



TECHNISCHE UNIVERSITÄT
ILMENAU

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

Advanced Tensor Based Signal Processing Techniques for Wireless Communication Systems and Biomedical Signal Processing

Kristina Naskovska

Technische Universität Ilmenau
Fakultät für Elektrotechnik und Informationstechnik
Institut für Informationstechnik
Fachgebiet Nachrichtentechnik



Advanced Tensor Based Signal Processing Techniques for Wireless Communication Systems and Biomedical Signal Processing

Kristina Naskovska
geboren am 05.05.1989 in Skopje

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

Anfertigung im:	Fachgebiet Nachrichtentechnik Institut für Informationstechnik Fakultät für Elektrotechnik und Informationstechnik
Gutachter:	Prof. Dr.-Ing. Martin Haardt Prof. Dr. André L. F. de Almeida Prof. Dr.-Ing. habil. Jens Haueisen
Tag der Einreichung:	26.02.2019
Tag der wissenschaftlichen Aussprache:	29.08.2019

urn:nbn:de:gbv:ilm1-2019000451

Acknowledgements

This thesis is a result of my work at Communication Research Laboratory, Ilmenau University of Technology and it would not have been possible without the gracious support of many people. Therefore, I would like to thank everyone who has helped me, supported me, guided me, or taught me.

I would especially like to thank my supervisor Prof. Martin Haardt for first giving me the opportunity to be his student and to work among his side. Moreover, I would sincerely like to thank him for the constant support, the creative ideas in the direction of my scientific work, for constantly checking my work, providing me with very useful suggestions in order to improve my work as well as for training and improving my soft skills. Simply, I thank him for all his teachings without which this work would not have been possible. Likewise, my special thanks goes to Prof. André L. F. de Almeida and Prof. Jens Haueisen that have spent time and effort in reviewing my thesis. In addition, my sincere thanks goes to Prof. André L. F. de Almeida for the cooperation during all these years as well as for his inspiring suggestions and ideas. I also owe a great debt of gratitude to Prof. Jens Haueisen for providing me with insides and for explaining me the concepts of the biomedical signal processing, as well as for the help in analyzing my results. Furthermore, I would also like to thank the rest of my dissertation examination committee, Prof. Heiko O. Jacobs, Prof. Jochen Seitz, and Prof. Gerald Schuller, for their kind help.

My work at the Communication Research Laboratory was characterized by a very pleasant working atmosphere and many moments that will be remembered. Therefore, I would like to thank my colleagues and all of my students. In particular, I would like to thank Sehr Ali Cheema, Yao Cheng, Jianshu Zhang, Mike Wolf, Jens Steinwandt, Bilal Zafar, Mikus Grasis, Khaled Nafez Rauf Ardah, Sepideh Gherekhloo, Liana Khamidullina, Damir Rakhimov, and Marko Hennhöfer. I really appreciate every technical or private discussion and exchange of ideas we had over the past years. Moreover, I extend my thanks to Ms. Christina Patotschka and Mr. Wolfgang Erdtmann for all the administrative help and fight against the chaos caused by staff, students, and bureaucracy.

I am also truly grateful to the employees of the Biomedical Engineering Group for their generous and enlightening guidance and for their help in analyzing my results. In particular, my sincere appreciation goes to Stefan Lau for his guidance and joint work, as well as to Uwe Graichen and Daniel Strohmeier for participating in the analysis of the data. To this end, I would like to thank Alexey Alexandrovich Korobkov from Kazan National Research Technical University n.a. A.N. Tupolev-KAI for being among my side in the analysis of the biomedical data.

Last but not least, I would like to thank my parents Biljana and Pero Naskovski as well as my sister Katerina Naskovska for their unconditional love and support in my life. In addition, thanks to the remaining members of my family and friends as well as my boyfriend Matthias Röding.

Abstract

Many observed signals in signal processing applications including wireless communications, biomedical signal processing, image processing, and machine learning are multi-dimensional. Tensors preserve the multi-dimensional structure and provide a natural representation of these signals/data. Moreover, tensors provide often an improved identifiability. Therefore, we benefit from using tensor algebra in the above mentioned applications and many more. In this thesis, we present the benefits of utilizing tensor algebra in two signal processing areas. These include signal processing for MIMO (Multiple-Input Multiple-Output) wireless communication systems and biomedical signal processing. Moreover, we contribute to the theoretical aspects of tensor algebra by deriving new properties and ways of computing tensor decompositions.

Often, we only have an element-wise or a slice-wise description of the signal model. This representation of the signal model does not reveal the explicit tensor structure. Therefore, the derivation of all tensor unfoldings is not always obvious. Consequently, exploiting the multi-dimensional structure of these models is not always straightforward. We propose an alternative representation of the element-wise multiplication or the slice-wise multiplication based on the generalized tensor contraction operator. Later in this thesis, we exploit this novel representation and the properties of the contraction operator such that we derive the final tensor models.

There exist a number of different tensor decompositions that describe different signal models such as the HOSVD (Higher Order Singular Value Decomposition), the CP/PARAFAC (Canonical Polyadic / PARallel FACtors) decomposition, the BTD (Block Term Decomposition), the PARATUCK2 (PARAFac and TUCker2) decomposition, and the PARAFAC2 (PARallel FACtors2) decomposition. Among these decompositions, the CP decomposition is most widely spread and used. Therefore, the development of algorithms for the efficient computation of the CP decomposition is important for many applications. The SECSI (Semi-Algebraic framework for approximate CP decomposition via SIMultaneous matrix diagonalization) framework is an efficient and robust tool for the calculation of the approximate low-rank CP decomposition via simultaneous matrix diagonalizations. In this thesis, we present five extensions of the SECSI framework that reduce the computational complexity of the original framework and/or introduce constraints to the factor matrices. Moreover, the PARAFAC2 decomposition and the PARATUCK2 decomposition are usually described using a slice-wise notation that can be expressed in terms of the generalized tensor contraction as proposed in this thesis. We exploit this novel representation to derive explicit tensor models for the PARAFAC2 decomposition and the PARATUCK2 decomposition. Furthermore, we use the PARAFAC2 model to derive an ALS (Alternating Least-Squares) algorithm for the computation of the PARAFAC2 decomposition.

Moreover, we exploit the novel contraction properties for element-wise and slice-wise multiplications to model MIMO multi-carrier wireless communication systems. We show that this very general

model can be used to derive the tensor model of the received signal for MIMO-OFDM (Multiple-Input Multiple-Output - Orthogonal Frequency Division Multiplexing), Khatri-Rao coded MIMO-OFDM, and randomly coded MIMO-OFDM systems. We propose the transmission techniques Khatri-Rao coding and random coding in order to impose an additional tensor structure of the transmit signal tensor that otherwise does not have a particular structure. Moreover, we show that this model can be extended to other multi-carrier techniques such as GFDM (Generalized Frequency Division Multiplexing). Utilizing these models at the receiver side, we design several types for receivers for these systems that outperform the traditional matrix based solutions in terms of the symbol error rate.

In the last part of this thesis, we show the benefits of using tensor algebra in biomedical signal processing by jointly decomposing EEG (ElectroEncephaloGraphy) and MEG (MagnetoEncephaloGraphy) signals. EEG and MEG signals are usually acquired simultaneously, and they capture aspects of the same brain activity. Therefore, EEG and MEG signals can be decomposed using coupled tensor decompositions such as the coupled CP decomposition. We exploit the proposed coupled SECSI framework (one of the proposed extensions of the SECSI framework) for the computation of the coupled CP decomposition to first validate and analyze the photic driving effect. Moreover, we validate the effects of skull defects on the measurement EEG and MEG signals by means of a joint EEG-MEG decomposition using the coupled SECSI framework. Both applications show that we benefit from coupled tensor decompositions and the coupled SECSI framework is a very practical tool for the analysis of biomedical data.

Zusammenfassung

Zahlreiche messbare Signale in verschiedenen Bereichen der digitalen Signalverarbeitung, z.B. in der drahtlosen Kommunikation, im Mobilfunk, biomedizinischen Anwendungen, der Bild- oder akustischen Signalverarbeitung und dem maschinellen Lernen sind mehrdimensional. Tensoren erhalten die mehrdimensionale Struktur und stellen eine natürliche Darstellung dieser Signale/Daten dar. Darüber hinaus bieten Tensoren oft eine verbesserte Trennbarkeit von enthaltenen Signalkomponenten. Daher profitieren wir von der Verwendung der Tensor-Algebra in den oben genannten Anwendungen und vielen mehr. In dieser Arbeit stellen wir die Vorteile der Nutzung der Tensor-Algebra in zwei Bereichen der Signalverarbeitung vor: drahtlose MIMO (Multiple-Input Multiple-Output) Kommunikationssysteme und biomedizinische Signalverarbeitung. Darüber hinaus tragen wir zu theoretischen Aspekten der Tensor-Algebra bei, indem wir neue Eigenschaften und Berechnungsmethoden für die Tensor-Zerlegung ableiten.

Oftmals verfügen wir lediglich über eine elementweise oder ebenenweise Beschreibung des Signalmodells, welche nicht die explizite Tensorstruktur zeigt. Daher ist die Ableitung aller Tensor-Unfoldings nicht offensichtlich, wodurch die multidimensionale Struktur dieser Modelle nicht trivial nutzbar ist. Wir schlagen eine alternative Darstellung der elementweisen Multiplikation oder der ebenenweisen Multiplikation auf der Grundlage des generalisierten Tensor-Kontraktionsoperators vor. Weiterhin nutzen wir diese neuartige Darstellung und deren Eigenschaften zur Ableitung der letztendlichen Tensor-Modelle.

Es existieren eine Vielzahl von Tensor-Zerlegungen, die verschiedene Signalmodelle beschreiben, wie die HOSVD (Higher Order Singular Value Decomposition), CP/PARAFAC (Canonical Polyadic/PARallel FACtors) Zerlegung, die BTD (Block Term Decomposition), die PARATUCK2- (PARAFac und TUCKer2) und die PARAFAC2-Zerlegung (PARAllel FACtors2). Dabei ist die CP-Zerlegung am weitesten verbreitet und wird findet in zahlreichen Gebieten Anwendung. Daher ist die Entwicklung von Algorithmen zur effizienten Berechnung der CP-Zerlegung von besonderer Bedeutung. Das SECSI (Semi-Algebraic Framework for approximate CP decomposition via SIMultaneous matrix diagonalization) Framework ist ein effizientes und robustes Werkzeug zur Berechnung der approximierten Low-Rank CP-Zerlegung durch simultane Matrixdiagonalisierung. In dieser Arbeit stellen wir fünf Erweiterungen des SECSI-Frameworks vor, welche die Rechenkomplexität des ursprünglichen Frameworks reduzieren bzw. Einschränkungen für die Faktormatrizen einführen. Darüber hinaus werden die PARAFAC2- und die PARATUCK2-Zerlegung in der Regel mit einer ebenenweisen Notation beschrieben, die sich in Form der allgemeinen Tensor-Kontraktion, wie sie in dieser Arbeit vorgeschlagen wird, ausdrücken lässt. Wir nutzen diese neuartige Darstellung, um explizite Tensormodelle für diese beiden Zerlegungen abzuleiten. Darüber hinaus verwenden wir das PARAFAC2-Modell, um einen ALS-Algorithmus (Alternating Least-Squares) für die Berechnung der PARAFAC2-Zerlegungen abzuleiten.

Weiterhin nutzen wir die neuartigen Kontraktionseigenschaften für elementweise und ebenenweise Multiplikationen, um MIMO Multi-Carrier-Mobilfunksysteme zu modellieren. Wir zeigen, dass dieses sehr allgemeine Modell verwendet werden kann, um das Tensor-Modell des empfangenen Signals für MIMO-OFDM- (Multiple- Input Multiple-Output - Orthogonal Frequency Division Multiplexing), Khatri-Rao codierte MIMO-OFDM- und zufällig codierte MIMO-OFDM-Systeme abzuleiten. Wir schlagen die Übertragungstechniken der Khatri-Rao-Kodierung und zufällige Kodierung vor, um eine zusätzliche Tensor-Struktur des Sendesignal-Tensors einzuführen, welcher gewöhnlich keine bestimmte Struktur aufweist. Darüber hinaus zeigen wir, dass dieses Modell auf andere Multi-Carrier-Techniken wie GFDM (Generalized Frequency Division Multiplexing) erweitert werden kann. Unter Verwendung dieser Modelle auf der Empfängerseite entwerfen wir verschiedene Typen von Empfängern für diese Systeme, die die traditionellen matrixbasierten Lösungen in Bezug auf die Symbolfehlerrate übertreffen.

Im letzten Teil dieser Arbeit zeigen wir die Vorteile der Verwendung von Tensor-Algebra in der biomedizinischen Signalverarbeitung durch die gemeinsame Zerlegung von EEG- (ElectroEncephaloGraphy) und MEG- (MagnetoEncephaloGraphy) Signalen. Diese werden in der Regel gleichzeitig erfasst, wobei sie gemeinsame Aspekte derselben Gehirnaktivität beschreiben. Daher können EEG- und MEG-Signale mit gekoppelten Tensor-Zerlegungen wie der gekoppelten CP-Zerlegung analysiert werden. Wir nutzen das vorgeschlagene gekoppelte SECSI-Framework (eine der vorgeschlagenen Erweiterungen des SECSI-Frameworks) für die Berechnung der gekoppelten CP-Zerlegung, um zunächst den photic driving effect zu validieren und zu analysieren. Darüber hinaus validieren wir die Auswirkungen von Schädeldefekten auf die Messsignale von EEG und MEG durch eine gemeinsame EEG-MEG-Zerlegung mit dem gekoppelten SECSI-Framework. Beide Anwendungen zeigen, dass wir von gekoppelten Tensor-Zerlegungen profitieren, wobei die Methoden des gekoppelten SECSI-Frameworks erfolgreich zur Analyse biomedizinischer Daten genutzt werden können.

Contents

Acknowledgements

Abstract **ii**

Zusammenfassung **v**

Contents **vii**

List of Tables **xi**

List of Figures **xiii**

List of Algorithms **xix**

1. Introduction and scope of the thesis **1**

1.1. Motivation and state of the art 1

1.2. Major contributions 4

2. Tensor Algebra **11**

2.1. Fundamental Concepts of Tensor Algebra 11

2.1.1. Fundamental Concepts and Definitions 12

2.1.2. Properties of Kronecker, Khatri-Rao, and Hadamard Products 19

2.1.3. Least-Squares Kronecker and Khatri-Rao factorizations 21

2.1.4. New Contraction Properties for Element-wise and Slice-wise Multiplication . 23

2.2. Tensor Decompositions 27

2.2.1. Higher-Order Singular Value Decomposition 28

2.2.2. Canonical Polyadic Decomposition 32

2.2.3. Block Term Decomposition 39

2.2.4. PARATUCK2 42

2.2.5. PARAFAC2 44

2.3. Applications of Tensor Algebra 45

2.4. Conclusions 48

3. Computation of the CP decomposition and the coupled CP decomposition **49**

3.1. Introduction to the SECSI framework 50

3.2. Truncated Semi-Algebraic framework for the approximate CP decomposition via
Simultaneous matrix diagonalization (T-SECSI) 55

3.2.1. Simulation Results 56

3.2.2. Summary 59

3.3. Semi-Algebraic framework for the approximate CP decomposition via Non-Symmetric
Simultaneous matrix diagonalization (NS-SECSI) 61

3.3.1. Simulation Results 64

3.3.2. Summary 68

3.4.	Semi-Algebraic framework for the approximate CP decomposition via Simultaneous matrix diagonalization for Symmetric Tensors (S-SECSI)	70
3.4.1.	Simulation Results	74
3.4.2.	Summary	78
3.5.	Semi-Algebraic framework for the approximate CP decomposition via Simultaneous matrix diagonalization for Non-Negative Tensors (SECSI ₊)	79
3.5.1.	Simulation Results	83
3.5.2.	Summary	84
3.6.	Semi-Algebraic framework for the approximate Coupled CP decomposition via Simultaneous matrix diagonalization (C-SECSI)	85
3.6.1.	Simulation Results	93
3.6.2.	Summary	102
3.7.	Conclusions	103
4.	PARATUCK2 and PARAFAC2 via constrained CP model	105
4.1.	PARATUCK2	105
4.1.1.	Summary	110
4.2.	PARAFAC2	110
4.2.1.	Simulation Results	115
4.2.2.	Summary	120
4.3.	Conclusions	120
5.	Application of Tensor Algebra to Wireless Communication Systems	121
5.1.	Orthogonal Frequency Division Multiplexing (OFDM)	121
5.1.1.	MIMO-OFDM via generalized tensor contraction	123
5.1.1.1.	Simulation Results	131
5.1.1.2.	Summary	133
5.1.2.	Khatri-Rao Coded MIMO-OFDM via generalized tensor contraction	134
5.1.2.1.	Simulation Results	139
5.1.2.2.	Summary	144
5.1.3.	Randomly Coded MIMO-OFDM via generalized tensor contraction	144
5.1.3.1.	Simulation Results	148
5.1.3.2.	Summary	151
5.2.	Generalized Frequency Division Multiplexing (GFDM)	151
5.2.1.	Simulation Results	158
5.2.2.	Summary	159
5.3.	Conclusions	160
6.	Application of Tensor Algebra to Biomedical Signal Processing	163
6.1.	Validation of the photic driving effect based on the coupled CP decomposition	164
6.1.1.	Measured EEG-MEG Signals	165
6.1.2.	Joint EEG-MEG Signal Decomposition	166
6.1.3.	Analysis	167
6.1.4.	Results	168
6.1.5.	Discussion and Summary	174
6.2.	Validation of a controlled experiment based on Joint EEG-MEG signal decomposition using the coupled SECSI framework	177
6.2.1.	Measured EEG-MEG Signals	177
6.2.2.	Joint EEG-MEG Signal Decomposition	178

6.2.3. Results	179
6.2.4. Discussion and Summary	184
6.3. Conclusions	185
7. Conclusions and Future Work	187
Appendix A. Acronyms, Symbols, and Notation	195
A.1. Acronyms	195
A.2. Symbols and Notation	199
Appendix B. Proofs and derivations	203
B.1. Derivation of the explicit tensor structure of a diagonalized tensor	203
B.2. Structure of the selection matrix	205
B.3. Derivation of the 4-way channel tensor in the frequency domain and its unfoldings .	206

List of Tables

2.1. Link between the diagonalized tensor structures and their generalized unfoldings. . . 26

3.1. Estimates of the factor matrices obtained from the diagonalization along the first mode. 90

5.1. Computational time required for each algorithm. 132

5.2. Parameters for Figs. 5.12, 5.13, and 5.14 142

6.1. Individual alpha frequency f_α in Hz for each volunteer. 165

6.2. Stimulation frequencies in f_α rations for the particular conditions. 165

6.3. Residual for the MEG and EEG signal tensors and reliability of the coupled CP decomposition. 179

List of Figures

1.1. Structure of the thesis.	7
2.1. 1-mode, 2-mode, and 3-mode unfoldings of a 3-way tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ using forward ordering of the n -mode vectors.	13
2.2. 1-mode, 2-mode, and 3-mode unfoldings of a 3-way tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ using reverse cyclical ordering of the n -mode vectors.	14
2.3. Visualization of the generalized unfolding $[\mathcal{A}]_{([1,2],[3,4])}$	14
2.4. Kronecker product between a tensor $\mathcal{A} \in \mathbb{C}^{2 \times 2 \times 2}$ and a matrix $\mathbf{B} \in \mathbb{C}^{2 \times 2}$ in terms of the scalar elements.	17
2.5. Generalized tensor contraction between the tensors $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$ and $\mathcal{B} \in \mathbb{C}^{M \times N \times K}$	18
2.6. Tensor concatenation along the 1-mode, the 2-mode, and the 3-mode.	18
2.7. A 3-way identity tensor $\mathcal{I}_{3,3}$ with dimensions $3 \times 3 \times 3$	19
2.8. A visualization of the diagonal structure of the tensors $\mathcal{D}^{(A)} = \mathcal{I}_{3,M} \times_3 \mathbf{A}^T \in \mathbb{C}^{M \times M \times N}$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B} \in \mathbb{C}^{M \times N \times N}$, for $M = 4$ and $N = 3$	25
2.9. A slice-wise multiplication between two tensors $\mathcal{A} \in \mathbb{C}^{M \times N \times K}$ and $\mathcal{B} \in \mathbb{C}^{N \times J \times K}$, $\mathcal{T}_{1(.,.,k)} = \mathcal{A}_{(.,.,k)} \mathcal{B}_{(.,.,k)}$, $\forall k = 1, \dots, K$	25
2.10. A visualization of the block diagonal structure of the unfoldings for the tensors $\mathcal{D}^{(A)} = \mathcal{I}_{3,M} \times_3 \mathbf{A}^T \in \mathbb{C}^{M \times M \times N}$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B} \in \mathbb{C}^{M \times N \times N}$, for $M = 4$ and $N = 3$	26
2.11. HOSVD of a 3-way tensor $\mathcal{X} = \mathcal{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$	29
2.12. CP decomposition of a 3-way tensor with rank $R = 3$	33
2.13. Generalized unfoldings of a 4-way tensor with a CP structure.	34
2.14. BTD in rank- (L, M, N) terms	40
2.15. Block diagonal structure of the BTD in rank- (L, M, N) terms	41
2.16. Slice-wise visualization of the PARATUCK2 decomposition for a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ with slices $\mathcal{X}_{(.,.,k)} = \mathbf{A} \cdot \mathcal{D}^{(A)}_{(.,.,k)} \cdot \mathbf{R} \cdot \mathcal{D}^{(B)}_{(.,.,k)} \cdot \mathbf{B}^T$	43
2.17. Slice-wise visualization of the PARAFAC2 decomposition for a tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ with slices $\mathbf{X}_k = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{B}_k^T$, $\forall k = 1, \dots, K$	45
3.1. Diagonalization of the tensor $\mathcal{S}_3^{[s]} \in \mathbb{C}^{R \times R \times M_3}$ along the 3-mode.	51
3.2. The SECSI framework for the computation of the CP decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ with rank R	54
3.3. CCDF of the TSFE for a real-valued tensor with dimensions $40 \times 40 \times 40$, tensor rank $R = 3$, and SNR = 20 dB.	57

3.4. CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 20 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient 0.9.	58
3.5. TMSFE as a function of the SNR for a real-valued tensor with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$	59
3.6. TMSFE as a function of the SNR for a complex-valued tensor with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns.	60
3.7. The NS-SECSI framework for the computation of the approximate CP decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$	62
3.8. CCDF of the TSFE for a real-valued tensor with dimensions $40 \times 40 \times 40$, tensor rank $R = 3$, and SNR = 25 dB.	64
3.9. CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 25 dB.	65
3.10. CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 10 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.	66
3.11. Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $30 \times 30 \times 30$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.	66
3.12. CCDF of the TSFE for a complex-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 15 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.	67
3.13. Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.	68
3.14. CCDF of the TSFE for a real-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 20 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.	69
3.15. The S-SECSI framework for the computation of the approximate CP decomposition of a symmetric tensor $\mathcal{X} \in \mathbb{C}^{M \times M \times M_3}$. The symmetry is with respect to the 1-mode and the 2-mode.	73
3.16. CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode, i.e., $\mathbf{F}_1 = \mathbf{F}_2$ from equation (3.1).	74
3.17. CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient 0.9.	75

3.18. Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $30 \times 30 \times 30$ and tensor rank $R = 3$. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.	75
3.19. CCDF of the TSFE for a real-valued tensor with dimensions $7 \times 7 \times 4$, tensor rank $R = 3$, and SNR = 35 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns.	76
3.20. CCDF of the TSFE for a complex-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.	77
3.21. Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.	77
3.22. CCDF of the TSFE for a real-valued tensor with dimensions $20 \times 20 \times 20$, tensor rank $R = 3$, and SNR = 10 dB.	84
3.23. Diagonalization of the tensors $\mathcal{S}_3^{(1)} \in \mathbb{C}^{R \times R \times M_3^{(1)}}$ and $\mathcal{S}_3^{(2)} \in \mathbb{C}^{R \times R \times M_3^{(2)}}$ along the 3-mode.	89
3.24. The C-SECSI framework for the computation of the coupled CP decomposition of two tensors $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$ that have the 1-mode in common.	91
3.25. Reliability as a function of the assumed rank \hat{R} for different SNRs.	93
3.26. Reliability as a function of the assumed rank \hat{R} for different ranks.	94
3.27. CCDF of the TSFE for real-valued tensors with dimensions $40 \times 4 \times 10$, tensor ranks $R_1 = R_2 = 3$, factor matrices with mutually correlated columns designed as sine functions, and SNR ₁ = SNR ₂ = 25 dB.	95
3.28. CCDF of the TSFE for real-valued tensors with dimensions $80 \times 80 \times 80$, tensor ranks $R_1 = R_2 = 4$, and SNR ₁ = SNR ₂ = 30 dB. The first (common) factor matrix has mutually correlated columns with a correlation coefficient $\rho_1 = 0.98$.	95
3.29. CCDF of the TSFE for complex-valued tensors with dimensions $4 \times 8 \times 7$, tensor rank $R_1 = R_2 = 3$, and SNR ₁ = SNR ₂ = 45 dB. The first (common) factor matrix has mutually correlated columns.	96
3.30. CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 8 \times 4$, tensor ranks $R_1 = R_2 = 3$, and SNR ₁ = SNR ₂ = 45 dB. The third factor matrices have mutually correlated columns.	97
3.31. CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 3 \times 4$, tensor ranks $R_1 = R_2 = 4$, and SNR ₁ = SNR ₂ = 30 dB. Both tensors are degenerate with respect to the 2-modes, i.e., $R_1 > M_2^{(1)}$ and $R_2 > M_2^{(2)}$.	98
3.32. CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 7 \times 7$, tensor ranks $R_1 = R_2 = 3$, SNR ₁ = SNR ₂ = 30 dB, and the factor matrices have mutually correlated columns with correlation coefficients $\rho_1 = 0.1$, $\rho_2 = 0.9$, and $\rho_3 = 0.1$.	99

3.33. TMSFE as a function of the SNR for complex-valued tensors with dimensions $4 \times 8 \times 7$, tensor ranks $R_1 = R_2 = 3$, where the second tensor has third factor matrix with mutually correlated columns with correlation coefficient of 0.98.	99
3.34. Probability of selecting the final estimate for the C-SECSI framework using the REC PS heuristic. The decomposed complex-valued tensors have dimensions $4 \times 7 \times 8$, common 1-mode, ranks $R_1 = R_2 = 3$, $\text{SNR}_1 = \text{SNR}_2 = 30$ dB, and correlated factor matrices with correlation coefficients ρ_1, ρ_2 , and ρ_3	100
3.35. TMSFE as a function of the SNR_2 for complex-valued tensors with dimensions $3 \times 8 \times 7$, tensor ranks $R_1 = R_2 = 3$, and $\text{SNR}_1 = 30$ dB.	101
4.1. Slice-wise visualization of the PARATUCK2 decomposition for a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ with slices $\mathcal{X}_{(:, :, k)} = \mathbf{A} \cdot \mathcal{D}^{(A)}_{(:, :, k)} \cdot \mathbf{R} \cdot \mathcal{D}^{(B)}_{(:, :, k)} \cdot \mathbf{B}^T$	106
4.2. Slice-wise visualization of the PARAFAC2 decomposition for a tensor $\mathcal{X}' \in \mathbb{R}^{I \times J \times K}$ with slices $\mathbf{X}'_k = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k, :)}) \cdot \mathbf{B}_k^T, \quad \forall k = 1, \dots, K$	111
4.3. MSRE as a function of the $\text{SNR} = 0, \dots, 40$ dB for a real-valued tensor with dimensions $8 \times 10 \times 12$, and 3 components. The results are averaged over 2000 realizations. All algorithms are initialized with the same initial factor matrices.	116
4.4. CCDF of the SRE for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and $\text{SNR} = 20$ dB. All algorithms are initialized with the same initial factor matrices. The vertical lines correspond to the mean values also depicted in Fig. 4.3.	117
4.5. CCDF of the number of iterations for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and $\text{SNR} = 20$ dB. All algorithms are initialized with the same initial factor matrices. The vertical lines represent the average number of iterations equal to 122, 137, and 157 for “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD”, respectively.	117
4.6. CCDF of the SRE for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and $\text{SNR} = 30$ dB. The matrix \mathbf{C} has collinear columns with a correlation coefficient 0.8. All algorithms are initialized with the same initial matrices. The vertical lines represent the mean values.	118
4.7. CCDF of the number of iterations for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and $\text{SNR} = 30$ dB. The matrix \mathbf{C} has mutually correlated columns with a correlation coefficient of 0.8. All algorithms are initialized with the same initial matrices. The vertical lines represent the average number of iterations equal to 278, 466, and 95 for “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD”, respectively.	119
5.1. A MIMO system with $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas.	124
5.2. Visualization of the channel tensors $\tilde{\mathcal{H}}_R^{(1)}$ and $\tilde{\mathcal{H}}_R^{(2)}$ as well as the channel matrices $\tilde{\mathbf{H}}_R^{(1)}$ and $\tilde{\mathbf{H}}_R^{(2)}$ for a MIMO system with $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas. The MIMO system is depicted in Fig. 5.1.	124
5.3. Visualization of the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$ for a MIMO-OFDM system with parameters $M_T = 2$, $M_R = 3$, and $N = 3$	125

5.4. Visualization of the arrangement of the pilot symbols and the data symbols for the first antenna with respect to the frame number and the subcarrier number for a MIMO-OFDM system with parameters $M_T = 2$, $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$.	127
5.5. Visualization of the arrangement of the pilot symbols and the data symbols in the first frame with respect to the antenna number and the subcarrier number for a MIMO-OFDM system with parameters $M_T = 2$, $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$.	127
5.6. SER for a 2×2 OFDM system with parameters $\Delta K = 2$, $\Delta F = 3$, $N = 128$, $K = 2$.	132
5.7. SER for a 2×2 OFDM system with parameters $\Delta K = 8$, $\Delta F = 3$, $N = 128$, $K = 8$.	133
5.8. SER for an OFDM system with parameters $\Delta K = 2$, $\Delta F = 4$, $N = 512$, $K = 2$, and $M_T \times M_R$ antennas depicted in the legend.	133
5.9. Visualization of the Khatri-Rao coding.	136
5.10. SER comparison for different numbers of transmitted blocks.	140
5.11. SER comparison for different numbers of transmit and receive antennas.	140
5.12. NMSE of the channel matrix for different scenarios after the LSKRF.	141
5.13. NMSE of the symbol matrix for different scenarios after the LSKRF.	141
5.14. Channel estimation error for different scenarios.	142
5.15. SER for 2×2 OFDM and KR coded OFDM systems, $N = 128$ and $\Delta F = 10$. The OFDM system has the following parameters $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. The KR coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 16-QAM. Hence, both systems transmit 2 Bits/Symbol.	143
5.16. SER for a 2×2 randomly coded OFDM system with parameters $N = 128$, $Q = 2$, K , ΔK , ΔF , and the symbols are modulated using 4-QAM. The parameters K , ΔK and ΔF are indicated in the legend.	149
5.17. SER for 2×2 KR coded OFDM, randomly coded OFDM, and traditional OFDM systems for $N = 128$ and $\Delta F = 10$. The KR coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 16-QAM. The randomly coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 4-QAM. The OFDM system has the following parameters $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. Therefore, all systems transmit 2 Bits/Symbol.	150
5.18. Visualization of the matrix $[\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot [\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T)]$ for a MIMO-OFDM system with parameters $M_T = 2$, $M_R = 3$, and $N = 3$.	156
5.19. Comparison of the SER for different pilot positions spacings in the frequency domain.	159
5.20. Comparison of the SER for different modulation order.	159
6.1. Visualization of the EEG and MEG tensors per volunteer and stimulation frequency	166
6.2. RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $1.1f_\alpha$.	169

