mitigated by the RR-ANOMAX scheme introduced in Section 16.3.4, which restores the required rank for high SNRs and thereby achieves a performance close to the upper bound. The WF-based heuristic for RR-ANOMAX performs nearly as good. As evident from Figure 16.5b, the MMSE

[6] Note that the performance of using the scaled DFT matrix is identical to using any other fixed orthogonal matrix, e.g., a scaled identity matrix.

and ZF transceivers are sensitive to spatial correlation at the relay, since they spatially cancel the self-interference.

In Figure 16.6 we investigate the "near-far" robustness of the different relay amplification strategies by varying the path gain of UT_2 (α_2) for fixed path gain of UT_1 ($\alpha_1 = 1$). Values of α_2 less than one correspond to a scenario where UT_2 is farther from the relay than UT_1. However, since the terminals' signals go through both channels $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$, both terminals are affected in a similar manner. RR-ANOMAX and WF-RR-ANOMAX perform equally well with an almost constant gap to the upper bound.

Figure 16.7 demonstrates the effect of increasing the number of antennas at the relay. We consider uncorrelated Rayleigh fading with $M_1 = M_2 = 6$ antennas at the UTs. RR-ANOMAX is displayed only for $M_{\mathrm{R}} = 2, 3, 4$, since the exhaustive search needed for RR-ANOMAX becomes computationally prohibitive for $M_{\mathrm{R}} > 4$. We observe that the sum-rate for ANOMAX increases only very little with increasing M_{R} which is due to the fact that ANOMAX tends to concentrate the energy on the dominant eigenmode of the channels. The algebraically simple WF-RR-ANOMAX technique is again very close to the upper bound.

The final set of simulation results shows the performance of the RAGES scheme for the special case of single-antenna terminals introduced in Section 16.4. We compare the sum-rate optimal semi-algebraic 2-D RAGES strategy with the simplified 1-D RAGES strategy. For reference, we also show the sum-rate optimal Polynmial Time DC (POTDC) based approach from [KVRH12, KRVH12] as well as the ANOMAX scheme from Section 16.3.1, the ZF and MMSE transceivers from [UK08], and the scaled DFT matrix. In Figure 16.8, we set $M_{\mathrm{R}} = 3$, $\rho_{\mathrm{R}} = 0$ and vary the SNR from 5 to 25 dB. On the other hand, for Figure 16.9, we fix the SNR to 0 dB and simulate a "near-far" effect by moving the relay. The "normalized relay position d" we show represents the distance between the relay and UT_1 divided by the distance between UT_2 and UT_1, assuming all three are located on one line. Therefore, $d = 0.5$ corresponds to the case where the relay is placed in the middle, whereas $d > 0.5$ means it is closer to UT_2 than to UT_1. We assume a path loss exponent of $n = 3$, i.e., the channel variances are chosen such that $\alpha_1^2/\alpha_2^2 = (1-d)^3/d^3$. Both simulation results confirm that 2-D RAGES and POTDC coincide, which is expected since they both yield the sum-rate optimal strategy in this setting. Moreover, 1-D RAGES performs as good as 2-D RAGES despite being computationally much less demanding. The fully algebraic ANOMAX scheme is also quite close to the maximum achievable sum-rate.

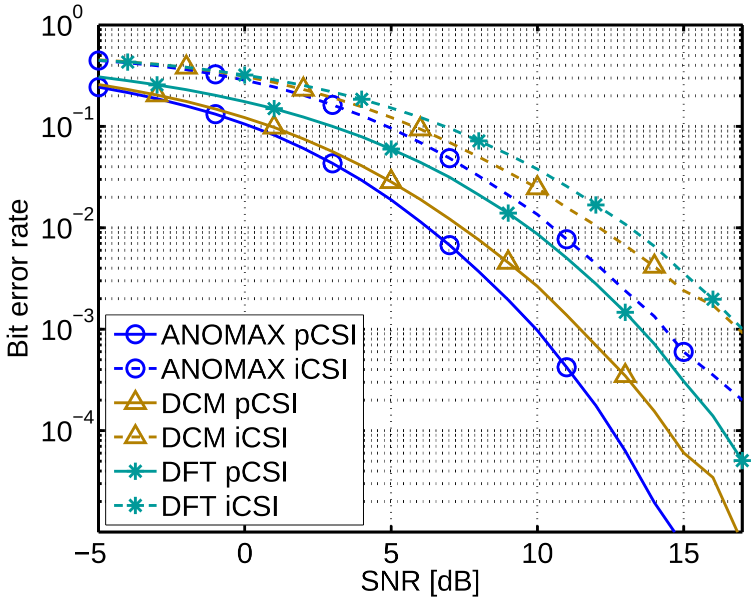
16.6. Summary

In this chapter of the thesis, we have discussed the choice of the relay amplification matrix for two-way relaying with a MIMO AF relay. After cancellation of the self-interference, the system is decomposed into two parallel single-user MIMO systems. Their effective channel matrices as well as their noise covariance matrices depend on the choice of the relay amplification matrix. Therefore, finding an optimal relaying strategy with respect to a suitable performance metric, e.g., sum-rate or bit error rate is a mathematically challenging task.

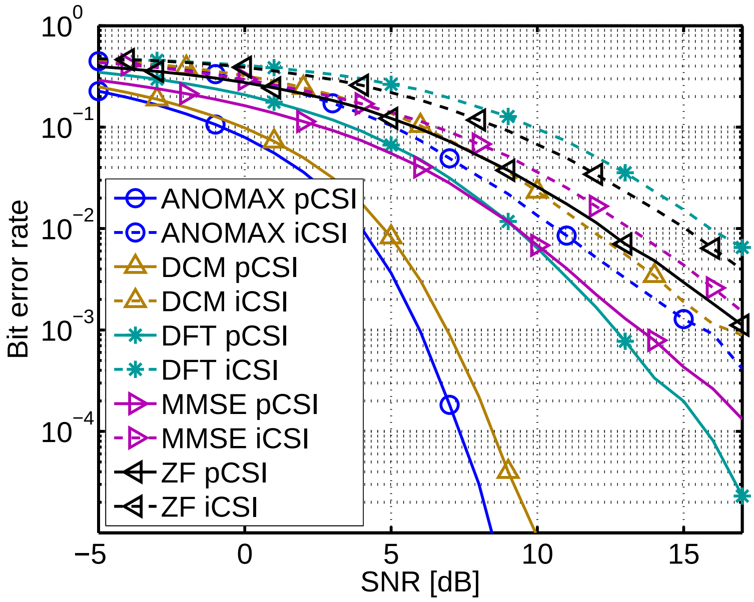
For the general MIMO case, we have introduced the Algebraic Norm Maximizing (ANOMAX) transmit strategy [RH09a] ANOMAX targets the maximization of the weighted sum of the Frobenius norms of the effective channel matrices. The resulting solution for the relay amplification matrix $\boldsymbol{G}$ is very simple to compute in closed-form. It tends to focus the energy in the effective channels on their dominant eigenmodes which results in a considerable SNR improvement for single-stream transmission. This results in a very good BER performance, providing reliable communication links and high diversity orders.

However, if we look at the maximum mutual information for higher Signal to Noise Ratios, the drawback of this approach becomes obvious: the low-rank nature of the effective channels is not suitable to support multiplexing of several data streams, resulting in a low spatial multiplexing gain. Therefore, for scenarios where the design target is to maximize the data rate, we provide the RR-ANOMAX algorithm [RH10a] as a simple extension to ANOMAX. By adjusting the singular value profile dependent on the current SNR, we can restore the required rank for high SNRs, yielding a performance that inherits the benefits of ANOMAX for low SNRs and still provides the full spatial multiplexing gain of $\min\{M_1, M_2, M_{\rm R}\}$ for high SNRs. This adaptation can be performed in a heuristic manner. We provide an example for a heuristic inspired by the Water Filling (WF) principle [Tel99]. Simulations demonstrate that the heuristic WF-RR-ANOMAX performs close to the upper bound while being very simple to compute. These simulations also show that WF-RR-ANOMAX performs many existing relay amplification strategies, such as the Dual Channel Matching (DCM) or the ZF and the MMSE transceivers.

As a third contribution, we consider the special case of single-antenna terminals. In this case, the sum-rate of the system can be explicitly expressed as a product of Rayleigh quotients. We show that such an optimization problem can be solved via Generalized Eigenvectors via the RAGES (Rate Maximization via Generalized Eigenvectors for Single-Antenna terminals) scheme. RAGES simplifies the search for the matrix $\boldsymbol{G}$ to a search over two real-valued scalar parameters. Since both have a physical interpretation, bounds for the search interval are available and the search can be efficiently implemented. We show that a simplified 1-D search where one of the two parameters is ignored performs just as well as the 2-D search, resulting in a very low computational complexity. Moreover, for the special case of reciprocal channels and white noise at the relay, the complexity can be further reduced by projecting the $M_{\mathrm{R}}^2 \times M_{\mathrm{R}}^2$ problem onto an equivalent 2×2 problem, rendering the search for the correct generalized eigenvector independent of M_{R}. Simulation results demonstrate that 2-D RAGES is in fact sum-rate optimal by comparing it to the Polynomial Time DC (POTDC) algorithm introduced by us in [KRVH12, KVRH12]. POTDC is a polynomial-time algorithm for solving a class of Difference of Convex functions (DC) problems that are in general NP-hard. As we show in [KRVH12, KVRH12], the maximization of the product of two Rayleigh quotients can be reformulated as DC problem where the cost function is a difference of logarithmic functions. POTDC solves this problem by approximating the log-function locally by a linear function in an iterative way. Thus, it solves a sequence of relaxed problems which have the same complexity as a semi-definite programming (SDP) problem. As we show in [KRVH12], POTDC is guaranteed to converge and its solution is guaranteed to be at least locally optimal, with strong numerical evidence for global optimality (though this has not been proven yet).

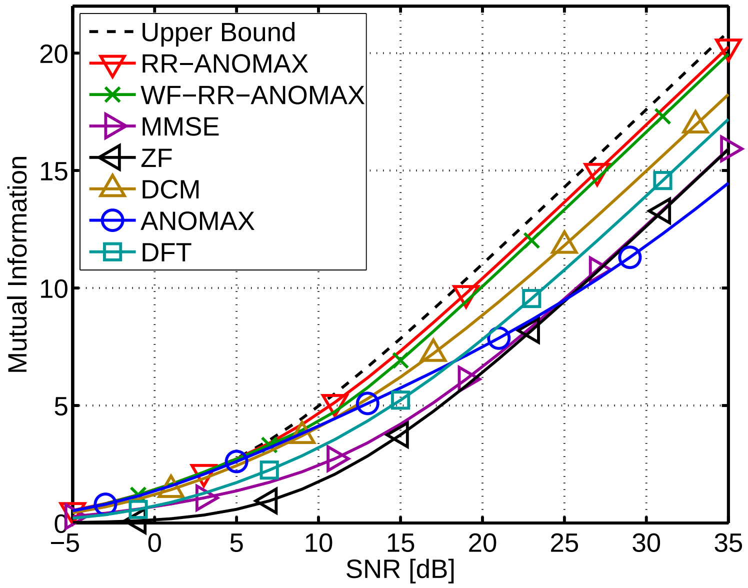


(a) $M_2 = M_2 = 4,\ M_{\mathrm{R}} = 2$

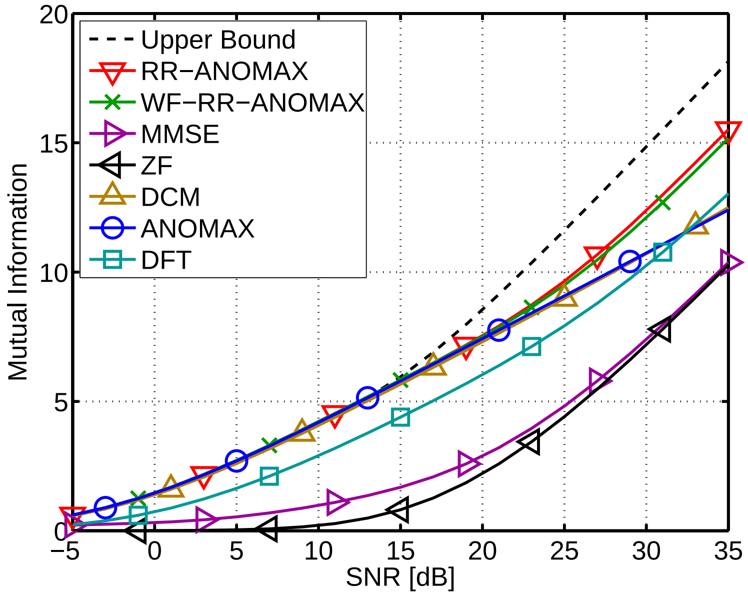


(b) $M_1 = M_2 = 2,\ M_{\mathrm{R}} = 5$

Figure 16.4.: Bit error rate using dominant eigenmode transmission (DET) and uncoded QPSK for correlated Rayleigh fading channels ($\rho_1 = \rho_2 = 0.9$, $\rho_{\mathrm{R}} = 0$) using various relaying strategies. Solid curves represent perfect CSI (pCSI), dashed lines imperfect CSI (iCSI) where the channels are estimated using the SLS-based iterative refinement for TENCE and 2 dedicated pilots. Left: $M_2 = M_2 = 4$, $M_{\mathrm{R}} = 2$. Right: $M_1 = M_2 = 2$, $M_{\mathrm{R}} = 5$.



(a) No spatial correlation



(b) Spatial correlation at the relay $\rho_R = 0.9$

Figure 16.5.: Maximum mutual information vs. the SNR for $M_1 = M_2 = 2$, $M_R = 4$ using Rayleigh fading channels. Left: no spatial correlation $\rho_1 = \rho_2 = \rho_R = 0$. Right: spatial correlation at the relay $\rho_1 = \rho_2 = 0$, $\rho_R = 0.9$.

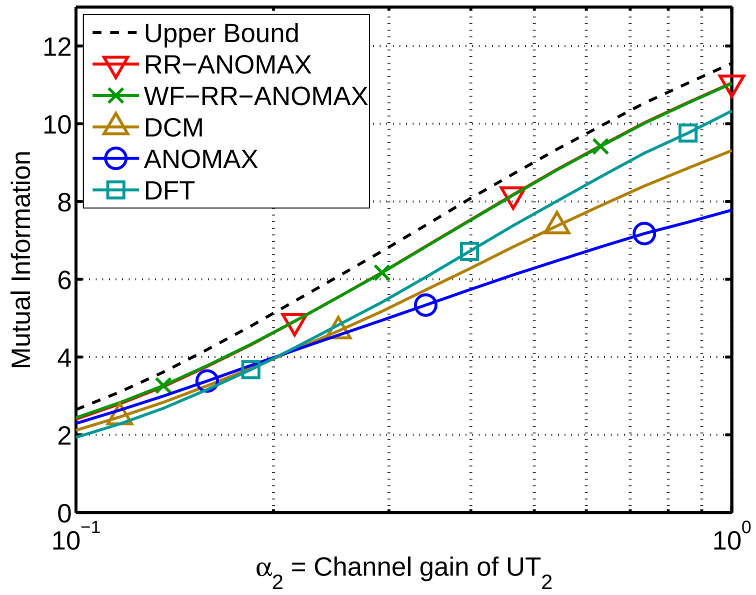


Figure 16.6.: Maximum mutual information vs. the path gain of UT_2 for $\alpha_1 = 1$, $M_1 = M_2 = M_{\mathrm{R}} = 3$, an SNR of 20 dB, and $\rho_1 = \rho_2 = \rho_{\mathrm{R}} = 0$.

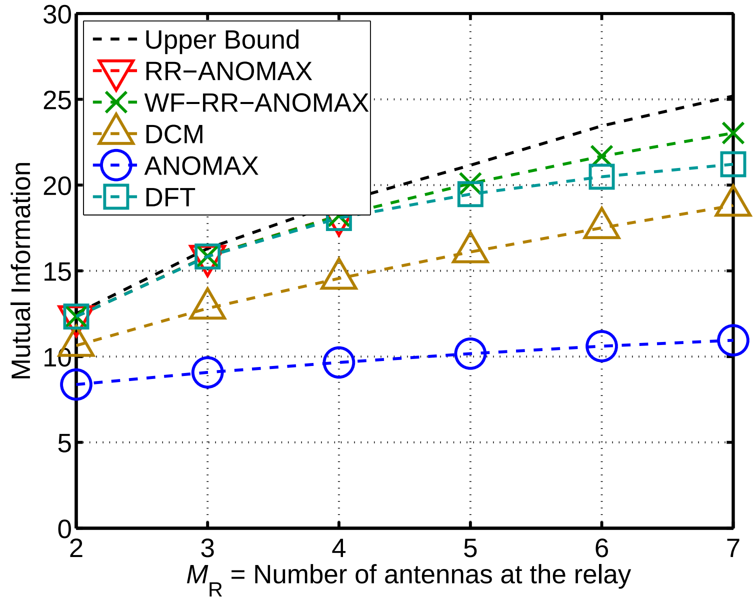


Figure 16.7.: Maximum mutual information vs. the number of antennas at the relay M_{R} for $M_1 = M_2 = 6$, an SNR of 20 dB, and $\rho_1 = \rho_2 = \rho_{\mathrm{R}} = 0$.

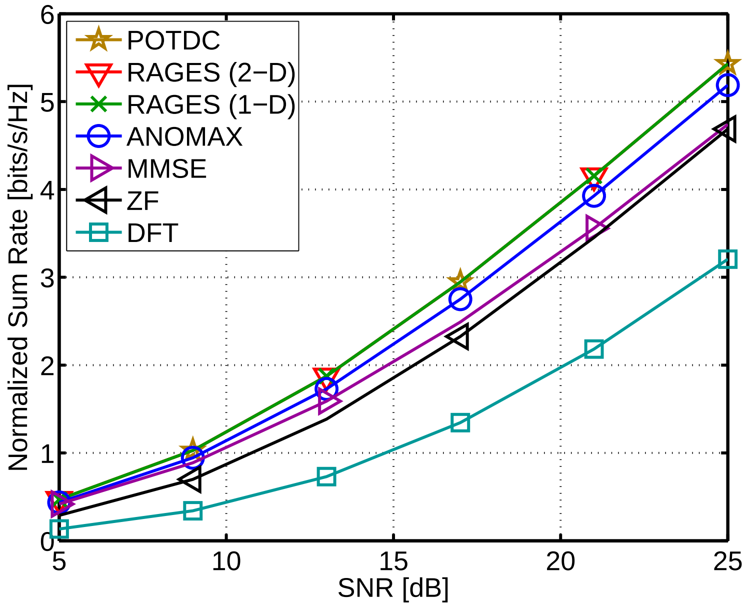


Figure 16.8.: Maximum mutual information vs. the path gain of UT_2 for $\alpha_1 = 1$, $M_1 = M_2 = M_R = 3$, an SNR of 20 dB, and $\rho_1 = \rho_2 = \rho_R = 0$.

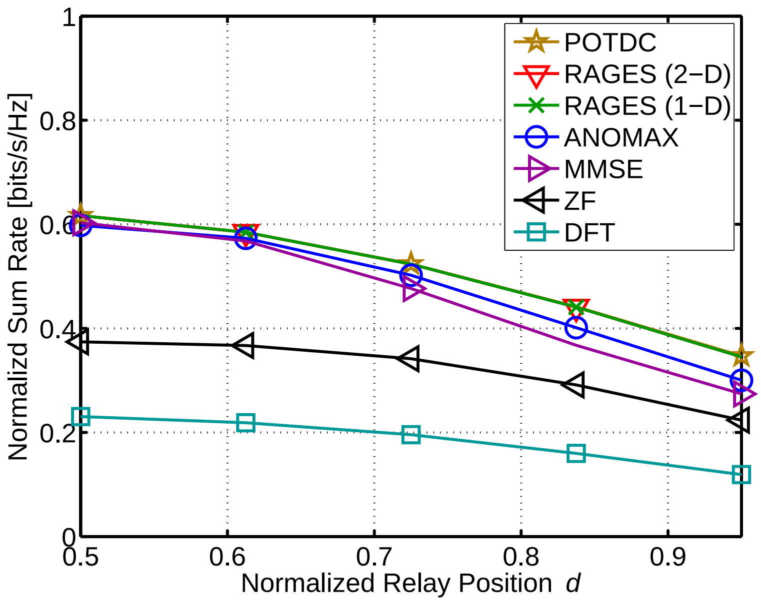


Figure 16.9.: Maximum mutual information vs. the number of antennas at the relay M_R for $M_1 = M_2 = 6$, an SNR of 20 dB, and $\rho_1 = \rho_2 = \rho_R = 0$.

17. Summary of two-way relaying

17.1. Summary of contributions

This part of the thesis discusses two-way relaying with a MIMO Amplify and Forward relay station. The two main aspects that are discussed represent the channel estimation problem as well as the choice of the relay amplification matrix. Our main novel contributions are:

- The algebraic Tensor-Based Channel Estimation (TENCE) algorithm [RH10c, RH09e], which provides both terminals with knowledge of the channel matrices between the terminals and the relay station. Based on this knowledge, all the effective channel matrices can be constructed, i.e., the self-interference channel (to subtract the own "echo" signal), the forward channel to the other UT (for transmit precoding), and the backward channel from the other UT (for spatial decoding). Note that TENCE applies to arbitrary antenna configurations and resolves all relevant scaling ambiguities.

- A Structured-Least Squares (SLS)-based iterative refinement for TENCE [RH10c, RH09d] which improves the estimation accuracy further by performing only 1-4 refinement steps. Such an iterative process is necessary since finding the channel matrices requires solving a quadratic Least Squares (LS) problem for which no closed-form LS-optimal solution exists.

- Design rules and recommendations for the training sequences to be used during the training phase for TENCE [RH10c].

- The Algebraic Norm Maximizing (ANOMAX) [RH09a] transmit strategy for choosing the relay amplification matrix. ANOMAX focuses on maximizing the Frobenius norms of the effective channel matrices which results in a strong concentration of the energy onto the dominant eigenmodes of the channels and thus a very good BER performance and a high diversity order, outperforming existing schemes [UK08, VH11]. Section 16.3.1 extends [RH09a] by an enhanced discussion where we prove that ANOMAX and DCM are identical for single-antenna terminals and that ANOMAX is rank-one if the weighting coefficient is set to zero or to one, see Appendices E.6 and E.7.

- The Rank-Restored ANOMAX (RR-ANOMAX) [RH10a] scheme which inherits the good performance of ANOMAX for low SNRs but fixes the problem that ANOMAX cannot provide the full spatial multiplexing gain for high SNRs (due to the low-rank nature of the resulting channel matrices) by adjusting the singular value profile of ANOMAX, dependent on the SNR. Such an adaptation can be performed in a heuristic manner, for instance, by a heuristic inspired by the Water Filling (WF) principle. As we show, the resulting WF-RR-ANOMAX scheme outperforms many other algebraic approaches [UK08, VH11] and yields a sum-rate close to the upper bound.

- The Rate Maximization via Generalized Eigenvectors for Single-Antenna terminals (RAGES) [RH10b]. RAGES represents a semi-algebraic scheme which provides the sum-rate optimal strategy in the special case of single-antenna terminals. As we demonstrate, the sum-rate in such a case can be expressed as the product of two Rayleigh quotients. The optimal relaying strategy is then found via a Generalized Eigenvector of two matrices that depend on two unknown real-valued scalar quantities. Since both scalar parameters have a physical interpretation, the search for them can be efficiently implemented. As we demonstrate, a simplified 1-D search where one of the two parameters is ignored provides a sum-rate that is very close to the optimum found via a 2-D search. Simulations verify that 2-D RAGES is in fact optimal by comparing it to the sum-rate optimal Polynomial Time DC (POTDC) algorithm proposed by us in [KVRH12, KRVH12] while being significantly simpler than existing schemes based on numerical optimization [ZLCC09]. Section 16.4 extends [RH10b] and [KRVH12] by introducing a low-complexity version for the special case of white noise at the relay, see Appendix E.9.

More related contributions that are not explicitly discussed in this thesis but worth mentioning are:

- The extension to multiple user pairs, for instance in the context of voluntary inter-operator physical resource sharing (SAPHYRE[1]) [JBF$^+$10], giving rise to the projection based separation of multiple operators (ProBaSeMO) [ZRH12a, RZHJ10]. This idea is also applicable to the communication between a base station and several terminals via one relay station [ZRH11]. For the special case of single-antenna terminals and a network of single-antenna relays, several optimization-based approaches are discussed in [ZRH$^+$12b]. Likewise, the sum-rate maximization with a MIMO AF relay and several pairs of single-antenna terminals is discussed in [ZBR$^+$12].

[1]SAPHYRE is an European research project, partly funded by the European Union under its FP7 ICT Objective 1.1 - The Network of the Future (FP7-ICT-248001), `http://saphyre.eu`.

- The adaptation of ANOMAX and RR-ANOMAX to the (one-way) interference channel with a MIMO AF relay is shown in [LRH11a, LRH11b, LZR+11].
- Near-Far Robustness and optimal power allocation for two-way relaying with a MIMO AF relay is discussed in [RH09c, RJH10].

17.2. Future work

The results that have been achieved so far open up some exciting directions for future research.

A major open direction is the analytical performance assessment of the algebraic schemes that have been proposed so far. In terms of the channel estimation schemes, this refers to deriving the MSE for TENCE and its SLS-based iterative refinement (possibly following some of the ideas that have been used to derive the performance of SLS-based ESPRIT in Section 12.4.6) and to compare it to the corresponding Cramér-Rao Bound in order to judge its efficiency.

For the relay amplification matrix designs an analytical characterization of the diversity order for particular fading distributions (e.g., Rayleigh fading) or the achievable sum-rate would be desirable. The same is true for the multi-pair extensions provided by the ProBaSeMO framework, where additionally the question of inter-pair fairness becomes relevant.

Note that a stronger analytical reasoning could lead to new ideas for a more rigorous Rank-Restored version of ANOMAX, which is so far only based on a heuristic adaptation of the singular values. We have seen that in some cases, there is still a gap to the upper bound in terms of the sum-rate, offering the potential to improve RR-ANOMAX further.

Another aspect that is still open is a proper extension of the algebraic schemes to the case of a network of MIMO relays. This becomes difficult mainly due to two facts: Firstly, each relay may have its own power constraint, which is difficult to incorporate. Secondly, each relay should only use the knowledge of its own channels to the UTs and not require (instantaneous) knowledge of the channels between the other relay stations and the UTs, since this would require significant signaling overhead. Of course, the potential improvement of having at least coarse knowledge (say, second-order statistics) of the other channels becomes interesting as well in order to judge whether a low-rate communication between relay stations could possibly pay off. Note that DCM provides a benchmark for this case since it is easily applied to relay networks and only relies on local channel knowledge [VH11].

It would also be desirable to compare our system assumptions to other alternatives in order to judge its practical significance in a broader context. Firstly, this includes a fair comparison of the AF relay we assumed to different relay operation modes (e.g., DF, EF, or CF) in terms of the achievable system performance (bit error rate, system sum rate, latency) and the required complexity (digital signal processing, hardware). Moreover, we have assumed that the UTs compute their precoding and decoding matrices from the effective channel matrices. This should be compared to the alternative approach where the UTs use their own channel matrix to the relay station for this task [XH10, LCSK10, JS10b, WT12].

Another aspect that has so far not been addressed is the application of the concepts to real-world conditions. While channel estimation errors have already been introduced, we are facing many more limitations when implementing the schemes in practice. For instance, perfect synchronization cannot be achieved and reciprocity (which is assumed for TENCE) will never exactly hold. Moreover, frequency-selective fading introduces additional challenges for which we need to introduce appropriate measures, e.g., chunk-wise adaptive precoding via OFDM (instead of performing the adaptation per subcarrier [OORW08, RFH08]). Furthermore, in presence of time-varying channels, the potential of deriving an adaptive channel tracking scheme based on TENCE could be investigated.

For the applicability to practical networks, it is also desirable to take Quality of Service (QoS) constraints into account, e.g., target bit error rates, maximum delays, rate requirements and so on. In light of the increasing share of battery-powered mobile devices and the call for green communications to reduce the world-wide carbon dioxide emissions, finding particularly energy efficient strategies becomes increasingly relevant. Our initial investigations on power allocation to achieve predefined SNR targets from [RH09c] can still be extended. We have only considered the minimization of the relay's transmit power for fixed powers at the UTs in [RH09c]. The minimization of the UTs transmit powers or the total network power would be a next step.

Part V.

Conclusions and Outlook

18. Conclusions

In this thesis, we investigate advanced algebraic concepts for finding efficient solutions to various applications of multi-channel digital signal processing. The main goal is to contribute methodically, i.e., discuss how such solutions can be found in general and demonstrate this approach with respect to different application areas. As we show, algebraic approaches can yield several benefits, e.g., a flexible control over the complexity-performance trade-off, a profound theoretical assessment of the achievable performance, or simply close-to-optimal low-complexity solutions.

In the first part of the thesis we summarize the tools for manipulating algebraic expressions that have been used throughout the thesis. We begin by a systematic treatment of various types of linear and quadratic forms and emphasize on the fact that they can be reduced into simple "canonical" versions. These often allow for direct solutions, e.g., Least Squares (LS)-optimal estimates of parameters in linear forms or (generalized) eigenvectors maximizing (ratios of) quadratic forms. We also discuss the LS-optimal factorization of Kronecker and Khatri-Rao products both for the case where one factor is known and the case where both factors are unknown. Additionally, we introduce the elementary concepts of multilinear algebra and discuss two of the most frequently used tensor decompositions, the Higher Order Singular Value Decomposition (HOSVD) and the Canonical Polyadic (CP) decomposition.

The second part of the thesis is devoted to the Semi-algebraic framework for approximate CP decomposition via Simultaneous matrix diagonalization (SECSI), first proposed by us in [RH08b, RH08a]. SECSI extends the known link between the CP decomposition and Simultaneous Matrix Diagonalizations (SMDs) [dL06] in a number of aspects. Firstly, we show that not only one but many SMDs can be constructed due to the symmetry of the problem and establish the full system of SMDs for the 3-D and the general R-D case. Secondly, we demonstrate that from each SMD, estimates for all loading matrices can be found. Thereby, we obtain several candidate solutions from which we select one final estimate in a subsequent step. Thirdly, we discuss that based on the choice which SMDs to solve and how to select the final estimate, many different algorithms can be defined within the framework, which allows us to flexibly control the performance-complexity trade-off. We propose several heuristic algorithms and compare their performance in simulations. We compare the achieved accuracy as well as the required run-time to state-of-the-art CP algorithms which shows the enhanced robustness and flexibility of the novel SECSI framework. It should be noted that the SECSI framework has already successfully been applied in several practical applications related to biomedical signal processing, i.e., to identify space-time-frequency components in event-related Electroencephalography (EEG) data [JRW+08, WRH+09], for a temporally resolved component analysis of dynamic EEG sources [WJG+10b, WJR+10], and for the decomposition of a space–time–wave-vector tensor obtained by a local spatial 3-D Fourier transform of the EEG measurement data [BCA+10, BCA+12]. Moreover, SECSI has been applied for model order selection in multidimensional data [DRWH10] and for parameter estimation from multidimensional signals with colored noise and imperfect antenna arrays [DSR+10].

In the third part of the thesis we study subspace-based multidimensional high-resolution parameter estimation. We introduce the concept of sampling multidimensional signals on separable R-dimensional grids. For data sampled in this manner we show that tensor algebra provides a convenient and natural description of the inherent multidimensional structure. Based on this description we introduce the tensor-based signal subspace estimate [RHD06, HRD08] which takes advantage of the structure and can be used to enhance arbitrary subspace-based multidimensional parameter estimation algorithms. We also prove a simple algebraic link between the matrix-based and the tensor-based subspace estimate (first shown by us for the special case $R = 2$ in [RBHW09]), showing that the latter can be computed by applying a structured projection to the former. We then introduce various enhancements to the family of Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT)-type algorithms which exploit the multidimensionality of the data (in form of the Tensor-ESPRIT-type algorithms [HRD08]), the non-circularity of the sources' amplitudes (in form of the NC ESPRIT-type algorithms [HR04]) or both jointly (in form of the NC Tensor-ESPRIT-type algorithms [RH09b]). For the tensor-case we additionally discuss the Tensor-Structure Structured Least Squares (TS-SLS) algorithm [RH07b], a tensor-based approach to solve the overdetermined shift invariance equations in ESPRIT. We demonstrate numerically that these modifications to ESPRIT yield an enhanced estimation accuracy by exploiting specific patterns in the data.

A particularly important aspect of the third part is the introduction of a generic framework for the analytical performance assessment of ESPRIT-type algorithms. We begin by extending an existing first-order perturbation expansion for the SVD-based subspace estimate [LLV93] to the HOSVD-based subspace estimate [RBHW09]. We then show how to obtain corresponding expansions for arbitrary ESPRIT-type algorithms based on LS [RBH10] and (1-D) Structured Least Squares (SLS) [RH11]. Since these are explicit, no assumptions about the statistics of the perturbations is required. We also provide closed-form expressions for the mean square estimation error (MSE) for the case of circularly symmetric white noise. Note that we do not need the noise to be Gaussian distributed. For the special case of a single source and uniform arrays, we simplify the MSEs into tractable closed-form expressions which allow insights into the efficiency of various ESPRIT-type algorithms. As we show, all ESPRIT-type algorithms based on LS have the same MSE for a single source and the asymptotic efficiency decreases with increasing array size. However, already one iteration of SLS reduces the MSE significantly, achieving an asymptotic efficiency very close to one.

The final part of the thesis discusses two-way relaying [RW05] with MIMO Amplify and Forward (AF) relays. We introduce the algebraic Tensor-Based Channel Estimation (TENCE) scheme [RH09e, RH10c] and its SLS-based iterative refinement [RH09d, RH10c] in order to obtain the required channel knowledge in such a system. As a side result, we find design rules and recommendations on the choice of the pilot sequences during the training phase. We then focus on the design of the relay amplification matrix in such a system. We first introduce the Algebraic Norm Maximizing (ANOMAX) transmit strategy [RH09a] that maximizes the Frobenius norm of the effective channel matrices, which yields a strong concentration of the energy onto the dominant eigenmode of these channels. This results in a very good Bit Error Rate (BER) performance, providing a high diversity order and hence a reliable communication link. However, for the case where our goal is to spatially multiplex several data streams, such a low-rank structure is undesirable since it fails to provide the full spatial multiplexing gain of $\min\{M_1, M_2, M_{\mathrm{R}}\}$. To address this issue we propose a simple algebraic modification of ANOMAX given by the Rank Restored ANOMAX (RR-ANOMAX) scheme [RH10a]. RR-ANOMAX adjusts the singular value profile of ANOMAX as a function of the Signal to Noise Ratio (SNR), restoring the required rank for high SNRs. As we demonstrate, already a very simple heuristic based on the Water Filling (WF) principle [Tel99] achieves a very good sum-rate close to the upper bound. Finally, for the special case of single-antenna terminals we propose the semi-algebraic sum-rate optimal Rate maximization via Generalized Eigenvectors for Single antenna terminals (RAGES) scheme [KRVH12, RH10b]. RAGES finds the optimal relay amplification matrix by searching generalized eigenvectors over a field described by two scalar real-valued parameters. Since both parameters have a physical meaning, the search can be implemented very efficiently and a simplified 1-D search performs very close to the optimal 2-D scheme. We verify the optimality by comparing RAGES to the sum-rate optimal convex optimization based Polynomial-Time DC (POTDC) algorithm [KRVH12].

Overall, the thesis demonstrates that many practical problems can efficiently be addressed in an algebraic or a semi-algebraic manner. We benefit from such solutions in multiple ways, e.g., a lower complexity (as for the algebraic approaches presented in the context of two-way relaying), enhanced flexibility (as for the SECSI framework), or the possibility to develop profound analytical results on their performance (as for the ESPRIT-type algorithms).

19. Future work

The thesis has addressed a broad spectrum of topics and, thereby, opened up many exciting directions for future research.

Beginning with the SECSI framework, its most serious shortcoming is the lack of analytical results for the achievable accuracy. A more profound theoretical understanding of the ultimate limit of Simultaneous Matrix Diagonalizations in general or, alternatively, the performance of specific algorithms [FG06, LA11] would allow to base the design of specific algorithms within the SECSI framework on more solid grounds. Ultimately, more sophisticated means of choosing which SMDs to solve and how to select the final estimate could lead to more robust and more efficient solutions. A less far-fetched goal is to alleviate the strict identifiability requirement that two of the modes of the tensor must be non-degenerate. Especially in the R-D case for $R > 3$ this can be achieved easily by means of "generalized" unfoldings [LA11]. We have developed an extension of the SECSI framework to take these generalized unfoldings into account and initial results on its performance are reported in [RSH12]. As expected, this extended framework outperforms SECSI in particular for $R > 3$. Since the generalized unfoldings offer a vast amount of options how to arrange the data, the number of potential model estimates becomes very large. Therefore, designing and evaluating appropriate heuristics for the extended framework is a challenging task with great potential for additional improvements. A third direction is to investigate the potential to implement SECSI in a truly parallelized fashion, which would allow for an efficient practical implementation on suitable DSP hardware. Also, an on-line implementation where the current solution is updated for every new batch of observed data may be required by certain applications.

Considering the subspace-based multidimensional parameter estimation algorithms discussed in the second part of the thesis, there are many open questions as well. Firstly, we have shown algebraic extensions using the family of ESPRIT-type algorithms as an example (cf. Tables 13.1 and 13.2). However, the approach is so generic that it can be applied to many other subspace-based algorithms as well. For instance, the tensor-based subspace estimate can be used to enhance the R-D RARE or the R-D MODE algorithms, too. A completely tensor-based formulation of the corresponding algorithms may lead to additional tensor gains that are specific to the algorithms (as in the TS-SLS algorithm for Tensor-ESPRIT). Likewise, the exploitation of non-circularity is possible for many other schemes (and has partly already been demonstrated, e.g., for the MUSIC algorithm). Moreover, finding a way to exploit also weak-sense non-circularity or the coexistence of circular and non-circular signals for ESPRIT is another open research item. For the analytical performance assessment, analytical results for some of the algorithms are still missing, e.g., the NC ESPRIT-type and the NC Tensor-ESPRIT-type algorithms (cf. Table 13.1). Also, the extension of the performance assessment of Structured Least Squares (SLS) to the Tensor-Structure Structured Least Squares (TS-SLS) algorithm is open. Another challenging task is to extend the simplified closed-form MSE expressions for the single-source case to the more relevant case of two sources. A description of the MSE in terms of the physical parameter such as the correlation and the spatial separation of the two sources may be very insightful. In this case, we observe a gain from using tensors in simulations and hence we would like to understand under which conditions this gain is particularly pronounced. It would also be desirable to extend the first-order perturbation framework to take into account the second-order terms since this provides insights into the performance in the threshold region for low SNRs. Finally, applying all these ideas to different algorithms, such as, R-D MUSIC, R-D MODE, or R-D RARE and investigating the tensor gain for these cases is a promising path as well.

Concerning the final part on two-way relaying with MIMO AF relays, many unanswered questions remain as well. A large open research area is the analytical performance assessment of the algebraic schemes that have been proposed in this area. This includes the algebraic channel estimation schemes TENCE and its SLS-based iterative refinement, where analytical results on the achievable channel estimation accuracy and a comparison to the corresponding Cramér-Rao Bound would be desirable. Also, the diversity order or the achievable sum-rate of the algebraic relay amplification matrix designs could be of large practical significance. A stronger analytical reasoning could also lead to new ideas for finding better versions of the Rank Restored version of ANOMAX, which is so far computed based on heuristics. An extension from the scenario with a single relay station to the case of multiple cooperating relay stations is another interesting open area. In this case, local power constraints, the proper exchange of channel state information, and network synchronization are major hurdles. Moreover, taking into account real-world conditions such as Quality of Service (QoS) constraints, frequency-selective fading, imperfect synchronization, or reciprocity imbalance is of significant practical interest. Such considerations help to judge the robustness of the proposed algorithms and may lead to new ideas how to improve them further with respect to real-world conditions. Finally, integrating the two-way relaying protocol into a larger wireless communication system and performing system-level simulations to assess its performance is an important step towards the adaptation of our developed ideas into future mobile communication standards.

Part VI.

Appendices

Appendix A.

Glossary of Acronyms, Symbols and Notation

A.1. Acronyms

AF	Amplify and Forward
ANC	Analog Network Coding
ANOMAX	Algebraic Norm Maximizing
ASK	Amplitude Shift Keying
BC	broadcast
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CoMP	Coordinated Multipoint
CP	Canonical Polyadic
DCM	Dual Channel Matching
DET	Dominant Eigenmode Transmission
DF	Decode and Forward
DFT	Discrete Fourier Transform
EEG	Electroencephalography
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
EVD	EigenValue Decomposition
FDMA	Frequency Division Multiple Access
HOSVD	Higher Order Singular Value Decomposition
LPA	Largest Principle Angle
LS	Least Squares
MAC	multiple access
MCS	Modulation and Coding Scheme
MDF	Multidimensional Folding
MIMO	Multiple Input Multiple Output
MODE	Method of Direction of Arrival Estimation
MUSIC	Multiple Signal Classification
MSE	Mean Squared Error
MSRE	Mean Square Reconstruction Error
MSK	Minimum Shift Keying
OFDM	Orthogonal Frequency Division Multiplexing
OQPSK	Offset Quadrature Phase Shift Keying
PCA	Principle Component Analysis
P2P	Point to point
PRIME	Polynomial Root Intersection for Multi-dimensional Estimation
RARE	Rank-Reduction Estimator

RF	Radio Frequency
SDMA	Space-Division Multiple Access
SLS	Structured Least Squares
SMUX	Spatial Multiplexing
SNR	Signal to Noise Ratio
SVD	Singular Value Decomposition
TDD	Time Division Duplexing
TDMA	Time Division Multiple Access
TLS	Total Least Squares
TMSFE	Total Mean Squared Factor Error
TS-SLS	Tensor-Structure Structured Least Squares
ULA	Uniform Linear Array
URA	Uniform Rectangular Array
WF	Water Filling
ZMCSCG	Zero Mean Circularly Symmetric Complex Gaussian

A.2. Symbols and Notation

$\mathbb{R}$	Set of real numbers
$\mathbb{C}$	Set of complex numbers
$\mathbb{Z}$	Set of integer numbers
e, π, $\jmath$	Euler's number, π, and imaginary unit: $\mathrm{e}^{\jmath\pi} + 1 = 0$
$\doteq$	Definition
a, b, c	scalars
$\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$	column vectors
$\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$	matrices
$\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$, $\boldsymbol{\mathcal{C}}$	tensors
$\mathrm{Re}\{x\}$	Real part of complex variable x
$\mathrm{Im}\{x\}$	Imaginary part of complex variable x
$\arg\{x\}$	Argument (phase) of complex variable x
x^*	Complex conjugate of x
$\mathbf{0}_{M\times N}$	Matrix of zeros of size $M \times N$
$\mathbf{1}_{M\times N}$	Matrix of ones of size $M \times N$
$\boldsymbol{I}_M$	Identity matrix of size $M \times M$
$\boldsymbol{\mathcal{I}}_{R,d}$	R-way identity tensor of size $d \times d \ldots \times d$
$\boldsymbol{\Pi}_M$	Exchange of size $M \times M$ with ones on its anti-diagonal and zeros elsewhere
$\boldsymbol{Q} \succcurlyeq \mathbf{0}$	$\boldsymbol{Q}$ is a positive-semidefinite matrix
$[\boldsymbol{A}]_{(i,j)}$	The (i,j)-element of the matrix $\boldsymbol{A}$
$[a_i]_{i=1,2,\ldots,I}$	An $I \times 1$ column vector $\boldsymbol{a}$ with i-th element a_i

$(\cdot)^{\mathrm{T}}$	matrix transpose
$(\cdot)^{\mathrm{H}}$	Hermitian transpose
$\lVert\cdot\rVert_2$	Euclidean (two-) norm
$\lVert\cdot\rVert_{\mathrm{F}}$	Frobenius norm
$\lVert\cdot\rVert_{\mathrm{H}}$	Higher-Order (Frobenius) norm
$\boldsymbol{A} \otimes \boldsymbol{B}$	Kronecker product between $\boldsymbol{A} \in \mathbb{C}^{M\times N}$ and $\boldsymbol{B} \in \mathbb{C}^{P\times Q}$ defined as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{1,1}\cdot\boldsymbol{B} & a_{1,2}\cdot\boldsymbol{B} & \cdots & a_{1,N}\cdot\boldsymbol{B} \\ a_{2,1}\cdot\boldsymbol{B} & a_{2,2}\cdot\boldsymbol{B} & \cdots & a_{2,N}\cdot\boldsymbol{B} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M,1}\cdot\boldsymbol{B} & a_{M,2}\cdot\boldsymbol{B} & \cdots & a_{M,N}\cdot\boldsymbol{B} \end{bmatrix}.$$

$\boldsymbol{A} \diamond \boldsymbol{B}$	Khatri-Rao (column-wise Kronecker) product between $\boldsymbol{A} \in \mathbb{C}^{M\times N}$ and $\boldsymbol{B} \in \mathbb{C}^{P\times N}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Schur (element-wise) product between $\boldsymbol{A} \in \mathbb{C}^{M\times N}$ and $\boldsymbol{B} \in \mathbb{C}^{M\times N}$
$\boldsymbol{A} \oslash \boldsymbol{B}$	Inverse Schur product (element-wise division) between $\boldsymbol{A} \in \mathbb{C}^{M\times N}$ and $\boldsymbol{B} \in \mathbb{C}^{M\times N}$
$\mathrm{vec}\{\cdot\}$	vec-operator: stack elements of a matrix/tensor into a column vector, begin with first (row) index, then proceed to second (column), third, etc.
$\mathrm{unvec}_{I\times J}\{\cdot\}$	inverse vec-operator: reshape elements of a vector back into a matrix/tensor of indicated size
$\mathrm{diag}\{\cdot\}$	transform a vector into a square diagonal matrix or extract main diagonal of a square matrix and place elements into a vector
$\mathrm{off}\{\cdot\}$	extracts all off-diagonal elements of a given square matrix $\boldsymbol{X} \in \mathbb{C}^{d\times d}$, i.e., $\mathrm{off}\{\boldsymbol{X}\} = \boldsymbol{X} - (\boldsymbol{X} \odot \boldsymbol{I}_d)$
$\mathrm{trace}\{\cdot\}$	trace of a matrix (sum of diagonal elements = sum of eigenvalues)
$\det\{\cdot\}$	determinant of a matrix (product of eigenvalues)
$\mathrm{rank}\{\cdot\}$	rank of a matrix
$\mathrm{EV}_k\{\cdot\}$	k-th eigenvalue of a matrix
$\mathrm{orth}\{\boldsymbol{A}\}$	returns a matrix $\boldsymbol{U} \in \mathbb{C}^{M\times r}$ with mutually orthonormal columns that span the same space as the columns of $\boldsymbol{A} \in \mathbb{C}^{M\times r}$ (assuming $r \le M$)
$\boldsymbol{A}^+$	Moore-Penrose pseudo inverse [Moo20, Pen55] of a matrix $\boldsymbol{A} \in \mathbb{C}^{M\times N}$, which we can compute via

- $\boldsymbol{A}^+ = \boldsymbol{V}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}}^{-1} \cdot \boldsymbol{U}_{\mathrm{s}}^{\mathrm{H}}$, where $\boldsymbol{A} = \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}} \cdot \boldsymbol{V}_{\mathrm{s}}^{\mathrm{H}}$ represents the economy-size SVD of $\boldsymbol{A}$ (cf. Section 3.2).

	• $\boldsymbol{A}^+ = (\boldsymbol{A}^{\mathrm{H}} \cdot \boldsymbol{A})^{-1} \cdot \boldsymbol{A}^{\mathrm{H}}$ if $\operatorname{rank}\{\boldsymbol{A}\} = N$ (full column rank)
	• $\boldsymbol{A}^+ = \boldsymbol{A}^{\mathrm{H}} \cdot (\boldsymbol{A} \cdot \boldsymbol{A}^{\mathrm{H}})^{-1}$ if $\operatorname{rank}\{\boldsymbol{A}\} = M$ (full row rank).
$[\boldsymbol{\mathcal{X}}]_{(n)}$	n-mode unfolding of tensor $\boldsymbol{\mathcal{X}}$ in reverse cyclical column ordering
$\boldsymbol{\mathcal{X}} \times_n \boldsymbol{U}$	n-mode product between tensor $\boldsymbol{\mathcal{X}}$ and matrix $\boldsymbol{U}$
$\boldsymbol{\mathcal{X}} \mathop{\times}\limits_{r=1}^{R}{}_r \boldsymbol{U}_r$	repeated n-mode products, short-hand notation for $\boldsymbol{\mathcal{X}} \times_1 \boldsymbol{U}_1 \ldots \times_R \boldsymbol{U}_R$
$[\boldsymbol{\mathcal{A}} \sqcup_n \boldsymbol{\mathcal{B}}]$	n-mode concatenation of tensors $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$
$\mathbb{E}\{X\}$	Expectation operator, i.e., mean of the random variable X
$\operatorname{Med}\{X\}$	Median of the random variable X, i.e., the point where $\Pr[X \leq \operatorname{Med}\{X\}] = 0.5$. Note that $\operatorname{Med}\{X\} = \mathbb{E}\{X\}$ only if X has a symmetric distribution.
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ, variance σ^2
$\mathcal{CN}(\mu, \sigma^2)$	circularly symmetric complex Gaussian distribution
$\delta[x]$	Kronecker δ-symbol, $\delta[x] = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x \neq 0 \end{cases}$
$\boldsymbol{Q}_p$	Left-$\boldsymbol{\Pi}$-real matrix satisfying $\boldsymbol{\Pi}_p \cdot \boldsymbol{Q}_p^* = \boldsymbol{Q}_p$.
$\boldsymbol{Q}_p^{(\mathrm{s})}$	Unitary sparse left-$\boldsymbol{\Pi}$-real given by [HN95]

$$\boldsymbol{Q}_{2n}^{(\mathrm{s})} = \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_n & \jmath \boldsymbol{I}_n \\ \boldsymbol{\Pi}_n & -\jmath \boldsymbol{\Pi}_n \end{bmatrix} \quad \text{and} \quad \boldsymbol{Q}_{2n+1}^{(\mathrm{s})} = \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_n & \boldsymbol{0}_{n \times 1} & \jmath \boldsymbol{I}_n \\ \boldsymbol{0}_{n \times 1}^{\mathrm{T}} & \sqrt{2} & \boldsymbol{0}_{n \times 1}^{\mathrm{T}} \\ \boldsymbol{\Pi}_n & \boldsymbol{0}_{n \times 1} & -\jmath \boldsymbol{\Pi}_n \end{bmatrix},$$

for even and for odd order, respectively.

$\boldsymbol{D}_N$	An $N \times N$ DFT matrix given by

$$\boldsymbol{D}_N = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & w_N & \ldots & w_N^{N-1} \\ 1 & w_N^2 & \ldots & w_N^{2(N-1)} \\ \vdots & \ldots & \vdots & \vdots \\ 1 & w_N^{N-1} & \ldots & w_N^{(N-1)^2} \end{bmatrix},$$

where $w_N = \mathrm{e}^{-\jmath \frac{2\pi}{N}}$.

Appendix B.

Proofs and derivations for Part I

B.1. Proof of Proposition 3.1.1

Since $\boldsymbol{A} \otimes \boldsymbol{X}$ and $\boldsymbol{X} \otimes \boldsymbol{A}$ are both matrices of linear forms, it is clear that their vectorized versions are linear forms as well. Therefore, they must be expressible as a coefficient matrix multiplying $\operatorname{vec}\{\boldsymbol{X}\}$. Finding the correct alignment of the entries of $\boldsymbol{A}$ into this bigger coefficient matrix is merely a matter of writing down the elements of $\operatorname{vec}\{\boldsymbol{A} \otimes \boldsymbol{X}\}$ and $\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\}$ in the correct order and then reading the corresponding coefficients out.

For instance, consider $\boldsymbol{X} \otimes \boldsymbol{A}$ and for simplicity assume $P = Q = 2$ such that $\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$. Moreover, let $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_1 & \dots \boldsymbol{a}_N \end{bmatrix}$. Then $\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\}$ can be written as

$$\operatorname{vec}\left\{\begin{bmatrix} x_{11} \cdot \begin{bmatrix} \boldsymbol{a}_1 & \dots \boldsymbol{a}_N \end{bmatrix} & x_{12} \cdot \begin{bmatrix} \boldsymbol{a}_1 & \dots \boldsymbol{a}_N \end{bmatrix} \\ x_{21} \cdot \begin{bmatrix} \boldsymbol{a}_1 & \dots \boldsymbol{a}_N \end{bmatrix} & x_{22} \cdot \begin{bmatrix} \boldsymbol{a}_1 & \dots \boldsymbol{a}_N \end{bmatrix} \end{bmatrix}\right\}$$
$$= \begin{bmatrix} x_{11} \cdot \boldsymbol{a}_1^{\mathrm{T}} & x_{21} \cdot \boldsymbol{a}_1^{\mathrm{T}} & \dots & x_{11} \cdot \boldsymbol{a}_N^{\mathrm{T}} & x_{21} \cdot \boldsymbol{a}_N^{\mathrm{T}} & x_{12} \cdot \boldsymbol{a}_1^{\mathrm{T}} & x_{22} \cdot \boldsymbol{a}_1^{\mathrm{T}} & \dots & x_{12} \cdot \boldsymbol{a}_N^{\mathrm{T}} & x_{22} \cdot \boldsymbol{a}_N^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}.$$

Consequently, the first $2M$ elements of $\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\}$ are equal to $\boldsymbol{a}_1$ scaled by x_{11}, then by x_{21}, the next $2M$ elements are $\boldsymbol{a}_2$ scaled by x_{11}, then by x_{21}, and so on. Therefore, the first half of $\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\}$ can be written as

$$\begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{a}_1 \\ \boldsymbol{a}_2 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{a}_2 \\ \vdots & \\ \boldsymbol{a}_N & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{a}_N \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_2 \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_N \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \end{bmatrix}. \tag{B.1}$$

Likewise, the second half can be expressed as

$$\begin{bmatrix} \boldsymbol{a}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{a}_1 \\ \boldsymbol{a}_2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{a}_2 \\ \vdots & \\ \boldsymbol{a}_N & \mathbf{0} \\ \mathbf{0} & \boldsymbol{a}_N \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_2 \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_N \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \end{bmatrix}. \tag{B.2}$$

Combining (B.1) and (B.2) we finally have

$$\operatorname{vec}\{\boldsymbol{X} \otimes \boldsymbol{A}\} = \begin{bmatrix} \begin{bmatrix} \boldsymbol{I}_2 \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_N \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} \boldsymbol{I}_2 \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_N \end{bmatrix} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{22} \end{bmatrix} = \left(\begin{bmatrix} \boldsymbol{I}_2 \otimes \boldsymbol{a}_1 \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{I}_2 \otimes \boldsymbol{a}_N \end{bmatrix} \otimes \boldsymbol{I}_2 \right) \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{22} \end{bmatrix}. \tag{B.3}$$

This shows the theorem for $P = Q = 2$, the extension to arbitrary P and Q is straightforward. The derivation for $\operatorname{vec}\{\boldsymbol{A} \otimes \boldsymbol{X}\}$ proceeds in a similar fashion. □

B.2. Proof of Proposition 3.1.2

We start by the vectorization of $\boldsymbol{A} \diamond \boldsymbol{X}$. By its definition it contains the vectors $\boldsymbol{a}_n \otimes \boldsymbol{x}_n$ for $n = 1, 2, \ldots, N$. For each of those, we can apply (3.30), i.e., $\boldsymbol{a}_n \otimes \boldsymbol{x}_n = (\boldsymbol{a}_n \otimes \boldsymbol{I}_P) \cdot \boldsymbol{x}_n$. Collecting these vectors for all n we obtain

$$\begin{aligned} \operatorname{vec}\{\boldsymbol{A} \diamond \boldsymbol{X}\} &= \begin{bmatrix} \boldsymbol{a}_1 \otimes \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{a}_N \otimes \boldsymbol{x}_N \end{bmatrix} = \begin{bmatrix} (\boldsymbol{a}_1 \otimes \boldsymbol{I}_P) \cdot \boldsymbol{x}_1 \\ \vdots \\ (\boldsymbol{a}_N \otimes \boldsymbol{I}_P) \cdot \boldsymbol{x}_N \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{a}_1 \otimes \boldsymbol{I}_P & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{a}_2 \otimes \boldsymbol{I}_P & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{a}_N \otimes \boldsymbol{I}_P \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix} \end{aligned} \tag{B.4}$$

$$= \left(\begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{a}_2 & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{a}_N \end{bmatrix} \otimes \boldsymbol{I}_P \right) \cdot \operatorname{vec}\{\boldsymbol{X}\} = ([\boldsymbol{I}_N \diamond \boldsymbol{A}] \otimes \boldsymbol{I}_P) \cdot \operatorname{vec}\{\boldsymbol{X}\}, \tag{B.5}$$

which is the desired result. For $\operatorname{vec}\{\boldsymbol{X} \diamond \boldsymbol{A}\}$, by applying similar reasoning, we can show

$$\operatorname{vec}\{\boldsymbol{X} \diamond \boldsymbol{A}\} = \begin{bmatrix} \boldsymbol{I}_P \otimes \boldsymbol{a}_1 & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_P \otimes \boldsymbol{a}_2 & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{I}_P \otimes \boldsymbol{a}_N \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix}. \tag{B.6}$$

However, this coefficient matrix cannot directly be broken into a Kronecker product between $\boldsymbol{I}_P$ and $\boldsymbol{I}_N \diamond \boldsymbol{A}$ since the columns would appear in the wrong order. Therefore, we compose it in a different manner. What we need is a block-diagonal matrix, where the n-th block is equal to $\boldsymbol{I}_P \otimes \boldsymbol{a}_n$, i.e., another block diagonal matrix containing P repeated copies of the n-th column of $\boldsymbol{A}$. Therefore, we first modify the matrix $\boldsymbol{A}$ such that its N columns are repated P times each, which can be accomplished by a multiplication with $\boldsymbol{I}_N \otimes \boldsymbol{1}_{P\times 1}^{\mathrm{T}}$. The resulting $M \times P \cdot N$ matrix is transformed into a block-diagonal matrix via the Khatri-Rao product with an identity matrix of size $P \cdot N \times P \cdot N$. This shows the second part of the proposition.

For the third part, we observe that the Schur product contains the terms $b_{k,\ell} \cdot x_{k,\ell}$, i.e., each element of $\boldsymbol{X}$ gets scaled by one element in $\boldsymbol{B}$. Therefore, the vectorized version of $\boldsymbol{X} \odot \boldsymbol{B}$ contains the elements of $\boldsymbol{X}$ in the same order as $\operatorname{vec}\{\boldsymbol{X}\}$ only multiplied with the corresponding element in $\boldsymbol{B}$. This can for instance be written as a pre-multiplication with a diagonal matrix, i.e., $\operatorname{diag}\{\operatorname{vec}\{\boldsymbol{B}\}\} \cdot \operatorname{vec}\{\boldsymbol{X}\}$. Alternatively, we can use a Khatri-Rao product and write $\left(\boldsymbol{I}_{P \cdot N} \diamond \operatorname{vec}\{\boldsymbol{B}\}^{\mathrm{T}}\right) \cdot \operatorname{vec}\{\boldsymbol{X}\}$. □

B.3. The PC-Kronecker product

The Kronecker product between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is a matrix which contains all pair-wise products of the elements of $\boldsymbol{A}$ and $\boldsymbol{B}$ [Bel60]. The ordering of these elements could be chosen in different ways. By convention, we have

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{1,1} \cdot \boldsymbol{B} & a_{1,2} \cdot \boldsymbol{B} & \cdots & a_{1,N} \cdot \boldsymbol{B} \\ a_{2,1} \cdot \boldsymbol{B} & a_{2,2} \cdot \boldsymbol{B} & \cdots & a_{2,N} \cdot \boldsymbol{B} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M,1} \cdot \boldsymbol{B} & a_{M,2} \cdot \boldsymbol{B} & \cdots & a_{M,N} \cdot \boldsymbol{B} \end{bmatrix} \tag{B.7}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\boldsymbol{B} \in \mathbb{C}^{P \times Q}$, and $a_{k,\ell} = [\boldsymbol{A}]_{(k,\ell)}$ denotes the (k,ℓ)-element of $\boldsymbol{A}$.

In most cases, this arrangement of elements is convenient, as many identities are readily expressed via this type of Kronecker product. There are, however, exceptions where a different ordering would be beneficial to arrive at more compact expressions. To this end, we define a "PC-Kronecker product" (PC = permuted columns) between $\boldsymbol{A}$ and $\boldsymbol{B}$ in the following manner

$$\boldsymbol{A} \boxtimes \boldsymbol{B} = \begin{bmatrix} \boldsymbol{A} \otimes \boldsymbol{b}_1 & \boldsymbol{A} \otimes \boldsymbol{b}_2 & \cdots & \boldsymbol{A} \otimes \boldsymbol{b}_Q \end{bmatrix}. \tag{B.8}$$

The PC-Kronecker product is equal to the Kronecker product except for a different ordering of its columns. Note that, like Kronecker products, multiple PC-Kronecker products are associative, i.e., $(\boldsymbol{A} \boxtimes \boldsymbol{B}) \boxtimes \boldsymbol{C} = \boldsymbol{A} \boxtimes (\boldsymbol{B} \boxtimes \boldsymbol{C})$. However, mixed Kronecker/PC-Kronecker products are not associative, i.e., $(\boldsymbol{A} \boxtimes \boldsymbol{B}) \otimes \boldsymbol{C} \neq \boldsymbol{A} \boxtimes (\boldsymbol{B} \otimes \boldsymbol{C})$.

With the help of the PC-Kronecker product, we can for instance directly compute the commutation matrices from [MN95], introduced in (3.34), via the simple relation $\boldsymbol{K}_{M,N} = \boldsymbol{I}_M \boxtimes \boldsymbol{I}_N$. For instance,

$$\boldsymbol{K}_{3,4} = \boldsymbol{I}_3 \boxtimes \boldsymbol{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{B.9}$$

Likewise, the permutation matrices introduced in (4.11) are readily computed as

$$\boldsymbol{P}^{(1)}_{M_1,M_2,M_3} = (\boldsymbol{I}_{M_3} \boxtimes \boldsymbol{I}_{M_2}) \otimes \boldsymbol{I}_{M_1} \tag{B.10}$$

$$\boldsymbol{P}^{(2)}_{M_1,M_2,M_3} = \boldsymbol{I}_{M_3} \otimes (\boldsymbol{I}_{M_2} \boxtimes \boldsymbol{I}_{M_1}) \tag{B.11}$$

$$\boldsymbol{P}^{(3)}_{M_1,M_2,M_3} = \boldsymbol{I}_{M_3} \boxtimes \boldsymbol{I}_{M_2} \boxtimes \boldsymbol{I}_{M_1}, \tag{B.12}$$

or, in the general R-D case,

$$\boldsymbol{P}^{(r)}_{M_1,M_2,\ldots,M_R} = \begin{cases} (\boldsymbol{I}_{M_R} \boxtimes \ldots \boxtimes \boldsymbol{I}_{M_{r+1}}) \otimes (\boldsymbol{I}_{M_r} \boxtimes \ldots \boxtimes \boldsymbol{I}_{M_1}) & r = 1, 2, \ldots, R-1 \\ \boldsymbol{I}_{M_R} \boxtimes \ldots \boxtimes \boldsymbol{I}_{M_2} \boxtimes \boldsymbol{I}_{M_1} & r = R. \end{cases} \tag{B.13}$$

B.4. Proof for Proposition 3.1.3

Let $\boldsymbol{Y} = \boldsymbol{A} \cdot \boldsymbol{X}$ and $\boldsymbol{Z} = \boldsymbol{B} \cdot \boldsymbol{X}^{\mathrm{H}} \cdot \boldsymbol{C}^{\mathrm{H}}$. Then, the left-hand side of (3.39) is equal to $\operatorname{trace}\{\boldsymbol{Y} \cdot \boldsymbol{Z}\}$. Using identity (3.8) we can rewrite this as

$$\operatorname{trace}\{\boldsymbol{Y} \cdot \boldsymbol{Z}\} = \operatorname{vec}\left\{\boldsymbol{Z}^{\mathrm{T}}\right\}^{\mathrm{T}} \cdot \operatorname{vec}\{\boldsymbol{Y}\} \tag{B.14}$$

$$= \operatorname{vec}\left\{\boldsymbol{C}^* \cdot \boldsymbol{X}^* \boldsymbol{B}^{\mathrm{T}}\right\}^{\mathrm{T}} \cdot \operatorname{vec}\{\boldsymbol{A} \cdot \boldsymbol{X}\} \tag{B.15}$$

However, (B.15) represents the inner product of two linear forms for which we can apply (3.7). We obtain

$$\operatorname{vec}\left\{\boldsymbol{C}^* \cdot \boldsymbol{X}^* \boldsymbol{B}^{\mathrm{T}}\right\} \cdot \operatorname{vec}\{\boldsymbol{A} \cdot \boldsymbol{X}\} = \left[(\boldsymbol{B} \otimes \boldsymbol{C}^*) \cdot \operatorname{vec}\{\boldsymbol{X}^*\}\right]^{\mathrm{T}} \cdot (\boldsymbol{I}_N \otimes \boldsymbol{A}) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{B.16}$$

$$= \operatorname{vec}\{\boldsymbol{X}^*\}^{\mathrm{T}} \cdot \left(\boldsymbol{B}^{\mathrm{T}} \otimes \boldsymbol{C}^{\mathrm{H}}\right) \cdot (\boldsymbol{I}_N \otimes \boldsymbol{A}) \cdot \operatorname{vec}\{\boldsymbol{X}\} \tag{B.17}$$

$$= \operatorname{vec}\{\boldsymbol{X}\}^{\mathrm{H}} \cdot \left(\boldsymbol{B}^{\mathrm{T}} \otimes \boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{A}\right) \cdot \operatorname{vec}\{\boldsymbol{X}\}, \tag{B.18}$$

where we have additionally used (3.17). This is the desired result. □

B.5. Diagonalizable matrices

In this appendix we discuss under which conditions a square matrix $\boldsymbol{X} \in \mathbb{C}^{N \times N}$ is diagonalizable which implies that it possesses N linearly independent eigenvectors, as mentioned in Section 3.2.

To this end we need to define the multiplicities of an eigenvalue. Let $p(\lambda) = \det\{\boldsymbol{X} - \lambda \cdot \boldsymbol{I}_N\} \in \mathbb{C}$ be the characteristic polynomial of the matrix $\boldsymbol{X}$ so that the eigenvalues of $\boldsymbol{X}$ are the roots of $p(\lambda)$. Then the **algebraic multiplicity** of a particular eigenvalue λ_n is defined as the multiplicity of the root λ_n of the polynomial $p(\lambda)$. On the other hand, the **geometric** multiplicity λ_n is defined as the dimension of the nullspace of the matrix $\boldsymbol{X} - \lambda_n \cdot \boldsymbol{I}_N$. For instance, if all eigenvalues are distinct, all geometric and algebraic multiplicities are equal to one.

Based on this definition we have the following result [GvL80]: a matrix is diagonalizable if for all eigenvalues, their algebraic and their geometric multiplicity agrees.

To illustrate this effect, let us consider a counter-example given by the matrix $\boldsymbol{X}$

$$\boldsymbol{X} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}. \tag{B.19}$$

The characteristic polynomial of $\boldsymbol{X}$ is given by $p(\lambda) = \lambda^2$. Therefore, the eigenvalues are $\lambda_1 = \lambda_2 = 0$. In other words, the matrix has the eigenvalue 0 with algebraic multiplicity equal to two. However, the geometric multiplicity of the eigenvalue 0 is only one since $\operatorname{rank}\{\boldsymbol{X} - 0 \cdot \boldsymbol{I}_2\} = \operatorname{rank}\{\boldsymbol{X}\} = 1$. Therefore, this particular matrix $\boldsymbol{X}$ is not diagonalizable.

B.6. Further notes on LS Kronecker/Khatri-Rao factorization when one factor is known

The explicit expressions for the Least-Squares Kronecker factors provided in (3.54)-(3.57) can be simplified by using (3.15) and (3.22) to expand the pseudo-inverse of the Kronecker and the Khatri-Rao products, respectively.

Let us start with the Kronecker factorizations (3.55) and (3.57). Introducing the short hand notation

$$\tilde{\boldsymbol{C}}_Q = \begin{bmatrix} \boldsymbol{I}_Q \otimes \boldsymbol{c}_1 \\ \boldsymbol{I}_Q \otimes \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{I}_Q \otimes \boldsymbol{c}_N \end{bmatrix} \tag{B.20}$$

equation (3.55) reads as $\operatorname{vec}\{\boldsymbol{Y}_{\mathrm{LS}}\} = \left(\tilde{\boldsymbol{C}}_Q \otimes \boldsymbol{I}_P\right)^+ \cdot \operatorname{vec}\{\boldsymbol{D}\}$. We can apply (3.15) to rewrite $\left(\tilde{\boldsymbol{C}}_Q \otimes \boldsymbol{I}_P\right)^+$ as $\left(\tilde{\boldsymbol{C}}_Q^+ \otimes \boldsymbol{I}_P\right)$. Furthermore, $\tilde{\boldsymbol{C}}_Q$ is of size $Q \cdot N \times Q$ and has has rank Q as long as $\boldsymbol{C}$ is not equal to the zero matrix. Consequently, the pseudo-inverse of $\tilde{\boldsymbol{C}}_Q$ can be written as $\tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \cdot \left(\tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \cdot \tilde{\boldsymbol{C}}_Q\right)^{-1}$. Via the definition of $\tilde{\boldsymbol{C}}_Q$ from (B.20) it is easy to see that

$$\tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \cdot \tilde{\boldsymbol{C}}_Q = \sum_{n=1}^{N} \left(\boldsymbol{I}_Q \otimes \boldsymbol{c}_n\right)^{\mathrm{H}} \cdot \left(\boldsymbol{I}_Q \otimes \boldsymbol{c}_n\right) \tag{B.21}$$

Applying (3.17), this can be further simplified into

$$\tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \cdot \tilde{\boldsymbol{C}}_Q = \sum_{n=1}^{N} \left(\boldsymbol{I}_Q \otimes \left(\boldsymbol{c}_n^{\mathrm{H}} \cdot \boldsymbol{c}_n\right)\right) = \boldsymbol{I}_Q \otimes \left(\sum_{n=1}^{N} \boldsymbol{c}_n^{\mathrm{H}} \cdot \boldsymbol{c}_n\right) = \boldsymbol{I}_Q \cdot \|\boldsymbol{C}\|_{\mathrm{F}}^2 \tag{B.22}$$

Consequently, the pseudo-inverse of $\tilde{\boldsymbol{C}}_Q$ is easily expressed as $\tilde{\boldsymbol{C}}_Q^{+} = \tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \cdot \|\boldsymbol{C}\|_{\mathrm{F}}^{-2}$. This eliminates all matrix inversions from (3.55). Using similar arguments for (3.57) we arrive at the simplified rules

$$\operatorname{vec}\{\boldsymbol{Y}_{\mathrm{LS}}\} = \|\boldsymbol{C}\|_{\mathrm{F}}^{-2} \cdot \left(\tilde{\boldsymbol{C}}_Q^{\mathrm{H}} \otimes \boldsymbol{I}_P\right) \cdot \operatorname{vec}\{\boldsymbol{D}\} \tag{B.23}$$

for $\boldsymbol{D} \approx \boldsymbol{C} \otimes \boldsymbol{Y}$ and

$$\operatorname{vec}\{\boldsymbol{Y}_{\mathrm{LS}}\} = \|\boldsymbol{C}\|_{\mathrm{F}}^{-2} \cdot \left(\boldsymbol{I}_Q \otimes \tilde{\boldsymbol{C}}_P\right) \cdot \operatorname{vec}\{\boldsymbol{D}\} \tag{B.24}$$

for $\boldsymbol{D} \approx \boldsymbol{Y} \otimes \boldsymbol{C}$, respectively.

In [LP93], the same problem is addressed and a direct solution for $\boldsymbol{Y}$ is given in a different form. For instance, the LS solution to $\boldsymbol{D} \approx \boldsymbol{C} \otimes \boldsymbol{Y}$ is expressed as

$$[\boldsymbol{Y}_{\mathrm{LS}}]_{(i,j)} = \frac{\operatorname{trace}\left\{\boldsymbol{D}_{i,j}^{\mathrm{H}} \cdot \boldsymbol{C}\right\}}{\operatorname{trace}\left\{\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{C}\right\}}, \tag{B.25}$$

where $\boldsymbol{D}_{i,j} = \boldsymbol{D}(i : P : M \cdot P, j : Q : N \cdot Q)$, i.e., $\boldsymbol{D}_{i,j}$ contains $M \times N$ elements from $\boldsymbol{D}$, starting with the (i,j) element and advancing by P and Q elements along columns and rows, respectively. Certainly, (B.25) is equivalent to the direct solution given in (B.23), since $\operatorname{trace}\left\{\boldsymbol{C}^{\mathrm{H}} \cdot \boldsymbol{C}\right\} = \|\boldsymbol{C}\|_{\mathrm{F}}^2$. However, it requires the definition of the block matrices $\boldsymbol{D}_{i,j}$ which (B.23) and (B.24) avoid.

In a similar manner, equation (3.54) for finding an LS Khatri-Rao factor of $\boldsymbol{B} \approx \boldsymbol{A} \diamond \boldsymbol{X}$ can be simplified by expanding the pseudo-inverse of $[\boldsymbol{I}_N \diamond \boldsymbol{A}] \otimes \boldsymbol{I}_P$. We can first use (3.15) and then (3.22) on the remaining pseudo-inverse of $[\boldsymbol{I}_N \diamond \boldsymbol{A}]$ (note that (3.22) is applicable because the matrix is of size $M \cdot N \times N$ and has full column rank as long as none of the columns of $\boldsymbol{A}$

is equal to the zero vector). This leads to the simplified rule

$$\mathrm{vec}\{\boldsymbol{X}_{\mathrm{LS}}\} = \left[\boldsymbol{I}_N \odot \left(\boldsymbol{A}^{\mathrm{H}} \cdot \boldsymbol{A}\right)\right]^{-1} \left([\boldsymbol{I}_N \diamond \boldsymbol{A}]^{\mathrm{H}} \otimes \boldsymbol{I}_P\right) \cdot \mathrm{vec}\{\boldsymbol{B}\} \tag{B.26}$$

$$= \mathrm{diag}\left\{[\|\boldsymbol{a}_1\|_2^2, \ldots, \|\boldsymbol{a}_N\|_2^2]\right\}^{-1} \cdot \left([\boldsymbol{I}_N \diamond \boldsymbol{A}]^{\mathrm{H}} \otimes \boldsymbol{I}_P\right) \cdot \mathrm{vec}\{\boldsymbol{B}\} \tag{B.27}$$

$$, = \left(\left[\boldsymbol{I}_N \diamond \boldsymbol{A}^{(0)}\right]^{\mathrm{H}} \otimes \boldsymbol{I}_P\right) \cdot \mathrm{vec}\{\boldsymbol{B}\} \tag{B.28}$$

where $\boldsymbol{a}_n$ is the n-th column of $\boldsymbol{A}$ and $\boldsymbol{A}^{(0)} = \boldsymbol{A} \cdot \mathrm{diag}\left\{[\|\boldsymbol{a}_1\|_2^2, \ldots, \|\boldsymbol{a}_N\|_2^2]\right\}^{-1}$. Using similar arguments for the LS Khatri-Rao factorization of $\boldsymbol{B} \approx \boldsymbol{X} \diamond \boldsymbol{A}$ we find the simplified version of (3.56) as

$$\mathrm{vec}\{\boldsymbol{X}_{\mathrm{LS}}\} = \left[\boldsymbol{I}_{P \cdot N} \diamond \left(\boldsymbol{A}^{(0)} \cdot \left[\boldsymbol{I}_N \otimes \boldsymbol{1}_{P \times 1}^{\mathrm{T}}\right]\right)\right]^{\mathrm{H}} \cdot \mathrm{vec}\{\boldsymbol{B}\}. \tag{B.29}$$

B.7. Further properties of the permutation matrices

As we show in this section, the permutation matrices allow manipulations on R-fold Kronecker products such as cyclical shifts. To this end, let $\boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{A}} \bigtimes_{r=1}^{R} {}_r \boldsymbol{X}_r$, where $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \ldots \times I_R}$, $\boldsymbol{\mathcal{B}} \in \mathbb{C}^{J_1 \times J_2 \ldots \times J_R}$, and $\boldsymbol{X}_r \in \mathbb{C}^{J_r \times I_r}$. Then, applying (4.11) to $\boldsymbol{\mathcal{B}}$, we obtain

$$\begin{aligned}
\mathrm{vec}\{\boldsymbol{\mathcal{B}}\} &= \boldsymbol{P}_\beta^{(r)} \cdot \mathrm{vec}\left\{[\boldsymbol{\mathcal{B}}]_{(r)}\right\} \\
\mathrm{vec}\left\{\boldsymbol{\mathcal{A}} \bigtimes_{r=1}^{R} {}_r \boldsymbol{X}_r\right\} &= \boldsymbol{P}_\beta^{(r)} \cdot \mathrm{vec}\left\{\boldsymbol{X}_r \cdot [\boldsymbol{\mathcal{A}}]_{(r)} \cdot (\boldsymbol{X}_{r+1} \otimes \boldsymbol{X}_{r+2} \otimes \ldots \otimes \boldsymbol{X}_{r-1})^{\mathrm{T}}\right\} \\
(\boldsymbol{X}_R \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \mathrm{vec}\{\boldsymbol{\mathcal{A}}\} &= \boldsymbol{P}_\beta^{(r)} \cdot (\boldsymbol{X}_{r+1} \otimes \ldots \otimes \boldsymbol{X}_{r-1} \otimes \boldsymbol{X}_r)\, \mathrm{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\} \\
(\boldsymbol{X}_R \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \boldsymbol{P}_\alpha^{(r)} \cdot \mathrm{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\} &= \boldsymbol{P}_\beta^{(r)} \cdot (\boldsymbol{X}_{r+1} \otimes \ldots \otimes \boldsymbol{X}_{r-1} \otimes \boldsymbol{X}_r) \cdot \mathrm{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(r)}\right\} \\
\Rightarrow (\boldsymbol{X}_R \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \boldsymbol{P}_\alpha^{(r)} &= \boldsymbol{P}_\beta^{(r)} \cdot (\boldsymbol{X}_{r+1} \otimes \ldots \otimes \boldsymbol{X}_{r-1} \otimes \boldsymbol{X}_r) \qquad \text{(B.30)}
\end{aligned}$$

where we have applied (4.5) in the first step, (4.20) as well as (3.7) in the second step, and (4.11) in the third step. Here, α and β represent $(I_1, I_2, \ldots, I_R)$ and $(J_1, J_2, \ldots, J_R)$, respectively. Equation (B.30) shows the following:

- The permutation matrices $\boldsymbol{P}^{(R)}$ can reverse the order in a Kronecker product.

$$(\boldsymbol{X}_R \otimes \boldsymbol{X}_{R-1} \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \boldsymbol{P}_\alpha^{(R)} = \boldsymbol{P}_\beta^{(R)} \cdot (\boldsymbol{X}_1 \otimes \boldsymbol{X}_2 \otimes \ldots \otimes \boldsymbol{X}_R) \tag{B.31}$$

- The permutation matrices $\boldsymbol{P}^{(r)}$, $r < R$ reverse the order of the Kronecker product **and**

then perform a cyclic shift by r terms

$$\begin{aligned}(\boldsymbol{X}_R \otimes \boldsymbol{X}_{R-1} \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \boldsymbol{P}_\alpha^{(1)} &= \boldsymbol{P}_\beta^{(1)} \cdot (\boldsymbol{X}_2 \otimes \boldsymbol{X}_3 \otimes \ldots \otimes \boldsymbol{X}_R \otimes \boldsymbol{X}_1) \\ (\boldsymbol{X}_R \otimes \boldsymbol{X}_{R-1} \otimes \ldots \otimes \boldsymbol{X}_1) \cdot \boldsymbol{P}_\alpha^{(2)} &= \boldsymbol{P}_\beta^{(2)} \cdot (\boldsymbol{X}_3 \otimes \boldsymbol{X}_4 \otimes \ldots \otimes \boldsymbol{X}_R \otimes \boldsymbol{X}_1 \otimes \boldsymbol{X}_2),\end{aligned}$$

etc., where a cyclic shift by one term is the operation that maps an ordered sequence $[x_1, x_2, \ldots, x_R]$ onto the sequence $[x_2, x_3, \ldots, x_R, x_1]$ and a cyclic shift by r terms is the result of applying this operation r times.

- The inverse permutation matrix is given by

$$\left(\boldsymbol{P}_{I_1,I_2,\ldots,I_R}^{(r)}\right)^{-1} = \boldsymbol{P}_{I_1,I_2,\ldots,I_R}^{(r)\mathrm{T}} = \boldsymbol{P}_{I_r,I_{r-1},\ldots,I_1,I_R,\ldots,I_{r+1}}^{(r)}. \tag{B.32}$$

- We can define matrices $\boldsymbol{Q}^{(r)} \doteq \boldsymbol{P}^{(R)^{\mathrm{T}}} \cdot \boldsymbol{P}^{(r)}$ which perform a cyclic shift by r terms without reversing the order:[1]

$$\begin{aligned}(\boldsymbol{X}_1 \otimes \boldsymbol{X}_2 \otimes \ldots \otimes \boldsymbol{X}_R) \cdot \boldsymbol{Q}_\alpha^{(1)} &= \boldsymbol{Q}_\beta^{(1)} \cdot (\boldsymbol{X}_2 \otimes \boldsymbol{X}_3 \otimes \ldots \otimes \boldsymbol{X}_R \otimes \boldsymbol{X}_1) \\ (\boldsymbol{X}_1 \otimes \boldsymbol{X}_2 \otimes \ldots \otimes \boldsymbol{X}_R) \cdot \boldsymbol{Q}_\alpha^{(2)} &= \boldsymbol{Q}_\beta^{(2)} \cdot (\boldsymbol{X}_3 \otimes \boldsymbol{X}_4 \otimes \ldots \otimes \boldsymbol{X}_R \otimes \boldsymbol{X}_1 \otimes \boldsymbol{X}_2),\end{aligned}$$

etc. Note that [MN79] shows that for the special case $R = 3$ such a cyclical shift can be achieved by virtue of commutation matrices in the following fashion:

$$\begin{aligned}(\boldsymbol{X}_1 \otimes \boldsymbol{X}_2 \otimes \boldsymbol{X}_3) \cdot \boldsymbol{K}_{I_1,I_2 \cdot I_3} &= \boldsymbol{K}_{J_1,J_2 \cdot J_3} (\boldsymbol{X}_2 \otimes \boldsymbol{X}_3 \otimes \boldsymbol{X}_1) \\ (\boldsymbol{X}_1 \otimes \boldsymbol{X}_2 \otimes \boldsymbol{X}_3) \cdot \boldsymbol{K}_{I_1 \cdot I_2,I_3} &= \boldsymbol{K}_{J_1 \cdot J_2,J_3} (\boldsymbol{X}_3 \otimes \boldsymbol{X}_1 \otimes \boldsymbol{X}_2).\end{aligned}$$

A corollary that follows from this is that the permutation matrices satisfy

$$\boldsymbol{K}_{I_1,I_2 \cdot I_3} = \boldsymbol{P}_{I_1,I_2,I_3}^{(R)^{\mathrm{T}}} \cdot \boldsymbol{P}_{I_1,I_2,I_3}^{(1)} = \boldsymbol{Q}_{I_1,I_2,I_3}^{(1)} = \boldsymbol{P}_{I_3,I_2,I_1}^{(R)} \cdot \boldsymbol{P}_{I_1,I_2,I_3}^{(1)} \tag{B.33}$$

$$\boldsymbol{K}_{I_1 \cdot I_2,I_3} = \boldsymbol{P}_{I_1,I_2,I_3}^{(R)^{\mathrm{T}}} \cdot \boldsymbol{P}_{I_1,I_2,I_3}^{(2)} = \boldsymbol{Q}_{I_1,I_2,I_3}^{(2)} = \boldsymbol{P}_{I_3,I_2,I_1}^{(R)} \cdot \boldsymbol{P}_{I_1,I_2,I_3}^{(2)}. \tag{B.34}$$

B.8. Proof for Theorem 4.1.1

Proving this theorem is mainly a manner of enumerating the elements of the tensor and its vectorized versions and then comparing the different results. For instance, to show (4.13), we

[1] The reason for the order-reversing to be included in the definition of the permutation matrices is that the reverse cyclical column-ordering rule for the matrix unfoldings from [dLdMV00b] is used for their definition.

can express $[\boldsymbol{\mathcal{A}}]_{(1)}$ as

$$[\boldsymbol{\mathcal{A}}]_{(1)} = [\boldsymbol{a}_{1,1}, \boldsymbol{a}_{1,2}, \ldots, \boldsymbol{a}_{1,M_3}, \boldsymbol{a}_{2,1}, \boldsymbol{a}_{2,2}, \ldots, \boldsymbol{a}_{M_2,M_3}] \tag{B.35}$$

where $\boldsymbol{a}_{k,\ell}$ are the 1-mode vectors of $\boldsymbol{\mathcal{A}}$, i.e., $\boldsymbol{a}_{k,\ell} = [a_{1,k,\ell}, a_{2,k,\ell}, \ldots, a_{M_1,k,\ell}]^{\mathrm{T}}$. Here, $a_{j,k,\ell}$ denotes the (j,k,ℓ)-element of $\boldsymbol{\mathcal{A}}$ for $j = 1,2,\ldots,M_1$, $k = 1,2,\ldots,M_2$ and $\ell = 1,2,\ldots,M_3$. Using this definition, $[\boldsymbol{\mathcal{A}}]_{(2)}^{\mathrm{T}}$ becomes

$$[\boldsymbol{\mathcal{A}}]_{(2)}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{a}_{1,1} & \boldsymbol{a}_{2,1} & \cdots & \boldsymbol{a}_{M_2,1} \\ \boldsymbol{a}_{1,2} & \boldsymbol{a}_{2,2} & \cdots & \boldsymbol{a}_{M_2,2} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{a}_{1,M_3} & \boldsymbol{a}_{2,M_3} & \cdots & \boldsymbol{a}_{M_2,M_3} \end{bmatrix} \tag{B.36}$$

Obviously, vectorizing (B.35) and (B.36) gives the same result, namely, the vectors $\boldsymbol{a}_{k,\ell}$ stacked on top of each other, where first the second index ℓ is varied and then the first index k.

Identities (4.14) and (4.15) are shown using similar arguments for the 2-mode vectors and the 3-mode vectors of $\boldsymbol{\mathcal{A}}$, respectively. $\square$

Appendix C.

Proofs and derivations for Part II

C.1. Proof of Theorem 7.1.1

To show the essential uniqueness of the SMD-based estimates of the loading matrices used in SECSI, we first need the following lemma about the uniqueness of the SMD:

Lemma C.1.1. *Consider the exact (noise-free) SMD of K matrix slices $\boldsymbol{X}_k = \boldsymbol{T} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{T}^{-1}$, $k = 1, 2, \ldots, K$. Then, the matrix $\boldsymbol{T}$ is identified only up to a permutation and a scaling ambiguity of its columns and the matrix $\boldsymbol{\Lambda}_n$ is identified only up to the same permutation of its diagonal elements.*

Proof. The existence of the ambiguities follows from the observation that for every slice k we have $\boldsymbol{T} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{T}^{-1} = \bar{\boldsymbol{T}} \cdot \bar{\boldsymbol{\Lambda}}_k \cdot \bar{\boldsymbol{T}}^{-1}$, where $\bar{\boldsymbol{T}} = \boldsymbol{T} \cdot \boldsymbol{P} \cdot \boldsymbol{D}$ and $\bar{\boldsymbol{\Lambda}}_k = \boldsymbol{P}^{-1} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{P}$. Here, $\boldsymbol{D} = \operatorname{diag}\{\boldsymbol{d}\}$ is an arbitrary diagonal matrix and $\boldsymbol{P}$ is a permutation matrix. Since $\boldsymbol{D}$ is diagonal we have $\boldsymbol{P} \cdot \boldsymbol{D} \cdot \boldsymbol{P}^{-1} = \operatorname{diag}\{\boldsymbol{P} \cdot \boldsymbol{d}\}$, i.e., $\boldsymbol{P} \cdot \boldsymbol{D} \cdot \boldsymbol{P}^{-1}$ is again a diagonal matrix with permuted elements on the main diagonal. Therefore, we can write

$$\begin{aligned} \bar{\boldsymbol{T}} \cdot \bar{\boldsymbol{\Lambda}}_k \cdot \bar{\boldsymbol{T}}^{-1} &= \boldsymbol{T} \cdot \boldsymbol{P} \cdot \boldsymbol{D} \cdot \boldsymbol{P}^{-1} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{P} \cdot \boldsymbol{D}^{-1} \cdot \boldsymbol{P}^{-1} \cdot \boldsymbol{T}^{-1} \\ &= \boldsymbol{T} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{P} \cdot \boldsymbol{D} \cdot \boldsymbol{P}^{-1} \cdot \boldsymbol{P} \cdot \boldsymbol{D}^{-1} \cdot \boldsymbol{P}^{-1} \cdot \boldsymbol{T}^{-1} \\ &= \boldsymbol{T} \cdot \boldsymbol{\Lambda}_k \cdot \boldsymbol{T}^{-1}, \end{aligned} \tag{C.1}$$

where in the first step we have used the fact that the product of diagonal matrices commutes. □

The theorem can be shown by applying the lemma to the SMDs solved in the SECSI framework. As an example, consider the SMD for $\boldsymbol{S}_{3,k}^{\text{lhs}}$ shown in (7.12). From Lemma C.1.1 we know that instead of $\boldsymbol{T}_1$ we may find the matrix $\bar{\boldsymbol{T}}_1 = \boldsymbol{T}_1 \cdot \boldsymbol{P} \cdot \boldsymbol{D}_1$.

Since we find $\hat{\boldsymbol{F}}^{(1)}$ based on the ambiguous estimate $\bar{\boldsymbol{T}}_1$ via $\hat{\boldsymbol{F}}^{(1)} = \boldsymbol{U}_1^{[\text{s}]} \cdot \bar{\boldsymbol{T}}_1$, we have

$$\hat{\boldsymbol{F}}^{(1)} = \boldsymbol{U}_1^{[\text{s}]} \cdot \bar{\boldsymbol{T}}_1 = \boldsymbol{U}_1^{[\text{s}]} \cdot \boldsymbol{T}_1 \cdot \boldsymbol{P} \cdot \boldsymbol{D}_1 = \boldsymbol{F}^{(1)} \cdot \boldsymbol{P} \cdot \boldsymbol{D}_1. \tag{C.2}$$

The matrix $\tilde{\boldsymbol{F}}^{(3)}$ is found by row-wise augmentation of the diagonals of $\boldsymbol{\Lambda}_k$. Since they are permuted by $\boldsymbol{P}$, we identify $\tilde{\boldsymbol{F}}^{(3)}$ up to a permutation ambiguity of its columns. Moreover, as $\tilde{\boldsymbol{F}}^{(3)}$ is a scaled version of $\boldsymbol{F}^{(3)}$ where the s_3-th row is scaled to one, a corresponding scaling ambiguity in identifying $\boldsymbol{F}^{(3)}$ is also present. Consequently, we have

$$\hat{\boldsymbol{F}}^{(3)} = \boldsymbol{F}^{(3)} \cdot \boldsymbol{P} \cdot \boldsymbol{D}_3. \tag{C.3}$$

The matrix $\boldsymbol{F}^{(2)}$ is then found via an LS fit according to $\hat{\boldsymbol{F}}^{(2)} = [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\hat{\boldsymbol{F}}^{(3)} \diamond \hat{\boldsymbol{F}}^{(1)}\right)^{+}\right]^{\mathrm{T}}$. Inserting (C.2) and (C.3) we obtain

$$\begin{aligned}
\hat{\boldsymbol{F}}^{(2)} &= [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\left(\boldsymbol{F}^{(3)} \cdot \boldsymbol{P} \cdot \boldsymbol{D}_3\right) \diamond \left(\boldsymbol{F}^{(1)} \cdot \boldsymbol{P} \cdot \boldsymbol{D}_1\right)\right)^{+}\right]^{\mathrm{T}} \\
&= [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\left[\left(\boldsymbol{F}^{(3)} \cdot \boldsymbol{P}\right) \diamond \left(\boldsymbol{F}^{(1)} \cdot \boldsymbol{P}\right)\right] \cdot \boldsymbol{D}_3 \cdot \boldsymbol{D}_1\right)^{+}\right]^{\mathrm{T}} \\
&= [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\left[\boldsymbol{F}^{(3)} \diamond \boldsymbol{F}^{(1)}\right] \cdot \boldsymbol{P} \cdot \boldsymbol{D}_3 \cdot \boldsymbol{D}_1\right)^{+}\right]^{\mathrm{T}} \\
&= [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\boldsymbol{P} \cdot \boldsymbol{D}_3 \cdot \boldsymbol{D}_1\right)^{-1} \cdot \left[\boldsymbol{F}^{(3)} \diamond \boldsymbol{F}^{(1)}\right]^{+}\right]^{\mathrm{T}} \\
&= [\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left(\boldsymbol{D}_3 \cdot \boldsymbol{D}_1\right)^{-1} \cdot \boldsymbol{P}^{\mathrm{T}} \cdot \left[\boldsymbol{F}^{(3)} \diamond \boldsymbol{F}^{(1)}\right]^{+}\right]^{\mathrm{T}} \\
&= \underbrace{[\boldsymbol{\mathcal{X}}]_{(2)} \cdot \left[\left[\boldsymbol{F}^{(3)} \diamond \boldsymbol{F}^{(1)}\right]^{+}\right]^{\mathrm{T}}}_{\boldsymbol{F}^{(2)}} \cdot \boldsymbol{P} \cdot \left(\boldsymbol{D}_1 \cdot \boldsymbol{D}_3\right)^{-1} \\
&= \boldsymbol{F}^{(2)} \cdot \boldsymbol{P} \cdot \left(\boldsymbol{D}_1 \cdot \boldsymbol{D}_3\right)^{-1},
\end{aligned} \tag{C.4}$$

where in the first step we have used (3.20), the second step follows from the fact that the Khatri-Rao product is computed column-wise and hence the column permutation can be moved to the right, and in the 4th step we have used that $\boldsymbol{P}^{-1} = \boldsymbol{P}^{\mathrm{T}}$, since $\boldsymbol{P}$ is a permutation matrix. This concludes the proof of the theorem. $\square$

Appendix D.

Proofs and derivations for Part III

D.1. A CP formulation for the array steering tensor

In this appendix we show that the definition of the array steering tensor via repeated $(R+1)$-mode concatenations given in (9.12) can be rewritten in form of a CP decomposition, as discussed in Section 4.2. To this end, note that we can write $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1 \times \ldots \times M_R \times d}$ as $\boldsymbol{\mathcal{A}} \times_{R+1} \boldsymbol{I}_d$. Inserting $\boldsymbol{\mathcal{A}} = [\boldsymbol{\mathcal{A}}_1 \sqcup_{R+1} \boldsymbol{\mathcal{A}}_2 \sqcup_{R+1} \ldots \sqcup_{R+1} \boldsymbol{\mathcal{A}}_d]$ from (9.12) we obtain

$$\begin{aligned}
\boldsymbol{\mathcal{A}} &= [\boldsymbol{\mathcal{A}}_1 \sqcup_{R+1} \boldsymbol{\mathcal{A}}_2 \sqcup_{R+1} \ldots \sqcup_{R+1} \boldsymbol{\mathcal{A}}_d] \times_{R+1} \boldsymbol{I}_d \\
&= [\boldsymbol{\mathcal{A}}_1 \sqcup_{R+1} \boldsymbol{\mathcal{A}}_2 \sqcup_{R+1} \ldots \sqcup_{R+1} \boldsymbol{\mathcal{A}}_d] \times_{R+1} \begin{bmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \ldots & \boldsymbol{e}_d \end{bmatrix} \\
&= \sum_{i=1}^{d} \boldsymbol{\mathcal{A}}_i \times_{R+1} \boldsymbol{e}_i \\
&= \sum_{i=1}^{d} \boldsymbol{\mathcal{A}}_i \circ \boldsymbol{e}_i,
\end{aligned} \tag{D.1}$$

where from the second to the third line we have applied (4.10) and $\boldsymbol{e}_i$ represents the i-th column of $\boldsymbol{I}_d$. Moreover, in the last step we have used the fact that the r-mode product with a column vector is the same as an outer product. Inserting (9.13) into (D.1) we obtain

$$\boldsymbol{\mathcal{A}} = \sum_{i=1}^{d} \boldsymbol{a}_i^{(1)} \circ \boldsymbol{a}_i^{(2)} \circ \ldots \circ \boldsymbol{a}_i^{(R)} \circ \boldsymbol{e}_i, \tag{D.2}$$

which is a CP decomposition in component form as shown in (4.30). Via (4.34) this can alternatively be expressed as

$$\begin{aligned}
\boldsymbol{\mathcal{A}} &= \boldsymbol{\mathcal{I}}_{R+1,d} \times_1 \boldsymbol{A}^{(1)} \ldots \times_R \boldsymbol{A}^{(R)} \times_{R+1} \begin{bmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \ldots & \boldsymbol{e}_d \end{bmatrix} \\
&= \boldsymbol{\mathcal{I}}_{R+1,d} \times_1 \boldsymbol{A}^{(1)} \ldots \times_R \boldsymbol{A}^{(R)} \times_{R+1} \boldsymbol{I}_d \\
&= \boldsymbol{\mathcal{I}}_{R+1,d} \times_1 \boldsymbol{A}^{(1)} \ldots \times_R \boldsymbol{A}^{(R)},
\end{aligned} \tag{D.3}$$

which shows the validity of (9.15).

D.2. Notes on 2-D arrays with equal beampatterns

In this appendix we comment on the separability and the centro-symmetry of 2-D arrays (cf. Section 9.2) where the elements have arbitrary beam patterns. Let $\boldsymbol{A} \in \mathbb{C}^{M\times d}$ be the array steering matrix of the antenna array. Then $\boldsymbol{A}$ can be written as

$$\boldsymbol{A} = \boldsymbol{G} \odot \bar{\boldsymbol{A}}, \tag{D.4}$$

where $\boldsymbol{G} \in \mathbb{C}^{M\times d}$ contains the beam patterns of the M antenna elements for the d sources, i.e., $[\boldsymbol{G}]_{(m,n)} = g_m(\theta_n, \alpha_n)$ for $m = 1, 2, \dots, M$ and $n = 1, 2, \dots d$. Moreover, $\bar{\boldsymbol{A}} \in \mathbb{C}^{M\times d}$ is the "normalized" array steering matrix which only contains the phase offsets between different elements due to their displacement. For instance, for an $M_1 \times M_2$ Uniform Rectangular Array (URA), $\bar{\boldsymbol{A}} = \bar{\boldsymbol{A}}^{(1)} \diamond \bar{\boldsymbol{A}}^{(2)}$, where $\bar{\boldsymbol{A}}^{(r)}$ are Vandermonde matrices of the form

$$\bar{\boldsymbol{A}}^{(r)} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \mathrm{e}^{\jmath\cdot\mu_1^{(r)}} & \mathrm{e}^{\jmath\cdot\mu_2(r)} & \cdots & \mathrm{e}^{\jmath\cdot\mu_d^{(r)}} \\ \vdots & \vdots & \vdots & \vdots \\ \mathrm{e}^{\jmath\cdot(M_r-1)\cdot\mu_1^{(r)}} & \mathrm{e}^{\jmath\cdot(M_r-1)\cdot\mu_2(r)} & \cdots & \mathrm{e}^{\jmath\cdot(M_r-1)\cdot\mu_d^{(r)}} \end{bmatrix}. \tag{D.5}$$

In general, for (D.4) to be separable, we need $\bar{\boldsymbol{A}}$ to be separable (for which the geometric alignment of the antenna elements needs to be separable, cf. Figure 9.3) and $\boldsymbol{G}$ to be separable, i.e., $\boldsymbol{G} = \boldsymbol{G}^{(1)} \diamond \boldsymbol{G}^{(2)}$. However, this is not necessary if all the elements share the *same beam pattern*, as we demonstrate next. In such a case, all the rows of $\boldsymbol{G}$ are identical and we can write

$$\boldsymbol{G} = \boldsymbol{1}_M \cdot \boldsymbol{g}^{\mathrm{T}}, \tag{D.6}$$

where $\boldsymbol{g}^{\mathrm{T}} = \begin{bmatrix} g(\theta_1, \alpha_1) & \dots & g(\theta_d, \alpha_d) \end{bmatrix} \in \mathbb{C}^{1\times d}$. Inserting (D.6) into (D.4) we can apply (3.21) to write $\boldsymbol{A}$ as

$$\boldsymbol{A} = \left(\boldsymbol{1}_M \cdot \boldsymbol{g}^{\mathrm{T}}\right) \odot \bar{\boldsymbol{A}} = \underbrace{\mathrm{diag}\left\{\boldsymbol{1}_M\right\}}_{\boldsymbol{I}_M} \cdot \bar{\boldsymbol{A}} \cdot \mathrm{diag}\left\{\boldsymbol{g}\right\} = \bar{\boldsymbol{A}} \cdot \mathrm{diag}\left\{\boldsymbol{g}\right\}. \tag{D.7}$$

However, the scaling by $\mathrm{diag}\{\boldsymbol{g}\}$ is irrelevant since on our data model $\boldsymbol{X} = \boldsymbol{A}\cdot\boldsymbol{S}+\boldsymbol{N}$ from (9.16) it can be incorporated into the symbol matrix $\boldsymbol{S}$. Therefore, it is sufficient that $\bar{\boldsymbol{A}}$ is separable into $\left(\bar{\boldsymbol{A}}^{(1)} \diamond \bar{\boldsymbol{A}}^{(2)}\right)$ if all elements share the same (arbitrary) complex beam pattern $g(\theta, \alpha)$.

Note that a similar argument is true for centro-symmetry. If the antenna configuration itself is centro-symmetric, i.e., $\boldsymbol{\Pi}_M \cdot \bar{\boldsymbol{A}}^* = \bar{\boldsymbol{A}} \cdot \bar{\boldsymbol{\Delta}}$, we can allow for arbitrary complex beam patterns $g(\theta, \alpha)$ as long as they are common to all elements. This is easily seen by inserting (D.7) into the condition (9.20)

$$\begin{aligned}\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* &= \boldsymbol{A} \cdot \boldsymbol{\Delta} \\ \Leftrightarrow \boldsymbol{\Pi}_M \cdot \bar{\boldsymbol{A}}^* \cdot \operatorname{diag}\{\boldsymbol{g}^*\} &= \bar{\boldsymbol{A}} \cdot \operatorname{diag}\{\boldsymbol{g}\} \cdot \boldsymbol{\Delta} \\ \Leftrightarrow \bar{\boldsymbol{A}} \cdot \bar{\boldsymbol{\Delta}} \cdot \operatorname{diag}\{\boldsymbol{g}^*\} &= \bar{\boldsymbol{A}} \cdot \operatorname{diag}\{\boldsymbol{g}\} \cdot \boldsymbol{\Delta} \\ \Leftrightarrow \boldsymbol{\Delta} &= \operatorname{diag}\{\boldsymbol{g}\}^+ \cdot \bar{\boldsymbol{\Delta}} \cdot \operatorname{diag}\{\boldsymbol{g}^*\}. \end{aligned} \tag{D.8}$$

This shows that $\boldsymbol{\Pi}_M \cdot \bar{\boldsymbol{A}}^* = \bar{\boldsymbol{A}} \cdot \bar{\boldsymbol{\Delta}}$ already implies the condition for centro-symmetry of the whole array $\boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* = \boldsymbol{A} \cdot \boldsymbol{\Delta}$ if all the elements share the same beam pattern and we choose $\boldsymbol{\Delta}$ according to (D.8).

D.3. Proof of Theorem 10.2.1

As shown in (10.10), the estimated signal subspace tensor can be computed via

$$\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \ldots \times_R \hat{\boldsymbol{U}}_R^{[\mathrm{s}]} \times_{R+1} \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}}. \tag{D.9}$$

Here, $\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]}$ represents the truncated version core tensor $\hat{\boldsymbol{\mathcal{S}}}$ from the HOSVD of $\boldsymbol{\mathcal{X}}$. In order to eliminate $\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]}$ we require the following lemma:

Lemma D.3.1. *The truncated core tensor $\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]}$ can be computed from $\boldsymbol{\mathcal{X}}$ directly via*

$$\hat{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{X}} \times_1 \hat{\boldsymbol{U}}_1^{[\mathrm{s}]^{\mathrm{H}}} \ldots \times_{R+1} \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}}. \tag{D.10}$$

Proof. To show (D.10) we insert the HOSVD of $\boldsymbol{\mathcal{X}}$ given by $\boldsymbol{\mathcal{X}} = \hat{\boldsymbol{\mathcal{S}}} \times_1 \hat{\boldsymbol{U}}_1 \ldots \times_{R+1} \hat{\boldsymbol{U}}_{R+1}$. Using the rules for n-mode products (cf. Section 4.1), we obtain

$$\hat{\boldsymbol{\mathcal{S}}} = \boldsymbol{\mathcal{S}} \times_1 \left(\hat{\boldsymbol{U}}_1^{[\mathrm{s}]^{\mathrm{H}}} \cdot \hat{\boldsymbol{U}}_1\right) \ldots \times_{R+1} \left(\hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}} \cdot \hat{\boldsymbol{U}}_{R+1}\right). \tag{D.11}$$

However, since the matrices of r-mode singular vectors $\hat{\boldsymbol{U}}_r$ are unitary, they satisfy $\hat{\boldsymbol{U}}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \hat{\boldsymbol{U}}_r = \left[\boldsymbol{I}_{p_r}, \, \boldsymbol{0}_{p_r \times (M_r - p_r)}\right]$. Consequently, we have

$$\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]} = \hat{\boldsymbol{\mathcal{S}}} \times_1 \left[\boldsymbol{I}_{p_1}, \, \boldsymbol{0}_{p_1 \times (M_1 - p_1)}\right] \ldots \times_{R+1} \left[\boldsymbol{I}_{p_{R+1}}, \, \boldsymbol{0}_{p_{R+1} \times (M_{R+1} - p_{R+1})}\right]. \tag{D.12}$$

Therefore, $\hat{\boldsymbol{\mathcal{S}}}^{[\mathrm{s}]}$ computed via (D.10) contains the first p_r elements of $\hat{\boldsymbol{\mathcal{S}}}$ in the r-th mode, which shows that it is indeed the truncated core tensor. $\square$

Next, we use Lemma D.3.1 to eliminate $\hat{\boldsymbol{\mathcal{S}}}$ in (D.9). We obtain

$$\begin{aligned}\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} &= \boldsymbol{\mathcal{X}} \times_1 \left(\hat{\boldsymbol{U}}_1^{[\mathrm{s}]} \cdot \hat{\boldsymbol{U}}_1^{[\mathrm{s}]^{\mathrm{H}}}\right) \ldots \times_R \left(\hat{\boldsymbol{U}}_R^{[\mathrm{s}]} \cdot \hat{\boldsymbol{U}}_R^{[\mathrm{s}]^{\mathrm{H}}}\right) \times_{R+1} \left(\hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}} \cdot \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}}\right) \\ &= \boldsymbol{\mathcal{X}} \times_1 \hat{\boldsymbol{T}}_1 \ldots \times_R \hat{\boldsymbol{T}}_R \times_{R+1} \left(\hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}} \cdot \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}}\right),\end{aligned} \tag{D.13}$$

where we have introduced the short hand notation $\hat{\boldsymbol{T}}_r = \hat{\boldsymbol{U}}_r^{[\mathrm{s}]} \cdot \hat{\boldsymbol{U}}_r^{[\mathrm{s}]^{\mathrm{H}}}$.

The next step is to compute the matrix $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$. Inserting (D.13) and using the rules for the unfoldings of repeated n-mode products, we obtain

$$\begin{aligned}\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}} &= \left[\left(\hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}} \cdot \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^{\mathrm{H}}}\right) \cdot [\boldsymbol{\mathcal{X}}]_{(R+1)} \cdot \left(\hat{\boldsymbol{T}}_1 \otimes \ldots \otimes \hat{\boldsymbol{T}}_R\right)^{\mathrm{T}}\right]^{\mathrm{T}} \\ &= \left(\hat{\boldsymbol{T}}_1 \otimes \ldots \otimes \hat{\boldsymbol{T}}_R\right) \cdot [\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{T}} \cdot \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^*} \cdot \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}},\end{aligned} \tag{D.14}$$

where in the last step we have used the fact that the transpose operator has no effect on $\hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}}$ since this matrix is diagonal.

As pointed out in Section 10.2, the link between the measurement matrix $\boldsymbol{X}$ and the measurement tensor $\boldsymbol{\mathcal{X}}$ is given by $\boldsymbol{X} = [\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{T}}$. Note that their SVDs are given by

$$\boldsymbol{X} = \hat{\boldsymbol{U}} \cdot \hat{\boldsymbol{\Sigma}} \cdot \hat{\boldsymbol{V}}^{\mathrm{H}} = \left[\hat{\boldsymbol{U}}_{\mathrm{s}}, \quad \hat{\boldsymbol{U}}_{\mathrm{n}}\right] \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{\mathrm{s}} & \boldsymbol{0}_{d\times(N-d)} \\ \boldsymbol{0}_{(M-d)\times d} & \hat{\boldsymbol{\Sigma}}_{\mathrm{n}} \end{bmatrix} \cdot \left[\hat{\boldsymbol{V}}_{\mathrm{s}}, \quad \hat{\boldsymbol{V}}_{\mathrm{n}}\right]^{\mathrm{H}} \tag{D.15}$$

$$[\boldsymbol{\mathcal{X}}]_{(R+1)} = \left[\hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]}, \quad \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{n}]}\right] \cdot \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]} & \boldsymbol{0}_{d\times(N-d)} \\ \boldsymbol{0}_{(M-d)\times d} & \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{n}]} \end{bmatrix} \cdot \left[\hat{\boldsymbol{V}}_{R+1}^{[\mathrm{s}]}, \quad \hat{\boldsymbol{V}}_{R+1}^{[\mathrm{n}]}\right]^{\mathrm{H}} \tag{D.16}$$

and hence linked through the following identities

$$\hat{\boldsymbol{U}}_{\mathrm{s}} = \hat{\boldsymbol{V}}_{R+1}^{[\mathrm{s}]^*},\ \hat{\boldsymbol{U}}_{\mathrm{n}} = \hat{\boldsymbol{V}}_{R+1}^{[\mathrm{n}]^*},\ \hat{\boldsymbol{V}}_{\mathrm{s}} = \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^*},\ \hat{\boldsymbol{V}}_{\mathrm{n}} = \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{n}]^*},\ \hat{\boldsymbol{\Sigma}}_{\mathrm{s}} = \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]}. \tag{D.17}$$

Consequently we can write

$$[\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{T}} \cdot \hat{\boldsymbol{U}}_{R+1}^{[\mathrm{s}]^*} \cdot \hat{\boldsymbol{\Sigma}}_{R+1}^{[\mathrm{s}]^{-1}} = \boldsymbol{X} \cdot \hat{\boldsymbol{V}}_{\mathrm{s}} \cdot \hat{\boldsymbol{\Sigma}}_{\mathrm{s}}^{-1} = \hat{\boldsymbol{U}} \cdot \hat{\boldsymbol{\Sigma}} \cdot \hat{\boldsymbol{V}}^{\mathrm{H}} \cdot \hat{\boldsymbol{V}}_{\mathrm{s}} \cdot \hat{\boldsymbol{\Sigma}}_{\mathrm{s}}^{-1} = \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \hat{\boldsymbol{\Sigma}}_{s} \cdot \hat{\boldsymbol{\Sigma}}_{\mathrm{s}}^{-1} = \hat{\boldsymbol{U}}_{\mathrm{s}}, \tag{D.18}$$

where we have use the orthonormality of $\hat{\boldsymbol{V}}$ which implies $\hat{\boldsymbol{V}}^{\mathrm{H}} \cdot \hat{\boldsymbol{V}}_{\mathrm{s}} = \left[\boldsymbol{I}_d, \ \boldsymbol{0}_{d\times(N-d)}\right]$. Finally, inserting (D.18) into (D.14) yields

$$\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)} = \left(\hat{\boldsymbol{T}}_1 \otimes \ldots \otimes \hat{\boldsymbol{T}}_R\right) \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}, \tag{D.19}$$

which is the desired result. □

Corollary D.3.2. *A corollary which follows from this theorem is that the exact subspace $\boldsymbol{U}_{\mathrm{s}}$ satisfies the following identity*

$$\boldsymbol{U}_{\mathrm{s}} = \left(\boldsymbol{T}_1 \otimes \ldots \otimes \boldsymbol{T}_R\right) \cdot \boldsymbol{U}_{\mathrm{s}}. \tag{D.20}$$

Proof. The corollary follows by considering the special case where $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0$ and hence $\hat{\boldsymbol{T}}_r = \boldsymbol{T}_r$ as well as $\hat{\boldsymbol{U}}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{s}}$. For this case we also have $\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)} = \left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)} = \boldsymbol{U}_{\mathrm{s}}$, where the last identity resembles the fact that in the noise-free case, the HOSVD-based subspace estimate coincides with the SVD-bases subspace estimate. □

D.4. Proof of Theorem 11.4.2

We begin by eliminating the Higher-Order Frobenius norm in the tensor-valued Least Squares problem. Considering the fact that it is defined as the square-root of the sum of the squared magnitude of all elements it is immediately evident that the Higher-Order Frobenius norm of a tensor is equal to the Frobenius norm of an arbitrary rearrangement of all the elements into a matrix. Specifically, we have for arbitrary tensors $\boldsymbol{\mathcal{Z}} \in \mathbb{C}^{I_1 \times I_2 \ldots \times I_R}$ the identity $\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{H}} = \left\|[\boldsymbol{\mathcal{Z}}]_{(r)}\right\|_{\mathrm{F}} = \left\|[\boldsymbol{\mathcal{Z}}]^{\mathrm{T}}_{(r)}\right\|_{\mathrm{F}}$, $\forall r = 1, 2, \ldots, R$, i.e., the Higher-Order Frobenius norm of $\boldsymbol{\mathcal{Z}}$ is equal to the Frobenius norm of all its unfoldings.