6.3. RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $0.55f_\alpha$	169
6.4. RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $0.7f_\alpha$	170
6.5. Normalized frequencies obtained from the analysis, weights, reliability, and residual corresponding to the EEG signals.	171
6.6. Normalized frequencies obtained from the analysis, weights, reliability, and residual corresponding to the MEG signals.	172
6.7. Scatter plot of the normalized frequencies obtained from the analysis for EEG, violin plots of the weights of the recruited and not-recruited components, reliability, and residual as a function of the normalized stimulation frequency.	173
6.8. Scatter plot of the normalized frequencies obtained from the analysis for MEG, violin plots of the weights of the recruited and not-recruited components, reliability, and residual as a function of the normalized stimulation frequency.	174
6.9. Visualization of the experimental setup for the skull defect study in [LFH14].	178
6.10. Components of the MEG signal tensor without a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.	180
6.11. Components of the MEG signal tensor with a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.	180
6.12. Components of the EEG signal tensor without a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.	181
6.13. Components of the EEG signal tensor with a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.	181
6.14. Measured MEG signals above two skull defects (row 1) and components of rank 3 coupled decomposition (rows 2-4) shown at selected source positions (columns) at the first peak in the time dimension. The iso-line increment is uniform within measurements and within components, respectively. The components are arranged in a meaningful sequence. The dipolar source is indicated with a black bar with two spheres marking the poles. Skull defects are marked by closed black lines indicating the inner, middle and outer boundaries of the defects.	182

6.15.	Measured EEG signals above two skull defects (row 1) and components of rank 3 coupled decomposition (rows 2-4) shown at selected source positions (columns) at the first peak in the time dimension. Formatting and markings equivalent to Fig. 6.14.	183
B.1.	Visualization of the structure of the matrices \mathbf{C} and $(\mathbf{e}_k \otimes \mathbf{C}_{(k,.)})$ and the tensor $\mathcal{I}_{3,R \times 3}(\mathbf{e}_k \otimes \mathbf{C}_{(k,.)})$, for $K = 3$, $R = 2$, and $k = 1, \dots, K$.	204
B.2.	Visualization of the structure of the matrix $(\mathbf{I}_K \diamond \mathbf{C}^T)^T$, for $K = 3$ and $R = 2$.	204
B.3.	Visualization of the structure of the tensor $\mathcal{T}_D \in \mathbb{C}^{J \times K \times K}$ in (B.3), for $J = 4$, $K = 3$, and $R = 2$.	205

List of Algorithms

2.1. Least-Squares Kronecker Factorization (LSKF)	22
2.2. Least-Squares Khatri-Rao Factorization (LSKRF)	23
4.1. PARAFAC2 via ALS (“P2-ALS”)	115
5.1. ZF receiver	128
5.2. Iterative Least-Squares with Projection (ILSP)	129
5.3. Iterative Least-Squares with Enumeration (ILSE)	129
5.4. Recursive Least-Squares with Projection (RLSP)	130
5.5. Recursive Least-Squares with Enumeration (RLSE)	131
5.6. Khatri-Rao (KR) receiver	138
5.7. Khatri-Rao receiver and its enhancement via Least-Squares (KR+LS)	139
5.8. Random Coding-Khatri-Rao (RC-KR) receiver	147
5.9. Random Coding-Khatri-Rao + ALS (RC-KR+ALS) receiver	148
5.10. Alternating Least-Squares (ALS) receiver	158

Chapter 1

Introduction and scope of the thesis

1.1 Motivation and state of the art

Tensors have been introduced for the first time at the end of the 19-th century and have been mainly used in differential calculus and physics [Com14]. At that time, the concept of tensors was not widely utilized because of the abstract tensor definition that tensors represent a mapping from one linear space to another, whose coordinates transform multi-linearly under a change of bases [Com14]. Later, this definition was simplified by considering multi-dimensional coordinate systems. With respect to a multi-dimensional coordinate system, a tensor is a multi-dimensional array of numbers whose dimensions correspond to the dimensions of the coordinate system [Com14]. Moreover, the concept of tensor decompositions has been introduced at the beginning of the 20-th century [KB09]. A tensor decomposition decomposes a given tensor into its elementary underlying components. Depending on the structure and the properties of these underlying components, different tensor decompositions have been defined including the Tucker decomposition [Tuc63], the CP/PARAFAC (Canonical Polyadic/PARallel FACtors) decomposition [CC70],[Kru77], the PARAFAC2 (PARallel FACtors2) decomposition [Har72], and the DEDICOM (DEcomposition into DIrectional COMponents) decomposition [Har78]. These tensor decompositions have been introduced to model and solve problems in psychometrics. Thus, the first application area of tensors and tensor decompositions is psychometrics. Since then, the number of tensor decompositions and the number of tensor applications have constantly been increased. Today, tensor algebra has application in thermometrics, statistics, image processing, signal processing for wireless communications, biomedical signal processing, data analytics, machine learning, and many more.

The observed signals in the above mentioned applications are multi-dimensional. For instance, in wireless communications the dimensions of the observed/measured signals correspond to time, frequency, antennas, and users. Moreover, in biomedical signal processing the dimensions of the measured signals correspond to time, space (channels), modality (electroencephalography, magnetoencephalography, electrocardiography), participant (volunteer), and experimental condition. Certainly, it is possible to arrange these observations in a matrix and to use matrix methods to analyze them. However, using this approach, we lose the interconnecting information that exists between the different dimensions. On the other side, tensors preserve and exploit the multi-dimensional structure while even providing an improved identifiability. In addition, the different tensor decompositions open different and flexible ways to model the observed signals.

The tensor decompositions that have been proposed in the more recent past include the HOSVD (Higher-Order Singular Value Decomposition) [DLDMV00b], the BTD (Block Term Decomposition) [DL08a,DL08b,DLN08], the NTF (Non-negative Tensor Factorization i.e., CP decomposition with non-negativity constraints) [CZPSI09], the HOGSVD (Higher-Order Generalized Singular Value Decomposition) [Pon10, PSvA11], and the CONFAC (CONstrained FACtor) decomposition [dAFM08,FdA14b]. Note that the different tensor decompositions are unique up to a permutation and scaling ambiguity under different conditions that should be derived for each decomposition separately. For instance, a sufficient condition for the uniqueness of the CP decomposition for 3-way tensors is provided in [Kru77,Kru89]. Moreover, the authors of [Com14,SDLF⁺17] provide some new results on the uniqueness of the CP decomposition. Note that we still study the theoretical aspects of tensor algebra especially the uniqueness properties of the tensor decompositions.

Very often a tensor is decomposed into the minimum number of rank one components using the CP decomposition. The CP decomposition is typically calculated via an iterative multi-linear ALS (Alternating Least Square) algorithm [CC70,KB09]. ALS based algorithms require a lot of iterations to calculate the CP decomposition and they have no convergence guarantee. Moreover, ALS based algorithms are less accurate in ill-conditioned scenarios, for instance, if the columns of the factor matrices are highly correlated. There are also ALS based algorithms for calculating the CP decomposition such as the ones presented in [BSG99] and [RCH08] that either introduce constraints to reduce the number of iterations or are based on line search, respectively. Alternatively, semi-algebraic solutions have been proposed in the literature based on SMDs (Simultaneous Matrix Diagonalizations). Such examples include [DL05], [RH08], [LA11], [RH13a], [LA14], and [RSH12]. In contrast to the remaining SMD based algorithms, the SECSI (Semi-Algebraic framework for the approximate CP decomposition) framework [RH08, RH13a, RSH12] calculates all possible SMDs and then selects the best available solution in a final step via appropriate heuristics. Therefore, it offers an efficient and robust computation of the approximate low-rank CP decomposition with complexity-accuracy trade-off. However, especially for tensors with large dimensions the SECSI framework might be computationally too expensive. Note that a closed-form solution for the computation of the CP decomposition exists only for some special cases such as the computation of the CP decomposition of a tensor with two slices and a tensor with rank two [RH13a]. Moreover, different applications impose different constraints on the CP decomposition. Therefore, the existing algorithms for the computation of the CP decomposition should be modified. For instance, non-negativity constraints on the factor matrices are considered in blind estimation applications and biomedical signal processing [CZPSI09]. Algorithms for the computation of the CP decomposition with non-negativity constraints are proposed in [LS15,AALM16] and [CFC15], based on the ADMM (Alternating Direction Method of Multipliers) and ALS, respectively. Note that the SECSI framework proposed in [RH08,RH13a] considers only real-valued and symmetry constraints on the factor matrices, but not non-negativity or sparsity constraints. Moreover, several combined signal processing applications such as joint processing of biomedical signals [BCA12,RDGD⁺15,ABS15] and array signal processing [SDDL18,SDL17a] benefit from coupled tensor decompositions. Therefore, for these applications, we require a coupled CP decomposition and algorithms for its computation

as proposed in [SDDL15, VDS⁺16]. However, these previous publications do not consider that the coupled tensors might be corrupted by noise with different variances. The authors of [FCC16] consider this, but they propose an ALS algorithm with normalization for which the noise variance should be known or estimated beforehand. Moreover, the ALS algorithm proposed in [FCC16] might lead to many iterations, and it is not robust in ill-posed scenarios for example, if the factor matrices contain collinear columns.

In contrast to the CP decomposition, the PARATUCK2 decomposition [HL96] and the PARAFAC2 decomposition [Har72] are more flexible tensor decompositions. Both decompositions are generalization of the CP decomposition, but the PARATUCK2 decomposition offers more flexibility in terms of the number of sets of underlying components. On the other hand, the PARAFAC2 decomposition allows the underlying components to vary along one of the tensor dimensions. Even though these decompositions are more flexible than CP, they are not as often used as the CP decomposition. However, lately there is an increased interest in these decompositions in various applications such as biomedical signal processing and data analytics [WJG⁺10, FdA14a, XF-dAS14, CHGH18, CSH18]. The smaller number of applications of PARATUCK2 and PARAFAC2 as compared to the CP decomposition might be due to the fact that there exist only some results on the uniqueness of these decompositions [Har72, HL96, FdA14b]. Moreover, in contrast to the many algorithms for the computation of the CP decomposition, there exist only few algorithms for the computation of the PARATUCK2 decomposition and the PARAFAC2 decomposition [Kie93, HL96, Bro98, KTBB99, Wei15]. The algorithms for the computation of the PARAFAC2 decomposition can be divided into two groups an indirect approach and a direct approach. The indirect approach fits the cross product of the slices of the tensor instead of the tensor slices in an ALS fashion [Kie93]. On the other hand, the direct fitting approach fits the tensor slices directly in an ALS fashion, thereby using two loops that lead to a significant number of iterations [Bro98, KTBB99, Wei15]. Furthermore, both decompositions PARATUCK2 and PARAFAC2 are defined using a slice-wise notation. Hence, there are no explicit tensor models that define these decompositions.

Tensor and tensor decompositions are very practical tools for modeling wireless multi-carrier systems such as OFDM (Orthogonal Frequency Division Multiplexing) systems [HYW⁺09]. The authors of [dAFX13] model a MIMO multi-carrier system using tensor algebra and the PARATUCK2 tensor decomposition resulting in a novel space, time, and frequency coding structure. Similarly, trilinear coding in space, time, and frequency is proposed for MIMO-OFDM systems based on the CP tensor decomposition in [dAF13b]. By exploiting tensor models, semi-blind receivers have been introduced for multi-carrier communications systems in [FdA14a] and [LdCSdA13]. However, all these publications use additional spreading that leads to a significantly reduced spectral efficiency to create the tensor structure. Moreover, previous publications on tensor models for multi-carrier communications systems [dAFX13], [dAF13b], [LdCSdA13], and [FdA14a] do not exploit the channel correlation between the adjacent subcarriers. Furthermore, all these publications rely on the subcarrier-wise description of the MIMO-OFDM system. Therefore, these models do

not reveal the explicit tensor structure of the received signal in MIMO-OFDM systems. Note that the OFDM is the most widely used multi-carrier technique in the many current wireless communication systems [HYW⁺09]. However, several new multi-carrier transmission techniques are considered as an alternative to OFDM for beyond 5G wireless communication systems. These multi-carrier transmission techniques include FBMC (Filter Bank Multi-Carrier) [PNCZ⁺16,FB11], UPMC (Universal Filtered Multi-Carrier) [SWC14], and GFDM (Generalized Frequency Division Multiplexing) [MDK⁺17,MMG⁺14]. In general, the tensor models corresponding to MIMO-OFDM can be extended to any of the multi-carrier techniques such as MIMO-FBMC, MIMO-UPMC, and MIMO-GFDM. In spite of this fact, only the authors of [KCdA17] propose a tensor based model for MIMO-FBMC systems. However, the authors of [KCdA17] derive the proposed semi-blind receiver using only slice-wise notation. An explicit tensor model is not proposed. Hence, we require a general tensor model for MIMO multi-carrier systems that can be extended to any multi-carrier technique. Moreover, the tensor structure of the FBMC, UPMC, and GFDM signals remain to be explored.

As previously mentioned, the measured signals in biomedical signal processing are multi-dimensional. In clinical studies, these signals are usually obtained using EEG (ElectroEncephaloGraphy) and MEG (MagnetoEncephaloGraphy). Therefore, we can analyze these signals using tensor decompositions as shown in [CLK⁺15,BAC⁺14,HCS⁺14]. Moreover, an identification of the signal components in EEG data based on the PARAFAC2 decomposition is performed in [WJG⁺10,WJR⁺10]. The authors of [CHGH18] exploit also the PARAFAC2 decomposition for the analysis of somatosensory evoked magnetic fields and somatosensory evoked electrical potentials. Typically, the EEG and MEG signals used in biomedical studies are simultaneously acquired. Thus, these signals simultaneously capture aspects of the same electric activity and therefore can be jointly analyzed. For instance, a joint EEG-MEG signal analysis can be performed using a coupled CP decomposition [BCA12]. However, the authors of [BCA12] use simulated signals not measurement signals. Similarly, the authors of [AKD11,ABS15] show that data fusion in metabolomics benefits from coupled matrix-tensor decompositions. Hence, the biomedical signal processing applications benefit from joint signal analysis using coupled tensor decompositions. Therefore, efficient algorithms for the computation of the coupled tensor decompositions have yet to be developed. Note that the coupling is only assumed, but not yet proven in all biomedical applications. This is because in most of the biomedical applications, we have only measured observations and it is not possible to generate simulated signals that confirm the coupling hypotheses. Therefore, the coupling of the EEG and MEG signals in different biomedical applications has yet to be proven and efficiently exploited.

1.2 Major contributions

Inspired by the previously presented state of the art, in this thesis we focus on the following objectives.

- *Efficient and robust computation of the CP decomposition and the coupled CP decomposition*

Efficient algorithms for the computation of the CP decomposition play an important role in all signal/data analysis applications. We propose five extensions of the SECSI framework for the computation of the CP decomposition and the coupled CP decomposition. These extensions include T-SECSI (Truncated SECSI) that reduces the computational complexity of the original SECSI framework, NS-SECSI (Non-Symmetric SECSI) that reduces the computational complexity of the original SECSI framework and computes the CP decomposition in a closed-form fashion, S-SECSI (Symmetric SECSI) that considers symmetry constraints on the factor matrices in a closed-form fashion, SECSI₊ (SECSI for non-negative tensors) that considers non-negativity constraints on the factor matrices, as well as C-SECSI (Coupled SECSI) for the computation of the coupled CP decomposition. In the future, it is possible to consider other constraints such as sparsity constraints on the factor matrices. Moreover, we use the proposed C-SECSI framework for the joint analysis of EEG and MEG signals and show its benefits in biomedical signal processing.

- *Contraction properties for element-wise multiplication and slice-wise multiplication*

In many tensor applications, we only have an element-wise description or a slice-wise description of the signal model. We propose to express these descriptions by using the generalized tensor contraction operator. In contrast to the element-wise and slice-wise multiplications, this novel representation leads to an explicit tensor model of the resulting tensor. The explicit tensor model reveals the complete tensor structure and allow us to derive all tensor unfoldings. Among the perspectives for a future work, we should study the properties of the Kronecker product between two tensors as it is a recurring structure when using the novel representation of the slice-wise multiplications. Moreover, we should study the uniqueness properties of the resulting models especially in the cases when the resulting tensor model is a constrained CP decomposition.

- *Derivation of explicit tensor models for the PARATUCK2 decomposition and the PARAFAC2 decomposition*

Using the new properties for the slice-wise multiplication based on the generalized tensor contraction, we derive explicit tensor models for the PARATUCK2 decomposition and the PARAFAC2 decomposition. By substituting the individual structure of the tensors involved in the contraction, we show that both decompositions satisfy constrained CP models. These models can be used later to further study the uniqueness properties of PARATUCK2 and PARAFAC2. Moreover, we can utilize these models for the derivation of new algorithms for the computation of the PARATUCK2 decomposition and the PARAFAC2 decomposition. In this thesis, we derive a single loop ALS algorithm for the computation of the PARAFAC2 decomposition.

- *Derivation of explicit tensor models for wireless multi-carrier MIMO communication systems*

It is evident that the typical description of multi-carrier systems using a subcarrier-wise

notation, i.e., using a slice-wise notation, can be expressed in terms of the generalized tensor contraction according to the new properties derived in this thesis. Hence, we exploit these properties to derive a general tensor model of the received signal in multi-carrier MIMO systems. We verify this model for MIMO-OFDM systems, such that we are able to derive the traditional zero forcing receiver as a benchmark in addition to new improved receivers. In this thesis, we also consider other transmission techniques for MIMO-OFDM that include Khatri-Rao coding and random coding. Using these techniques, we impose a CP structure on the transmitted signal. Note that the new general model for multi-carrier systems still holds even though the transmitted signal has a different structure compared to the uncoded case. Therefore, we exploit the same contraction based tensor model to design receivers for Khatri-Rao coded MIMO-OFDM and randomly coded MIMO-OFDM systems. Moreover, we show that the general tensor model can be used to model a MIMO-GFDM system and accordingly can be utilized for an efficient receiver design. In the future, we should consider MIMO-FBMC and MIMO-UFMC systems as well. Moreover, the general tensor model presented in this thesis can be easily extended to one-way and two-way relaying systems. Hence, we can use our tensor model to derive the explicit tensor model of the received signal in multi-carrier relaying systems.

The structure of the thesis is visualized in Fig. 1.1. The rest of this thesis is organized in six chapters. In Chapter 2, we review the fundamental concepts of tensor algebra, selected tensor decompositions, and some applications of tensor algebra. In the section devoted to fundamental concepts of tensor algebra, we review some basic definitions, basic properties of tensor algebra, the properties of the Kronecker product, the properties of the Khatri-Rao product, and the properties of the Hadamard product. Moreover, in this section, we review the least-squares Kronecker factorization of a matrix [VLP93, VLP97, dCFR18] and the least-squares Khatri-Rao factorization of a matrix [RH09b] that can factorize these products in a least-squares sense in the case of noise corrupted observations. Furthermore, we present our contributions to the theoretical aspects of tensor algebra that include an alternative representation of the element-wise multiplication and slice-wise multiplication between two arrays. We propose to represent these multiplications using the generalized tensor contraction operator. These novel properties are used later in this thesis as a fundamental first step. By substituting the individual tensor structure of the tensors involved in the contraction, we are able to derive an explicit tensor model of the overall tensor. Some of the aforementioned properties have been published in [NCdAH18]. In addition to the fundamentals of tensor algebra, we review some tensor decompositions including the HOSVD, the CP decomposition, the BTD, the PARATUCK2 decomposition, and the PARAFAC2 decomposition in Chapter 2. Finally, we list some of the many applications of tensor algebra with a special focus on applications in wireless communication systems and biomedical signal processing that are in the main scope of this thesis.

We devote Chapter 3 to the efficient computation of the CP decomposition and the coupled CP decomposition that is the basis of many signal/data analysis applications. To be more precise, we

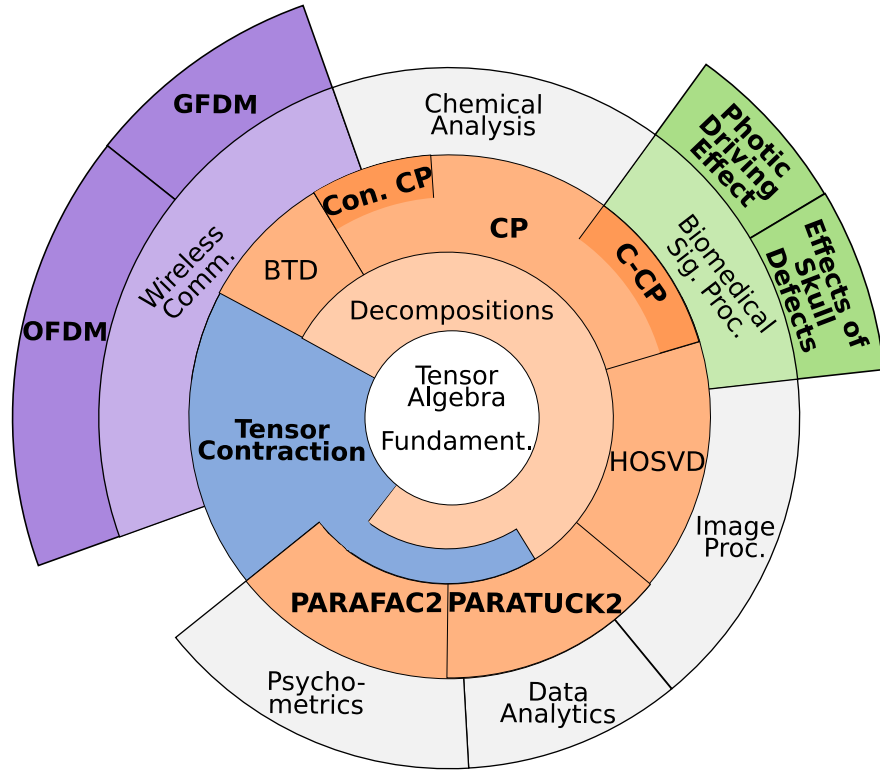


Figure 1.1.: Structure of the thesis.

propose extensions of the SECSI framework that reduce the computational complexity or introduce constraints to the factor matrices. T-SECSI has a reduced computational complexity than the original framework because it diagonalizes a core tensor of smaller dimensions. However, T-SECSI still diagonalizes the core tensor using symmetric SMDs and computes six initial estimates of the factor matrices for a 3-way tensor as the SECSI framework. Moreover, the NS-SECSI framework additionally reduces the computational complexity by considering a non-symmetric SMD instead of two symmetric SMDs. Hence, NS-SECSI computes only three initial estimates of the factor matrices and thereby reduces also the complexity of the search for the final solution. Moreover, considering a closed-form solution of the non-symmetric SMDs, we propose an NS-SECSI framework that approximates the CP decomposition in a novel closed-form fashion, for the first time. Recall that previous publications propose a closed-form solution of the CP decomposition only for special cases that include tensors with rank two and tensors with two slices [RH13a]. We denote this framework by NS-SECSI-NS-IDIEM, where NS-IDIEM (Non-Symmetric Improved DIagonalization using Equivalent Matrices) is an extended version of the IDIEM (Improved DIagonalization using Equivalent Matrices) algorithm [CB12], [CKM⁺14] that provides a closed-form solution for the non-symmetric SMD problem. We also propose a closed-form solution of the CP decomposition for symmetric tensors. We denote this framework by S-SECSI. Furthermore, by considering non-negativity constraints on the factor matrices we derive the SECSI₊ framework for the computation of the CP decomposition for nonnegative tensors. Finally, for coupled tensors (tensors that have at least one mode in common) we propose C-SECSI for the efficient and robust computation of

the coupled CP decomposition. The NS-SECSI framework and the C-SECSI framework have been published in [NHT⁺16] and [NH16], respectively.

In Chapter 4, we express the slice-wise multiplication between two tensors via the new properties of the contraction operator to derive explicit tensor models for the PARATUCK2 decomposition and the PARAFAC2 decomposition. First, we show that these two decompositions represent a slice-wise multiplication between two tensors. Next, by substituting the individual tensor structure of the tensors involved in the contraction, we derive novel explicit tensor models of the overall tensor. The resulting tensor models correspond to constrained CP models, and they do not depend on a chosen unfolding and capture all of the unfoldings at the same time. Using the constrained CP model of the PARAFAC2 decomposition, we propose a single loop ALS algorithm for the computation of the PARAFAC2 decomposition. In contrast to the state-of-the-art algorithms, the proposed algorithm requires fewer iterations. This algorithm has been published in [NCdAH18].

We devote Chapter 5 to tensor based signal processing for wireless communication systems. In this chapter, we express the received signal for MIMO-OFDM systems that traditionally has only a subcarrier-wise (slice-wise) description via the generalized tensor contraction. The proposed tensor model represents a contraction between a channel tensor and a transmit signal tensor, and it provides a new, compact, and flexible formulation of the MIMO-OFDM system. The resulting tensor model is then derived by substituting the structure of the channel tensor and the transmit signal tensor. Moreover, this model represents a very general description that can be extended to any transmission technique for MIMO-OFDM. For instance, in this chapter we present the extension of this model to the Khatri-Rao coded MIMO-OFDM. In this case, the transmitted signal tensor contains Khatri-Rao coded symbols and therefore it has an additional CP structure. Note that this Khatri-Rao coding introduces an additional spreading of the data symbols. To increase the spectral efficiency, we introduce a random coding such that the "coding matrix" contains random data symbols. Therefore, the proposed randomly coded MIMO-OFDM system has a higher spectral efficiency than the Khatri-Rao coded MIMO-OFDM while retaining the CP structure of the transmit signal tensor. In addition to MIMO-OFDM systems, we show that MIMO-GFDM systems can also be modeled using the generalized tensor contraction. Exploiting the resulting tensor models at the receiver facilitates the design of several types of receivers for MIMO-OFDM and MIMO-GFDM systems. The aforementioned models and the proposed receivers with the exception of the randomly coded MIMO-OFDM have already been published in [NHdA18], [NHdA17], and partially in [NCH⁺17].

Furthermore, in Chapter 6 we focus on tensor based signal analysis for biomedical applications. In particular, we show the benefits of using the C-SECSI framework proposed in Chapter 3 for the joint analysis of EEG and MEG signals. It is assumed that the EEG and MEG signals can be coupled because they are typically acquired simultaneously and capture the same brain activity. In this chapter, we present two applications of the joint EEG-MEG signal analysis using the C-SECSI framework for the analysis of measurement data. In the first application, we provide a validation of the photic driving effect using the C-SECSI framework. The photic driving effect is

represented by the resulting oscillations in the human brain that are caused by stimulation of the brain by repetitive light flashes that is also known as IPS (Intermittent Photic Stimulation). The measured EEG and MEG signals used in this thesis are result from an IPS experiment on twelve healthy volunteers using twenty IPS frequencies [SSK⁺16]. IPS causes a frequency entrainment that is indicated by the synchronization of the individual brain rhythm of each volunteer with the photic stimulation frequency. In this thesis, we analyze the frequency entrainment after the joint decomposition of the EEG and MEG signals based on the coupled CP decomposition. In the second application, we present a validation of a controlled experiment based on a joint EEG-MEG signal decomposition in order to show the effect of skull defects on the measurement signals. The effect of the skull defects on the EEG and MEG signals in rabbits is studied in a controlled experimental setup [LFH14] using a sinusoidal constant-current as a source. In this thesis, we analyze the signals from the experimental setup described in [LFH14] using the coupled CP decomposition and show that the tensor decomposition produces meaningful components with respect to the experimental setup. Both applications considered in this thesis show that the joint EEG-MEG signal analysis using the C-SECSI framework is a robust method for the extraction and separation of meaningful components from multi-dimensional biomedical measurement signals. These applications have been published in [NKHH17] and [NLA⁺17].

Finally, we conclude this thesis and present the future work in Chapter 7. In Appendix A, we list the acronyms, the symbols, and the used notation. Some proofs and derivations that are required on several occasions in this thesis are provided in Appendix B.

Chapter 2

Tensor Algebra

A tensor represents a multi-dimensional array of numbers with respect to a multi-dimensional coordinate system. Tensors preserve the multi-dimensional data structure and provide an improved identifiability. Therefore, tensor algebra has a broad range of applications such as chemometrics, psychometrics, numerical mathematics, image signal processing, signal processing for communications, biomedical signal processing, and many more [KB09, CMP⁺15].

This chapter contains three sections devoted to the fundamental concepts of tensor algebra, tensor decompositions, and applications of tensor algebra. In the section about fundamental concepts of tensor algebra, we review some basic definitions from tensor algebra, the properties of the Kronecker product, the properties of the Khatri-Rao product, and the properties of the Hadamard product. Moreover, in tensor algebra applications we often observe data that have a structure of a Khatri-Rao product or a Kronecker product. In the case of noise corrupted data, we can factorize these products in a least-squares sense in order to identify the underlying parameters. Therefore, in the section fundamental concepts of tensor algebra we also review the least-squares Kronecker factorization and the least-squares Khatri-Rao factorization of matrices. Furthermore, in some applications we encounter data models described using a slice-wise multiplication or an element-wise multiplication. This slice-wise or element-wise description does not reveal the tensor structure explicitly. As an alternative, we propose to represent the slice-wise multiplication of two tensors by a generalized contraction of two tensors. Therefore, in the section devoted to fundamental concepts of tensor algebra, we present novel contraction properties for element-wise and slice-wise multiplications.