Based on this idea we can reformulate any tensor-valued LS problem into a matrix-valued LS problem. For the problem (11.13) at hand we use the $(R+1)$-mode unfolding and obtain

$$\begin{aligned}
\hat{\boldsymbol{\Psi}}^{(r)}_{\mathrm{LS}} = \arg\max_{\boldsymbol{\Psi}} \Bigg\| \boldsymbol{\Psi} \cdot \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \left(\boldsymbol{I}_{M_1} \otimes \ldots \otimes \boldsymbol{I}_{M_r-1} \otimes \boldsymbol{J}^{(r)}_1 \otimes \boldsymbol{I}_{M_{r+1}} \otimes \ldots \otimes \boldsymbol{I}_{M_R}\right)^{\mathrm{T}} \\
- \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \left(\boldsymbol{I}_{M_1} \otimes \ldots \otimes \boldsymbol{I}_{M_r-1} \otimes \boldsymbol{J}^{(r)}_2 \otimes \boldsymbol{I}_{M_{r+1}} \otimes \ldots \otimes \boldsymbol{I}_{M_R}\right)^{\mathrm{T}} \Bigg\|^2_{\mathrm{F}} \\
= \arg\max_{\boldsymbol{\Psi}} \left\| \boldsymbol{\Psi} \cdot \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \tilde{\boldsymbol{J}}^{(r)^{\mathrm{T}}}_1 - \left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \tilde{\boldsymbol{J}}^{(r)^{\mathrm{T}}}_2 \right\|^2_{\mathrm{F}}
\end{aligned} \tag{D.21}$$

where we have used (4.5) in the first step and the definition of $\tilde{\boldsymbol{J}}_n^{(r)}$ from (11.5) in the second step. Equation (D.21) is a matrix-valued Least Squares problem of the form $\arg\min_{\boldsymbol{X}} \|\boldsymbol{X}\boldsymbol{A} - \boldsymbol{B}\|_{\mathrm{F}}^2$ which has the closed-form solution $\boldsymbol{X}_{\mathrm{LS}} = \boldsymbol{B} \cdot \boldsymbol{A}^+$. Consequently, we obtain

$$\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)} = \left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \tilde{\boldsymbol{J}}_2^{(r)^{\mathrm{T}}} \cdot \left(\left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)} \cdot \tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{T}}}\right)^+ \tag{D.22}$$

Since the pseudo-inverse of a matrix satisfies $(\boldsymbol{A}^{\mathrm{T}})^+ = (\boldsymbol{A}^+)^{\mathrm{T}}$, we can simplify (D.22) into

$$\begin{aligned}\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(r)} &= \left[\tilde{\boldsymbol{J}}_2^{(r)} \cdot \left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right]^{\mathrm{T}} \cdot \left[\left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right)^+\right]^{\mathrm{T}} \\ &= \left[\left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right)^+ \tilde{\boldsymbol{J}}_2^{(r)} \cdot \left[\hat{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right]^{\mathrm{T}},\end{aligned} \tag{D.23}$$

which proves the theorem. □

D.5. Proof of Theorem 11.5.1

According to Theorem 11.5.1 we have to show that (11.26) is true. Inserting (11.27) into (11.26) this is equivalent to showing

$$\begin{gathered}\begin{bmatrix}\boldsymbol{J}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{\Pi}_M\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{A} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*\end{bmatrix} \cdot \boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{J}_2 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{\Pi}_M\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{A} \\ \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*\end{bmatrix} \\ \begin{bmatrix}\boldsymbol{J}_1 \cdot \boldsymbol{A} \\ \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{\Pi}_M \cdot \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*\end{bmatrix} \cdot \boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{J}_2 \cdot \boldsymbol{A} \\ \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{\Pi}_M \cdot \boldsymbol{\Pi}_M \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*\end{bmatrix}\end{gathered} \tag{D.24}$$

The first $M^{(\mathrm{sel})}$ rows of (D.24) are given by $\boldsymbol{J}_1 \cdot \boldsymbol{A} \cdot \boldsymbol{\Phi} = \boldsymbol{J}_2 \cdot \boldsymbol{A}$, which was assumed for the Theorem. The second $M^{(\mathrm{sel})}$ rows can be simplified by multiplying from the left with $\boldsymbol{\Pi}_{M^{(\mathrm{sel})}}$ and then using the fact that $\boldsymbol{\Pi}_M \cdot \boldsymbol{\Pi}_M = \boldsymbol{I}_M$. We obtain

$$\boldsymbol{J}_2 \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Phi} = \boldsymbol{J}_1 \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^* \tag{D.25}$$

Since $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$ are diagonal, their order can be exchanged. Multiplying with $\boldsymbol{\Psi}$ from the right hand side twice then cancels $\boldsymbol{\Psi}$ since $\boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi} = \boldsymbol{I}_d$. We are left with

$$\boldsymbol{J}_2 \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Phi} = \boldsymbol{J}_1 \cdot \boldsymbol{A}^*$$

$$\boldsymbol{J}_2 \cdot \boldsymbol{A}^* = \boldsymbol{J}_1 \cdot \boldsymbol{A}^* \cdot \boldsymbol{\Phi}^*, \tag{D.26}$$

where in the last step we have multiplied with $\boldsymbol{\Phi}^*$ from the right-hand side and used the fact that $\boldsymbol{\Phi}^* \cdot \boldsymbol{\Phi} = \boldsymbol{I}_d$. Finally, conjugating (D.26) shows that the condition is indeed equivalent to $\boldsymbol{J}_1 \cdot \boldsymbol{A} \cdot \boldsymbol{\Phi} = \boldsymbol{J}_2 \cdot \boldsymbol{A}$, which was assumed for the theorem. This concludes the proof. □

D.6. Proof of Theorem 11.6.1

To show (11.38) we first define $\tilde{\boldsymbol{\mathcal{A}}} = \boldsymbol{\mathcal{A}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} (\boldsymbol{\Psi}^* \cdot \boldsymbol{\Psi}^*)$. By virtue of this definition, $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$ can be written as $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} = \left[\boldsymbol{\mathcal{A}} \sqcup_r \tilde{\boldsymbol{\mathcal{A}}}\right]$ (cf. (11.37)). Therefore, (11.38) becomes

$$\left[\boldsymbol{\mathcal{A}} \sqcup_r \tilde{\boldsymbol{\mathcal{A}}}\right] \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)} = \left[\boldsymbol{\mathcal{A}} \sqcup_r \tilde{\boldsymbol{\mathcal{A}}}\right] \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)}. \tag{D.27}$$

Applying (4.8) and (4.9) we can rewrite (D.27) as

$$\begin{aligned}
&\left[\left(\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)}\right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_2^{(r)} \boldsymbol{\Pi}_{M_r})\right)\right] \times_{R+1} \boldsymbol{\Phi}^{(r)} \\
=&\left[\left(\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)}\right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_1^{(r)} \boldsymbol{\Pi}_{M_r})\right)\right] \\
&\left[\left(\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)}\right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_2^{(r)} \boldsymbol{\Pi}_{M_r}) \times_{R+1} \boldsymbol{\Phi}^{(r)}\right)\right] \\
=&\left[\left(\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)}\right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_1^{(r)} \boldsymbol{\Pi}_{M_r})\right)\right]
\end{aligned} \tag{D.28}$$

where we have inserted the definition of $\boldsymbol{J}_1^{(\mathrm{nc})(r)}$ and $\boldsymbol{J}_2^{(\mathrm{nc})(r)}$ from (11.29) and (11.30). Obviously, the parts on the left-hand side of the r-mode concatenation operator are equal, since $\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)}$. What remains to be shown is that

$$\tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_2^{(r)} \boldsymbol{\Pi}_{M_r}) \times_{R+1} \boldsymbol{\Phi}^{(r)} = \tilde{\boldsymbol{\mathcal{A}}} \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_1^{(r)} \boldsymbol{\Pi}_{M_r}). \tag{D.29}$$

Inserting the definition of $\tilde{\boldsymbol{\mathcal{A}}}$, (D.29) becomes

$$\begin{aligned}
&\boldsymbol{\mathcal{A}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_2^{(r)}) \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Phi}^{(r)} \\
=&\boldsymbol{\mathcal{A}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_r (\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}} \boldsymbol{J}_1^{(r)}) \ldots \times_R \boldsymbol{\Pi}_{M_R}.
\end{aligned} \tag{D.30}$$

Conjugating (D.30) and multiplying with $\boldsymbol{\Pi}_{M_q}$ along all modes $q = 1, 2, \ldots, R$, $q \neq r$ and with $\boldsymbol{\Pi}_{M_r^{(\mathrm{sel})}}$ along mode r we have

$$\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)^*} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)}. \tag{D.31}$$

Finally, inserting $\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)}$ into (D.31) we are left with $\boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} = \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)}$ which is obviously a true statement. This concludes the proof of the theorem. □

D.7. Proof of Theorem 11.6.2

Let us consider the noise-free case $\boldsymbol{\mathcal{N}} = \boldsymbol{\mathcal{O}}$. In this case, we have

$$\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_{R+1} \boldsymbol{S}^{\mathrm{T}} \tag{D.32}$$

from (11.36). Moreover, since the HOSVD of $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$ is given by $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = \boldsymbol{\mathcal{S}}^{[\mathrm{s}](r)} \times_1 \boldsymbol{U}_1^{[\mathrm{s}](r)} \ldots \times_R \boldsymbol{U}_R^{[\mathrm{s}](r)} \times_{R+1} \boldsymbol{U}_{R+1}^{[\mathrm{s}](r)}$, we can compute $\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)}$ via

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} = \boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} \times_{R+1} \left(\boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)^{-1}} \cdot \left(\boldsymbol{U}_{R+1}^{[\mathrm{s}](r)} \right)^{\mathrm{H}} \right) \tag{D.33}$$

(cf. (11.39)). Combining (D.32) and (D.33) we obtain

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_{R+1} \underbrace{\left(\boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)^{-1}} \cdot \left(\boldsymbol{U}_{R+1}^{[\mathrm{s}](r)} \right)^{\mathrm{H}} \cdot \boldsymbol{S}^{\mathrm{T}} \right)}_{\boldsymbol{T}_r} \tag{D.34}$$

To proceed, we require the following lemma:

Lemma D.7.1. *The r-mode augmented measurement tensors $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$ satisfy*

$$\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} \right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} \right]_{(R+1)}^{\mathrm{H}} = 2 \,\mathrm{Re} \left\{ [\boldsymbol{\mathcal{X}}]_{(R+1)} \cdot [\boldsymbol{\mathcal{X}}]_{(R+1)}^{\mathrm{H}} \right\}. \tag{D.35}$$

Proof. The $R+1$-mode unfolding of $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$ contains all $(R+1)$-mode vectors of $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$. Since $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$ is the r-mode concatenation of $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R}$, it is evident that its $(R+1)$-mode vectors contain the $(R+1)$-mode vectors of $\boldsymbol{\mathcal{X}}$ and the $(R+1)$-mode vectors of $\boldsymbol{\mathcal{X}}^*$, in some order. Hence, we can write

$$\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} \right]_{(R+1)} = \left[[\boldsymbol{\mathcal{X}}]_{(R+1)} \quad [\boldsymbol{\mathcal{X}}]_{(R+1)}^* \right] \cdot \boldsymbol{P}_r, \tag{D.36}$$

where $\boldsymbol{P}_r$ is a permutation matrix. Since for any permutation matrix we have $\boldsymbol{P}_r^{\mathrm{H}} = \boldsymbol{P}_r^{-1}$, we can write $\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}^{\mathrm{H}} = \left[\boldsymbol{\mathcal{X}}\right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}\right]_{(R+1)}^{\mathrm{H}} + \left[\boldsymbol{\mathcal{X}}\right]_{(R+1)}^{*} \cdot \left[\boldsymbol{\mathcal{X}}\right]_{(R+1)}^{\mathrm{T}} = 2\,\mathrm{Re}\left\{\left[\boldsymbol{\mathcal{X}}\right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}\right]_{(R+1)}^{\mathrm{H}}\right\}$, which proves the lemma. $\square$

From the lemma we conclude that $\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}^{\mathrm{H}}$ is not a function of r. This is significant as we need the matrices $\boldsymbol{\Psi}^{(r)}$ to have common eigenvectors for the pairing in R-D ESPRIT and, therefore, $\boldsymbol{T}_r$ in (D.34) should be independent of r. To show this we observe that $\boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)}$ and $\boldsymbol{U}_{R+1}^{[\mathrm{s}](r)}$ represent the set of right singular vectors and the set of singular values of $\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}$ and hence they can alternatively be computed via the eigenvalue decomposition of $\left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)} \cdot \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}\right]_{(R+1)}^{\mathrm{H}}$. Consequently, $\boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)}$ and $\boldsymbol{U}_{R+1}^{[\mathrm{s}](r)}$ must be equal for all $r = 1, 2, \ldots, R$ due to (D.35). Remembering that $\boldsymbol{T}_r$ in (D.34) is defined as $\boldsymbol{T}_r = \boldsymbol{\Sigma}_{R+1}^{[\mathrm{s}](r)^{-1}} \cdot \left(\boldsymbol{U}_{R+1}^{[\mathrm{s}](r)}\right)^{\mathrm{H}} \cdot \boldsymbol{S}^{\mathrm{T}}$ we immediately find that $\boldsymbol{T}_r$ is also equal for all $r = 1, 2, \ldots, R$. Therefore, we have $\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_{R+1} \boldsymbol{T}$. Inserting this relation into the shift invariance equations (11.38) we obtain

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \boldsymbol{T}^{-1} \times_{R+1} \boldsymbol{\Phi}^{(r)} = \boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)} \times_{R+1} \boldsymbol{T}^{-1} \tag{D.37}$$

$$\boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_1^{(\mathrm{nc})(r)} \times_{R+1} \left(\boldsymbol{T} \cdot \boldsymbol{\Phi}^{(r)} \cdot \boldsymbol{T}^{-1}\right) = \boldsymbol{\mathcal{U}}^{[\mathrm{s}](r)} \times_r \boldsymbol{J}_2^{(\mathrm{nc})(r)}, \tag{D.38}$$

which proves the theorem. $\square$

Note that when computing the truncated HOSVD of $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}$, special attention must be paid with respect to the n-ranks. While for R-D Tensor-ESPRIT-type algorithms the n-ranks of $\boldsymbol{\mathcal{X}}$ are always less than or equal to d, for R-D NC Tensor-ESPRIT the n-ranks can exceed d. This is easy to see from the fact that the noise-free observation tensor factors into $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = \boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} \times_{R+1} \boldsymbol{S}^{\mathrm{T}}$ and hence its n-ranks are determined by the n-ranks of $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$. Consider the r-mode unfolding of $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}$. Using the definition of $\tilde{\boldsymbol{\mathcal{A}}}$ as in Appendix D.6 such that $\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)} = \left[\boldsymbol{\mathcal{A}} \sqcup_r \tilde{\boldsymbol{\mathcal{A}}}\right]$ we can express the r-mode unfolding via (4.4) as

$$\left[\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}\right]_{(r)} = \begin{bmatrix} \left[\boldsymbol{\mathcal{A}}\right]_{(r)} \\ \left[\tilde{\boldsymbol{\mathcal{A}}}\right]_{(r)} \end{bmatrix}. \tag{D.39}$$

Applying property (4.25) to the definition of $\boldsymbol{\mathcal{A}}$ given in (9.15) we obtain for $\left[\boldsymbol{\mathcal{A}}\right]_{(r)}$

$$\left[\boldsymbol{\mathcal{A}}\right]_{(r)} = \boldsymbol{A}^{(r)} \cdot \underbrace{\left(\boldsymbol{A}^{(r+1)} \diamond \ldots \diamond \boldsymbol{A}^{(R)} \diamond \boldsymbol{I}_d \diamond \boldsymbol{A}^{(1)} \diamond \ldots \diamond \boldsymbol{A}^{(r-1)}\right)^{\mathrm{T}}}_{\boldsymbol{B}^{(r)^{\mathrm{T}}}}. \tag{D.40}$$

Likewise, for $[\tilde{\boldsymbol{\mathcal{A}}}]_{(r)}$ we can show using similar reasoning

$$[\tilde{\boldsymbol{\mathcal{A}}}]_{(r)} = \boldsymbol{\Pi}_{M_r} \boldsymbol{A}^{(r)^*} \cdot \underbrace{\left((\boldsymbol{\Pi}_{M_{r+1}} \boldsymbol{A}^{(r+1)^*}) \diamond \ldots \diamond (\boldsymbol{\Pi}_{M_R} \boldsymbol{A}^{(R)^*}) \diamond \boldsymbol{\Psi}^{*^2} \diamond (\boldsymbol{\Pi}_{M_1} \boldsymbol{A}^{(1)^*}) \diamond \ldots \diamond (\boldsymbol{\Pi}_{M_{r-1}} \boldsymbol{A}^{(r-1)^*}) \right)^{\mathrm{T}}}_{\tilde{\boldsymbol{B}}^{(r)^{\mathrm{T}}}}. \tag{D.41}$$

Therefore, $[\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}]_{(r)}$ becomes

$$\left[\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}\right]_{(r)} = \begin{bmatrix} \boldsymbol{A}^{(r)} \cdot \boldsymbol{B}^{(r)^{\mathrm{T}}} \\ \boldsymbol{\Pi}_{M_r} \cdot \boldsymbol{A}^{(r)^*} \cdot \tilde{\boldsymbol{B}}^{(r)^{\mathrm{T}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}^{(r)} & \boldsymbol{0}_{M_r \times d} \\ \boldsymbol{0}_{M_r \times d} & \boldsymbol{\Pi}_{M_r} \cdot \boldsymbol{A}^{(r)^*} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{B}^{(r)} & \tilde{\boldsymbol{B}}^{(r)} \end{bmatrix}^{\mathrm{T}}. \tag{D.42}$$

Equation (D.42) shows that $[\boldsymbol{\mathcal{A}}^{(\mathrm{nc},r)}]_{(r)}$ can be expressed as the product of a matrix of size $2M_r \times 2d$ which has rank $2d$ if $\boldsymbol{A}^{(r)}$ is full column rank and the transpose of a matrix of size $M \cdot d/M_r \times 2d$. Note that the second matrix has a rank higher than d if $\boldsymbol{B}^{(r)}$ and $\tilde{\boldsymbol{B}}^{(r)}$ do not have the same column space. This is the case if the array is not centro-symmetric in some mode $q = 1, 2, \ldots, R$, $q \neq r$. Consequently, if the array is not fully centro-symmetric, the n-ranks of the augmented array steering tensor can exceed d (for $n = 1, 2, \ldots, R$), in the worst case they can reach $2d$. This has to be taken into account when computing the truncated HOSVD for the HOSVD-based signal subspace estimate, e.g., by truncating only up to $2d$ in the first R modes to be on the safe side.

D.8. Proof of Theorem 11.6.3

Let $\boldsymbol{\mathcal{Z}}^{(\mathrm{nc},r)}$ be the forward-backward averaged r-mode augmented measurement tensor, i.e., $\boldsymbol{\mathcal{Z}}^{(\mathrm{nc},r)} = \left[\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} \sqcup_{R+1} (\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)^*} \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R} \times_{R+1} \boldsymbol{\Pi}_N)\right]$. Since $\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)} = [\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}}]$ where $\tilde{\boldsymbol{\mathcal{X}}} = \boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R}$ we can expand $\boldsymbol{\mathcal{Z}}^{(\mathrm{nc},r)}$ into

$$\boldsymbol{\mathcal{Z}}^{(\mathrm{nc},r)} = \left[[\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}}] \sqcup_{R+1} [(\boldsymbol{\mathcal{X}} \times_{R+1} \boldsymbol{\Pi}_N) \sqcup_r (\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \boldsymbol{\Pi}_N)] \right] \tag{D.43}$$

using (4.8), (4.9), and (4.10). In the next step, let us compute the $(R+1)$-mode product between $\boldsymbol{\mathcal{Z}}^{(\mathrm{nc},r)}$ and $\boldsymbol{Q}_{2N}^{\mathrm{H}}$. Using the sparse left-$\boldsymbol{\Pi}$-real matrices $\boldsymbol{Q}_p^{(\mathrm{s})}$ introduced in [HN95] (cf. Appendix A.2), we obtain

$$
\begin{aligned}
&\mathcal{Z}^{(\mathrm{nc},r)} \times_{R+1} \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_N & \boldsymbol{\Pi}_N \\ -\jmath \boldsymbol{I}_N & \jmath \boldsymbol{\Pi}_N \end{bmatrix} \\
=& \left[\left[\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}} \right] \sqcup_{R+1} \left[(\boldsymbol{\mathcal{X}} \times_{R+1} \boldsymbol{\Pi}_N) \sqcup_r (\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \boldsymbol{\Pi}_N) \right] \right] \times_{R+1} \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_N & \boldsymbol{\Pi}_N \\ -\jmath \boldsymbol{I}_N & \jmath \boldsymbol{\Pi}_N \end{bmatrix} \\
=& \frac{1}{\sqrt{2}} \cdot \left[\left(\boldsymbol{\mathcal{X}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ -\jmath \boldsymbol{I}_N \end{bmatrix} \right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ -\jmath \boldsymbol{I}_N \end{bmatrix} \right) \right] \\
+& \frac{1}{\sqrt{2}} \cdot \left[\left(\boldsymbol{\mathcal{X}} \times_{R+1} \left(\begin{bmatrix} \boldsymbol{\Pi}_N \\ \jmath \boldsymbol{\Pi}_N \end{bmatrix} \cdot \boldsymbol{\Pi}_N \right) \right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \left(\begin{bmatrix} \boldsymbol{\Pi}_N \\ \jmath \boldsymbol{\Pi}_N \end{bmatrix} \cdot \boldsymbol{\Pi}_N \right) \right) \right] \\
=& \frac{1}{\sqrt{2}} \cdot \left[\left(\boldsymbol{\mathcal{X}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ -\jmath \boldsymbol{I}_N \end{bmatrix} \right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ -\jmath \boldsymbol{I}_N \end{bmatrix} \right) \right] + \frac{1}{\sqrt{2}} \cdot \left[\left(\boldsymbol{\mathcal{X}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ \jmath \boldsymbol{I}_N \end{bmatrix} \right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \begin{bmatrix} \boldsymbol{I}_N \\ \jmath \boldsymbol{I}_N \end{bmatrix} \right) \right] \\
=& \frac{1}{\sqrt{2}} \cdot \left[\left(\boldsymbol{\mathcal{X}} \times_{R+1} \begin{bmatrix} 2\boldsymbol{I}_N \\ \boldsymbol{0}_{N \times N} \end{bmatrix} \right) \sqcup_r \left(\tilde{\boldsymbol{\mathcal{X}}} \times_{R+1} \begin{bmatrix} 2\boldsymbol{I}_N \\ \boldsymbol{0}_{N \times N} \end{bmatrix} \right) \right] \\
=& \frac{2}{\sqrt{2}} \cdot \left[\left[\boldsymbol{\mathcal{X}} \sqcup_{R+1} \boldsymbol{\mathcal{O}}_{M_1 \times \ldots \times M_R \times N} \right] \sqcup_r \left[\tilde{\boldsymbol{\mathcal{X}}} \sqcup_{R+1} \boldsymbol{\mathcal{O}}_{M_1 \times \ldots \times M_R \times N} \right] \right]
\end{aligned} \tag{D.44}
$$

where we have again used (4.8), (4.9), and (4.10). and the obvious fact that $[\boldsymbol{\mathcal{A}} \sqcup_r \boldsymbol{\mathcal{B}}] + [\boldsymbol{\mathcal{C}} \sqcup_r \boldsymbol{\mathcal{D}}] = [(\boldsymbol{\mathcal{A}} + \boldsymbol{\mathcal{C}}) \sqcup_r (\boldsymbol{\mathcal{B}} + \boldsymbol{\mathcal{D}})]$. Equation (D.44) shows that the last N "virtual snapshots" (i.e., $(R+1)$-mode slices) become zero. Therefore, we can skip the zero blocks and continue the proof only for the non-zero part $\sqrt{2} \cdot \left[\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}} \right] \in \mathbb{C}^{M_1 \times \ldots \times 2M_r \times \ldots \times M_R \times N}$.

To proceed further, we need the following lemma:

Lemma D.8.1. *Let* $\bar{\boldsymbol{\mathcal{X}}}^{(r)} = \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}}$. *Then, the following identity is true* $\tilde{\boldsymbol{\mathcal{X}}} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} = \bar{\boldsymbol{\mathcal{X}}}^{(r)^*} \times_r \boldsymbol{\Pi}_{M_r}$.

Proof. The identity follows by expanding $\tilde{\boldsymbol{\mathcal{X}}}$ into $\tilde{\boldsymbol{\mathcal{X}}} = \boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{\Pi}_{M_1} \ldots \times_R \boldsymbol{\Pi}_{M_R}$. Using the rules of n-mode products we can then simplify $\tilde{\boldsymbol{\mathcal{X}}} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}}$ into

$$
\begin{aligned}
&\tilde{\boldsymbol{\mathcal{X}}} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} \\
=& \boldsymbol{\mathcal{X}}^* \times_1 (\boldsymbol{Q}_{M_1}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_{M_1}) \ldots \times_{r-1} (\boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_{M_{r-1}}) \times_r \boldsymbol{\Pi}_{M_r} \times_{r+1} (\boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_{M_{r+1}}) \ldots \times_R (\boldsymbol{Q}_{M_R}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_{M_R}) \\
=& \boldsymbol{\mathcal{X}}^* \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{T}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{T}} \times_r \boldsymbol{\Pi}_{M_r} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{T}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{T}} \\
=& \left(\boldsymbol{\mathcal{X}} \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_r \boldsymbol{\Pi}_{M_r} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} \right)^* \\
=& \bar{\boldsymbol{\mathcal{X}}}^{(r)^*} \times_r \boldsymbol{\Pi}_{M_r},
\end{aligned} \tag{D.45}
$$

where we have used the fact that $\boldsymbol{Q}_p^{\mathrm{H}} \cdot \boldsymbol{\Pi}_p = \boldsymbol{Q}_p^{\mathrm{T}}$ since $\boldsymbol{Q}_p$ is left-$\boldsymbol{\Pi}$-real. This concludes the proof of the lemma. $\square$

Via Lemma D.8.1 it is easy to see that the multiplication of the remaining part $\sqrt{2}\cdot\left[\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}}\right]$ with $\boldsymbol{Q}_{M_q}^{\mathrm{H}}$ in all modes $q = 1, 2, \ldots, R$, $q \neq r$ yields

$$\sqrt{2}\cdot\left[\boldsymbol{\mathcal{X}} \sqcup_r \tilde{\boldsymbol{\mathcal{X}}}\right] \times_1 \boldsymbol{Q}_{M_1}^{\mathrm{H}} \ldots \times_{r-1} \boldsymbol{Q}_{M_{r-1}}^{\mathrm{H}} \times_{r+1} \boldsymbol{Q}_{M_{r+1}}^{\mathrm{H}} \ldots \times_R \boldsymbol{Q}_{M_R}^{\mathrm{H}} = \sqrt{2}\cdot\left[\bar{\boldsymbol{\mathcal{X}}} \sqcup_r \bar{\boldsymbol{\mathcal{X}}}^* \times_r \boldsymbol{\Pi}_{M_r}\right]. \tag{D.46}$$

The final step for the computation of $\mathcal{T}(\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)})$ is the multiplication with $\boldsymbol{Q}_{2M_r}^{\mathrm{H}}$ in mode r. Skipping the zero blocks for brevity, we obtain

$$\begin{aligned}
\mathcal{T}(\boldsymbol{\mathcal{X}}^{(\mathrm{nc},r)}) &= \sqrt{2}\cdot\left[\bar{\boldsymbol{\mathcal{X}}} \sqcup_r \bar{\boldsymbol{\mathcal{X}}}^* \times_r \boldsymbol{\Pi}_{M_r}\right] \times_r \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{I}_{M_r} & \boldsymbol{\Pi}_{M_r} \\ -\jmath \boldsymbol{I}_{M_r} & \jmath \boldsymbol{\Pi}_{M_r} \end{bmatrix} \\
&= \left(\bar{\boldsymbol{\mathcal{X}}} \times_r \begin{bmatrix} \boldsymbol{I}_{M_r} \\ -\jmath \boldsymbol{I}_{M_r} \end{bmatrix}\right) + \left(\bar{\boldsymbol{\mathcal{X}}}^* \times_r \left(\begin{bmatrix} \boldsymbol{\Pi}_{M_r} \\ \jmath \boldsymbol{\Pi}_{M_r} \end{bmatrix} \cdot \boldsymbol{\Pi}_{M_r}\right)\right) \\
&= \left(\bar{\boldsymbol{\mathcal{X}}} \times_r \begin{bmatrix} \boldsymbol{I}_{M_r} \\ -\jmath \boldsymbol{I}_{M_r} \end{bmatrix}\right) + \left(\bar{\boldsymbol{\mathcal{X}}}^* \times_r \begin{bmatrix} \boldsymbol{I}_{M_r} \\ \jmath \boldsymbol{I}_{M_r} \end{bmatrix}\right) \\
&= \left[(\bar{\boldsymbol{\mathcal{X}}} + \bar{\boldsymbol{\mathcal{X}}}^*) \sqcup_r (-\jmath \bar{\boldsymbol{\mathcal{X}}} + \jmath \bar{\boldsymbol{\mathcal{X}}}^*)\right] \\
&= 2\cdot\left[\mathrm{Re}\left\{\bar{\boldsymbol{\mathcal{X}}}\right\} \sqcup_r \mathrm{Im}\left\{\bar{\boldsymbol{\mathcal{X}}}\right\}\right]
\end{aligned} \tag{D.47}$$

which proves the theorem. $\square$

D.9. Derivation of the TS-SLS update rule

In this section we sketch the derivation of the update rule for the TS-SLS algorithm. Note that it was first proposed in [RH07b], however, due to space limitations the derivation could not be included there. We start with the cost function (11.47). For simplicity we only consider $r = 1$. The first step is to expand the residual tensor $\boldsymbol{\mathcal{R}}_{k+1}^{(1)}$ in terms of the updates of the k-th iteration $\Delta\Delta\boldsymbol{\mathcal{S}}_k$, $\Delta\Delta\boldsymbol{U}_{1,k}$, $\Delta\Delta\boldsymbol{U}_{2,k}$, and $\Delta\Delta\boldsymbol{\Psi}_k^{(1)}$. The expansion contains one "zero-th order" term (i.e., not dependent on any of the update terms) equal to $\boldsymbol{\mathcal{R}}_k^{(1)}$, linear terms for $\Delta\Delta\boldsymbol{\mathcal{S}}_k$, $\Delta\Delta\boldsymbol{U}_{1,k}$, $\Delta\Delta\boldsymbol{U}_{2,k}$, and $\Delta\Delta\boldsymbol{\Psi}_k^{(1)}$, and many higher-order terms which are the result of cross products between the update terms. If we assume the updates to be small, all higher-order terms can be neglected and we only consider the linear terms. We are left with

$$\boldsymbol{\mathcal{R}}_{k+1}^{(1)} \approx \boldsymbol{\mathcal{R}}_k^{(1)} + \Delta\Delta\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \Delta\Delta\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}$$

$$+\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}$$
$$+\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \Delta\Delta\boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \Delta\Delta\boldsymbol{U}_{2,k}$$
$$+\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \Delta\Delta\boldsymbol{\Psi}_k^{(1)}, \tag{D.48}$$

where $\boldsymbol{U}_{1,k} = \hat{\boldsymbol{U}}_1 + \Delta\boldsymbol{U}_{1,k}$, $\boldsymbol{U}_{2,k} = \hat{\boldsymbol{U}}_2 + \Delta\boldsymbol{U}_{2,k}$, $\boldsymbol{\mathcal{S}}_k = \hat{\boldsymbol{\mathcal{S}}} + \Delta\boldsymbol{\mathcal{S}}_k$, and $\boldsymbol{\Psi}_k^{(1)} = \hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(1)} + \Delta\boldsymbol{\Psi}_k^{(1)}$.

In the next step we apply the vec-operator to transform the linear dependencies from (D.48) into a canonical matrix-vector form. For the first two terms connected to $\Delta\Delta\boldsymbol{\mathcal{S}}_k$ we apply property (4.20) and obtain

$$\begin{aligned}&\operatorname{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \Delta\Delta\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}\right\}\\ =&\underbrace{\left(\boldsymbol{\Psi}_k^{(1)} \otimes \boldsymbol{U}_{2,k} \otimes (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) - \boldsymbol{I}_d \otimes \boldsymbol{U}_{2,k} \otimes (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k})\right)}_{\boldsymbol{F}_4^{(1)}} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k\right\}.\end{aligned} \tag{D.49}$$

For the second set of terms we first compute a one-mode unfolding and then reorder its vectorized version via the definition of the permutation matrices from (4.11). We can write

$$\begin{aligned}&\operatorname{vec}\left\{\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}\right\}\\ =&\boldsymbol{P}^{(1)} \cdot \operatorname{vec}\left\{\boldsymbol{J}_1^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k} \cdot \left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)}\right]_{(1)} - \boldsymbol{J}_2^{(1)} \cdot \Delta\Delta\boldsymbol{U}_{1,k} \cdot \left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k}\right]_{(1)}\right\}\\ =&\underbrace{\boldsymbol{P}^{(1)} \cdot \left(\left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)}\right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{J}_1^{(1)} - \left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k}\right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{J}_2^{(1)}\right)}_{\boldsymbol{F}_2^{(1)}} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\}\end{aligned} \tag{D.50}$$

where in the second step we have used property (3.7). The two terms connected to $\Delta\Delta\boldsymbol{U}_{2,k}$ are rewritten in a similar manner

$$\begin{aligned}&\operatorname{vec}\left\{\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \Delta\Delta\boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} - \boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \Delta\Delta\boldsymbol{U}_{2,k}\right\}\\ =&\boldsymbol{P}^{(2)} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k} \cdot \left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_3 \boldsymbol{\Psi}_k^{(1)}\right]_{(2)} - \Delta\Delta\boldsymbol{U}_{2,k} \cdot \left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k})\right]_{(2)}\right\}\\ =&\boldsymbol{P}^{(2)} \cdot \underbrace{\left(\left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_3 \boldsymbol{\Psi}_k^{(1)}\right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_2} - \left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_2^{(1)} \cdot \boldsymbol{U}_{1,k})\right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_2}\right)}_{\boldsymbol{F}_3^{(1)}} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\}\end{aligned} \tag{D.51}$$

Finally, the remaining term connected to $\Delta\Delta\boldsymbol{\Psi}_k^{(1)}$ becomes

$$\begin{aligned}
&\operatorname{vec}\left\{\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k} \times_3 \Delta\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\
=&\boldsymbol{P}^{(3)} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(1)} \cdot \left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}\right]_{(3)}\right\} \\
=&\boldsymbol{P}^{(3)} \cdot \underbrace{\left(\left[\boldsymbol{\mathcal{S}}_k \times_1 (\boldsymbol{J}_1^{(1)} \cdot \boldsymbol{U}_{1,k}) \times_2 \boldsymbol{U}_{2,k}\right]_{(3)}^{\mathrm{T}} \otimes \boldsymbol{I}_d\right)}_{\boldsymbol{F}_1^{(1)}} \cdot \operatorname{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(1)}\right\}. \qquad \text{(D.52)}
\end{aligned}$$

Therefore, collecting (D.49), (D.50), (D.51), and (D.52), we have the following first-order expansion for the residual tensor

$$\operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_{k+1}^{(1)}\right\} \approx \operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_{k}^{(1)}\right\} + \begin{bmatrix}\boldsymbol{F}_1^{(1)} & \boldsymbol{F}_2^{(1)} & \boldsymbol{F}_3^{(1)} & \boldsymbol{F}_4^{(1)}\end{bmatrix} \cdot \begin{bmatrix}\operatorname{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_{k}\right\}\end{bmatrix}. \qquad \text{(D.53)}$$

The next step is to realize that the cost function for TS-SLS is a sum of norms which can be written as the norm of a vector containing the vectorized versions of $\boldsymbol{\mathcal{R}}^{(1)}$, $\Delta\boldsymbol{U}_1$, $\Delta\boldsymbol{U}_2$, and $\Delta\boldsymbol{\mathcal{S}}$. Therefore, the cost function in the k-th iteration step can be written as

$$\begin{aligned}
&\begin{bmatrix}\operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_{k+1}^{(1)}\right\} \\ \kappa_1^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{U}_{1,k+1}\right\} \\ \kappa_2^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{U}_{2,k+1}\right\} \\ \kappa_3^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_{k+1}\right\}\end{bmatrix} \approx \begin{bmatrix}\operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_{k}^{(1)}\right\} \\ \kappa_1^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{U}_{1,k}\right\} \\ \kappa_2^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{U}_{2,k}\right\} \\ \kappa_3^{(1)} \cdot \operatorname{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_{k}\right\}\end{bmatrix} \\
&+ \underbrace{\begin{bmatrix}\boldsymbol{F}_1^{(1)} & \boldsymbol{F}_2^{(1)} & \boldsymbol{F}_3^{(1)} & \boldsymbol{F}_4^{(1)} \\ \mathbf{0} & \kappa_1^{(1)} \cdot \boldsymbol{I}_{M_1 \cdot p_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \kappa_2^{(1)} \cdot \boldsymbol{I}_{M_2 \cdot p_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \kappa_3^{(1)} \cdot \boldsymbol{I}_{p_1 \cdot p_2 \cdot d}\end{bmatrix}}_{\boldsymbol{F}} \cdot \begin{bmatrix}\operatorname{vec}\left\{\Delta\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_{k}\right\}\end{bmatrix}, \qquad \text{(D.54)}
\end{aligned}$$

where we have inserted (D.53) and the three update equations $\Delta\boldsymbol{U}_{1,k+1} = \Delta\boldsymbol{U}_{1,k} + \Delta\Delta\boldsymbol{U}_{1,k}$, $\Delta\boldsymbol{U}_{2,k+1} = \Delta\boldsymbol{U}_{2,k} + \Delta\Delta\boldsymbol{U}_{2,k}$, and $\Delta\boldsymbol{\mathcal{S}}_{k+1} = \Delta\boldsymbol{\mathcal{S}}_{k} + \Delta\Delta\boldsymbol{\mathcal{S}}_{k}$.

Consequently, after linearization the resulting optimization problem is equivalent to minimizing the norm of (D.54), which is a linear least squares problem that is readily solved via the pseudo-inverse of $\boldsymbol{F}$. This concludes the derivation for $r = 1$. For convenience, the update matrices $\boldsymbol{F}_n^{(1)}$, $n = 1, 2, 3, 4$ are summarized again below.

$$\boldsymbol{F}_1^{(1)} = \boldsymbol{P}^{(3)} \cdot \left(\left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \boldsymbol{J}_1^{(1)} \cdot \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \times_2 \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right]_{(3)}^{\mathrm{T}} \otimes \boldsymbol{I}_d \right)$$

$$\begin{aligned} \boldsymbol{F}_2^{(1)} = \boldsymbol{P}^{(1)} \cdot \Bigg(& \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_2 \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \times_3 \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(1)} + \Delta \boldsymbol{\Psi}_k^{(1)} \right) \right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{J}_1^{(1)} \\ & - \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_2 \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{J}_2^{(1)} \Bigg) \end{aligned}$$

$$\begin{aligned} \boldsymbol{F}_3^{(1)} = \boldsymbol{P}^{(2)} \cdot \Bigg(& \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \boldsymbol{J}_1^{(1)} \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \times_3 \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(1)} + \Delta \boldsymbol{\Psi}_k^{(1)} \right) \right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_2} \\ & - \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \boldsymbol{J}_2^{(1)} \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_2} \Bigg) \end{aligned}$$

$$\begin{aligned} \boldsymbol{F}_4^{(1)} = & \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(1)} + \Delta \boldsymbol{\Psi}_k^{(1)} \right) \otimes \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \otimes \left(\boldsymbol{J}_1^{(1)} \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \right) \\ & - \boldsymbol{I}_d \otimes \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \otimes \left(\boldsymbol{J}_2^{(1)} \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \right). \end{aligned}$$

The derivation for $r = 2$ proceeds in a similar manner. The resulting matrices $\boldsymbol{F}_n^{(2)}$ for $n = 1, 2, 3, 4$ are given by

$$\boldsymbol{F}_1^{(2)} = \boldsymbol{P}^{(3)} \cdot \left(\left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \times_2 \boldsymbol{J}_1^{(2)} \cdot \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right]_{(3)}^{\mathrm{T}} \otimes \boldsymbol{I}_d \right)$$

$$\begin{aligned} \boldsymbol{F}_2^{(2)} = \boldsymbol{P}^{(1)} \cdot \Bigg(& \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_2 \boldsymbol{J}_1^{(2)} \cdot \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \times_3 \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(2)} + \Delta \boldsymbol{\Psi}_k^{(2)} \right) \right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_1} \\ & - \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_2 \boldsymbol{J}_2^{(2)} \cdot \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right]_{(1)}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_1} \Bigg) \end{aligned}$$

$$\begin{aligned} \boldsymbol{F}_3^{(2)} = \boldsymbol{P}^{(2)} \cdot \Bigg(& \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \times_3 \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(2)} + \Delta \boldsymbol{\Psi}_k^{(2)} \right) \right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{J}_1^{(2)} \\ & - \left[\left(\hat{\boldsymbol{\mathcal{S}}} + \Delta \boldsymbol{\mathcal{S}}_k \right) \times_1 \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \right]_{(2)}^{\mathrm{T}} \otimes \boldsymbol{J}_2^{(2)} \Bigg) \end{aligned}$$

$$\begin{aligned} \boldsymbol{F}_4^{(2)} = & \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{(2)} + \Delta \boldsymbol{\Psi}_k^{(2)} \right) \otimes \left(\boldsymbol{J}_1^{(2)} \cdot \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right) \otimes \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right) \\ & - \boldsymbol{I}_d \otimes \left(\boldsymbol{J}_2^{(2)} \cdot \left(\hat{\boldsymbol{U}}_2 + \Delta \boldsymbol{U}_{2,k} \right) \right) \otimes \left(\hat{\boldsymbol{U}}_1 + \Delta \boldsymbol{U}_{1,k} \right). \end{aligned}$$

D.10. Proof of Proposition 12.3.1

We start by inserting $\hat{\boldsymbol{U}}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s}}$ and $\hat{\boldsymbol{T}}_r = \boldsymbol{T}_r + \Delta \boldsymbol{T}_r$ into (10.11). Then we obtain

$$\left[\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]} \right]_{(R+1)}^{\mathrm{T}} = \left[(\boldsymbol{T}_1 + \Delta \boldsymbol{T}_1) \otimes (\boldsymbol{T}_2 + \Delta \boldsymbol{T}_2) \otimes \ldots (\boldsymbol{T}_R + \Delta \boldsymbol{T}_R) \right] \cdot (\boldsymbol{U}_{\mathrm{s}} + \Delta \boldsymbol{U}_{\mathrm{s}})$$

$$= \underbrace{[\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}}}_{\boldsymbol{U}_{\mathrm{s}}} + [\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \dots \boldsymbol{T}_R] \cdot \Delta \boldsymbol{U}_{\mathrm{s}}$$
$$+ [\Delta \boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}} + \dots + [\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 \otimes \dots \Delta \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\}, \quad \text{(D.55)}$$

since all terms that contain more than one perturbation term can be absorbed into $\mathcal{O}\left\{\Delta^2\right\}$. The first term in (D.55) represents the exact signal subspace (cf. Corollary D.3.2), hence the remaining terms are the first order expansion of $\left[\Delta \hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]^{\mathrm{T}}_{(R+1)}$. As the first term of this expansion already agrees with Proposition 12.3.1, we still need to show that for the remaining terms we have for $r = 1, 2, \dots, R$

$$[\boldsymbol{T}_1 \otimes \dots \otimes \Delta \boldsymbol{T}_r \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}} = \left[\boldsymbol{T}_1 \otimes \dots \otimes (\boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}}) \otimes \dots \boldsymbol{T}_R\right] \cdot \boldsymbol{U}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\}. \quad \text{(D.56)}$$

As a first step, we expand the left-hand side of (D.56) by applying Corollary D.3.2

$$\begin{aligned}
[\boldsymbol{T}_1 \otimes \dots \otimes \Delta \boldsymbol{T}_r \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}} &= [\boldsymbol{T}_1 \otimes \dots \otimes \Delta \boldsymbol{T}_r \otimes \dots \boldsymbol{T}_R] \cdot [\boldsymbol{T}_1 \otimes \dots \otimes \boldsymbol{T}_r \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}} \\
&= [(\boldsymbol{T}_1 \cdot \boldsymbol{T}_1) \otimes \dots \otimes (\Delta \boldsymbol{T}_r \cdot \boldsymbol{T}_r) \otimes \dots (\boldsymbol{T}_R \cdot \boldsymbol{T}_R)] \cdot \boldsymbol{U}_{\mathrm{s}} \\
&= [\boldsymbol{T}_1 \otimes \dots \otimes (\Delta \boldsymbol{T}_r \cdot \boldsymbol{T}_r) \otimes \dots \boldsymbol{T}_R] \cdot \boldsymbol{U}_{\mathrm{s}}, \qquad \text{(D.57)}
\end{aligned}$$

where we have used the fact that the matrices $\boldsymbol{T}_r$ are projection matrices and hence idempotent, i.e., $\boldsymbol{T}_r \cdot \boldsymbol{T}_r = \boldsymbol{T}_r$. Hence, what remains to be shown is that $\Delta \boldsymbol{T}_r \cdot \boldsymbol{T}_r = \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \mathcal{O}\left\{\Delta^2\right\}$. Since $\hat{\boldsymbol{T}}_r = \hat{\boldsymbol{U}}_r^{[\mathrm{s}]} \cdot \hat{\boldsymbol{U}}_r^{[\mathrm{s}]^{\mathrm{H}}}$ and $\hat{\boldsymbol{U}}_r^{[\mathrm{s}]} = \boldsymbol{U}_r^{[\mathrm{s}]} + \Delta \boldsymbol{U}_r^{[\mathrm{s}]}$, a first order expansion for $\Delta \boldsymbol{T}_r$ is obtained via

$$\begin{aligned}
\hat{\boldsymbol{T}}_r &= \left(\boldsymbol{U}_r^{[\mathrm{s}]} + \Delta \boldsymbol{U}_r^{[\mathrm{s}]}\right) \cdot \left(\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \Delta \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}}\right) \\
&= \boldsymbol{T}_r + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \Delta \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \Delta \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \mathcal{O}\left\{\Delta^2\right\} \\
\Rightarrow \Delta \boldsymbol{T}_r &= \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \Delta \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \Delta \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \mathcal{O}\left\{\Delta^2\right\}, \qquad \text{(D.58)}
\end{aligned}$$

where in general we have $\Delta \boldsymbol{U}_r^{[\mathrm{s}]} = \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]} + \mathcal{O}\left\{\Delta^2\right\}$ (cf. (12.10)). Using this expansion in (D.58) we obtain

$$\begin{aligned}
\Delta \boldsymbol{T}_r \cdot \boldsymbol{T}_r &= \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \left(\boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]}\right)^{\mathrm{H}} \cdot \boldsymbol{T}_r + \left(\boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]}\right) \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r + \mathcal{O}\left\{\Delta^2\right\} \\
&= \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot \boldsymbol{U}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r \\
&\quad + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{s}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r + \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r + \mathcal{O}\left\{\Delta^2\right\}
\end{aligned}$$

$$= \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot \underbrace{\boldsymbol{U}_r^{[\mathrm{n}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r}_{\mathbf{0}_{M_r - p_r \times M_r}} + \boldsymbol{U}_r^{[\mathrm{s}]} \cdot \underbrace{\left(\boldsymbol{\Gamma}_r^{[\mathrm{s}]} + \boldsymbol{\Gamma}_r^{[\mathrm{s}]^{\mathrm{H}}}\right)}_{\mathbf{0}_{p_r \times p_r}} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r + \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} \cdot \underbrace{\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot \boldsymbol{T}_r}_{\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}}} + \mathcal{O}\left\{\Delta^2\right\}$$
$$= \boldsymbol{U}_r^{[\mathrm{n}]} \cdot \boldsymbol{\Gamma}_r^{[\mathrm{n}]} \cdot \boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} + \mathcal{O}\left\{\Delta^2\right\}, \tag{D.59}$$

which is the desired result. Note that $\boldsymbol{\Gamma}_r^{[\mathrm{s}]} + \boldsymbol{\Gamma}_r^{[\mathrm{s}]^{\mathrm{H}}} = \mathbf{0}_{p_r \times p_r}$ follows from the fact that $\boldsymbol{\Gamma}_r^{[\mathrm{s}]}$ is a skew-Hermitian matrix. This property is apparent from its definition (12.12), which we restate here for convenience

$$\boldsymbol{\Gamma}_r^{[\mathrm{s}]} = \boldsymbol{D}_r \odot \underbrace{\left(\boldsymbol{U}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)} \cdot \boldsymbol{V}_r^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_r^{[\mathrm{s}]} + \boldsymbol{\Sigma}_r^{[\mathrm{s}]} \cdot \boldsymbol{V}_r^{[\mathrm{s}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)}^{\mathrm{H}} \cdot \boldsymbol{U}_r^{[\mathrm{s}]}\right)}_{\bar{\boldsymbol{\Gamma}}_r^{[\mathrm{s}]}}. \tag{D.60}$$

By its definition, $\bar{\boldsymbol{\Gamma}}_r^{[\mathrm{s}]} = \bar{\boldsymbol{\Gamma}}_r^{[\mathrm{s}]^{\mathrm{H}}}$. Moreover, since $[\boldsymbol{D}_r]_{(k,\ell)} = \frac{1}{\sigma_\ell^{(r)^2} - \sigma_k^{(r)^2}}$ for $k \neq \ell$ we have $[\boldsymbol{D}_r]_{(k,\ell)} = -[\boldsymbol{D}_r]_{(\ell,k)}$ and hence $\boldsymbol{D}_r = -\boldsymbol{D}_r^{\mathrm{H}}$. Finally, the product of a Hermitian and a skew-Hermitian matrix is skew-Hermitian again. □

D.11. Sketch of derivation of (12.20)

The first-order expansion for 1-D Standard ESPRIT shown in (12.20) is essentially based on three steps: (1) the perturbation of μ_k in terms of $\lambda_k = \mathrm{e}^{\jmath \mu_k}$; (2) the perturbation of λ_k in terms $\boldsymbol{\Psi}$; (3) the perturbation of $\boldsymbol{\Psi}$ in terms of the signal subspace estimation error $\Delta \boldsymbol{U}_{\mathrm{s}}$.

For the first step we compute a Taylor series expansion of $\lambda = \mathrm{e}^{\jmath\mu}$ at a particular point $\lambda_k = \mathrm{e}^{\jmath \mu_k}$. We obtain

$$\begin{aligned} \mathrm{e}^{\jmath(\mu_k + \Delta\mu_k)} &= \mathrm{e}^{\jmath\mu_k} + \jmath \Delta\mu_k \cdot \mathrm{e}^{\jmath\mu_k} + \mathcal{O}\left\{\Delta^2\right\} \\ &\approx \underbrace{\mathrm{e}^{\jmath\mu_k}}_{\lambda_k} \cdot (1 + \jmath\Delta\mu_k). \end{aligned} \tag{D.61}$$

Consequently, for a slightly perturbed eigenvalue $\lambda_k + \Delta\lambda_k$ we can write up to first order

$$\begin{aligned} \lambda_k + \Delta\lambda_k &\approx \lambda_k \cdot (1 + \jmath\Delta\mu_k) \\ \frac{\Delta\lambda_k}{\lambda_k} &\approx 0 + \jmath\Delta\mu_k \\ \Rightarrow \Delta\mu_k &\approx \mathrm{Im}\left\{\frac{\Delta\lambda_k}{\lambda_k}\right\}. \end{aligned} \tag{D.62}$$

For the second step [LLV93] borrows a result from [LT78] which provides a first-order expansion of the eigendecomposition. Let $\boldsymbol{\Psi} = \boldsymbol{Q} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{-1}$ be the eigendecomposition of $\boldsymbol{\Psi}$ in terms of the matrix of eigenvectors $\boldsymbol{Q}$ and the diagonal matrix of eigenvalues $\boldsymbol{\Lambda} = \mathrm{diag}\{[\lambda_1, \ldots, \lambda_d]\}$. Then in [LT78] it is shown that the eigendecomposition of a perturbed matrix $\boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}$ can be expressed as

$$\boldsymbol{\Psi} + \Delta\boldsymbol{\Psi} = \boldsymbol{Q} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{-1} + \boldsymbol{Q} \cdot \Delta\boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{-1} + \mathcal{O}\left\{\Delta^2\right\}, \tag{D.63}$$

i.e., the perturbation $\Delta\boldsymbol{\Psi}$ has a first-order effect only on the *eigenvalues*, the perturbation term for the *eigenvectors* is already a second-order term. Rearranging (D.63) we can express $\Delta\lambda_k$ via

$$\begin{aligned} \Delta\boldsymbol{\Lambda} &= \underbrace{\boldsymbol{Q}^{-1}}_{\boldsymbol{P}} \cdot \Delta\boldsymbol{\Psi} \cdot \boldsymbol{Q} + \mathcal{O}\left\{\Delta^2\right\} \\ \Rightarrow \Delta\lambda_k &= \boldsymbol{p}_k^{\mathrm{T}} \cdot \Delta\boldsymbol{\Psi} \cdot \boldsymbol{q}_k + \mathcal{O}\left\{\Delta^2\right\}, \end{aligned} \tag{D.64}$$

where in the last step we have used the fact that $\boldsymbol{\Lambda}$ is diagonal and we have introduced $\boldsymbol{q}_k$ and $\boldsymbol{p}_k^{\mathrm{T}}$ as the k-th column and the k-th row of $\boldsymbol{Q}$ and $\boldsymbol{P} = \boldsymbol{Q}^{-1}$, respectively.

For the third step we consider the shift invariance equations in terms of the estimated signal subspace $\hat{\boldsymbol{U}}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{s}} + \Delta\boldsymbol{U}_{\mathrm{s}}$, where $\Delta\boldsymbol{U}_{\mathrm{s}}$ represents the estimation error. We also write $\hat{\boldsymbol{\Psi}}_{\mathrm{LS}} = \boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}$ and try to find a first order expansion for $\Delta\boldsymbol{\Psi}$ by rearranging the shift invariance equations. We obtain [RH89a]

$$\begin{aligned} \boldsymbol{J}_1 \cdot (\boldsymbol{U}_{\mathrm{s}} + \Delta\boldsymbol{U}_{\mathrm{s}}) \cdot (\boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}) &\approx \boldsymbol{J}_2 \cdot (\boldsymbol{U}_{\mathrm{s}} + \Delta\boldsymbol{U}_{\mathrm{s}}) \\ \boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi} &\approx \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}} + \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \\ \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi} &\approx \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \\ \Rightarrow \Delta\boldsymbol{\Psi} &\approx (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ \cdot (\boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} - \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi}) \end{aligned} \tag{D.65}$$

where the $\approx$ sign represents equality up to first order term and hence all second order terms have been neglected. Also, in the third line we have canceled two terms by exploiting the fact that $\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} = \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}$.

Combining (D.62), (D.64), and (D.65), we obtain

$$\Delta\mu_k \approx \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ (\boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} - \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi}) \cdot \boldsymbol{q}_k / \lambda_k\right\} \tag{D.66}$$

Equation (D.66) can be simplified by multiplying out the last bracket. Since $\boldsymbol{\Psi} \cdot \boldsymbol{q}_k = \lambda_k \cdot \boldsymbol{q}_k$

we can write

$$\Delta\mu_k \approx \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ \left(\boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k/\lambda_k - \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k \cdot \lambda_k/\lambda_k\right)\right\}$$
$$\approx \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^+ \left(\boldsymbol{J}_2/\lambda_k - \boldsymbol{J}_1\right) \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k\right\}, \qquad \text{(D.67)}$$

which is the desired result. □

D.12. Proof of Theorem 12.4.1

For R-D Standard ESPRIT, the explicit first-order expansion of the estimation error for the k-th spatial frequency in the r-th mode is given by (12.20), which we restate here for convenience

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{p}_k^{(r)^{\mathrm{T}}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}\right)^+ \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)}/\lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right] \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k^{(r)}\right\} + \mathcal{O}\left\{\Delta^2\right\} \qquad \text{(D.68)}$$

Note that inside the $\mathrm{Im}\{.\}$ operator we have a linear form in $\Delta\boldsymbol{U}_{\mathrm{s}}$, which is itself a linear form in the perturbation $\boldsymbol{N}$. Therefore, we can write

$$\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} = \mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}} \cdot \mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} \qquad \text{(D.69)}$$

Here $\boldsymbol{r}_k^{(r)^{\mathrm{T}}} = \boldsymbol{q}_k^{(r)^{\mathrm{T}}} \otimes \left(\boldsymbol{p}_k^{(r)^{\mathrm{T}}} \cdot \left(\tilde{\boldsymbol{J}}_1^{(r)} \cdot \boldsymbol{U}_{\mathrm{s}}\right)^+ \cdot \left[\tilde{\boldsymbol{J}}_2^{(r)}/\lambda_k^{(r)} - \tilde{\boldsymbol{J}}_1^{(r)}\right]\right)$ follows directly by applying property (3.7) for matrix product linear forms.

Moreover, $\boldsymbol{W}_{\mathrm{mat}} = \left(\boldsymbol{\Sigma}_{\mathrm{s}}^{-1} \cdot \boldsymbol{V}_{\mathrm{s}}^{\mathrm{T}}\right) \otimes \left(\boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}}\right)$ follows by applying the same rule to the first order expansion of $\Delta\boldsymbol{U}_{\mathrm{s}} = \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}} \cdot \boldsymbol{N} \cdot \boldsymbol{V}_{\mathrm{s}} \cdot \boldsymbol{\Sigma}_{\mathrm{s}}^{-1}$ (cf. (12.4)).

In order to expand $\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\}$ using (D.69), we observe that for arbitrary complex vectors $\boldsymbol{z}_1, \boldsymbol{z}_2$ we have

$$\mathrm{Im}\left\{\boldsymbol{z}_1^{\mathrm{T}} \cdot \boldsymbol{z}_2\right\} = \mathrm{Im}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\} + \mathrm{Re}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\} \quad \text{and hence}$$
$$\mathrm{Im}\left\{\boldsymbol{z}_1^{\mathrm{T}} \cdot \boldsymbol{z}_2\right\}^2 = \mathrm{Im}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{z}_1\right\} + \mathrm{Re}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{z}_1\right\}$$
$$+ \mathrm{Im}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{z}_1\right\} + \mathrm{Re}\left\{\boldsymbol{z}_1\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{z}_1\right\} \qquad \text{(D.70)}$$

Using (D.69) in $\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\}$ and applying (D.70) for $\boldsymbol{z}_1^{\mathrm{T}} = \boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}$ and $\boldsymbol{z}_2 = \mathrm{vec}\left\{\boldsymbol{N}\right\}$ we find

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \mathbb{E}\left\{\mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\}\right\}$$

$$
\begin{aligned}
&+ \mathbb{E}\left\{\mathrm{Re}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\}\right\} \\
&+ \mathbb{E}\left\{\mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\}\right\} \\
&+ \mathbb{E}\left\{\mathrm{Re}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}} \cdot \mathrm{Im}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\}\right\}
\end{aligned}
\tag{D.71}
$$

Since the only random quantity in (D.71) is the noise matrix $\boldsymbol{N}$ we can move the quantities related to $\boldsymbol{W}_{\mathrm{mat}}$ and $\boldsymbol{r}_k$ out of the expectation operator. We are then left with the covariance matrices of the real part and the imaginary part of the noise, respectively, as well as with the cross-covariance matrix between the real and the imaginary part. However, since the noise is assumed to be circularly symmetric (cf. Section 9.2.4), zero mean, and i.i.d., we obtain

$$
\mathbb{E}\left\{\mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}}\right\} = \mathbb{E}\left\{\mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}}\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \boldsymbol{I}_{M \cdot N} \tag{D.72}
$$

$$
\mathbb{E}\left\{\mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}}\right\} = \mathbb{E}\left\{\mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}\right\}\right\}^{\mathrm{T}}\right\} = \boldsymbol{0}_{M \cdot N \times M \cdot N} \tag{D.73}
$$

Note that Gaussianity is not needed for these properties to hold. Consequently, the MSE expressions are still valid if the noise is not Gaussian, as long as it is still circularly symmetric, zero mean, and i.i.d. With these results, the MSE simplifies into

$$
\begin{aligned}
\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} &= \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Im}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\} + \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \mathrm{Re}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\} \cdot \mathrm{Re}\left\{\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\} \\
&= \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\|\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)}\right\|_2^2
\end{aligned}
\tag{D.74}
$$

which is the desired result. $\square$

The procedure for 2-D Standard Tensor-ESPRIT is in fact quite similar. The first step is to express the estimation error in $\mu_k^{(r)}$ in terms of the perturbation $\mathrm{vec}\left\{\boldsymbol{N}\right\} = \mathrm{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(3)}^{\mathrm{T}}\right\}$ (cf. (9.16)). This expression takes the form

$$
\Delta\mu_k^{(r)} = \mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \mathrm{vec}\left\{\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} = \mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{ten}} \cdot \mathrm{vec}\left\{\boldsymbol{N}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{D.75}
$$

since $\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\mathrm{s}]}\right]_{(R+1)}^{\mathrm{T}}$ depends linearly on $\mathrm{vec}\left\{\boldsymbol{N}\right\}$. Due to the fact that (D.75) has the same form as (D.69), the second step to expand the MSE expressions follows the same lines as for

R-D Standard ESPRIT, which immediately shows that the MSE becomes

$$\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \frac{\sigma_{\rm n}^2}{2} \cdot \left\| \boldsymbol{W}_{\rm ten}^{\rm T} \cdot \boldsymbol{r}_k^{(r)} \right\|_2^2 \tag{D.76}$$

Therefore, the final ingredient is finding an explicit expression for $\boldsymbol{W}_{\rm ten}$ which satisfies

$$\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\rm s]}\right]_{(3)}^{\rm T} = \boldsymbol{W}_{\rm ten} \cdot {\rm vec}\left\{\boldsymbol{N}\right\} + \mathcal{O}\left\{\Delta^2\right\}. \tag{D.77}$$

Recall from Proposition 12.3.1 that for $R = 2$, the HOSVD-based signal subspace estimation error $\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\rm s]}\right]_{(R+1)}$ can be expanded into

$$\left[\Delta\hat{\boldsymbol{\mathcal{U}}}^{[\rm s]}\right]_{(3)}^{\rm T} = (\boldsymbol{T}_1 \otimes \boldsymbol{T}_2) \cdot \Delta\boldsymbol{U}_{\rm s} + \left(\left[\Delta\boldsymbol{U}_1^{[\rm s]} \cdot \boldsymbol{U}_1^{[\rm s]^{\rm H}}\right] \otimes \boldsymbol{T}_2\right) \cdot \boldsymbol{U}_{\rm s} \quad + \left(\boldsymbol{T}_1 \otimes \left[\Delta\boldsymbol{U}_2^{[\rm s]} \cdot \boldsymbol{U}_2^{[\rm s]^{\rm H}}\right]\right) \cdot \boldsymbol{U}_{\rm s} + \mathcal{O}\left\{\Delta^2\right\}, \tag{D.78}$$

where $\Delta\boldsymbol{U}_{\rm s}$, $\Delta\boldsymbol{U}_1^{[\rm s]}$, and $\Delta\boldsymbol{U}_2^{[\rm s]}$ are given by

$$\Delta\boldsymbol{U}_{\rm s} = \boldsymbol{U}_{\rm n} \cdot \boldsymbol{U}_{\rm n}^{\rm H} \cdot \boldsymbol{N} \cdot \boldsymbol{V}_{\rm s} \cdot \boldsymbol{\Sigma}_{\rm s}^{-1} = \boldsymbol{V}_3^{[\rm n]^*} \cdot \boldsymbol{V}_3^{[\rm n]^{\rm T}} \cdot \boldsymbol{N} \cdot \boldsymbol{U}_3^{[\rm s]^*} \cdot \boldsymbol{\Sigma}_3^{[\rm s]^{-1}} \quad \text{and} \tag{D.79}$$

$$\Delta\boldsymbol{U}_r^{[\rm s]} = \boldsymbol{U}_r^{[\rm n]} \cdot \boldsymbol{U}_r^{[\rm n]^{\rm H}} \cdot [\boldsymbol{\mathcal{N}}]_{(r)} \cdot \boldsymbol{V}_r^{[\rm s]} \cdot \boldsymbol{\Sigma}_r^{[\rm s]^{-1}} \quad \text{for } r = 1, 2. \tag{D.80}$$

The first term in (D.78) is easily vectorized by applying property (3.7) for matrix product linear forms which yields the first term of $\boldsymbol{W}_{\rm ten}$ as

$$\begin{aligned} {\rm vec}\left\{(\boldsymbol{T}_1 \otimes \boldsymbol{T}_2) \cdot \Delta\boldsymbol{U}_{\rm s}\right\} &= {\rm vec}\left\{(\boldsymbol{T}_1 \otimes \boldsymbol{T}_2) \cdot \boldsymbol{V}_3^{[\rm n]^*} \cdot \boldsymbol{V}_3^{[\rm n]^{\rm T}} \cdot \boldsymbol{N} \cdot \boldsymbol{U}_3^{[\rm s]^*} \cdot \boldsymbol{\Sigma}_3^{[\rm s]^{-1}}\right\} \\ &= \left(\boldsymbol{U}_3^{[\rm s]^*} \cdot \boldsymbol{\Sigma}_3^{[\rm s]^{-1}}\right)^{\rm T} \otimes \left[(\boldsymbol{T}_1 \otimes \boldsymbol{T}_2) \cdot \boldsymbol{V}_3^{[\rm n]^*} \cdot \boldsymbol{V}_3^{[\rm n]^{\rm T}}\right] \cdot {\rm vec}\left\{\boldsymbol{N}\right\} \\ &= \left(\boldsymbol{\Sigma}_3^{[\rm s]^{-1}} \cdot \boldsymbol{U}_3^{[\rm s]^{\rm H}}\right) \otimes \left[(\boldsymbol{T}_1 \otimes \boldsymbol{T}_2) \cdot \boldsymbol{V}_3^{[\rm n]^*} \cdot \boldsymbol{V}_3^{[\rm n]^{\rm T}}\right] \cdot {\rm vec}\left\{\boldsymbol{N}\right\} \end{aligned} \tag{D.81}$$

However, for the second term in (D.78) we get

$$\begin{aligned} &{\rm vec}\left\{\left(\left[\boldsymbol{U}_1^{[\rm n]} \cdot \boldsymbol{U}_1^{[\rm n]^{\rm H}} \cdot [\boldsymbol{\mathcal{N}}]_{(1)} \cdot \boldsymbol{V}_1^{[\rm s]} \cdot \boldsymbol{\Sigma}_1^{[\rm s]^{-1}} \cdot \boldsymbol{U}_1^{[\rm s]^{\rm H}}\right] \otimes \boldsymbol{T}_2\right) \cdot \boldsymbol{U}_{\rm s}\right\} \\ &= \left(\boldsymbol{U}_{\rm s}^{\rm T} \otimes \boldsymbol{I}_M\right) \cdot {\rm vec}\left\{\left[\boldsymbol{U}_1^{[\rm n]} \cdot \boldsymbol{U}_1^{[\rm n]^{\rm H}} \cdot [\boldsymbol{\mathcal{N}}]_{(1)} \cdot \boldsymbol{V}_1^{[\rm s]} \cdot \boldsymbol{\Sigma}_1^{[\rm s]^{-1}} \cdot \boldsymbol{U}_1^{[\rm s]^{\rm H}}\right] \otimes \boldsymbol{T}_2\right\} \end{aligned} \tag{D.82}$$

by inserting (D.80) for $\Delta\boldsymbol{U}_1^{[\rm s]}$. To proceed we need to vectorize a Kronecker product. This is discussed in Section 3.1.2. Applying Proposition 3.1.1 we obtain

$$\begin{aligned}
&\left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \operatorname{vec}\left\{\left[\boldsymbol{U}_1^{[\mathrm{n}]} \cdot \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(1)} \cdot \boldsymbol{V}_1^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \cdot \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}}\right] \otimes \boldsymbol{T}_2\right\} \\
&= \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_2 \cdot \operatorname{vec}\left\{\left[\boldsymbol{U}_1^{[\mathrm{n}]} \cdot \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}} \cdot [\boldsymbol{\mathcal{N}}]_{(1)} \cdot \boldsymbol{V}_1^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \cdot \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}}\right]\right\} \\
&= \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_2 \cdot \left[\left(\boldsymbol{V}_1^{[\mathrm{s}]} \cdot \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \cdot \boldsymbol{U}_1^{[\mathrm{s}]^{\mathrm{H}}}\right)^{\mathrm{T}} \otimes \left(\boldsymbol{U}_1^{[\mathrm{n}]} \cdot \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}}\right)\right] \operatorname{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(1)}\right\}
\end{aligned} \tag{D.83}$$

where the matrix $\bar{\boldsymbol{T}}_2$ is constructed from the columns of $\boldsymbol{T}_2$ given by $\boldsymbol{t}_{2,m}$ for $m = 1, 2, \ldots, M_2$ in the following manner

$$\bar{\boldsymbol{T}}_2 = \boldsymbol{I}_{M_1} \otimes \begin{bmatrix} \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,1} \\ \vdots \\ \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,M_2} \end{bmatrix} \tag{D.84}$$

The final step is to rearrange the elements of $\operatorname{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(1)}\right\}$ so that they appear in the same order as in $\operatorname{vec}\{\boldsymbol{N}\}$. However, since $\boldsymbol{N} = [\boldsymbol{\mathcal{N}}]_{(3)}^{\mathrm{T}}$, this can easily be achieved in the following manner

$$\operatorname{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(1)}\right\} = \boldsymbol{K}_{M_2 \times (M_1 \cdot N)} \cdot \operatorname{vec}\{\boldsymbol{N}\} \tag{D.85}$$

where $\boldsymbol{K}_{M_2 \times (M_1 \cdot N)}$ is the commutation matrix (cf. equation (4.16)). This completes the derivation of the second term of $\boldsymbol{W}_{\mathrm{ten}}$. The third term is obtained in a similar manner. In this case, no permutation is needed, since $\operatorname{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(2)}\right\} = \operatorname{vec}\left\{[\boldsymbol{\mathcal{N}}]_{(3)}^{\mathrm{T}}\right\} = \operatorname{vec}\{\boldsymbol{N}\}$ (cf. equation (4.14)). □

D.13. Proof of Theorem 12.4.2

For simplicity, we show the proof for 1-D Unitary ESPRIT only. However, the ideas used in the proof carry over to the R-D case straightforwardly. Moreover, since the real-valued transformation used for R-D Unitary Tensor-ESPRIT is in fact the same (except for a specific structure of the left-$\boldsymbol{\Pi}$-real matrices, which is irrelevant for the proof), it also applies to the tensor case.

The theorem may seem obvious at first sight, as the real-valued transformation should not affect the performance at all. However, to show it rigorously, the only starting point we have available is the first-order perturbation expansion of the shift invariance equations. Since the parameters are extracted from the real-valued shift invariance equations in a different manner as in the complex-valued case (e.g., using the arctangent function), for a rigorous proof we need to develop a perturbation expansion for the real-valued shift invariance equations used in 1-D Unitary ESPRIT and then show its equivalence to (12.32). To this end, let $\boldsymbol{X}_0^{(\mathrm{fba})} \in \mathbb{C}^{M\times 2N}$ be the forward-backward averaged measurement matrix according to (12.31) and let $\varphi(\boldsymbol{X}_0^{(\mathrm{fba})}) = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{X}_0^{(\mathrm{fba})} \cdot \boldsymbol{Q}_{2N} \in \mathbb{R}^{M\times 2N}$ be the transformed real-valued measurement matrix. Their SVDs can be expressed as

$$\boldsymbol{X}_0^{(\mathrm{fba})} = \begin{bmatrix}\boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} & \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})}\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{\Sigma}_{\mathrm{s}}^{(\mathrm{fba})} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{V}_{\mathrm{s}}^{(\mathrm{fba})} & \boldsymbol{V}_{\mathrm{n}}^{(\mathrm{fba})}\end{bmatrix}^{\mathrm{H}} \tag{D.86}$$

$$\varphi(\boldsymbol{X}_0^{(\mathrm{fba})}) = \begin{bmatrix}\boldsymbol{E}_{\mathrm{s}} & \boldsymbol{E}_{\mathrm{n}}\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{\Sigma}_{\mathrm{s}}{}^{(\varphi)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix} \cdot \begin{bmatrix}\boldsymbol{W}_{\mathrm{s}} & \boldsymbol{W}_{\mathrm{n}}\end{bmatrix}^{\mathrm{H}}.$$

Since $\varphi(\boldsymbol{X}_0^{(\mathrm{fba})}) = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{X}_0^{(\mathrm{fba})} \cdot \boldsymbol{Q}_{2N}$ and the matrices $\boldsymbol{Q}_p$ are unitary, it is easy to see that an SVD of $\varphi(\boldsymbol{X}_0^{(\mathrm{fba})})$ is given by choosing

$$\boldsymbol{E}_{\mathrm{s}} = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}, \quad \boldsymbol{E}_{\mathrm{n}} = \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})}, \quad \boldsymbol{\Sigma}_{\mathrm{s}}{}^{(\varphi)} = \boldsymbol{\Sigma}_{\mathrm{s}}^{(\mathrm{fba})} \tag{D.87}$$

$$\boldsymbol{W}_{\mathrm{s}} = \boldsymbol{Q}_{2N}^{\mathrm{H}} \cdot \boldsymbol{V}_{\mathrm{s}}^{(\mathrm{fba})}, \quad \boldsymbol{W}_{\mathrm{n}} = \boldsymbol{Q}_{2N}^{\mathrm{H}} \cdot \boldsymbol{V}_{\mathrm{n}}^{(\mathrm{fba})}.$$

The shift invariance equation for 1-D Unitary ESPRIT can be written as

$$\boldsymbol{K}_1 \cdot \boldsymbol{E}_{\mathrm{s}} \cdot \boldsymbol{\Upsilon} = \boldsymbol{K}_2 \cdot \boldsymbol{E}_{\mathrm{s}} \tag{D.88}$$

where $\boldsymbol{\Upsilon} = \boldsymbol{R} \cdot \boldsymbol{\Omega} \cdot \boldsymbol{R}^{-1}$ and $\boldsymbol{\Omega} = \operatorname{diag}\left\{\left[\omega_1, \ldots, \omega_d\right]\right\}$ with $\omega_k = \tan(\mu_k/2)$, $k = 1, 2, \ldots, d$. Moreover, $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ are the real-values selection matrices (cf. (11.9) and (11.10)) which can be written as

$$\boldsymbol{K}_1 = 2 \cdot \operatorname{Re}\left\{\boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{Q}_M\right\} = \boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot (\boldsymbol{J}_1 + \boldsymbol{J}_2) \cdot \boldsymbol{Q}_M \tag{D.89}$$

$$\boldsymbol{K}_2 = 2 \cdot \operatorname{Im}\left\{\boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{Q}_M\right\} = \jmath \cdot \boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot (\boldsymbol{J}_1 - \boldsymbol{J}_2) \cdot \boldsymbol{Q}_M. \tag{D.90}$$

The second form follows from expanding the real part and the imaginary part according to $2 \cdot \operatorname{Re}\{x\} = x + x^*$ and $2 \cdot \operatorname{Im}\{x\} = -\jmath x + \jmath x^*$. The conjugated term $\boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{T}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{Q}^*$ can be simplified into $\boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{Q}$ using the fact that $\boldsymbol{J}_1 = \boldsymbol{\Pi}_{M^{(\mathrm{sel})}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{\Pi}_M$ (since the array must be centro-symmetric) and the fact that $\boldsymbol{Q}_p^* = \boldsymbol{\Pi}_p \cdot \boldsymbol{Q}_p$ (since $\boldsymbol{Q}_p$ is left-$\boldsymbol{\Pi}$-real).

The complex-valued shift invariance equation for the forward-backward averaged data has the following form

$$\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{\Psi} = \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}, \tag{D.91}$$

where $\boldsymbol{\Psi} = \boldsymbol{Q}^{(\mathrm{fba})} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$, $\boldsymbol{\Lambda} = \mathrm{diag}\left\{\left[\lambda_1, \dots, \lambda_d\right]\right\}$ with $\lambda_k = \mathrm{e}^{\jmath\mu_k}$, $k = 1, 2, \dots, d$.

Before we proceed, we require the following two lemmas:

Lemma D.13.1. *The following identites are satisfied*

$$\left(\boldsymbol{J}_1 + \boldsymbol{J}_2\right) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} = \boldsymbol{J}_1 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})} \cdot \breve{\boldsymbol{\Psi}} \tag{D.92}$$

$$\left(\boldsymbol{J}_1 - \boldsymbol{J}_2\right) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} = \boldsymbol{J}_2 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})} \cdot \mathring{\boldsymbol{\Psi}}, \tag{D.93}$$

where $\breve{\boldsymbol{\Psi}} = \boldsymbol{I}_d + \boldsymbol{\Psi} = \boldsymbol{Q}^{(\mathrm{fba})} \cdot \left(\boldsymbol{I}_d + \boldsymbol{\Lambda}\right) \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$ *and* $\mathring{\boldsymbol{\Psi}} = -\boldsymbol{I}_d + \boldsymbol{\Psi}^{-1} = \boldsymbol{Q}^{(\mathrm{fba})} \cdot \left(-\boldsymbol{I}_d + \boldsymbol{\Lambda}^{-1}\right) \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$.

Proof. The identities follow straightforwardly from $\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{\Psi} = \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$. For the first identity we simply add $\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$ to both sides of the equation and then factor out the common terms. For the second identity we subtract $\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$, factor out the common terms, and finally replace $\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$ by $\boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{\Psi}^{-1}$. □

Lemma D.13.2. *The solution* $\boldsymbol{\Psi}$ *to* (D.91) *and the solution* $\boldsymbol{\Upsilon}$ *to* (D.88) *have the same eigenvectors, i.e.,* $\boldsymbol{Q}^{(\mathrm{fba})} = \boldsymbol{R}$. *Moreover, their eigenvalues are related as* $\omega_k = \jmath\frac{1-\lambda_k}{1+\lambda_k}$ *or equivalently* $\lambda_k = \frac{1+\jmath\omega_k}{1-\jmath\omega_k}$.

Proof. The lemma follows by performing the steps that were used in the derivation for Unitary ESPRIT to transform the complex-valued shift invariance equation into the real-valued shift invariance equation ([HN95, HRD08]) in the reverse order. Starting from $\boldsymbol{\Upsilon} = \left(\boldsymbol{K}_1 \cdot \boldsymbol{E}_{\mathrm{s}}\right)^{+} \cdot \boldsymbol{K}_2 \cdot \boldsymbol{E}_{\mathrm{s}}$ and replacing $\boldsymbol{E}_{\mathrm{s}}$ with the help of (D.87) and $\boldsymbol{K}_n$ with the help of (D.89) and (D.90) we get

$$\begin{aligned}
\boldsymbol{\Upsilon} &= \left(\boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot \left(\boldsymbol{J}_1 + \boldsymbol{J}_2\right) \cdot \boldsymbol{Q}_M \cdot \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}\right)^{+} \cdot \jmath \cdot \boldsymbol{Q}_{M^{(\mathrm{sel})}}^{\mathrm{H}} \cdot \left(\boldsymbol{J}_1 - \boldsymbol{J}_2\right) \cdot \boldsymbol{Q}_M \cdot \boldsymbol{Q}_M^{\mathrm{H}} \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \\
&= \left(\left(\boldsymbol{J}_1 + \boldsymbol{J}_2\right) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}\right)^{+} \cdot \jmath \cdot \left(\boldsymbol{J}_1 - \boldsymbol{J}_2\right) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \\
&= \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})} \cdot \breve{\boldsymbol{\Psi}}\right)^{+} \cdot \jmath \cdot \boldsymbol{J}_2 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})} \cdot \mathring{\boldsymbol{\Psi}} \\
&= \jmath \cdot \breve{\boldsymbol{\Psi}}^{-1} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})}\right)^{+} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{U}_{s}^{(\mathrm{fba})} \cdot \mathring{\boldsymbol{\Psi}} \\
&= \jmath \cdot \breve{\boldsymbol{\Psi}}^{-1} \cdot \boldsymbol{\Psi} \cdot \mathring{\boldsymbol{\Psi}} \\
&= \jmath \cdot \boldsymbol{Q}^{(\mathrm{fba})} \cdot \left(\boldsymbol{I}_d + \boldsymbol{\Lambda}\right)^{-1} \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}} \cdot \boldsymbol{Q}^{(\mathrm{fba})} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}} \cdot \boldsymbol{Q}^{(\mathrm{fba})} \cdot \left(-\boldsymbol{I}_d + \boldsymbol{\Lambda}^{-1}\right) \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}
\end{aligned}$$

$$= \jmath \cdot \boldsymbol{Q}^{(\mathrm{fba})} \cdot \underbrace{(\boldsymbol{I}_d + \boldsymbol{\Lambda})^{-1} \cdot (\boldsymbol{I}_d - \boldsymbol{\Lambda})}_{\mathrm{diag}\left\{\left[\frac{1-\lambda_k}{1+\lambda_k}\right]_{k=1,2,\ldots,d}\right\}} \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$$

$$= \boldsymbol{Q}^{(\mathrm{fba})} \cdot \boldsymbol{\Omega} \cdot \boldsymbol{Q}^{(\mathrm{fba})^{-1}}, \tag{D.94}$$

where we have used Lemma D.13.1 in the second step. □

Since the real-valued shift-invariance equation in (D.88) has the same algebraic form as its complex-valued counterpart (D.91), the same arguments can be applied to develop a first-order perturbation expansion. In fact, following the three steps discussed in Appendix D.11, we find that the second step (cf. (D.64)) and the third step (cf. (D.65)) lead to the same result (where $\boldsymbol{J}_1, \boldsymbol{J}_2, \boldsymbol{U}_\mathrm{s}$, and $\boldsymbol{\Psi}$ are consistently exchanged by $\boldsymbol{K}_1, \boldsymbol{K}_2, \boldsymbol{E}_\mathrm{s}$, and $\boldsymbol{\Upsilon}$, respectively). For the first step we need a Taylor series expansion of $\omega_k = \tan(\mu_k/2)$ which is given by

$$\omega_k + \Delta\omega \approx \underbrace{\tan(\mu_k/2)}_{\omega_k} + \Delta\mu \cdot \underbrace{\left(\frac{\tan^2(\mu_k/2)}{2} + \frac{1}{2}\right)}_{\frac{\omega_k^2+1}{2}}$$

$$\Rightarrow \Delta\mu \approx \Delta\omega \cdot \frac{2}{\omega_k^2 + 1}. \tag{D.95}$$

Combining (D.95) with the corresponding real-valued expressions for (D.64) and (D.65) we obtain

$$\Delta\mu_k = \boldsymbol{p}_k^{(\mathrm{fba})^\mathrm{T}} \cdot (\boldsymbol{K}_1 \cdot \boldsymbol{E}_\mathrm{s})^+ \cdot (\boldsymbol{K}_2 - \omega_k \cdot \boldsymbol{K}_1) \cdot \Delta\boldsymbol{E}_\mathbf{s} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{2}{\omega_k^2 + 1} \tag{D.96}$$

where $\boldsymbol{q}_k^{(\mathrm{fba})}$ is the k-th column of $\boldsymbol{Q}^{(\mathrm{fba})}$ and $\boldsymbol{p}_k^{(\mathrm{fba})^\mathrm{T}}$ is the k-th row of $\boldsymbol{P}^{(\mathrm{fba})} = \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$. Moreover, the perturbation of the real-valued subspace $\boldsymbol{E}_\mathrm{s}$ is expanded in terms of the transformed noise contribution $\varphi(\boldsymbol{N}^{(\mathrm{fba})}) = \boldsymbol{Q}_M^\mathrm{H} \cdot \boldsymbol{N}^{(\mathrm{fba})} \cdot \boldsymbol{Q}_{2N}$ as

$$\Delta\boldsymbol{E}_\mathbf{s} = \boldsymbol{E}_\mathrm{n} \cdot \boldsymbol{E}_\mathrm{n}^\mathrm{H} \cdot \varphi(\boldsymbol{N}^{(\mathrm{fba})}) \cdot \boldsymbol{W}_\mathrm{s} \cdot \boldsymbol{\Sigma}_\mathrm{s}^{(\varphi)^{-1}}. \tag{D.97}$$

Inserting (D.97) into (D.96) and applying the identities (D.87) as well as (D.89) and (D.90) we obtain

$$\Delta\mu_k = \boldsymbol{p}_k^{(\mathrm{fba})^\mathrm{T}} \cdot \left((\boldsymbol{J}_1 + \boldsymbol{J}_2) \cdot \boldsymbol{U}_\mathrm{s}^{(\mathrm{fba})}\right)^+ \cdot \left(\jmath(\boldsymbol{J}_1 - \boldsymbol{J}_2) - \omega_k(\boldsymbol{J}_1 + \boldsymbol{J}_2)\right) \cdot \boldsymbol{U}_\mathrm{n}^{(\mathrm{fba})} \cdot \boldsymbol{U}_\mathrm{n}^{(\mathrm{fba})^\mathrm{H}}$$
$$\cdot \boldsymbol{N}^{(\mathrm{fba})} \cdot \boldsymbol{V}_s^{(\mathrm{fba})} \cdot \boldsymbol{\Sigma}_\mathrm{s}^{(\mathrm{fba})^{-1}} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{2}{\omega_k^2 + 1}$$

$$=\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \left((\boldsymbol{J}_1 + \boldsymbol{J}_2) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \right)^+ \cdot \left(\jmath(\boldsymbol{J}_1 - \boldsymbol{J}_2) - \omega_k(\boldsymbol{J}_1 + \boldsymbol{J}_2) \right) \cdot \Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{2}{\omega_k^2 + 1}, \tag{D.98}$$

where $\Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} = \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})} \cdot \boldsymbol{U}_{\mathrm{n}}^{(\mathrm{fba})^{\mathrm{H}}} \cdot \boldsymbol{N}^{(\mathrm{fba})} \cdot \boldsymbol{V}_s^{(\mathrm{fba})} \cdot \boldsymbol{\Sigma}_{\mathrm{s}}^{(\mathrm{fba})^{-1}}$. To simplify (D.98) further, consider the term $(\jmath(\boldsymbol{J}_1 - \boldsymbol{J}_2) - \omega_k(\boldsymbol{J}_1 + \boldsymbol{J}_2))$ first. Using the relation $\omega_k = \jmath\frac{1-\lambda_k}{1+\lambda_k}$ from Lemma D.13.2 we can rewrite this term as

$$\begin{aligned}
\jmath \cdot (\boldsymbol{J}_1 - \boldsymbol{J}_2) - \jmath \cdot \frac{1-\lambda_k}{1+\lambda_k} \cdot (\boldsymbol{J}_1 + \boldsymbol{J}_2) &= \jmath \cdot \boldsymbol{J}_1 \cdot \left(1 - \frac{1-\lambda_k}{1+\lambda_k}\right) - \jmath \cdot \boldsymbol{J}_2 \cdot \left(1 + \frac{1-\lambda_k}{1+\lambda_k}\right) \\
&= \jmath \cdot \boldsymbol{J}_1 \cdot \frac{2 \cdot \lambda_k}{1+\lambda_k} - \jmath \cdot \boldsymbol{J}_2 \cdot \frac{2}{1+\lambda_k} \\
&= \jmath \cdot (\boldsymbol{J}_1 \cdot \lambda_k - \boldsymbol{J}_2) \cdot \frac{2}{1+\lambda_k}.
\end{aligned} \tag{D.99}$$

Moreover, the term $\frac{2}{\omega_k^2+1}$ can be expressed in terms of λ_k via Lemma D.13.2. We obtain

$$\begin{aligned}
\frac{2}{\omega_k^2 + 1} &= \frac{2}{\left(\jmath\frac{1-\lambda_k}{1+\lambda_k}\right)^2 + 1} = \frac{2 \cdot (\lambda_k + 1)^2}{-(\lambda_k - 1)^2 + (\lambda_k + 1)^2} \\
&= \frac{2 \cdot (\lambda_k + 1)^2}{4\lambda_k} = \frac{(\lambda_k + 1)^2}{2\lambda_k}.
\end{aligned} \tag{D.100}$$

Inserting (D.99) and (D.100) into (D.98) and replacing $(\boldsymbol{J}_1 + \boldsymbol{J}_2) \cdot \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})}$ via (D.92) we get

$$\begin{aligned}
\Delta\mu_k =& \jmath \cdot \underbrace{\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \breve{\boldsymbol{\Psi}}^{-1}}_{\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot (1+\lambda_k)^{-1}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_s^{(\mathrm{fba})}\right)^+ \cdot (\boldsymbol{J}_1 \cdot \lambda_k - \boldsymbol{J}_2) \cdot \Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{2}{1+\lambda_k} \cdot \frac{(\lambda_k+1)^2}{2\lambda_k} \\
=& \jmath \cdot \boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \frac{1}{1+\lambda_k} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_s^{(\mathrm{fba})}\right)^+ \cdot (\boldsymbol{J}_1 \cdot \lambda_k - \boldsymbol{J}_2) \cdot \Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{(\lambda_k+1)}{\lambda_k} \\
=& \jmath \cdot \boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_s^{(\mathrm{fba})}\right)^+ \cdot (\boldsymbol{J}_1 - \boldsymbol{J}_2/\lambda_k) \cdot \Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})} \cdot \frac{\lambda_k}{1+\lambda_k} \cdot \frac{\lambda_k+1}{\lambda_k} \\
=& -\jmath \cdot \boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_s^{(\mathrm{fba})}\right)^+ \cdot (\boldsymbol{J}_2/\lambda_k - \boldsymbol{J}_1) \cdot \Delta \boldsymbol{U}_{\mathrm{s}}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})},
\end{aligned} \tag{D.101}$$

where $\boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot \breve{\boldsymbol{\Psi}}^{-1} = \boldsymbol{p}_k^{(\mathrm{fba})^{\mathrm{T}}} \cdot (1+\lambda_k)^{-1}$ follows because $\breve{\boldsymbol{\Psi}}$ has the eigendecomposition $\breve{\boldsymbol{\Psi}} = \boldsymbol{Q}^{(\mathrm{fba})} \cdot (\boldsymbol{I}_d + \boldsymbol{\Lambda}) \cdot \boldsymbol{P}^{(\mathrm{fba})}$ where $\boldsymbol{P}^{(\mathrm{fba})} = \boldsymbol{Q}^{(\mathrm{fba})^{-1}}$ and hence $\breve{\boldsymbol{\Psi}}^{-1} = \boldsymbol{Q}^{(\mathrm{fba})} \cdot (\boldsymbol{I}_d + \boldsymbol{\Lambda})^{-1} \cdot \boldsymbol{P}^{(\mathrm{fba})}$.

As a final step we notice that (D.101) must be real-valued since we have started from the purely real-valued expansion (D.96) and only used equivalence transforms to arrive at (D.101). However, if $-\jmath \cdot z \in \mathbb{R}$ for $z \in \mathbb{C}$ this implies that $\mathrm{Re}\{z\} = 0$ and hence $-\jmath \cdot z = \mathrm{Im}\{z\}$. Consequently, (D.101) can also be written as

$$\Delta\mu_k = \mathrm{Im}\left\{\boldsymbol{p}_k^{(\mathrm{fba})^\mathrm{T}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_s^{(\mathrm{fba})}\right)^+ \cdot (\boldsymbol{J}_2/\lambda_k - \boldsymbol{J}_1) \cdot \Delta\boldsymbol{U}_\mathrm{s}^{(\mathrm{fba})} \cdot \boldsymbol{q}_k^{(\mathrm{fba})}\right\}. \tag{D.102}$$

Obviously, (D.102) is the same as the first-order expansion for FBA-ESPRIT shown in (12.32). This concludes the proof of the theorem. □

D.14. Proof of Theorem 12.4.3

As pointed out in Section 12.4.4, the inclusion of Forward-Backward-Averaging leads to a very similar model, where all quantities originating from the noise-free observation $\boldsymbol{X}_0$ (or $\boldsymbol{\mathcal{X}}_0$) are replaced by the corresponding quantities for $\boldsymbol{X}_0^{(\mathrm{fba})}$ (or $\boldsymbol{\mathcal{X}}_0^{(\mathrm{fba})}$). This also applies to large parts of the derivation of the MSE, which can be performed in a quite similar manner as shown in the proof of Theorem 12.4.1.

For instance, for R-D Standard ESPRIT with Forward-Backward-Averaging, we arrive at a relation similar to (D.71) which reads as

$$\begin{aligned}
&\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} \\
&= \mathbb{E}\left\{\mathrm{Im}\left\{\boldsymbol{r}_k^{(r)(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^\mathrm{T} \cdot \mathrm{Im}\left\{\boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)(\mathrm{fba})}\right\}\right\} \\
&+ \mathbb{E}\left\{\mathrm{Re}\left\{\boldsymbol{r}_k^{(r)(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^\mathrm{T} \cdot \mathrm{Re}\left\{\boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)(\mathrm{fba})}\right\}\right\} \\
&+ \mathbb{E}\left\{\mathrm{Im}\left\{\boldsymbol{r}_k^{(r)(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^\mathrm{T} \cdot \mathrm{Re}\left\{\boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)(\mathrm{fba})}\right\}\right\} \\
&+ \mathbb{E}\left\{\mathrm{Re}\left\{\boldsymbol{r}_k^{(r)(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})}\right\} \cdot \mathrm{Im}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\} \cdot \mathrm{Re}\left\{\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^\mathrm{T} \cdot \mathrm{Im}\left\{\boldsymbol{W}_\mathrm{mat}^{(\mathrm{fba})^\mathrm{T}} \cdot \boldsymbol{r}_k^{(r)(\mathrm{fba})}\right\}\right\}
\end{aligned} \tag{D.103}$$

At this point, the derivation has to be extended to take into account the effect of the Forward-Backward-Averaging onto the noise. In particular, we need the covariance matrices of the real part and the imaginary part of the augmented noise matrix, respectively, as well as the cross-covariance matrix between the real part and the imaginary part.

To this end, we can express $\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}$ as

$$\mathrm{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\} = \mathrm{vec}\left\{\left[\boldsymbol{N}, \quad \boldsymbol{\Pi}_M \cdot \boldsymbol{N}^* \cdot \boldsymbol{\Pi}_N\right]\right\} = \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{N}\right\} \\ (\boldsymbol{\Pi}_N \otimes \boldsymbol{\Pi}_M) \cdot \mathrm{vec}\left\{\boldsymbol{N}^*\right\} \end{bmatrix}$$

$$
\begin{aligned}
&= \begin{bmatrix} \operatorname{vec}\{\operatorname{Re}\{\boldsymbol{N}\}\} \\ \boldsymbol{\Pi}_{M\cdot N}\cdot\operatorname{vec}\{\operatorname{Re}\{\boldsymbol{N}\}\} \end{bmatrix} + \jmath\cdot \begin{bmatrix} \operatorname{vec}\{\operatorname{Im}\{\boldsymbol{N}\}\} \\ -\boldsymbol{\Pi}_{M\cdot N}\cdot\operatorname{vec}\{\operatorname{Im}\{\boldsymbol{N}\}\} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{I}_{M\cdot N} \\ \boldsymbol{\Pi}_{M\cdot N} \end{bmatrix}\cdot\operatorname{vec}\{\operatorname{Re}\{\boldsymbol{N}\}\} + \jmath\cdot\begin{bmatrix} \boldsymbol{I}_{M\cdot N} \\ -\boldsymbol{\Pi}_{M\cdot N} \end{bmatrix}\cdot\operatorname{vec}\{\operatorname{Im}\{\boldsymbol{N}\}\}
\end{aligned} \tag{D.104}
$$

Equation (D.104) allows us to express the real part and the imaginary part of $\boldsymbol{N}^{(\mathrm{fba})}$ in terms of the real part and the imaginary part of $\boldsymbol{N}$, respectively. The first immediate observation is that the real part and the imaginary part of the noise matrix $\boldsymbol{N}$ do not mix and hence

$$
\begin{aligned}
&\mathbb{E}\left\{\operatorname{Re}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}\cdot\operatorname{Im}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^{\mathrm{T}}\right\} \\
=&\mathbb{E}\left\{\operatorname{Im}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}\cdot\operatorname{Re}\left\{\operatorname{vec}\left\{\boldsymbol{N}\right\}^{(\mathrm{fba})}\right\}^{\mathrm{T}}\right\} = \boldsymbol{0}_{M\cdot N\times M\cdot N}
\end{aligned} \tag{D.105}
$$

However, for the covariance matrix of the real part of $\boldsymbol{N}^{(\mathrm{fba})}$ we obtain

$$
\begin{aligned}
&\mathbb{E}\left\{\operatorname{Re}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}\cdot\operatorname{Re}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^{\mathrm{T}}\right\} \\
=&\begin{bmatrix} \boldsymbol{I}_{M\cdot N} \\ \boldsymbol{\Pi}_{M\cdot N} \end{bmatrix}\cdot\underbrace{\mathbb{E}\left\{\operatorname{Re}\{\operatorname{vec}\{\boldsymbol{N}\}\}\cdot\operatorname{Re}\{\operatorname{vec}\{\boldsymbol{N}\}\}^{\mathrm{T}}\right\}}_{\frac{\sigma_{\mathrm{n}}^2}{2}\cdot\boldsymbol{I}_{M\cdot N\times M\cdot N}}\cdot\begin{bmatrix} \boldsymbol{I}_{M\cdot N} \\ \boldsymbol{\Pi}_{M\cdot N} \end{bmatrix}^{\mathrm{T}} \\
=&\frac{\sigma_{\mathrm{n}}^2}{2}\cdot\begin{bmatrix} \boldsymbol{I}_{M\cdot N} & \boldsymbol{\Pi}_{M\cdot N} \\ \boldsymbol{\Pi}_{M\cdot N} & \boldsymbol{I}_{M\cdot N} \end{bmatrix} = \frac{\sigma_{\mathrm{n}}^2}{2}\cdot\left(\boldsymbol{I}_{2MN}+\boldsymbol{\Pi}_{2MN}\right)
\end{aligned} \tag{D.106}
$$

Likewise, the covariance matrix of the imaginary part of $\boldsymbol{N}^{(\mathrm{fba})}$ becomes

$$
\begin{aligned}
\mathbb{E}\left\{\operatorname{Im}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}\cdot\operatorname{Im}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\right\}^{\mathrm{T}}\right\} &= \frac{\sigma_{\mathrm{n}}^2}{2}\cdot\begin{bmatrix} \boldsymbol{I}_{M\cdot N} & -\boldsymbol{\Pi}_{M\cdot N} \\ -\boldsymbol{\Pi}_{M\cdot N} & \boldsymbol{I}_{M\cdot N} \end{bmatrix} \\
&= \frac{\sigma_{\mathrm{n}}^2}{2}\cdot\left(\boldsymbol{I}_{2MN}-\boldsymbol{\Pi}_{2MN}\right)
\end{aligned} \tag{D.107}
$$

Note that this is quite an interesting result by itself: It is often claimed that Forward-Backward-Averaging does not alter the noise statistics which can be seen from the fact that $\mathbb{E}\left\{\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}\cdot\operatorname{vec}\left\{\boldsymbol{N}^{(\mathrm{fba})}\right\}^{\mathrm{H}}\right\} = \sigma_{\mathrm{n}}^2\cdot\boldsymbol{I}_{2MN}$ (which can also be verified at this point by summing (D.106) and (D.107)). While it is true that the (complex) entries of $\boldsymbol{N}^{(\mathrm{fba})}$ are still uncorrelated, the derivations of this section show that this is not true any more for the real part and the imaginary part of the elements of $\boldsymbol{N}^{(\mathrm{fba})}$. Note that the first terms in (D.106) and (D.107) are both equal to $\sigma_{\rm n}^2/2 \cdot \boldsymbol{I}_{2MN}$ and hence similar to the corresponding result (D.72) in the proof of Theorem 12.4.1 for R-D Standard ESPRIT. This explains that the resulting MSE expression contains $\frac{\sigma_{\rm n}^2}{2} \cdot \left\| \boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}} \right\|_2^2$ as a first term, following the same arguments as in the proof for Theorem 12.4.1. The remaining terms in (D.106) and (D.107) are given by $\pm\sigma_{\rm n}^2/2 \cdot \boldsymbol{\Pi}_{2MN}$. Inserting these into (D.103) we obtain

$$\begin{aligned}
&\mathrm{Im}\left\{\boldsymbol{r}_k^{(r)^{({\rm fba})^{\rm T}}} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})}\right\} \cdot \boldsymbol{\Pi}_{2MN} \cdot \mathrm{Im}\left\{\boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}}\right\} \\
&- \mathrm{Re}\left\{\boldsymbol{r}_k^{(r)^{({\rm fba})^{\rm T}}} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})}\right\} \cdot \boldsymbol{\Pi}_{2MN} \cdot \mathrm{Re}\left\{\boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}}\right\} \\
&= - \mathrm{Re}\left\{\boldsymbol{r}_k^{(r)^{({\rm fba})^{\rm T}}} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})} \cdot \boldsymbol{\Pi}_{2MN} \cdot \boldsymbol{W}_{\rm mat}^{({\rm fba})^{\rm T}} \cdot \boldsymbol{r}_k^{(r)^{({\rm fba})}}\right\}
\end{aligned} \tag{D.108}$$

where we have used the fact that for arbitrary complex vectors $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ we have

$$\mathrm{Re}\left\{\boldsymbol{z}_1^{\rm T} \cdot \boldsymbol{z}_2\right\} = \mathrm{Re}\left\{\boldsymbol{z}_1^{\rm T}\right\} \cdot \mathrm{Re}\left\{\boldsymbol{z}_2\right\} - \mathrm{Im}\left\{\boldsymbol{z}_1^{\rm T}\right\} \cdot \mathrm{Im}\left\{\boldsymbol{z}_2\right\}. \tag{D.109}$$

This completes the proof for the MSE expression for R-D Unitary ESPRIT given in equation (12.36). The proof for (12.37) for 2-D Unitary Tensor-ESPRIT proceeds in a completely analogous fashion as we only replace $\boldsymbol{W}_{\rm mat}^{({\rm fba})}$ by $\boldsymbol{W}_{\rm ten}^{({\rm fba})}$ consistently. □

D.15. Proof of Theorem 12.4.4

Without regularization, the cost function for 1-D Structured Least Squares can be expressed as [Haa97b]

$$\hat{\boldsymbol{\Psi}}_{\rm SLS} = \underset{\boldsymbol{\Psi}, \Delta\overline{\boldsymbol{U}}_{\rm s}}{\arg\min} \left\| \boldsymbol{J}_1 \cdot \left(\hat{\boldsymbol{U}}_{\rm s} + \Delta\overline{\boldsymbol{U}}_{\rm s}\right) \cdot \boldsymbol{\Psi} - \boldsymbol{J}_2 \cdot \left(\hat{\boldsymbol{U}}_{\rm s} + \Delta\overline{\boldsymbol{U}}_{\rm s}\right) \right\|_{\rm F}^2. \tag{D.110}$$

where we have used $\Delta\overline{\boldsymbol{U}}_{\rm s}$ only to avoid confusion with the $\Delta\boldsymbol{U}_{\rm s}$ associated to the estimation error in $\hat{\boldsymbol{U}}_{\rm s}$. Note that the cost function is solved in an iterative manner starting with $\Delta\overline{\boldsymbol{U}}_{\rm s} = \boldsymbol{0}_{M\times d}$ and with $\boldsymbol{\Psi} = \boldsymbol{\Psi}_{\rm LS}$, where $\boldsymbol{\Psi}_{\rm LS} = \left(\boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\rm s}\right)^+ \cdot \left(\boldsymbol{J}_2 \cdot \hat{\boldsymbol{U}}_{\rm s}\right)$ represents the LS solution to the shift invariance equation. As we compute only a single iteration we find one update term for $\hat{\boldsymbol{U}}_{\rm s}$ and one for $\boldsymbol{\Psi}_{\rm LS}$ which we denote as $\Delta\boldsymbol{U}_{\rm s,SLS}$ and $\Delta\boldsymbol{\Psi}_{\rm SLS}$ (i.e., $\Delta\boldsymbol{U}_{\rm s,SLS}$ represents the $\Delta\overline{\boldsymbol{U}}_{\rm s}$ which minimizes the linearized version of (D.110)). In other words, the cost function becomes

$$\hat{\boldsymbol{\Psi}}_{\rm SLS} = \hat{\boldsymbol{\Psi}}_{\rm LS} + \Delta\boldsymbol{\Psi}_{\rm SLS} \quad \text{where} \tag{D.111}$$

$$
\begin{aligned}
\Delta\boldsymbol{\Psi}_{\mathrm{SLS}} &= \underset{\Delta\boldsymbol{\Psi},\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}}{\arg\min} \left\| \boldsymbol{J}_1 \cdot \left(\hat{\boldsymbol{U}}_{\mathrm{s}} + \Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right) \cdot \left(\boldsymbol{\Psi}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}\right) - \boldsymbol{J}_2 \cdot \left(\hat{\boldsymbol{U}}_{\mathrm{s}} + \Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right) \right\|_{\mathrm{F}}^2 \\
&= \underset{\Delta\boldsymbol{\Psi},\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}}{\arg\min} \left\| \boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi}_{\mathrm{LS}} - \boldsymbol{J}_2 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} + \boldsymbol{J}_1 \cdot \Delta\overline{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi}_{\mathrm{LS}} + \boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi} - \boldsymbol{J}_2 \cdot \Delta\overline{\boldsymbol{U}}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\} \right\|_{\mathrm{F}}^2 \\
&= \underset{\Delta\boldsymbol{\Psi},\Delta\boldsymbol{U}_{\mathrm{s}}}{\arg\min} \left\| \boldsymbol{R}_{\mathrm{LS}} + \boldsymbol{J}_1 \cdot \Delta\overline{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi}_{\mathrm{LS}} + \boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi} - \boldsymbol{J}_2 \cdot \Delta\overline{\boldsymbol{U}}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\} \right\|_{\mathrm{F}}^2
\end{aligned}
\tag{D.112}
$$

where we have defined the matrix $\boldsymbol{R}_{\mathrm{LS}} = \boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \boldsymbol{\Psi}_{\mathrm{LS}} - \boldsymbol{J}_2 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}$ which contains the residual error in the shift invariance equation after the LS fit. Since (D.112) is linearized by skipping the quadratic terms in $\mathcal{O}\left\{\Delta^2\right\}$, it is easily solved by an LS fit. To express the result in closed-form we vectorize (D.112) using the fact that $\|\boldsymbol{A}\|_{\mathrm{F}} = \|\mathrm{vec}\{\boldsymbol{A}\}\|_2$ and obtain

$$
\begin{aligned}
\Delta\boldsymbol{\psi}_{\mathrm{SLS}} &= \underset{\mathrm{vec}\{\Delta\boldsymbol{\Psi}\},\mathrm{vec}\left\{\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right\}}{\arg\min} \Big\| \boldsymbol{r}_{\mathrm{LS}} + \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) \cdot \mathrm{vec}\left\{\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right\} \\
&\qquad + \left(\boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}\right)\right) \cdot \mathrm{vec}\{\Delta\boldsymbol{\Psi}\} - \left(\boldsymbol{I}_d \otimes \boldsymbol{J}_2\right) \cdot \mathrm{vec}\left\{\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\} \Big\|_2^2 \\
\Rightarrow \Delta\boldsymbol{\psi}_{\mathrm{SLS}} &= \underset{\mathrm{vec}\{\Delta\boldsymbol{\Psi}\},\mathrm{vec}\left\{\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right\}}{\arg\min} \left\| \boldsymbol{r}_{\mathrm{LS}} + \hat{\boldsymbol{F}}_{\mathrm{SLS}} \cdot \begin{bmatrix} \mathrm{vec}\{\Delta\boldsymbol{\Psi}\} \\ \mathrm{vec}\left\{\Delta\overline{\boldsymbol{U}}_{\mathrm{s}}\right\} \end{bmatrix} + \mathcal{O}\left\{\Delta^2\right\} \right\|_2^2
\end{aligned}
\tag{D.113}
$$

$$
\Rightarrow \begin{bmatrix} \Delta\boldsymbol{\psi}_{\mathrm{SLS}} \\ \Delta\boldsymbol{u}_{\mathrm{s,SLS}} \end{bmatrix} = -\hat{\boldsymbol{F}}_{\mathrm{SLS}}^{+} \cdot \boldsymbol{r}_{\mathrm{LS}} \tag{D.114}
$$

where we have introduced the vectorized quantities $\boldsymbol{r}_{\mathrm{LS}} = \mathrm{vec}\{\boldsymbol{R}_{\mathrm{LS}}\}$, $\Delta\boldsymbol{\psi}_{\mathrm{SLS}} = \mathrm{vec}\{\Delta\boldsymbol{\Psi}_{\mathrm{SLS}}\}$, and $\Delta\boldsymbol{u}_{\mathrm{s,SLS}} = \mathrm{vec}\{\Delta\boldsymbol{U}_{\mathrm{s,SLS}}\}$, respectively. We have also skipped the quadratic terms $\mathcal{O}\left\{\Delta^2\right\}$ in (D.113) for the solution in (D.114). Moreover, the matrix $\hat{\boldsymbol{F}}_{\mathrm{SLS}}$ becomes

$$
\hat{\boldsymbol{F}}_{\mathrm{SLS}} = \left[\boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}\right),\ \left(\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) - \left(\boldsymbol{I}_d \otimes \boldsymbol{J}_2\right)\right] \in \mathbb{C}^{(M-1)d \times (d^2 + M \cdot d)} \tag{D.115}
$$

Therefore, our next goal is to find a first order expansion of $\Delta\boldsymbol{\psi}_{\mathrm{SLS}}$ in (D.114). This looks difficult at first sight as it involves an expansion of a pseudo-inverse due to $\hat{\boldsymbol{F}}_{\mathrm{SLS}}$. However, this step simplifies significantly by realizing that $\hat{\boldsymbol{F}}_{\mathrm{SLS}}$ can be expressed as $\hat{\boldsymbol{F}}_{\mathrm{SLS}} = \boldsymbol{F}_{\mathrm{SLS}} + \Delta\hat{\boldsymbol{F}}_{\mathrm{SLS}}$, where

$$
\begin{aligned}
\boldsymbol{F}_{\mathrm{SLS}} &= \left[\boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}\right),\ \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) - \left(\boldsymbol{I}_d \otimes \boldsymbol{J}_2\right)\right] \in \mathbb{C}^{(M-1)d \times (d^2 + M \cdot d)} \\
\Delta\boldsymbol{F}_{\mathrm{SLS}} &= \left[\boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}}\right),\ \left(\Delta\boldsymbol{\Psi}_{\mathrm{LS}}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right)\right]
\end{aligned}
\tag{D.116}
$$

where $\hat{\boldsymbol{\Psi}}_{\mathrm{LS}} = \boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}_{\mathrm{LS}}$. Since $\boldsymbol{F}_{\mathrm{SLS}}$ is not random (i.e., only dependent on $\boldsymbol{X}_0$ but not on $\boldsymbol{N}$) and $\Delta\boldsymbol{\Psi}_{\mathrm{LS}} = \mathbf{0}_{d\times d} + \mathcal{O}\{\Delta\}$ (cf. (D.65)), i.e., at least linear in the perturbation, we have

$$\hat{\boldsymbol{F}}_{\mathrm{SLS}}^{+} = \boldsymbol{F}_{\mathrm{SLS}}^{+} + \mathcal{O}\{\Delta\}. \tag{D.117}$$

This relation only describes the zero-th term of the expansion of $\hat{\boldsymbol{F}}_{\mathrm{SLS}}^{+}$. However, as we see below, the linear term is not needed for a first order expansion of $\Delta\boldsymbol{\psi}_{\mathrm{SLS}}$. Continuing with (D.114), the second term of the right-hand side is given by $\boldsymbol{r}_{\mathrm{LS}}$ for which we can write

$$\begin{aligned}
\boldsymbol{r}_{\mathrm{LS}} &= \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}} \cdot \hat{\boldsymbol{\Psi}}_{\mathrm{LS}} - \boldsymbol{J}_2 \cdot \hat{\boldsymbol{U}}_{\mathrm{s}}\right\} \\
&= \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot (\boldsymbol{U}_{\mathrm{s}} + \Delta\boldsymbol{U}_{\mathrm{s}}) \cdot (\boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}_{\mathrm{LS}}) - \boldsymbol{J}_2 \cdot (\boldsymbol{U}_{\mathrm{s}} + \Delta\boldsymbol{U}_{\mathrm{s}})\right\} \\
&= \mathrm{vec}\left\{\underbrace{\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} - \boldsymbol{J}_2 \cdot \boldsymbol{U}_{\mathrm{s}}}_{\mathbf{0}_{(M-1)\times d}} + \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi}_{\mathrm{LS}} - \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\}\right\} \\
&= \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot \Delta\boldsymbol{\Psi}_{\mathrm{LS}} - \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} + \mathcal{O}\left\{\Delta^2\right\}\right\}
\end{aligned} \tag{D.118}$$

Moreover, as shown in [LLV93] (cf. (D.65)), $\Delta\boldsymbol{\Psi}_{\mathrm{LS}}$ can be expressed in terms of $\Delta\boldsymbol{U}_{\mathrm{s}}$ via

$$\Delta\boldsymbol{\Psi}_{\mathrm{LS}} = (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} - (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi} + \mathcal{O}\left\{\Delta^2\right\}. \tag{D.119}$$

Using this expansion in (D.118) we obtain

$$\begin{aligned}
\boldsymbol{r}_{\mathrm{LS}} &= \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi}\right\} + \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}}\right\} \\
&\quad - \mathrm{vec}\left\{\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_1 \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{\Psi}\right\} - \mathrm{vec}\left\{\boldsymbol{J}_2 \cdot \Delta\boldsymbol{U}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\} \\
&= \boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\} \quad \text{where}
\end{aligned} \tag{D.120}$$
$$\boldsymbol{W}_{\mathrm{R,U}} = \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) + \boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_2\right) - \boldsymbol{\Psi}^{\mathrm{T}} \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_1\right) - (\boldsymbol{I}_d \otimes \boldsymbol{J}_2).$$

Using (D.120) and (D.117) in (D.114) we find that

$$\begin{aligned}
\begin{bmatrix} \Delta\boldsymbol{\psi}_{\mathrm{SLS}} \\ \Delta\boldsymbol{u}_{\mathrm{s,SLS}} \end{bmatrix} &= -\hat{\boldsymbol{F}}_{\mathrm{SLS}}^{+} \cdot \boldsymbol{r}_{\mathrm{LS}} = -\left(\boldsymbol{F}_{\mathrm{SLS}}^{+} + \mathcal{O}\{\Delta\}\right) \cdot \left(\boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\}\right) \\
&= -\boldsymbol{F}_{\mathrm{SLS}}^{+} \cdot \boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\}
\end{aligned} \tag{D.121}$$

Note that from (D.116) it follows that $\boldsymbol{F}_{\mathrm{SLS}}$ has full row-rank and hence its pseudo-inverse can be expressed as $\boldsymbol{F}_{\mathrm{SLS}}^{+} = \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}} \cdot (\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}})^{-1}$. This allows us to extract $\Delta\boldsymbol{\psi}_{\mathrm{SLS}}$ from (D.121) since $\Delta\boldsymbol{u}_{\mathrm{s,SLS}}$ is not explicitly needed as long as only one SLS iteration is performed. We obtain

$$\Delta\boldsymbol{\psi}_{\mathrm{SLS}} = -\left(\boldsymbol{I}_d \otimes (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}\right) \cdot (\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}})^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{D.122}$$

The final step in SLS-based ESPRIT is to replace the LS-based estimate $\hat{\boldsymbol{\Psi}}_{\mathrm{LS}}$ by $\hat{\boldsymbol{\Psi}}_{\mathrm{SLS}} = \hat{\boldsymbol{\Psi}}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}_{\mathrm{SLS}} = \boldsymbol{\Psi} + \Delta\boldsymbol{\Psi}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}_{\mathrm{SLS}}$. Following the first-order expansion for Standard ESPRIT from [LLV93] (cf. (D.67)) we obtain

$$\begin{aligned}\Delta\mu_k &= \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\Delta\boldsymbol{\Psi}_{\mathrm{LS}} + \Delta\boldsymbol{\Psi}_{\mathrm{SLS}}) \cdot \boldsymbol{q}_k\right\}/\mathrm{e}^{\jmath\mu_k} + \mathcal{O}\left\{\Delta^2\right\}\\ &= \mathrm{Im}\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot \Delta\boldsymbol{\Psi}_{\mathrm{LS}} \cdot \boldsymbol{q}_k\right\}/\mathrm{e}^{\jmath\mu_k} + \mathrm{Im}\left\{\left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \boldsymbol{p}_k^{\mathrm{T}}\right) \cdot \Delta\boldsymbol{\psi}_{\mathrm{SLS}}\right\}/\mathrm{e}^{\jmath\mu_k} + \mathcal{O}\left\{\Delta^2\right\}\end{aligned} \tag{D.123}$$

Since the first term is exactly the same as for LS-based 1-D Standard ESPRIT, we can use the result from [LLV93]. Inserting the expansion for $\Delta\boldsymbol{\psi}_{\mathrm{SLS}}$ from (D.122) we obtain

$$\begin{aligned}\Delta\mu_k = \mathrm{Im}&\left\{\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \left(\frac{\boldsymbol{J}_2}{\mathrm{e}^{\jmath\mu_k}} - \boldsymbol{J}_1\right) \cdot \Delta\boldsymbol{U}_{\mathrm{s}} \cdot \boldsymbol{q}_k\right\}\\ &- \mathrm{Im}\left\{\left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}\right) \cdot (\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}})^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\}/\mathrm{e}^{\jmath\mu_k} + \mathcal{O}\left\{\Delta^2\right\}\end{aligned} \tag{D.124}$$

Finally, rearranging the first term as $\boldsymbol{a}^{\mathrm{T}} \cdot \boldsymbol{B} \cdot \boldsymbol{c} = (\boldsymbol{c}^{\mathrm{T}} \otimes \boldsymbol{a}^{\mathrm{T}}) \cdot \mathrm{vec}\left\{\boldsymbol{B}\right\}$ we have

$$\begin{aligned}\Delta\mu_k &= \mathrm{Im}\left\{\boldsymbol{q}_k^{\mathrm{T}} \otimes \left(\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \left(\frac{\boldsymbol{J}_2}{\mathrm{e}^{\jmath\mu_k}} - \boldsymbol{J}_1\right)\right) \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\}\\ &- \mathrm{Im}\left\{\left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \boldsymbol{p}_k^{\mathrm{T}} \cdot \frac{(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}}{\mathrm{e}^{\jmath\mu_k}}\right) \cdot (\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}})^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\}\\ &= \mathrm{Im}\left\{\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}\right\} + \mathcal{O}\left\{\Delta^2\right\}, \quad \text{where}\\ \boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} &= \boldsymbol{q}_k^{\mathrm{T}} \otimes \left[\boldsymbol{p}_k^{\mathrm{T}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{U}_s)^{+} \cdot \left(\frac{\boldsymbol{J}_2}{\mathrm{e}^{\jmath\mu_k}} - \boldsymbol{J}_1\right)\right] - \left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \left[\boldsymbol{p}_k^{\mathrm{T}} \cdot \frac{(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}}{\mathrm{e}^{\jmath\mu_k}}\right]\right) \cdot \left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}},\end{aligned} \tag{D.125}$$

which is the desired result (12.38). Equation (12.39) follows from (12.38) by inserting the first order expansion for $\mathrm{vec}\left\{\Delta\boldsymbol{U}_{\mathrm{s}}\right\}$ in terms of $\mathrm{vec}\left\{\boldsymbol{N}\right\}$ as shown in Section D.12. □

D.16. Proof of Theorem 12.4.5

This theorem consists of several parts which we address in separate subsections.