Moreover, there exist a number of different tensor decompositions. In the section tensor decompositions, we review the HOSVD (Higher Order Singular Value Decomposition), the CP/PARAFAC (Canonical Polyadic/PARallel FACtors) decomposition, the BTD (Block Term Decomposition), the PARATUCK2 (PARAFAC and TUCKer2) decomposition, and the PARAFAC2 (PARallel FACtors 2) decomposition. As previously mentioned, the tensor decompositions have many applications. In the last section of this chapter, we list some of these applications.

2.1 Fundamental Concepts of Tensor Algebra

The goal of this section is to introduce the necessary notation and tensor algebra definitions used throughout this thesis. Accordingly, first the fundamental concepts and tensor algebra definitions are provided followed by some properties of the Kronecker product, Khatri-Rao product, Hadamard

product, and operators like the vectorization and diagonalization operator. Moreover, we include a brief overview of often used tools, i.e., the least-squares Khatri-Rao Kronecker factorization and the least-squares Kronecker factorization. Our contribution to the tensor contraction operator and its properties is presented at the end of this section. A detailed list of the used notation is available in Appendix A.2.

2.1.1 Fundamental Concepts and Definitions

An N -way tensor represents an N -dimensional array of numbers. We denote an N -way tensor by $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$. As a matrix is an element of an outer product of two linear spaces, a tensor is an element from an outer product of N linear spaces [KB09, DLDMV00b, CMP⁺15, CLdA09, SDLF⁺17].

$$\mathcal{A} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \dots \circ \mathbf{a}_N$$

Each of the linear spaces is represented by the column vector \mathbf{a}_i of length I_n , $\forall n = 1, 2, \dots, N$. The symbol \circ denotes the outer product.

The n -mode vectors or fibers are obtained by fixing every index but one [DLDMV00b, KB09]. A tensor of order three, $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ has 1-mode, 2-mode, and 3-mode vectors (column, row, and tube fibers). They are denoted by $\mathcal{A}_{(:, i_2, i_3)}$, $\mathcal{A}_{(i_1, :, i_3)}$, and $\mathcal{A}_{(i_1, i_2, :)}$. Moreover, the two dimensional matrices resulting from varying two indices and fixing the rest are called slices [KB09]. For a tensor of order three $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$, we can define horizontal, lateral, and frontal slices denoted by $\mathcal{A}_{(i_1, :, :)}$, $\mathcal{A}_{(:, i_2, :)}$, and $\mathcal{A}_{(:, :, i_3)}$, respectively.

Moreover, a matricization (which means transforming a tensor into a matrix) is also called an unfolding or flattening [KB09, DLDMV00b, CMP⁺15, CLdA09, SDLF⁺17]. There are three unfoldings, 1-mode unfolding, 2-mode unfolding, and 3-mode unfolding for a tensor of order three, $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$. We denote these tensor unfoldings by $[\mathcal{A}]_{(1)} \in \mathbb{C}^{I_1 \times I_2 I_3}$, $[\mathcal{A}]_{(2)} \in \mathbb{C}^{I_2 \times I_1 I_3}$ and $[\mathcal{A}]_{(3)} \in \mathbb{C}^{I_3 \times I_1 I_2}$. The matricization is performed by arranging the corresponding n -mode vectors into a matrix. Different orderings of the n -mode vectors lead to different tensor unfoldings. According to the forward ordering, for the n -mode unfolding of an N -way tensor ($[\mathcal{A}]_{(n)}$) the n -mode vectors are ordered in increasing order. Hence, we vary the indices starting with the first index (which varies the fastest) until the N -th index with the exception of the n -th index. A visualization of the three different unfoldings of a tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ using the forward ordering is depicted in Fig. 2.1. Another common ordering is the reverse cyclical ordering. According to the reverse cyclical ordering, the matricization $[\mathcal{A}]_{(n)}$ of an N -way tensor is performed such that we first vary the $(n-1)$ -th index, then the $(n-2)$ -th, in a reverse order up to the first index. Afterwards, we start over with the N -th index and keep varying the indices in decreasing order until the $(n+1)$ -th index. The three unfoldings of a 3-way tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ according to the reverse cyclical ordering are visualized in Fig. 2.2. The different ordering types consecutively lead to different definitions of the tensor operations. Therefore, one must be careful which ordering is used and consistently use only one ordering while deriving new concepts or results. In general,

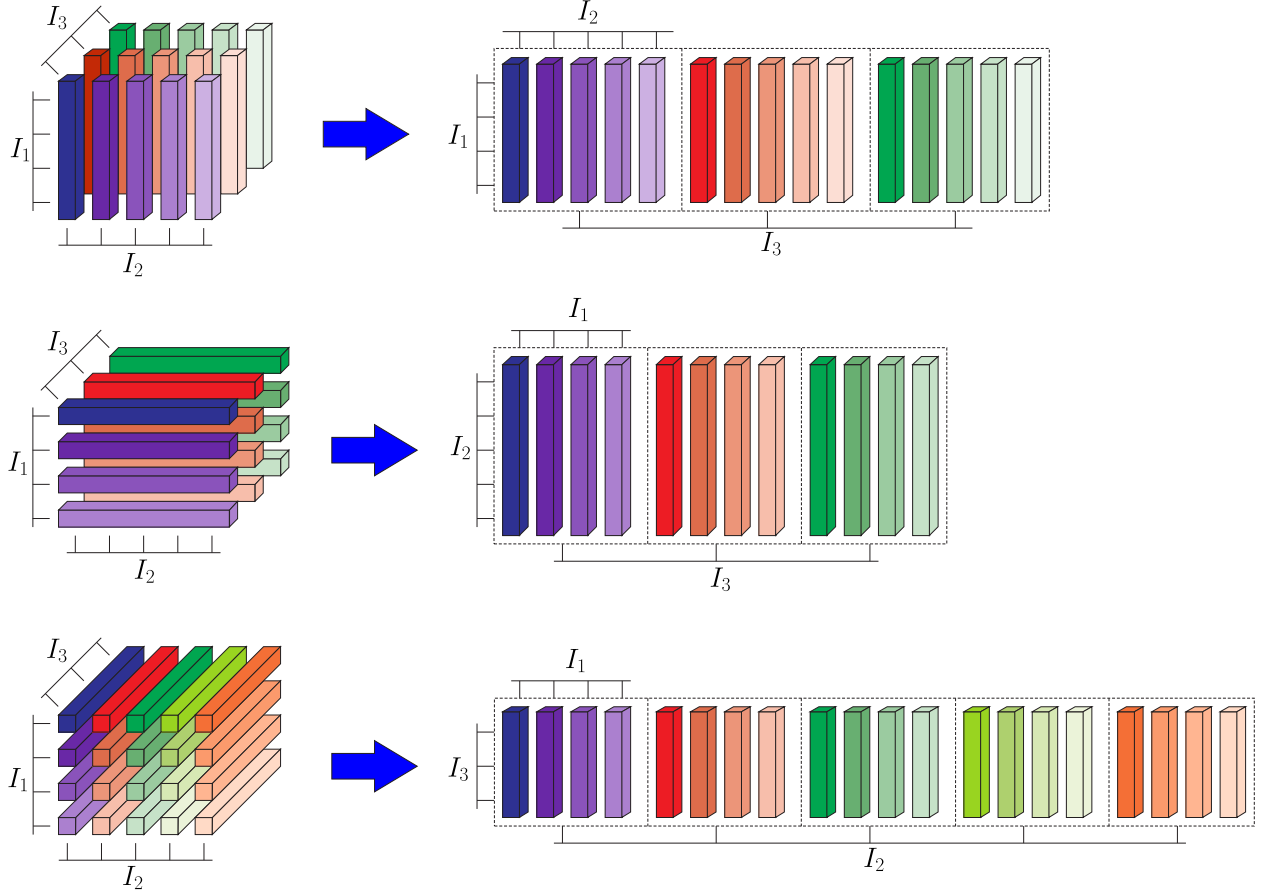


Figure 2.1.: 1-mode, 2-mode, and 3-mode unfoldings of a 3-way tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ using forward ordering of the n -mode vectors.

throughout this thesis we use the forward ordering unless stated otherwise.

In addition to the n -mode unfoldings, generalized matrix unfoldings can be defined by using two subsets of any of the N dimensions [LA11, RSH12]. For instance, the set of modes $(1, 2, \dots, N)$ of an N -way tensor \mathcal{A} can be divided into two non-overlapping, P and $N - P$ dimensional subsets, $\alpha^{(1)} = [\alpha_1 \dots \alpha_P]$ and $\alpha^{(2)} = [\alpha_{P+1} \dots \alpha_N]$, respectively. This leads to the generalized unfolding $[\mathcal{A}]_{(\alpha^{(1)}, \alpha^{(2)})}$, where the indices contained in $\alpha^{(1)}$ vary along the rows and the indices contained in $\alpha^{(2)}$ vary along the columns. Here, the index α_1 varies the fastest between the rows, the index α_{P+1} varies the fastest between the columns, P is any number between one and N , and α_n is any of the tensor dimensions. For instance, let us assume a 4-way tensor $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$. In the generalized unfolding $[\mathcal{A}]_{([1,2],[3,4])}$ the 1-st mode varies faster than the 2-nd mode along the rows and the 3-rd mode varies faster than the 4-th mode along the columns. A visualization of this generalized unfolding and the index ordering is depicted in Fig. 2.3. Hence, the n -mode unfoldings defined previously (Figs. 2.1 and 2.2) are special cases of the generalized unfoldings. For instance, the 2-mode unfolding of the tensor $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$ according to the forward ordering and according to the reverse cyclic ordering can be expressed as $[\mathcal{A}]_{(2,[1,3,4])}$ and $[\mathcal{A}]_{(2,[1,4,3])}$, respectively.

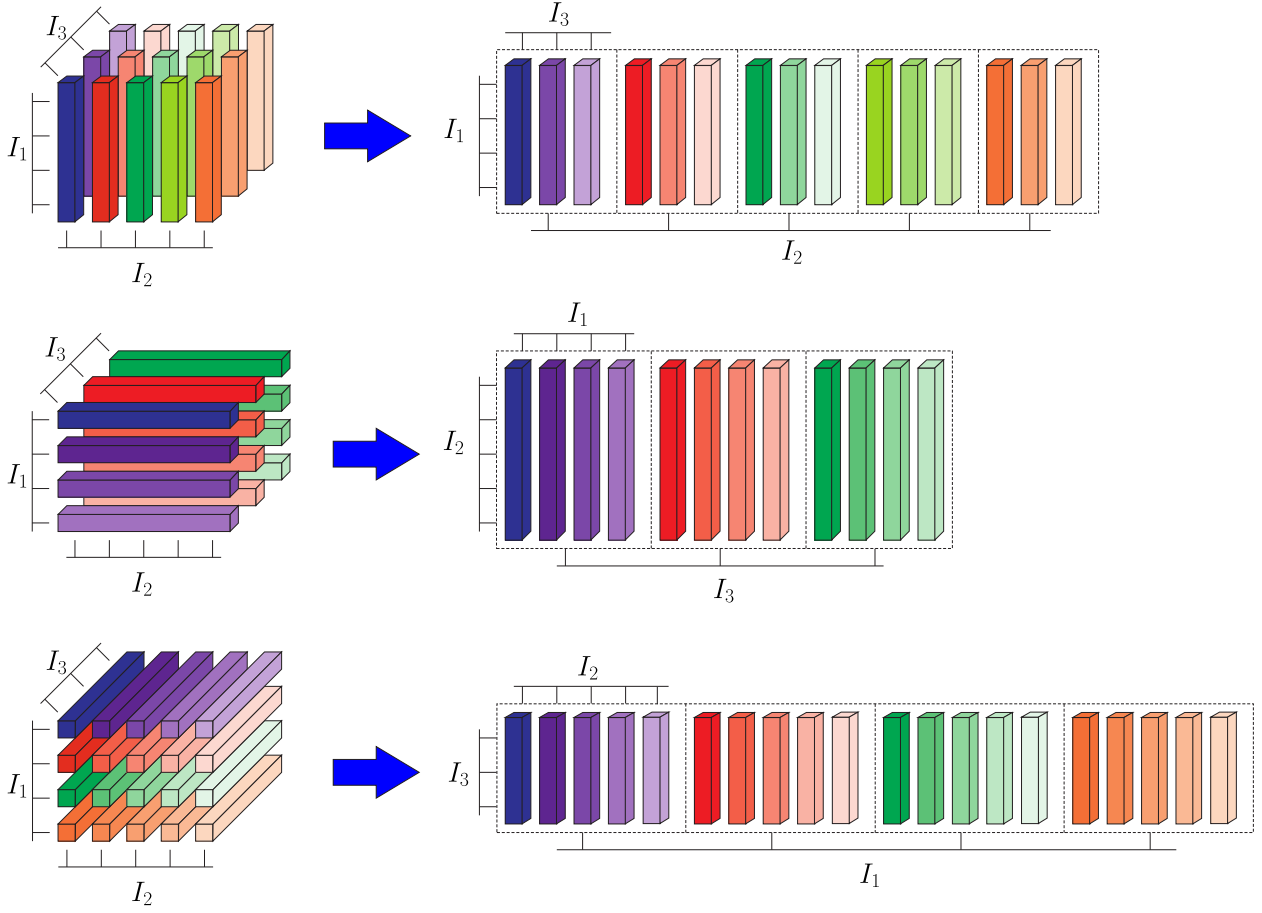


Figure 2.2.: 1-mode, 2-mode, and 3-mode unfoldings of a 3-way tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ using reverse cyclical ordering of the n -mode vectors.

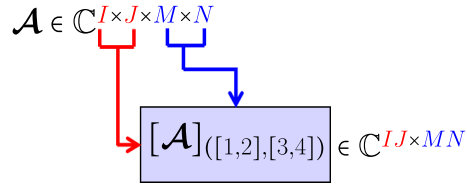


Figure 2.3.: Visualization of the generalized unfolding $[\mathcal{A}]_{([1,2],[3,4])}$.

We can define a linear and a bilinear function in terms of a vector or a matrix multiplication, respectively. Assuming a column vector $\mathbf{x} \in \mathbb{C}^N$ with elements x_n and a column vector $\mathbf{a} \in \mathbb{C}^N$ with elements a_n , a linear function $f(\mathbf{x})$ that is linear in all x_n is defined as

$$f(\mathbf{x}) = \sum_{n=1}^N a_n \cdot x_n = \mathbf{x}^T \cdot \mathbf{a},$$

where T denotes vector/matrix transpose. Now, let us assume that the column vectors $\mathbf{x} \in \mathbb{C}^N$ and $\mathbf{y} \in \mathbb{C}^M$ contain the elements x_n and y_m , respectively. For a bilinear function $f(\mathbf{x}, \mathbf{y})$ that is linear

in all x_n for fixed y_m and linear in all y_m for fixed x_n , we have

$$f(\mathbf{x}, \mathbf{y}) = \sum_{n=1}^N \sum_{m=1}^M a_{n,m} \cdot x_n \cdot y_m = \mathbf{x}^T \cdot \mathbf{A} \cdot \mathbf{y},$$

where $\mathbf{A} \in \mathbb{C}^{N \times M}$ and $\mathbf{A}_{(n,m)} = a_{n,m}$. Next, we assume that the column vectors $\mathbf{x} \in \mathbb{C}^N$, $\mathbf{y} \in \mathbb{C}^M$, and $\mathbf{z} \in \mathbb{C}^P$ contain the elements x_n , y_m , and z_p , respectively. Moreover, we assume that the tensor $\mathcal{A} \in \mathbb{C}^{N \times M \times P}$ contains the elements $a_{n,m,p}$. For a trilinear function $f(\mathbf{x}, \mathbf{y}, \mathbf{z})$ that is linear in all x_n for fixed y_m and z_p , linear in all y_m for fixed x_n and z_p , and linear in all z_p for fixed x_n and y_m , we have

$$f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{n=1}^N \sum_{m=1}^M \sum_{p=1}^P a_{n,m,p} \cdot x_n \cdot y_m \cdot z_p. \quad (2.1)$$

This trilinear function cannot be defined in terms of a vector or a matrix multiplication and therefore an extension to multi-linear algebra is required. Hence, for an N -way tensor the tensor algebra defines a multiplication along the n -th mode [KB09, DLDMV00b, CMP⁺15, CLdA09, SDLF⁺17]. This n -mode product between a tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ and a matrix $\mathbf{X} \in \mathbb{C}^{J \times I_n}$ is denoted by $\mathcal{T} = \mathcal{A} \times_n \mathbf{X} \in \mathbb{C}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$. Using the element-wise notation for the n -mode product, we have

$$(\mathcal{A} \times_n \mathbf{X})_{(i_1, i_2, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N)} = \sum_{i_n=1}^{I_n} \mathcal{A}_{(i_1, i_2, \dots, i_N)} \cdot \mathbf{X}_{(j, i_n)}.$$

The n -mode product can also be defined by means of the tensor unfoldings. Accordingly, the n -mode product can be computed using the n -mode unfolding of the tensor, i.e.,

$$\mathcal{T} = \mathcal{A} \times_n \mathbf{X} \Leftrightarrow [\mathcal{T}]_{(n)} = \mathbf{X} \cdot [\mathcal{A}]_{(n)}. \quad (2.2)$$

Finally, the trilinear function in equation (2.1) can be expressed as

$$f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{n=1}^N \sum_{m=1}^M \sum_{p=1}^P a_{n,m,p} \cdot x_n \cdot y_m \cdot z_p = \mathcal{A} \times_1 \mathbf{x} \times_2 \mathbf{y} \times_3 \mathbf{z},$$

using tensor notation. Accordingly, the tensor $\mathcal{A} \in \mathbb{C}^{N \times M \times P}$ has elements $\mathcal{A}_{(n,m,p)} = a_{n,m,p}$.

Assuming a tensor \mathcal{X} and the matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} of compatible dimensions, the properties for the n -mode product can be summarized as follows. First, the order of multiplication along different modes is irrelevant, i.e.,

$$\mathcal{Y} = \mathcal{X} \times_m \mathbf{A} \times_n \mathbf{B} = \mathcal{X} \times_n \mathbf{B} \times_m \mathbf{A} \\ \text{if } m \neq n.$$

For the multiplication along the same mode, we get

$$\mathcal{Y} = (\mathcal{X} \times_n \mathbf{A}) \times_n \mathbf{D} = \mathcal{X} \times_n (\mathbf{D} \cdot \mathbf{A}).$$

Moreover, we assume the tensors $\mathcal{Y} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ and $\mathcal{X} \in \mathbb{C}^{J_1 \times J_2 \times J_3}$ and the matrices $\mathbf{A} \in \mathbb{C}^{I_1 \times J_1}$, $\mathbf{B} \in \mathbb{C}^{I_2 \times J_2}$, and $\mathbf{C} \in \mathbb{C}^{I_3 \times J_3}$. If $\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$ using the forward ordering, we get the following properties

$$\begin{aligned} [\mathcal{Y}]_{(1)} &= \mathbf{A} \cdot [\mathcal{X}]_{(1)} \cdot (\mathbf{C} \otimes \mathbf{B})^T, \\ [\mathcal{Y}]_{(2)} &= \mathbf{B} \cdot [\mathcal{X}]_{(2)} \cdot (\mathbf{C} \otimes \mathbf{A})^T, \text{ and} \\ [\mathcal{Y}]_{(3)} &= \mathbf{C} \cdot [\mathcal{X}]_{(3)} \cdot (\mathbf{B} \otimes \mathbf{A})^T, \end{aligned}$$

where \otimes denotes the Kronecker product (for the definition of the Kronecker product see Appendix A.2 equation (A.3)). In general, for $\mathcal{Y} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$, $\mathcal{X} \in \mathbb{C}^{J_1 \times J_2 \times \dots \times J_N}$, and $\mathbf{M}^{(n)} \in \mathbb{C}^{I_n \times J_n}$, $\forall n = 1, \dots, N$, if

$$\mathcal{Y} = \mathcal{X} \times_1 \mathbf{M}^{(1)} \times_2 \mathbf{M}^{(2)} \times_3 \dots \times_N \mathbf{M}^{(N)} \quad (2.3)$$

using the forward ordering of the n -mode vectors, we have

$$[\mathcal{Y}]_{(n)} = \mathbf{M}^{(n)} \cdot [\mathcal{X}]_{(n)} \cdot \left(\mathbf{M}^{(N)} \otimes \dots \otimes \mathbf{M}^{(n+1)} \otimes \mathbf{M}^{(n-1)} \otimes \dots \otimes \mathbf{M}^{(1)} \right)^T. \quad (2.4)$$

On the other hand, using the cyclic ordering, we have

$$[\mathcal{Y}]_{(n)} = \mathbf{M}^{(n)} \cdot [\mathcal{X}]_{(n)} \cdot \left(\mathbf{M}^{(n+1)} \otimes \dots \otimes \mathbf{M}^{(N)} \otimes \mathbf{M}^{(1)} \otimes \dots \otimes \mathbf{M}^{(n-1)} \right)^T. \quad (2.5)$$

Moreover, for the generalized unfolding $[\mathcal{X}]_{([1,2,\dots,L],[L+1,\dots,N])}$, we have

$$\begin{aligned} [\mathcal{Y}]_{([1,2,\dots,L],[L+1,\dots,N])} &= \\ &= \left(\mathbf{M}^{(L)} \otimes \dots \otimes \mathbf{M}^{(2)} \otimes \mathbf{M}^{(1)} \right) \cdot [\mathcal{X}]_{([1,2,\dots,L],[L+1,\dots,N])} \cdot \left(\mathbf{M}^{(N)} \otimes \mathbf{M}^{(N-1)} \otimes \dots \otimes \mathbf{M}^{(L+1)} \right)^T. \end{aligned} \quad (2.6)$$

Furthermore, let us consider the $\text{vec}(\cdot)$ operator that transforms a tensor or a matrix into a column vector, thereby the stacking of the elements is performed in an increasing order, starting from the first mode, until the N -th mode for an N -way tensor. A vectorization of a tensor \mathcal{Y} can also be defined in terms of a tensor unfolding [FdA14b], where \mathcal{Y} is given in (2.3).

$$\text{vec}(\mathcal{Y}) = [\mathcal{Y}]_{([1,2,\dots,N],0)} = \left(\mathbf{M}^{(N)} \otimes \dots \otimes \mathbf{M}^{(2)} \otimes \mathbf{M}^{(1)} \right) \cdot [\mathcal{X}]_{([1,2,\dots,N],0)} \quad (2.7)$$

Note that for equation (2.3) we also use the shorthand notation $\mathcal{Y} = \mathcal{X} \times_{n=1}^N \mathbf{M}^{(n)}$.

Similar to matrices, a Kronecker product between two tensors $\mathcal{A} \in \mathbb{C}^{M \times N \times L}$ and $\mathcal{B} \in \mathbb{C}^{P \times Q \times R}$ can be defined as $\mathcal{K} = \mathcal{A} \otimes \mathcal{B} \in \mathbb{C}^{PM \times QN \times RL}$ [Cic14]. Moreover, the Kronecker product between a

tensor $\mathcal{A} \in \mathbb{C}^{M \times N \times L}$ and a matrix $B \in \mathbb{C}^{P \times Q}$ equals $\mathcal{K} = \mathcal{A} \otimes B \in \mathbb{C}^{PM \times QN \times L}$. For $\mathcal{A} \in \mathbb{C}^{2 \times 2 \times 2}$ and $B \in \mathbb{C}^{2 \times 2}$, we depict the result in Fig. 2.4 in terms of the scalar elements.

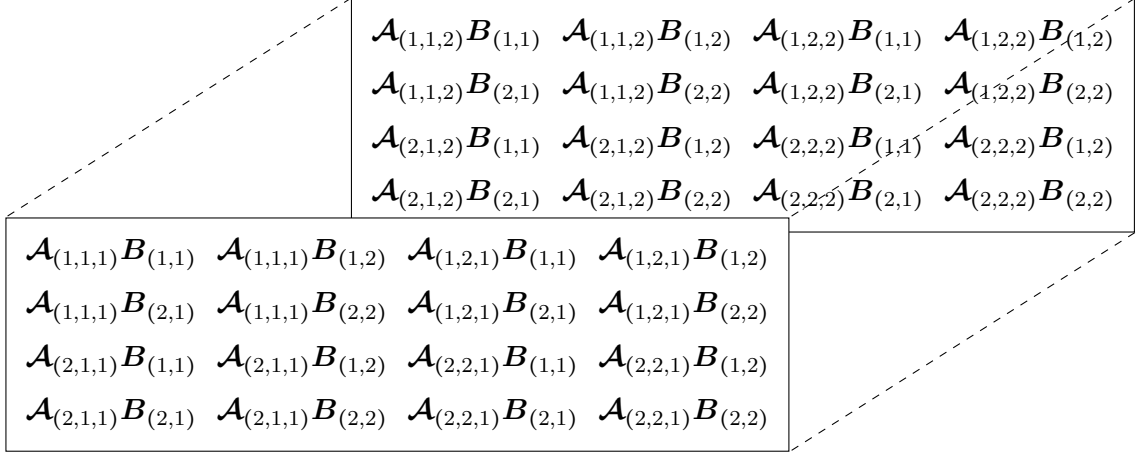


Figure 2.4.: Kronecker product between a tensor $\mathcal{A} \in \mathbb{C}^{2 \times 2 \times 2}$ and a matrix $B \in \mathbb{C}^{2 \times 2}$ in terms of the scalar elements.

The Khatri-Rao product between two matrices that is the column-wise Kronecker product we denote by \diamond (see Appendix A.2 equation (A.4)).

Furthermore, the tensor contraction $\mathcal{A} \bullet_n^m \mathcal{B}$ between two tensors $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ and $\mathcal{B} \in \mathbb{C}^{J_1 \times J_2 \times \dots \times J_N}$ represents an inner product of the n -th mode of \mathcal{A} with the m -th mode of \mathcal{B} , provided that $I_n = J_m$ [Cic14, HRDG08, CGLM08].

$$\mathcal{T}_{(i_1, \dots, i_{n-1}, i_{n+1}, \dots, i_N, j_1, \dots, j_{m-1}, j_{m+1}, \dots, j_M)} = (\mathcal{A} \bullet_n^m \mathcal{B})_{(i_1, \dots, i_{n-1}, i_{n+1}, \dots, i_N, j_1, \dots, j_{m-1}, j_{m+1}, \dots, j_M)} \quad (2.8)$$

$$\mathcal{T}_{(i_1, \dots, i_{n-1}, i_{n+1}, \dots, i_N, j_1, \dots, j_{m-1}, j_{m+1}, \dots, j_M)} = \sum_{i=1}^{I_n} \mathcal{A}_{(i_1, \dots, i_{n-1}, i, i_{n+1}, \dots, i_N)} \cdot \mathcal{B}_{(j_1, \dots, j_{m-1}, i, j_{m+1}, \dots, j_M)} \quad (2.9)$$

The resulting tensor \mathcal{T} contains the remaining dimensions of both tensors. Hence, the dimensions $I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N \times J_1 \times \dots \times J_{m-1} \times J_{m+1} \times \dots \times J_M$, correspond to the dimensions of the tensor \mathcal{A} in an increasing order except for the contracted n -th dimension followed by the dimensions of the tensor \mathcal{B} in an increasing order except for the contracted m -th dimension.

In Fig. 2.5a, we depict an example of the tensor contraction $\mathcal{T} = \mathcal{A} \bullet_{\frac{1}{3}}^{\frac{1}{3}} \mathcal{B} \in \mathbb{C}^{I \times J \times N \times N \times K}$ between the tensors $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$ and $\mathcal{B} \in \mathbb{C}^{M \times N \times K}$. Moreover, the contraction along several modes of compatible dimensions is also possible, which is called generalized contraction. Accordingly, the generalized contraction along two modes (double contraction) is denoted as $\mathcal{A} \bullet_{n,k}^{m,l} \mathcal{B}$. In particular, element-wise the contraction along two modes between the tensors $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$ and $\mathcal{B} \in \mathbb{C}^{M \times N \times K}$ is defined as [Cic14],

$$(\mathcal{A} \bullet_{\frac{1}{3}, \frac{1}{4}}^{\frac{1}{2}, \frac{2}{2}} \mathcal{B})_{(i,j,k)} \triangleq \sum_{n=1}^N \sum_{m=1}^M \mathcal{A}_{(i,j,m,n)} \cdot \mathcal{B}_{(m,n,k)} = \mathcal{T}_{(i,j,k)}.$$

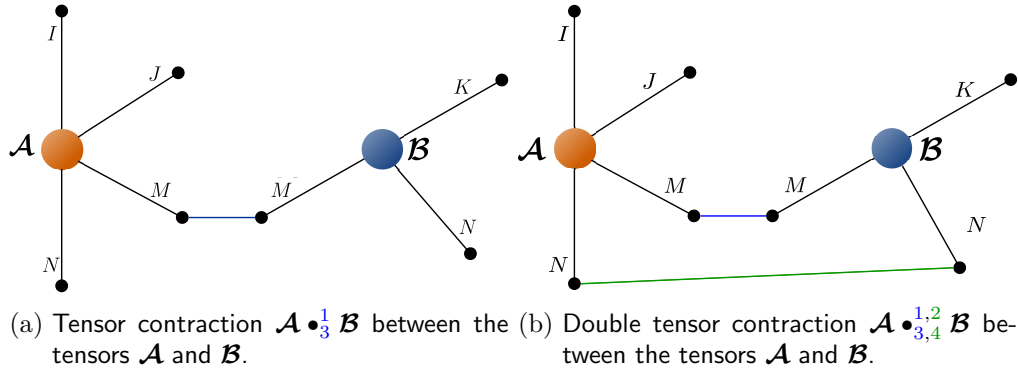


Figure 2.5.: Generalized tensor contraction between the tensors $\mathcal{A} \in \mathbb{C}^{I \times J \times M \times N}$ and $\mathcal{B} \in \mathbb{C}^{M \times N \times K}$.