D.16.1. MSE for Standard ESPRIT

We start by simplifying the MSE expression for 1-D Standard ESPRIT. In the case of a single source we can write

$$\boldsymbol{X}_0 = \boldsymbol{a}(\mu) \cdot \boldsymbol{s}^{\mathrm{T}}, \tag{D.126}$$

where $\boldsymbol{a} \in \mathbb{C}^{M \times 1}$ is the array steering vector and $\boldsymbol{s} \in \mathbb{C}^{N \times 1}$ contains the source symbols. Let $\hat{P}_{\mathrm{T}} = \|\boldsymbol{s}\|_2^2 / N$ be the empirical source power. Furthermore, since we assume a Uniform Linear Array (ULA) of isotropic elements, $\boldsymbol{a}(\mu)$ is given by $\boldsymbol{a}(\mu) = [1, \mathrm{e}^{\jmath\mu}, \mathrm{e}^{2\jmath\mu}, \ldots, \mathrm{e}^{(M-1)\jmath\mu}]$. Note that $\|\boldsymbol{a}(\mu)\|_2^2 = M$. For notational convenience, we drop the explicit dependence of $\boldsymbol{a}$ on μ and write just $\boldsymbol{a}(\mu) = \boldsymbol{a}$ in the sequel. The selection matrices $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ are then chosen as

$$\boldsymbol{J}_1 = \begin{bmatrix} \boldsymbol{I}_{M-1} & \boldsymbol{0}_{M-1 \times 1} \end{bmatrix} \quad \boldsymbol{J}_2 = \begin{bmatrix} \boldsymbol{0}_{M-1 \times 1} & \boldsymbol{I}_{M-1} \end{bmatrix} \tag{D.127}$$

for maximum overlap, i.e., $M^{(\mathrm{sel})} = M - 1$. Since (D.126) is a rank-one matrix, we can directly identify the subspaces with the array steering vector and the source symbol matrix, namely

$$\boldsymbol{U}_{\mathrm{s}} = \boldsymbol{u}_{\mathrm{s}} = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2} \cdot \mathrm{e}^{\jmath\varphi_1} = \frac{1}{\sqrt{M}} \cdot \boldsymbol{a} \cdot \mathrm{e}^{\jmath\varphi_1} \tag{D.128}$$

$$\boldsymbol{V}_{\mathrm{s}} = \boldsymbol{v}_{\mathrm{s}} = \frac{\boldsymbol{s}^*}{\|\boldsymbol{s}\|_2} \cdot \mathrm{e}^{-\jmath\varphi_1} = \frac{1}{\sqrt{\hat{P}_{\mathrm{T}} \cdot N}} \cdot \boldsymbol{s}^* \cdot \mathrm{e}^{-\jmath\varphi_1} \tag{D.129}$$

$$\boldsymbol{\Sigma}_{\mathrm{s}} = \sigma_{\mathrm{s}} = \sqrt{M \cdot N \cdot \hat{P}_{\mathrm{T}}}. \tag{D.130}$$

Here, the quantity $\mathrm{e}^{\jmath\varphi_1}$ stems from the fact that the SVD is only unique up to one arbitrary phase term per column of $\boldsymbol{U}$ if the opposite phase term is included in the corresponding column of $\boldsymbol{V}$. For the MSE expression from Theorem 12.4.1 we also require the quantity $\boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}}$, which resembles a projection matrix on the noise subspace. However, since the signal subspace is spanned by $\boldsymbol{a}$ we can write $\boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}} = \boldsymbol{I}_M - \frac{\boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}}{\|\boldsymbol{a}\|_2^2} = \boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}$. The MSE expression for 1-D Standard ESPRIT also include the eigenvectors of $\boldsymbol{\Psi}$ denoted by $\boldsymbol{p}_k$ and $\boldsymbol{q}_k$. However, for the special case discussed here, $\boldsymbol{\Psi}$ is scalar and given by $\boldsymbol{\Psi} = \boldsymbol{\Phi} = \mathrm{e}^{\jmath\mu}$. Consequently, we have $\boldsymbol{p}_k = \boldsymbol{q}_k = 1$ for the eigenvectors[1].

Combining these expressions and inserting into (12.24) we have $\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \sigma_{\mathrm{n}}^2/2 \cdot$

[1]We could account for the scaling ambiguity of the eigenvectors by setting $\boldsymbol{p}_k = \alpha$ and $\boldsymbol{q}_k = \alpha^{-1}$ for some $\alpha \in \mathbb{C}_{\neq 0}$. However, this scalar is irrelevant as it cancels immediately in the expression (12.26) for $\boldsymbol{r}_k^{(r)}$.

$\left\|\boldsymbol{W}_{\text{mat}}^{\text{T}} \cdot \boldsymbol{r}\right\|_2^2 = \sigma_{\text{n}}^2/2 \cdot \left\|\boldsymbol{r}^{\text{T}} \cdot \boldsymbol{W}_{\text{mat}}\right\|_2^2$ with

$$\boldsymbol{r}_k^{(r)} = \boldsymbol{r} = \left[\left(\boldsymbol{J}_1 \frac{\boldsymbol{a}}{\sqrt{M}} \mathrm{e}^{\jmath\varphi_1}\right)^+ \left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right)\right]^{\text{T}} \tag{D.131}$$

$$\boldsymbol{W}_{\text{mat}} = \left(\frac{1}{\sqrt{M \cdot N \cdot \hat{P}_{\text{T}}}} \cdot \frac{\boldsymbol{s}^{\text{H}}}{\sqrt{\hat{P}_{\text{T}} \cdot N}} \cdot \mathrm{e}^{-\jmath\varphi_1}\right) \otimes \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\text{H}}\right) \tag{D.132}$$

Note that $\boldsymbol{W}_{\text{mat}}$ is the Kronecker product of a $1 \times N$ vector and an $M \times M$ matrix. Hence, $\boldsymbol{r}^{\text{T}} \cdot \boldsymbol{W}_{\text{mat}}$ can be written as

$$\underbrace{\left(\frac{1}{\sqrt{M \cdot N \cdot \hat{P}_{\text{T}}}} \cdot \frac{\boldsymbol{s}^{\text{H}}}{\sqrt{\hat{P}_{\text{T}} \cdot N}} \cdot \mathrm{e}^{-\jmath\varphi_1}\right)}_{\tilde{\boldsymbol{s}}^{\text{T}}} \otimes \underbrace{\left[\left(\boldsymbol{J}_1 \frac{\boldsymbol{a}}{\sqrt{M}} \mathrm{e}^{\jmath\varphi_1}\right)^+ \left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \cdot \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\text{H}}\right)\right]}_{\tilde{\boldsymbol{a}}^{\text{T}}}. \tag{D.133}$$

Therefore, the MSE can be expressed as $\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \sigma_{\text{n}}^2/2 \cdot \left\|\tilde{\boldsymbol{s}}^{\text{T}} \otimes \tilde{\boldsymbol{a}}^{\text{T}}\right\|_2^2$, which is equal to $\mathbb{E}\left\{(\Delta\mu_k^{(r)})^2\right\} = \sigma_{\text{n}}^2/2 \cdot \left\|\tilde{\boldsymbol{s}}^{\text{T}}\right\|_2^2 \cdot \left\|\tilde{\boldsymbol{a}}^{\text{T}}\right\|_2^2$ according to Property (3.16).

Since $\tilde{\boldsymbol{s}}^{\text{T}}$ is a scaled version of $\boldsymbol{s}^{\text{H}}$ and $\left\|\boldsymbol{s}^{\text{H}}\right\|_2^2 = N \cdot \hat{P}_{\text{T}}$ we find that the first term in the MSE expression can conveniently be expressed as

$$\left\|\tilde{\boldsymbol{s}}^{\text{T}}\right\|_2^2 = \frac{1}{M \cdot N \cdot \hat{P}_{\text{T}}} \cdot \frac{\hat{P}_{\text{T}} \cdot N}{\hat{P}_{\text{T}} \cdot N} = \frac{1}{M \cdot N \cdot \hat{P}_{\text{T}}}. \tag{D.134}$$

Next, we proceed to simplify $\tilde{\boldsymbol{a}}^{\text{T}}$ further. Firstly, since only the norm of $\tilde{\boldsymbol{a}}^{\text{T}}$ matters, the phase term $\mathrm{e}^{\jmath\varphi_1}$ can be canceled. We then expand the pseudo-inverse of $\boldsymbol{J}_1 \cdot \boldsymbol{a}$ using the rule $\boldsymbol{x}^+ = \boldsymbol{x}^{\text{H}} / \left\|\boldsymbol{x}\right\|_2^2$ which yields

$$\begin{aligned}
\tilde{\boldsymbol{a}}^{\text{T}} &= \sqrt{M} \frac{\boldsymbol{a}^{\text{H}} \boldsymbol{J}_1^{\text{H}}}{\left\|\boldsymbol{J}_1 \boldsymbol{a}\right\|_2^2} \left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \cdot \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\text{H}}\right) \\
&= \sqrt{M} \frac{\boldsymbol{a}^{\text{H}}}{M-1} \boldsymbol{J}_1^{\text{H}} \cdot \left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \cdot \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\text{H}}\right),
\end{aligned} \tag{D.135}$$

where $\left\|\boldsymbol{J}_1 \boldsymbol{a}\right\|_2^2 = M - 1$ follows from the fact that all elements of $\boldsymbol{a}$ contains only phase terms. Combining the last two brackets we can rewrite this expression into

$$\tilde{\boldsymbol{a}}^{\text{T}} = \sqrt{M} \frac{\boldsymbol{a}^{\text{H}}}{M-1} \boldsymbol{J}_1^{\text{H}} \cdot \left(\left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) - \frac{1}{M} \cdot \left(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\text{H}}\right)$$

$$
\begin{aligned}
&= \frac{\sqrt{M}}{M-1} \boldsymbol{a}^{\mathrm{H}} \boldsymbol{J}_1^{\mathrm{H}} \cdot \left((\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) - \frac{1}{M} \cdot \underbrace{(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} \cdot \boldsymbol{a} - \boldsymbol{J}_1 \cdot \boldsymbol{a})}_{\mathbf{0}} \cdot \boldsymbol{a}^{\mathrm{H}} \right) \\
&= \frac{\sqrt{M}}{M-1} \boldsymbol{a}^{\mathrm{H}} \boldsymbol{J}_1^{\mathrm{H}} \cdot (\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) \\
&= \frac{\sqrt{M}}{M-1} \Big(\underbrace{\boldsymbol{a}^{\mathrm{H}} \boldsymbol{J}_1^{\mathrm{H}} \cdot \boldsymbol{J}_2/\mathrm{e}^{\jmath\mu}}_{\tilde{\boldsymbol{a}}_1^{\mathrm{T}}} - \underbrace{\boldsymbol{a}^{\mathrm{H}} \boldsymbol{J}_1^{\mathrm{H}} \cdot \boldsymbol{J}_1}_{\tilde{\boldsymbol{a}}_2^{\mathrm{T}}} \Big)
\end{aligned} \tag{D.136}
$$

Note that in the second step we have used the fact that $\boldsymbol{a}$ satisfies the shift invariance equation, i.e., $\boldsymbol{J}_1 \cdot \boldsymbol{a} \cdot \mathrm{e}^{\jmath\mu} = \boldsymbol{J}_2 \cdot \boldsymbol{a}$, which can be rewritten as $\boldsymbol{J}_2 \cdot \boldsymbol{a}/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1 \cdot \boldsymbol{a} = \mathbf{0}$. Since we have $\boldsymbol{a}^{\mathrm{H}} = [1, \mathrm{e}^{-\jmath\mu}, \mathrm{e}^{-2\jmath\mu}, \ldots, \mathrm{e}^{-(M-1)\jmath\mu}]$ it is easy to see that

$$
\begin{aligned}
\tilde{\boldsymbol{a}}_1^{\mathrm{T}} &= [0, 1, \mathrm{e}^{-\jmath\mu}, \mathrm{e}^{-2\jmath\mu}, \ldots, \mathrm{e}^{-(M-3)\jmath\mu}, \mathrm{e}^{-(M-2)\jmath\mu}]/\mathrm{e}^{\jmath\mu} \\
&= [0, \mathrm{e}^{-\jmath\mu}, \mathrm{e}^{-2\jmath\mu}, \ldots, \mathrm{e}^{-(M-2)\jmath\mu}, \mathrm{e}^{-(M-1)\jmath\mu}] \\
\tilde{\boldsymbol{a}}_2^{\mathrm{T}} &= [1, \mathrm{e}^{-\jmath\mu}, \mathrm{e}^{-2\jmath\mu}, \ldots, \mathrm{e}^{-(M-2)\jmath\mu}, 0] \quad \text{and hence} \\
\tilde{\boldsymbol{a}}_1^{\mathrm{T}} - \tilde{\boldsymbol{a}}_2^{\mathrm{T}} &= [-1, 0, \ldots, 0, \mathrm{e}^{-(M-1)\jmath\mu}]
\end{aligned} \tag{D.137}
$$

Consequently, we find $\left\|\tilde{\boldsymbol{a}}^{\mathrm{T}}\right\|_2^2 = \frac{M}{(M-1)^2} \cdot 2$. Combining this result with (D.134) we finally have

$$
\mathbb{E}\left\{ (\Delta\mu_k^{(r)})^2 \right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\|\tilde{\boldsymbol{s}}^{\mathrm{T}}\right\|_2^2 \cdot \left\|\tilde{\boldsymbol{a}}^{\mathrm{T}}\right\|_2^2 = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \frac{1}{M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot 2 \cdot \frac{M}{(M-1)^2} \tag{D.138}
$$

$$
= \frac{\sigma_{\mathrm{n}}^2}{N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{(M-1)^2} \tag{D.139}
$$

which is the desired result. □

D.16.2. MSE for Unitary ESPRIT

The second part of the theorem is to show that for a single source, the MSE for Unitary ESPRIT is the same as the MSE for Standard ESPRIT. Firstly, we expand $\boldsymbol{X}_0^{(\mathrm{fba})}$ and find

$$
\boldsymbol{X}_0^{(\mathrm{fba})} = \begin{bmatrix} \boldsymbol{a} \cdot \boldsymbol{s}^{\mathrm{T}} & \boldsymbol{\Pi}_M \cdot \boldsymbol{a}^* \cdot \boldsymbol{s}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_N \end{bmatrix} = \boldsymbol{a} \cdot \begin{bmatrix} \boldsymbol{s}^{\mathrm{T}} & \mathrm{e}^{-\jmath \cdot \mu \cdot (M-1)} \cdot \boldsymbol{s}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_N \end{bmatrix} = \boldsymbol{a} \cdot \overline{\boldsymbol{s}}^{\mathrm{T}} \tag{D.140}
$$

where we have used the fact that for a ULA we have $\boldsymbol{\Pi}_M \cdot \boldsymbol{a}^* = \boldsymbol{a} \cdot \mathrm{e}^{-\jmath \cdot \mu \cdot m_{\mathrm{c}}}$, where m_c depends on the choice of the phase center. If it is chosen in the middle of the array we have $m_{\mathrm{c}} = 0$, if it is chosen at the first element as in (9.19) we have $m_{\mathrm{c}} = M - 1$. The particular choice of the phase center is irrelevant and it is not difficult to show that it also cancels in the subsequent derivation. However, since the derivation is significantly more complicated to present with an arbitrary choice of the phase center we stick to (9.19) for simplicity, and hence we set $m_{\rm c} = M - 1$. Moreover, we have defined $\overline{\boldsymbol{s}}$ to be

$$\overline{\boldsymbol{s}} = \begin{bmatrix} \boldsymbol{s} \\ {\rm e}^{-\jmath \cdot \mu \cdot (M-1)} \cdot \boldsymbol{\Pi}_N \cdot \boldsymbol{s}^* \end{bmatrix} \tag{D.141}$$

Note that $\overline{\boldsymbol{s}}^{\rm H} \overline{\boldsymbol{s}} = \boldsymbol{s}^{\rm H} \cdot \boldsymbol{s} + \boldsymbol{s}^{\rm T} \cdot \boldsymbol{\Pi}_N \cdot \boldsymbol{\Pi}_N \cdot \boldsymbol{s}^* = 2 \cdot \boldsymbol{s}^{\rm H} \cdot \boldsymbol{s}$. As for Standard ESPRIT, we relate (D.140) to its SVD and obtain

$$\boldsymbol{u}_{\rm s}^{\rm (fba)} = \frac{\boldsymbol{a}}{\sqrt{M}} = \boldsymbol{u}_{\rm s} \cdot {\rm e}^{\jmath \varphi_1}, \quad \boldsymbol{v}_{\rm s}^{\rm (fba)} = \frac{\overline{\boldsymbol{s}}^*}{\sqrt{2 \cdot N \cdot \hat{P}_{\rm T}}} \cdot {\rm e}^{-\jmath \varphi_1}, \quad \sigma_{\rm s}^{\rm (fba)} = \sqrt{2 \cdot M \cdot N \cdot \hat{P}_{\rm T}} \tag{D.142}$$

An important consequence we can draw from (D.142) is that the column space $\boldsymbol{u}_{\rm s}$ remains unaffected from the forward-backward-averaging. Therefore we also have $\boldsymbol{U}_{\rm n}^{\rm (fba)} = \boldsymbol{U}_{\rm n}$ and hence $\boldsymbol{U}_{\rm n}^{\rm (fba)} \boldsymbol{U}_{\rm n}^{\rm (fba)^H} = \boldsymbol{I}_M - \frac{\boldsymbol{a} \cdot \boldsymbol{a}^{\rm H}}{M}$. Following the lines of the derivation for 1-D Standard ESPRIT we can show that

$$\boldsymbol{r}^{\rm (fba)^T} \cdot \boldsymbol{W}_{\rm mat}^{\rm (fba)} = \tilde{\overline{\boldsymbol{s}}}^{\rm T} \otimes \tilde{\boldsymbol{a}}^{\rm T}, \tag{D.143}$$

where $\tilde{\overline{\boldsymbol{s}}}$ is given by

$$\tilde{\overline{\boldsymbol{s}}} = \frac{1}{\sqrt{2 \cdot M \cdot N \cdot \hat{P}_{\rm T}}} \cdot \frac{\overline{\boldsymbol{s}}^*}{\sqrt{2 \cdot N \cdot \hat{P}_{\rm T}}} \cdot {\rm e}^{-\jmath \varphi_1} \tag{D.144}$$

and $\tilde{\boldsymbol{a}}$ is the same as in the derivation for 1-D Standard ESPRIT (cf. equation (D.133)).

According to Theorem 12.4.3, the MSE for Unitary ESPRIT can be computed as

$$\frac{\sigma_{\rm n}^2}{2} \cdot \left(\boldsymbol{r}^{\rm (fba)^T} \cdot \boldsymbol{W}_{\rm mat}^{\rm (fba)} \cdot \left[\boldsymbol{r}^{\rm (fba)^T} \cdot \boldsymbol{W}_{\rm mat}^{\rm (fba)} \right]^{\rm H} - {\rm Re} \left\{ \boldsymbol{r}^{\rm (fba)^T} \cdot \boldsymbol{W}_{\rm mat}^{\rm (fba)} \cdot \boldsymbol{\Pi}_{2MN} \cdot \left[\boldsymbol{r}^{\rm (fba)^T} \cdot \boldsymbol{W}_{\rm mat}^{\rm (fba)} \right]^{\rm T} \right\} \right).$$

Using (D.143) and the fact that $\boldsymbol{\Pi}_{2MN} = \boldsymbol{\Pi}_{2N} \otimes \boldsymbol{\Pi}_M$, this expression can be written into

$$\begin{aligned} & \frac{\sigma_{\rm n}^2}{2} \cdot \left(\tilde{\overline{\boldsymbol{s}}}^{\rm T} \cdot \tilde{\overline{\boldsymbol{s}}}^* \cdot \tilde{\boldsymbol{a}}^{\rm T} \cdot \tilde{\boldsymbol{a}}^* - \tilde{\overline{\boldsymbol{s}}}^{\rm T} \cdot \boldsymbol{\Pi}_{2N} \cdot \tilde{\overline{\boldsymbol{s}}} \cdot \tilde{\boldsymbol{a}}^{\rm T} \cdot \boldsymbol{\Pi}_M \cdot \tilde{\boldsymbol{a}} \right) \\ = & \frac{\sigma_{\rm n}^2}{2} \cdot \left(\left\| \tilde{\overline{\boldsymbol{s}}} \right\|_2^2 \left\| \tilde{\boldsymbol{a}} \right\|_2^2 - \tilde{\overline{\boldsymbol{s}}}^{\rm T} \cdot \boldsymbol{\Pi}_{2N} \cdot \tilde{\overline{\boldsymbol{s}}} \cdot \tilde{\boldsymbol{a}}^{\rm T} \cdot \boldsymbol{\Pi}_M \cdot \tilde{\boldsymbol{a}} \right) \end{aligned} \tag{D.145}$$

Since $\tilde{\boldsymbol{a}}$ is the same as in (D.136) we know that $\|\tilde{\boldsymbol{a}}\|_2^2 = 2 \cdot \frac{M}{(M-1)^2}$. Moreover, $\left\| \tilde{\overline{\boldsymbol{s}}} \right\|_2^2 = \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\rm T}}$

follows directly from (D.144). For the second term in (D.145) we have

$$
\begin{aligned}
&\tilde{\bar{\boldsymbol{s}}}^{\mathrm{T}} \cdot \boldsymbol{\Pi}_{2N} \cdot \tilde{\bar{\boldsymbol{s}}} = \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \bar{\boldsymbol{s}}^{\mathrm{H}} \cdot \boldsymbol{\Pi}_{2N} \cdot \bar{\boldsymbol{s}}^{*} \cdot \mathrm{e}^{-2\jmath\varphi} \\
&= \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \left(\boldsymbol{s}^{\mathrm{H}} \cdot \boldsymbol{s} \cdot \mathrm{e}^{\jmath\mu(M-1)} + \boldsymbol{s}^{\mathrm{T}} \cdot \boldsymbol{\Pi}_N \cdot \boldsymbol{\Pi}_N \cdot \boldsymbol{s}^{*} \cdot \mathrm{e}^{\jmath\mu(M-1)} \right) \cdot \mathrm{e}^{-2\jmath\varphi} \\
&= \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \left(N \cdot \hat{P}_{\mathrm{T}} + N \cdot \hat{P}_{\mathrm{T}} \right) \cdot \mathrm{e}^{\jmath\mu(M-1)} \cdot \mathrm{e}^{-2\jmath\varphi} \\
&= \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \mathrm{e}^{\jmath\mu(M-1)} \cdot \mathrm{e}^{-2\jmath\varphi}
\end{aligned} \tag{D.146}
$$

Similarly we can simplify $\tilde{\boldsymbol{a}}^{\mathrm{T}} \cdot \boldsymbol{\Pi}_M \cdot \tilde{\boldsymbol{a}}$ by using (D.136) and (D.137) into

$$
\begin{aligned}
\tilde{\boldsymbol{a}}^{\mathrm{T}} \cdot \boldsymbol{\Pi}_M \cdot \tilde{\boldsymbol{a}} &= \frac{M}{(M-1)^2} \cdot \left[-1, 0, \ldots, 0, \mathrm{e}^{-(M-1)\jmath\mu}\right] \cdot \boldsymbol{\Pi}_M \cdot \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ \mathrm{e}^{-(M-1)\jmath\mu} \end{bmatrix} \cdot \mathrm{e}^{2\jmath\varphi} \\
&= \frac{M}{(M-1)^2} \cdot \left(-\mathrm{e}^{-(M-1)\jmath\mu} - \mathrm{e}^{-(M-1)\jmath\mu} \right) \cdot \mathrm{e}^{2\jmath\varphi} \\
&= -2 \cdot \frac{M}{(M-1)^2} \cdot \mathrm{e}^{-(M-1)\jmath\mu} \cdot \mathrm{e}^{2\jmath\varphi}.
\end{aligned} \tag{D.147}
$$

Combining the results from (D.146) and (D.147) into (D.145) we finally obtain for the MSE

$$
\begin{aligned}
&\frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left(\frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot 2 \cdot \frac{M}{(M-1)^2} + \frac{1}{2 \cdot M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \mathrm{e}^{\jmath\mu(M-1)} \cdot \mathrm{e}^{-2\jmath\varphi} \cdot 2 \cdot \frac{M}{(M-1)^2} \cdot \mathrm{e}^{-(M-1)\jmath\mu} \cdot \mathrm{e}^{2\jmath\varphi} \right) \\
&= \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left(\frac{1}{N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{(M-1)^2} + \frac{1}{N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{(M-1)^2} \cdot \right) \\
&= \frac{\sigma_{\mathrm{n}}^2}{N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{1}{(M-1)^2}
\end{aligned} \tag{D.148}
$$

which is equal to the result for 1-D Standard ESPRIT from (D.139) and hence proves this part of the theorem. □

D.16.3. Cramér-Rao Bound

The third part of the theorem is to simplify the deterministic Cramér-Rao Bound (CRB) for the special case of a single source. To this end, a closed-form expression for the deterministic

CRB for this setting is given by [SN89]

$$\boldsymbol{C} = \frac{\sigma_\mathrm{n}^2}{2 \cdot N} \cdot \mathrm{Re}\left\{\left[\boldsymbol{D}^\mathrm{H} \cdot \left(\boldsymbol{I}_M - \boldsymbol{A} \cdot \left(\boldsymbol{A}^\mathrm{H} \cdot \boldsymbol{A}\right)^{-1} \cdot \boldsymbol{A}^\mathrm{H}\right) \cdot \boldsymbol{D}\right] \odot \hat{\boldsymbol{R}}_\mathrm{S}^\mathrm{T}\right\}^{-1} \tag{D.149}$$

where $\hat{\boldsymbol{R}}_\mathrm{S} = \frac{1}{N} \cdot \boldsymbol{S} \cdot \boldsymbol{S}^\mathrm{H}$ is the sample covariance matrix of the source symbols and $\boldsymbol{D} \in \mathbb{C}^{M \times d}$ is the matrix of partial derivatives of the array steering vectors with respect to the parameters of interest. In the case $d = 1$, we have $\hat{\boldsymbol{R}}_\mathrm{S} = \|\boldsymbol{s}\|_2^2 / N = \hat{P}_\mathrm{T}$ and the CRB expression simplifies into

$$\begin{aligned} C &= \frac{\sigma_\mathrm{n}^2}{2 \cdot N \cdot \hat{P}_\mathrm{T}} \cdot \mathrm{Re}\left\{\boldsymbol{d}^\mathrm{H} \cdot \left(\boldsymbol{I}_M - \frac{\boldsymbol{a} \cdot \boldsymbol{a}^\mathrm{H}}{M}\right) \cdot \boldsymbol{d}\right\}^{-1} \\ &= \frac{1}{2 \cdot \hat{\rho}} \cdot \mathrm{Re}\left\{\boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{d} - \frac{1}{M} \cdot \boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^\mathrm{H} \cdot \boldsymbol{d}\right\}^{-1} \\ &= \frac{1}{2 \cdot \hat{\rho}} \cdot \left[\boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{d} - \frac{1}{M} \cdot |\boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{a}|^2\right]^{-1} \end{aligned} \tag{D.150}$$

Since for a ULA the array steering vector can be expressed as $\boldsymbol{a} = \begin{bmatrix} 1 & \mathrm{e}^{\jmath\mu} & \mathrm{e}^{2\jmath\mu} & \dots & \mathrm{e}^{(M-1)\jmath\mu} \end{bmatrix}$ we have

$$\boldsymbol{d} = \frac{\partial \boldsymbol{a}}{\partial \mu} = \jmath \cdot \begin{bmatrix} 0 & \mathrm{e}^{\jmath\mu} & 2 \cdot \mathrm{e}^{2\jmath\mu} & \dots & (M-1) \cdot \mathrm{e}^{(M-1)\jmath\mu} \end{bmatrix}. \tag{D.151}$$

Consequently the terms $\boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{d}$ and $\boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{a}$ become

$$\begin{aligned} \boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{d} &= \sum_{m=0}^{M-1} m^2 = \frac{1}{6} \cdot (M-1) \cdot M \cdot (2M-1) \\ \boldsymbol{d}^\mathrm{H} \cdot \boldsymbol{a} &= -\jmath \sum_{m=0}^{M-1} m = -\jmath \cdot \frac{1}{2} \cdot (M-1) \cdot M \end{aligned} \tag{D.152}$$

Using these expressions in (D.150), we obtain

$$\begin{aligned} C &= \frac{1}{2 \cdot \hat{\rho}} \cdot \left[\frac{1}{6} \cdot (M-1) \cdot M \cdot (2M-1) - \frac{1}{M} \cdot \left|-\jmath \cdot \frac{1}{2} \cdot (M-1) \cdot M\right|^2\right]^{-1} \\ &= \frac{1}{2 \cdot \hat{\rho}} \cdot \left[\frac{1}{6} \cdot (M-1) \cdot M \cdot (2M-1) - \frac{1}{4} \cdot (M-1)^2 \cdot M\right]^{-1} \\ &= \frac{1}{2 \cdot \hat{\rho}} \cdot \left[\frac{1}{12} \cdot (M-1) \cdot M \cdot \left(2 \cdot (2M-1) - 3 \cdot (M-1)\right)\right]^{-1} \\ &= \frac{1}{2 \cdot \hat{\rho}} \cdot \left[\frac{1}{12} \cdot (M-1) \cdot M \cdot (M+1)\right]^{-1} \end{aligned}$$

$$= \frac{1}{\hat{\rho}} \cdot \frac{6}{(M-1) \cdot M \cdot (M+1)} \tag{D.153}$$

which is the desired result. □

D.17. Proof of Theorem 12.4.6

The first order expansion for the estimation error of the k-th spatial frequency provided in (12.38) can be expressed as

$$\Delta\mu_{k,\mathrm{SLS}} = \mathrm{Im}\left\{(\boldsymbol{r}_{k,\mathrm{LS}} - \Delta\boldsymbol{r}_{k,\mathrm{SLS}})^{\mathrm{T}} \cdot \boldsymbol{W}_{\mathrm{mat}} \cdot \mathrm{vec}\{\boldsymbol{N}\}\right\} + \mathcal{O}\left\{\Delta^2\right\} \tag{D.154}$$

where $\Delta\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}}$ is given by

$$\begin{aligned}
\Delta\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} &= \left(\boldsymbol{q}_k^{\mathrm{T}} \otimes \left[\boldsymbol{p}_k^{\mathrm{T}} \cdot \frac{(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{\mathrm{H}}}{\mathrm{e}^{\jmath\mu_k}}\right]\right) \cdot \left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}} \\
\boldsymbol{W}_{\mathrm{R,U}} &= \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) + \boldsymbol{I}_d \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_2\right) - \boldsymbol{\Psi}^{\mathrm{T}} \otimes \left(\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}} (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}})^{+} \cdot \boldsymbol{J}_1\right) - (\boldsymbol{I}_d \otimes \boldsymbol{J}_2) \\
\boldsymbol{F}_{\mathrm{SLS}} &= \left[\boldsymbol{I}_d \otimes (\boldsymbol{J}_1 \cdot \boldsymbol{U}_{\mathrm{s}}),\ \left(\boldsymbol{\Psi}^{\mathrm{T}} \otimes \boldsymbol{J}_1\right) - (\boldsymbol{I}_d \otimes \boldsymbol{J}_2)\right]
\end{aligned}$$

For a single source, we have $\boldsymbol{p}_k = \boldsymbol{q}_k = 1$, $\boldsymbol{\Psi} = \Psi = \mathrm{e}^{\jmath\mu}$, and $\boldsymbol{U}_{\mathrm{s}} = \boldsymbol{a}/\sqrt{M}$, and therefore $\Delta\boldsymbol{r}_{k,\mathrm{SLS}}^{\mathrm{T}} = \Delta\boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}}$ simplifies to

$$\begin{aligned}
\Delta\boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}} &= \frac{(\boldsymbol{J}_1 \cdot \boldsymbol{a})^{\mathrm{H}}}{\sqrt{M} \cdot \mathrm{e}^{\jmath\mu}} \cdot \left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1} \cdot \boldsymbol{W}_{\mathrm{R,U}} \\
\boldsymbol{W}_{\mathrm{R,U}} &= (\mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1) + \left(\boldsymbol{J}_1 \cdot \boldsymbol{a} (\boldsymbol{J}_1 \cdot \boldsymbol{a})^{+} \cdot \boldsymbol{J}_2\right) - \mathrm{e}^{\jmath\mu} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a} (\boldsymbol{J}_1 \cdot \boldsymbol{a})^{+} \cdot \boldsymbol{J}_1\right) - \boldsymbol{J}_2 \\
\boldsymbol{F}_{\mathrm{SLS}} &= \left[\boldsymbol{J}_1 \cdot \frac{\boldsymbol{a}}{\sqrt{M}},\ \mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1 - \boldsymbol{J}_2\right]
\end{aligned} \tag{D.155}$$

We can write $\boldsymbol{W}_{\mathrm{R,U}}$ as

$$\boldsymbol{W}_{\mathrm{R,U}} = \left(\boldsymbol{J}_1 \cdot \boldsymbol{a} (\boldsymbol{J}_1 \cdot \boldsymbol{a})^{+} - \boldsymbol{I}_{M-1}\right) \cdot (\boldsymbol{J}_2 - \mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1) \tag{D.156}$$

Moreover, we need to simplify the term $\left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1}$. To this end, note that $\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}$ can be written into

$$\begin{aligned}
\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}} &= \frac{1}{M} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}} \cdot \boldsymbol{J}_1^{\mathrm{H}} + (\mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1 - \boldsymbol{J}_2) \cdot (\mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1 - \boldsymbol{J}_2)^{\mathrm{H}} \\
&= \frac{1}{M} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}} \cdot \boldsymbol{J}_1^{\mathrm{H}} + 2 \cdot \boldsymbol{I}_{M-1} - \boldsymbol{J}_1 \cdot \boldsymbol{J}_2^{\mathrm{H}} \cdot \mathrm{e}^{\jmath\mu} - \boldsymbol{J}_2 \cdot \boldsymbol{J}_1^{\mathrm{H}} \cdot \mathrm{e}^{-\jmath\mu}
\end{aligned}$$

$$= \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\} \cdot \left(\frac{1}{M} \cdot \mathbf{1}_{(M-1)\times(M-1)} + 2 \cdot \boldsymbol{I}_{M-1} - \boldsymbol{J}_1 \cdot \boldsymbol{J}_2^{\mathrm{H}} - \boldsymbol{J}_2 \cdot \boldsymbol{J}_1^{\mathrm{H}}\right) \cdot \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \tag{D.157}$$

In the last step we have applied the identity $\boldsymbol{x} = \boldsymbol{x} \odot \mathbf{1}_N = \operatorname{diag}\{\boldsymbol{x}\} \cdot \mathbf{1}_N,\ \forall \boldsymbol{x} \in \mathbb{C}^{N\times 1}$ to simplify the first term. For the second term we have used the fact that the vector $\boldsymbol{a}$ contains only phase term so that $|a_m|^2 = 1\ \forall m = 1, 2, \ldots, M$, where a_m is the m-th element of the vector $\boldsymbol{a}$. Finally, the third and the fourth term follows because $\boldsymbol{a}$ satisfies the shift invariance equation and therefore $a_m = a_{m-1} \cdot \mathrm{e}^{\jmath\mu}\ \forall m = 2, 3, \ldots, M$. Equation (D.157) shows that the inverse of $\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}$ can be expressed as

$$\begin{aligned}\left(\boldsymbol{F}_{\mathrm{SLS}} \cdot \boldsymbol{F}_{\mathrm{SLS}}^{\mathrm{H}}\right)^{-1} &= \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\} \cdot \boldsymbol{G}^{-1} \cdot \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}}, \quad \text{where} \\ \boldsymbol{G} &= \frac{1}{M} \cdot \mathbf{1}_{(M-1)\times(M-1)} + 2 \cdot \boldsymbol{I}_{M-1} - \boldsymbol{J}_1 \cdot \boldsymbol{J}_2^{\mathrm{H}} - \boldsymbol{J}_2 \cdot \boldsymbol{J}_1^{\mathrm{H}}\end{aligned} \tag{D.158}$$

since $\left(\operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}\right)^{-1} = \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}}$. To proceed further, we require the following lemma:

Lemma D.17.1. *The inverse of the matrix* $\boldsymbol{G} = \frac{1}{M} \cdot \mathbf{1}_{(M-1)\times(M-1)} + 2 \cdot \boldsymbol{I}_{M-1} - \boldsymbol{J}_1 \cdot \boldsymbol{J}_2^{\mathrm{H}} - \boldsymbol{J}_2 \cdot \boldsymbol{J}_1^{\mathrm{H}}$ *is given by the following expression*

$$\left[\boldsymbol{G}^{-1}\right]_{(m_1,m_2)} = \begin{cases} \frac{1}{M} \cdot \left((M - m_1) \cdot m_2 - 3 \cdot \frac{m_1 \cdot (M-m_1) \cdot m_2 \cdot (M-m_2)}{M^2+11}\right) & m_1 \geq m_2 \\ \frac{1}{M} \cdot \left(m_1 \cdot (M - m_2) - 3 \cdot \frac{m_1 \cdot (M-m_1) \cdot m_2 \cdot (M-m_2)}{M^2+11}\right) & m_1 < m_2 \end{cases} \tag{D.159}$$
$$m_1, m_2 = 1, 2, \ldots, M-1$$

Proof. This lemma can be proved in different ways. One way is to multiply (D.159) with $\boldsymbol{G}$ and show that the resulting matrix is equal to an identity matrix. A more constructive proof is to find the inverse of $\boldsymbol{G}$ by computing the inverse of the tri-diagonal matrix $\boldsymbol{G}_0 = 2 \cdot \boldsymbol{I}_{M-1} - \boldsymbol{J}_1 \cdot \boldsymbol{J}_2^{\mathrm{H}} - \boldsymbol{J}_2 \cdot \boldsymbol{J}_1^{\mathrm{H}}$ (which is equal to 2 on its main diagonal and -1 on its first upper and its first lower off-diagonal, respectively) first. It is easy to show that the (m_1, m_2)-element of $\boldsymbol{G}_0^{-1}$ is equal to $\frac{1}{M} \cdot (M - m_1) \cdot m_2$ for $m_1 \geq m_2$ and $\frac{1}{M} \cdot m_1 \cdot (M - m_2)$ for $m_1 < m_2$. Then, we can compute the inverse of $\boldsymbol{G}$ by applying the matrix inversion lemma since $\boldsymbol{G} = \boldsymbol{G}_0 + \frac{1}{M} \cdot \mathbf{1}_{(M-1)\times 1} \cdot \mathbf{1}_{1\times(M-1)}$ and therefore $\boldsymbol{G}^{-1} = \boldsymbol{G}_0^{-1} - \frac{\boldsymbol{G}_0^{-1} \cdot \mathbf{1}_{(M-1)\times 1} \cdot \frac{1}{M} \cdot \mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}_0^{-1}}{1 + \frac{1}{M} \cdot \mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}_0^{-1} \cdot \mathbf{1}_{(M-1)\times 1}}$. This expression involves the column sum of $\boldsymbol{G}_0^{-1}$, i.e., the vector $\boldsymbol{G}_0^{-1} \cdot \mathbf{1}_{(M-1)\times 1}$ which can be shown to be equal to $\left[\frac{1}{2} \cdot m \cdot (M - m)\right]_{m=1,2,\ldots,M}$. Here, the notation $[a_i]_{i=1,2,\ldots,I}$ refers to an $I \times 1$ column vector $\boldsymbol{a}$ with i-th element a_i. The denominator of $\boldsymbol{G}^{-1}$ therefore becomes $1 + \frac{1}{M} \sum_{m=1}^{M-1} \frac{1}{2} \cdot m \cdot (M - m)$ which is equal to $1 + \frac{1}{2 \cdot M} \frac{1}{6}(M-1)M(M+1) = \frac{1}{12}(12 + M^2 - 1) = \frac{1}{12}(M^2 + 11)$. $\square$

Collecting our intermediate results from (D.155), (D.157), and (D.156) we have for $\Delta \boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}}$

$$\begin{aligned}
\Delta \boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}} &= \frac{(\boldsymbol{J}_1 \boldsymbol{a})^{\mathrm{H}}}{\sqrt{M} \cdot \mathrm{e}^{\jmath\mu}} \cdot \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\} \cdot \boldsymbol{G}^{-1} \cdot \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a} \left(\boldsymbol{J}_1 \cdot \boldsymbol{a}\right)^{+} - \boldsymbol{I}_{M-1}\right) \cdot \left(\boldsymbol{J}_2 - \mathrm{e}^{\jmath\mu} \cdot \boldsymbol{J}_1\right) \\
&= \frac{\mathbf{1}_{1\times(M-1)}}{\sqrt{M}} \cdot \boldsymbol{G}^{-1} \cdot \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a} \left(\boldsymbol{J}_1 \cdot \boldsymbol{a}\right)^{+} - \boldsymbol{I}_{M-1}\right) \cdot \left(\boldsymbol{J}_2 / \mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \\
&= \frac{\mathbf{1}_{1\times(M-1)}}{\sqrt{M}} \cdot \boldsymbol{G}^{-1} \cdot \Big(\underbrace{\operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{a}}_{\mathbf{1}_{(M-1)\times 1}} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a}\right)^{+} \cdot \left(\boldsymbol{J}_2 / \mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \\
&\quad - \left(\operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \cdot \boldsymbol{J}_2 / \mathrm{e}^{\jmath\mu} - \operatorname{diag}\{\boldsymbol{J}_1 \cdot \boldsymbol{a}\}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \right) \Big) \\
&= \frac{\mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}^{-1}}{\sqrt{M}} \cdot \left(\mathbf{1}_{(M-1)\times 1} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a}\right)^{+} \cdot \left(\boldsymbol{J}_2 / \mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) - \left(\boldsymbol{J}_2 \cdot \operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}} - \boldsymbol{J}_1 \cdot \operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}} \right) \right) \\
&= \frac{1}{\sqrt{M}} \underbrace{\mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}^{-1} \cdot \mathbf{1}_{(M-1)\times 1}}_{\gamma(M)} \cdot \left(\boldsymbol{J}_1 \cdot \boldsymbol{a}\right)^{+} \cdot \left(\boldsymbol{J}_2 / \mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1\right) \\
&\quad - \frac{\mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}^{-1}}{\sqrt{M}} \cdot \left(\boldsymbol{J}_2 - \boldsymbol{J}_1\right) \cdot \operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}} \qquad \text{(D.160)}
\end{aligned}$$

Consequently, the matrix $\boldsymbol{G}^{-1}$ enters this vector only via its row-sum $\boldsymbol{g}^{\mathrm{T}} = \mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}^{-1}$. Summing (D.159) over m_1 we obtain

$$\begin{aligned}
\boldsymbol{g}^{\mathrm{T}} = \mathbf{1}_{1\times(M-1)} \cdot \boldsymbol{G}^{-1} &= \left[\frac{1}{2} \cdot m \cdot (M-m) - \frac{3}{M} \cdot \frac{1}{6} (M-1) M (M+1) \cdot \frac{m \cdot (M-m)}{M^2+11} \right]_{m=1,2,\ldots,M-1}^{\mathrm{T}} \\
&= \left[m \cdot (M-m) \right]_{m=1,2,\ldots,M-1}^{\mathrm{T}} \cdot \frac{1}{2} \cdot \left(1 - \frac{M^2-1}{M^2+11} \right) \\
&= \left[m \cdot (M-m) \right]_{m=1,2,\ldots,M-1}^{\mathrm{T}} \cdot \frac{1}{2} \cdot \frac{M^2+11-M^2+1}{M^2+11} \\
&= \left[m \cdot (M-m) \right]_{m=1,2,\ldots,M-1}^{\mathrm{T}} \cdot \frac{6}{M^2+11}, \qquad \text{(D.161)}
\end{aligned}$$

We can find the scalar $\gamma(M)$ in (D.160) by summing the elements of $\boldsymbol{g}^{\mathrm{T}}$. We obtain

$$\gamma(M) = \frac{6}{M^2+11} \sum_{m=1}^{M-1} m \cdot (M-m) = \frac{6}{M^2+11} \cdot \frac{1}{6} \cdot (M-1) M (M+1) = \frac{(M-1)M(M+1)}{M^2+11}$$

For the second term in (D.160) we need to simplify $\boldsymbol{g}^{\mathrm{T}} \cdot (\boldsymbol{J}_2 - \boldsymbol{J}_1)$. Note that the matrix $\boldsymbol{J}_2 - \boldsymbol{J}_1$

has the following form

$$\boldsymbol{J}_2 - \boldsymbol{J}_1 = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} \tag{D.162}$$

Therefore, the multiplication with $\boldsymbol{J}_2 - \boldsymbol{J}_1$ leads to a differentiation, i.e.,

$$\boldsymbol{g}^{\mathrm{T}} \cdot (\boldsymbol{J}_2 - \boldsymbol{J}_1) = \boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}} = [-g_1,\ g_1 - g_2,\ g_2 - g_3, \dots, g_{M-2} - g_{M-1},\ g_{M-1}] \tag{D.163}$$

where g_m is the m-th element of $\boldsymbol{g}^{\mathrm{T}}$. Using (D.161) we can compute the m-th element of the vector $\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}$, which we denote as $g_{\mathrm{D},m}$ as

$$\begin{aligned} g_{\mathrm{D},m} &= \frac{6}{M^2+11} \cdot \begin{cases} -M+1 & m = 1 \\ 2m - M - 1 & m = 2, 3, \dots, M-1 \\ M - 1 & m = M \end{cases} \\ &= \frac{6}{M^2+11} \cdot (2m - M - 1), \quad m = 1, 2, \dots, M \end{aligned} \tag{D.164}$$

Collecting our intermediate results, we have shown that $\boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}}$ can be written as

$$\begin{aligned} \boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}} &= \boldsymbol{r}_{\mathrm{LS}}^{\mathrm{T}} - \Delta\boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}} \\ &= \left(\frac{\boldsymbol{J}_1 \cdot \boldsymbol{a}}{\sqrt{M}}\right)^{+} \cdot (\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) - \frac{\gamma(M)}{\sqrt{M}} \cdot (\boldsymbol{J}_1 \cdot \boldsymbol{a})^{+} \cdot (\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) + \frac{1}{\sqrt{M}} \cdot \boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}} \cdot \mathrm{diag}\{\boldsymbol{a}\}^{\mathrm{H}} \\ &= \sqrt{M} \cdot \left(\left(1 - \frac{\gamma(M)}{M}\right) \cdot (\boldsymbol{J}_1\boldsymbol{a})^{+} \cdot (\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) + \frac{1}{M} \cdot \boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}} \cdot \mathrm{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\right) \end{aligned} \tag{D.165}$$

where $\boldsymbol{r}_{\mathrm{LS}}$ has been taken from (D.131). The next step to computing the mean square error is to calculate the vector $\boldsymbol{W}_{\mathrm{mat}}^{\mathrm{T}} \cdot \boldsymbol{r}_{\mathrm{SLS}}$. The first few steps in computing this product are very similar to the LS case. Following (D.133) we find that in the SLS case, the resulting vector is again equal to a Kronecker product of the same vector $\tilde{\boldsymbol{s}}^{\mathrm{T}}$ and a modified vector $\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}$, i.e.,

$$\begin{aligned} &\boldsymbol{r}_{\mathrm{SLS}}^{\mathrm{T}} \cdot \boldsymbol{W}_{\mathrm{mat}} = \tilde{\boldsymbol{s}}^{\mathrm{T}} \otimes \tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}, \quad \text{where} \\ &\quad \tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}} = \sqrt{M} \cdot \left(\left(1 - \frac{\gamma(M)}{M}\right) \cdot (\boldsymbol{J}_1\boldsymbol{a})^{+} \cdot (\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu} - \boldsymbol{J}_1) + \frac{1}{M} \cdot \boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}} \cdot \mathrm{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\right) \cdot \left(\boldsymbol{I}_M - \frac{\boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}}{M}\right) \end{aligned} \tag{D.166}$$

For the first term of $\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}$ we find by applying similar arguments as in (D.136)

$$\begin{aligned}
&\sqrt{M}\cdot\left(1-\frac{\gamma(M)}{M}\right)\cdot(\boldsymbol{J}_1\boldsymbol{a})^{+}\cdot(\boldsymbol{J}_2/\mathrm{e}^{\jmath\mu}-\boldsymbol{J}_1)\cdot\left(\boldsymbol{I}_M-\frac{\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}}{M}\right)\\
&=\frac{\sqrt{M}}{M-1}\cdot\left(1-\frac{(M-1)M(M+1)}{M(M^2+11)}\right)\cdot[-1,0,\dots,0,\mathrm{e}^{-(M-1)\jmath\mu}]\\
&=12\cdot\frac{\sqrt{M}}{(M-1)(M^2+11)}\cdot[-1,0,\dots,0,\mathrm{e}^{-(M-1)\jmath\mu}] \qquad \text{(D.167)}
\end{aligned}$$

For the second term we can write

$$\begin{aligned}
\frac{\sqrt{M}}{M}\cdot\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}\cdot\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\cdot\left(\boldsymbol{I}_M-\frac{\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}}{M}\right)&=\frac{\sqrt{M}}{M}\cdot\left(\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}\cdot\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}}-\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}\cdot\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\cdot\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}/M\right)\\
&=\frac{\sqrt{M}}{M}\cdot\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}\cdot\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}} \qquad \text{(D.168)}
\end{aligned}$$

where we have used the fact that $\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\cdot\boldsymbol{a}=\boldsymbol{1}_{M\times 1}$ and $\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}=\boldsymbol{g}^{\mathrm{T}}\cdot(\boldsymbol{J}_2-\boldsymbol{J}_1)$ so that $\boldsymbol{g}_{\mathrm{D}}^{\mathrm{T}}\cdot\operatorname{diag}\{\boldsymbol{a}\}^{\mathrm{H}}\cdot\boldsymbol{a}=\boldsymbol{g}^{\mathrm{T}}\cdot(\boldsymbol{J}_2-\boldsymbol{J}_1)\cdot\boldsymbol{1}_{M\times 1}=\boldsymbol{g}^{\mathrm{T}}\cdot\boldsymbol{0}_{(M-1)\times 1}$ since $(\boldsymbol{J}_2-\boldsymbol{J}_1)$ differentiates the vector of ones. Combining (D.167) and (D.168) we find for $\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}$

$$\begin{aligned}
\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}=\;&12\cdot\frac{\sqrt{M}}{(M-1)(M^2+11)}\cdot[-1,0,\dots,0,\mathrm{e}^{-(M-1)\jmath\mu}]+\frac{\sqrt{M}}{M}\cdot\frac{6}{M^2+11}\cdot\\
&[(-M+1),(-M+3)\mathrm{e}^{-\jmath\mu},(-M+5)\mathrm{e}^{-2\jmath\mu},\dots,(M-1)\cdot\mathrm{e}^{-(M-1)\jmath\mu}] \qquad \text{(D.169)}
\end{aligned}$$

We conclude that the two vectors $\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}$ consists of have the same phase in each element and can hence be conveniently combined. When computing the squared norm of $\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}$ by summing the squared magnitude of all elements the phase terms cancel which also confirms the intuition the the result should be independent of the particular position μ. We obtain

$$\begin{aligned}
\left\|\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}\right\|_2^2=\;&\sqrt{M}^2\cdot\left(\frac{-12}{(M-1)(M^2+11)}+\frac{6\cdot(-M+1)}{M(M^2+11)}\right)^2\\
&+\sum_{m=2}^{M-1}\frac{36}{M(M^2+11)^2}\cdot(-M+2m-1)^2\\
&+\sqrt{M}^2\cdot\left(\frac{12}{(M-1)(M^2+11)}+\frac{6\cdot(M-1)}{M(M^2+11)}\right)^2\\
=\;&2\cdot M\cdot\frac{6^2}{(M^2+11)^2}\cdot\left(\frac{2M+(M-1)^2}{M(M-1)}\right)^2+\frac{36}{M(M^2+11)^2}\cdot\underbrace{\sum_{m=2}^{M-1}(-M+2m-1)^2}_{\frac{1}{3}(M-3)(M-2)(M-1)}
\end{aligned}$$

$$
\begin{aligned}
&= 2 \cdot \frac{36}{(M^2+11)^2} \cdot \frac{(M^2+1)^2}{M(M-1)^2} + \frac{12(M-3)(M-2)(M-1)}{M(M^2+11)^2} \\
&= \frac{12}{M(M^2+11)^2} \left(\frac{6(M^2+1)^2}{(M-1)^2} + (M-3)(M-2)(M-1) \right) \\
&= \frac{12}{M(M^2+11)^2(M-1)^2} \cdot \left(6(M^2+1)^2 + (M-3)(M-2)(M-1)^3\right) \\
&= \frac{12}{M(M^2+11)^2(M-1)^2} \cdot M \cdot (M^4 - 2M^3 + 24M^2 - 22M + 23) \\
&= 12 \cdot \frac{M^4 - 2M^3 + 24M^2 - 22M + 23}{(M^2+11)^2(M-1)^2}
\end{aligned} \tag{D.170}
$$

The mean square error is given by (cf. equation (D.138))

$$
\mathbb{E}\left\{(\Delta\mu)^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\|\tilde{\boldsymbol{s}}^{\mathrm{T}}\right\|_2^2 \cdot \left\|\tilde{\boldsymbol{a}}_{\mathrm{SLS}}^{\mathrm{T}}\right\|_2^2 \tag{D.171}
$$

Inserting (D.134) and (D.170) we have

$$
\begin{aligned}
\mathbb{E}\left\{(\Delta\mu)^2\right\} &= \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \frac{1}{M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot 12 \cdot \frac{M^4 - 2M^3 + 24M^2 - 22M + 23}{(M^2+11)^2(M-1)^2} \\
&= \frac{\sigma_{\mathrm{n}}^2}{N \cdot \hat{P}_{\mathrm{T}}} \cdot 6 \cdot \frac{M^4 - 2M^3 + 24M^2 - 22M + 23}{M(M^2+11)^2(M-1)^2},
\end{aligned} \tag{D.172}
$$

which is the desired result. □

D.18. Proof of Theorem 12.4.7

D.18.1. R-D Standard ESPRIT

The proof for the R-D extension is in fact quite similar to the proof for the 1-D case provided in Section D.16. In fact, (D.126) is still valid, the only difference being that $\boldsymbol{a}(\mu)$ becomes $\boldsymbol{a}(\mu^{(1)}) \otimes \ldots \otimes \boldsymbol{a}(\mu^{(R)}) = \boldsymbol{a}(\boldsymbol{\mu})$. However, the first steps of the derivation can still be performed in the very same way. We obtain the MSE for R-D Standard ESPRIT as

$$
\mathbb{E}\left\{\left(\Delta\mu^{(r)}\right)^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \left\|\boldsymbol{r}^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}}\right\|_2^2 = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \|\tilde{\boldsymbol{s}}\|_2^2 \cdot \left\|\tilde{\boldsymbol{a}}^{(r)}\right\|_2^2, \tag{D.173}
$$

where $\tilde{\boldsymbol{s}}$ is the same as in the 1-D case (cf. equation (D.133)) and $\tilde{\boldsymbol{a}}^{(r)}$ is given by

$$\tilde{\boldsymbol{a}}^{(r)^{\mathrm{T}}} = \sqrt{M}\frac{\boldsymbol{a}^{\mathrm{H}}\tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}}}{\left\|\tilde{\boldsymbol{J}}_1^{(r)}\boldsymbol{a}\right\|_2^2}\left(\tilde{\boldsymbol{J}}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)}\right)\cdot\left(\boldsymbol{I}_M - \frac{1}{M}\cdot\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}\right). \tag{D.174}$$

Since $\tilde{\boldsymbol{J}}_1^{(r)}$ selects the $M_r - 1$ out of M_r elements in the r-th mode, we have $\left\|\tilde{\boldsymbol{J}}_1^{(r)}\boldsymbol{a}\right\|_2^2 = \frac{M}{M_r}\cdot(M_r - 1)$. Moreover, two of the four terms in (D.174) cancel since the array steering vector satisfies the shift invariance equation in the r-th mode, i.e., $\tilde{\boldsymbol{J}}_1^{(r)}\cdot\boldsymbol{a}\cdot\mathrm{e}^{\jmath\cdot\mu^{(r)}} = \tilde{\boldsymbol{J}}_2^{(r)}\cdot\boldsymbol{a}$. We obtain

$$\begin{aligned}
\tilde{\boldsymbol{a}}^{(r)^{\mathrm{T}}} =& \sqrt{M}\frac{\boldsymbol{a}^{\mathrm{H}}\tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}}}{\frac{M}{M_r}\cdot(M_r-1)}\left(\left(\tilde{\boldsymbol{J}}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)}\right) - \frac{1}{M}\cdot\left(\tilde{\boldsymbol{J}}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)}\right)\cdot\boldsymbol{a}\cdot\boldsymbol{a}^{\mathrm{H}}\right) \\
=& \sqrt{M}\frac{\boldsymbol{a}^{\mathrm{H}}\tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}}}{\frac{M}{M_r}\cdot(M_r-1)}\left(\left(\tilde{\boldsymbol{J}}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)}\right) - \frac{1}{M}\cdot\underbrace{\left(\tilde{\boldsymbol{J}}_2^{(r)}\cdot\boldsymbol{a}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)}\cdot\boldsymbol{a}\right)}_{\boldsymbol{0}}\cdot\boldsymbol{a}^{\mathrm{H}}\right) \\
=& \frac{\sqrt{M}\cdot M_r}{M\cdot(M_r-1)}\cdot\left(\boldsymbol{a}^{\mathrm{H}}\cdot\tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}}\cdot\tilde{\boldsymbol{J}}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} - \boldsymbol{a}^{\mathrm{H}}\cdot\tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}}\cdot\tilde{\boldsymbol{J}}_1^{(r)}\right)
\end{aligned} \tag{D.175}$$

Since the the array steering vector $\boldsymbol{a}$ and the selection matrices $\tilde{\boldsymbol{J}}_\ell^{(r)}$ can be factored into Kronecker products according to $\boldsymbol{a} = \boldsymbol{a}^{(1)}\otimes\ldots\boldsymbol{a}^{(R)}$ and $\tilde{\boldsymbol{J}}_\ell^{(r)} = \boldsymbol{I}_{\prod_{n=1}^{r-1}M_n}\otimes\boldsymbol{J}_\ell^{(r)}\otimes\boldsymbol{I}_{\prod_{n=r+1}^{R}M_n}$, for $\ell = 1,2$ and $r = 1,2,\ldots,R$, all "unaffected" modes can be factored out of (D.175) and we have

$$\tilde{\boldsymbol{a}}^{(r)^{\mathrm{T}}} = \frac{\sqrt{M}\cdot M_r}{M\cdot(M_r-1)}\cdot\left(\boldsymbol{a}^{(1)}\otimes\ldots\otimes\boldsymbol{a}^{(r-1)}\right)^{\mathrm{H}}\otimes\left(\tilde{\boldsymbol{a}}_1^{(r)} - \tilde{\boldsymbol{a}}_2^{(r)}\right)^{\mathrm{T}}\otimes\left(\boldsymbol{a}^{(r+1)}\otimes\ldots\otimes\boldsymbol{a}^{(R)}\right)^{\mathrm{H}} \tag{D.176}$$

where $\tilde{\boldsymbol{a}}_1^{(r)}$ and $\tilde{\boldsymbol{a}}_2^{(r)}$ are given by

$$\tilde{\boldsymbol{a}}_1^{(r)^{\mathrm{T}}} = \boldsymbol{a}^{(r)^{\mathrm{H}}}\cdot\boldsymbol{J}_1^{(r)^{\mathrm{H}}}\cdot\boldsymbol{J}_2^{(r)}/\mathrm{e}^{\jmath\cdot\mu^{(r)}} \quad \text{and} \quad \tilde{\boldsymbol{a}}_2^{(r)^{\mathrm{T}}} = \boldsymbol{a}^{(r)^{\mathrm{H}}}\cdot\boldsymbol{J}_1^{(r)^{\mathrm{H}}}\cdot\boldsymbol{J}_1^{(r)}. \tag{D.177}$$

Following the same reasoning as for (D.137) we find

$$\tilde{\boldsymbol{a}}_1^{(r)^{\mathrm{T}}} - \tilde{\boldsymbol{a}}_2^{(r)^{\mathrm{T}}} = \left[-1, 0, \ldots, 0, \mathrm{e}^{-(M_r-1)\jmath\mu^{(r)}}\right] \tag{D.178}$$

Consequently, the desired norm $\left\|\tilde{\boldsymbol{a}}^{(r)}\right\|_2^2$ is directly found to be

$$\begin{aligned}\left\|\tilde{\boldsymbol{a}}^{(r)}\right\|_2^2 =& \frac{M \cdot M_r^2}{M^2 \cdot (M_r - 1)^2} \cdot \left(\prod_{n=1}^{r-1} \left\|\boldsymbol{a}^{(n)}\right\|_2^2\right) \cdot 2 \cdot \left(\prod_{n=r+1}^{R} \left\|\boldsymbol{a}^{(n)}\right\|_2^2\right) = 2 \cdot \frac{M_r^2}{M \cdot (M_r - 1)^2} \cdot \frac{M}{M_r} \\ =& 2 \cdot \frac{M_r}{(M_r - 1)^2}.\end{aligned} \tag{D.179}$$

Therefore, the MSE expression for R-D Standard ESPRIT is given by

$$\mathbb{E}\left\{\left(\Delta\mu^{(r)}\right)^2\right\} = \frac{\sigma_{\mathrm{n}}^2}{2} \cdot \frac{1}{M \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot 2 \cdot \frac{M_r}{(M_r - 1)^2} = \frac{\sigma_{\mathrm{n}}^2}{N \cdot \hat{P}_{\mathrm{T}}} \cdot \frac{M_r}{M \cdot (M_r - 1)^2} \tag{D.180}$$

which proofs the first part of the theorem. □

D.18.2. R-D Unitary ESPRIT

The second part of the theorem is to prove that in the R-D case the performance of R-D Unitary ESPRIT and R-D Standard ESPRIT are the same as long as a single source is present. However, for this part, no changes have to be made compared to Section D.16.2: As it was shown there, Forward-Backward-Averaging only affects $\boldsymbol{v}_{\mathrm{s}}$ and has no effect on $\boldsymbol{u}_{\mathrm{s}}$ or $\boldsymbol{U}_{\mathrm{n}}$. Applying the same steps here immediately proves this part of the theorem.

D.18.3. Cramér-Rao Bound

The third part is the simplification of the Cramér-Rao Bound. In the R-D case, the CRB is given by

$$\boldsymbol{C} = \frac{\sigma_{\mathrm{n}}^2}{2 \cdot N} \cdot \mathrm{Re}\left\{\left[\boldsymbol{D}^{(R)^{\mathrm{H}}} \cdot \left(\boldsymbol{I}_M - \boldsymbol{A} \cdot \left(\boldsymbol{A}^{\mathrm{H}} \cdot \boldsymbol{A}\right)^{-1} \cdot \boldsymbol{A}^{\mathrm{H}}\right) \cdot \boldsymbol{D}^{(R)}\right] \odot \left(\mathbf{1}_{R\times R} \otimes \hat{\boldsymbol{R}}_{\mathrm{S}}^{\mathrm{T}}\right)\right\}^{-1} \tag{D.181}$$

where $\boldsymbol{D}^{(R)} \in \mathbb{C}^{M\times(d\cdot R)}$ contains the partial derivatives of the array steering vectors $\boldsymbol{a}_n$ with respect to $\mu_n^{(r)}$ for $n = 1, 2, \ldots, d$ and $r = 1, 2, \ldots, R$. For the special case of a single source, the CRB simplifies into

$$\begin{aligned}\boldsymbol{C} &= \frac{\sigma_{\mathrm{n}}^2}{2 \cdot N} \cdot \mathrm{Re}\left\{\left[\boldsymbol{D}^{(R)^{\mathrm{H}}} \cdot \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}\right) \cdot \boldsymbol{D}^{(R)}\right] \odot \left(\mathbf{1}_{R\times R} \otimes \hat{P}_{\mathrm{T}}\right)\right\}^{-1} \\ &= \frac{\sigma_{\mathrm{n}}^2}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \mathrm{Re}\left\{\underbrace{\boldsymbol{D}^{(R)^{\mathrm{H}}} \cdot \left(\boldsymbol{I}_M - \frac{1}{M} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}\right) \cdot \boldsymbol{D}^{(R)}}_{\boldsymbol{J}}\right\}^{-1}\end{aligned} \tag{D.182}$$

The columns of $\boldsymbol{D}^{(R)} \in \mathbb{C}^{M \times R}$ are given by $\tilde{\boldsymbol{d}}^{(r)} = \frac{\partial \boldsymbol{a}}{\partial \mu^{(r)}} \in \mathbb{C}^{M \times 1}$. Using the fact that $\boldsymbol{a} = \boldsymbol{a}^{(1)} \otimes \ldots \otimes \boldsymbol{a}^{(R)}$ we obtain

$$\tilde{\boldsymbol{d}}^{(r)} = \boldsymbol{a}^{(1)} \otimes \ldots \otimes \boldsymbol{a}^{(r-1)} \otimes \boldsymbol{d}^{(r)} \otimes \boldsymbol{a}^{(r+1)} \otimes \ldots \otimes \boldsymbol{a}^{(R)} \tag{D.183}$$

where $\boldsymbol{d}^{(r)} = \frac{\partial \boldsymbol{a}^{(r)}}{\partial \mu^{(r)}} \in \mathbb{C}^{M_r \times 1} = \jmath \cdot [0, \mathrm{e}^{\jmath \mu^{(r)}}, 2 \cdot \mathrm{e}^{2 \jmath \mu^{(r)}}, \ldots, (M-1)\mathrm{e}^{(M-1)\jmath \mu^{(r)}}]$. Therefore, the elements of the matrix $\boldsymbol{J}$ are given by

$$[\boldsymbol{J}]_{(r_1,r_2)} = \tilde{\boldsymbol{d}}^{(r_1)^{\mathrm{H}}} \cdot \tilde{\boldsymbol{d}}^{(r_2)} - \frac{1}{M} \cdot \tilde{\boldsymbol{d}}^{(r_1)^{\mathrm{H}}} \cdot \boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}} \cdot \tilde{\boldsymbol{d}}^{(r_2)} \tag{D.184}$$

With the help of (D.183) we find for the diagonal elements ($r_1 = r_2 = r$)

$$\begin{aligned} \tilde{\boldsymbol{d}}^{(r)^{\mathrm{H}}} \cdot \tilde{\boldsymbol{d}}^{(r)} &= \boldsymbol{a}^{(1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(1)} \cdot \ldots \cdot \boldsymbol{a}^{(r-1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(r-1)} \cdot \boldsymbol{d}^{(r)^{\mathrm{H}}} \cdot \boldsymbol{d}^{(r)} \cdot \boldsymbol{a}^{(r+1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(r+1)} \cdot \ldots \cdot \boldsymbol{a}^{(R)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(R)} \\ &= (M_1 \cdot \ldots \cdot M_{r-1}) \cdot \left(\sum_{m=0}^{M_r - 1} m^2 \right) \cdot (M_{r+1} \cdot \ldots \cdot M_R) \\ &= \frac{1}{6} \cdot M \cdot (M_r - 1) \cdot (2M_r - 1) \end{aligned} \tag{D.185}$$

and similarly

$$\begin{aligned} \tilde{\boldsymbol{d}}^{(r)^{\mathrm{H}}} \cdot \boldsymbol{a} &= \boldsymbol{a}^{(1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(1)} \cdot \ldots \cdot \boldsymbol{a}^{(r-1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(r-1)} \cdot \boldsymbol{d}^{(r)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(r)} \cdot \boldsymbol{a}^{(r+1)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(r+1)} \cdot \ldots \cdot \boldsymbol{a}^{(R)^{\mathrm{H}}} \cdot \boldsymbol{a}^{(R)} \\ &= (M_1 \cdot \ldots \cdot M_{r-1}) \cdot \left(-\jmath \sum_{m=0}^{M_r - 1} m \right) \cdot (M_{r+1} \cdot \ldots \cdot M_R) \\ &= -\jmath \cdot M \cdot \frac{1}{2} \cdot (M_r - 1) \end{aligned} \tag{D.186}$$

Combining these two results we have for $[\boldsymbol{J}]_{(r,r)}$

$$\begin{aligned} [\boldsymbol{J}]_{(r,r)} &= \frac{1}{6} \cdot M \cdot (M_r - 1) \cdot (2M_r - 1) - \frac{1}{M} (-\jmath) \cdot M \cdot \frac{1}{2} \cdot (M_r - 1) \cdot \jmath \cdot M \cdot \frac{1}{2} \cdot (M_r - 1) \\ &= \frac{1}{6} \cdot M \cdot (M_r - 1) \cdot (2M_r - 1) - \frac{1}{4} \cdot (M_r - 1) M \cdot (M_r - 1) \\ &= \frac{1}{12} \cdot M \cdot (M_r - 1)\big(2(2M_r - 1) - 3(M_r - 1)\big) \\ &= \frac{1}{12} \cdot M \cdot (M_r - 1)(M_r + 1) \end{aligned} \tag{D.187}$$

On the other hand, for the off-diagonal elements we obtain

$$\begin{aligned}\tilde{\boldsymbol{d}}^{(r_1)^{\mathrm{H}}} \cdot \tilde{\boldsymbol{d}}^{(r_2)} &= \frac{M}{M_{r_1} \cdot M_{r_2}} \cdot \left(-\jmath \frac{1}{2} \cdot (M_{r_1} - 1) \cdot M_{r_1}\right) \cdot \left(\jmath \frac{1}{2} \cdot (M_{r_2} - 1) \cdot M_{r_2}\right) \\ &= \frac{1}{4} M \cdot (M_{r_1} - 1) \cdot M_{r_1} \cdot (M_{r_2} - 1) \cdot M_{r_2} \end{aligned} \tag{D.188}$$

and therefore for $[\boldsymbol{J}]_{(r_1,r_2)}$, $r_1 \neq r_2$

$$\begin{aligned}[\boldsymbol{J}]_{(r_1,r_2)} =& \frac{1}{4} M \cdot (M_{r_1} - 1) \cdot M_{r_1} \cdot (M_{r_2} - 1) \cdot M_{r_2} \\ & - \frac{1}{M} (-\jmath) \cdot M \cdot \frac{1}{2} \cdot (M_{r_1} - 1) \cdot (\jmath) \cdot M \cdot \frac{1}{2} \cdot (M_{r_2} - 1) \\ =& \frac{1}{4} M \cdot (M_{r_1} - 1) \cdot M_{r_1} \cdot (M_{r_2} - 1) \cdot M_{r_2} - \frac{1}{4} M \cdot (M_{r_1} - 1) \cdot M_{r_1} \cdot (M_{r_2} - 1) \cdot M_{r_2} \\ =& 0 \end{aligned} \tag{D.189}$$

This shows that $\boldsymbol{J}$ is diagonal and real-valued. Consequently, the CRB becomes

$$\boldsymbol{C} = \frac{\sigma_{\mathrm{n}}^2}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \cdot \mathrm{Re}\{\boldsymbol{J}\}^{-1} = \mathrm{diag}\left\{\left[C^{(1)}, \ \ldots, C^{(R)}\right]\right\} \quad \text{where} \tag{D.190}$$

$$C^{(r)} = \frac{\sigma_{\mathrm{n}}^2}{2 \cdot N \cdot \hat{P}_{\mathrm{T}}} \frac{12}{M \cdot (M_r - 1) \cdot (M_r + 1)} = \frac{1}{\hat{\rho}} \cdot \frac{6}{M \cdot (M_r - 1) \cdot (M_r + 1)},$$

which is the desired result. □

D.18.4. R-D Standard Tensor-ESPRIT

The fourth part of the theorem is to show that the MSE of R-D Standard Tensor-ESPRIT is the same as the MSE for R-D Standard ESPRIT for $d = 1$. Since we have only shown the expressions for R-D Standard Tensor-ESPRIT in the special case $R = 2$, we will also assume this case here.

Note that the MSE expression for Tensor-ESPRIT is in fact quite similar to the one for matrix-based ESPRIT with the only difference being that the matrix $\boldsymbol{W}_{\mathrm{mat}}$ is replaced by the matrix $\boldsymbol{W}_{\mathrm{ten}}$ which we restate here for convenience

$$\begin{aligned}\boldsymbol{W}_{\mathrm{ten}} =& \left(\boldsymbol{\Sigma}_3^{[\mathrm{s}]^{-1}} \boldsymbol{U}_3^{[\mathrm{s}]^{\mathrm{H}}}\right) \otimes \left([\boldsymbol{T}_1 \otimes \boldsymbol{T}_2] \boldsymbol{V}_3^{[\mathrm{n}]^*} \boldsymbol{V}_3^{[\mathrm{n}]^{\mathrm{T}}}\right) \\ &+ \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \bar{\boldsymbol{T}}_2 \left(\boldsymbol{U}_1^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \boldsymbol{V}_1^{[\mathrm{s}]^{\mathrm{T}}} \otimes \boldsymbol{U}_1^{[\mathrm{n}]} \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}}\right) \cdot \boldsymbol{K}_{M_2 \times (M_1 \cdot N)} \\ &+ \left(\boldsymbol{U}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \bar{\boldsymbol{T}}_1 \left(\boldsymbol{U}_2^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_2^{[\mathrm{s}]^{-1}} \boldsymbol{V}_2^{[\mathrm{s}]^{\mathrm{T}}} \otimes \boldsymbol{U}_2^{[\mathrm{n}]} \boldsymbol{U}_2^{[\mathrm{n}]^{\mathrm{H}}}\right) \quad \text{where} \end{aligned} \tag{D.191}$$

$$\bar{\boldsymbol{T}}_1 = \begin{bmatrix} \boldsymbol{I}_{M_2} \otimes \boldsymbol{t}_{1,1} \\ \vdots \\ \boldsymbol{I}_{M_2} \otimes \boldsymbol{t}_{1,M_1} \end{bmatrix} \otimes \boldsymbol{I}_{M_2}, \quad \bar{\boldsymbol{T}}_2 = \boldsymbol{I}_{M_1} \otimes \begin{bmatrix} \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,1} \\ \vdots \\ \boldsymbol{I}_{M_1} \otimes \boldsymbol{t}_{2,M_2} \end{bmatrix},$$

and $\boldsymbol{t}_{r,m}$ is the m-th column of $\boldsymbol{T}_r$.

To simplify this expression for the special case $d = 1$, we express $\boldsymbol{\mathcal{X}}_0$ and its unfoldings as

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{a}^{(1)} \circ \boldsymbol{a}^{(2)} \circ \boldsymbol{s} \in \mathbb{C}^{M_1 \times M_2 \times N} \tag{D.192}$$

$$[\boldsymbol{\mathcal{X}}_0]_{(1)} = \boldsymbol{a}^{(1)} \cdot \left(\boldsymbol{a}^{(2)} \otimes \boldsymbol{s}\right)^{\mathrm{T}}, \quad [\boldsymbol{\mathcal{X}}_0]_{(2)} = \boldsymbol{a}^{(2)} \cdot \left(\boldsymbol{s} \otimes \boldsymbol{a}^{(1)}\right)^{\mathrm{T}}, \quad [\boldsymbol{\mathcal{X}}_0]_{(3)} = \boldsymbol{s} \cdot \left(\boldsymbol{a}^{(1)} \otimes \boldsymbol{a}^{(2)}\right)^{\mathrm{T}}$$

Consequently, we can relate the necessary subspaces of the unfoldings of $\boldsymbol{\mathcal{X}}_0$ to $\boldsymbol{s}$ and $\boldsymbol{a}^{(r)}$ via

$$\boldsymbol{u}_1^{[\mathrm{s}]} = \frac{\boldsymbol{a}^{(1)}}{\sqrt{M_1}}, \quad \boldsymbol{u}_2^{[\mathrm{s}]} = \frac{\boldsymbol{a}^{(2)}}{\sqrt{M_2}}, \quad \boldsymbol{u}_3^{[\mathrm{s}]} = \frac{\boldsymbol{s}}{\sqrt{N \cdot \hat{P}_{\mathrm{T}}}} \tag{D.193}$$

$$\boldsymbol{U}_1^{[\mathrm{n}]} = \boldsymbol{I}_{M_1} - \frac{\boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}}}{M_1}, \quad \boldsymbol{U}_2^{[\mathrm{n}]} = \boldsymbol{I}_{M_2} - \frac{\boldsymbol{a}^{(2)} \cdot \boldsymbol{a}^{(2)^{\mathrm{H}}}}{M_2}$$

$$\boldsymbol{\Sigma}_1^{[\mathrm{s}]} = \boldsymbol{\Sigma}_2^{[\mathrm{s}]} = \boldsymbol{\Sigma}_3^{[\mathrm{s}]} = \sqrt{M \cdot N \cdot \hat{P}_{\mathrm{T}}}$$

$$\boldsymbol{v}_1^{[\mathrm{s}]} = \frac{\left(\boldsymbol{a}^{(2)} \otimes \boldsymbol{s}\right)^*}{\sqrt{M_2 \cdot N \cdot \hat{P}_{\mathrm{T}}}}, \quad \boldsymbol{v}_2^{[\mathrm{s}]} = \frac{\left(\boldsymbol{s} \otimes \boldsymbol{a}^{(1)}\right)^*}{\sqrt{M_1 \cdot N \cdot \hat{P}_{\mathrm{T}}}}, \quad \boldsymbol{v}_3^{[\mathrm{s}]} = \boldsymbol{u}_{\mathrm{s}} = \frac{\boldsymbol{a}}{\sqrt{M}}$$

$$\boldsymbol{V}_3^{[\mathrm{n}]^*} \cdot \boldsymbol{V}_3^{[\mathrm{n}]^{\mathrm{T}}} = \boldsymbol{U}_{\mathrm{n}} \cdot \boldsymbol{U}_{\mathrm{n}}^{\mathrm{H}} = \boldsymbol{I} - \frac{\boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}}{M} \tag{D.194}$$

Moreover, we have for $\boldsymbol{T}_r$

$$\boldsymbol{T}_r = \boldsymbol{u}_r^{[\mathrm{s}]} \cdot \boldsymbol{u}_r^{[\mathrm{s}]^{\mathrm{H}}} = \frac{\boldsymbol{a}^{(r)} \cdot \boldsymbol{a}^{(r)^{\mathrm{H}}}}{M_r}, \quad \text{for } r = 1, 2 \text{ and thus}$$

$$\boldsymbol{T}_1 \otimes \boldsymbol{T}_2 = \frac{\boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}}}{M_1} \otimes \frac{\boldsymbol{a}^{(2)} \cdot \boldsymbol{a}^{(2)^{\mathrm{H}}}}{M_2} = \frac{(\boldsymbol{a}^{(1)} \otimes \boldsymbol{a}^{(2)}) \cdot (\boldsymbol{a}^{(1)} \otimes \boldsymbol{a}^{(2)})^{\mathrm{H}}}{M} = \frac{\boldsymbol{a} \cdot \boldsymbol{a}^{\mathrm{H}}}{M} \tag{D.195}$$

From (D.194) and (D.195) it immediately follows that the first term in $\boldsymbol{W}_{\mathrm{ten}}$ cancels as it contains $[\boldsymbol{T}_1 \otimes \boldsymbol{T}_2] \cdot \boldsymbol{V}_3^{[\mathrm{n}]^*} \cdot \boldsymbol{V}_3^{[\mathrm{n}]^{\mathrm{T}}}$. We also find $\boldsymbol{t}_{r,m} = \boldsymbol{a}^{(r)} \cdot \mathrm{e}^{-\jmath \mu^{(r)} (m-1)} / M_r$ for $r = 1, 2$ and $m = 1, 2, \ldots, M_r$. To simplify the remaining two terms in $\boldsymbol{W}_{\mathrm{ten}}$ we first look at some of their components. We start by $\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_1$. Using the identity $\boldsymbol{u}_{\mathrm{s}} = \boldsymbol{a}/\sqrt{M}$ and the explicit expression for $\boldsymbol{t}_{r,m}$ we may write

$$\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_1 = \frac{1}{\sqrt{M}} \left(\boldsymbol{a}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_1 = \frac{1}{\sqrt{M}} \left(\boldsymbol{a}^{(1)^{\mathrm{T}}} \otimes \boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{I}_{M_1} \otimes \boldsymbol{I}_{M_2}\right) \cdot \bar{\boldsymbol{T}}_1$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{M}} \left(\left[1, \mathrm{e}^{\jmath\mu^{(1)}}, \ldots, \mathrm{e}^{\jmath\mu^{(1)}(M_1-1)} \right] \otimes \boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{I}_{M_1} \otimes \boldsymbol{I}_{M_2} \right) \cdot \bar{\boldsymbol{T}}_1 \\
&= \frac{1}{\sqrt{M}} \left(\left[1, \mathrm{e}^{\jmath\mu^{(1)}}, \ldots, \mathrm{e}^{\jmath\mu^{(1)}(M_1-1)} \right] \otimes \boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{I}_{M_1} \otimes \boldsymbol{I}_{M_2} \right) \\
&\quad \cdot \frac{1}{M_r} \cdot \begin{bmatrix} \boldsymbol{I}_{M_2} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \\ \boldsymbol{I}_{M_2} \otimes \boldsymbol{a}^{(1)} \cdot \mathrm{e}^{-\jmath\mu^{(1)}} \otimes \boldsymbol{I}_{M_2} \\ \vdots \\ \boldsymbol{I}_{M_2} \otimes \boldsymbol{a}^{(1)} \cdot \mathrm{e}^{-\jmath\mu^{(1)}(M_1-1)} \otimes \boldsymbol{I}_{M_2} \end{bmatrix} \\
&= \frac{1}{M_r\sqrt{M}} \sum_{m=1}^{M_r} \mathrm{e}^{\jmath\mu^{(1)}(m-1)} \cdot \left(\boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{I}_{M_1} \otimes \boldsymbol{I}_{M_2} \right) \cdot \left(\boldsymbol{I}_{M_2} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \right) \cdot \mathrm{e}^{-\jmath\mu^{(1)}(m-1)} \\
&= \frac{1}{M_r\sqrt{M}} \left(\boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \right) \cdot \underbrace{\sum_{m=1}^{M_r} 1}_{M_r} = \frac{1}{\sqrt{M}} \left(\boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \right),
\end{aligned}
\tag{D.196}
$$

where we have used the fact that $\boldsymbol{x}^{\mathrm{T}} \otimes \boldsymbol{Y} = \left[x_1 \cdot \boldsymbol{Y}, \ldots, x_N \cdot \boldsymbol{Y} \right]$, where x_n is the n-th element of $\boldsymbol{x} \in \mathbb{C}^{N\times 1}$, and and $\boldsymbol{Y}$ is a matrix of arbitrary size. By applying similar reasoning to $\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M \right) \cdot \bar{\boldsymbol{T}}_2$ we can show

$$
\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M \right) \cdot \bar{\boldsymbol{T}}_2 = \frac{1}{\sqrt{M}} \left(\boldsymbol{a}^{(1)^{\mathrm{T}}} \otimes \boldsymbol{I}_{M_1} \otimes \boldsymbol{a}^{(2)} \right) \tag{D.197}
$$

The remaining terms in $\boldsymbol{W}_{\mathrm{ten}}$ we have not yet associated with $\boldsymbol{s}$ and $\boldsymbol{a}^{(r)}$ yet are $\boldsymbol{U}_r^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_r^{[\mathrm{s}]^{-1}} \boldsymbol{V}_r^{[\mathrm{s}]^{\mathrm{T}}} \otimes \boldsymbol{U}_r^{[\mathrm{n}]} \boldsymbol{U}_r^{[\mathrm{n}]^{\mathrm{H}}}$. Using the relations (D.193)-(D.194) we can write

$$
\begin{aligned}
\left(\boldsymbol{U}_1^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \boldsymbol{V}_1^{[\mathrm{s}]^{\mathrm{T}}} \right) \otimes \left(\boldsymbol{U}_1^{[\mathrm{n}]} \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}} \right) &= \frac{1}{MN\hat{P}_{\mathrm{T}}} \cdot \left(\boldsymbol{a}^{(1)^*} \cdot \left(\boldsymbol{a}^{(2)} \otimes \boldsymbol{s} \right)^{\mathrm{H}} \right) \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \\
\left(\boldsymbol{U}_2^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_2^{[\mathrm{s}]^{-1}} \boldsymbol{V}_2^{[\mathrm{s}]^{\mathrm{T}}} \right) \otimes \left(\boldsymbol{U}_2^{[\mathrm{n}]} \boldsymbol{U}_2^{[\mathrm{n}]^{\mathrm{H}}} \right) &= \frac{1}{MN\hat{P}_{\mathrm{T}}} \cdot \left(\boldsymbol{a}^{(2)^*} \cdot \left(\boldsymbol{s} \otimes \boldsymbol{a}^{(1)} \right)^{\mathrm{H}} \right) \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}}
\end{aligned}
$$

where we have used the short-hand notation $\boldsymbol{\Pi}^{\perp}_{\boldsymbol{x}} = \boldsymbol{I}_N - \frac{\boldsymbol{x}\boldsymbol{x}^{\mathrm{H}}}{\boldsymbol{x}^{\mathrm{H}}\boldsymbol{x}}$ for the projection matrix onto the orthogonal complement of the vector $\boldsymbol{x} \in \mathbb{C}^{N\times 1}$. Combining these intermediate result, the last term in $\boldsymbol{W}_{\mathrm{ten}}$ can be expressed as

$$
\begin{aligned}
&\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M \right) \cdot \bar{\boldsymbol{T}}_1 \cdot \left(\boldsymbol{U}_2^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_2^{[\mathrm{s}]^{-1}} \boldsymbol{V}_2^{[\mathrm{s}]^{\mathrm{T}}} \right) \otimes \left(\boldsymbol{U}_2^{[\mathrm{n}]} \boldsymbol{U}_2^{[\mathrm{n}]^{\mathrm{H}}} \right) \\
&= \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left(\boldsymbol{a}^{(2)^{\mathrm{T}}} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \right) \cdot \left[\left(\boldsymbol{a}^{(2)^*} \cdot \left(\boldsymbol{s} \otimes \boldsymbol{a}^{(1)} \right)^{\mathrm{H}} \right) \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right] \\
&= \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left\| \boldsymbol{a}^{(2)} \right\|_2^2 \cdot \left(\boldsymbol{s} \otimes \boldsymbol{a}^{(1)} \right)^{\mathrm{H}} \otimes \left[\left(\boldsymbol{a}^{(1)} \otimes \boldsymbol{I}_{M_2} \right) \cdot \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right]
\end{aligned}
$$

$$= \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left(M_2 \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \boldsymbol{a}^{(1)^{\mathrm{H}}} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right) \tag{D.198}$$

With similar arguments, the second term in $\boldsymbol{W}_{\mathrm{ten}}$ can be simplified into

$$\begin{aligned}
&\left(\boldsymbol{u}_{\mathrm{s}}^{\mathrm{T}} \otimes \boldsymbol{I}_M\right) \cdot \bar{\boldsymbol{T}}_2 \cdot \left(\boldsymbol{U}_1^{[\mathrm{s}]^*} \boldsymbol{\Sigma}_1^{[\mathrm{s}]^{-1}} \boldsymbol{V}_1^{[\mathrm{s}]^{\mathrm{T}}}\right) \otimes \left(\boldsymbol{U}_1^{[\mathrm{n}]} \boldsymbol{U}_1^{[\mathrm{n}]^{\mathrm{H}}}\right) \cdot \boldsymbol{K}_{M_2 \times (M_1 \cdot N)} \\
=& \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left(M_1 \cdot \boldsymbol{a}^{(2)^{\mathrm{H}}} \otimes \boldsymbol{s}^{\mathrm{H}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{a}^{(2)} \right) \cdot \boldsymbol{K}_{M_2 \times (M_1 \cdot N)} \\
=& \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left(M_1 \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{a}^{(2)} \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \right)
\end{aligned} \tag{D.199}$$

where the last step is a special case of Property (3.35) for commutation matrices.

Using (D.198) and (D.199) in (D.191), we obtain

$$\begin{aligned}
\boldsymbol{W}_{\mathrm{ten}} &= \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \left(M_1 \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{a}^{(2)} \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} + M_2 \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \boldsymbol{a}^{(1)^{\mathrm{H}}} \otimes \boldsymbol{a}^{(1)} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right) \\
&= \frac{1}{MN\hat{P}_{\mathrm{T}}\sqrt{M}} \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \left(M_1 \cdot \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \left(\boldsymbol{a}^{(2)} \cdot \boldsymbol{a}^{(2)^{\mathrm{H}}}\right) + M_2 \cdot \left(\boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}}\right) \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right) \\
&= \frac{1}{N\hat{P}_{\mathrm{T}}\sqrt{M}} \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \left(\boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \frac{\boldsymbol{a}^{(2)} \cdot \boldsymbol{a}^{(2)^{\mathrm{H}}}}{M_2} + \frac{\boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}}}{M_1} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right) \\
&= \frac{1}{N\hat{P}_{\mathrm{T}}\sqrt{M}} \cdot \boldsymbol{s}^{\mathrm{H}} \otimes \left(\boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} + \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right)
\end{aligned} \tag{D.200}$$

where we applied the rule $\boldsymbol{x} \otimes \boldsymbol{y}^{\mathrm{T}} = \boldsymbol{x} \cdot \boldsymbol{y}^{\mathrm{T}}$ and used the short-hand notation $\boldsymbol{\Pi}_{\boldsymbol{x}} = \frac{\boldsymbol{x}\boldsymbol{x}^{\mathrm{H}}}{\boldsymbol{x}^{\mathrm{H}}\boldsymbol{x}}$ for the projection matrix onto the vector $\boldsymbol{x} \in \mathbb{C}^{N \times 1}$. Comparing (D.200) with (D.132) and its R-D extension in (D.174) we find that for a single source, $\boldsymbol{W}_{\mathrm{mat}}$ and $\boldsymbol{W}_{\mathrm{ten}}$ are in fact quite similar, the only difference being that $\boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}}$ is replaced by $\boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} + \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}}$. Therefore, to show the R-D Standard ESPRIT and R-D Standard Tensor-ESPRIT have the same MSE for $d = 1$, it is sufficient to show that the corresponding terms $\tilde{\boldsymbol{a}}^{(r)}$ are the same, i.e., that

$$\boldsymbol{a}^{\mathrm{H}} \tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}} \left(\tilde{\boldsymbol{J}}_2^{(r)} / \mathrm{e}^{\jmath \cdot \mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)} \right) \cdot \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}} = \boldsymbol{a}^{\mathrm{H}} \tilde{\boldsymbol{J}}_1^{(r)^{\mathrm{H}}} \left(\tilde{\boldsymbol{J}}_2^{(r)} / \mathrm{e}^{\jmath \cdot \mu^{(r)}} - \tilde{\boldsymbol{J}}_1^{(r)} \right) \cdot \left(\boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} + \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}^{\perp}_{\boldsymbol{a}^{(2)}} \right) \tag{D.201}$$

for $r = 1, 2$. Note that the left-hand side of (D.201) was shown to be equal to (cf. equation (D.176))

$$\begin{aligned}
&\left(\tilde{\boldsymbol{a}}_1^{(1)} - \tilde{\boldsymbol{a}}_2^{(1)} \right)^{\mathrm{T}} \otimes \boldsymbol{a}^{(2)^{\mathrm{T}}}, \quad \text{for } r = 1 \text{ and } \quad \boldsymbol{a}^{(1)^{\mathrm{T}}} \otimes \left(\tilde{\boldsymbol{a}}_1^{(2)} - \tilde{\boldsymbol{a}}_2^{(2)} \right)^{\mathrm{T}} \quad \text{for } r = 2 \quad \text{where} \\
&\tilde{\boldsymbol{a}}_1^{(r)^{\mathrm{T}}} = \boldsymbol{a}^{(r)^{\mathrm{H}}} \cdot \boldsymbol{J}_1^{(r)^{\mathrm{H}}} \cdot \boldsymbol{J}_2^{(r)} / \mathrm{e}^{\jmath \cdot \mu^{(r)}} \quad \text{and} \quad \tilde{\boldsymbol{a}}_2^{(r)^{\mathrm{T}}} = \boldsymbol{a}^{(r)^{\mathrm{H}}} \cdot \boldsymbol{J}_1^{(r)^{\mathrm{H}}} \cdot \boldsymbol{J}_1^{(r)}.
\end{aligned} \tag{D.202}$$

Expanding the corresponding right-hand side of (D.201) we have for $r = 1$

$$
\begin{aligned}
&\boldsymbol{a}^{\mathrm{H}} \tilde{\boldsymbol{J}}_1^{(1)^{\mathrm{H}}} \left(\tilde{\boldsymbol{J}}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \tilde{\boldsymbol{J}}_1^{(1)} \right) \cdot \left(\boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}}^{\perp} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} + \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}^{\perp} \right) \\
&= \left[\left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \right] \cdot \left(\boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}}^{\perp} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} + \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \otimes \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}^{\perp} \right) \\
&= \left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}}^{\perp} \right) \otimes \left(\underbrace{\boldsymbol{a}^{(2)^{\mathrm{H}}} \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}}_{\boldsymbol{a}^{(2)^{\mathrm{H}}}} \right) \\
&+ \left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(1)}} \right) \otimes \left(\underbrace{\boldsymbol{a}^{(2)^{\mathrm{H}}} \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}^{\perp}}_{\boldsymbol{0}_{1 \times M_2}} \right)
\end{aligned} \tag{D.203}
$$

where we have used the fact that $\boldsymbol{a} = \boldsymbol{a}^{(1)} \otimes \boldsymbol{a}^{(2)}$ and $\tilde{\boldsymbol{J}}_\ell^{(1)} = \boldsymbol{J}_\ell^{(1)} \otimes \boldsymbol{I}_{M_2}$ for $\ell = 1, 2$. Moreover, $\boldsymbol{a}^{(2)^{\mathrm{H}}} \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}} = \boldsymbol{a}^{(2)^{\mathrm{H}}}$ and $\boldsymbol{a}^{(2)^{\mathrm{H}}} \cdot \boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}^{\perp} = \boldsymbol{0}_{1 \times M_2}$ follow from the fact that $\boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}$ and $\boldsymbol{\Pi}_{\boldsymbol{a}^{(2)}}^{\perp}$ represent projection matrices onto the vector $\boldsymbol{a}^{(2)}$ and its orthogonal complement space, respectively. Therefore, the right-hand side of (D.201) can be written as

$$
\begin{aligned}
&\left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \cdot \left(\boldsymbol{I}_{M_1} - \frac{1}{M_1} \boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}} \right) \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \\
&= \left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \\
&- \frac{1}{M_1} \left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} - \boldsymbol{J}_1^{(1)} \right) \cdot \boldsymbol{a}^{(1)} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}} \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \\
&= \left(\tilde{\boldsymbol{a}}^{(1)^{\mathrm{T}}} - \tilde{\boldsymbol{a}}^{(2)^{\mathrm{T}}} \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} - \frac{1}{M_1} \left(\boldsymbol{a}^{(1)^{\mathrm{H}}} \boldsymbol{J}_1^{(1)^{\mathrm{H}}} \underbrace{\left(\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} \boldsymbol{a}^{(1)} - \boldsymbol{J}_1^{(1)} \boldsymbol{a}^{(1)} \right)}_{\boldsymbol{0}_{(M_2 - 1) \times 1}} \cdot \boldsymbol{a}^{(1)^{\mathrm{H}}} \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}} \\
&= \left(\tilde{\boldsymbol{a}}^{(1)^{\mathrm{T}}} - \tilde{\boldsymbol{a}}^{(2)^{\mathrm{T}}} \right) \otimes \boldsymbol{a}^{(2)^{\mathrm{H}}}
\end{aligned} \tag{D.204}
$$

where $\boldsymbol{J}_2^{(1)} / \mathrm{e}^{\jmath \mu^{(1)}} \boldsymbol{a}^{(1)} - \boldsymbol{J}_1^{(1)} \boldsymbol{a}^{(1)} = \boldsymbol{0}_{(M_2 - 1) \times 1}$ follows from the fact that $\boldsymbol{a}^{(1)}$ satisfies the shift invariance equation for $r = 1$. This shows that the left-hand side and the right-hand side of (D.201) are equal for $r = 1$. The proof for $r = 2$ proceeds in an analogous fashion. Consequently, we have shown that for $d = 1$

$$
\boldsymbol{r}^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{mat}} = \boldsymbol{r}^{(r)^{\mathrm{T}}} \cdot \boldsymbol{W}_{\mathrm{ten}}, \quad \text{for } r = 1, 2 \tag{D.205}
$$

and hence the MSE for 2-D Standard ESPRIT and 2-D Standard Tensor-ESPRIT are in fact equal. □

Note that even though many terms have canceled in this proof, e.g., the first term of $\boldsymbol{W}_{\mathrm{ten}}$ has canceled completely, the second term has canceled for $r = 2$, and the third term for $r = 1$, this is specific to the special case $d = 1$ we considered here. Therefore, albeit the proof may suggest that the MSE for 2-D Tensor-ESPRIT can be simplified in more general settings, this is unfortunately not the case.

D.18.5. R-D Unitary Tensor-ESPRIT

The fifth and final part of the theorem is to show that the MSE for R-D Unitary ESPRIT is again equal to the MSE for R-D Standard ESPRIT in case of a single source. Again, there is no need to derive this in full detail. As it was shown in Section D.16.2, Forward-Backward-Averaging has no effect on $\boldsymbol{u}_s$ or $\boldsymbol{U}_{\rm n}$ but only affects $\boldsymbol{v}_s$ and $\boldsymbol{V}_{\rm n}$. This carries over to the tensor case where only the quantities involving the symbols are affected. However, since the "symbol part" and the "array part" can always be factorized (cf. equation (D.200)), the arguments from Section D.16.2 can still be applied to prove this part of the theorem. $\square$

Appendix E.

Proofs and derivations for Part IV

E.1. Derivation of the iterative refinement update rule for TENCE

In order to derive the SLS-based update rule for the iterative refinement algorithm for TENCE we first need the following lemma:

Lemma E.1.1. *For a tensor* $\boldsymbol{\mathcal{S}} \in \mathbb{C}^{N_1 \times N_2 \times N_3}$ *and matrices* $\boldsymbol{A} \in \mathbb{C}^{N_1 \times M_1}$ *and* $\boldsymbol{B} \in \mathbb{C}^{N_2 \times M_2}$ *the following identities hold:*

$$\begin{aligned} \operatorname{vec}\left\{\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right\} &= \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \left(\boldsymbol{I}_{M_1} \otimes \left[\boldsymbol{\mathcal{S}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]^{\mathrm{T}}_{(1)}\right) \cdot \operatorname{vec}\left\{\boldsymbol{A}\right\} \\ &= \boldsymbol{P}^{(1)}_{M_1,M_2,N_3} \cdot \left(\boldsymbol{I}_{M_2} \otimes \left[\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}}\right]^{\mathrm{T}}_{(2)}\right) \cdot \operatorname{vec}\left\{\boldsymbol{B}\right\}, \end{aligned} \tag{E.1}$$

where $\boldsymbol{P}^{(n)}_{I,J,K}$ *are the permutation matrices defined in* (4.11).

Proof. From the definition of the permutation matrices we know that

$$\operatorname{vec}\left\{\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right\} = \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \operatorname{vec}\left\{\left[\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]_{(3)}\right\}. \tag{E.2}$$

Applying Theorem 4.1.1, this can be reformulated into

$$\operatorname{vec}\left\{\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right\} = \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \operatorname{vec}\left\{\left[\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]^{\mathrm{T}}_{(1)}\right\}. \tag{E.3}$$

Expanding the 1-mode product with the help of (4.5), we obtain

$$\begin{aligned} \operatorname{vec}\left\{\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right\} &= \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \operatorname{vec}\left\{\left(\boldsymbol{A}^{\mathrm{T}} \cdot \left[\boldsymbol{\mathcal{S}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]_{(1)}\right)^{\mathrm{T}}\right\} \\ &= \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \operatorname{vec}\left\{\left[\boldsymbol{\mathcal{S}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]^{\mathrm{T}}_{(1)} \cdot \boldsymbol{A}\right\}. \end{aligned} \tag{E.4}$$

We can now use property (3.7) to isolate $\boldsymbol{A}$. We get

$$\operatorname{vec}\left\{\boldsymbol{\mathcal{S}} \times_1 \boldsymbol{A}^{\mathrm{T}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right\} = \boldsymbol{P}^{(3)}_{M_1,M_2,N_3} \cdot \left(\boldsymbol{I}_{M_1} \otimes \left[\boldsymbol{\mathcal{S}} \times_2 \boldsymbol{B}^{\mathrm{T}}\right]^{\mathrm{T}}_{(1)}\right) \cdot \operatorname{vec}\left\{\boldsymbol{A}\right\}, \tag{E.5}$$

which is the first line of the lemma. The proof of the second line is accomplished in a similar fashion. □

We start by expressing the cost function (15.29) in a more compact form by applying identity (4.1) for the higher-order norm of tensors. Then, we obtain the following alternative representation of (15.29)

$$J(\Delta \boldsymbol{H}_k) = \left\| \left[\ \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\}^{\mathrm{T}}, \ \kappa_1 \cdot \mathrm{vec}\left\{\Delta \boldsymbol{H}_{1,k}\right\}^{\mathrm{T}}, \ \kappa_2 \cdot \mathrm{vec}\left\{\Delta \boldsymbol{H}_{2,k}\right\}^{\mathrm{T}}, \ \right]^{\mathrm{T}} \right\|_2^2. \tag{E.6}$$

Next, we expand the residual tensor $\boldsymbol{\mathcal{R}}_{k+1}$ in terms of the updates for the channel matrices. Using the update rules (15.31) and (15.32) in (15.30) for $\boldsymbol{\mathcal{R}}_{k+1}$ we obtain

$$\begin{aligned}
\boldsymbol{\mathcal{R}}_{k+1} =& \tilde{\boldsymbol{\mathcal{Y}}}_1 - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k+1}\right)^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_{k+1}\right)^{\mathrm{T}} \\
=& \tilde{\boldsymbol{\mathcal{Y}}}_1 - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k} + \Delta\Delta \boldsymbol{H}_{1,k}\right)^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_k + \Delta\Delta \boldsymbol{H}_k\right)^{\mathrm{T}} \\
=& \tilde{\boldsymbol{\mathcal{Y}}}_1 - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k}\right)^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_k\right)^{\mathrm{T}} - \boldsymbol{\mathcal{G}} \times_1 \Delta\Delta \boldsymbol{H}_{1,k}^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_k\right)^{\mathrm{T}} \\
& - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k}\right)^{\mathrm{T}} \times_2 \Delta\Delta \boldsymbol{H}_k^{\mathrm{T}} - \boldsymbol{\mathcal{G}} \times_1 \Delta\Delta \boldsymbol{H}_{1,k}^{\mathrm{T}} \times_2 \Delta\Delta \boldsymbol{H}_k^{\mathrm{T}} \\
\approx& \boldsymbol{\mathcal{R}}_k - \boldsymbol{\mathcal{G}} \times_1 \Delta\Delta \boldsymbol{H}_{1,k}^{\mathrm{T}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_k\right)^{\mathrm{T}} - \boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k}\right)^{\mathrm{T}} \times_2 \Delta\Delta \boldsymbol{H}_k^{\mathrm{T}},
\end{aligned} \tag{E.7}$$

where in the last step we have neglected the higher-order terms in $\Delta\Delta \boldsymbol{H}_{1,k}$ and $\Delta\Delta \boldsymbol{H}_{2,k}$. Therefore (E.7) is a linear function in these terms. In order to use this linear function in (E.6), we apply the vec-operator and use Lemma E.1.1 to reorder the terms. Then,

$$\begin{aligned}
\mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_{k+1}\right\} \approx \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\} &- \boldsymbol{P}^{(3)}_{M_1, M_1+M_2, N_{\mathrm{R}}} \cdot \left(\boldsymbol{I}_{M_1} \otimes \left[\boldsymbol{\mathcal{G}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta \boldsymbol{H}_k\right)^{\mathrm{T}}\right]_{(1)}^{\mathrm{T}}\right) \cdot \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_{1,k}\right\} \\
&- \boldsymbol{P}^{(1)}_{M_1, M_1+M_2, N_{\mathrm{R}}} \cdot \left(\boldsymbol{I}_{M_1+M_2} \otimes \left[\boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta \boldsymbol{H}_{1,k}\right)^{\mathrm{T}}\right]_{(2)}^{\mathrm{T}}\right) \cdot \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_k\right\}.
\end{aligned}$$

In order to separate the update terms $\Delta\Delta \boldsymbol{H}_{1,k}$ and $\Delta\Delta \boldsymbol{H}_{2,k}$ we apply the following identity

$$\mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_k\right\} = \mathrm{vec}\left\{\left[\Delta\Delta \boldsymbol{H}_{1,k}, \ \Delta\Delta \boldsymbol{H}_{2,k}\right]\right\} = \begin{bmatrix} \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_{2,k}\right\} \end{bmatrix}, \tag{E.8}$$

which follows from the definition of the vec-operator. Equation (E.8) allows to express the update equation for the residual tensor $\boldsymbol{\mathcal{R}}_k$ in the following convenient fashion

$$\mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_{k+1}\right\} = \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\} - \boldsymbol{F}_k^{(1)} \cdot \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_{1,k}\right\} - \boldsymbol{F}_k^{(2)} \cdot \mathrm{vec}\left\{\Delta\Delta \boldsymbol{H}_{2,k}\right\}, \tag{E.9}$$

where the matrices $\boldsymbol{F}_k^{(1)}$ and $\boldsymbol{F}_k^{(2)}$ are given by

$$\begin{aligned}\boldsymbol{F}_k^{(1)} =& \boldsymbol{P}^{(3)}_{M_1,M_1+M_2,N_{\mathrm{R}}} \cdot \left(\boldsymbol{I}_{M_1} \otimes \left[\boldsymbol{\mathcal{G}} \times_2 \left(\hat{\boldsymbol{H}} + \Delta\boldsymbol{H}_k\right)^{\mathrm{T}}\right]^{\mathrm{T}}_{(1)}\right) \\ &+ \boldsymbol{P}^{(1)}_{M_1,M_1+M_2,N_{\mathrm{R}}} \cdot \left(\boldsymbol{I}_{M_1+M_2} \otimes \left[\boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta\boldsymbol{H}_{1,k}\right)^{\mathrm{T}}\right]^{\mathrm{T}}_{(2)}\right) \cdot \boldsymbol{J}_1 \qquad \text{(E.10)}\end{aligned}$$

$$\boldsymbol{F}_k^{(2)} = \boldsymbol{P}^{(1)}_{M_1,M_1+M_2,N_{\mathrm{R}}} \cdot \left(\boldsymbol{I}_{M_1+M_2} \otimes \left[\boldsymbol{\mathcal{G}} \times_1 \left(\hat{\boldsymbol{H}}_1 + \Delta\boldsymbol{H}_{1,k}\right)^{\mathrm{T}}\right]^{\mathrm{T}}_{(2)}\right) \cdot \boldsymbol{J}_2 \qquad \text{(E.11)}$$

$$\boldsymbol{J}_1 = \begin{bmatrix} \boldsymbol{I}_{M_1 \cdot M_{\mathrm{R}}} \\ \boldsymbol{0}_{M_2 \cdot M_{\mathrm{R}} \times M_1 \cdot M_{\mathrm{R}}} \end{bmatrix} \quad \boldsymbol{J}_2 = \begin{bmatrix} \boldsymbol{0}_{M_1 \cdot M_{\mathrm{R}} \times M_2 \cdot M_{\mathrm{R}}} \\ \boldsymbol{I}_{M_2 \cdot M_{\mathrm{R}}} \end{bmatrix}.$$

Next, we insert (E.9) as well as (15.31) and (15.32) into the cost function (E.6) for the $(k+1)$-th iteration which yields

$$J(\Delta\boldsymbol{H}_{k+1}) = \left\| \begin{bmatrix} \operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\} \\ \kappa_1 \cdot \operatorname{vec}\left\{\Delta\boldsymbol{H}_{1,k}\right\} \\ \kappa_2 \cdot \operatorname{vec}\left\{\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix} + \boldsymbol{F}_k \cdot \begin{bmatrix} \operatorname{vec}\left\{\Delta\Delta\boldsymbol{H}_{1,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix} \right\|_2^2$$

$$\boldsymbol{F}_k = \begin{bmatrix} -\boldsymbol{F}_k^{(1)} & -\boldsymbol{F}_k^{(2)} \\ \kappa_1 \cdot \boldsymbol{I}_{M_1 \cdot M_{\mathrm{R}}} & \boldsymbol{0}_{M_1 \cdot M_{\mathrm{R}} \times M_2 \cdot M_{\mathrm{R}}} \\ \boldsymbol{0}_{M_2 \cdot M_{\mathrm{R}} \times M_1 \cdot M_{\mathrm{R}}} & \kappa_2 \cdot \boldsymbol{I}_{M_2 \cdot M_{\mathrm{R}}} \end{bmatrix}. \qquad \text{(E.12)}$$

Consequently, the cost function has been rewritten as a linear least squares problem in the update terms $\Delta\Delta\boldsymbol{H}_{1,k}$ and $\Delta\Delta\boldsymbol{H}_{2,k}$. Therefore, the least squares solution of (E.12) with respect to these terms is given by

$$\begin{bmatrix} \operatorname{vec}\left\{\Delta\Delta\boldsymbol{H}_{1,k}\right\} \\ \operatorname{vec}\left\{\Delta\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix} = -\boldsymbol{F}_k^{+} \cdot \begin{bmatrix} \operatorname{vec}\left\{\boldsymbol{\mathcal{R}}_k\right\} \\ \kappa_1 \cdot \operatorname{vec}\left\{\Delta\boldsymbol{H}_{1,k}\right\} \\ \kappa_2 \cdot \operatorname{vec}\left\{\Delta\boldsymbol{H}_{2,k}\right\} \end{bmatrix}. \qquad \text{(E.13)}$$

E.2. Proof of Proposition 16.1.1

We want to show that the choice of γ given in (16.6) provides a transmit power that is upper-bounded by $P_{\mathrm{T,R}}$. To this end we start from the average power transmitted by the relay given by (16.5) which is replicated here for convenience

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \gamma^2 \cdot \operatorname{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{R}_1 \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{R}_2 \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\}$$

Since the trace is linear we can break this expression into the sum of three trace operations

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{R}_1 \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{R}_2 \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} \tag{E.14}$$

Next, we rearrange the first two terms using the rule $\mathrm{trace}\{\boldsymbol{A} \cdot \boldsymbol{B}\} = \mathrm{trace}\{\boldsymbol{B} \cdot \boldsymbol{A}\}$ and obtain

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{R}_1 \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})}\right\} + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{R}_2 \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})}\right\} + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} \tag{E.15}$$

To upper-bound (E.15) we apply the bound $\mathrm{trace}\{\boldsymbol{A} \cdot \boldsymbol{B}\} \leq \mathrm{trace}\{\boldsymbol{A}\} \cdot \mathrm{trace}\{\boldsymbol{B}\}$. Since $\mathrm{trace}\{\boldsymbol{R}_i\} = P_{\mathrm{T},i}$ this step yields

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \gamma^2 \cdot P_{\mathrm{T},1} \cdot \mathrm{trace}\left\{\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})}\right\} + \gamma^2 \cdot P_{\mathrm{T},2} \cdot \mathrm{trace}\left\{\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})}\right\} + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} \tag{E.16}$$

Applying the rule $\mathrm{trace}\{\boldsymbol{A} \cdot \boldsymbol{B}\} = \mathrm{trace}\{\boldsymbol{B} \cdot \boldsymbol{A}\}$ again, we have

$$\begin{aligned}\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} &\leq \gamma^2 \cdot P_{\mathrm{T},1} \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} + \gamma^2 \cdot P_{\mathrm{T},2} \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} \\ &\quad + \gamma^2 \cdot \mathrm{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\} \\ &= \gamma^2 \cdot \mathrm{trace}\left\{P_{\mathrm{T},1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + P_{\mathrm{T},2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\}\end{aligned} \tag{E.17}$$

Finally, substituting γ from (16.6) into (E.17) we have

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} \leq P_{\mathrm{T,R}} \tag{E.18}$$

which proves the proposition.

E.3. Proof of Proposition 16.1.2

We want to show that the relay's transmit power using γ from (16.6) converges to $\frac{P_{\mathrm{T,R}}}{\min\{M_{\mathrm{U}}, M_{\mathrm{R}}\}}$ for high SNRs if the users apply spatial multiplexing to transmit their data. We begin with the case where both effective channels $\boldsymbol{H}_{1,2}^{(\mathrm{e})}$ and $\boldsymbol{H}_{2,1}^{(\mathrm{e})}$ are square and full rank, which implies $M_{\mathrm{U}} \leq M_{\mathrm{R}}$. In this case, the capacity-achieving power distribution across the M_{U} eigenmodes approaches a uniform power allocation as the SNR approaches infinity. Consequently, the input covariance matrices converge to a scaled identity, i.e., $\lim_{P_{\rm N}\to 0} \boldsymbol{R}_i = \frac{P_{{\rm T},i}}{M_{\rm U}} \cdot \boldsymbol{I}_{M_{\rm U}}$, since $\operatorname{trace}\{\boldsymbol{R}_i\} = P_{{\rm T},i}$. Inserting the limiting input covariance matrices into (16.5) we obtain

$$\begin{aligned}\lim_{P_{\rm N}\to 0} \mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} &= \gamma^2 \cdot \operatorname{trace}\left\{\frac{P_{\rm T,1}}{M_{\rm U}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \frac{P_{\rm T,2}}{M_{\rm U}} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\rm f)} \cdot \boldsymbol{H}_2^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \boldsymbol{G} \cdot \boldsymbol{R}_{\rm N,R} \cdot \boldsymbol{G}^{\rm H}\right\} \\ &= \frac{\gamma^2 \cdot}{M_{\rm U}} \operatorname{trace}\left\{P_{\rm T,1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + P_{\rm T,2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\rm f)} \cdot \boldsymbol{H}_2^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \boldsymbol{G} \cdot \boldsymbol{R}_{\rm N,R} \cdot \boldsymbol{G}^{\rm H}\right\}.\end{aligned}$$

If we choose γ according to (16.6) we have

$$\begin{aligned}\lim_{P_{\rm N}\to 0} \mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} &= \frac{P_{\rm T,R}}{M_{\rm U}} \cdot \frac{\operatorname{trace}\left\{P_{\rm T,1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + P_{\rm T,2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\rm f)} \cdot \boldsymbol{H}_2^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \boldsymbol{G} \cdot \boldsymbol{R}_{\rm N,R} \cdot \boldsymbol{G}^{\rm H}\right\}}{\operatorname{trace}\left\{P_{\rm T,1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + P_{\rm T,2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\rm f)} \cdot \boldsymbol{H}_2^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \boldsymbol{G} \cdot \boldsymbol{R}_{\rm N,R} \cdot \boldsymbol{G}^{\rm H}\right\}} \\ &= \frac{P_{\rm T,R}}{M_{\rm U}} \qquad \text{(E.19)}\end{aligned}$$

which proves the proposition for the first case.

For $M_{\rm R} < M_{\rm U}$ the effective channel matrices become rank-deficient, since their rank is at most[1] $M_{\rm R}$. For $\operatorname{rank}(\boldsymbol{H}_{1,2}^{(\rm e)}) = \operatorname{rank}(\boldsymbol{H}_{2,1}^{(\rm e)}) = M_{\rm R}$, the capacity-achieving transmit covariance matrices become $\boldsymbol{R}_i = \frac{P_{{\rm T},i}}{M_{\rm R}} \cdot \boldsymbol{V}_{3-i,i}^{[\rm s]} \boldsymbol{V}_{3-i,i}^{[\rm s]^{\rm H}}$, where $\boldsymbol{V}_{j,i}^{[\rm s]}$ represents the $M_{\rm R}$ dominant right singular vectors of the effective channel matrix $\boldsymbol{H}_{j,i}^{(\rm e)}$. Consequently, it converges to a scaled version of the projection matrix onto the row space of $\boldsymbol{H}_{j,i}^{(\rm e)}$. However, since $\boldsymbol{H}_{j,i}^{(\rm e)} = \boldsymbol{H}_j^{(\rm b)} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_i^{(\rm f)}$, the row space of $\boldsymbol{H}_{j,i}^{(\rm e)}$ coincides with the row space of $\boldsymbol{H}_i^{(\rm f)}$. Therefore we have

$$\lim_{P_{\rm N}\to 0} \operatorname{trace}\left\{\boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{R}_1 \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H}\right\} = \lim_{P_{\rm N}\to 0} \operatorname{trace}\left\{\boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{R}_1\right\} \qquad \text{(E.20)}$$

$$= \operatorname{trace}\left\{\boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)}\right\} \cdot \frac{P_{\rm T,1}}{M_{\rm R}} = \frac{1}{M_{\rm R}} \cdot \operatorname{trace}\left\{P_{\rm T,1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H}\right\} \qquad \text{(E.21)}$$

where in the first step we have used the identity $\operatorname{trace}\{\boldsymbol{A} \cdot \boldsymbol{B}\} = \operatorname{trace}\{\boldsymbol{B} \cdot \boldsymbol{A}\}$ and in the second step the fact that $\boldsymbol{R}_1$ converges to a scaled version of the projection matrix onto the row space of $\boldsymbol{H}_1^{(\rm f)}$, which leaves the matrix inside the trace unaltered, except for the scaling. Proceeding in the same manner with the second term we eventually find

$$\lim_{P_{\rm N}\to 0} \mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \frac{1}{M_{\rm R}} \cdot \gamma^2 \cdot \operatorname{trace}\left\{P_{\rm T,1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\rm f)} \cdot \boldsymbol{H}_1^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + P_{\rm T,2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\rm f)} \cdot \boldsymbol{H}_2^{(\rm f)^{\rm H}} \cdot \boldsymbol{G}^{\rm H} + \boldsymbol{G} \cdot \boldsymbol{R}_{\rm N,R} \cdot \boldsymbol{G}^{\rm H}\right\}$$

[1] It is lower only if the channel matrices have rank-deficiencies (which happens with probability zero if channels are drawn from continuous distributions) or the relay amplification matrix is rank-deficient. However, in this case, the bound becomes simply tighter, as we can replace $\min\{M_{\rm U}, M_{\rm R}\}$ with the actual rank of the effective channel matrices. The proof still follows the same lines as the one presented here.

$$= \frac{P_{\mathrm{T,R}}}{M_{\mathrm{R}}} \cdot \frac{\mathrm{trace}\left\{P_{\mathrm{T},1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + P_{\mathrm{T},2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\}}{\mathrm{trace}\left\{P_{\mathrm{T},1} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + P_{\mathrm{T},2} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}^{\mathrm{H}} + \boldsymbol{G} \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}^{\mathrm{H}}\right\}}$$
$$= \frac{P_{\mathrm{T,R}}}{M_{\mathrm{R}}} \tag{E.22}$$

which proves the proposition for the second case.