This example, $\mathcal{A} \bullet_{3,4}^{1,2} \mathcal{B}$ is visualized in Fig. 2.5b. It represents a contraction of the 3-rd and the 4-th mode of \mathcal{A} with the 1-st and the 2-nd mode of \mathcal{B} , respectively. Using the concept of generalized unfoldings, it can be shown that the tensor contraction satisfies

$$[\mathcal{A} \bullet_{3,4}^{1,2} \mathcal{B}]_{([1,2],3)} = [\mathcal{A}]_{([1,2],[3,4])} \cdot [\mathcal{B}]_{([1,2],3)} = \quad (2.10)$$

$$[\mathcal{A} \bullet_{4,3}^{2,1} \mathcal{B}]_{([1,2],3)} = [\mathcal{A}]_{([1,2],[4,3])} \cdot [\mathcal{B}]_{([2,1],3)}. \quad (2.11)$$

The equations (2.10) and (2.11) can be extended to any suitable set of modes and tensors. From the definition of the contraction operator follows that the order of the tensors is irrelevant. The resulting tensors have only different ordering of the dimensions. For instance, $\mathcal{A} \bullet_{3,4}^{1,2} \mathcal{B} = \mathcal{T}_1$ and $\mathcal{B} \bullet_{1,2}^{3,4} \mathcal{A} = \mathcal{T}_2$. Both tensors $\mathcal{T}_1 \in \mathbb{C}^{I \times J \times K}$ and $\mathcal{T}_2 \in \mathbb{C}^{K \times I \times J}$ contain the same elements, but their dimensions are permuted. Since we always specify which mode is unfolded or multiplied, the permuted dimensions are irrelevant for us. Note that the tensor contraction is a generalization of the n -mode product and matrix multiplication. To this end, the n -mode product between a tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ and a matrix $\mathbf{X} \in \mathbb{C}^{J \times I_n}$ defined in equation (2.2) can be express as a contraction, i.e., $\mathcal{A} \bullet_n^2 \mathbf{X}$. Furthermore, the matrix products $\mathbf{B} \cdot \mathbf{X}$ and $\mathbf{B}^T \cdot \mathbf{X}$ can also be expressed in terms of contraction, i.e., $\mathbf{B} \cdot \mathbf{X} = \mathbf{B} \bullet_2^1 \mathbf{X}$ and $\mathbf{B}^T \cdot \mathbf{X} = \mathbf{B} \bullet_1^1 \mathbf{X}$.

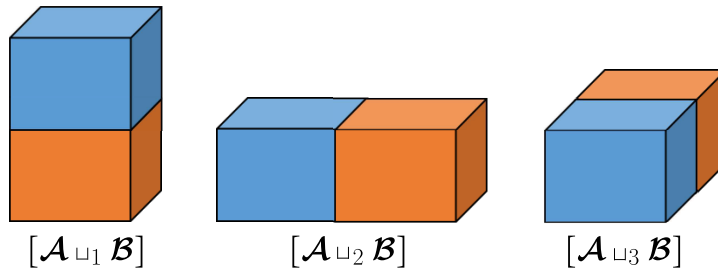


Figure 2.6.: Tensor concatenation along the 1-mode, the 2-mode, and the 3-mode.

Let $[\mathcal{A} \sqcup_n \mathcal{B}]$ denote the concatenation along the n -mode of two tensors $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ and $\mathcal{B} \in \mathbb{C}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$ of compatible dimensions [HRDG08]. A visualization of the concatenation between two 3-way tensors \mathcal{A} and \mathcal{B} along mode one, mode two, and mode three is depicted

in Fig. 2.6.

A super-diagonal identity N -way tensor with dimensions $R \times R \times \dots \times R$ we denote as $\mathcal{I}_{N,R}$. All elements of an identity tensor are zeros except for the elements with all equal indices ($i_1 = i_2 = \dots = i_N$), which equal one [KB09]. An identity tensor $\mathcal{I}_{3,3}$ has a structure depicted in Fig. 2.7.

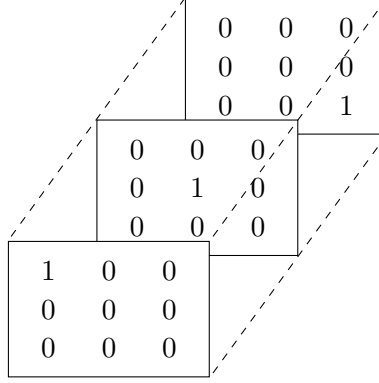


Figure 2.7.: A 3-way identity tensor $\mathcal{I}_{3,3}$ with dimensions $3 \times 3 \times 3$

Finally, an extension of the Frobenius norm to tensors is called higher-order norm. This higher-order norm of the tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ is defined as [DLDMV00b, KB09]

$$\|\mathcal{A}\|_{\text{H}} = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} |\mathcal{A}_{(i_1, i_2, \dots, i_N)}|^2} = \|\text{vec}(\mathcal{A})\|_2 = \left\| [\mathcal{A}]_{(n)} \right\|_{\text{F}}, \forall n = 1, 2, \dots, N,$$

$$\|\mathcal{A}\|_{\text{H}} = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$$

where $\|\cdot\|_{\text{H}}$, $\|\cdot\|_2$, and $\|\cdot\|_{\text{F}}$ denotes the higher-order norm, the Euclidean (two)-norm, and the Frobenius norm, respectively. Moreover, $\langle \cdot \rangle$ denotes the scalar product that is defined in (A.2).

2.1.2 Properties of Kronecker, Khatri-Rao, and Hadamard Products

The tensor unfoldings can be expressed in terms of Kronecker products as shown in (2.4), (2.5), and (2.6). Therefore, during the derivation of new concepts using tensor algebra, the properties of the Kronecker product and the Khatri-Rao product are essential tools. To this end, we summarize the properties of the Kronecker product, the Khatri-Rao product, the Hadamard product, and the vectorization operator in this section.

In the past, the Hadamard and Kronecker products have been studied and applied in matrix theory, system theory, statistics, and many other fields. Thus, many references summarize and derive new properties for these products. Such references include, [Neu69], [Bre78], [Liu99], [LT08], [PP08], and many more.

For the matrices $\mathbf{A} \in \mathbb{C}^{M_1 \times M_2}$, $\mathbf{B} \in \mathbb{C}^{N_1 \times N_2}$, $\mathbf{C} \in \mathbb{C}^{M_1 \times M_2}$, $\mathbf{E} \in \mathbb{C}^{N_2 \times M_2}$, $\mathbf{F} \in \mathbb{C}^{M_2 \times M_2}$, $\mathbf{G} \in \mathbb{C}^{N_2 \times M_2}$, and $\mathbf{H} \in \mathbb{C}^{N_2 \times N_2}$ of compatible dimensions hold the following properties [Bre78], [Liu99],

[LT08], [PP08],

$$(\mathbf{A} \otimes \mathbf{B})^H = \mathbf{A}^H \otimes \mathbf{B}^H \quad (2.12)$$

$$(\mathbf{A} \otimes \mathbf{B})^+ = \mathbf{A}^+ \otimes \mathbf{B}^+ \quad (2.13)$$

$$\|\mathbf{A} \otimes \mathbf{B}\|_F = \|\mathbf{A}\|_F \|\mathbf{B}\|_F \quad (2.14)$$

$$(\mathbf{A} + \mathbf{C}) \diamond \mathbf{E} = \mathbf{A} \diamond \mathbf{E} + \mathbf{C} \diamond \mathbf{E} \quad (2.15)$$

$$(\mathbf{A} \diamond \mathbf{C}) \diamond \mathbf{E} = \mathbf{A} \diamond (\mathbf{C} \diamond \mathbf{E}) \quad (2.16)$$

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{F} \otimes \mathbf{G}) = (\mathbf{A}\mathbf{F}) \otimes (\mathbf{B}\mathbf{G}) \quad (2.17)$$

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{F} \diamond \mathbf{G}) = (\mathbf{A}\mathbf{F}) \diamond (\mathbf{B}\mathbf{G}) \quad (2.18)$$

$$(\mathbf{A} \diamond \mathbf{G})^H (\mathbf{A} \diamond \mathbf{G}) = (\mathbf{A}^H \mathbf{A}) \odot (\mathbf{G}^H \mathbf{G}) \quad (2.19)$$

$$(\mathbf{A} \diamond \mathbf{G})^+ = [(\mathbf{A}^H \mathbf{A}) \odot (\mathbf{G}^H \mathbf{G})]^+ (\mathbf{A} \diamond \mathbf{G})^H \quad (2.20)$$

$$(\mathbf{A} \diamond \mathbf{G}) \odot (\mathbf{C} \diamond \mathbf{E}) = (\mathbf{A} \odot \mathbf{C}) \diamond (\mathbf{G} \odot \mathbf{E}) \quad (2.21)$$

$$(\mathbf{I}_{N_2} \otimes \mathbf{F})(\mathbf{H} \otimes \mathbf{I}_{M_2}) = (\mathbf{H} \otimes \mathbf{I}_{M_2})(\mathbf{I}_{N_2} \otimes \mathbf{F}) \quad (2.22)$$

where H and $^+$ denotes Hermitian transpose and matrix pseudo-inverse, respectively. We symbolize the Hadamard (element-wise) product by \odot .

The $\text{diag}(\cdot)$ operator transforms a vector into a diagonal matrix. On the other hand, by applying $\text{diag}(\cdot)$ onto a matrix we transform the elements on the main diagonal into a column vector. The properties of the $\text{vec}(\cdot)$ and $\text{diag}(\cdot)$ operator shown in [Neu69, Bre78, PP08, LT08] include

$$\text{trace}(\mathbf{A}\mathbf{B}) = \text{trace}(\mathbf{B}\mathbf{A}) = \text{vec}(\mathbf{A}^T)^T \text{vec}(\mathbf{B}), \quad (2.23)$$

$$\text{vec}(\mathbf{A}\mathbf{X}\mathbf{B}) = (\mathbf{B}^T \otimes \mathbf{A}) \text{vec}(\mathbf{X}), \quad (2.24)$$

$$\text{vec}(\mathbf{A}\mathbf{D}\mathbf{B}) = (\mathbf{B}^T \diamond \mathbf{A}) \text{diag}(\mathbf{D}), \quad (2.25)$$

$$(\mathbf{A} \diamond \mathbf{C}) \text{diag}(\mathbf{f}) = (\mathbf{A} \text{diag}(\mathbf{f})) \diamond \mathbf{C} = \mathbf{A} \diamond (\mathbf{C} \text{diag}(\mathbf{f})), \quad \text{and} \quad (2.26)$$

$$\text{diag}(\mathbf{g}) \mathbf{A} \text{diag}(\mathbf{f}) = \mathbf{A} \odot (\mathbf{g}\mathbf{f}^T), \quad (2.27)$$

where $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{B} \in \mathbb{C}^{N \times M}$, $\mathbf{C} \in \mathbb{C}^{K \times N}$, $\mathbf{D} \in \mathbb{C}^{N \times N}$, $\mathbf{f} \in \mathbb{C}^{N \times 1}$, $\mathbf{g} \in \mathbb{C}^{K \times 1}$, and $\mathbf{X} \in \mathbb{C}^{N \times N}$ are compatibly partitioned matrices. Also, we assume that \mathbf{D} is a diagonal matrix.

Moreover, the Khatri-Rao and the Hadamard products can be transformed into Kronecker product by means of selection matrices [Liu99, LT08]. Let us take into account the matrices $\mathbf{A} \in \mathbb{C}^{M \times N}$, $\mathbf{B} \in \mathbb{C}^{K \times N}$, $\mathbf{C} \in \mathbb{C}^{M \times N}$, $\mathbf{X} \in \mathbb{C}^{N \times N}$, the two selection matrices $\mathbf{J}_N \in \mathbb{R}^{N^2 \times N}$ and $\mathbf{J}_M \in \mathbb{R}^{M^2 \times M}$, and an identity matrix \mathbf{I}_N with dimensions $N \times N$. Using the selection matrices with a structure (the structure of this selection matrix is depicted in Appendix B.2.)

$$\mathbf{J}_N = \mathbf{I}_N \diamond \mathbf{I}_N = [\mathcal{I}_{3,N}]_{(1)}^T = [\mathcal{I}_{3,N}]_{(2)}^T = [\mathcal{I}_{3,N}]_{(3)}^T, \quad (2.28)$$

it can be shown that

$$\mathbf{A} \diamond \mathbf{B} = (\mathbf{A} \otimes \mathbf{B}) \mathbf{J}_N \quad (2.29)$$

$$\mathbf{A} \odot \mathbf{C} = \mathbf{J}_M^T (\underbrace{\mathbf{A} \otimes \mathbf{C}}_{\mathbf{J}_N}) \mathbf{J}_N \quad (2.30)$$

$$\mathbf{A} \odot \mathbf{C} = \mathbf{J}_M^T (\mathbf{A} \diamond \mathbf{C}) \quad (2.31)$$

$$\text{diag}(\mathbf{X}) = \mathbf{J}_N^T \text{vec}(\mathbf{X}) \quad (2.32)$$

Further details and references regarding the relation between the different products are available in [Liu99, LT08]. However, most of the properties summarized in this section can be easily proven by means of the selection matrix defined in (2.28). Note that this selection matrix also represents 1-mode unfolding, 2-mode unfolding, and 3-mode unfolding of a 3-way identity tensor. These unfoldings are all equivalent to one another (see equation (2.28)).

2.1.3 Least-Squares Kronecker and Khatri-Rao factorizations

Very often in signal processing applications, we need to solve a so-called least-squares Kronecker or Khatri-Rao factorization. In this section, we provide an overview of the algorithms for the computation of these factorizations and their uniqueness properties.

Assume that $\mathbf{A} = \mathbf{X} \otimes \mathbf{Y} + \mathbf{N} \in \mathbb{C}^{KI \times LJ}$, where $\mathbf{X} \in \mathbb{C}^{I \times J}$ and $\mathbf{Y} \in \mathbb{C}^{K \times L}$. The matrix $\mathbf{N} \in \mathbb{C}^{KI \times LJ}$ represents an uncorrelated ZMCSCG (Zero Mean Circularly Symmetric Complex Gaussian) noise, thus \mathbf{A} is a noisy observation of the Kronecker product between the matrices \mathbf{X} and \mathbf{Y} . The LSKF (Least-Squares Kronecker Factorization) provides an estimate of the matrices \mathbf{X} and \mathbf{Y} in an LS (Least-Squares) sense. To this end, the goal of a LSKF is the estimation of the matrices \mathbf{X} and \mathbf{Y} from the matrix $\mathbf{A} \in \mathbb{C}^{KI \times LJ}$ that is an approximation of their Kronecker product $\mathbf{A} \approx \mathbf{X} \otimes \mathbf{Y} \in \mathbb{C}^{KI \times LJ}$. Such an algorithm was first presented in [VLP93, VLP97]. The LSKF is based on a rank one approximation and it is summarized in Algorithm 2.1. The rank one approximation is computed from the truncated SVD (Singular Value Decomposition) that provides the best rank-one approximation of the reshaped matrix $\bar{\mathbf{A}} \in \mathbb{C}^{KL \times IJ}$ in an LS sense. The reshaping is performed by the means of the permutation matrix \mathbf{P} [dCFR18].

$$\mathbf{P} = \mathbf{I}_J \otimes \left(\sum_{i=1}^I \sum_{l=1}^L \mathbf{e}_{I,i} \mathbf{e}_{L,l}^T \otimes \mathbf{e}_{L,i} \mathbf{e}_{I,i}^T \right) \otimes \mathbf{I}_K,$$

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \text{ for } I = 2, J = 1, L = 3, K = 2 \quad (2.33)$$

Moreover, $\mathbf{e}_{I,i} \in \mathbb{R}^{I \times 1}$ is a pinning vector corresponding to the i -th column of an identity matrix of size $I \times I$. Note that the operator $\text{unvec}_{M \times N}(\cdot)$ transforms a vector into a matrix (or a tensor) of indicated size, i.e., $M \times N$. The $\text{unvec}_{M \times N}(\cdot)$ operator represents the inverse operation of the $\text{vec}(\cdot)$ operator (Appendix A.2). Using the LSKF, we can estimate the unknown matrices $\mathbf{X} \in \mathbb{C}^{I \times J}$ and $\mathbf{Y} \in \mathbb{C}^{K \times L}$ up to one complex scaling factor ambiguity, since $\mathbf{X} \otimes \mathbf{Y} = (c\mathbf{X}) \otimes (\frac{1}{c}\mathbf{Y})$, $\forall c \in \mathbb{C}$, $c \neq 0$. Furthermore, in [dCFR18] LSKF algorithms based on ALS (Alternating Least-Squares) and ALMS (Alternating Least Mean Squares) are considered. However, both of these algorithms are iterative with no guarantee of convergence. In this thesis, the acronym LSKF always refers to the Algorithm 2.1. The extension to a Kronecker product of multiple matrices $\mathbf{X}_1 \otimes \mathbf{X}_2 \otimes \dots \otimes \mathbf{X}_N$ is also possible using the concept of N -way arrays and their rank one approximation. Such a generalization of the SVD to the higher order SVD (HOSVD) [DLDMV00b] of an N -way tensor is discussed in Section 2.2.

Algorithm 2.1: Least-Squares Kronecker Factorization (LSKF)

Data: Given a matrix $\mathbf{A} \in \mathbb{C}^{KI \times LJ}$ that is an approximation of the Kronecker product between the matrices $\mathbf{X} \in \mathbb{C}^{I \times J}$ and $\mathbf{Y} \in \mathbb{C}^{K \times L}$, i.e., $\mathbf{A} \approx \mathbf{X} \otimes \mathbf{Y} \in \mathbb{C}^{KI \times LJ}$.

Reshape the input matrix $\mathbf{A} \in \mathbb{C}^{KI \times LJ}$ into $\bar{\mathbf{A}} = \text{unvec}_{KL \times IJ}(\mathbf{P} \cdot \text{vec}(\mathbf{A}))$, with $\mathbf{P} = \mathbf{I}_J \otimes \left(\sum_{i=1}^I \sum_{l=1}^L \mathbf{e}_{I,i} \mathbf{e}_{L,l}^T \otimes \mathbf{e}_{L,i} \mathbf{e}_{I,i}^T \right) \otimes \mathbf{I}_K$ in equation (2.33).

Compute the SVD of the matrix $\bar{\mathbf{A}} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^H$.

Compute $\hat{\mathbf{x}} = \sqrt{\sigma_1} \cdot \mathbf{v}_1^*$ where $\sigma_1 = \mathbf{\Sigma}_{(1,1)}$ and $\mathbf{v}_1 = \mathbf{V}_{(:,1)}$ is the most dominant singular value and the corresponding right singular vector, respectively.

Compute $\hat{\mathbf{y}} = \sqrt{\sigma_1} \cdot \mathbf{u}_1$ where $\sigma_1 = \mathbf{\Sigma}_{(1,1)}$ and $\mathbf{u}_1 = \mathbf{U}_{(:,1)}$ is the most dominant singular value and the corresponding left singular vector, respectively.

Result: $\hat{\mathbf{X}} = \text{unvec}_{I \times J}(\hat{\mathbf{x}})$ and $\hat{\mathbf{Y}} = \text{unvec}_{K \times L}(\hat{\mathbf{y}})$

Algorithm 2.2: Least-Squares Khatri-Rao Factorization (LSKRF)

Data: Given a matrix $\mathbf{B} \in \mathbb{C}^{PM \times N}$ that is an approximation of the Khatri-Rao product between the matrices $\mathbf{Z} \in \mathbb{C}^{M \times N}$ and $\mathbf{W} \in \mathbb{C}^{P \times N}$, i.e., $\mathbf{B} \approx \mathbf{Z} \diamond \mathbf{W} \in \mathbb{C}^{KI \times LJ}$

for $n = 1 : N$ **do**

 Select the n -th column of the matrix \mathbf{B} , i.e., $\mathbf{B}_{(:,n)} \in \mathbb{C}^{PM \times 1}$.

 Reshape the vector $\mathbf{B}_{(:,n)}$ into a matrix $\bar{\mathbf{B}}_n = \text{unvec}_{P \times M}(\mathbf{B}_{(:,n)})$.

 Compute the SVD of the matrix $\bar{\mathbf{B}}_n = \mathbf{U}_n \mathbf{\Sigma}_n \mathbf{V}_n^H$.

 Compute the n -th column of the matrix $\hat{\mathbf{Z}}$, $\hat{\mathbf{Z}}_{(:,n)} = \sqrt{\sigma_{n,1}} \cdot \mathbf{v}_{n,1}^*$ where $\sigma_{n,1} = \Sigma_{n(1,1)}$ and $\mathbf{v}_{n,1} = \mathbf{V}_{n(:,1)}$ is the most dominant singular value and the corresponding right singular vector, respectively.

 Compute the n -th column of the matrix $\hat{\mathbf{W}}$, $\hat{\mathbf{W}}_{(:,n)} = \sqrt{\sigma_{n,1}} \cdot \mathbf{u}_{n,1}$ where $\sigma_{n,1} = \Sigma_{n(1,1)}$ and $\mathbf{u}_{n,1} = \mathbf{U}_{n(:,1)}$ is the most dominant singular value and the corresponding left singular vector, respectively.

end

Result: $\hat{\mathbf{Z}}$ and $\hat{\mathbf{W}}$

Similar to the LSKF, the LSKRF (Least-Squares Khatri-Rao Factorization) provides an estimate of the matrices $\mathbf{Z} \in \mathbb{C}^{M \times N}$ and $\mathbf{W} \in \mathbb{C}^{P \times N}$ given a matrix $\mathbf{B} \in \mathbb{C}^{PM \times N}$. The matrix $\mathbf{B} = \mathbf{Z} \diamond \mathbf{W} + \mathbf{N}$ is a noisy observation of the Khatri-Rao product between the matrices \mathbf{Z} and \mathbf{W} . The matrix $\mathbf{N} \in \mathbb{C}^{PM \times N}$ represents the noise matrix containing ZMCSCG noise. The LSKRF algorithm, summarized in Algorithm 2.2 has been proposed and utilized in [RH09a, RH09b] and [RH10]. Since the Khatri-Rao product is a column-wise Kronecker product, this LSKRF algorithm computes the best rank-one approximation in a column-wise fashion. Using the LSKRF, the matrices $\mathbf{Z} \in \mathbb{C}^{M \times N}$ and $\mathbf{W} \in \mathbb{C}^{P \times N}$ can be estimated up to one complex scaling factor ambiguity per column. Hence, the estimated matrices are equal to $\hat{\mathbf{Z}} = \mathbf{Z} \mathbf{\Lambda}$ and $\hat{\mathbf{W}} = \mathbf{W} \mathbf{\Lambda}^{-1}$, where $\mathbf{\Lambda} \in \mathbb{C}^{N \times N}$ is a diagonal matrix with diagonal elements equal to the N complex scaling ambiguities. The factorization of a Khatri-Rao product of multiple matrices $\mathbf{X}_1 \diamond \mathbf{X}_2 \diamond \dots \diamond \mathbf{X}_N$ using the concept of N -way tensors is proposed in [RH13a].

2.1.4 New Contraction Properties for Element-wise and Slice-wise Multiplication

In many tensor applications, we only have an element-wise or a slice-wise description of our data/signal model. For instance, there exist only a slice-wise description of the PARATUCK2 decomposition and the PARAFAC2 decomposition corresponding to a certain unfolding of the overall tensor [HL96, Har72]. In the same way, some proposed tensor based models for MIMO-ODFM (Multiple Input Multiple Output-Orthogonal Frequency Division Multiplexing) communication sys-

tems have only an element-wise or a slice-wise representation [dAFX13,FdA14a]. Further examples include the slice-wise description of two-way relaying MIMO communication systems [ZNNH15,XFdAS14]. This description of the signal models does not reveal the tensor structure explicitly. Hence, the derivation of all tensor unfoldings is not always obvious. Therefore, we propose to express the slice-wise multiplication of two tensors in terms of the tensor contraction operator. In this section, we summarize novel properties of the contraction operator for element-wise and slice-wise multiplications [NCdAH18]. Later in this thesis, we exploit these properties for the representation of the PARATUCK2 decomposition, the PARAFAC2 decomposition, and multi-carrier MIMO wireless communication systems in terms of generalized tensor contraction. Next, by substituting the individual tensor structure of the tensors involved in the contraction, we are able to derive novel explicit models of the overall tensor. The resulting tensor models do not depend on a chosen unfolding and capture all of the unfoldings at the same time.

Hadamard product via generalized tensor contraction

First, let us consider a Hadamard product (element-wise multiplication) between two vectors $\mathbf{a} \in \mathbb{C}^{M \times 1}$ and $\mathbf{b} \in \mathbb{C}^{M \times 1}$, $c_{(m)} = a_{(m)}b_{(m)}$, $\forall m = 1, \dots, M$ ($\mathbf{c} \in \mathbb{C}^{M \times 1}$). The Hadamard product can be expressed via the multiplication of a diagonal matrix and a vector, i.e., $\mathbf{a} \odot \mathbf{b} = \text{diag}(\mathbf{a}) \mathbf{b} = \text{diag}(\mathbf{b}) \mathbf{a}$. As explained in Section 2.1.1, the matrix multiplication is equivalent to the contraction \bullet_2^1 . Therefore, we get

$$\mathbf{a} \odot \mathbf{b} = \text{diag}(\mathbf{a}) \bullet_2^1 \mathbf{b} = \text{diag}(\mathbf{b}) \bullet_2^1 \mathbf{a}.$$

Likewise, for a Hadamard product between two row vectors, we get $\mathbf{a}^T \odot \mathbf{b}^T = \mathbf{a}^T \bullet_2^1 \text{diag}(\mathbf{b}^T) = \mathbf{b}^T \bullet_2^1 \text{diag}(\mathbf{a}^T)$.

Next, for the Hadamard product between two matrices $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{M \times N}$, $\mathbf{C}_{(m,n)} = \mathbf{A}_{(m,n)} \mathbf{B}_{(m,n)}$, $\forall m = 1, \dots, M$ and $n = 1, \dots, N$, we can show that $\mathbf{C} = \mathbf{A} \odot \mathbf{B} = \mathcal{D}_A \bullet_{2,4}^{1,2} \mathbf{B} = \mathcal{D}_B \bullet_{2,4}^{1,2} \mathbf{A}$. Here $\mathcal{D}_A \in \mathbb{C}^{M \times M \times N \times N}$ and $\mathcal{D}_B \in \mathbb{C}^{M \times M \times N \times N}$ are diagonal 4-way tensors with the non-zero elements $\mathcal{D}_{A(m,m,n,n)} = \mathbf{A}_{(m,n)}$ and $\mathcal{D}_{B(m,m,n,n)} = \mathbf{B}_{(m,n)}$, respectively. As an alternative, we also have

$$\mathbf{C} = \mathbf{A} \odot \mathbf{B} = \mathcal{D}^{(A)} \bullet_{2,3}^{1,3} \mathcal{D}^{(B)},$$

where the diagonal 3-way tensors have the following non-zero elements $\mathcal{D}^{(A)}_{(m,m,n)} = \mathbf{A}_{(m,n)}$ and $\mathcal{D}^{(B)}_{(m,m,n)} = \mathbf{B}_{(m,n)}$. Moreover, these diagonal 3-way tensors can be either defined in terms of slices,

$$\mathcal{D}^{(A)}_{(:, :, n)} = \text{diag}(\mathbf{A}_{(:, n)}), \forall n = 1, \dots, N \quad \mathcal{D}^{(B)}_{(m, :, :)} = \text{diag}(\mathbf{B}_{(m, :)}), \forall m = 1, \dots, M$$

or using tensor notation $\mathcal{D}^{(A)} = \mathcal{I}_{3,M} \times_3 \mathbf{A}^T$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B}$. We depict the structure of these two tensors in Fig. 2.8.

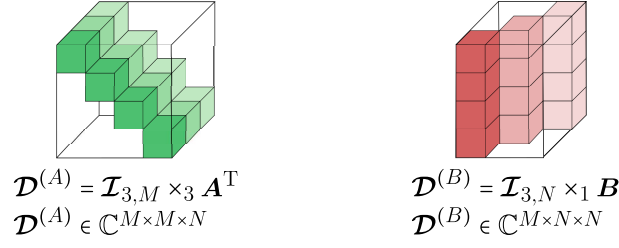


Figure 2.8.: A visualization of the diagonal structure of the tensors $\mathcal{D}^{(A)} = \mathcal{I}_{3,M} \times_3 \mathbf{A}^T \in \mathbb{C}^{M \times M \times N}$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B} \in \mathbb{C}^{M \times N \times N}$, for $M = 4$ and $N = 3$.

Slice-wise multiplication via generalized tensor contraction

A slice-wise multiplication between two tensors $\mathcal{A} \in \mathbb{C}^{M \times N \times K}$ and $\mathcal{B} \in \mathbb{C}^{N \times J \times K}$ is defined as $\mathcal{T}_{1(.,.,k)} = \mathcal{A}_{(.,.,k)} \mathcal{B}_{(.,.,k)}$, $\forall k = 1, \dots, K$. We depict this slice-wise multiplication in Fig. 2.9. To express this slice-wise multiplication we can diagonalize \mathcal{B} to obtain

$$\mathcal{T}_1 = \mathcal{A} \bullet_{2,3}^{1,4} \mathcal{D}_B \in \mathbb{C}^{M \times J \times K},$$

where $\mathcal{D}_B \in \mathbb{C}^{N \times J \times K \times K}$ has non-zero elements $\mathcal{D}_{B(n,j,k,k)} = \mathcal{B}_{(n,j,k)}$ or $\mathcal{D}_{B(n,j,.,.)} = \text{diag}(\mathcal{B}_{(n,j,.)})$, for $n = 1, \dots, N$ and $j = 1, \dots, J$. Further combinations are also possible that lead to the same result, for instance, $\mathcal{T}_2 = \mathcal{D}_B \bullet_{1,4}^{2,3} \mathcal{A} \in \mathbb{C}^{J \times K \times M}$ or $\mathcal{T}_3 = \mathcal{D}_A \bullet_{2,4}^{1,3} \mathcal{B} \in \mathbb{C}^{M \times K \times J}$ with $\mathcal{D}_{A(m,n,k,k)} = \mathcal{A}_{(m,n,k)}$ as diagonal elements (non-zero elements of \mathcal{D}_A). Note that the tensors \mathcal{T}_1 , \mathcal{T}_2 , and \mathcal{T}_3 contain the same elements, but have permuted dimensions. However, the permuted order of the dimensions is not relevant, because we always explicitly declare which dimension is multiplied or unfolded.

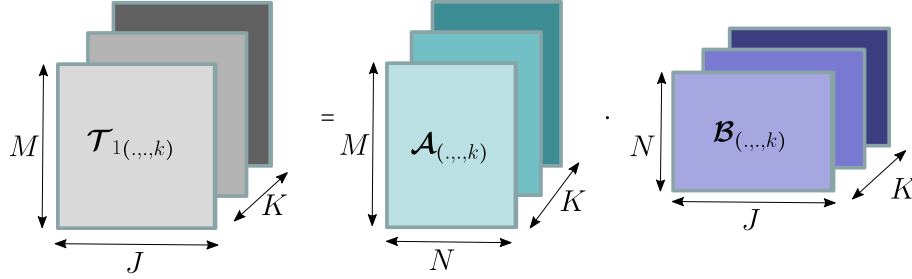


Figure 2.9.: A slice-wise multiplication between two tensors $\mathcal{A} \in \mathbb{C}^{M \times N \times K}$ and $\mathcal{B} \in \mathbb{C}^{N \times J \times K}$, $\mathcal{T}_{1(.,.,k)} = \mathcal{A}_{(.,.,k)} \mathcal{B}_{(.,.,k)}$, $\forall k = 1, \dots, K$.

Representation of diagonal matrices and diagonal tensors in terms of Khatri-Rao products

An explicit expression of the diagonalized tensor can be obtained by expressing its generalized unfolding in terms of a Khatri-Rao product. First, let us consider the column vector $\mathbf{a} \in \mathbb{C}^M$. It can be easily shown that

$$\text{diag}(\mathbf{a}) = \mathbf{I}_M \diamond \mathbf{a}^T.$$

Now, let us consider the reshaping of the matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ into a diagonal tensor $\mathcal{D}^{(A)} = \mathcal{I}_{3,M \times 3} \mathbf{A}^T$ (see Fig. 2.8). By studying the resulting tensor structure, the tensor unfoldings, and the properties of the Khatri-Rao product, we get

$$[\mathcal{D}^{(A)}]_{([3,2],[1])} = \mathbf{I}_M \diamond \mathbf{A}^T.$$

Likewise, for the tensor $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B} \in \mathbb{C}^{M \times N \times N}$ and the matrix $\mathbf{B} \in \mathbb{C}^{M \times N}$, we have $[\mathcal{D}^{(B)}]_{([1,3],[2])} = \mathbf{I}_N \diamond \mathbf{B}$. In Fig. 2.8, we show the diagonal structure of both tensors $\mathcal{D}^{(A)}$ and $\mathcal{D}^{(B)}$. Moreover, in Fig. 2.10 we illustrate the block diagonal structure of the tensors' unfoldings.

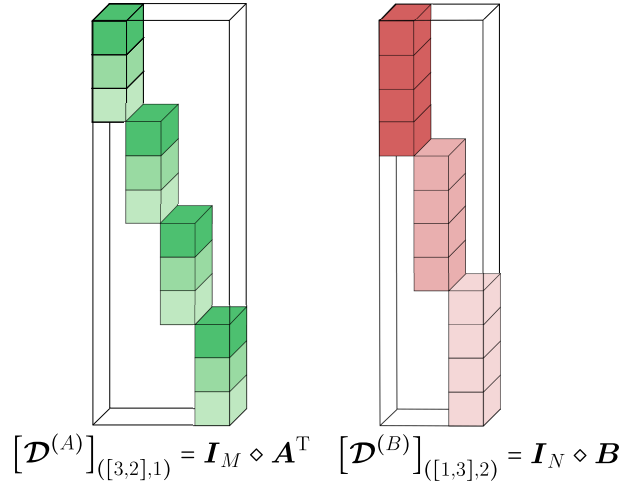


Figure 2.10.: A visualization of the block diagonal structure of the unfoldings for the tensors $\mathcal{D}^{(A)} = \mathcal{I}_{3,M} \times_3 \mathbf{A}^T \in \mathbb{C}^{M \times M \times N}$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,N} \times_1 \mathbf{B} \in \mathbb{C}^{M \times N \times N}$, for $M = 4$ and $N = 3$.

dimensions	non-zero elements	generalized unfoldings
$\mathbf{a} \in \mathbb{C}^{M \times 1}, \mathbf{D} \in \mathbb{C}^{M \times M}$	$\mathcal{D}_{(m,m)} = \mathbf{a}_{(m)}$	$\mathbf{D} = \mathbf{I}_M \diamond \mathbf{a}^T$
$\mathbf{A} \in \mathbb{C}^{M \times N}, \mathcal{D} \in \mathbb{C}^{M \times N \times N}$	$\mathcal{D}_{(m,n,n)} = \mathbf{A}_{(m,n)}$	$[\mathcal{D}]_{([1,3],[2])} = \mathbf{I}_N \diamond \mathbf{A}$
$\mathbf{A} \in \mathbb{C}^{M \times N}, \mathcal{D} \in \mathbb{C}^{M \times M \times N}$	$\mathcal{D}_{(m,m,n)} = \mathbf{A}_{(m,n)}$	$[\mathcal{D}]_{([3,2],[1])} = \mathbf{I}_M \diamond \mathbf{A}^T$
$\mathbf{A} \in \mathbb{C}^{M \times N}, \mathcal{D} \in \mathbb{C}^{M \times M \times N \times N}$	$\mathcal{D}_{(m,m,n,n)} = \mathbf{A}_{(m,n)}$	$[\mathcal{D}]_{([1,3],[2,4])} = \mathbf{I}_M \diamond \text{vec}(\mathbf{A})^T$
$\mathcal{A} \in \mathbb{C}^{M \times N \times K}, \mathcal{D} \in \mathbb{C}^{M \times N \times K \times K}$	$\mathcal{D}_{(m,n,k,k)} = \mathbf{A}_{(m,n,k)}$	$[\mathcal{D}]_{([1,2,4],[3])} = \mathbf{I}_K \diamond [\mathcal{A}]_{([1,2],[3])}$
$\mathcal{A} \in \mathbb{C}^{M \times N \times K}, \mathcal{D} \in \mathbb{C}^{M \times M \times N \times K}$	$\mathcal{D}_{(m,m,n,k)} = \mathbf{A}_{(m,n,k)}$	$[\mathcal{D}]_{([3,4,2],[1])} = \mathbf{I}_M \diamond [\mathcal{A}]_{([2,3],[1])}$

Table 2.1.: Link between the diagonalized tensor structures and their generalized unfoldings.

The expression of the diagonalized tensor in terms of its generalized unfolding and the Khatri-Rao product can also be obtained for N -way tensors. Hence, there exists a link between the diagonalized tensor structures and their generalized unfoldings. This generalized unfolding can always be expressed as a Khatri-Rao product between an identity matrix and a generalized unfolding of the tensor to be diagonalized, where the dimensions that are diagonalized are in the columns of the second matrix. This notation will be used later in this thesis and it is illustrated in Table 2.1.

The element-wise or slice-wise multiplication between two arrays (vectors/matrices/tensors) of the same order can be written in terms of a contraction if the unaffected mode vectors are transformed into a diagonal matrix (by adding an additional array dimension). This diagonalization can be performed using the Khatri-Rao product as shown in Table 2.1.

2.2 Tensor Decompositions

Tensor decompositions factorize a given tensor into a core tensor and factor matrices. Depending on the structure of the core tensor, the factor matrices, the number of core tensors, or the number of the factor matrices, different tensor decompositions/factorizations are defined. Many references, such as [KB09,DLDMV00b,CMP⁺15,CLdA09], and [SDLF⁺17] provide an overview of the different tensor decompositions, their properties, and applications. In this section, we review the basic tensor decompositions relevant for this thesis. These tensor decompositions include the HOSVD, the CP decomposition, the BTd, the PARATUCK2 decomposition, and the PARAFAC2 decomposition.

We can differentiate between an exact tensor decomposition and an approximate low-rank tensor decomposition. For instance, let us consider a noisy observation of a low-rank tensor $\mathcal{X}_0 \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ ¹ given as

$$\mathcal{X} = \mathcal{X}_0 + \mathcal{N} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}, \quad (2.34)$$

where $\mathcal{N} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ is a noise tensor. For the noisy tensor \mathcal{X} we can compute an exact tensor decomposition such as the HOSVD or the BTd. However, we can compute an approximate low-rank tensor decomposition² for the noisy tensor \mathcal{X} . Depending on the statistical properties of the noise tensor \mathcal{N} , we can define different cost functions for the estimation of the low-rank tensor \mathcal{X}_0 based on equation (2.34). The different cost functions lead to different algorithms for the computation of the low-rank tensor decompositions. In this thesis, we consider ZMCSCG noise and an estimation of the low-rank tensor in an LS sense.

¹ By low-rank tensor we mean that the tensor \mathcal{X}_0 satisfies a low-rank tensor model such as the economy size HOSVD, the CP decomposition, the BTd, the PARATUCK2 decomposition, or the PARAFAC2 decomposition. Moreover, we assume that the number of the underlying components is smaller than the maximum of the tensor dimensions.

² In case of a noisy observation, we can compute an approximation of any of the low-rank tensor decompositions including the truncated HOSVD, the CP decomposition, the BTd, the PARATUCK2 decomposition, and the PARAFAC2 decomposition. The particular structure of the low-rank tensor decomposition depends on the structure of the noiseless tensor \mathcal{X}_0 . Note that we consider the BTd as an exact tensor decomposition and as a low-rank tensor decomposition. This is due to the fact that BTd is a generalization of the Tucker and the CP decomposition.

2.2.1 Higher-Order Singular Value Decomposition

Noiseless data model

The Tucker tensor decomposition [Tuc63] factorizes a given tensor $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ into a core tensor $\mathcal{H} \in \mathbb{C}^{r_1 \times r_2 \times \dots \times r_N}$ and factor matrices $\mathbf{G}_n \in \mathbb{C}^{I_n \times r_n}$, $\forall n = 1, \dots, N$.

$$\mathcal{X} = \mathcal{H} \times_1 \mathbf{G}_1 \times_2 \mathbf{G}_2 \dots \times_N \mathbf{G}_N$$

This tensor decomposition does not consider any constraints on the factors or the core tensor. Therefore, it is the most general tensor decomposition. By considering orthogonality constraints for the factor matrices and the core tensor, the Tucker decomposition is transformed into the HOSVD. The HOSVD is also known as multi-linear SVD [DLDMV00b]. However, many authors refer to the HOSVD as the Tucker decomposition [KB09, CMP⁺15]. Moreover, the HOSVD represents an extension of the matrix SVD to multi-dimensional arrays/tensors. For a tensor $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$, we get [DLDMV00b]

$$\mathcal{X} = \mathcal{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \dots \times_N \mathbf{U}_N = \mathcal{S} \times_{n=1}^N \mathbf{U}_n.$$

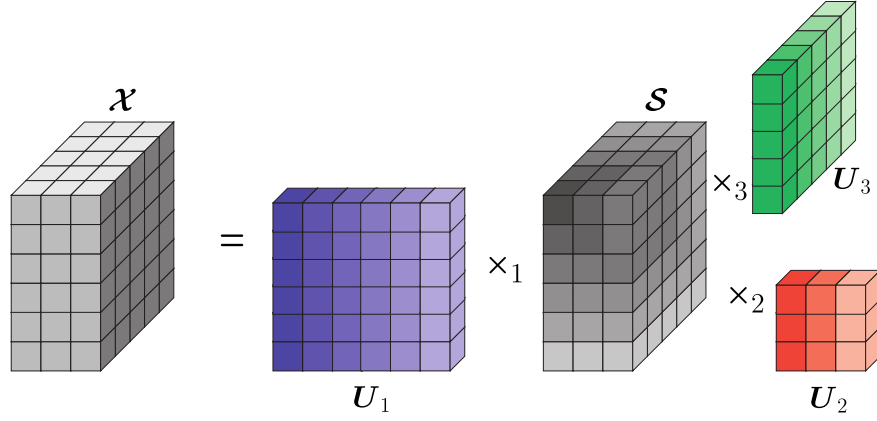
The core tensor $\mathcal{S} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ has the property of all-orthogonality. In other words, the subtensors $\mathcal{S}_{i_n=\alpha} = \mathcal{S}_{(\dots, \alpha, \dots)} \in \mathbb{C}^{I_1 \times \dots \times I_{n-1} \times 1 \times I_{n+1} \times \dots \times I_N}$ obtained by fixing the n -th index to α and $\mathcal{S}_{i_n=\beta} = \mathcal{S}_{(\dots, \beta, \dots)} \in \mathbb{C}^{I_1 \times \dots \times I_{n-1} \times 1 \times I_{n+1} \times \dots \times I_N}$ are orthogonal for all possible values of n , α , and β provided that $\alpha \neq \beta$, i.e., $\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0$, where $\alpha \neq \beta$, $\forall n, \alpha, \beta$, and $\langle \cdot \rangle$ denotes the scalar product. The scalar product between two tensors is defined in (A.2). Moreover, the subtensors are ordered in a decreasing order of their higher-order norm, i.e., $\|\mathcal{S}_{i_n=1}\|_{\text{H}} \geq \|\mathcal{S}_{i_n=2}\|_{\text{H}} \geq \dots \geq \|\mathcal{S}_{i_n=I_n}\|_{\text{H}} \geq 0$ [DLDMV00b]. The matrices $\mathbf{U}_n \in \mathbb{C}^{I_n \times I_n}$ are unitary matrices and they represent the basis of the n -mode space of \mathcal{X} [DLDMV00b]. Therefore, the loading matrices can be computed as the left singular vectors from the SVDs of the n -mode unfoldings.

$$[\mathcal{X}]_{(n)} = \mathbf{U}_n \mathbf{\Sigma}_n \mathbf{V}_n^{\text{H}} \in \mathbb{C}^{I_n \times I_{\bar{n}}}, \quad I_{\bar{n}} = I_1 \cdot I_2 \dots I_{n-1} \cdot I_{n+1} \dots I_N \quad (2.35)$$

The matrices $\mathbf{\Sigma}_n \in \mathbb{R}^{I_n \times I_{\bar{n}}}$ and $\mathbf{V}_n \in \mathbb{R}^{I_{\bar{n}} \times I_{\bar{n}}}$ correspond to the singular values and the right singular vectors of the matrix $[\mathcal{X}]_{(n)}$, respectively.

Exploiting the unitary property of the matrices $\mathbf{U}_n, \forall n = 1, \dots, N$, the core tensor can be calculated as $\mathcal{S} = \mathcal{X} \times_{n=1}^N \mathbf{U}_n^{\text{H}}$. Moreover, the n -mode unfolding of the core tensor \mathcal{S} is related to the singular values of the n -mode unfolding of \mathcal{X} , such that $[\mathcal{S}]_{(n)} \cdot [\mathcal{S}]_{(n)}^{\text{H}} = \mathbf{\Sigma}_n \mathbf{\Sigma}_n^{\text{T}} \in \mathbb{R}^{I_n \times I_n}$. The HOSVD of a 3-way tensor is depicted in Fig. 2.11.

The rank of the n -mode unfolding indicates the n -rank of the tensor \mathcal{X} , i.e., $n\text{-rank}(\mathcal{X}) = \text{rank}([\mathcal{X}]_{(n)}) \leq I_n$. The n -ranks of a tensor are not necessarily equal to one another. If a tensor $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ has n -ranks (multi-linear ranks) smaller than the corresponding dimension,

Figure 2.11.: HOSVD of a 3-way tensor $\mathcal{X} = \mathcal{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$

$d_n = n\text{-rank}(\mathcal{X}) < I_n$, we can define an economy size HOSVD. For the economy size HOSVD, we have

$$\mathcal{X} = \mathcal{S}^{[s]} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s]} \dots \times_N \mathbf{U}_N^{[s]}. \quad (2.36)$$

The loading matrices $\mathbf{U}_n^{[s]} \in \mathbb{C}^{I_n \times d_n}$ contain only the first d_n columns of the matrices \mathbf{U}_n in equation (2.35). For the corresponding core tensor $\mathcal{S}^{[s]} \in \mathbb{C}^{d_1 \times d_2 \times \dots \times d_N}$, we get $\mathcal{S}^{[s]} = \mathcal{X} \times_{n=1}^N \mathbf{U}_n^{[s]H}$.

Noise corrupted data model

Let us assume a noise corrupted signal tensor $\mathcal{X} = \mathcal{X}_0 + \mathcal{N} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$, where \mathcal{X}_0 is the noiseless signal tensor and \mathcal{N} is the noise tensor. The SVD of the n -mode unfolding of the noise corrupted tensor \mathcal{X} is given by

$$[\mathcal{X}]_{(n)} = \underbrace{\begin{bmatrix} \hat{\mathbf{U}}_n^{[s]} & \hat{\mathbf{U}}_n^{[n]} \end{bmatrix}}_{\mathbf{U}_n} \underbrace{\begin{bmatrix} \hat{\Sigma}_n^{[s]} & \mathbf{0}_{d_n \times (I_{\bar{n}} - d_n)} \\ \mathbf{0}_{(I_n - d_n) \times d_n} & \hat{\Sigma}_n^{[n]} \end{bmatrix}}_{\Sigma_n} \underbrace{\begin{bmatrix} \hat{\mathbf{V}}_n^{[s]H} \\ \hat{\mathbf{V}}_n^{[n]H} \end{bmatrix}}_{\mathbf{V}_n^H}, \quad I_{\bar{n}} = I_1 \cdot I_2 \dots I_{n-1} \cdot I_{n+1} \dots I_N,$$

where $\hat{\mathbf{U}}_n^{[s]} \in \mathbb{C}^{I_n \times d_n}$, $\hat{\mathbf{U}}_n^{[n]} \in \mathbb{C}^{I_n \times (I_n - d_n)}$, $\hat{\Sigma}_n^{[s]} \in \mathbb{R}^{d_n \times d_n}$, $\hat{\Sigma}_n^{[n]} \in \mathbb{R}^{(I_n - d_n) \times (I_n - d_n)}$, $\hat{\mathbf{V}}_n^{[s]} \in \mathbb{C}^{I_{\bar{n}} \times d_n}$, and $\hat{\mathbf{V}}_n^{[n]} \in \mathbb{C}^{I_{\bar{n}} \times (I_{\bar{n}} - d_n)}$. We symbolize the unitary bases corresponding to the signal components by the superscript $(.)^{[s]}$ and the ones corresponding to the noise components by the superscript $(.)^{[n]}$. Moreover, the matrix $\mathbf{0}_{M \times L}$ denotes a matrix of zeros with dimensions $M \times L$. The $d_n = n\text{-rank}(\mathcal{X}_0) \leq I_n$ are the corresponding n -ranks of the noiseless signal tensor \mathcal{X}_0 . Furthermore, the matrices $\hat{\mathbf{U}}_n^{[s]} \in \mathbb{C}^{I_n \times d_n}$ and $\hat{\mathbf{U}}_n^{[n]} \in \mathbb{C}^{I_n \times (I_n - d_n)}$ represent the estimated bases for the n -mode signal space and its orthogonal complement, respectively. For the noiseless signal tensor, we can

define a truncated HOSVD in the following way

$$\mathbf{x}_0 \approx \hat{\mathbf{S}}^{[s]} \times_{n=1}^N \hat{\mathbf{U}}_n^{[s]}, \quad (2.37)$$

where the corresponding truncated core tensor $\hat{\mathbf{S}}^{[s]} \in \mathbb{C}^{d_1 \times d_2 \times \dots \times d_N}$ is computed as $\hat{\mathbf{S}}^{[s]} = \mathbf{x} \times_{n=1}^N \hat{\mathbf{U}}_n^{[s]H}$. Hence, the truncated HOSVD is a practical tool for noise suppression and estimation of the signal subspace.

Model order estimation

In order to compute the truncated HOSVD or to perform a denoising of a noisy tensor, the n -ranks of the noiseless tensor (d_n) should be known. Therefore, a multi-linear rank estimation from a noisy observation is a very important task in tensor based signal processing and data analytics. Naturally, the methods for the rank estimation of a matrix can be extended to tensors by using the n -mode unfoldings. For instance, the MDL (Minimum Description Length) model order estimation method proposed in [Ris78] can be used for each of the tensor unfoldings. Let us assume the noisy observation of the tensor $\mathbf{x}_0 \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ given by $\mathbf{x} = \mathbf{x}_0 + \mathbf{N} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$, where the elements of the noise tensor \mathbf{N} are i.i.d. ZMCSCG distributed with variance σ_N^2 . For the n -mode unfolding, we have $[\mathbf{x}]_{(n)} = [\mathbf{x}_0]_{(n)} + [\mathbf{N}]_{(n)}$. The covariance matrix $\mathbf{R}_n = \mathbb{E} \left\{ [\mathbf{x}]_{(n)} [\mathbf{x}]_{(n)}^H \right\} \in \mathbb{C}^{I_n \times I_n}$ is also equal to $\mathbf{R}_n = \mathbf{R}_{0,n} + \sigma_N^2 \cdot \mathbf{I}_{I_n}$, where $\mathbf{R}_{0,n} = \mathbb{E} \left\{ [\mathbf{x}_0]_{(n)} [\mathbf{x}_0]_{(n)}^H \right\} \in \mathbb{C}^{I_n \times I_n}$. Therefore, the eigenvalues of \mathbf{R}_n are

$$\lambda_i = \begin{cases} \lambda_{0,i} + \sigma_N^2, & \text{if } 1 \leq i \leq d_n \\ \sigma_N^2, & \text{if } d_n + 1 \leq i \leq I_n \end{cases},$$

where d_n is the n -rank of \mathbf{x}_0 and the eigenvalues $\lambda_{0,i}$ correspond to the eigenvalues of $\mathbf{R}_{0,n}$. If these eigenvalues are plotted in a decreasing order, the curve has an “L” shape. In practice, we only have an estimate of the covariance matrix, $\hat{\mathbf{R}}_n = \frac{1}{I_{\bar{n}}} [\mathbf{x}]_{(n)} [\mathbf{x}]_{(n)}^H$ with $I_{\bar{n}} = I_1 \cdot I_2 \dots I_{n-1} \cdot I_{n+1} \dots I_N$. Therefore, we can compute the estimated eigenvalues. To this end, d_n can be estimated based on the MDL criterion as [Ris78, YLC17, dCHRDG07]

$$\hat{d}_n = \arg \min_r -2I_{\bar{n}}(I_n - r) \log \left\{ \frac{\prod_{i=r+1}^{I_n} \lambda_i^{1/(I_n-r)}}{\frac{1}{I_n-r} \sum_{i=r+1}^{I_n} \lambda_i} \right\} + r(2I_n - r) \log I_{\bar{n}}.$$

Other methods exist for model order estimation of tensors. Such methods include the evaluation of an “L”-shaped curve [QBLH06, VDS⁺16] or the MEET (Modified Eigenvalues Estimator for Tucker rank determination) algorithm and the SCORE algorithm [YLC17]. The MEET and the SCORE algorithms exploit the HOSVD decomposition and therefore they are more robust against noise than the matrix based solutions. Moreover, robust and enhanced methods for model order estimation for N -way tensors that exploit the tensor structure are also proposed in [dCHRDG07, dCRHdS13]

and [CH13].

Computation of the HOSVD

As previously mentioned, the loading matrices of the HOSVD are easily computed from the left singular vectors of the tensor unfoldings. Afterwards, the core tensor is estimated using the loading matrices [DLDMV00b, KB09]. The truncated HOSVD can be computed from the HOSVD by means of truncation, i.e., by taking into account only the first d_n columns of the loading matrices \mathbf{U}_n leading to $\hat{\mathbf{U}}_n^{[s]} \in \mathbb{C}^{I_n \times d_n}$ and computing the truncated core tensor as explained after equation (2.37). However, the truncated HOSVD is not the optimal low rank approximation in the Frobenius norm sense. The accuracy can be increased with iterative algorithms such as the algorithm proposed in [DLDMV00a]. This HOOI (Higher-order Orthogonal Iteration) algorithm computes the left singular vectors of a representative matrix of the corresponding unfolding, instead of the actual tensor unfolding. For time-varying applications, once a previous estimate of the HOSVD is available, an efficient update of the decomposition can be computed according to [MSK09]. Moreover, in [YFLZ16] an iterative method for the computation of the Tucker decomposition for tensors with missing entries is proposed. This method is also capable of determining the multi-linear ranks and it is initialized using the HOSVD. The HOSVD with n -rank = 1, $\forall n = 1, \dots, N$ is equivalent to the rank one CP decomposition. Therefore, for rank one tensors the efficient estimate of the factors presented in [dSCdA15b] can be used to compute the HOSVD, even though it was originally developed for the computation of the CP decomposition.

Coupled truncated HOSVD

Coupled tensors are tensors that have at least one mode in common. Such tensors have a coupled HOSVD decomposition. In the noisy case, an approximate coupled truncated HOSVD can be computed. This coupled truncated HOSVD for two noise corrupted tensors $\mathbf{X}^{(1)} \in \mathbb{C}^{M_1 \times M_2^{(1)} \times M_3^{(1)}}$ and $\mathbf{X}^{(2)} \in \mathbb{C}^{M_1 \times M_2^{(2)} \times M_3^{(2)}}$ that have the mode one in common is given by

$$\mathbf{X}^{(1)} = \mathbf{X}_0^{(1)} + \mathcal{N}^{(1)} \approx \hat{\mathbf{S}}^{[s],(1)} \times_1 \hat{\mathbf{U}}_1^{[s]} \times_2 \hat{\mathbf{U}}_2^{[s],(1)} \times_3 \hat{\mathbf{U}}_3^{[s],(1)} \quad (2.38)$$

$$\mathbf{X}^{(2)} = \mathbf{X}_0^{(2)} + \mathcal{N}^{(2)} \approx \hat{\mathbf{S}}^{[s],(2)} \times_1 \hat{\mathbf{U}}_1^{[s]} \times_2 \hat{\mathbf{U}}_2^{[s],(2)} \times_3 \hat{\mathbf{U}}_3^{[s],(2)}. \quad (2.39)$$

In (2.38) and (2.39), it is assumed that all multi-linear ranks for both noiseless tensors $\mathbf{X}_0^{(1)}$ and $\mathbf{X}_0^{(2)}$ are equal to R , where R is less or equal than the minimum of the tensor dimensions. The tensors $\hat{\mathbf{S}}^{[s],(1)} \in \mathbb{C}^{R \times R \times R}$ and $\hat{\mathbf{S}}^{[s],(2)} \in \mathbb{C}^{R \times R \times R}$ are the truncated core tensors and the loading matrices $\hat{\mathbf{U}}_1^{[s]} \in \mathbb{C}^{M_1 \times R}$, $\hat{\mathbf{U}}_2^{[s],(i)} \in \mathbb{C}^{M_2^{(i)} \times R}$ and $\hat{\mathbf{U}}_3^{[s],(i)} \in \mathbb{C}^{M_3^{(i)} \times R}$ have unitary columns and span the column space of the n -mode unfolding of $\mathbf{X}_0^{(i)}$, for $n = 1, 2, 3$ and $i = 1, 2$, respectively. Note that the matrix $\hat{\mathbf{U}}_1^{[s]}$ spans the column space of the 1-mode unfolding of the tensors $\mathbf{X}_0^{(1)}$ and $\mathbf{X}_0^{(2)}$.

The common factor matrices corresponding to the truncated HOSVD of the tensors $\mathbf{X}^{(1)}$ and

$\mathcal{X}^{(2)}$ can be approximated jointly, using the economy size SVD,

$$\begin{bmatrix} [\mathcal{X}^{(1)}]_{(1)} & [\mathcal{X}^{(2)}]_{(1)} \end{bmatrix} \approx \hat{\mathbf{U}}_1^{[s]} \cdot \hat{\mathbf{\Sigma}}_1^{[s]} \cdot \hat{\mathbf{V}}_1^{[s]H} \in \mathbb{C}^{M_1 \times M_2^{(1)} M_3^{(1)} + M_2^{(2)} M_3^{(2)}}.$$

The remaining matrices $\hat{\mathbf{U}}_2^{[s],(i)}$ and $\hat{\mathbf{U}}_3^{[s],(i)}$ are computed separately, using the economy size SVD of each of the unfoldings, i.e., $[\mathcal{X}^{(i)}]_{(n)} \approx \hat{\mathbf{U}}_n^{[s],(i)} \cdot \hat{\mathbf{\Sigma}}_n^{[s],(i)} \cdot \hat{\mathbf{V}}_n^{[s],(i)H}$, for $n = 2, 3$ and $i = 1, 2$. Moreover, the core tensors are also computed separately, $\hat{\mathcal{S}}^{[s],(1)} = \mathcal{X}^{(1)} \times_1 \hat{\mathbf{U}}_1^{[s]H} \times_2 \hat{\mathbf{U}}_2^{[s],(1)H} \times_3 \hat{\mathbf{U}}_3^{[s],(1)H}$ and $\hat{\mathcal{S}}^{[s],(2)} = \mathcal{X}^{(2)} \times_1 \hat{\mathbf{U}}_1^{[s]H} \times_2 \hat{\mathbf{U}}_2^{[s],(2)H} \times_3 \hat{\mathbf{U}}_3^{[s],(2)H}$.

Applications of the truncated HOSVD

The truncated HOSVD is a practical tool for dimensions reduction, signal subspace estimation, and noise suppression. For instance, we can use the truncated HOSVD as an initial step when computing the CP decomposition [RH08, RSH12, RH13a]. Thus, we use the truncated HOSVD to reduce the problem dimensionality and to suppress noise. For a tensor with missing entries the algorithm proposed in [YFLZ16] can be used as an alternative initial step for the SECSI framework instead of the HOSVD. Tensor based subspace estimation using the truncated HOSVD is performed in [HRDG08, RH13b, CRKH14, SDLF⁺17]. Moreover, the truncated HOSVD has an application in time-varying multi-dimensional harmonic retrieval for source separation [CRKH14, RH13b]. Many applications in compressed sensing, such as the recovery of signals from compressed measurements for MRI signals, hyper-spectral imaging, deblurring via image filtering, and tensor completion problems exploit the truncated HOSVD [CC13b, CC13a]. The truncated HOSVD can be utilized in video processing for tracking motion trajectories [MSK09] and image denoising [YLC17]. Furthermore, it can be used in speech or text processing applications for topic modeling [SDLF⁺17] or for chemical analysis, psychometrics [CC70], and many other applications [KB09].

2.2.2 Canonical Polyadic Decomposition

The CP decomposition and tensor rank

In addition to the HOSVD, the CP decomposition is another extension of the SVD to multi-linear arrays. Some authors refer to the CP decomposition as PARAFAC (Parallel Factors) analysis, CAN-DECOMP (CANonical DECOMPosition), or CAND (CANonical Decomposition) [CC70, KB09]. It decomposes a tensor into a sum of the minimum number R of rank one components. The CP decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ is defined as [KB09, CLdA09, CMP⁺15, Kru77]

$$\mathcal{X} = \sum_{r=1}^R \mathbf{f}_1^{(r)} \circ \mathbf{f}_2^{(r)} \circ \mathbf{f}_3^{(r)} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3, \quad (2.40)$$

where the column vectors $\mathbf{f}_1^{(r)}$, $\mathbf{f}_2^{(r)}$, and $\mathbf{f}_3^{(r)}$ represent the r -th columns of the matrices $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2 \in \mathbb{C}^{M_2 \times R}$, and $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$, respectively. The tensor rank is the minimum number of rank one components that generate \mathcal{X} . According to equation (2.40) the tensor rank equals R . The CP decomposition for a 3-way tensor with rank $R = 3$ is visualized in Fig. 2.12. Note that the factor matrices are not unitary, meaning that the R underlying parallel components are not necessarily mutually orthogonal. Therefore, the CP decomposition is a very practical tool for many applications.