E.4. Proof of Lemma 16.1.4

Let us start with the power of the desired signal components. For the received power at UT_1 $P_{\mathrm{R},1}$, we expand (16.9) and obtain

$$P_{\mathrm{R},1} = \mathbb{E}\left\{\left|h_{1,2}^{(\mathrm{e})} \cdot x_2\right|^2\right\} = \left|h_{1,2}^{(\mathrm{e})}\right|^2 \cdot \underbrace{\mathbb{E}\left\{|x_2|^2\right\}}_{P_{\mathrm{T},2}} = \left|\boldsymbol{h}_1^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{h}_2^{(\mathrm{f})}\right|^2 \cdot P_{\mathrm{T},2}$$
$$= \boldsymbol{h}_1^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{h}_2^{(\mathrm{f})} \cdot \boldsymbol{h}_2^{(\mathrm{f})^{\mathrm{H}}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}} \cdot \boldsymbol{h}_1^{(\mathrm{b})^*} \cdot P_{\mathrm{T},2}, \tag{E.23}$$

where in the last step we have used the fact that $|z|^2 = z \cdot z^*$ for arbitrary $z \in \mathbb{C}$. Equation (E.23) is quadratic in the elements of $\boldsymbol{G}_\gamma$. To transform it into a canonical quadratic form in $\boldsymbol{g}_\gamma = \mathrm{vec}\{\boldsymbol{G}_\gamma\}$ we can directly apply (3.41) and obtain

$$P_{\mathrm{R},1} = \mathrm{vec}\{\boldsymbol{G}_\gamma\}^{\mathrm{H}} \cdot \left[\left(\boldsymbol{h}_2^{(\mathrm{f})} \cdot \boldsymbol{h}_2^{(\mathrm{f})^{\mathrm{H}}}\right)^{\mathrm{T}} \otimes \left(\boldsymbol{h}_1^{(\mathrm{b})^*} \cdot \boldsymbol{h}_1^{(\mathrm{b})^{\mathrm{T}}}\right)\right] \cdot \mathrm{vec}\{\boldsymbol{G}_\gamma\} \cdot P_{\mathrm{T},2}$$
$$= \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \underbrace{\left[\left(\boldsymbol{h}_2^{(\mathrm{f})} \cdot \boldsymbol{h}_2^{(\mathrm{f})^{\mathrm{H}}}\right) \otimes \left(\boldsymbol{h}_1^{(\mathrm{b})} \cdot \boldsymbol{h}_1^{(\mathrm{b})^{\mathrm{H}}}\right)\right]^{\mathrm{T}}}_{\boldsymbol{K}_{2,1}} \cdot \boldsymbol{g}_\gamma \cdot P_{\mathrm{T},2}, \tag{E.24}$$

which shows (16.10). The corresponding expansion of (16.9) in terms of $P_{\mathrm{R},2}$ yields (16.11) using similar arguments.

Concerning the power of the effective noise component $\tilde{n}_i$, we expand $\tilde{P}_{\mathrm{N},i} = \mathbb{E}\left\{|\tilde{n}_i|^2\right\}$ and find

$$\tilde{P}_{\mathrm{N},i} = \mathbb{E}\left\{\left|\boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{n}_{\mathrm{R}} + n_i\right|^2\right\}$$
$$= \mathbb{E}\left\{\boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{n}_{\mathrm{R}} \cdot \boldsymbol{n}_{\mathrm{R}}^{\mathrm{H}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}} \cdot \boldsymbol{h}_i^{(\mathrm{b})^*}\right\} + \mathbb{E}\left\{|n_i|^2\right\}$$
$$= \boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \underbrace{\mathbb{E}\left\{\boldsymbol{n}_{\mathrm{R}} \cdot \boldsymbol{n}_{\mathrm{R}}^{\mathrm{H}}\right\}}_{\boldsymbol{R}_{\mathrm{N,R}}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}} \cdot \boldsymbol{h}_i^{(\mathrm{b})^*} + \underbrace{\mathbb{E}\left\{|n_i|^2\right\}}_{P_{\mathrm{N},i}}$$
$$= \boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{T}}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{R}_{\mathrm{N,R}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}} \cdot \boldsymbol{h}_i^{(\mathrm{b})^*} + P_{\mathrm{N},i}, \tag{E.25}$$

where in the first step we have used the fact that the noise at the relay is independent of the noise at the terminals. As before, we can apply (3.41) to transform the first term of (E.25) into a canonical quadratic form and obtain

$$\begin{aligned}\tilde{P}_{\mathrm{N},i} &= \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \left[\boldsymbol{R}_{\mathrm{N,R}}^{\mathrm{T}} \otimes \left(\boldsymbol{h}_i^{(\mathrm{b})^*} \cdot \boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{T}}}\right)\right] \cdot \boldsymbol{g}_\gamma + P_{\mathrm{N},i} \\ &= \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \underbrace{\left[\boldsymbol{R}_{\mathrm{N,R}} \otimes \left(\boldsymbol{h}_i^{(\mathrm{b})} \cdot \boldsymbol{h}_i^{(\mathrm{b})^{\mathrm{H}}}\right)\right]^{\mathrm{T}}}_{\boldsymbol{J}_i} \cdot \boldsymbol{g}_\gamma + P_{\mathrm{N},i},\end{aligned} \tag{E.26}$$

which shows (16.12). Finally, concerning the transmit power of the relay, we expand $\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\}$ by inserting (14.2). We obtain

$$\begin{aligned}\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} &= \mathbb{E}\left\{\|\boldsymbol{G}_\gamma \cdot \boldsymbol{r}\|_2^2\right\} = \mathbb{E}\left\{\boldsymbol{r}^{\mathrm{H}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}} \cdot \boldsymbol{G}_\gamma \cdot \boldsymbol{r}^2\right\} = \mathbb{E}\left\{\mathrm{trace}\left\{\boldsymbol{G}_\gamma \cdot \boldsymbol{r} \cdot \boldsymbol{r}^{\mathrm{H}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}}\right\}\right\} \\ &= \mathrm{trace}\left\{\boldsymbol{G}_\gamma \cdot \mathbb{E}\left\{\boldsymbol{r} \cdot \boldsymbol{r}^{\mathrm{H}}\right\} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}}\right\} = \mathrm{trace}\left\{\boldsymbol{G}_\gamma \cdot \boldsymbol{R}_{\mathrm{R}} \cdot \boldsymbol{G}_\gamma^{\mathrm{H}}\right\},\end{aligned} \tag{E.27}$$

where we have applied the identity $\boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{x} = \mathrm{trace}\left\{\boldsymbol{x} \cdot \boldsymbol{x}^{\mathrm{H}}\right\}$, which holds for arbitrary $\boldsymbol{x} \in \mathbb{C}^N$. Equation (E.27) is a special case of (3.40) and can hence be transformed into a canonical quadratic form as well. Applying (3.42) for $\boldsymbol{R} = \boldsymbol{R}_{\mathrm{R}}$ and $\boldsymbol{X} = \boldsymbol{G}_\gamma$ we directly obtain

$$\mathbb{E}\left\{\|\bar{\boldsymbol{r}}\|_2^2\right\} = \boldsymbol{g}_\gamma^{\mathrm{H}} \cdot \left(\boldsymbol{R}_{\mathrm{R}}^{\mathrm{T}} \otimes \boldsymbol{I}_{M_{\mathrm{R}}}\right) \cdot \boldsymbol{g}_\gamma, \tag{E.28}$$

which shows (16.13).

E.5. Derivation of ANOMAX

The ANOMAX solution is easily derived via algebraic manipulation of the cost function (16.16) which is restated here for convenience

$$\boldsymbol{G} = \underset{\boldsymbol{G} \mid \|\boldsymbol{G}\|_{\mathrm{F}}=1}{\arg\max} \left(\left\|\boldsymbol{H}_1^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})}\right\|_{\mathrm{F}}^2 + \left\|\boldsymbol{H}_2^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_1^{(\mathrm{f})}\right\|_{\mathrm{F}}^2\right). \tag{E.29}$$

Since $\|\boldsymbol{X}\|_{\mathrm{F}} = \|\mathrm{vec}\{\boldsymbol{X}\}\|_2$ for any matrix $\boldsymbol{X}$ we can rewrite (E.29) in terms of $\boldsymbol{g} = \mathrm{vec}\{\boldsymbol{G}\}$. Applying the rule $\mathrm{vec}\left\{\boldsymbol{A} \cdot \boldsymbol{X} \cdot \boldsymbol{B}^{\mathrm{T}}\right\} = (\boldsymbol{B} \otimes \boldsymbol{A}) \cdot \mathrm{vec}\{\boldsymbol{X}\}$ (cf. equation 3.7), we obtain

$$\boldsymbol{g} = \underset{\boldsymbol{g} \mid \|\boldsymbol{g}\|_2=1}{\arg\max} \left(\left\|\left(\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_1^{(\mathrm{b})}\right) \cdot \boldsymbol{g}\right\|_2^2 + \left\|\left(\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_2^{(\mathrm{b})}\right) \cdot \boldsymbol{g}\right\|_2^2\right). \tag{E.30}$$

Expanding the two-norms via $\|\boldsymbol{x}\|_2^2 = \boldsymbol{x}^{\mathrm{H}} \cdot \boldsymbol{x}$ we get

$$\begin{aligned}
\boldsymbol{g} &= \underset{\boldsymbol{g}\|\|\boldsymbol{g}\|_2=1}{\arg\max} \quad \Big(\boldsymbol{g}^{\mathrm{H}} \cdot \Big(\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_1^{(\mathrm{b})}\Big)^{\mathrm{H}} \cdot \Big(\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_1^{(\mathrm{b})}\Big) \cdot \boldsymbol{g} \\
&\qquad + \boldsymbol{g}^{\mathrm{H}} \cdot \Big(\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_2^{(\mathrm{b})}\Big)^{\mathrm{H}} \cdot \Big(\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_2^{(\mathrm{b})}\Big) \cdot \boldsymbol{g}\Big) \qquad \text{(E.31)} \\
&= \underset{\boldsymbol{g}\|\|\boldsymbol{g}\|_2=1}{\arg\max} \quad \boldsymbol{g}^{\mathrm{H}} \cdot \Big[\Big(\boldsymbol{H}_2^{(\mathrm{f})^*} \otimes \boldsymbol{H}_1^{(\mathrm{b})^{\mathrm{H}}}\Big) \cdot \Big(\boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_1^{(\mathrm{b})}\Big) + \Big(\boldsymbol{H}_1^{(\mathrm{f})^*} \otimes \boldsymbol{H}_2^{(\mathrm{b})^{\mathrm{H}}}\Big) \cdot \Big(\boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{T}}} \otimes \boldsymbol{H}_2^{(\mathrm{b})}\Big)\Big] \cdot \boldsymbol{g} \\
&= \underset{\boldsymbol{g}\|\|\boldsymbol{g}\|_2=1}{\arg\max} \quad \boldsymbol{g}^{\mathrm{H}} \cdot \underbrace{\Big[\Big(\boldsymbol{H}_2^{(\mathrm{f})^*} \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{T}}}\Big) \otimes \Big(\boldsymbol{H}_1^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_1^{(\mathrm{b})}\Big) + \Big(\boldsymbol{H}_1^{(\mathrm{f})^*} \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{T}}}\Big) \otimes \Big(\boldsymbol{H}_2^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_2^{(\mathrm{b})}\Big)\Big]}_{\boldsymbol{R}_{\mathrm{ANO}}} \cdot \boldsymbol{g} \\
&= \underset{\boldsymbol{g}\|\|\boldsymbol{g}\|_2=1}{\arg\max} \quad \boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{R}_{\mathrm{ANO}} \cdot \boldsymbol{g} \qquad \text{(E.32)}
\end{aligned}$$

where from the second to the third step we have used the rule $(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot (\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{A} \cdot \boldsymbol{C}) \otimes (\boldsymbol{B} \cdot \boldsymbol{D})$. As shown in Section 3.2, the solution to (E.32) is given by eigenvector of $\boldsymbol{R}_{\mathrm{ANO}}$ which corresponds to its largest eigenvalue, scaled to norm one. This proves the first form of ANOMAX.

To prove the second and the third form it is sufficient to see that $\left(\boldsymbol{K}_{\mathrm{ANO}}^{\mathrm{T}}\right)^{\mathrm{H}} \cdot \boldsymbol{K}_{\mathrm{ANO}}^{\mathrm{T}} = \boldsymbol{R}_{\mathrm{ANO}}$, where $\boldsymbol{K}_{\mathrm{ANO}}$ is defined in (16.19). Therefore, the eigenvectors of $\boldsymbol{R}_{\mathrm{ANO}}$ coincide with the right singular vectors of $\boldsymbol{K}_{\mathrm{ANO}}^{\mathrm{T}}$ (cf. Section 3.2, equation (3.51)) and the conjugates of the left singular vectors of $\boldsymbol{K}_{\mathrm{ANO}}$.

E.6. Proof of Proposition 16.3.1

Here we would like to show that DCM and ANOMAX provide the identical relay amplification matrix for the special case of single antenna terminals $M_1 = M_2 = 1$. For notational convenience, we consider the case where reciprocity holds, such that $\boldsymbol{h}_1^{(\mathrm{f})} = \boldsymbol{h}_1^{(\mathrm{b})^{\mathrm{T}}} = \boldsymbol{h}_1$ and $\boldsymbol{h}_2^{(\mathrm{f})} = \boldsymbol{h}_2^{(\mathrm{b})^{\mathrm{T}}} = \boldsymbol{h}_2$. The extension to the non-reciprocal case follows the same lines as this proof. In the reciprocal case, the matrix $\boldsymbol{R}_{\mathrm{ANO}}$ takes the following simpler form

$$\boldsymbol{R}_{\mathrm{ANO}} = \left(\boldsymbol{h}_2 \cdot \boldsymbol{h}_2^{\mathrm{H}}\right)^{\mathrm{T}} \otimes \left(\boldsymbol{h}_1 \cdot \boldsymbol{h}_1^{\mathrm{H}}\right)^{\mathrm{T}} + \left(\boldsymbol{h}_1 \cdot \boldsymbol{h}_1^{\mathrm{H}}\right)^{\mathrm{T}} \otimes \left(\boldsymbol{h}_2 \cdot \boldsymbol{h}_2^{\mathrm{H}}\right)^{\mathrm{T}} \qquad \text{(E.33)}$$

To prove proposition 16.3.1 we first require the following lemma:

Lemma E.6.1. *If the channel vectors are not colinear, then the matrix $\boldsymbol{R}_{\mathrm{ANO}}$ has an ordered* *eigendecomposition given by*

$$\boldsymbol{R}_{\text{ANO}} = [\boldsymbol{q}_1,\ \boldsymbol{q}_2] \cdot \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \cdot [\boldsymbol{q}_1,\ \boldsymbol{q}_2]^{\text{H}}, \tag{E.34}$$

where the eigenvectors $\boldsymbol{q}_n$ *and the eigenvalues* λ_n *are*

$$\boldsymbol{q}_1 = \frac{1}{\sqrt{2} \cdot \sqrt{\|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 + \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2}} \cdot \Big((\boldsymbol{h}_1^* \otimes \boldsymbol{h}_2^*) + (\boldsymbol{h}_2^* \otimes \boldsymbol{h}_1^*) \Big) \tag{E.35}$$

$$\boldsymbol{q}_2 = \frac{1}{\sqrt{2} \cdot \sqrt{\|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 - \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2}} \cdot \Big((\boldsymbol{h}_1^* \otimes \boldsymbol{h}_2^*) - (\boldsymbol{h}_2^* \otimes \boldsymbol{h}_1^*) \Big) \tag{E.36}$$

$$\lambda_1 = \|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 + \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 \tag{E.37}$$

$$\lambda_2 = \|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 - \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2. \tag{E.38}$$

Otherwise, the eigendecomposition reduces to $\boldsymbol{R}_{\text{ANO}} = \boldsymbol{q}_1 \cdot \lambda_1 \cdot \boldsymbol{q}_1^{\text{H}}$.

Proof. To prove this lemma we require the following steps:

1. Show that $\boldsymbol{R}_{\text{ANO}}$ is at most rank two. This is obvious from (E.33): $\boldsymbol{R}_{\text{ANO}}$ is the sum of two matrices which are both rank-one, since the Kronecker product of rank-one matrices is again rank-one.

2. Show that $\boldsymbol{R}_{\text{ANO}} \cdot \boldsymbol{q}_n = \lambda_n \cdot \boldsymbol{q}_n$ for $n = 1, 2$. For instance, for $n = 1$ this is achieved by multiplying (E.33) and (E.35) out and simplifying the four resulting cross terms via the rule $(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot (\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{A} \cdot \boldsymbol{C}) \otimes (\boldsymbol{B} \cdot \boldsymbol{D})$.

3. Show that $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ are orthonormal, i.e, $\boldsymbol{q}_k^{\text{H}} \cdot \boldsymbol{q}_\ell = \delta[k - \ell]$. This step requires inserting (E.35) and (E.36) and simplifying the result as in the previous step.

4. Show that $\lambda_1 \geq \lambda_2$. This is again obvious since

$$\begin{aligned} \|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 + \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 &\geq \|\boldsymbol{h}_1\|_2^2 \cdot \|\boldsymbol{h}_2\|_2^2 - \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 \\ \Leftrightarrow \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 \geq - \left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 &\Leftrightarrow 2\left|\boldsymbol{h}_1^{\text{H}} \cdot \boldsymbol{h}_2\right|^2 \geq 0 \end{aligned} \tag{E.39}$$

5. For the special case of colinear channels, λ_2 becomes zero and $\boldsymbol{R}_{\text{ANO}}$ reduces to a rank-one matrix. This is also seen from (E.33) since the two rank-one matrices are identical in this case.

□

Lemma E.6.1 provides a lot of insight into the algebraic structure of the ANOMAX solution for the single-antenna case. Since ANOMAX selects $\boldsymbol{g}$ as the dominant eigenvector of $\boldsymbol{R}_{\mathrm{ANO}}$ we can directly conclude that it will be equal to $\boldsymbol{q}_1$. However, the Dual Channel Matching (DCM) solution for the general case, which reads as

$$\boldsymbol{G}_{\mathrm{DCM}} = \boldsymbol{H}_2^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_1^{(\mathrm{f})^{\mathrm{H}}} + \boldsymbol{H}_1^{(\mathrm{b})^{\mathrm{H}}} \cdot \boldsymbol{H}_2^{(\mathrm{f})^{\mathrm{H}}} \tag{E.40}$$

simplifies to

$$\boldsymbol{G}_{\mathrm{DCM}} = \boldsymbol{h}_2^* \cdot \boldsymbol{h}_1^{\mathrm{H}} + \boldsymbol{h}_1^* \cdot \boldsymbol{h}_2^{\mathrm{H}} \tag{E.41}$$

for the single-antenna reciprocal case which is discussed here. Consequently, $\mathrm{vec}\{\boldsymbol{G}_{\mathrm{DCM}}\} = (\boldsymbol{h}_1^* \otimes \boldsymbol{h}_2^*) + (\boldsymbol{h}_2^* \otimes \boldsymbol{h}_1^*)$ which is a scaled version of $\boldsymbol{q}_1$. This already proves proposition 16.3.1.

We additionally observe that the maximum value of the ANOMAX cost function is equal to λ_1. This shows that ANOMAX is most efficient for the case of colinear channels and least efficient for the case of orthogonal channels where $\boldsymbol{h}_1^{\mathrm{H}} \cdot \boldsymbol{h}_2 = 0$. In fact, for this special case we have $\lambda_1 = \lambda_2$ so instead of $\boldsymbol{q}_1$ we could also choose $\boldsymbol{q}_2$ or any affine combination of $w \cdot \boldsymbol{q}_1 + (1-w) \cdot \boldsymbol{q}_2$ for $w \in \mathbb{R}_{[0,1]}$. However, this has no impact on the value of the cost function as all these linear combinations yield the same value λ_1.

E.7. Proof of Proposition 16.3.2

For $\beta = 1$, the cost function for ANOMAX simplifies into

$$\boldsymbol{G}_{\mathrm{opt}} = \underset{\boldsymbol{G} | \|\boldsymbol{G}\|_{\mathrm{F}}=1}{\arg\max} \left\| \boldsymbol{H}_1^{(\mathrm{b})} \cdot \boldsymbol{G} \cdot \boldsymbol{H}_2^{(\mathrm{f})} \right\|_{\mathrm{F}}^2 . \tag{E.42}$$

Let the SVDs of $\boldsymbol{H}_1^{(\mathrm{b})}$ and $\boldsymbol{H}_2^{(\mathrm{f})}$ be given by

$$\boldsymbol{H}_1^{(\mathrm{b})} = \boldsymbol{U}_1^{(\mathrm{b})} \cdot \boldsymbol{\Sigma}_1^{(\mathrm{b})} \cdot \boldsymbol{V}_1^{(\mathrm{b})^{\mathrm{H}}} \tag{E.43}$$

$$\boldsymbol{H}_2^{(\mathrm{f})} = \boldsymbol{U}_2^{(\mathrm{f})} \cdot \boldsymbol{\Sigma}_2^{(\mathrm{f})} \cdot \boldsymbol{V}_2^{(\mathrm{f})^{\mathrm{H}}}, \tag{E.44}$$

$$\tag{E.45}$$

where $\boldsymbol{U}_1^{(\mathrm{b})} \in \mathbb{C}^{M_1 \times \min\{M_1, M_{\mathrm{R}}\}}$, $\boldsymbol{\Sigma}_1^{(\mathrm{b})} \in \mathbb{C}^{\min\{M_1,M_{\mathrm{R}}\},\min\{M_1,M_{\mathrm{R}}\}}$, $\boldsymbol{V}_1^{(\mathrm{b})} \in \mathbb{C}^{M_{\mathrm{R}} \times \min\{M_1,M_{\mathrm{R}}\}}$, $\boldsymbol{U}_2^{(\mathrm{f})} \in \mathbb{C}^{M_{\mathrm{R}} \times \min\{M_2,M_{\mathrm{R}}\}}$, $\boldsymbol{\Sigma}_2^{(\mathrm{f})} \in \mathbb{C}^{\min\{M_2,M_{\mathrm{R}}\},\min\{M_2,M_{\mathrm{R}}\}}$, and $\boldsymbol{V}_2^{(\mathrm{f})} \in \mathbb{C}^{M_2 \times \min\{M_2,M_{\mathrm{R}}\}}$. Then, $\boldsymbol{G}$ can be written as

$$\boldsymbol{G} = \boldsymbol{V}_1^{(\mathrm{b})} \cdot \boldsymbol{\Gamma} \cdot \boldsymbol{U}_2^{(\mathrm{f})^{\mathrm{H}}} = \sum_{k=1}^{\min\{M_1, M_{\mathrm{R}}\}} \sum_{\ell=1}^{\min\{M_2, M_{\mathrm{R}}\}} \gamma_{k,\ell} \cdot \boldsymbol{v}_{1,k}^{(\mathrm{b})} \cdot \boldsymbol{u}_{2,\ell}^{(\mathrm{f})^{\mathrm{H}}} \tag{E.46}$$

without loss of generality, since any left singular vector orthogonal to $\boldsymbol{V}_1^{(\mathrm{b})}$ and any right singular vector orthogonal to $\boldsymbol{U}_2^{(\mathrm{f})^{\mathrm{H}}}$ does not contribute to the cost function. Here, $\gamma_{k,\ell} = [\boldsymbol{G}]_{(k,\ell)}$ and the vectors $\boldsymbol{v}_{1,k}^{(\mathrm{b})}$ and $\boldsymbol{u}_{2,\ell}^{(\mathrm{f})}$ represent the k-th and the ℓ-th column of $\boldsymbol{V}_1^{(\mathrm{b})}$ and $\boldsymbol{U}_2^{(\mathrm{f})}$, respectively. Note that $\boldsymbol{\Gamma}$ is of size $\min\{M_1, M_{\mathrm{R}}\} \times \min\{M_2, M_{\mathrm{R}}\}$ and the constraint $\|\boldsymbol{G}\|_{\mathrm{F}}^2 = 1$ translates to $\|\boldsymbol{\Gamma}\|_{\mathrm{F}}^2 = \sum_{k,\ell} |\gamma_{k,\ell}|^2 = 1$. Inserting this form of $\boldsymbol{G}$ into the cost function, we have

$$\begin{aligned} \left\| \boldsymbol{U}_1^{(\mathrm{b})} \cdot \boldsymbol{\Sigma}_1^{(\mathrm{b})} \cdot \boldsymbol{\Gamma} \cdot \boldsymbol{\Sigma}_2^{(\mathrm{f})} \cdot \boldsymbol{V}_2^{(\mathrm{f})^{\mathrm{H}}} \right\|_{\mathrm{F}}^2 &= \left\| \boldsymbol{\Sigma}_1^{(\mathrm{b})} \cdot \boldsymbol{\Gamma} \cdot \boldsymbol{\Sigma}_2^{(\mathrm{f})} \right\|_{\mathrm{F}}^2 \\ &= \sum_{k=1}^{\min\{M_1, M_{\mathrm{R}}\}} \sum_{\ell=1}^{\min\{M_2, M_{\mathrm{R}}\}} |\gamma_{k,\ell}|^2 \cdot \left(\sigma_{1,k}^{(\mathrm{b})} \cdot \sigma_{2,\ell}^{(\mathrm{f})} \right)^2, \end{aligned} \tag{E.47}$$

where $\sigma_{1,k}^{(\mathrm{b})}$ are the singular values of $\boldsymbol{H}_1^{(\mathrm{b})}$ and $\sigma_{2,\ell}^{(\mathrm{f})}$ are the singular values of $\boldsymbol{H}_2^{(\mathrm{f})}$. Since both singular value profiles are ordered and we have the constraint $\sum_{k,\ell} |\gamma_{k,\ell}|^2 = 1$ it is best to set $\gamma_{1,1} = 1$ and $\gamma_{k,\ell} = 0$ for $k > 1$ or $\ell > 1$. Consequently, $\boldsymbol{G}_{\mathrm{opt}}$ is given by

$$\boldsymbol{G}_{\mathrm{opt}} = \boldsymbol{v}_{1,1}^{(\mathrm{b})} \cdot \boldsymbol{u}_{2,1}^{(\mathrm{f})^{\mathrm{H}}}. \tag{E.48}$$

This concludes the proof of the proposition for $\beta = 1$. For $\beta = 0$, the proof proceeds analogously.

□

E.8. Proof of Proposition 16.4.1

The first obvious step is that the factor 1/2 as well as the $\log_2$ function can be dropped, since $1/2 \log_2(x)$ represents a monotonically increasing function for $x > 0$. The cost function can then be written as

$$J(\boldsymbol{g}) = \left(1 + \frac{P_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}}\right) \cdot \left(1 + \frac{P_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},2}}\right) \tag{E.49}$$

$$= \frac{\tilde{P}_{\mathrm{R},1}}{\tilde{P}_{\mathrm{N},1}} \cdot \frac{\tilde{P}_{\mathrm{R},2}}{\tilde{P}_{\mathrm{N},2}}, \tag{E.50}$$

where the desired signal power $P_{\mathrm{R},i}$ and the noise power $\tilde{P}_{\mathrm{N},i}$ at UT_i, $i = 1, 2$ are defined in Lemma 16.1.4. Now, the main trick to transform this problem into (16.32) is to incorporate the power constraint $P_{\mathrm{T,R}} = \boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g} \le P_{\mathrm{T,R}}^{\max}$ into the cost function. To this end, we need the following lemma:

Lemma E.8.1. *The vector $\boldsymbol{g}_{\mathrm{opt}}$ maximizing the cost function $J(\boldsymbol{g})$ satisfies the power constraint with equality, i.e., $\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_{\mathrm{opt}} = P_{\mathrm{T,R}}^{\max}$.*

Proof. Firstly, inserting (16.10), (16.11), and (16.12) into (E.49), we have

$$J(\boldsymbol{g}) = \left(1 + \frac{\boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{K}_{2,1} \cdot \boldsymbol{g} \cdot P_{\mathrm{T},2}}{\boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{g} + P_{\mathrm{N},1}}\right) \cdot \left(1 + \frac{\boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{K}_{1,2} \cdot \boldsymbol{g} \cdot P_{\mathrm{T},1}}{\boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{g} + P_{\mathrm{N},2}}\right). \tag{E.51}$$

Based on this form of the cost function we can prove the lemma by contradiction. Assume that an optimal $\boldsymbol{g}_{\mathrm{opt}}$ exists which satisfies $\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_{\mathrm{opt}} < P_{\mathrm{T,R}}^{\max}$. Then, we can find a constant $c > 1$, such that $\bar{\boldsymbol{g}}_{\mathrm{opt}} = c \cdot \boldsymbol{g}_{\mathrm{opt}}$ satisfies $\bar{\boldsymbol{g}}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \bar{\boldsymbol{g}}_{\mathrm{opt}} = P_{\mathrm{T,R}}^{\max}$. However, inserting $\bar{\boldsymbol{g}}_{\mathrm{opt}}$ into the cost function $\boldsymbol{J}(\boldsymbol{g})$ we have

$$\begin{aligned} J(\bar{\boldsymbol{g}}_{\mathrm{opt}}) &= \left(1 + \frac{c^2 \cdot \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{2,1} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},2}}{c^2 \cdot \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{g}_{\mathrm{opt}} + P_{\mathrm{N},1}}\right) \cdot \left(1 + \frac{c^2 \cdot \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{1,2} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},1}}{c^2 \cdot \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{g}_{\mathrm{opt}} + P_{\mathrm{N},2}}\right) \\ &= \left(1 + \frac{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{2,1} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},2}}{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{J}_1 \cdot \boldsymbol{g}_{\mathrm{opt}} + \frac{P_{\mathrm{N},1}}{c^2}}\right) \cdot \left(1 + \frac{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{1,2} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},1}}{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{J}_2 \cdot \boldsymbol{g}_{\mathrm{opt}} + \frac{P_{\mathrm{N},2}}{c^2}}\right) \end{aligned} \tag{E.52}$$

which is monotonically increasing in c since $P_{\mathrm{N},i} > 0$. Since $c > 1$, we therefore have $J(\bar{\boldsymbol{g}}_{\mathrm{opt}}) > J(\boldsymbol{g}_{\mathrm{opt}})$, contradicting the assumption that $\boldsymbol{g}_{\mathrm{opt}}$ was optimal. Hence, our assumption was false, which proves the lemma. □

Applying this lemma to the maximization of $J(\boldsymbol{g})$ subject to $P_{\mathrm{T,R}} \le P_{\mathrm{T,R}}^{\max}$ it becomes evident that the inequality constraint can be replaced by an equality constraint. Hence, we have

$$\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_{\mathrm{opt}} = P_{\mathrm{T,R}}^{\max}. \tag{E.53}$$

Using (E.53) and (16.12), we can express $\tilde{P}_{\mathrm{N},i}$ as

$$\tilde{P}_{\mathrm{N},i} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{J}_i \cdot \boldsymbol{g}_{\mathrm{opt}} + P_{\mathrm{N},i} \cdot \frac{\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_{\mathrm{opt}}}{P_{\mathrm{T,R}}^{\max}} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \underbrace{\left(\boldsymbol{J}_i + \frac{P_{\mathrm{N},i}}{P_{\mathrm{T,R}}^{\max}} \cdot \boldsymbol{Q}\right)}_{\tilde{\boldsymbol{J}}_i} \cdot \boldsymbol{g}_{\mathrm{opt}}. \tag{E.54}$$

Moreover, since $\tilde{P}_{\mathrm{R},i} \doteq P_{\mathrm{R},i} + \tilde{P}_{\mathrm{N},i}$ we obtain by combining (E.53) with (16.10) and (16.11)

$$\tilde{P}_{\mathrm{R},1} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{2,1} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},2} + \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \tilde{\boldsymbol{J}}_1 \cdot \boldsymbol{g}_{\mathrm{opt}} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \left(\boldsymbol{K}_{2,1} \cdot P_{\mathrm{T},2} + \tilde{\boldsymbol{J}}_1\right) \cdot \boldsymbol{g}_{\mathrm{opt}} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_1 \cdot \boldsymbol{g}_{\mathrm{opt}}$$
$$\tilde{P}_{\mathrm{R},2} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{K}_{1,2} \cdot \boldsymbol{g}_{\mathrm{opt}} \cdot P_{\mathrm{T},1} + \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \tilde{\boldsymbol{J}}_2 \cdot \boldsymbol{g}_{\mathrm{opt}} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \left(\boldsymbol{K}_{1,2} \cdot P_{\mathrm{T},1} + \tilde{\boldsymbol{J}}_2\right) \cdot \boldsymbol{g}_{\mathrm{opt}} = \boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_2 \cdot \boldsymbol{g}_{\mathrm{opt}}.$$

Inserting these intermediate results into the cost function (E.50), our equivalent optimization problem can be stated as

$$\boldsymbol{g}_{\mathrm{opt}} = \underset{\boldsymbol{g} | \boldsymbol{g}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g} = P_{\mathrm{T,R}}^{\max}}{\arg\max} \; J(\boldsymbol{g})$$
$$J(\boldsymbol{g}) = \frac{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_1 \cdot \boldsymbol{g}}{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{J}}_1 \cdot \boldsymbol{g}} \cdot \frac{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{K}}_2 \cdot \boldsymbol{g}}{\boldsymbol{g}^{\mathrm{H}} \cdot \tilde{\boldsymbol{J}}_2 \cdot \boldsymbol{g}}. \tag{E.55}$$

The final step to showing the proposition is to realize that $J(\boldsymbol{g})$ is in fact a homogeneous function in the norm of $\boldsymbol{g}$, i.e., $J(\boldsymbol{g}) = J(c \cdot \boldsymbol{g})\ \forall c \neq 0$. Therefore, we can drop the equality constraint as well and solve the unconstrained maximization problem provided that the resulting $\boldsymbol{g}_{\mathrm{opt}}$ is properly rescaled to satisfy $\boldsymbol{g}_{\mathrm{opt}}^{\mathrm{H}} \cdot \boldsymbol{Q} \cdot \boldsymbol{g}_{\mathrm{opt}} = P_{\mathrm{T,R}}^{\max}$. This concludes the proof of the proposition. □

E.9. Lowering the complexity for RAGES

For white noise at the relay and reciprocal channels, the rate-optimal relay amplification matrix has the form $\boldsymbol{G} = \boldsymbol{H}^* \cdot \boldsymbol{B} \cdot \boldsymbol{H}^{\mathrm{H}}$, as shown in [ZLCC09][2], where $\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h}_1 & \boldsymbol{h}_2 \end{bmatrix} \in \mathbb{C}^{M_{\mathrm{R}} \times 2}$.

Via (3.7) we find that the vector $\boldsymbol{g} = \mathrm{vec}\{\boldsymbol{G}\}$ can then be written as

$$\boldsymbol{g} = \mathrm{vec}\{\boldsymbol{G}\} = \mathrm{vec}\left\{\boldsymbol{H}^* \cdot \boldsymbol{B} \cdot \boldsymbol{H}^{\mathrm{H}}\right\} = (\boldsymbol{H}^* \otimes \boldsymbol{H}^*) \cdot \underbrace{\mathrm{vec}\{\boldsymbol{B}\}}_{\boldsymbol{b}}$$
$$= (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{b} \tag{E.56}$$

In order to proceed we require the following lemma:

Lemma E.9.1. *The following identities are true*

$$\tilde{\boldsymbol{K}}_1 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* = (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{C}_1$$
$$\tilde{\boldsymbol{K}}_2 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* = (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{C}_2$$

[2] Note that [ZLCC09] uses a different parametrization. The authors write $\boldsymbol{G} = \boldsymbol{U}^* \cdot \boldsymbol{B} \cdot \boldsymbol{U}^{\mathrm{H}}$, where $\boldsymbol{U} \in \mathbb{C}^{M \times 2}$ denote the two left singular vectors of the matrix $\boldsymbol{H}$. However, since $\boldsymbol{H} = \boldsymbol{U} \cdot \boldsymbol{T}$ for a non-singular matrix $\boldsymbol{T} \in \mathbb{C}^{2 \times 2}$, this is equivalent to the form we use here.

$$\begin{aligned}\tilde{\boldsymbol{J}}_1 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* &= (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{D}_1 \\ \tilde{\boldsymbol{J}}_1 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* &= (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{D}_2,\end{aligned} \tag{E.57}$$

where the matrices $\boldsymbol{C}_1, \boldsymbol{C}_2, \boldsymbol{D}_1, \boldsymbol{D}_2 \in \mathbb{C}^{2\times 2}$ *are given by*

$$\boldsymbol{C}_1 = \begin{bmatrix} P_{\mathrm{N,R}}\alpha_1^2 + \rho_1\beta_1 & P_{\mathrm{N,R}}\alpha_{2,1} & \rho_1 P_{\mathrm{T},1}\alpha_{2,1} & 0 \\ 0 & \rho_1\beta_1 & 0 & \rho_1 P_{\mathrm{T},1}\alpha_{2,1} \\ P_{\mathrm{T},2}\alpha_{1,2}(\rho_1 + \alpha_1^2) & P_{\mathrm{T},2}\alpha_1^2\alpha_2^2 & (\rho_1 + \alpha_1^2)\beta_2 & \alpha_{2,1}\beta_2 \\ 0 & \rho_1 P_{\mathrm{T},2}\alpha_{1,2} & 0 & \rho_1\beta_2 \end{bmatrix}$$

$$\boldsymbol{C}_2 = \begin{bmatrix} \rho_2\beta_1 & 0 & \rho_2 P_{\mathrm{T},1}\alpha_{2,1} & 0 \\ \alpha_{1,2}\beta_1 & (\rho_2 + \alpha_2^2)\beta_1 & P_{\mathrm{T},1}\alpha_1^2\alpha_2^2 & P_{\mathrm{T},1}\alpha_{2,1}(\rho_2 + \alpha_2^2) \\ \rho_2 P_{\mathrm{T},2}\alpha_{1,2} & 0 & \rho_2\beta_2 & 0 \\ 0 & \rho_2 P_{\mathrm{T},2}\alpha_{1,2} & P_{\mathrm{N,R}}\alpha_{1,2} & P_{\mathrm{N,R}}\alpha_2^2 + \rho_2\beta_2 \end{bmatrix}$$

$$\boldsymbol{D}_1 = \begin{bmatrix} P_{\mathrm{N,R}}\alpha_1^2 + \rho_1\beta_1 & P_{\mathrm{N,R}}\alpha_{2,1} & \rho_1 P_{\mathrm{T},1}\alpha_{2,1} & 0 \\ 0 & \rho_1\beta_1 & 0 & \rho_1 P_{\mathrm{T},1}\alpha_{2,1} \\ \rho_1 P_{\mathrm{T},2}\alpha_{1,2} & 0 & P_{\mathrm{N,R}}\alpha_1^2 + \rho_1\beta_2 & P_{\mathrm{N,R}}\alpha_{2,1} \\ 0 & \rho_1 P_{\mathrm{T},2}\alpha_{1,2} & 0 & \rho_1\beta_2 \end{bmatrix}$$

$$\boldsymbol{D}_2 = \begin{bmatrix} \rho_2\beta_1 & 0 & \rho_2 P_{\mathrm{T},1}\alpha_{2,1} & 0 \\ P_{\mathrm{N,R}}\alpha_{1,2} & P_{\mathrm{N,R}}\alpha_2^2 + \rho_2\beta_1 & 0 & \rho_2 P_{\mathrm{T},1}\alpha_{2,1} \\ \rho_2 P_{\mathrm{T},2}\alpha_{1,2} & 0 & \rho_2\beta_2 & 0 \\ 0 & \rho_2 P_{\mathrm{T},2}\alpha_{1,2} & P_{\mathrm{N,R}}\alpha_{1,2} & P_{\mathrm{N,R}}\alpha_2^2 + \rho_2\beta_2 \end{bmatrix}$$

using the short-hand notations $\alpha_k^2 = \|\boldsymbol{h}_k\|_2^2$, $\alpha_{k,\ell} = \boldsymbol{h}_k^{\mathrm{H}} \cdot \boldsymbol{h}_\ell$, $\beta_k = P_{\mathrm{T},k}\alpha_k^2 + P_{\mathrm{N,R}}$, *and* $\rho_k = \frac{P_{\mathrm{N},k}}{P_{\mathrm{T,R}}^{\max}}$ *for* $k, \ell \in \{1, 2\}$

Proof. This lemma is easily shown by inserting the definitions of $\tilde{\boldsymbol{K}}_n$ and $\tilde{\boldsymbol{J}}_n$ shown in Proposition 16.4.1. For instance, $\tilde{\boldsymbol{K}}_1$ is defined as a scalar combination of the matrices $\boldsymbol{K}_{2,1}$, $\boldsymbol{J}_1$, and $\boldsymbol{Q}$. However, as evident from their definition in Lemma 16.1.4, for the special case treated here, these three matrices are linear combinations of $(\boldsymbol{h}_2 \otimes \boldsymbol{h}_1)^* \cdot (\boldsymbol{h}_2 \otimes \boldsymbol{h}_1)^{\mathrm{T}}$, $(\boldsymbol{I}_{M_{\mathrm{R}}} \otimes \boldsymbol{h}_1)^* \cdot (\boldsymbol{I}_{M_{\mathrm{R}}} \otimes \boldsymbol{h}_1)^{\mathrm{T}}$, $(\boldsymbol{h}_1 \otimes \boldsymbol{I}_{M_{\mathrm{R}}})^* \cdot (\boldsymbol{h}_1 \otimes \boldsymbol{I}_{M_{\mathrm{R}}})^{\mathrm{T}}$, $(\boldsymbol{h}_2 \otimes \boldsymbol{I}_{M_{\mathrm{R}}})^* \cdot (\boldsymbol{h}_2 \otimes \boldsymbol{I}_{M_{\mathrm{R}}})^{\mathrm{T}}$, and $\boldsymbol{I}_{M_{\mathrm{R}}^2}$. Therefore, the product with $\boldsymbol{H}^* \otimes \boldsymbol{H}^*$ yields a matrix for which each column is a linear combination of $\boldsymbol{h}_k^* \otimes \boldsymbol{h}_\ell^*$ for $k, \ell = 1, 2$. Determining the coefficients of these linear combinations is straightforward. □

Inserting (E.56) into the generalized eigenproblem for RAGES from (16.34) and applying

Lemma E.9.1 we find

$$\begin{aligned}
\left(\tilde{\boldsymbol{K}}_1 + \rho_{\text{sig}} \cdot \tilde{\boldsymbol{K}}_2\right) \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{b} &= \frac{\tilde{P}_{\text{R},1}}{\tilde{P}_{\text{N},1}} \cdot \left(\tilde{\boldsymbol{J}}_1 + \rho_{\text{noi}} \cdot \tilde{\boldsymbol{J}}_2\right) \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{b} \\
\left(\tilde{\boldsymbol{K}}_1 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* + \rho_{\text{sig}} \cdot \tilde{\boldsymbol{K}}_2 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^*\right) \cdot \boldsymbol{b} &= \frac{\tilde{P}_{\text{R},1}}{\tilde{P}_{\text{N},1}} \cdot \left(\tilde{\boldsymbol{J}}_1 \cdot + \rho_{\text{noi}} \cdot \tilde{\boldsymbol{J}}_2 \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^*\right) \cdot \boldsymbol{b} \\
(\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot (\boldsymbol{C}_1 + \rho_{\text{sig}} \cdot \boldsymbol{C}_2) \cdot \boldsymbol{b} &= \frac{\tilde{P}_{\text{R},1}}{\tilde{P}_{\text{N},1}} \cdot (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot (\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2) \cdot \boldsymbol{b}. \qquad \text{(E.58)}
\end{aligned}$$

Therefore, the necessary condition for $\boldsymbol{g} = (\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot \boldsymbol{b}$ to be optimal is equivalent to the condition (E.58) for $\boldsymbol{b}$. Obviously, for (E.58) to be satisfied, $\boldsymbol{b}$ must be a generalized eigenvector of $(\boldsymbol{C}_1 + \rho_{\text{sig}} \cdot \boldsymbol{C}_2) \in \mathbb{C}^{2\times 2}$ and $(\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2) \in \mathbb{C}^{2\times 2}$. To see this, we simply multiply (E.58) from the left-hand side with the pseudo-inverse of $(\boldsymbol{H} \otimes \boldsymbol{H})^* \cdot (\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2)$. If $(\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2)$ is invertible[3] this yields

$$(\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2)^{-1} \cdot (\boldsymbol{C}_1 + \rho_{\text{sig}} \cdot \boldsymbol{C}_2) \cdot \boldsymbol{b} = \frac{\tilde{P}_{\text{R},1}}{\tilde{P}_{\text{N},1}} \cdot \boldsymbol{b}. \qquad \text{(E.59)}$$

Therefore, instead of computing the generalized eigenvectors of the $M_\text{R}^2 \times M_\text{R}^2$ matrices $\tilde{\boldsymbol{K}}_1 + \rho_{\text{sig}} \cdot \tilde{\boldsymbol{K}}_2$ and $\tilde{\boldsymbol{J}}_1 + \rho_{\text{noi}} \cdot \tilde{\boldsymbol{J}}_2$ shown in (16.34), we can resort to computing the generalized eigenvectors of the 2×2 matrices $\boldsymbol{C}_1 + \rho_{\text{sig}} \cdot \boldsymbol{C}_2$ and $\boldsymbol{D}_1 + \rho_{\text{noi}} \cdot \boldsymbol{D}_2$ shown in (E.59), which significantly lowers the computational complexity. The computation of $\boldsymbol{C}_1$, $\boldsymbol{C}_2$, $\boldsymbol{D}_1$, $\boldsymbol{D}_2$ requires only linear complexity in M_R, all subsequent computations are independent of M_R.

[3]It can be verified that this matrix has full rank for finite SNRs. This is also reasonable since otherwise we would get infinite generalized eigenvalues which would correspond to an infinite sum-rate.

Bibliography

Publications and Technical Documents as Co- or First Author

[CLZ+12] Y. Cheng, S. Li, J. Zhang, F. Roemer, M. Haardt, Y. Zhou, and M. Dong, "Linear precoding-based geometric mean decomposition (LP-GMD) for Multi-User MIMO systems," in *Proc. Int. Symp. Wireless Comm. Systems (ISWCS 2012)*, Paris, France, Aug. 2012.

[CSR+12] Y. Cheng, N. Song, F. Roemer, M. Haardt, H. Henniger, R. Metzig, and E. Diedrich, "Satellite ground stations with electronic beam steering," in *Proc. 1st Intl. Conf. in Europe about Space and Satellite Telecomm. (ESTEL 2012)*, Rome, Italy, Oct. 2012.

[DHR08] J. P. C. L. Da Costa, M. Haardt, and F. Roemer, "Robust methods based on the HOSVD for estimating the model order in PARAFAC models," in *Proc. 5-th IEEE Sensor Array and Multich. Sig. Proc. Workshop (SAM 2008)*, Darmstadt, Germany, Jul. 2008, pp. 510 – 514.

[DHR09] ——, "Sequential GSVD based prewhitening for multidimensional HOSVD based subspace estimation," in *Proc. ITG Workshop on Smart Antennas (WSA'09)*, Berlin, Germany, Feb. 2009.

[DHRD07] J. P. C. L. Da Costa, M. Haardt, F. Roemer, and G. Del Galdo, "Enhanced model order estimation using higher-order arrays," in *Proc. 41-st Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2007, pp. 412 – 416.

[DRH09] J. P. C. L. Da Costa, F. Roemer, and M. Haardt, "Deterministic prewhitening to improve subspace based parameter estimation techniques in severely colored noise environments," in *Proc. 54th Int. Scientif. Coll. (IWK) Ilmenau*, Ilmenau, Germany, Sep. 2009.

[DRH10] ——, "Iterative sequential GSVD (I-S-GSVD) based prewhitening for multidimensional HOSVD based subspace estimation without knowledge of the noise covariance information," in *Proc. ITG Workshop on Smart Antennas (WSA'10)*, Bremen, Germany, Feb. 2010.

[DRHdS11] J. P. C. L. Da Costa, F. Roemer, M. Haardt, and R. T. de Sousa Jr., "Multi-Dimensional model order selection," *EURASIP Journal on Advances in Signal Processing*, vol. 26, Jul. 2011, review article.

[DRWH10] J. P. C. L. Da Costa, F. Roemer, M. Weis, and M. Haardt, “Robust R-D parameter estimation via closed-form PARAFAC,” in *Proc. ITG Workshop on Smart Antennas (WSA'10)*, Bremen, Germany, Feb. 2010.

[DSR+10] J. P. C. L. Da Costa, D. Schulz, F. Roemer, M. Haardt, and J. A. Apolinario Jr., “Robust R-D parameter estimation via Closed-Form PARAFAC in Kronecker colored environment,” in *Proc. Int. Symp. Wireless Comm. Systems (ISWCS 2010)*, York, UK, Sep. 2010.

[DTRH09] J. P. C. L. Da Costa, A. Thakre, F. Roemer, and M. Haardt, “Comparison of model order selection techniques for high-resolution parameter estimation algorithms,” in *Proc. 54th Int. Scientif. Coll. (IWK) Ilmenau*, Ilmenau, Germany, Sep. 2009.

[GZRH12] S. Gherekhloo, B. Zafar, F. Roemer, and M. Haardt, “Impact of synchronization errors on Alamouti-STBC-based cooperative MIMO schemes,” in *Proc. 7-th IEEE Sensor Array and Multich. Sig. Proc. Workshop (SAM 2012)*, Hoboken, NJ, Jul. 2012.

[HR04] M. Haardt and F. Roemer, “Enhancements of unitary ESPRIT for non-circular sources,” in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2004)*, vol. II, Montreal, Canada, May 2004, pp. 101 – 104.

[HRD08] M. Haardt, F. Roemer, and G. Del Galdo, “Higher-order SVD based subspace estimation to improve the parameter estimation accuracy in multi-dimensional harmonic retrieval problems,” *IEEE Trans. Sig. Proc.*, vol. 56, pp. 3198 – 3213, Jul. 2008.

[JBF+10] E. Jorswieck, L. Badia, T. Fahldieck, D. Gesbert, S. Gustafsson, M. Haardt, K. Ho, E. Karipidis, A. Kortke, E. Larsson, H. Mark, M. Nawrocki, V. Palestini, R. Piesiewicz, P. Priotti, F. Roemer, M. Schubert, J. Sykora, P. Trommelen, B. van den Ende, and M. Zorzi, “Resource sharing in wireless networks: The SAPHYRE approach,” in *Proc. Future Network and Mobile Summit 2010*, Florence, Italy, Jun. 2010.

[JRW+08] D. Jannek, F. Roemer, M. Weis, M. Haardt, and P. Husar, “Identification of signal components in multi-channel EEG signals via closed-form PARAFAC analysis and appropriate preprocessing,” in *Proc. 4th Europ. Med. and Biol. Engineering Conf. (EMBEC 2008)*, Antwerp, Belgium, Nov. 2008.

[KRVH12] A. Khabbazibasmenj, F. Roemer, S. A. Vorobyov, and M. Haardt, “Sum-rate maximization in two-way AF MIMO relaying: Polynomial time solutions to a class of DC programming problems,” *IEEE Trans. Sig. Proc.*, no. 10, pp. 5478–5493, Oct. 2012.

[KTS+10] P. Komulainen, A. Toelli, B. Song, F. Roemer, E. Bjoernson, and M. Bengtsson, "CSI acquisition concepts for advanced antenna schemes in the WINNER+ project," in *Proc. Future Network and Mobile Summit 2010*, Florence, Italy, Jun. 2010.

[KVRH12] A. Khabbazibasmenj, S. A. Vorobyov, F. Roemer, and M. Haardt, "Polynomial-time DC (POTDC) for sum-rate maximization in two-way AF MIMO relaying," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2012)*, Kyoto, Japan, Mar. 2012.

[LCZ+12] S. Li, Y. Cheng, J. Zhang, F. Roemer, B. Song, M. Haardt, Y. Zhou, and M. Dong, "Efficient spatial scheduling and precoding algorithms for MC MU MIMO system," in *Proc. Int. Symp. Wireless Comm. Systems (ISWCS 2012)*, Paris, France, Aug. 2012.

[LRH11a] J. Li, F. Roemer, and M. Haardt, "Efficient relay sharing (EReSh) between multiple operators in amplify-and-forward relaying systems," in *Proc. of the IEEE Int. Workshop on Comp. Adv. in Multi-Sensor Adaptive Proc. (CAMSAP 2011)*, San Juan, Puerto Rico, Dec. 2011, pp. 249 – 252.

[LRH11b] ——, "Spectrum and infrastructure sharing in the MIMO interference relay channels," in *Proc. 19-th European Sig. Proc. Conf. (EUSIPCO-2011)*, Barcelona, Spain, Aug. 2011.

[LZR+11] J. Li, J. Zhang, F. Roemer, M. Haardt, C. Scheunert, E. Jorswieck, M. Hekrdla, and J. Sykora, "Relay-assisted spectrum and infrastructure sharing between multiple operators," in *Proc. Future Network and Mobile Summit 2011*, Warsaw, Poland, Jun. 2011.

[LZRH11] Z. Lu, A. M. Zoubir, F. Roemer, and M. Haardt, "Source enumeration using the bootstrap for very few samples," in *Proc. 19-th European Sig. Proc. Conf. (EUSIPCO-2011)*, Barcelona, Spain, Aug. 2011.

[MCR+12] M. Muma, Y. Cheng, F. Roemer, M. Haardt, and A. M. Zoubir, "Robust source number enumeration for r-dimensional arrays in case of brief sensor failures," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2012)*, Kyoto, Japan, Mar. 2012.

[OORW08] M. Olsson, A. Osseiran, F. Roemer, and T. Wild, "Multi-antenna processing in the WINNER air interface," in *Proc. XXIX Gen. Ass. of the Int. Union of Radio Science (URSI 2008)*, Chicago, IL, Aug. 2008.

[RBH10] F. Roemer, H. Becker, and M. Haardt, "Analytical performance analysis for multi-dimensional Tensor-ESPRIT-type parameter estimation algorithms," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2010)*, Dallas, TX, Mar. 2010.

[RBHW09] F. Roemer, H. Becker, M. Haardt, and M. Weis, "Analytical performance evaluation for HOSVD-based parameter estimation schemes," in *Proc. of the IEEE Int. Workshop on Comp. Adv. in Multi-Sensor Adaptive Proc. (CAMSAP 2009)*, Aruba, Dutch Antilles, Dec. 2009.

[RFH08] F. Roemer, M. Fuchs, and M. Haardt, "Distributed MIMO systems with spatial reuse for high-speed-indoor mobile radio access," in *Proc. 20-th Wireless World Research Forum (WWRF)*, Ottawa, ON, Canada, Apr. 2008.

[RH05] F. Roemer and M. Haardt, "Using 3-D unitary ESPRIT on a hexagonal shaped ESPAR antenna for 1-D and 2-D direction of arrival estimation," in *Proc. ITG/IEEE Workshop on Smart Antennas (WSA'05)*, Duisburg, Germany, Apr. 2005.

[RH06] ——, "Efficient 1-D and 2-D DOA estimation for non-circular sources with hexagonal shaped ESPAR arrays," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2006)*, Toulouse, France, May 2006, pp. 881 – 884.

[RH07a] ——, "Deterministic Cramér-Rao bounds for strict sense non-circular sources," in *Proc. ITG/IEEE Workshop on Smart Antennas (WSA'07)*, Vienna, Austria, Feb. 2007.

[RH07b] ——, "Tensor-structure structured least squares (TS-SLS) to improve the performance of multi-dimensional ESPRIT-type algorithms," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2007)*, vol. II, Honolulu, HI, Apr. 2007, pp. 893 – 896.

[RH08a] ——, "A closed-form solution for multilinear PARAFAC decompositions," in *Proc. 5-th IEEE Sensor Array and Multich. Sig. Proc. Workshop (SAM 2008)*, Darmstadt, Germany, Jul. 2008, pp. 487 – 491.

[RH08b] ——, "A closed-form solution for parallel factor (PARAFAC) analysis," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2008)*, Las Vegas, NV, Apr. 2008, pp. 2365 – 2368.

[RH09a] ——, "Algebraic Norm-Maximizing (ANOMAX) transmit strategy for Two-Way relaying with MIMO amplify and forward relays," *IEEE Sig. Proc. Lett.*, vol. 16, Oct. 2009.

[RH09b] ——, "Multidimensional unitary Tensor-ESPRIT for Non-Circular sources," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2009)*, Taipei, Taiwan, Apr. 2009.

[RH09c] ——, "Near-far robustness and optimal power allocation for two-way relaying with MIMO amplify and forward relays," in *Proc. of the IEEE Int. Workshop on Comp. Adv. in Multi-Sensor Adaptive Proc. (CAMSAP 2009)*, Aruba, Dutch Antilles, Dec. 2009.

[RH09d] ——, "Structured least squares (SLS) based enhancements of Tensor-Based channel estimation (TENCE) for Two-Way relaying with multiple antennas," in *Proc. ITG Workshop on Smart Antennas (WSA'09)*, Berlin, Germany, Feb. 2009.

[RH09e] ——, "Tensor-Based channel estimation (TENCE) for Two-Way relaying with multiple antennas and spatial reuse," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2009)*, Taipei, Taiwan, Apr. 2009.

[RH10a] ——, "A low-complexity relay transmit strategy for two-way relaying with MIMO amplify and forward relays," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2010)*, Dallas, TX, Mar. 2010.

[RH10b] ——, "Sum-rate maximization in two-way relaying systems with MIMO amplify and forward relays via generalized eigenvectors," in *Proc. 18-th European Sig. Proc. Conf. (EUSIPCO-2010)*, Aalborg, Denmark, Aug. 2010.

[RH10c] ——, "Tensor-Based channel estimation (TENCE) and iterative refinements for Two-Way relaying with multiple antennas and spatial reuse," *IEEE Trans. Sig. Proc.*, vol. 58, pp. 5720 – 5735, Nov. 2010.

[RH11] ——, "Analytical performance assessment of 1-D structured least squares," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2011)*, Prague, Czech Republic, May 2011.

[RH12] ——, "Efficient joint azimuth and elevation estimation using hexagonal antenna arrays," in *The Digital Signal Processing Handbook*, V. K. Madisetti and D. B. Williams, Eds. CRC Press and IEEE Press, Boca Raton, FL, 2012, 2nd edition, 2nd reprint, submitted.

[RHD06] F. Roemer, M. Haardt, and G. Del Galdo, "Higher order SVD based subspace estimation to improve multi-dimensional parameter estimation algorithms," in *Proc. 40-th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2006, pp. 961 – 965.

[RJH10] F. Roemer, E. Jorswieck, and M. Haardt, "Efficient spatial processing and resource allocation for amplify and forward two-way relaying," in *Cross Layer Designs in WLAN Systems*, N. Zorba, C. Skianis, and C. Verikoukis, Eds. Troubador Publishing Ltd, Leicester, UK, Sep. 2010.

[RSH12] F. Roemer, C. Schroeter, and M. Haardt, "A semi-algebraic framework for approximate CP decompositions via joint matrix diagonalization and generalized unfoldings," in *Proc. 46-th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2012.

[RSS+11] F. Roemer, N. Sarmadi, B. Song, M. Haardt, M. Pesavento, and A. B. Gershman, "Tensor-Based Semi-Blind channel estimation for MIMO OSTBC-Coded systems," in *Proc. 45-th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2011.

[RZHJ10] F. Roemer, J. Zhang, M. Haardt, and E. Jorswieck, "Spectrum and infrastructure sharing in wireless networks: A case study with Relay-Assisted communications," in *Proc. Future Network and Mobile Summit 2010*, Florence, Italy, Jun. 2010.

[SHR+08] V. Stankovic, M. Haardt, F. Roemer, S. Gale, and A. Jeffries (section editors), "Multi-user MIMO systems," in *Technologies for the Wireless Future: Wireless World Research Forum (WWRF), Volume 3*, K. David, Ed. John Wiley & Sons, Ltd, Chichester, England, Sep. 2008, pp. 234–242.

[SRH08] B. Song, F. Roemer, and M. Haardt, "Efficient channel quantization scheme for multi-user MIMO broadcast channels with RBD precoding," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2008)*, Las Vegas, NV, Apr. 2008, pp. 2389 – 2392.

[SRH10a] ——, "Blind estimation of SIMO channels using a tensor-based subspace method," in *Proc. 44-th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2010.

[SRH10b] ——, "Flexible coordinated beamforming (FlexCoBF) algorithm for the downlink of Multi-User MIMO systems," in *Proc. ITG Workshop on Smart Antennas (WSA'10)*, Bremen, Germany, Feb. 2010.

[SRH10c] ——, "Using a new structured joint congruence (STJOCO) transformation of Hermitian matrices for precoding in multi-user MIMO systems," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2010)*, Dallas, TX, Mar. 2010.

[THRG10] A. Thakre, M. Haardt, F. Roemer, and K. Giridhar, "Tensor-Based spatial smoothing (TB-SS) using multiple snapshots," *IEEE Trans. Sig. Proc.*, vol. 25, no. 5, pp. 2715–2728, May 2010.

[WJG+10a] M. Weis, D. Jannek, T. Guenther, P. Husar, M. Haardt, and F. Roemer, "Space-Time-Frequency component analysis of visual evoked potentials based on the PARAFAC2 model," in *Proc. 55th Int. Scientif. Coll. (IWK) Ilmenau*, Ilmenau, Germany, Sep. 2010.

[WJG+10b] M. Weis, D. Jannek, T. Guenther, P. Husar, F. Roemer, and M. Haardt, "Temporally resolved multi-way component analysis of dynamic sources in event-related EEG data using PARAFAC2," in *Proc. 18-th European Sig. Proc. Conf. (EUSIPCO-2010)*, Aalborg, Denmark, Aug. 2010.

[WJR+10] M. Weis, D. Jannek, F. Roemer, T. Guenther, M. Haardt, and P. Husar, "Multi-Dimensional PARAFAC2 component analysis of Multi-Channel EEG data including temporal tracking," in *Proc. 32nd Int. Conf. IEEE Eng. in Med. and Biol. Soc.*, Buenos Aires, Argentina, Sep. 2010.

[WRH+09] M. Weis, F. Roemer, M. Haardt, D. Jannek, and P. Husar, "Multi-dimensional Space-Time-Frequency component analysis of event-related EEG data using closed-form PARAFAC," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2009)*, Taipei, Taiwan, Apr. 2009.

[WRHH12] M. Weis, F. Roemer, M. Haardt, and P. Husar, "Dual-Symmetric parallel factor analysis using Procrustes estimation and Khatri-Rao factorization," in *Proc. 20-th European Sig. Proc. Conf. (EUSIPCO-2012)*, Bucharest, Romania, Aug. 2012.

[YSRH12] A. Yeredor, B. Song, F. Roemer, and M. Haardt, "A "Sequentially drilled" joint congruence (SeDJoCo) transformation with applications in blind source separation and Multi-User MIMO systems," *IEEE Trans. Sig. Proc.*, vol. 60, no. 6, pp. 2744–2757, Mar. 2012.

[ZBR+12] J. Zhang, N. Bornhorst, F. Roemer, M. Haardt, and M. Pesavento, "Optimal and suboptimal beamforming for Multi-Operator Two-Way relaying with a MIMO Amplify-and-Forward relay," in *Proc. 16th ITG Workshop on Smart Antennas (WSA'12)*, Dresden, Germany, Mar. 2012.

[ZRH11] J. Zhang, F. Roemer, and M. Haardt, "Beamforming design for multi-user two-way relaying with MIMO amplify and forward relays," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2011)*, Prague, Czech Republic, May 2011.

[ZRH12a] ——, "Relay assisted physical resource sharing: Projection based separation of multiple operators (ProBaSeMO) for Two-Way relaying with MIMO amplify and forward relays," *IEEE Trans. Sig. Proc.*, vol. 60, no. 9, pp. 4834–4848, Sep. 2012.

[ZRH+12b] J. Zhang, F. Roemer, M. Haardt, A. Khabbazibasmenj, and S. A. Vorobyov, "Sum rate maximization for multi-pair two-way relaying with single-antenna amplify and forward relays," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2012)*, Kyoto, Japan, Mar. 2012.

References by Other Authors

[3GP06] 3GPP, TR 25.814, "Physical layer aspects for evolved universal terrestrial radio access (UTRA)," www.3gpp.org, Oct. 2006.

[3GP10] 3GPP TR 36.814 V9.0.0, "Further advancements for E-UTRA physical layer aspects," www.3gpp.org, Mar. 2010.

[3GP11] 3GPP, TR 25.956 v10.0, "Repeater planning guidelines and system analysis," www.3gpp.org, Apr. 2011.

[AB00] C. A. Andersson and R. Bro, "The N-way toolbox for MATLAB," *Chemometrics and Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1–4, 2000.

[AB03] C. M. Andersen and R. Bro, "Practical aspects of PARAFAC modeling of fluorescence excitation-emission data," *Journal of Chemometrics*, vol. 17, pp. 200–215, 2003.

[ACH07] J. Andrews, W. Choi, and R. Heath, Jr., "Overcoming interference in spatial multiplexing MIMO cellular networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 95–104, Dec. 2007.

[ACKY05] E. Acar, S. A. Camtepe, M. S. Krishnamoorthy, and B. Yener, "Modeling and multiway analysis of chatroom tensors," *Lecture Notes in Computer Science*, vol. 3495, pp. 256–268, 2005.

[AD06] H. Abeida and J.-P. Delmas, "MUSIC-like estimation of direction of arrival for noncircular sources," *IEEE Transactions on Signal Processing*, vol. 54, no. 7, pp. 2678–2690, Jul. 2006.

[ADK11] E. Acar, D. M. Dunlavy, and T. G. Kolda, "A scalable optimization approach for fitting canonical tensor decompositions," *Journal of Chemometrics*, vol. 25, no. 2, pp. 67–86, Feb. 2011.

[AFCC04] L. Albera, A. Ferréol, P. Comon, and P. Chevalier, "Blind identification of overcomplete mixtures of sources (BIOME)," *Elsevier J. Linear Algebra Appl.*, vol. 391, pp. 3–30, Nov. 2004.

[AGU10] O. Amin, B. Gedik, and M. Uysal, "Channel estimation for amplify-and forward relaying: cascaded against disintegrated estimators," *IET Communications*, vol. 4, no. 10, Jan. 2010.

[AH99] C. A. Andersson and R. Henrion, "A general algorithm for obtaining simple structure of core arrays in N-way PCA with application to fluorimetric data," *Computational Statistics and Data Analysis*, vol. 31, pp. 255–278, 1999.

[AHB07] E. E. Abdallah, A. B. Hamza, and P. Bhattacharya, "MPEG video watermarking using tensor singular value decomposition," *Lecture Notes in Computer Science*, vol. 4633, pp. 772–783, 2007.

[AKD09] E. Acar, T. G. Kolda, and D. M. Dunlavy, "An optimization approach for fitting canonical tensor decompositions," Sandia National Laboratories, Albuquerque, NM and Livermore, CA, Tech. Rep. SAND2009-0857, Feb. 2009.

[AP10] S. Abdallah and I. N. Psaromiligkos, "Semi-blind channel estimation for amplify-and-forward two-way relay networks employing constant-modulus constellations," in *Proc. 44th Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2010.

[AR88] K. S. Arun and B. D. Rao, "An improved Toeplitz approximation method," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. IV, New York, NY, Apr. 1988, pp. 2352–2355.

[Bar83] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Boston, MA, May 1983, pp. 336–339.

[BCA+10] H. Becker, P. Comon, L. Albera, M. Haardt, and I. Merlet, "Multiway space-time-wave-vector analysis for source localization and extraction," in *Proc. 18-th European Signal Processing Conference (EUSIPCO 2010)*, Aalborg, Denmark, Aug. 2010, pp. 1349–1353.

[BCA+12] ——, "Multi-way space-time-wave-vector analysis for EEG source separation," *EURASIP Signal Processing*, vol. 92, no. 4, pp. 1021–1031, Apr. 2012.

[Bel60] R. Bellman, *Introduction to Matrix Analysis.* New York: McGraw Hill, 1960.

[BFY04] J. Boyer, D. D. Falconer, and H. Yanikomeroglu, "Multihop diversity in wireless relaying channels," *IEEE Transactions on Communications*, vol. 52, pp. 1820–1830, Oct. 2004.

[BG06] M. Biguesh and A. B. Gershman, "Training based MIMO channel estimation: a study of estimator tradeoffs and optimal training signals," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 884–893, Mar. 2006.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proceedings of the International Conference on Communications*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[BHS05] R. Bro, R. A. Harshman, and N. D. Sidiropoulos, "Modeling multi-way data with linearly dependent loadings," Tech. Report 2005-176, KVL, Tech. Rep., 2005.

[BHSL09] R. Bro, R. A. Harshman, N. D. Sidiropoulos, and M. E. Lundy, "Modeling multi-way data with linearly dependent loadings," *J. Chemometrics*, vol. 23, no. 7-8, pp. 324–340, Jul. 2009.

[BK03] R. Bro and H. A. L. Kiers, "A new efficient method for determining the number of components in PARAFAC models," *Journal of Chemometrics*, vol. 17, pp. 274–286, 2003.

[BK07] B. W. Bader and T. G. Kolda, "Efficient MATLAB computations with sparse and factored tensors," *SIAM Journal on Scientific Computing*, vol. 30, no. 1, pp. 205–231, Dec. 2007.

[BK+12] B. W. Bader, T. G. Kolda *et al.*, "Matlab tensor toolbox version 2.5," Jan. 2012. [Online]. Available: http://www.sandia.gov/~tgkolda/TensorToolbox/

[BL86] A. Bax and L. Lerner, "Two-dimensional NMR spectroscopy," *J. Amer. Assoc. for the Advancement of Science*, vol. 232, pp. 960–967, May 1986.

[BL03] S. Buzzi and M. Lops, "Performance analysis for the improved linear multiuser detectors in bpsk-modulated ds-cdma systems," *IEEE Transactions on Communications*, vol. 51, no. 1, pp. 37–42, Jan. 2003.

[Boy06] R. Boyer, "Three-way arrays for harmonic retrieval: the colored noise case," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, May 2006.

[Boy08] ——, "Decoupled root-MUSIC algorithm for multidimensional harmonic retrieval," in *Proc. of the 9th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Recife, Brazil, Jul. 2008.

[BPG+09] G. Boudreau, J. Panicker, N. Guo, R. Chang, N. Wang, and S. Vrzic, "Interference coordination and cancellation for 4G networks," *IEEE Communications Magazine*, vol. 47, no. 4, pp. 74–81, Apr. 2009.

[Bre78] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on Circuits and Systems*, vol. CAS-25, no. 9, pp. 772–781, Sep. 1978.

[Bri75] D. R. Brillinger, *Time Series: Data Analysis and Theory.* New York: Holt, Rhinehart and Winston, 1975.

[BRK07] F. Belloni, A. Richter, and V. Koivunen, "DOA estimation via manifold separation for arbitrary array structures," *IEEE Transactions on Signal Processing*, vol. 55, pp. 4800–4810, Oct. 2007.

[BSG99] R. Bro, N. D. Sidiropoulos, and G. B. Giannakis, "A fast least squares algorithm for separating trilinear mixtures," in *Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA99)*, Aussois, France, Jan. 1999.

[BV04] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[Car91] J.-F. Cardoso, "Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1991)*, Toronto, ON, Canada, May 1991.

[Cat44] R. B. Cattell, "Parallel proportional profiles and other principles for determining the choice of factors by rotation," *Psychometrika*, vol. 9, no. 4, Dec. 1944.

[CB07] P. Chevalier and A. Blin, "Widely linear MVDR beamformers for the reception of an unknown signal corrupted by noncircular interferences," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, Nov. 2007.

[CC70] J. D. Carroll and J. J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.

[CE79] T. M. Cover and A. A. El Gamal, "Capacity theorems for the relay channel," *IEEE Transactions on Information Theory*, vol. IT-25, pp. 572–584, Sep. 1979.

[CGHN09] T. Cui, F. Gao, T. Ho, and A. Nallanathan, "Distributed space-time coding for two-way wireless relay networks," *IEEE Trans. on Signal Processing*, vol. 57, no. 2, pp. 658–671, Feb. 2009.

[CGR+12] P. Cheng, L. Gui, Y. Rui, Y. J. Guo, X. Huang, and W. Zhang, "Compressed sensing based channel estimation for two-way relay networks," *IEEE Wireless Communication Letters*, vol. 1, no. 3, pp. 201–203, Jun. 2012.

[Chu72] D. Chu, "Polyphase codes with good periodic correlation properties," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 531–532, Jul. 1972.

[CLdA09] P. Comon, X. Luciani, and A. L. F. de Almeida, "Tensor decompositions, Alternating Least Squares and other tales," *Journal of Chemometrics*, vol. 23, pp. 393–405, Aug. 2009.

[CLdA11] ——, "The tensor package v1.0," Feb. 2011. [Online]. Available: http://www.i3s.unice.fr/~pcomon/TensorPackage.html

[CLSO12] H. Chung, N. Lee, B. Shim, and T. Oh, "On the beamforming design for MIMO multi-pair two-way relay channels," *IEEE Transactions on Vehicular Technology*, Jul. 2012, accepted for publication.

[Cos83] M. Costa, "Writing on dirty paper," *IEEE Transactions on Information Theory*, vol. 29, no. 12, pp. 439–441, May 1983.

[CP06] P. Chevalier and F. Pipon, "New insights into optimal widely linear array receivers for the demodulation of BPSK, MSK and GMSK signals corrupted by non circular interferences - application to SAIC," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 870–883, Mar. 2006.

[CS93] J. F. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *IEE Proceedings-F*, vol. 140, no. 6, pp. 362–270, Dec. 1993.

[CS03] G. Caire and S. Shamai, "On the achievable throughput of a multiantenna Gaussian broadcast channel," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1691–1706, Jul. 2003.

[CSK09] X. Chen, D. Schonfeld, and A. Khokhar, "View-invariant tensor null space representation for multiple motion trajectory retrieval and classification," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'09)*, Taipei, Taiwan, Apr. 2009.

[CST10] P. Comon, M. Sørensen, and E. P. Tsigaridas, "Decomposing tensors with structured matrix factors reduces to rank-1 approximations," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP 2010)*, Dallas, USA, Mar. 2010.

[CtBdLC09] P. Comon, J. M. F. ten Berge, L. de Lathauwer, and J. Castaing, "Generic and typical ranks of multi-way arrays," *Linear Algebra and its Applications*, vol. 430, pp. 2997–3007, Jun. 2009.

[CWS01] P. Chargé, Y. Wang, and J. Saillard, "A non circular sources direction finding method using polynomial rooting," *Signal Processing*, no. 81, pp. 1765–1770, 2001.

[DA04] J. P. Delmas and H. Abeida, "Stochastic Cramér-Rao bound for noncircular signals with application to DOA estimation," *IEEE Transactions on Signal Processing*, vol. 52, no. 11, pp. 3192–3199, 2004.

[DD94] M. A. Doron and E. Doron, "Wavefield modeling and array processing, part II—Algorithm," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2571–2580, Oct. 1994.

[DF03] D. S. Dummit and R. M. Foote, *Abstract Algebra*, 3rd ed. Wiley, 2003.

[DHL+11] M. Dohler, R. W. Heath, Jr., A. Lozano, C. B. Papadias, and R. A. Valenzuela, "Is the PHY layer dead?" *IEEE Communications Magazine*, vol. 49, no. 4, pp. 159–165, Apr. 2011.

[dL04a] L. de Lathauwer, "Computation of the canonical decomposition by means of a simultaneous generalized Schur decomposition," *SIAM Journal Matrix. Anal. Appl.*, vol. 26, no. 2, pp. 295–327, 2004.

[dL04b] ——, "First-order perturbation analysis of the best rank-(r_1, r_2, r_3) approximation in multilinear algebra," *Journal of Chemometrics*, vol. 18, no. 1, pp. 2–11, 2004.

[dL06] ——, "A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalizations," *SIAM Journal Matrix. Anal. Appl.*, vol. 28, no. 3, pp. 642–666, 2006.

[dL08] ——, "Decompositions of a higher-order tensor in block terms — part II: Definitions and uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1033–1066, 2008.

[dLdB08] L. de Lathauwer and A. de Baynast, "Blind deconvolution of DS-CDMA signals by means of decomposition in rank-$(1, l, l)$ terms," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1562–1571, Apr. 2008.

[dLdMV00a] L. de Lathauwer, B. de Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[dLdMV00b] ——, "On the best rank-1 and rank-$(r_1, r_2, \ldots, r_n)$ approximation of higher-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, p. 1324–1342, 2000.

[dSL08] V. de Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Anal. Appl.*, vol. 30, pp. 1084–1127, 2008.

[dVVdL+07] M. de Vos, A. Vergult, L. de Lathauwer, W. de Clercq, S. van Huffel, P. Dupont, A. Palmini, and W. V. Paesschen, "Canonical decomposition of ictal scalp EEG reliably detects the seizure onset zone," *NeuroImage*, vol. 37, pp. 844–854, 2007.

[EJS82] J. Evans, J. Johnson, and D. Sun, "Application of advanced signal processing techniques to angle of arrival estimation in ATC navigation and surveillance systems," Tech. Rep. 582, M.I.T. Lincoln Laboratory, Cambridge, MA, Tech. Rep., Jun. 1982.

[EK06] J. Eriksson and V. Koivunen, "Complex random vectors and ICA models: Identifiability, uniqueness, and separability," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1017–1029, Mar. 2006.

[ES94] A. Eriksson and P. Stoica, "Optimally weighted ESPRIT for direction estimation," *Signal Processing*, vol. 38, pp. 223–229, 1994.

[ESS93] A. Eriksson, P. Stoica, and T. Söderström, "Second-order properties of MUSIC and ESPRIT estimates of sinusoidal frequencies in high SNR scenarios," *IEE Proceedings-F*, vol. 140, no. 4, Aug. 1993.

[EY36] C. Eckart and G. Young, "The approximation of one matrix by another one of lower rank," *Psychometrika*, vol. 1, pp. 211–218, 1936.

[FG06] T. Fu and X. Gao, "Simultaneous diagonalization with similarity transformation for non-defective matrices," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, Toulouse, France, May 2006.

[Fri90] B. Friedlander, "A sensitivity analysis of MUSIC algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-38, pp. 1740–1751, 1990.

[FW92] B. Friedlander and A. J. Weiss, "Direction finding using spatial smoothing with interpolated arrays," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 28, pp. 574–587, Apr. 1992.

[GCN08] F. Gao, T. Cui, and A. Nallanathan, "On channel estimation and optimal training design for amplify and forward relay networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 5, pp. 1907–1916, May 2008.

[GNW08] F. Gao, A. Nallanathan, and Y. Wang, "Improved MUSIC under the coexistence of both circular and noncircular sources," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3033–3038, Jul. 2008.

[GR11] J. Gora and S. Redana, "In-band and Out-band relaying configurations for dual-carrier LTE-Advanced system," in *IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2011)*, Toronto, Canada, Sep. 2011.

[GRP10] A. B. Gershman, M. Rübsamen, and M. Pesavento, "One- and two-dimensional direction-of-arrival estimation: An overview of search-free techniques," *Signal Processing*, vol. 90, pp. 1338–1349, 2010.

[GSL03] W. H. Gerstacker, R. Schober, and A. Lampe, "Receivers with widely linear processing for frequency-selective channels," *IEEE Transactions on Communications*, vol. 51, no. 9, pp. 1512–1523, Sep. 2003.

[GvL80] G. H. Golub and C. F. van Loan, "An analysis of the total least squares problem," *Numer. Anal.*, vol. 17, pp. 883–893, 1980.

[GvL96] ——, *Matrix computations*, 3rd ed. John Hopkins University Press, Oct. 1996.

[GZL09a] F. Gao, R. Zhang, and Y.-C. Liang, "Channel estimation for OFDM modulated two-way relay networks," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4443–4455, Nov. 2009.

[GZL09b] ——, "Optimal channel estimation and training design for two-way relay networks," *IEEE Transactions on Communications*, vol. 57, no. 10, pp. 3024–3033, Oct. 2009.

[GZMB10] X. Guo, S. Zhu, S. Miron, and D. Brie, "Approximate joint diagonalization by nonorthogonal nonparametric Jacobi transformations," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2010)*, Dallas, TX, Mar. 2010.

[Haa97a] M. Haardt, "Efficient One-, Two-, and Multidimensional High-Resolution Array Signal Processing," Ph.D. dissertation, Technische Universität München, Shaker Verlag, Aachen, 1997.

[Haa97b] ——, "Structured least squares to improve the performance of ESPRIT-type algorithms," *IEEE Transactions on Signal Processing*, vol. 45, pp. 792–799, Mar. 1997.

[Har70] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis," *UCLA working papers in phonetics*, vol. 16, pp. 1–84, 1970.

[Har72] ——, "PARAFAC2: Mathematical and technical notes," *UCLA Working Papers in Phonetics*, vol. 22, pp. 30–44, 1972.

[Has90] J. Hastad, "Tensor rank is NP-complete," *J. Algorithms*, vol. 11, pp. 644–654, 1990.

[HF94] G. F. Hatke and K. W. Forsythe, "A class of polynomial rooting algorithm for joint azimuth/elevation estimation using multidimensional arrays," in *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1994, pp. 694–699.

[Hit27] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, vol. 6, pp. 164–189, 1927.

[HK81] T. Han and K. Kobayashi, "A new achievable rate region for the interference channel," *IEEE Transactions on Information Theory*, vol. IT-27, no. 1, pp. 49–60, Jan. 1981.

[HKE+07] I. Hammerstrom, M. Kuhn, C. Esli, J. Zhao, A. Wittneben, and G. Bauch, "MIMO two-way relaying with transmit CSI at the relay," in *Proc. IEEE 8th Workshop on Sig. Proc. Adv. in Wireless Comm. (SPAWC 2007)*, Helsinki, Finland, Jun. 2007.

[HKT05] W. Hackbusch, B. N. Khoromskij, and E. E. Tyrtyshnikov, "Hierarchical Kronecker tensor-product approximations," *Journal of Numerical Mathematics*, vol. 13, pp. 119–156, 2005.

[HN95] M. Haardt and J. A. Nossek, "Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1232–1242, May 1995.

[HN98] ——, "Simultaneous Schur decomposition of several non-symmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems," *IEEE Transactions on Signal Processing*, vol. 46, no. 1, pp. 161–169, Jan. 1998.

[HNSG10] V. Havary-Nassab, S. Shahbazpanahi, and A. Grami, "Optimal distributed beamforming for two-way relay networks," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1238–1250, Mar. 2010.

[HS91] Y. Hua and T. K. Sarkar, "On SVD for estimating generalized eigenvalues of singular matrix pencil in noise," *IEEE Transactions on Signal Processing*, vol. 39, pp. 892–900, Apr. 1991.

[HTR04] M. Haardt, R. S. Thomä, and A. Richter, "Multidimensional high-resolution parameter estimation with applications to channel sounding," in *High-Resolution and Robust Signal Processing*, Y. Hua, A. Gershman, and Q. Chen, Eds. New York, NY: Marcel Dekker, 2004, pp. 255–338, chapter 5.

[IAvHdL10] M. Ishteva, P.-A. Absil, S. van Huffel, and L. de Lathauwer, "Local minima of the best low multilinear rank approximation of tensors," in *Proc. of the 19th International Symposium on Mathematical Theory of Networks and Systems (MTNS 2010)*, Budapest, Hungary, Jul. 2010.

[JGGN10] B. Jiang, F. Gao, X. Gao, and A. Nallanathan, "Channel estimation and training design for two-way relay networks with power allocation," *IEEE Transactions on Wireless Communications*, vol. 9, no. 6, pp. 2022–2032, Jun. 2010.

[JLL09] M. Jin, G. Liao, and J. Li, "Joint DOD and DOA estimation for bistatic MIMO radar," *Signal Processing*, vol. 89, no. 2, pp. 244–251, Feb. 2009.

[JS10a] E. Jorswieck and A. Sezgin, "Transmit strategies for the MIMO two-way amplify-forward channel with multiple relays and MMSE receiver," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP 2010)*, Dallas, TX, Mar. 2010.

[JS10b] J. Joung and A. H. Sayed, "Multiuser two-way amplify-and-forward relay processing and power control methods for beamforming systems," *IEEE Transactions on Signal Processing*, vol. 58, pp. 1833–1846, Mar. 2010.

[KAB83] S. Y. Kung, K. S. Arun, and D. V. Bhaskar Rao, "State space and SVD based approximation methods for the harmonic retrieval problem," *Journal of the Optical Society of America*, vol. 73, pp. 1799–1811, Dec. 1983.

[KB86] M. Kaveh and A. J. Barabell, "The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, pp. 331–341, Apr. 1986.

[KB06] T. Kolda and B. Bader, "The TOPHITS model for higher-order web link analysis," in *Proceedings of the SIAM Data Mining Conference and the Workshop on Link Analysis, Counterterrorism and Security*, Bethesda, MD, Apr. 2006.

[KB09] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, Sep. 2009.

[KdL80] P. M. Kroonenberg and J. de Leeuw, "Principal component analysis of three-mode data by means of alternating least squares algorithms," *Psychometrika*, vol. 45, no. 1, pp. 69–97, Mar. 1980.

[KGK07] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," in *Proc. ACM Conf. Appl., Techn., Archit., and Prot. for Comp. Comm. (SIGCOMM 2007)*, Kyoto, Japan, Aug. 2007, pp. 497–408.

[KH11] T. Kong and Y. Hua, "Optimal design of source and relay pilots for MIMO relay channel estimation," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4438–4446, Sep. 2011.

[KKE11] A. Khina, Y. Kochman, and U. Erez, "Physical-layer MIMO relaying," in *Proc. Int. Symposium on Information Theory (ISIT 2011)*, St. Petersburg, Russia, Aug. 2011, pp. 2437–2441.

[KNW86] A. Kapteyn, H. Neudecker, and T. Wansbeek, "An approach to n-mode components analysis," *Psychometrica*, vol. 51, pp. 269–275, 1986.

[KR68] C. G. Khatri and C. R. Rao, "Solutions to some functional equations and their applications to characterization of probability distributions," *Sankhya: The Indian Journal of Statistics*, vol. series A, 30, pp. 167–180, 1968.

[Kru77] J. B. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra Applications*, vol. 18, pp. 95–138, 1977.

[Kru89] ——, "Rank, decomposition, and uniqueness for 3-way and N-way arrays," in *Multiway Data Analysis*, R. Coppi and S. Bolasco, Eds. North-Holland, Amsterdam, 1989, pp. 7–18.

[KT83] R. Kumaresan and D. W. Tufts, "Estimating the angles of arrival of multiple plane waves," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 134–139, Jan. 1983.

[KYKI10] H.-M. Kim, T.-W. Yune, D. Kim, and G.-H. Im, "Decision-directed channel estimation for SC-FDE in amplify-and-forward relaying networks," *IEEE Communication Letters*, vol. 14, no. 3, Mar. 2010.

[LA11] X. Luciani and L. Albera, "Semi-algebraic canonical decomposition of multi-way arrays and joint eigenvalue decomposition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2011)*, Prague, Czech Republic, May 2011.

[LB08] D. Letexier and S. Bourennane, “Noise removal from hyperspectral images by multidimensional filtering,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 7, Jul. 2008.

[LCSK10] N. Lee, C.-B. Chae, O. Simeone, and J. Kang, “On the optimization of two-way AF MIMO relay channel with beamforming,” in *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2010.

[Lee80] A. Lee, “Centrohermitian and skew-centrohermitian matrices,” *Linear Algebra and its Applications*, vol. 29, pp. 205–210, 1980.

[LJS06] P. Larsson, N. Johansson, and K.-E. Sunell, “Coded bidirectional relaying,” in *Proc. of the IEEE 63rd Vehicular Technology Conference (VTC ’06)*, vol. 2, Melbourne, Australia, May 2006, pp. 851–855.

[LL06] J. Liu and X. Liu, “An eigenvector-based approach for multidimensional frequency estimation with improved identifiability,” *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4543–4556, Dec. 2006.

[LLM08] J. Liu, X. Liu, and X. Ma, “First-order perturbation analysis of singular vectors in singular value decomposition,” *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3044–3049, Jul. 2008.

[LLV93] F. Li, H. Liu, and R. J. Vaccaro, “Performance analysis for DOA estimation algorithms: Unification, simplifications, and observations,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 4, pp. 1170–1184, Oct. 1993.

[LLXZ12] A. Liu, G. Liao, Q. Xu, and C. Zeng, “A circularity-based DOA estimation method under coexistence of noncircular and circular signals,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Kyoto, Japan, Mar. 2012.

[LP93] C. F. V. Loan and N. Pitsianis, “Approximation with Kronecker products,” *Linear Algebra for Large-Scale and Real-Time Applications*, pp. 293–314, 1993.

[LPC08] N. Lee, H. Park, and J. Chun, “Linear precoder and decoder design for two-way AF MIMO relaying system,” in *Proc. IEEE Vehicular Technology Conference (VTC)*, May 2008, pp. 1221–1225.

[LSPL10] K.-J. Lee, H. Sung, H. Park, and I. Lee, “Joint optimization for one and two-way MIMO AF multiple-relay systems,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 12, pp. 3671–3681, Dec. 2010.

[LSS06] X. Lin, N. B. Shroff, and R. Srikant, “A tutorial on cross-layer optimization in wireless networks,” *IEEE Journal on Selected Areas in Communications*, no. 8, Aug. 2006.

[LT78] P. Lancaster and M. Tismentsky, *The Theory of Matrices*, 2nd ed. New York Acadmic Press, 1978.

[LV92] F. Li and R. J. Vaccaro, "Performance degradation of DOA estimators due to unknown noise fields," *IEEE Transactions on Signal Processing*, vol. 40, no. 3, pp. 686–690, Mar. 1992.

[LV08] P. Lioliou and M. Viberg, "Least-squares based channel estimation for MIMO relays," in *Proc. ITG/IEEE Workshop on Smart Antennas (WSA '08)*, Darmstadt, Germany, Feb. 2008, pp. 90–95.

[LVBL98] M. S. Lobo, L. Vandenberghe, S. Boys, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, no. 1–3, pp. 193–228, Nov. 1998.

[LY06a] K. Lee and A. Yener, "Iterative power allocation algorithms for Amplify/Estimate/Compress-and-Forward multi-band relay channels," in *40th Annual Conference on Information Sciences and Systems (CISS 2006)*, Princeton, NJ, Mar. 2006.

[LY06b] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, Aug. 2006.

[LYC08] N. Lee, H.-J. Yang, and J. Chun, "Achievable sum-rate maximizing AF relay beamforming scheme in two-way relay channels," in *Proc. IEEE Intl. Conf. Comm. (ICC 2008)*, Beijing, China, May 2008.

[LZZL09] M. Liu, J. Zhang, Y. Zhang, and Y. Liu, "A channel estimation scheme for amplify-and-forward OFDM relay networks," in *Proc. Vehicular Technology Conference (VTC Fall)*, Anchorage, AK, Sep. 2009.

[Möc88] J. Möcks, "Topographic components model for event-related potentials and some biophysical considerations," *IEEE Transactions on Biomedical Engineering*, vol. 35, pp. 482–484, 1988.

[MGH08] M. Milojevic, G. D. Galdo, and M. Haardt, "Tensor-based framework for the prediction of frequency-selective time-variant MIMO channels," in *Proc. International ITG/IEEE Workshop on Smart Antennas (WSA'08)*, Darmstadt, Germany, Feb. 2008, pp. 147–152.

[MHH+06] M. Mørup, L. K. Hansen, C. S. Herrmann, J. Parnas, and S. M. Arnfred, "Parallel factor analysis as an exploratory tool for wavelet transformed event-related EEG," *NeuroImage*, vol. 29, pp. 938–947, 2006.

[MHZ96] C. P. Mathews, M. Haardt, and M. D. Zoltowski, "Performance analysis of closed-form, ESPRIT based 2-D angle estimator for rectangular arrays," *IEEE Signal Processing Letters*, vol. 3, pp. 124–126, Apr. 1996.

[MLM05] S. Miron, N. Le Bihan, and J. I. Mars, "Vector-sensor MUSIC for polarized seismic sources localization," *EURASIP Journal on Applied Signal Processing*, vol. 10, pp. 74–84, 2005.

[MN79] J. R. Magnus and H. Neudecker, "The commutation matrix: some properties and applications," *The Annals of Statistics*, vol. 7, pp. 381–394, 1979.

[MN95] ——, *Matrix differential calculus with applications in statistics and econometrics.* John Wiley and Sons, 1995.

[Moo20] E. H. Moore, "On the reciprocal of the general algebraic matrix," *Bulletin of the American Mathematical Society*, vol. 26, pp. 394–395, 1920.

[MOZL09] J. Ma, P. Orlik, J. Zhang, and G. Y. Li, "Pilot matrix design for interim channel estimation in two-hop MIMO AF relay systems," in *Proc. Intl. Comm. Conf. (ICC 2009)*, Dresden, Germany, Jun. 2009.

[MSPM04] K. N. Mokios, N. D. Sidiropoulos, M. Pesavento, and C. E. Mecklenbräuker, "On 3-D harmonic retrieval for wireless channel sounding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, Toulouse, Canada, May 2004, pp. 89–92.

[Mun00] J. Munkres, *Topology*, 2nd ed. Prentice Hall, 2000.

[MvL08] C. D. M. Martin and C. F. van Loan, "A Jacobi-type method for computing orthogonal tensor decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 30, pp. 1219–1232, 2008.

[MZ94] C. P. Mathews and M. D. Zoltowski, "Performance analysis of the UCA-ESPRIT algorithm for circular ring arrays," *IEEE Transactions on Signal Processing*, vol. 42, no. 9, Sep. 1994.

[NBK04] R. U. Nabar, H. Bölcskei, and F. W. Kneubühler, "Fading relay channels: performance limits and space-time signal design," *IEEE Journal on Selected Areas in Communications*, vol. 22, pp. 1099–1109, Aug. 2004.

[NdLK08] C. Navasca, L. de Lathauwer, and S. Kindermann, "Swamp reducing technique for tensor decompositions," in *Proceedings of the 16th European Signal Processing Conference (EUSIPCO 2008)*, Lausanne, Switzerland, Aug. 2008.

[Neu69] H. Neudecker, "Some theorems on matrix differentiation with special reference to Kronecker matrix products," *Journal of the American Statistical Association*, vol. 64, pp. 953–963, 1969.

[NM93] F. D. Neeser and J. L. Massey, "Proper complex random processes with applications to information theory," *IEEE Transactions on Information Theory*, vol. 39, no. 4, Jul. 1993.

[NS09] D. Nion and N. D. Sidiropoulos, "Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor," *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2299–2310, Jun. 2009.

[NS10] ——, "Tensor algebra and multidimensional harmonic retrieval in signal processing for MIMO radar," *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5693–5705, Nov. 2010.

[OB07] T. J. Oechtering and H. Boche, "Bidirectional relaying using interference cancellation," in *Proc. ITG/IEEE Int. Workshop on Smart Antennas (WSA '07)*, Vienna, Austria, Feb. 2007.

[Oll08] E. Ollila, "On the circularity of a complex random variable," *IEEE Signal Processing Letters*, vol. 15, pp. 841–844, 2008.

[OS99] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. New Jersey: Prentice-Hall, 1999.

[OSBB08] T. J. Oechtering, C. Schnurr, I. Bjelakovic, and H. Boche, "Broadcast capacity region of two-phase bidirectional relaying," *IEEE Transactions on Information Theory*, vol. 54, no. 1, pp. 454–458, Jan. 2008.

[Ose11] I. V. Oseledets, "Tensor-Train decompositions," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.

[OVK91] B. Ottersten, M. Viberg, and T. Kailath, "Performance analysis of the Total Least Squares ESPRIT algorithm," *IEEE Transactions on Signal Processing*, vol. 39, no. 5, pp. 1122–1135, May 1991.

[OWS88] H. Ouibrahim, D. D. Weiner, and T. K. Sarkar, "Matrix pencil approach to angle-of-arrival estimation," in *Proceedings of the 20th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1988, pp. 203–206.

[Paa00] P. Paatero, "Construction and analysis of degenerate PARAFAC models," *J. Chemometrics*, vol. 14, pp. 285–289, 2000.

[PB97] B. Picinbono and P. Bondon, "Second-order statistics of complex signals," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 411–420, Feb. 1997.

[PC95] B. Picinbono and P. Chevalier, "Widely linear estimation with complex data," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, Aug. 1995.

[PDF+08] S. Parkvall, E. Dahlman, A. Furuskar, Y. Jading, M. Olsson, S. Wanstedt, and K. Zangi, "LTE-advanced - evolving LTE towards IMT-advanced," in *Proc. IEEE Vehicular Technology Conference (VTC Fall 2008)*, Calgary, AB, Canada, Sep. 2008.

[PdLvH05] J. M. Papy, L. de Lathauwer, and S. van Huffel, "Exponential data fitting using multilinear algebra: The single-channel and multi-channel case," *Numerical Linear Algebra with Applications*, vol. 12, pp. 809–826, Jun. 2005.

[Pen55] R. Penrose, "A generalized inverse for matrices," *Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, 1955.

[PF88] B. Porat and B. Friedlander, "Analysis of the asymptotic relative efficiency of the MUSIC algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 4, pp. 532–544, Apr. 1988.

[PGH00] M. Pesavento, A. B. Gershman, and M. Haardt, "On unitary Root-MUSIC with a real-valued eigendecomposition: A theoretical and experimental performance study," *IEEE Transactions on Signal Processing*, vol. 48, pp. 1306–1314, May 2000.

[PGW02] M. Pesavento, A. B. Gershman, and K. M. Wong, "Direction finding in partly calibrated sensor arrays composed of multiple subarrays," *IEEE Transactions on Signal Processing*, vol. 50, no. 9, pp. 2103–2115, Sep. 2002.

[Pic94] B. Picinbono, "On circularity," *IEEE Transactions on Signal Processing*, vol. 42, no. 12, pp. 3473–3482, Dec. 1994.

[Pic96] ——, "Second-order complex random vectors and normal distributions," *IEEE Transactions on Signal Processing*, vol. 44, no. 10, Oct. 1996.

[Pis73] V. F. Pisarenko, "The retrieval of harmonics from a covariance function," *Geophys. J. Royal Astronomical Soc.*, vol. 33, pp. 347–366, 1973.

[PK89a] S. U. Pillai and B. H. Kwon, "Forward / backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. AASP-37, pp. 8–15, Jan. 1989.

[PK89b] ——, "Performance analysis of MUSIC-type high resolution estimators for direction finding in correlated and coherent scenes," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. AASP-37, no. 8, Aug. 1989.

[PLNG10] T.-H. Pham, Y.-C. Liang, A. Nallanathan, and H. K. Garg, "Optimal training sequences for channel estimation in bi-directional relay networks with multiple antennas," *IEEE Trans. on Communications*, vol. 58, no. 2, pp. 474–479, Feb. 2010.

[PMB04] M. Pesavento, C. F. Mecklenbräuker, and J. F. Böhme, "Multi-dimensional rank reduction estimator for parametric MIMO channel estimation," *EURASIP J. Appl. Signal Proc.*, pp. 1354–1363, Aug. 2004.

[PNG03] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications.* Cambridge University Press, 2003.

[PP08] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," http://matrixcookbook.com (until 2010), Nov. 2008.

[PWS+04] R. Pabst, B. H. Walke, D. C. Schultz, P. Herhold, H. Yanikomeroglu, S. Mukherjee, H. Viswanathan, M. Lott, W. Zirwas, M. Dohler, H. Aghvami, D. D. Falconer, and G. P. Fettweis, "Relay-based deployment concepts for wireless and mobile broadband radio," *IEEE Communications Magazine*, vol. 42, pp. 80 – 89, Sep. 2004.

[RA92] B. D. Rao and K. S. Arun, "Model based processing of signals: A state space approach," *Proceedings of the IEEE*, vol. 80, pp. 283–309, Feb. 1992.

[RC05] M. Rajih and P. Comon, "Enhanced line search: A novel method to accelerate PARAFAC," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO 2005)*, Antalya, Turkey, Sep. 2005.

[RCH08] M. Rajih, P. Comon, and R. Harshman, "Enhanced line search: A novel method to accelerate PARAFAC," *SIAM J. Matr. Anal. Appl.*, vol. 30, no. 3, pp. 1148–1171, 2008.

[RG09] M. Rübsamen and A. Gershman, "Direction-of-arrival estimation for non-uniform sensor arrays: from manifold separation to Fourier domain MUSIC methods," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 588–599, Feb. 2009.

[RH89a] B. D. Rao and K. V. S. Hari, "Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1990–1995, Dec. 1989.

[RH89b] ——, "Performance analysis of Root-MUSIC," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1939–1948, Dec. 1989.

[RHST01] A. Richter, D. Hampicke, G. Sommerkorn, and R. S. Thomä, "MIMO measurement and joint M-D parameter estimation of mobile radio channels," in *Proc. IEEE Vehicular Technology Conference (VTC Spring 2001)*, Rhodes, Greece, May 2001.

[RK87] R. Roy and T. Kailath, "Total least-squares ESPRIT," in *Proc. 21st Asilomar Conf. Circuits, Syst., Computing*, Pacific Grove, CA, Nov. 1987.

[RKX12] Y. Rong, M. R. A. Khandaker, and Y. Xiang, "Channel estimation of dual-hop MIMO relay system via parallel factor analysis," *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 2224–2233, Jun. 2012.

[Roe06] F. Roemer, "Advances in subspace-based parameter estimation: Tensor-ESPRIT-type methods and non-circular sources," Diplomarbeit, Ilmenau University of Technology, Communications Research Laboratory, Germany, Oct. 2006.

[RPK86] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT - a subspace rotation approach to estimation of parameters of cisoids in noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. AASP-34, pp. 1340–1342, Oct. 1986.

[Rud76] W. Rudin, *Principles of Mathematical Analysis*, 3rd ed. McGraw-Hill, 1976.

[RVGS05] Y. Rong, S. A. Vorobyov, A. B. Gershman, and N. D. Sidiropoulos, "Blind Spatial Signature Estimation via Time-Varying User Power Loading and Parallel Factor Analysis," *IEEE Trans. Signal Processing*, vol. 53, no. 5, pp. 1697 – 1710, May 2005.

[RW05] B. Rankov and A. Wittneben, "Spectral efficient signaling for half-duplex relay channels," in *Proceedings of the 39th Annual Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Oct. 2005, pp. 1066–1071.

[RW07] ——, "Spectral efficient protocols for half-duplex fading relay channels," *IEEE Journal on Selected Areas in Communications*, vol. 25, Feb. 2007.

[RWW09] T. Riihonen, S. Werner, and R. Wichman, "Spatial loop interference suppression in full-duplex MIMO relays," in *Proc. 43-rd Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2009, pp. 1508 – 1512.

[RZZ93] C. R. Rao, L. C. Zhao, and B. Zhou, "A novel algorithm for 2-dimensional frequency estimation," in *Proc. 27th Asilomar Conf. Circuits, Syst., Computing*, Pacific Grove, CA, Nov. 1993, pp. 199–202.

[SAKH11] A. Sezgin, A. S. Avestimehr, M. A. Khajehnejad, and B. Hassibi, "Divide-and-conquer: Approaching the capacity of the two-pair bidirectional Gaussian relay network," *IEEE Transactions on Information Theory*, Sep. 2011.

[SBG00] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2377–2388, Aug. 2000.

[Sch79] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," in *in Proc. RADC Spectrum Estimation Workshop*, Oct. 1979.

[Sch85] W. Scharlau, "Quadratic and Hermitian forms," in *Grundlehren der mathematischen Wissenschaften*, vol. 270. Springer, 1985.

[Sch86] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. AP-34, no. 3, pp. 243–258, Mar. 1986.

[Sch91] L. L. Scharf, *Statistical Signal Processing.* Reading, MA: Addison-Wesley Publishing Company, 1991.

[SD12] S. Shahbazpanahi and M. Dong, "A semi-closed-form solution to optimal distributed beamforming for two-way relay networks," *IEEE Transactions on Signal Processing*, no. 3, pp. 1511–1516, Mar. 2012.

[SDHW12] N. Song, R. C. De Lamare, M. Haardt, and M. Wolf, "Adaptive widely linear reduced-rank interference suppression based on the multi-stage Wiener filter," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4003–4016, Aug. 2012.

[SdLW+11] J. Steinwandt, R. C. de Lamare, L. Wang, N. Song, and M. Haardt, "Widely linear adaptive beamforming algorithm based on the conjugate gradient method," in *Proc. International ITG Workshop on Smart Antennas (WSA 2011)*, Aachen, Germany, Feb. 2011.

[SE07] B. Savas and L. Elden, "Handwritten digit classification using higher order singular value decomposition," *Pattern Recognition*, vol. 40, pp. 993–1003, 2007.

[SGB00] N. Sidiropoulos, G. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Transactions on Signal Processing*, vol. 48, pp. 810–823, 2000.

[SH08] V. Stankovic and M. Haardt, "Generalized design of multi-user MIMO precoding matrices," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 953–961, Mar. 2008.

[Sha48] C. E. Shannon, "The mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, Jul. 1948.

[Sha61] ——, "Two-way communication channels," *Proc. 4th Berkeley Symp. on Mathematical Statistics and Probability*, vol. 1, pp. 611–644, 1961.

[SK90] E. Sanchez and B. R. Kowalski, "Tensorial resolution: a direct trilinear decomposition," *J. Chemom.*, vol. 4, pp. 29–45, 1990.

[SK93] A. L. Swindlehurst and T. Kailath, "Azimuth/elevation direction finding using regular array geometries," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 1, pp. 145–156, Jan. 1993.

[SL01] A. Shashua and A. Levin, "Linear image coding for regression and classification using the tensor-rank principle," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Kauai, HI, USA, Dec. 2001, pp. 42–49.

[SMB01] M. Steinbauer, A. F. Molisch, and E. Bonek, "The double-directional radio channel," *IEEE Transactions on Antennas and Propagation*, vol. 43, no. 4, Aug. 2001.

[Smi05] S. T. Smith, "Statistical resolution limits and the complexified Cramér-Rao bound," *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1597–1609, May 2005.

[SN89] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramér-Rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[SORK92] A. L. Swindlehurst, B. Ottersten, R. Roy, and T. Kailath, "Multiple invariance ESPRIT," *IEEE Transactions on Signal Processing*, vol. 40, no. 4, pp. 867–881, Apr. 1992.

[SS91] P. Stoica and T. Söderström, "Statistical analysis of MUSIC and subspace rotation estimates of sinusoidal frequencies," *IEEE Transactions on Signal Processing*, vol. 39, no. 8, Aug. 1991.

[SS03] P. J. Schreier and L. L. Scharf, "Second-order analysis of improper complex random vectors and processes," *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 714–725, Mar. 2003.

[SS05] ——, "Detection and estimation of improper complex random signals," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 306–312, Jan. 2005.

[SS10] ——, *Statistical Signal Processing of Complex-Valued Data: The Theory of Improper and Noncircular Signals.* Cambridge University Press, 2010.

[SSH04] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multi-user MIMO channels," *IEEE Transactions on Signal Processing*, vol. 52, pp. 461–471, Feb. 2004.

[SSW+11] N. Song, J. Steinwandt, L. Wang, R. C. de Lamare, and M. Haardt, "Non-data-aided adaptive beamforming algorithm based on the widely linear auxiliary vector filter," in *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP 2011)*, Prague, Czech Republic, May 2011, pp. 2636–2639.

[Ste93] G. W. Stewart, "On the early history of the singular value decomposition," *SIAM Review*, vol. 35, no. 4, pp. 551–566, 1993.

[STF06] J. Sun, D. Tao, and C. Faloutsos, "Beyond streams and graphs: Dynamic tensor analysis," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '06).* Philadelphia, PA: ACM Press, Aug. 2006, pp. 374–383.

[Str93] G. Strang, *Introduction to Linear Algebra.* Wellesley, MA: Wellesley-Cambridge Press, 1993.

[SWK85] T. J. Shan, M. Way, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. AASP-33, pp. 806–811, Aug. 1985.

[Tak24] T. Takagi, "On an algebraic problem related to an analytic theorem of Carathédory and Fejér and on an allied theorem of Landau," *Japan J. Math*, vol. 1, pp. 82–93, 1924.

[Tel99] E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecomm. ETT*, vol. 10, no. 6, pp. 585–596, Nov. 1999.

[Ten91] J. M. F. Ten Berge, "Kruskal's polynomial for 2 x 2 x 2 arrays and a generalization to 2 x n x n arrays," *Psychometrika*, vol. 56, pp. 631–636, 1991.

[Ten00] ——, "Kruskal's polynomial for $2 \times 2 \times 2$ arrays and a generalization to $2 \times n \times n$ arrays," *Psychometrika*, vol. 65, pp. 525–532, 2000.

[THG09a] A. Thakre, M. Haardt, and K. Giridhar, "Single snapshot r-d unitary esprit using an augmentation of the tensor order," in *Prof. IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2009)*, Dec. 2009.

[THG09b] ——, "Single snapshot spatial smoothing with improved effective array aperture," *IEEE Signal Processing Letters*, vol. 16, pp. 505–509, Jun. 2009.

[THR+00] R. S. Thomä, D. Hampicke, A. Richter, G. Sommerkorn, A. Schneider, U. Trautwein, and W. Wirnitzer, "Identification of time-variant directional mobile radio channels," *IEEE Trans. on Instrumentation and Measurement*, vol. 49, pp. 357–364, Apr. 2000.

[TLRT05] R. S. Thomä, M. Landmann, A. Richter, and U. Trautwein, "Multidimensional high-resolution channel sounding," in *Smart Antennas in Europe State of the Art, EURASIP Book Series on Signal Processing and Communications*, T. Kaiser, Ed., vol. 3. Cairo, Egypt: Hindawi Publishing, 2005.

[TSC88] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance analysis and code construction," *IEEE Transactions on Information Theory*, vol. 44, no. 2, pp. 744–765, 1988.

[TSQ09] L. B. Thiagarajan, S. Sun, and T. Q. S. Quek, "Carrier frequency offset and channel estimation in space-time non-regenerative two-way relay network," in *Proc. IEEE 10th Workshop on Sig. Proc. Adv. in Wireless Comm. (SPAWC 2009)*, Perugia, Italy, Jun. 2009.

[Tuc66] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, pp. 279–311, 1966.

[TW12] M. Tao and R. Wang, "Robust relay beamforming for two-way relay networks," *IEEE Communication Letters*, vol. 16, no. 7, Jul. 2012.

[UK08] T. Unger and A. Klein, "Duplex schemes in multiple antenna two-hop relaying," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, 2008, doi: 10.1155/2008/128592.

[Van00] C. F. Van Loan, "The ubiquitous Kronecker product," *Journal of Computational and Applied Mathematics*, vol. 123, pp. 85–100, 2000.

[Van02] H. L. Van Trees, *Optimum Array Processing: Detection, Estimation, and Modulation Theory, Part IV.* New York: Wiley, 2002.

[VBPP05] D. Vlasic, M. Brand, H. Pfister, and J. Popovic, "Face transfer with multilinear models," *ACM Transactions on Graphics*, vol. 24, pp. 426–433, 2005.

[vdM71] E. C. van der Meulen, "Three terminal communication channels," *Adv. Appl. Probab*, vol. 3, pp. 120–154, 1971.

[vdVP96] A.-J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1136–1155, May 1996.

[VH11] R. Vaze and R. W. Heath Jr, "On the capacity and diversity-multiplexing tradeoff of the two-way relay channel," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4219–4234, Jul. 2011.

[vHV91] S. van Huffel and J. Vandewalle, *The Total Least Squares problem: Computational Aspects and Analysis.* Philadelphia: SIAM, 1991.

[vL09] C. van Loan, "The Kronecker Product SVD," talk, slides available at http://www.cs.cornell.edu/cv , 2009.

[VLV07] S. Venkatesan, A. Lozano, and R. Valenzuela, "Network MIMO: Overcoming intercell interference in indoor wireless systems," in *Proc. 41-st Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2007, pp. 83 – 87.

[VOK91] M. Viberg, B. Ottersten, and T. Kailath, "Detection and estimation in sensor arrays using weighted subspace fitting," *IEEE Transactions on Signal Processing*, vol. 39, no. 11, pp. 2436– 2449, Sep. 1991.

[VRSG05] S. A. Vorobyov, Y. Rong, N. D. Sidiropoulos, and A. B. Gershman, "Robust iterative fitting of multilinear models," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2678–2689, Aug. 2005.

[VT02a] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensor-faces," *Lecture Notes in Computer Science*, vol. 2350, pp. 447–460, 2002.

[VT02b] ——, "Multilinear image analysis for facial recognition," in *Proceedings of the 16th International Conference on Pattern Recognition*, Quebec, Canada, Aug. 2002, pp. 511–514.

[VT04] ——, "Tensortextures: multilinear image-based rendering," *ACM Transactions on Graphics*, vol. 23, pp. 336–342, 2004.

[WA03] H. Wang and N. Ahuja, "Facial expression decomposition," in *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV 2003)*, vol. 2, Nice, France, Oct. 2003, pp. 958–965.

[WdLM11] T. Wang, R. C. de Lamare, and P. D. Mitchell, "Low-complexity set-membershop channel estimation for cooperative wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 6, Jul. 2011.

[WGZT10] G. Wang, F. Gao, X. Zhang, and C. Tellambura, "Superimposed training-based joint CFO and channel estimation for CP-OFDM modulated two-way relay networks," *EURASIP Journal on Wireless Communications and Networking - Special issue on physical-layer network coding for wireless cooperative networks*, vol. 2010, no. 1, Jan. 2010.

[WIN08] WINNER-II D6.13.14, "WINNER II system concept description," IST-4-027756 WINNER II, www.ist-winner.org, Jan. 2008.

[WJS08] K. Werner, M. Jansson, and P. Stoica, "On estimation of covariance matrices with Kronecker product structure," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 478–491, Feb. 2008.

[WT12] R. Wang and M. Tao, "Joint source and relay precoding designs for MIMO two-way relaying based on MSE criterion," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1352–1365, Mar. 2012.

[XH10] S. Xu and Y. Hua, "Source-relay optimization for a two-way MIMO relay system," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Dallas, TX, Mar. 2010.

[XRK94] G. Xu, R. H. Roy, and T. Kailath, "Detection of number of sources via exploitation of centro-symmetry property," *IEEE Transactions on Signal Processing*, vol. 42, no. 1, pp. 102–112, Jan. 1994.

[Xu02] Z. Xu, "Perturbation analysis for subspace decomposition with applications in subspace-based algorithms," *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2820–2830, Nov. 2002.

[YCHY10] W. Yang, Y. Cai, J. Hu, and W. Yang, "Channel estimation for two-way relay OFDM networks," *EURASIP Journal on Wireless Communications and Networking - Special issue on physical-layer network coding for wireless cooperative networks*, vol. 2010, no. 3, Jan. 2010.

[YCP11] H. J. Yang, J. Chun, and A. Paulraj, "Aymptotic capacity of the separated MIMO two-way relay channel," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7542–7554, Nov. 2011.

[Yer02] A. Yeredor, "Non-orthogonal Joint Diagonalization in the Least-Squares Sense With Application in Blind Source Separation," *IEEE Transactions on Signal Processing*, vol. 50, pp. 1545–1553, Jul. 2002.

[YHXM09] Y. Yang, H. Hu, J. Xu, and G. Mao, "Relay technologies for WiMAX and LTE-advanced mobile systems," *IEEE Communications Magazine*, vol. 47, pp. 100–105, Oct. 2009.

[YWK07] B. Yi, S. Wang, and S. Y. Kwon, "On MIMO relay with finite-rate feedback and imperfect channel estimation," in *Proc. IEEE Global Comm. Conf. (GLOBECOM 2007)*, Washington, DC, Nov. 2007, pp. 3878–3882.

[ZCW03] A. Zoubir, P. Chargé, and Y. Wang, "Non circular sources localization with ESPRIT," in *Proc. European Conference on Wireless Technology (ECWT 2003)*, Munich, Germany, Oct. 2003.

[ZFDW00] T. Zwick, C. Fischer, D. Didascalou, and W. Wiesbeck, "A stochastic spatial channel model based on wave-propagation modeling," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 1, pp. 6–15, Jan. 2000.

[ZHM+00] T. Zwick, D. Hampicke, J. Maurer, A. Richter, G. Sommerkorn, R. Thomä, and W. Wiesbeck, "Results of double-directional channel sounding measurements," in *Proc. IEEE Vehicular Technology Conference (VTC Spring 2000)*, Tokyo, Japan, May 2000, pp. 2497–2501.

[ZHR+04] T. Zwick, D. Hampicke, A. Richter, G. Sommerkorn, R. Thomä, and W. Wiesbeck, "A novel antenna concept for double-directional channel measurements," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 2, pp. 527–537, Mar. 2004.

[ZKB12] Y. Zhang, M.-A. Khalighi, and S. Bourennane, "Iterative channel estimation and data detection for amplify-and-forward relay networks," *IEEE Communication Letters*, vol. 16, no. 5, pp. 710–713, May 2012.

[ZKM92] M. D. Zoltowski, G. M. Kautz, and C. P. Mathews, "Performance analysis of eigenstructure based DOA estimators employing conjugate symmetric beamformers," in *IEEE Sixth SP Workshop on Statistical Signal and Array Processing*, Victoria, BC, Canada, Oct. 1992.

[ZKWB08] J. Zhan, M. Kuhn, A. Wittneben, and G. Bauch, "Self-interference aided channel estimation in two-way relaying systems," in *Proc. IEEE Global Comm. Conf. (GLOBECOM 2008)*, New Orleans, LA, Dec. 2008.

[ZLCC09] R. Zhang, Y.-C. Liang, C. C. Chai, and S. Cui, "Optimal beamforming for two-way multi-antenna relay channel with analogue network coding," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 699–712, Jun. 2009.

[ZLW04] X. Zhengyuan, P. Liu, and X. Wang, "Blind multiuser detection: From MOE to subspace methods," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 510–524, Feb. 2004.

[Zol88] M. Zoltowski, "Novel techniques for estimation of signal parameters based on matrix pencil, subspace rotations, and total least squares," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 4, New York, NY, Apr. 1988, pp. 2861–2864.

[ZRH12c] J. Zhang, F. Roemer, and M. Haardt, "Distributed beamforming for two-way relaying networks with individual power constraints," in *Proc. 46-th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2012.

[ZS89] M. Zoltowski and D. Stavrinides, "Sensor array signal processing via a Procrustes rotations based eigenanalysis of the ESPRIT data pencil," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 6, pp. 832–861, Jun. 1989.

[ZZC11] M. Zeng, R. Zhang, and S. Cui, "On design of collaborative beamforming for two-way relay networks," *IEEE Transactions on Signal Processing*, no. 5, May 2011.

Index

Erklärung

Ich versichere, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Bei der Auswahl und Auswertung folgenden Materials haben mir die nachstehend aufgeführten Personen in der jeweils beschriebenen Weise entgeltlich/unentgeltlich geholfen:

1. ....
2. ....
3. ....

Weitere Personen waren an der inhaltlich-materiellen Erstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlungs- bzw. Beratungsdiensten (Promotionsberater oder anderer Personen) in Anspruch genommen. Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalte der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer Prüfungsbehörde vorgelegt.

Ich bin darauf hingewiesen worden, dass die Unrichtigkeit der vorstehenden Erklärung als Täuschungsversuch bewertet wird und gemäß § 7 Abs. 10 der Promotionsordnung den Abbruch des Promotionsverfahrens zur Folge hat.

(Ort, Datum) (Unterschrift)