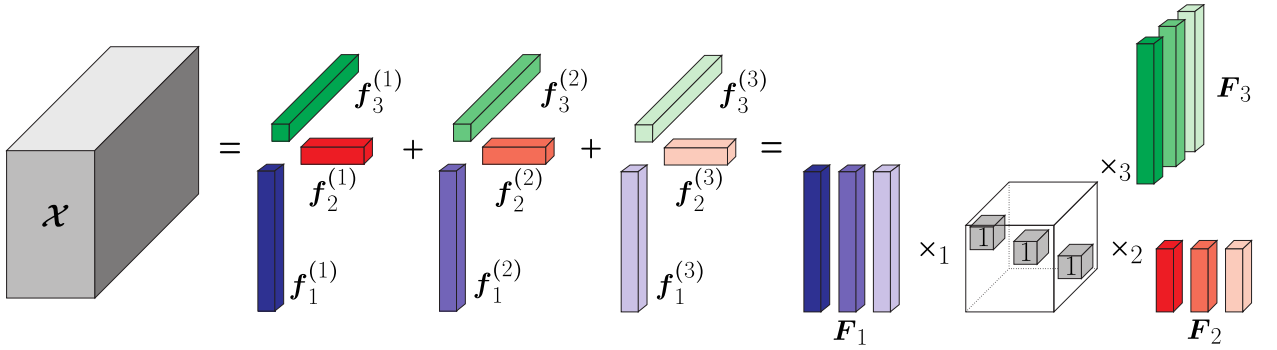


Figure 2.12.: CP decomposition of a 3-way tensor with rank $R = 3$.

The Kruskal rank of a matrix \mathbf{F} is the largest number k such that any combination of k columns of the matrix \mathbf{F} are linearly independent [Kru77]. We denote the Kruskal rank by k -rank. A sufficient condition for the uniqueness of the CP decomposition can be derived using the Kruskal rank [Kru77, Kru89, Com14]. This sufficient condition for the uniqueness of the CP decomposition for 3-way tensors is given by [Kru77, Kru89, Com14]

$$k\text{-rank}(\mathbf{F}_1) + k\text{-rank}(\mathbf{F}_2) + k\text{-rank}(\mathbf{F}_3) \leq 2R + 2. \quad (2.41)$$

Using the properties of the Kronecker and Khatri-Rao product from Section 2.1.2 equation (2.29), for the unfoldings defined in equation (2.4) it is easy to show that

$$[\mathcal{X}]_{(1)} = \mathbf{F}_1 \cdot (\mathbf{F}_3 \diamond \mathbf{F}_2)^T, \quad [\mathcal{X}]_{(2)} = \mathbf{F}_2 \cdot (\mathbf{F}_3 \diamond \mathbf{F}_1)^T, \quad \text{and} \quad [\mathcal{X}]_{(3)} = \mathbf{F}_3 \cdot (\mathbf{F}_2 \diamond \mathbf{F}_1)^T. \quad (2.42)$$

The extension of the CP decomposition to N -way tensors is straightforward. For instance, for a 4-way tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3 \times M_4}$ with rank R , we have

$$\mathcal{X} = \mathcal{I}_{4,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3 \times_4 \mathbf{F}_4,$$

where $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2 \in \mathbb{C}^{M_2 \times R}$, $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$, and $\mathbf{F}_4 \in \mathbb{C}^{M_4 \times R}$. For the unfoldings of the 4-way

tensor, we get

$$\begin{aligned} [\mathcal{X}]_{(1)} &= \mathbf{F}_1 \cdot (\mathbf{F}_4 \diamond \mathbf{F}_3 \diamond \mathbf{F}_2)^T, & [\mathcal{X}]_{(2)} &= \mathbf{F}_2 \cdot (\mathbf{F}_4 \diamond \mathbf{F}_3 \diamond \mathbf{F}_1)^T, \\ [\mathcal{X}]_{(3)} &= \mathbf{F}_3 \cdot (\mathbf{F}_4 \diamond \mathbf{F}_2 \diamond \mathbf{F}_1)^T, & \text{and} & \quad [\mathcal{X}]_{(4)} = \mathbf{F}_4 \cdot (\mathbf{F}_3 \diamond \mathbf{F}_2 \diamond \mathbf{F}_1)^T. \end{aligned}$$

Similar, for the generalized unfolding $[\mathcal{X}]_{([1,2],[3,4])}$ from equations (2.29) and (2.6), we get [RSH12]

$$[\mathcal{X}]_{([1,2],[3,4])} = (\mathbf{F}_2 \diamond \mathbf{F}_1) \cdot (\mathbf{F}_4 \diamond \mathbf{F}_3)^T.$$

In the same fashion, other generalized unfoldings can be defined. Some of them, their index ordering, and definition in terms of the factor matrices are depicted in Fig. 2.13.

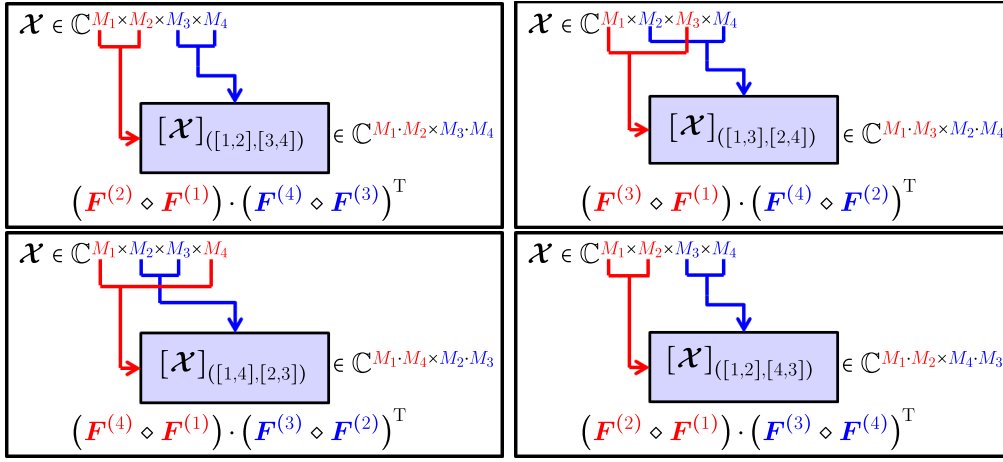


Figure 2.13.: Generalized unfoldings of a 4-way tensor with a CP structure.

The tensor rank is the minimum number of rank one components leading to the exact decomposition of a given low-rank tensor. This corresponds to the definition of the matrix rank for matrices. However, the tensor rank can only be revealed by the CP decomposition and it does not have to equal the n -ranks. The n -ranks defined in Section 2.2.1 correspond to the ranks of the unfoldings, not to the tensor rank. Moreover, a given tensor can have different ranks over \mathbb{C} and \mathbb{R} [CLdA09]. A maximal tensor rank is the maximum achievable rank. It can exceed the tensor dimensions. On the other hand, a typical rank is any rank that appears with a non-zero probability if the tensor elements are drawn randomly from a continuous distribution. Unlike matrix, a tensor can have several typical ranks and these do not always equal the maximal rank. If for a given tensor size there is only one typical rank, this one is called generic rank [CLdA09]. Tensors often have no generic rank over \mathbb{R} , but they always have one over \mathbb{C} [KB09, Kru89, CLdA09]. The maximal rank of a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ is upper bounded by $\min(M_1 M_2, M_1 M_3, M_2 M_3)$ [Kru89]. Some useful results on maximal rank and typical rank of tensors are available in [KB09, SDLF⁺17] and [LC10]. Also, note that the typical rank may change when the tensor is constrained in some way, for instance, if the frontal slabs are symmetric [SDLF⁺17]. The authors in [LS01, SDLF⁺17] proved

the following upper and lower bound for the tensor rank R .

$$\max(R_1, R_2, R_3) \leq R \leq \min(R_1 R_2, R_2 R_3, R_1 R_3)$$

Here, R_1 , R_2 , and R_3 denote the n -ranks ($n = 1, 2, 3$) for a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$.

The CP decomposition is unique up to a permutation and scaling ambiguity under mild conditions, i.e., if the sufficient condition in (2.41) is satisfied. Recognize that equation (2.40) still holds for either

$$\mathcal{X} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \mathbf{P} \times_2 \mathbf{F}_2 \mathbf{P} \times_3 \mathbf{F}_3 \mathbf{P} \quad \text{or} \quad \mathcal{X} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \mathbf{\Lambda}_1 \times_2 \mathbf{F}_2 \mathbf{\Lambda}_2 \times_3 \mathbf{F}_3 \mathbf{\Lambda}_3,$$

provided that the matrix $\mathbf{P} \in \mathbb{R}^{R \times R}$ is a permutation matrix³, and the matrices $\mathbf{\Lambda}_1 \in \mathbb{C}^{R \times R}$, $\mathbf{\Lambda}_2 \in \mathbb{C}^{R \times R}$, and $\mathbf{\Lambda}_3 \in \mathbb{C}^{R \times R}$ are diagonal matrices such that $\mathbf{\Lambda}_1 \mathbf{\Lambda}_2 \mathbf{\Lambda}_3 = \mathbf{I}_R$. Therefore, the factor matrices of a CP decomposition can be identified up to permutation and one complex scaling ambiguity per column. The uniqueness of the CP decomposition is investigated even nowadays. Basic uniqueness results are included in [DDL13a] and [DDL13b]. Further uniqueness results, including two-slab tensors and symmetric cases are available in [SDLF⁺17]. Moreover, uniqueness results for the CP decomposition with correlated factor matrices are available in [LC10].

Computation of the CP decomposition

The computation of the CP decomposition is considered to be an NP-hard problem⁴, but the praxis proves to be easier. However, the computation of the CP decomposition is more expensive than the computation of the HOSVD. There are many different types of algorithms for the computation of the CP decomposition and they can be categorized as follows. In case of noisy observations, we can compute only an approximate low-rank CP decomposition.

- ALS is a very simple iterative algorithm for the computation of the CP decomposition [CC70, Bro97, KB09, SvBDL13, dScdA15b, SDLF⁺17]. In every iteration, a factor matrix is computed from the tensor unfoldings (cf. equation (2.42)) in an LS sense assuming that the other factor matrices are known. The ALS algorithm has no guarantee of convergence and the number of iteration can be very high. Some versions of the ALS algorithm include dimensionality reduction [Bro97], for instance, based on the truncated HOSVD [BSG99, CFC16]. Moreover, it is possible to introduce constraints (e.g., correlation constraints on the factor matrices) to the ALS algorithm in order to reduce the number of iterations [FGC18].
- Enhanced line search algorithms exploit the multi-dimensional structure of the tensor while computing the CP decomposition. The line search based algorithms have an improved accu-

³ A permutation matrix $\mathbf{P} \in \mathbb{R}^{R \times R}$ contains R elements equal to ones and the remaining elements are zeros. Moreover, every row and column contain only a single element equal to one. A permutation matrix is obtained by permuting the rows or the columns of an identity matrix according to some permutation.

⁴ An NP-hard problem is a problem for which there exist no known polynomial algorithm that can solve this problem. Therefore, the complexity to find a solution to this problem grows exponentially with the problem size.

racy, but they are computationally more expensive than ALS [RCH08,SDLF⁺17].

- GD (Gradient descent) methods for the computation of the CP decomposition compute each of the factor matrices from the tensor unfoldings. Unlike ALS, the GD algorithms use a step in the opposite direction of the gradient instead of computing an LS fit. Another option is the usage of CG (Conjugate Gradient) based algorithms, which are faster than the well known gradient descent based algorithms [dSCdA15b]. In case of very large tensors, and/or large rank, using GD can be computationally very expensive. Therefore, as an alternative an SGD (Stochastic Gradient Descent) can be used [SDLF⁺17].
- Quasi-Newton and NLS (Nonlinear Least Squares) methods can also be used for fitting a CP model [SDLF⁺17, SvBDL15, SvBDL13, VDS⁺16]. The Quasi-Newton algorithms are iterative algorithms that compute the update of the factor matrices based on a Newton descent that includes the computation of the gradient and an approximation of the Hessian of the cost function. Similar, the NLS algorithms use Gauss-Newton or Levenberg-Marquardt methods and a linear approximation of the Hessian. Moreover, an efficient algorithm for the computation of the CP decomposition based on damped Gauss-Newton and an inverse approximate Hessian in a block form that can handle collinear factors is proposed in [PTC13].
- A deflation-based computation of the CP decomposition is based on rank one approximations [dSCdA15a, dSCdA15b]. The algorithm proposed in [dSCdA15a, dSCdA15b] computes the rank one components sequentially and computes a residual in an iterative matter. The authors in [dSCdA15b] propose both, an iterative and an algebraic solution for a deflation-based algorithm for rank one tensors. The extension of the deflation-based algorithm to the rank R CP decomposition is provided in [dSCdA15a].
- Semi-algebraic computation and algebraic computation of the CP decomposition involve conversion of the CP model into an SMD (Simultaneous Matrix Diagonalization) and then using diagonalization algorithms in order to obtain the factor matrices [DL05, LA11, LA14, RH13a]. The SECSI (Semi-Algebraic framework for approximate CP decomposition via SImultaneous matrix diagonalization) framework is an efficient tool for the computation of the CP decomposition based on matrix diagonalizations. It calculates all possible SMDs and then selects the best available solution in a final step via different heuristics [RH08, RSH12, RH13a, NHT⁺16]. For N -way ($N > 3$) tensors it is recommended to use the SECSI framework based on generalized unfoldings SECSI-GU (SECSI-Generalized Unfoldings) [RSH12].

Further details regarding the computation of the approximate CP decomposition are provided in Chapter 3, where we also present our contributions to the computation of the CP decomposition.

All of the above mentioned algorithms for the computation of the CP decomposition assume that the tensor rank is known. The estimation of the tensor rank from a noisy observation is an ill-posed problem, even more difficult than the estimation of the n -ranks. Some authors propose to estimate the tensor rank by fitting a CP decomposition for different ranks and choosing the rank that leads to the smallest residual. However, this method is not always reliable. An algorithm for the estimation

of the tensor rank known as CORCONDIA that is based on a core consistency check is proposed in [BK03]. More recently, its extension to sparse tensors was presented in [PF15]. Less complex algorithms for rank estimation are proposed in [HRDG08, dCHRDG07, dCRHdS13, LdCS⁺16] based on the minimum description length applied onto the eigenvalues of the generalized tensor unfoldings. The authors of [HAK⁺17] propose a method that estimates both the tensor rank and the factor matrices of the CP decomposition, by introducing a group sparsity constraint on the factor matrices. The advantage of this method is that it cannot overestimate the tensor rank. Thus, the resulting CP decomposition does not contain artificial components.

A CR-bound (Cramér-Rao-bound) is the lower bound on the variance of an unbiased estimator of a deterministic parameter [Cra46]. The CR-bound of the approximate low-rank CP decomposition for 3-way and 4-way arrays is provided in [LS01, SDLF⁺17]. A CRI-bound (Cramér-Rao-Induced-bound) for the approximate low-rank CP decomposition of a noisy tensor is derived in [TPK13]. This CRI-bound represents the angular error between the estimated and true components.

Symmetry, non-negativity, real-valued, and sparsity constraints for the CP decomposition

Different applications impose different constraints on the CP decomposition [SDLF⁺17]. We can summarize them as follows.

- Symmetry or Hermitian (conjugate) symmetry arises in higher order statistics, higher-order derivatives, and blind source identification. Two symmetric cases can be considered. One, if just two of the tensor modes are symmetric and another, if all modes are symmetric. For instance, for a 3-way tensor with a CP structure $\mathcal{X} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3$, we can have a symmetric case such that $\mathbf{F}_1 = \mathbf{F}_2$, or $\mathbf{F}_1 = \mathbf{F}_2 = \mathbf{F}_3$. A method for the efficient computation of a symmetric CP decomposition is proposed in Section 3.4.
- Non-negativity constraints are often considered in data analytics such as biomedical data, image processing, and blind estimation applications. Also, the power spectral density of signals does not have negative values. The non-negativity constraints can improve the identifiability of a tensor. The CP tensor decomposition with non-negativity constraints is also known as NTF (Non-negative Tensor Factorization) [CZPSI09]. It can be approximated using ADMM (Alternating Direction Method of Multipliers) [LS15]. An extension of the SECSI framework [RH13a] for the computation of a CP decomposition with non-negativity constraints is presented in Section 3.5.
- Real-valued factor matrices are a common constraint in chemistry and communications when dealing with power spectral density. The real valued constraints are easy to incorporate. However, the tensor rank over complex-valued and real-valued fields can be different. Therefore, we have to take into account if the factor matrices are real-valued or complex-valued.
- Sparsity constraints are also applied in data analysis applications for social network data or rating sites data [PFS12] and image processing [CC13a, CC13b]. The authors of [PFS12] propose a ParCube algorithm for the computation of the CP decomposition for big sparse

tensors. Moreover, in [HAK⁺17], the authors propose a computation of the CP decomposition by imposing group sparsity on the factor matrices that does not require a priori knowledge of the tensor rank.

In general, constraints have a positive influence on the computation of an approximate CP decomposition. They can restore the identifiability in otherwise non-identifiable cases [dAFM08, CZPSI09, SDLF⁺17]. The constraints can also improve the accuracy of the estimated factors in ill-posed scenarios like barely identifiable (highly correlated columns of the factor matrices) scenarios, and/or low SNR (Signal to Noise Ratio) scenarios. For instance, the authors of [FGC18] show that the convergence of the ALS approach can be improved by introducing mutual correlation constraints of the modes. Moreover, the constraints ensure a reasonable physical interpretation and physical meaning of the factor matrices. How to impose constraints on the CP decomposition depends on the specific algorithm for the computation of the CP decomposition. In this thesis, we present extensions of the SECSI framework [RH13a] to the robust computation of the approximate low-rank CP decomposition with non-negativity and symmetry constraints. These extensions are presented in Section 3.5 and Section 3.4, respectively. The SECSI framework [RH13a, RSH12, NHT⁺16] already considers real-valued and complex valued factor matrices.

Coupled CP decomposition

Furthermore, two different tensors $\mathcal{X}^{(1)} \in \mathbb{C}^{M_1 \times M_2^{(1)} \times M_3^{(1)}}$ and $\mathcal{X}^{(2)} \in \mathbb{C}^{M_1 \times M_2^{(2)} \times M_3^{(2)}}$ have a coupled CP decomposition if they have at least one common factor matrix. Assuming that the two tensors have the first factor matrix as a common one, the coupled CP decomposition is defined as

$$\begin{aligned}\mathcal{X}^{(1)} &= \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2^{(1)} \times_3 \mathbf{F}_3^{(1)}, \\ \mathcal{X}^{(2)} &= \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2^{(2)} \times_3 \mathbf{F}_3^{(2)},\end{aligned}$$

where the tensor rank of both tensors is equal to R . The factor matrices have dimensions $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2^{(i)} \in \mathbb{C}^{M_2^{(i)} \times R}$, and $\mathbf{F}_3^{(i)} \in \mathbb{C}^{M_3^{(i)} \times R}$, for $i = 1, 2$. In order to jointly decompose the tensors, the existing algorithms for the computation of the CP decomposition have to be modified. The computation of the coupled CP decomposition based on ALS is analyzed in [FCC16] and [CFC16]. An extension of the SECSI framework for the calculation of the coupled CP decomposition is proposed in [NH16] and in Section 3.6.

Applications of the CP decomposition

The CP decomposition has a broad range of applications including diverse branches of signal processing, audio processing, speech processing, biomedical engineering, chemometrics, and machine learning [KB09, CMP⁺15]. The rank one components of the CP decomposition capture the essential properties of complex signals, therefore the CP decomposition is an efficient tool for data analytics [CMP⁺15]. To this end, the coupled CP decomposition is also a valuable tool for data/signal anal-

ysis in biomedical applications [NKHH17, NLA⁺17]. The estimation of emission factors in amino acids fluorescence data or fluorescence spectroscopy can be performed in terms of the CP decomposition [HAK⁺17, LA14]. The CP decomposition is also widely used in wireless communication. For instance, the signals transmitted by different users correspond to rank one components. CP based applications also include harmonic retrieval [CMP⁺15, SDLF⁺17], estimation and tracking of the direction of arrival or the direction of departure [CMP⁺15, NS09], near field localization [PHT⁺18], CDMA (Code-Division Multiple Access) [SGB00], OFDM communications systems type applications [dAF13a, NHdA17, NHdA18], and many more. Moreover, the CP decomposition can be used for the imputation of missing entries [AKDM11, BMG13, SDLF⁺17], the joint identification and extraction of multiple sources from noisy observations [LC10, CL11, SDLF⁺17], and subspace tracking [NS09, NAMLT16]. The CP decomposition can be used for the analysis of rating websites such as movie websites [SDLF⁺17] and for Gaussian mixture parameter estimation [SDLF⁺17]. Furthermore, the CP decomposition has many dictionary based applications like chat or conversation topic modeling from previously defined sets of topics [SDLF⁺17]. The identification of the estimated sources according to a known dictionary can also be achieved using the CP decomposition [CG18]. Finally, data mining and structured data fusion can also benefit from the CP decomposition [SvBDL15, KB09].

2.2.3 Block Term Decomposition

The BTM is a generalization of the CP decomposition and the Tucker decomposition. It decomposes a higher order tensor in block terms of lower ranks [DL08a, DL08b, DLN08]. The BTM and its uniqueness properties are presented in [DL08b]. Depending on the different block term ranks, three different decompositions in block terms are defined, the decomposition in rank- $(L_r, L_r, 1)$ terms, the decomposition in rank- (L, M, N) terms, and the decomposition in rank- $(L, M, .)$ terms [DL08b, DLN08]. In this section, we provide a brief overview of these three decompositions.

Decomposition in rank- $(L_r, L_r, 1)$ terms

The decomposition in R rank- $(L_r, L_r, 1)$ terms of a tensor $\mathcal{T} \in \mathbb{C}^{I \times J \times K}$ is defined as

$$\mathcal{T} = \sum_{r=1}^R (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r, \quad (2.43)$$

where the matrices $\mathbf{A}_r \in \mathbb{C}^{I \times L_r}$ and $\mathbf{B}_r \in \mathbb{C}^{J \times L_r}$ have ranks $L_r = \text{rank}(\mathbf{A}_r) = \text{rank}(\mathbf{B}_r)$ and $\mathbf{c}_r \in \mathbb{C}^K$ is a rank one column vector, $\forall r = 1, \dots, R$ [DL08b, DLN08]. If we define the block matrices $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_R]$, and $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R]$, for the following matrix representations

of the tensor \mathcal{T} , we have

$$\begin{aligned} [\mathcal{T}]_{([2,1],3)} &= [(A_1 \diamond B_1) \cdot \mathbf{1}_{L_1}, \dots, (A_R \diamond B_R) \cdot \mathbf{1}_{L_R}] \cdot C^T, \\ [\mathcal{T}]_{([3,2],1)} &= (B \boxtimes_R C) \cdot A^T, \text{ and} \\ [\mathcal{T}]_{([1,3],2)} &= (C \boxtimes_R A) \cdot B^T. \end{aligned} \quad (2.44)$$

The symbol \boxtimes_R denotes the partition-wise Kronecker product. For the partitioned matrices $A = [A_1, \dots, A_R]$ and $B = [B_1, \dots, B_R]$, the partition-wise Kronecker product between these two matrices is defined as $A \boxtimes_R B = [A_1 \otimes B_1, \dots, A_R \otimes B_R]$ [DL08a, DL08b]. It is obvious that the R terms in (2.43) can be arbitrary permuted and the matrix A_r can be multiplied from the right hand side by a nonsingular matrix $F_r \in \mathbb{C}^{L_r \times L_r}$ provided that the matrix B_r^T is multiplied from the left hand side by F_r^{-1} . The decomposition is essentially unique when it is subject only to these trivial indeterminacies. The mild conditions under which the BTB in rank- $(L_r, L_r, 1)$ terms is essentially unique are proven in [DL08b]. The BTB in rank- $(L_r, L_r, 1)$ terms can be computed based on ALS using the tensor unfoldings in equation (2.44). The corresponding updates are provided in [DLN08].

Decomposition in rank- (L, M, N) terms

The BTB in rank- (L, M, N) terms decomposes a given tensor $\mathcal{T} \in \mathbb{C}^{I \times J \times K}$ into a sum of R rank- (L, M, N) terms [DL08b].

$$\mathcal{T} = \sum_{r=1}^R \mathcal{D}_r \times_1 A_r \times_2 B_r \times_3 C_r \quad (2.45)$$

The tensor $\mathcal{D}_r \in \mathbb{C}^{L \times M \times N}$ has 1-rank, 2-rank, and 3-rank equal to L , M , and N , respectively. Moreover, the matrices $A_r \in \mathbb{C}^{I \times L}$, $B_r \in \mathbb{C}^{J \times M}$, and $C_r \in \mathbb{C}^{K \times N}$ have full column rank. The BTB in rank- (L, M, N) terms is depicted in Fig. 2.14. When $R = 1$, the BTB in rank- (L, M, N) terms is equivalent to the Tucker decomposition (see Section 2.2.1). On the other hand, the BTB in R terms with rank- (L, M, N) , for $L = M = N = 1$ is the equivalent to the CP decomposition (see Section 2.2.2). Therefore, the BTB represents a generalization of the CP decomposition and the Tucker decomposition that offers more flexibility in terms of the model order (in comparison to the multi-linear ranks and the tensor rank of the CP decomposition).

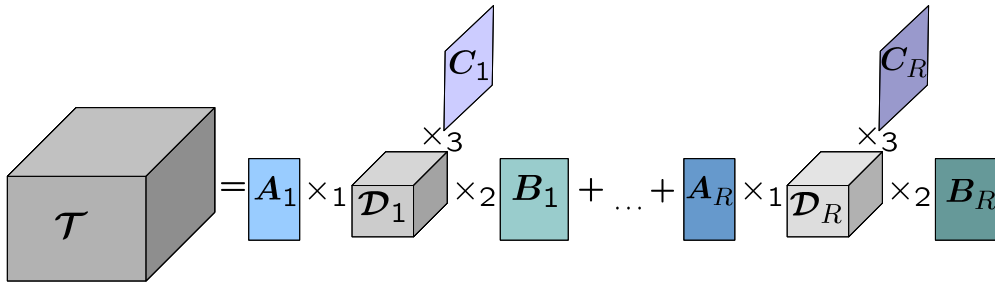
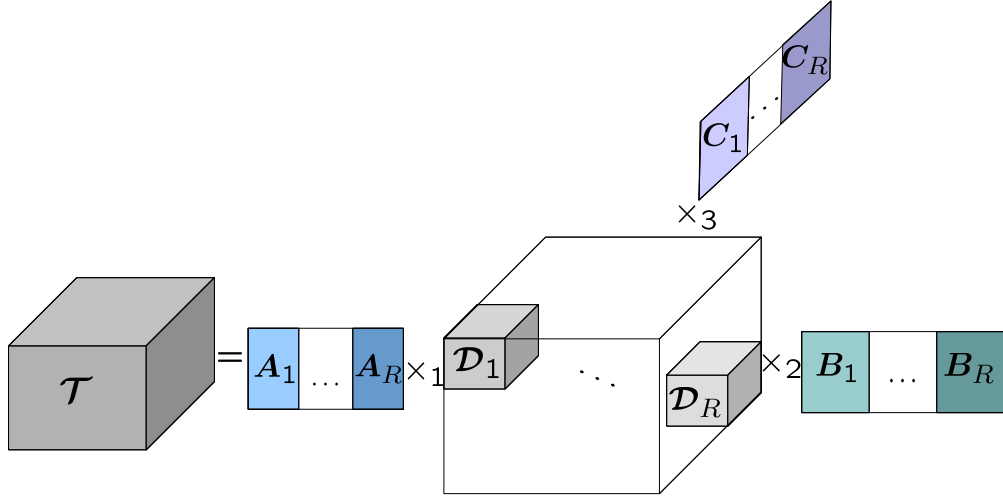


Figure 2.14.: BTB in rank- (L, M, N) terms

Figure 2.15.: Block diagonal structure of the BTM in rank- (L, M, N) terms

Using the partitioned matrices $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_R] \in \mathbb{C}^{I \times LR}$, $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_R] \in \mathbb{C}^{J \times MR}$, $\mathbf{C} = [\mathbf{C}_1, \dots, \mathbf{C}_R] \in \mathbb{C}^{K \times NR}$, and the block diagonal tensor $\mathcal{D} \in \mathbb{C}^{LR \times MR \times NR}$ that is built from the tensors \mathcal{D}_r , the BTM can be rewritten as $\mathcal{T} = \mathcal{D} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$. The BTM in this format and the corresponding partitioned matrices and tensors are depicted in Fig. 2.15. This block diagonal formulation is more compact and practical. Therefore, it can be used to derive any of the tensor unfoldings. Using the unfolding properties (2.4)-(2.7) the following unfoldings of the tensor \mathcal{T} can be derived [DLN08]

$$\begin{aligned}
 [\mathcal{T}]_{([3,2],1)} &= (\mathbf{B} \otimes \mathbf{C}) \cdot [\mathcal{D}]_{([3,2],1)} \cdot \mathbf{A}^T = (\mathbf{B} \boxtimes_R \mathbf{C}) \cdot \text{blkdiag}([\mathcal{D}_1]_{([3,2],1)}, \dots, [\mathcal{D}_R]_{([3,2],1)}) \cdot \mathbf{A}^T, \\
 [\mathcal{T}]_{([1,3],2)} &= (\mathbf{C} \otimes \mathbf{A}) \cdot [\mathcal{D}]_{([1,3],2)} \cdot \mathbf{B}^T = (\mathbf{C} \boxtimes_R \mathbf{A}) \cdot \text{blkdiag}([\mathcal{D}_1]_{([1,3],2)}, \dots, [\mathcal{D}_R]_{([1,3],2)}) \cdot \mathbf{B}^T, \\
 [\mathcal{T}]_{([2,1],3)} &= (\mathbf{A} \otimes \mathbf{B}) \cdot [\mathcal{D}]_{([2,1],3)} \cdot \mathbf{C}^T = (\mathbf{A} \boxtimes_R \mathbf{B}) \cdot \text{blkdiag}([\mathcal{D}_1]_{([2,1],3)}, \dots, [\mathcal{D}_R]_{([2,1],3)}) \cdot \mathbf{C}^T, \\
 \text{and } [\mathcal{T}]_{([3,2,1],0)} &= (\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}) \cdot [\mathcal{D}]_{([3,2,1],0)} = (\mathbf{A} \boxtimes_R \mathbf{B} \boxtimes_R \mathbf{C}) \begin{bmatrix} [\mathcal{D}_1]_{([3,2,1],0)} \\ \vdots \\ [\mathcal{D}_R]_{([3,2,1],0)} \end{bmatrix}.
 \end{aligned}$$

The block diagonalization operator denoted by $\text{blkdiag}(\cdot)$ is defined in (A.1). The authors of [DLN08] propose an ALS algorithm for the computation of the BTM in rank- (L, M, N) terms based on the above given tensor unfoldings. The BTM in rank- (L, M, N) terms is essentially unique under mild conditions. Accordingly, the uniqueness properties of the decomposition are presented in [DL08b]. The BTM in rank- (L, N, M) terms can be easily generalized to higher order tensors. This generalization and the corresponding unfoldings are used later in this thesis (see Chapter 4 and Chapter 5).

Decomposition in rank- (L, M, \cdot) terms

The decomposition in rank- (L, M, \cdot) terms is also referred to as type-2 BTD. The name type-2 corresponds to the Tucker2 decomposition (the Tucker2 decomposition of a 3-way tensor assumes that one of the factor matrices is an identity matrix). For a tensor $\mathcal{T} \in \mathbb{C}^{I \times J \times K}$, the BTD in rank- (L, M, \cdot) terms is defined as [DL08b, DLN08]

$$\mathcal{T} = \sum_{r=1}^R \mathcal{D}_r \times_1 \mathbf{A}_r \times_2 \mathbf{B}_r,$$

where $\mathcal{D}_r \in \mathbb{C}^{L \times M \times K}$ has 1-rank and 2-rank equal to L and M , respectively. Moreover, \mathbf{A}_r has dimensions $I \times L$ ($I \geq L$) and \mathbf{B}_r has dimensions $J \times M$ ($J \geq M$). Both matrices have full column rank. When $R = 1$, the BTD in rank- (L, M, \cdot) terms is equivalent to the Tucker2 decomposition. The BTD in rank- (L, M, \cdot) terms is essentially unique under mild conditions. The uniqueness results are provided in [DL08b].

The decomposition can be computed based on ALS [DLN08]. The ALS updates are easily derived from $\mathcal{T} = \mathcal{D} \times_1 \mathbf{A} \times_2 \mathbf{B}$, where $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_R]$, and \mathcal{D} is a block diagonal tensor build from the tensors \mathcal{D}_r , for $r = 1, \dots, R$.

Computation of the BTD

As previously explained, the BTD can be computed based on ALS [DLN08]. Moreover, the authors of [SvBDL13] develop nonlinear least squares methods which are memory efficient, gradient-based methods for the computation of the BTD. A complexity analysis of the algorithms for the computation of both the BTD and the CP decomposition is provided in [SvBDL13]. The BTD has a higher computational complexity than the CP decomposition.

Applications of the BTD

The BTD as a generalization of the CP decomposition and the Tucker decomposition has many applications such as blind source separation [CMP⁺15] and data fusion [SvBDL13]. In [ZNH14], the BTD is used for channel estimation in two-way relaying MIMO systems with multiple amplify-and-forward relays. Moreover, the BTD has applications in biomedical signal processing. For instance, the BTD can be used for modelling epileptic seizures from EEG (ElectroEncephaloGraphy) recordings [HCS⁺14]. The authors of [RdAZ16] also exploit the BTD for the identification of a cardiac arrhythmia from ECG (ElectroCardioGram) signals.

2.2.4 PARATUCK2

The PARATUCK2 decomposition is a very flexible tensor decomposition representing a mixture of the CP decomposition also known as PARAFAC and the Tucker decomposition (see Section 2.2.2

and Section 2.2.1, respectively). Let $\mathbf{A} \in \mathbb{C}^{I \times J}$ and $\mathbf{B} \in \mathbb{C}^{T \times P}$ be two matrices containing the elements $a_{i,j}$ and $b_{t,p}$ and representing two different sets of latent components, respectively. The PARATUCK2 tensor decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ containing these matrices is defined as [HL96]

$$\mathcal{X}_{(.,.,k)} = \mathbf{A} \cdot \mathcal{D}_{(.,.,k)}^{(A)} \cdot \mathbf{R} \cdot \mathcal{D}_{(.,.,k)}^{(B)} \cdot \mathbf{B}^T, \quad \forall k = 1, 2, \dots, K, \quad (2.46)$$

where the matrix $\mathbf{R} \in \mathbb{C}^{J \times P}$ indicates the interaction between the two different sets of latent components. The 3-mode slices of the two tensors $\mathcal{D}^{(A)} \in \mathbb{C}^{J \times J \times K}$ and $\mathcal{D}^{(B)} \in \mathbb{C}^{P \times P \times K}$ are diagonal matrices with diagonal elements equal to $d_{j,k}^{(A)}$ and $d_{p,k}^{(B)}$, respectively. The PARATUCK2 tensor decomposition is illustrated in Fig. 2.16.

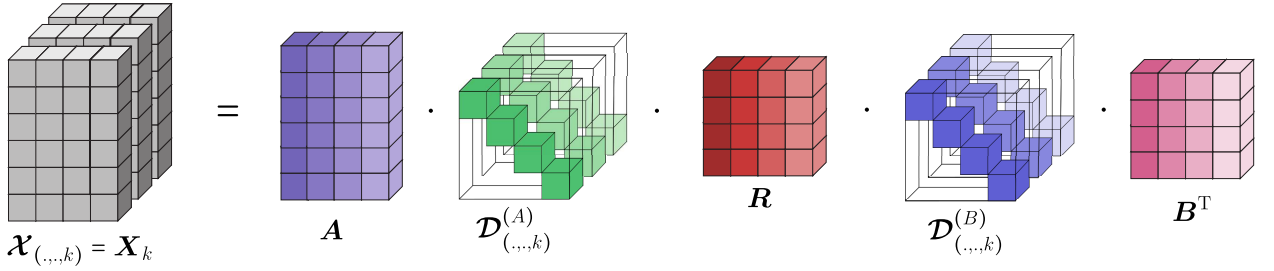


Figure 2.16.: Slice-wise visualization of the PARATUCK2 decomposition for a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ with slices $\mathcal{X}_{(.,.,k)} = \mathbf{A} \cdot \mathcal{D}_{(.,.,k)}^{(A)} \cdot \mathbf{R} \cdot \mathcal{D}_{(.,.,k)}^{(B)} \cdot \mathbf{B}^T$.

Even though the PARATUCK2 decomposition has been proposed a long time ago [HL96], there exist no explicit tensor model for this decomposition. As depicted in Fig. 2.16 the PARATUCK2 decomposition represents a slice-wise multiplication between two tensors. Recall that we have proposed an alternative representation of the slice-wise multiplication based on generalized tensor contraction in Section 2.1.4. In Section 4.1, we exploit this representation to derive a new tensor model for the PARATUCK2 decomposition. By substituting the structure of both tensors involved in the contraction, we show that the PARATUCK2 decomposition fits a constrained CP model.

The uniqueness properties of PARATUCK2 are proven in [HL96], but only for cases where $J = P$. The PARATUCK2 decomposition is unique up to a permutation and scaling ambiguity. For instance, in [HL96] it is shown that the PARATUCK2 decomposition is unique if \mathbf{A} , \mathbf{B} , and \mathbf{R} have full column rank, \mathbf{R} has no zero elements, $J = P = 2$, and $K \geq 9$. For $J = P = 3$ the tensor should have at least $K = 36$ slices in order the decomposition to be unique. Also, the authors of [HL96] prove the uniqueness of the PARATUCK2 for a symmetric case, i.e., $\mathcal{D}^{(A)} = \mathcal{D}^{(B)}$. In this case, the decomposition is unique if \mathbf{A} , \mathbf{B} , and \mathbf{R} have full column rank, \mathbf{R} has no zero elements, for $J = P = 2$, and $K \geq 5$, or for $J = P = 3$ and $K \geq 15$. Even more, the authors of [HL96] include uniqueness results for the DEDICOM (DEcomposition into DIrectional COmponents) [Har78, KB09]. The DEDICOM is a symmetric PARATUCK2 decomposition, where $\mathcal{D}^{(A)} = \mathcal{D}^{(B)}$ and $\mathbf{A} = \mathbf{B}$.

Based on the PARATUCK2 decomposition, it is possible to analyze a large amount of user information, such as user's behavior on social websites, shopping websites, profiling smart con-

tracts, etc. This then leads to the prediction of the user's behavior in the future [CSH18]. Moreover, PARATUCK2 describes various wireless communication systems [SGB00, FdA14a] (see Section 2.3). Similarly, the authors of [dAFX13] propose a space-time-frequency model for MIMO communication systems based on the PARATUCK2 decomposition. These applications assume MIMO-OFDM communication systems. In addition, to tensor based modeling of OFDM systems, we have proposed a PARATUCK2 based model for MIMO-GFDM (Generalized Frequency Division Multiplexing) systems in [NCH⁺17]. Furthermore, a semi-blind receiver for a relay assisted MIMO communication system based on the CP decomposition and the PARATUCK2 decomposition is proposed in [ZNNH15, XFdaS14].

2.2.5 PARAFAC2

The PARAFAC2 decomposition is a generalization of the PARAFAC/CP tensor decomposition. According to [Har72], the PARAFAC2 decomposition of a tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ is defined in the following slice-wise fashion,

$$\mathcal{X}_{(:, :, k)} = \mathbf{X}_k = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k, :)}) \cdot \mathbf{B}_k^T, \quad \forall k = 1, \dots, K. \quad (2.47)$$

It can be interpreted as a coupled matrix decomposition of K matrices $\mathbf{X}_k \in \mathbb{R}^{I \times J}$,⁵ where the matrix $\mathbf{A} \in \mathbb{R}^{I \times R}$ is the coupled mode. The rows of the matrix $\mathbf{C} \in \mathbb{R}^{K \times R}$ contain the weights that correspond to the R underlying components. The second mode is not coupled and therefore each \mathbf{X}_k matrix has a different loading factor $\mathbf{B}_k^T \in \mathbb{R}^{R \times J}$. In general, the PARAFAC2 decomposition is not unique. However, in [Har72] it has been shown that it is essentially unique under mild conditions, i.e., if $\mathbf{B}_k^T \cdot \mathbf{B}_k = \mathbf{F}^T \cdot \mathbf{F}$ such that $\mathbf{B}_k^T = \mathbf{F}^T \cdot \mathbf{V}_k$ and $\mathbf{V}_k \cdot \mathbf{V}_k^T = \mathbf{I}_R$, where $\mathbf{F} \in \mathbb{R}^{R \times R}$ and $\mathbf{V}_k \in \mathbb{R}^{R \times J}$. This is known as the Harshman constraint [Har72].

The slice-wise description of PARAFAC2 is visualized in Fig. 2.17, where it is obvious that the PARAFAC2 decomposition can be regarded as a slice-wise multiplication of two tensors. This slice-wise multiplication can be expressed in terms of the generalized contraction as proposed in Section 2.1.4. Exploiting the tensor contraction properties and the structure of the tensors involved in the contraction, we get a new tensor model for PARAFAC2. In Section 4.2, we show that PARAFAC2 fits a constrained CP model. The resulting tensor structure enables a simultaneous view of all dimensions, leading to an efficient computation of the PARAFAC2 decomposition.

The PARAFAC2 tensor decomposition has many applications in multi-dimensional data analytics. For instance, it can be used in biomedical applications when analyzing time-shifted signals. In [WJR⁺10] and [WJG⁺10], the PARAFAC2 decomposition is used for the identification of the dominant signal components in EEG signals resulting from visual-evoked potentials for each of the different time-shifted channels. Moreover, in [CHGH18], PARAFAC2 is used for the analysis

⁵ Note that in general each matrix \mathbf{X}_k can have different dimensions, i.e., $\mathbf{X}_k \in \mathbb{R}^{I_k \times J_k}$. However, we assume that all matrices have the same dimensions for notation simplicity, $I_1 = \dots = I_K = I$ and $J_1 = \dots = J_K = J$. The tensor notation introduced later in this thesis (see Section 4.2) still holds for different dimensions of the matrices \mathbf{X}_k if we zero pad the matrices to the maximum dimension, i.e., $I = \max(I_1, \dots, I_K)$ and $J = \max(J_1, \dots, J_K)$.

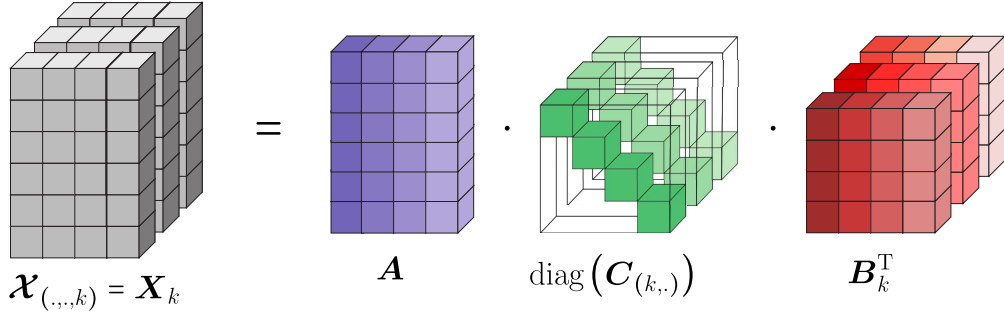


Figure 2.17.: Slice-wise visualization of the PARAFAC2 decomposition for a tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ with slices $\mathbf{X}_k = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{B}_k^T$, $\forall k = 1, \dots, K$.

of somatosensory evoked magnetic fields and somatosensory evoked electrical potentials. The authors of [BAK99] present a model of chromatographic data with time shifts based on PARAFAC2. Furthermore, an algorithm for the computation of the coupled PARAFAC2 decomposition is proposed in [CNH⁺18] for the joint analysis of somatosensory evoked magnetic fields and electrical potentials.

2.3 Applications of Tensor Algebra

Tensor based signal processing offers an improved identifiability, uniqueness, and more efficient denoising compared to matrix based techniques. A good overview of some tensor based signal processing applications is provided in [CMP⁺15], [KB09], and [Cic14]. In this section, we summarize some of the applications of tensor algebra.

Applications in wireless communication systems

Tensors are used to model wireless communication systems due to the multi-dimensional structure of the signals (time, frequency, space, users, etc.). For instance, a signal separation in wireless communications can be performed based on the CP decomposition [CMP⁺15, SDFL⁺17]. Using the CP structure, the authors of [SB02] propose Khatri-Rao space-time codes for MIMO wireless communication systems. Later, the authors of [dAF13a] propose a space-time-frequency coding technique using nested CP models for MIMO-OFDM systems. A model based on the generalized PARATUCK decomposition for MIMO-OFDM-CDMA wireless systems is presented in [FdA14a] using a 4-way tensor. Utilizing this generalized tensor structure, the authors in [FdA14a] propose two types of semi-blind receivers, an ALS based receiver and an LSKRF based receiver. Moreover, a semi-blind receiver for MIMO communication systems based on the PARATUCK2 decomposition is proposed in [dAFX13]. Furthermore, the authors in [dAFM08] derive new precoding and transmit techniques for MIMO systems based on a constrained CP model. In [NHdA17, NHdA18], we show that the received signal in MIMO-OFDM satisfies a constrained CP model (see Section 5.1). The utilization of the model proposed in [NHdA17] leads to an improved receiver design based

on the Khatri-Rao factorization. In addition to tensor based modeling for OFDM systems, a PARATUCK2 based model for MIMO-GFDM systems is presented in [NCH⁺17] (see Section 5.2). All of the previously mentioned references exploit the tensor structure for the design of semi-blind receivers. Note that using tensor algebra, we can estimate both the channel and the data symbols without any prior knowledge (this corresponds to a blind receiver). However, we can estimate the channel and the transmitted data symbols up to a scaling and permutation ambiguity. To resolve the scaling ambiguity we require an initial estimate of the communication channel and/or pilot symbols (this corresponds to a semi-blind receiver). In this thesis, we present the tensor models for MIMO-OFDM and MIMO-GFDM systems [NHdA17,NHdA18,NCH⁺17]. The details are described in Section 5.1 and Section 5.2, respectively.

In [SGB00], a multi-user communication system is modeled in terms of the CP decomposition. The proposed tensor model leads to the design of a semi-blind receiver for CDMA systems. Similarly, in [dAF13b], a Khatri-Rao coding in space and frequency for multi-user MIMO-OFDM systems is proposed based on the CP tensor decomposition.

Moreover, tensor models are often exploited for modeling two-way relaying systems. Such CP based algorithms are presented in [RH10] and [RKX12] for channel estimation in a two-way wireless relaying system. A semi-blind receiver for a relay assisted MIMO communication system based on the CP and the PARATUCK2 decomposition is proposed in [ZNNH15,XFdAS14]. Moreover, a nested Tucker model is used in [SdAH17] to model two-hop MIMO relaying systems. We show in [ZNH14] that a two-way relaying system with multiple relays can be modeled according to the BTD. These tensor based techniques provide an improved estimation accuracy and require less training data as compared to the traditional matrix based solutions.

Tensor based subspace estimation is usually performed using the truncated HOSVD for noise reduction [CRKH14,RH13b,SDLF⁺17]. Using the CP decomposition, we can also perform joint identification and extraction of multiple sources from noisy observations [LC10,CL11,SDLF⁺17], and subspace tracking [NS09,NAMLT16] for the estimation of the underlying features. Moreover, tensor algebra is a practical tool for source separation in time-varying applications such as multi-dimensional harmonic retrieval [HRDG08,RH13b,CRKH14]. The authors of [NS09,CMP⁺15,SDLF⁺17] present CP based applications for the estimation and tracking of the directions of arrival or the directions of departure. A near field localization based on the CP decomposition is performed in [PHT⁺18]. Moreover, the coupled CP tensor decomposition is suitable for several combined signal processing applications such as multirate sampling for array signal processing [SDL17a,SDL17b].

Applications in biomedical engineering

Tensors and tensor decompositions are widely used in neuroscience for the analysis of EEG and/or MEG (MagnetoEncephaloGraphy) signals [KB09]. In [CLK⁺15,BAC⁺14], tensor decompositions are applied for the analysis of EEG signals. Blind source separation for event related sources can be performed based on signal subspace tracking using the CP decomposition. For instance, in

[NBR⁺14], an extraction of event related sources is tested on EEG, ECG, and MEG data, and it is also shown that the CP based analysis is more robust against outliers. The CP decomposition is also used for the imputation of missing entries in big tensors resulting from MRI (Magnetic Resonance Imaging) images [MMG15]. The authors in [BMG13] also introduce a BSUM (Block Successive Upper Bound Minimization) algorithm for tensor imputation, which is applied on MRI images as well as RNA (RiboNucleic Acid) sequencing.

Moreover, in [CHGH18] the authors exploit the PARAFAC2 decomposition for the analysis of somatosensory evoked magnetic fields and somatosensory evoked electrical potentials. An extension of this method is proposed in [CNH⁺18], where the authors propose a coupled PARAFAC2 decomposition for the joint analysis of somatosensory evoked magnetic fields and electrical potentials. An identification of the signal components in EEG data based on the PARAFAC2 decomposition is also performed in [WJG⁺10, WJR⁺10]. The authors of [AKD11, ABS15] show that data fusion in metabolomics benefits from coupled matrix-tensor decompositions. Furthermore, a coupled CP is a valuable tool for data/signal analysis in biomedical applications for the joint analysis of EEG and MEG signals [BCA12, NKHH17, NLA⁺17]. In Section 6.1 and Section 6.2, we present our contribution to the joint analysis of EEG and MEG signals using the C-SECSI (Coupled-Semi-Algebraic framework for approximate coupled CP decomposition via SIMultaneous matrix diagonalization) framework for the computation of the coupled CP decomposition.

Other applications

In addition to the applications of tensor algebra in wireless communication systems and biomedical signal processing, tensor algebra has many more applications. These include data analytics in chemical analysis, psychometrics, image processing, video processing, topic modelling, etc.

A recovery of missing entries is performed using the HOSVD decomposition in [YFLZ16]. On the other hand, an imputation of missing entries based on the CP decomposition is presented in [AKDM11, BMG13, MMG15, SDLF⁺17]. The CP decomposition is applicable for data analytics from rating websites such as movie websites and Gaussian mixture parameter estimation [SDLF⁺17]. The coupled CP decomposition is applicable for data fusion of heterogeneous data sets of multiple sources, e.g., data from social websites and review websites can be processed jointly [AKD11]. Clustering of heterogeneous data sets is also proposed in [ABS15] based on the coupled CP decomposition.

Moreover, the authors of [CC70] propose one of the first applications of the CP decomposition in psychometrics. The estimation of the emission factors in amino acids fluorescence data or fluorescence spectroscopy can be performed in terms of the CP decomposition as shown in [HAK⁺17, LA14]. Furthermore, using the PARAFAC2 decomposition we describe data models that additionally vary along one of the tensor dimension (e.g., time shifts). The authors of [BAK99] present a model of a chromatographic data based on PARAFAC2.

Dimensionality reduction can be performed using the truncated HOSVD. For instance, the SECSI

framework exploits the truncated HOSVD for dimensionality reduction as an initial step of the computation of the CP decomposition [RH13a]. The truncated HOSVD is also used in applications such as image processing, including hyper-spectral imaging, deblurring via image filtering, and tensor completion problem [CC13a, CC13b]. Moreover, the HOSVD is often utilized in video processing for tracking motion trajectories [MSK09].

Furthermore, speech or text processing applications for topic modeling using the CP decomposition are presented in [SDLF⁺17]. Similar, an identification of estimated sources according to a known dictionary can be achieved using the CP decomposition [CG18].

Naturally, tensors and tensor decompositions play an important role in big data applications. In [SPF14], the authors propose a novel approach for parallel and distributed computation of low-rank tensor decompositions for data analytics and data compression. As shown in [CSH18], based on PARATUCK2 it is also possible to analyze large amounts of users information and to predicted the user's behavior in the future. For instance, the PARATUCK2 decomposition models smart contracts activities such that it highlights the time dependent latent parameters [CSH18]. These latent parameters are then modeled by the long short term memory network for predictive analytics [CSH18].

2.4 Conclusions

In this chapter, we provide the used notation and fundamental concepts of tensor algebra including the required definitions, properties of the Kronecker product, properties of the Khatri-Rao product, properties of the Hadamard product, LSKF (Least-Squared Kronecker Factorization), and LSKRF (Least-Squared Khatri-Rao Factorization). Moreover, we present our contribution to the fundamental concepts of tensor algebra for the generalized tensor contraction operator. In particular, we propose an alternative representation of the element-wise multiplication and slice-wise multiplication between two arrays (vector/matrices/tensors) based on generalized contraction. In contrast to the element-wise or slice-wise representations, this novel representation facilitates the derivation of the explicit tensor structure and all corresponding tensor unfoldings as we show in Chapter 4 and Chapter 5 of this thesis. Some parts of our contributions for the element-wise and slice-wise multiplications via generalized unfoldings has already been published in [NCdAH18]. Furthermore, in this chapter we review the basic tensor decompositions that will be used in the sequel the HOSVD, the CP decomposition, the BTD, the PARATUCK2 decomposition, and the PARAFAC2 decomposition. We also list some of the many applications of tensor algebra mainly including applications to wireless communication systems and biomedical signal processing that are the main scope of this thesis.

Chapter 3

Computation of the CP decomposition and the coupled CP decomposition

The CP decomposition decomposes a tensor $\mathcal{X}_0 \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ into R rank one components.

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3 \quad (3.1)$$

The matrices $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2 \in \mathbb{C}^{M_2 \times R}$, and $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$ are the factor matrices containing the signatures for each of the modes, and R is the tensor rank. The construction of the tensor unfoldings, the uniqueness properties of the CP decomposition, the extension to N -way arrays, and other details have already been discussed in Section 2.2.2. In Section 2.2.2, we also briefly discuss the computation of the CP decomposition. The algorithms for the computation of the CP decomposition can be categorized as ALS based algorithms, line search algorithms, GD based algorithms, NLS based algorithms, deflation-based algorithms, and semi-algebraic algorithms. The semi-algebraic algorithms exploit the CP tensor structure and convert the CP decomposition into an SMD. The link between the CP decomposition and an SMD was introduced in [DL05]. Later a complete semi-algebraic algorithm for the computation of CP is proposed in [LA11]. An improvement of this algorithm is then proposed in [LA14]. The semi-algebraic algorithm [LA11, LA14] known as DIAG (Direct ALgorithm for canonical polyadic decomposition) considers only the diagonalization of the tensor along one mode, and therefore it solves only one SMD. The earlier version of the algorithm [LA11] is denoted by SALT (Semi-ALgebraic Tensor decomposition). The algorithms SALT and DIAG use different algorithms for the computation of the SMD. The SALT algorithm utilizes Jacobi-like algorithm called JET (Joint Eigenvalue decomposition algorithm based on Triangular matrices). The DIAG algorithm utilizes the JD TM (Joint Diagonalization algorithm based on Targeting hyperbolic Matrices) algorithm that uses a polar matrix factorization to compute the SMD. On the other hand, the SECSI framework proposed in [RH08, RH13a] considers the diagonalization of the tensor along all modes and thereby it solves all possible SMDs. An improved extension of the framework for an N -way ($N > 3$) tensor based on generalized unfoldings is proposed in [RSH12], SECSI-GU. The SECSI framework computes multiple initial estimates of the factor matrices. Then, in a subsequent step a final estimate is chosen based on different heuristics. The different heuristics lead to a complexity-accuracy trade-off of the SECSI framework [RH13a].

In this chapter, we present extensions of the SECSI framework [RH08, RH13a, RSH12] that reduce the computation complexity and/or introduce constraints. These extensions include the truncated

SECSI (T-SECSI) [NHT⁺16], non-symmetric SECSI (NS-SECSI) [NHT⁺16], SECSI for symmetric tensors (S-SECSI), SECSI for non-negative tensors (SECSI₊), and an extension of SECSI to the coupled SECSI (C-SECSI) [NH16] for the computation of the coupled CP decomposition. We devote a section of this chapter to each of the proposed extensions of the SECSI framework, Sections 3.2-3.6, respectively. For simpler differentiation of our contributions for the proposed SECSI extensions and the original SECSI framework, we briefly review the link between the CP decomposition and the SMD and the SECSI framework proposed in [RH13a] in the following section.

3.1 Introduction to the SECSI framework

In practice, we have a noisy observation of the signal tensor $\mathcal{X}_0 \in \mathbb{C}^{M_1 \times M_2 \times M_3}$, i.e., $\mathcal{X} = \mathcal{X}_0 + \mathcal{N}$. The tensor \mathcal{N} represents ZMCSCG noise with variance σ_N^2 . From this noisy observation we can compute only a low-rank approximation of the observed signal \mathcal{X}_0 . For notation simplicity we consider the noiseless case first. In the presence of noise, all following relations still hold approximately. Moreover, we assume that $R \leq \min(M_1, M_2, M_3)$. However, the SECSI framework can also compute the CP decomposition of rank deficient tensor if the 3-way tensor is rank deficient in up to two modes [RH13a].

First, consider the link between the CP decomposition and the truncated HOSVD given by

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3 = \mathcal{S}^{[s]} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s]} \times_3 \mathbf{U}_3^{[s]}, \quad (3.2)$$

where the truncated core tensor $\mathcal{S}^{[s]} \in \mathbb{C}^{R \times R \times R}$ and the matrices $\mathbf{U}_1^{[s]} \in \mathbb{C}^{M_1 \times R}$, $\mathbf{U}_2^{[s]} \in \mathbb{C}^{M_2 \times R}$, and $\mathbf{U}_3^{[s]} \in \mathbb{C}^{M_3 \times R}$ correspond to the truncated HOSVD (for its definition see equation (2.37)). The factor matrices, $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2 \in \mathbb{C}^{M_2 \times R}$, and $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$ correspond to the CP decomposition in (3.1). The 1-mode unfolding of the tensor \mathcal{X}_0 satisfies

$$\begin{aligned} [\mathcal{X}_0]_{(1)} &= \mathbf{U}_1^{[s]} \cdot \left([\mathcal{S}^{[s]}]_{(1)} \cdot [\mathbf{U}_3^{[s]} \otimes \mathbf{U}_2^{[s]}]^T \right) \\ &= \mathbf{F}_1 \cdot \left([\mathcal{I}_{3,R}]_{(1)} \cdot [\mathbf{F}_3 \otimes \mathbf{F}_2]^T \right). \end{aligned}$$

Note that the matrices $\mathbf{U}_1^{[s]}$ and \mathbf{F}_1 span the column space of $[\mathcal{X}_0]_{(1)}$. Hence, there exists an invertible matrix $\mathbf{T}_1 \in \mathbb{C}^{R \times R}$ such that $\mathbf{F}_1 = \mathbf{U}_1^{[s]} \cdot \mathbf{T}_1$. Similarly, for the other two modes, we have $\mathbf{F}_2 = \mathbf{U}_2^{[s]} \cdot \mathbf{T}_2$ and $\mathbf{F}_3 = \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3$. By substituting these matrices in equation (3.2), we have

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{U}_1^{[s]} \cdot \mathbf{T}_1 \times_2 \mathbf{U}_2^{[s]} \cdot \mathbf{T}_2 \times_3 \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3 = \mathcal{S}^{[s]} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s]} \times_3 \mathbf{U}_3^{[s]}.$$

The above equation represents the fundamental link between the truncated HOSVD and the CP decomposition. Hence, for the truncated core tensor, we get $\mathcal{S}^{[s]} = \mathcal{I}_{3,R} \times_1 \mathbf{T}_1 \times_2 \mathbf{T}_2 \times_3 \mathbf{T}_3$. Moreover,

observing

$$\mathcal{S}^{[s]} \times_1 \mathbf{T}_1^{-1} \times_2 \mathbf{T}_2^{-1} \times_3 \mathbf{T}_3^{-1} = \mathcal{I}_{3,R}, \quad (3.3)$$

we see that the invertible matrices \mathbf{T}_1^{-1} , \mathbf{T}_2^{-1} , and \mathbf{T}_3^{-1} diagonalize the core tensor $\mathcal{S}^{[s]}$. The extensions of the SECSI framework presented in the remainder of this chapter differ in the way of solving this diagonalization problem and the different constraints imposed on the factor matrices \mathbf{F}_1 , \mathbf{F}_2 , and \mathbf{F}_3 . The original SECSI framework diagonalizes the truncated core tensor based on symmetric SMDs. After the 3-mode multiplication of the core tensor in (3.3) by $\mathbf{U}_3^{[s]}$, i.e., $\mathcal{S}_3 = \mathcal{S}^{[s]} \times_3 \mathbf{U}_3^{[s]}$, we get

$$\mathcal{S}_3 \times_1 \mathbf{T}_1^{-1} \times_2 \mathbf{T}_2^{-1} = \mathcal{I}_{3,R} \times_3 \underbrace{\mathbf{U}_3^{[s]} \mathbf{T}_3}_{\mathbf{F}_3}. \quad (3.4)$$

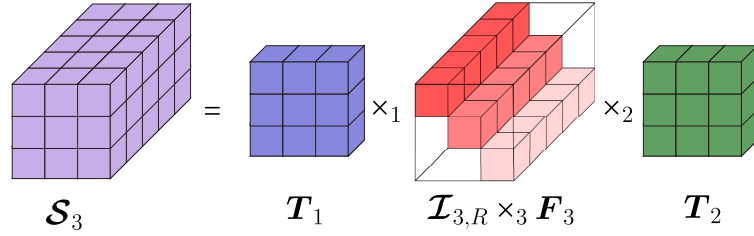


Figure 3.1.: Diagonalization of the tensor $\mathcal{S}_3^{[s]} \in \mathbb{C}^{R \times R \times M_3}$ along the 3-mode.

In Fig. 3.1, we visualize the above equation (3.4). We see that the invertible transform matrices \mathbf{T}_1 and \mathbf{T}_2 diagonalize the matrices $\mathcal{S}_{3(.,.,m_3)}$ (the 3-mode slides of \mathcal{S}_3) jointly for all $m_3 = 1, \dots, M_3$. Note that we can obtain

$$\mathbf{T}_1^{-1} \cdot \mathcal{S}_{3(.,.,m_3)} \cdot \mathbf{T}_2^{-T} = \text{diag}(\mathbf{F}_{3(m_3,.)}) \quad (3.5)$$

by multiplying the equation (3.4) along the 3-mode with the transpose of a pinning vector $\mathbf{e}_{m_3} \in \mathbb{R}^{M_3 \times 1}$, $\forall m_3 = 1, \dots, M_3$. The superscript $^{-T}$ denotes transposition and matrix inversion. The equation (3.5) reveals the link between the CP decomposition and an SMD. To be more precise, equation (3.5) represents a non-symmetric SMD. The original SECSI framework [RH13a] proposes to convert this non-symmetric SMD into two symmetric SMDs. By eliminating one of the transform matrices from the right-hand side, we have

$$\begin{aligned} \mathcal{S}_3^{\text{rhs}}(.,.,m_3) &= \mathcal{S}_{3(.,.,m_3)} \cdot \mathcal{S}_{3(.,.,p)}^{-1} = \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)}) \cdot \mathbf{T}_2^T \cdot \mathbf{T}_2^{-T} \cdot \text{diag}(\mathbf{F}_{3(p,.)})^{-1} \cdot \mathbf{T}_1^{-1} \\ &= \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)} \oslash \mathbf{F}_{3(p,.)}) \cdot \mathbf{T}_1^{-1}. \end{aligned} \quad (3.6)$$

By eliminating one of the transform matrices from the left-hand side, we have

$$\begin{aligned} \mathcal{S}_3^{\text{lhs}}(.,.,m_3) &= (\mathcal{S}_{3(.,.,p)}^{-1} \cdot \mathcal{S}_{3(.,.,m_3)})^T = \mathbf{T}_2 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)}) \cdot \mathbf{T}_1^T \cdot \mathbf{T}_1^{-T} \cdot \text{diag}(\mathbf{F}_{3(p,.)})^{-1} \cdot \mathbf{T}_2^{-1} \\ &= \mathbf{T}_2 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)} \oslash \mathbf{F}_{3(p,.)}) \cdot \mathbf{T}_2^{-1}. \end{aligned} \quad (3.7)$$

The symbol \oslash denotes the inverse Hadamard product (element-wise division). In general, the pivoting slice $\mathcal{S}_{3(\dots,p)}$ can be chosen randomly, however, a clever choice is based on the conditioning number, i.e., $p = \arg \min_{m_3=1,\dots,M_3} \text{cond} \{ \mathcal{S}_{3(\dots,m_3)} \}$. Note that the resulting diagonal elements are scaled version of the m_3 -th row of the factor matrix \mathbf{F}_3 . However, this scaling corresponds to the scaling ambiguity of the CP decomposition introduced in Section 2.2.2. The authors of [RH13a] recommend the algorithm proposed in [FG06] for the computation of the symmetric SMDs. After the diagonalization of the tensors $\mathcal{S}_3^{\text{lhs}}$ and $\mathcal{S}_3^{\text{rhs}}$ along the 3-mode we obtain the transform matrices \mathbf{T}_1 and \mathbf{T}_2 and two estimates of the matrix \mathbf{F}_3 . Therefore, from equation (3.6) and the link between the truncated HOSVD and the CP decomposition, we obtain $\mathbf{F}_1 = \mathbf{U}_1^{[\text{s}]} \cdot \mathbf{T}_1$ from the transform matrix \mathbf{T}_1 and \mathbf{F}_3 from the diagonal elements of the diagonalized tensor. The remaining factor matrix is then estimated based on an LS fit, i.e., $\mathbf{F}_2 = [\mathcal{X}_0]_{(2)} (\mathbf{F}_3 \oslash \mathbf{F}_1)^{-\text{T}}$. Similarly, from the diagonalization of the tensor $\mathcal{S}_3^{\text{lhs}}$ in (3.7) and the link between the truncated HOSVD and the CP decomposition we compute $\mathbf{F}_2 = \mathbf{U}_2^{[\text{s}]} \cdot \mathbf{T}_2$, \mathbf{F}_3 from the diagonal elements, and \mathbf{F}_1 via an LS fit. Hence, we obtain two sets of estimates for the factor matrices. However, the truncated core tensor $\mathcal{S}^{[\text{s}]}$ (see equation (3.3)) can also be diagonalized along the 1-mode and the 2-mode after the multiplication along the corresponding mode with $\mathbf{U}_1^{[\text{s}]}$ and $\mathbf{U}_2^{[\text{s}]}$, respectively. The diagonalization of the core tensor along the remaining two modes results in four additional sets of estimates of the factor matrices.

The resulting six initial sets of estimates of the factor matrices are depicted in Fig. 3.2 for a 3-way tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ with rank R . Each of these initial estimates is depicted by one parallel branch. Moreover, in Fig. 3.2 we indicate whether the factor matrices are estimated from a transform matrix, from the diagonal elements, or via an LS fit. The final estimate can then be chosen based on different heuristics. The authors of [RH13a] propose the following criteria, BM (Best Matching), REC PS (REConstruction criterion Paired Solutions), RES (RESiduals criterion), and CON PS (CONditioning criterion Paired Solutions). The BM solves all SMDs and all different combinations of the factor matrices that can be selected, while searching for the best available solution (an exhaustive search). According to BM, the final estimate is the one that leads to the lowest reconstruction error. The reconstruction error is calculated according to

$$\text{REC} = \frac{\|\hat{\mathcal{X}} - \mathcal{X}\|_{\text{H}}^2}{\|\mathcal{X}\|_{\text{H}}^2}, \quad (3.8)$$

where $\hat{\mathcal{X}}$ denotes the estimated tensor and \mathcal{X} denotes the noisy input tensor. The heuristic REC PS also solves all SMDs, but only considers combinations originating from the same SMD. For instance, for a 3-way tensor the heuristic REC PS choses one of the six initial estimates that leads to the smallest REC error as a final estimate. Moreover, RES also solves all SMDs, but as a final estimate we choose the factor matrices corresponding to the symmetric SMD that results in the smallest residual error after the diagonalization. On the other hand, the CON PS solves only two SMDs chosen based on the conditioning number of the slices prior to the diagonalization. Hence, the most computationally expensive selection criterion is the BM, followed by REC PS, and RES.

The least computational expensive criterion is the CON PS.

The authors of [CBW⁺17, CBC⁺17] present the performance analysis of the SECSI framework. This performance analysis is obtained using a first-order perturbation analysis. The closed-form expressions of the relative mean square error are derived for each of the estimated factor matrices and they are formulated in terms of the noise variance. Moreover, the authors of [CCH⁺18] present a performance analysis of SECSI-GU based on a first-order perturbation analysis. The authors of [RSH12] recommend the SECSI-GU framework for the computation of the CP decomposition for N -way ($N > 3$) tensors.

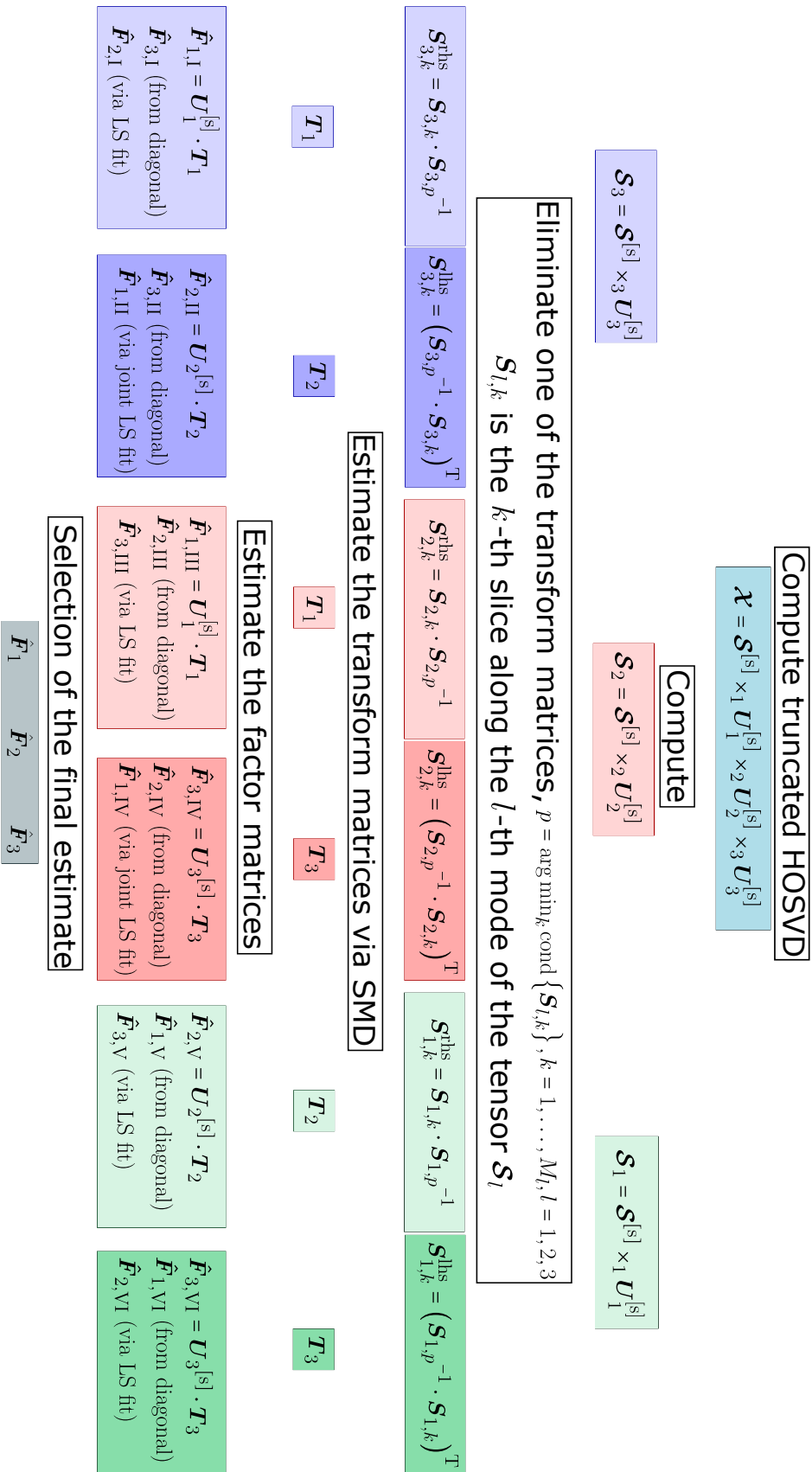


Figure 3.2.: The SECSI framework for the computation of the CP decomposition of a tensor $\chi \in \mathbb{C}^{M_1 \times M_2 \times M_3}$ with rank R .

3.2 Truncated Semi-Algebraic framework for the approximate CP decomposition via Simultaneous matrix diagonalization (T-SECSI)

As summarized in the beginning of this chapter, the SECSI framework [RH08, RSH12, RH13a] calculates all possible SMDs, and then selects the best available solution in a final step via appropriate heuristics. The different heuristics offer a complexity-accuracy trade-off. However, for some tensor applications like biomedical signal processing and big data applications the tensor dimensions $M_1 \times M_2 \times M_3$ are very large. For such applications even the least computationally expensive heuristics (CON) might still be computationally too expensive to solve. That is because SECSI diagonalizes a tensor of dimension $R \times R \times M_n$, where R is the tensor rank and M_n corresponds to one of the tensor dimensions M_1 , M_2 , or M_3 (the dimension along which the diagonalization takes place). Therefore, we propose a truncated SECSI (T-SECSI) framework [NHT⁺16] that reduces the computational complexity of the original SECSI framework [RH08, RSH12, RH13a].

Considering equation (3.3), we define the tensor $\mathcal{T}_3 = \mathcal{I}_{3,R} \times_3 \mathbf{T}_3 \in \mathbb{C}^{R \times R \times R}$ such that we get

$$\mathcal{S}^{[s]} \times_1 \mathbf{T}_1^{-1} \times_2 \mathbf{T}_2^{-1} = \mathcal{T}_3. \quad (3.9)$$

Note that \mathcal{T}_3 contains diagonal slices along the third mode. Hence, we need to diagonalize the truncated core tensor $\mathcal{S}^{[s]}$ along the 3-mode, or in other words we need to estimate the matrices \mathbf{T}_1 and \mathbf{T}_2 that diagonalize the tensor $\mathcal{S}^{[s]}$ along the 3-mode. As compared to the equation (3.4) where the tensor that is diagonalized has dimensions $R \times R \times M_3$ (also depicted in Fig. 3.1) here the tensor that is diagonalized $\mathcal{S}^{[s]}$ has dimensions $R \times R \times R$. Thus, we diagonalize only the truncated tensor that has reduced dimensions, $R \ll M_n$ (for low-rank tensors with large dimensions), for $n = 1, 2, 3$.

In order to obtain the set of matrices for the symmetric SMD, one of the transform matrices has to be eliminated and the truncated core tensor has to be sliced accordingly. When we use the third mode of the tensor as presented up to now, the diagonal matrices are aligned along the 3-mode of the tensor. In order to select the 3-mode slices, we multiply along the 3-mode with the transpose of a pinning vector \mathbf{e}_r that is the r -th column of an $R \times R$ identity matrix. Thus, each of the corresponding slices is defined as $\mathcal{S}^{[s]}_{(:, :, r)} = \mathcal{S}^{[s]} \times_3 \mathbf{e}_r^T$ and $\mathcal{T}_{3(:, :, r)} = \mathcal{T}_3 \times_3 \mathbf{e}_r^T = \text{diag}(\mathbf{T}_{3(r, \cdot)})$, for the left-hand and the right-hand side of equation (3.9). The elimination of one of the transform matrices is similar to the original SECSI framework. For the inversion, from the right-hand side and from the left-hand side, we have

$$\begin{aligned} \mathcal{S}^{\text{rhs}}_{(:, :, r)} &= \mathcal{S}^{[s]}_{(:, :, r)} \cdot \mathcal{S}^{[s]}_{(:, :, p)}^{-1} = \mathbf{T}_1 \cdot \text{diag}(\mathbf{T}_{3(r, \cdot)}) \cdot \mathbf{T}_2^T \cdot \mathbf{T}_2^{-T} \cdot \text{diag}(\mathbf{T}_{3(p, \cdot)})^{-1} \cdot \mathbf{T}_1^{-1} \\ &= \mathbf{T}_1 \cdot \text{diag}(\mathbf{T}_{3(r, \cdot)} \oslash \mathbf{T}_{3(p, \cdot)}) \cdot \mathbf{T}_1^{-1}, \text{ and} \\ \mathcal{S}^{\text{lhs}}_{(:, :, r)} &= \left(\mathcal{S}^{[s]}_{(:, :, p)}^{-1} \cdot \mathcal{S}^{[s]}_{(:, :, r)} \right)^T = \mathbf{T}_2 \cdot \text{diag}(\mathbf{T}_{3(r, \cdot)}) \cdot \mathbf{T}_1^T \cdot \mathbf{T}_1^{-T} \cdot \text{diag}(\mathbf{T}_{3(p, \cdot)})^{-1} \cdot \mathbf{T}_2^{-1} \\ &= \mathbf{T}_2 \cdot \text{diag}(\mathbf{T}_{3(r, \cdot)} \oslash \mathbf{T}_{3(p, \cdot)}) \cdot \mathbf{T}_2^{-1}, \end{aligned}$$

respectively. The pivoting slice $\mathcal{S}^{[s]}_{(...,p)}$ is chosen based on the conditioning number like in the original SECSI framework. The above equations represent two symmetric SMDs. Note that we have a set of R equations instead of the M_3 ($M_3 \geq R$) equations of the original SECSI framework, which reduces the computational complexity of the symmetric SMD. For the computation of the SMD we consider two algorithms which are proposed in [FG06] and [LA10,LA14]. The JD (Joint eigenvalue Decomposition) algorithm proposed in [FG06] computes the SMD based on unitary and non-unitary similarity transformations. The JD TM algorithm uses a polar matrix factorization to compute the SMD. The authors of [BCW⁺17] perform a performance analysis of these algorithms based on first-order perturbation analysis. Using these algorithms, an estimate of the matrix \mathbf{T}_1 is achieved from the diagonalization of $\mathcal{S}^{\text{lhs}}_{(...,r)}$ and \mathbf{T}_3 is calculated from the diagonal elements of \mathcal{T}_3 . Hence, the first set of the estimates of the factor matrices corresponding to the CP decomposition is obtained according to

$$\hat{\mathbf{F}}_{1,\text{I}} = \mathbf{U}_1^{[s]} \cdot \mathbf{T}_1 \quad \hat{\mathbf{F}}_{3,\text{I}} = \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3 \quad \hat{\mathbf{F}}_{2,\text{I}} = [\mathcal{X}]_{(2)} (\hat{\mathbf{F}}_3 \diamond \hat{\mathbf{F}}_1)^{-\text{T}}.$$

Moreover, from the diagonalization of $\mathcal{S}^{\text{rhs}}_{(...,r)}$ we estimate \mathbf{T}_2 and calculate \mathbf{T}_3 from the diagonal elements. As a result, we obtain the second set of factor matrices.

$$\hat{\mathbf{F}}_{2,\text{II}} = \mathbf{U}_2^{[s]} \cdot \mathbf{T}_2 \quad \hat{\mathbf{F}}_{3,\text{II}} = \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3 \quad \hat{\mathbf{F}}_{1,\text{II}} = [\mathcal{X}]_{(1)} (\hat{\mathbf{F}}_3 \diamond \hat{\mathbf{F}}_2)^{-\text{T}}.$$

Moreover, similar to the original SECSI framework the two additional tensor modes can be exploited such that four more sets of factor matrices are estimated. Accordingly, the core tensor should be sliced along its 1-mode and 2-mode, one of the transom matrices should be eliminated, and then diagonalized via symmetric SMDs. The remaining steps of choosing the final solution are identical as in the original SECSI framework. The T-SECSI framework can be visualized similarly to the original SECSI framework in Fig. 3.2 with the exception of the second step and the fact the SMDs have reduced dimensions as previously explained.

3.2.1 Simulation Results

In this subsection, we compare the proposed truncated extension of SECSI, denoted by T-SECSI, with the original SECSI framework [RH13a]. More specifically, we compare the accuracy of SECSI using the JD algorithm [FG06], SECSI using the JD TM algorithm [LA14], T-SECSI using the JD algorithm, and T-SECSI using the JD TM algorithm. All of the algorithms exploit the REC PS heuristic. We denote these algorithms by SECSI REC PS, SECSI-JD TM REC PS, T-SECSI-JD REC PS, and T-SECSI-JD TM REC PS. For the comparisons, we generate tensors \mathcal{X}_0 with rank R according to the CP decomposition in equation (3.1). The factor matrices \mathbf{F}_1 , \mathbf{F}_2 , and \mathbf{F}_3 have i.i.d. zero mean Gaussian distributed random entries with variance one for real-valued tensors or ZMCSCG random entries with variance one for complex-valued tensors. Moreover, for some simulation scenarios we use tensors that have correlated factor matrices. Therefore, we add

correlation using a correlation matrix $\mathbf{R}(\rho_n) \in \mathbb{R}^{R \times R}$

$$\begin{aligned} \mathbf{F}_n &\leftarrow \mathbf{F}_n \cdot \mathbf{R}(\rho_n) \\ \mathbf{R}(\rho_n) &= (1 - \rho_n) \cdot \mathbf{I}_{R \times R} + \frac{\rho_n}{R} \cdot \mathbf{1}_{R \times R}, \end{aligned} \quad (3.10)$$

where ρ_n is the correlation factor corresponding to the n -th factor matrix and $\mathbf{1}_{R \times R}$ denotes a matrix of ones. Afterwards, we add i.i.d. zero mean Gaussian noise with variance σ_N^2 . The noisy observation of the tensor is $\mathcal{X} = \mathcal{X}_0 + \mathcal{N}$, where \mathcal{N} is the noise tensor. The resulting SNR in dB is $\text{SNR} = 10 \log_{10} \left(\frac{\|\mathcal{X}_0^{(1)}\|_H^2}{\|\mathcal{N}^{(1)}\|_H^2} \right)$.

In order to evaluate the accuracy of the estimated factor matrices $\hat{\mathbf{F}}_n$, we define the TSFE (Total Squared Factor Error) as

$$\text{TSFE} = \frac{1}{N} \sum_{n=1}^N \min_{\mathbf{P} \in \mathcal{M}_{\text{PD}}(R)} \frac{\|\hat{\mathbf{F}}_n \cdot \mathbf{P} - \mathbf{F}_n\|_F^2}{\|\mathbf{F}_n\|_F^2}, \quad (3.11)$$

where $\mathcal{M}_{\text{PD}}(R)$ is a set of permutation matrices of size $R \times R$, R is the tensor rank, and N is the tensor dimensionality. The permutation matrix \mathbf{P} resolves the permutation ambiguity that arises from the computation of the CP decomposition.

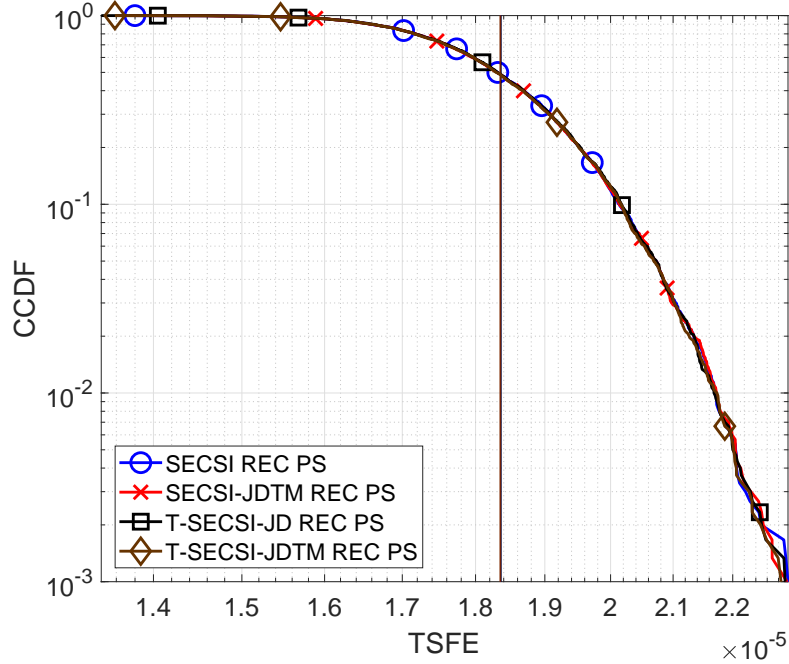


Figure 3.3.: CCDF of the TSFE for a real-valued tensor with dimensions $40 \times 40 \times 40$, tensor rank $R = 3$, and SNR = 20 dB.

In Fig. 3.3, we depict the CCDF (Complementary Cumulative Distribution Function) of the TSFE for 3000 realizations. The decomposed tensor is real-valued with dimensions $40 \times 40 \times 40$

and rank $R = 3$. The results depicted in Fig. 3.3 correspond to $\text{SNR} = 20$ dB. By depicting the CCDF of the TSFE we visualize the outliers in addition to the resulting TSFE. Moreover, we depict the mean of the TSFE by the vertical lines in Fig. 3.3. In Fig. 3.3, all algorithms SECSI REC PS, SECSI-JTDM REC PS, T-SECSI-JD REC PS, and T-SECSI-JDTM REC PS have the same accuracy. Note that for both algorithms JD and JDTM the maximum number of iterations is set to 50.

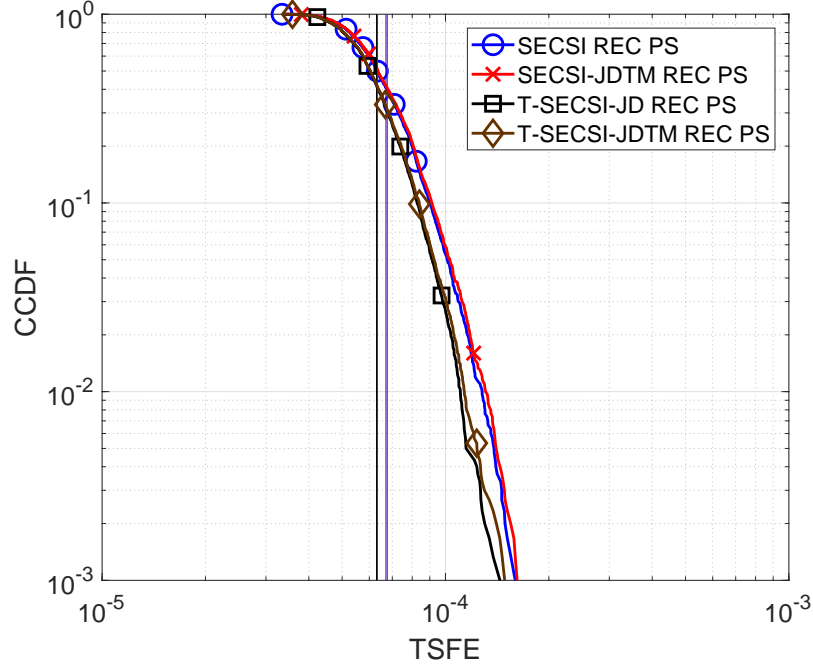


Figure 3.4.: CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and $\text{SNR} = 20$ dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient 0.9.

Next, we consider a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and $\text{SNR} = 20$ dB. Moreover, the first factor matrix \mathbf{F}_1 has mutually correlated columns with correlation coefficient $\rho_1 = 0.9$. The correlation is added as described in (3.10). In Fig. 3.4, we show the CCDF of the resulting TSFE for 3000 realizations. Both versions of the T-SECSI framework T-SECSI-JD and T-SECSI-JDTM have even an improved accuracy as compared to the SECSI framework.

Moreover, in Figs. 3.5 and 3.6 we depict the TMSFE (Total Mean Squared Factor Error) that is the averaged TSFE in equation (3.11) over 3000 realizations as a function of the SNR. In Fig. 3.5, we depict the TMSFE for SNR values between 0 dB and 40 dB. The synthetic tensor is real-valued with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$. Unlike the first two examples, here we observe a decreased accuracy of the T-SECSI framework as compared to the SECSI framework. Moreover, the accuracy gap increases with the increased SNR.

In Fig. 3.6, we illustrate the TMSFE for SNR values between 0 dB and 45 dB of a complex-valued

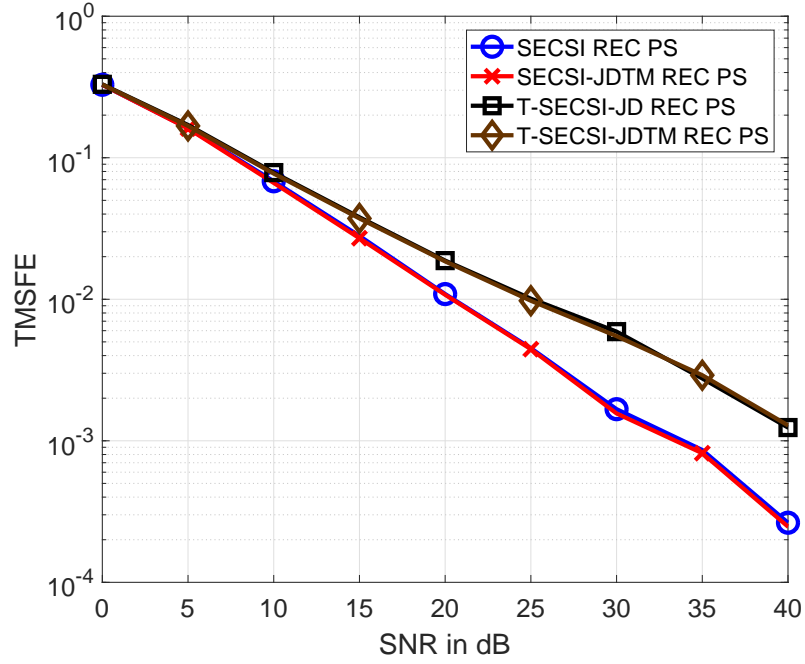


Figure 3.5.: TMSFE as a function of the SNR for a real-valued tensor with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$.

tensor. The tensor has dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$. The first factor matrix is

$$\mathbf{F}_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0.95 & 0.95 \\ 1 & 0.95 & 1 \\ 1 & 1 & 0.95 \end{bmatrix}.$$

Notice that this matrix \mathbf{F}_1 has highly correlated columns. The computation of the CP decomposition of the tensor with the 1-mode defined by the matrix \mathbf{F}_1 represents an ill-conditioned scenario. This is due to the fact that the three components are very similar to each other and therefore difficult to separate. The difficulty of this scenario is also reflected in Fig. 3.6. In this case, a higher SNR is required to achieve reasonable accuracy as compared to the uncorrelated case presented in Fig. 3.5. Moreover, the SECSI framework provides more accurate estimates of the factor matrices than T-SECSI according to Fig. 3.6.

3.2.2 Summary

The T-SECSI framework computes six independent, initial sets of estimates of the factor matrices for a 3-way tensor. Compared to the SECSI framework that also computes six initial solutions (these solutions are depicted in Fig. 3.2), T-SECSI has a lower computational complexity. The reduced computational complexity is due to the dimensionality reduction of the SMD problems.

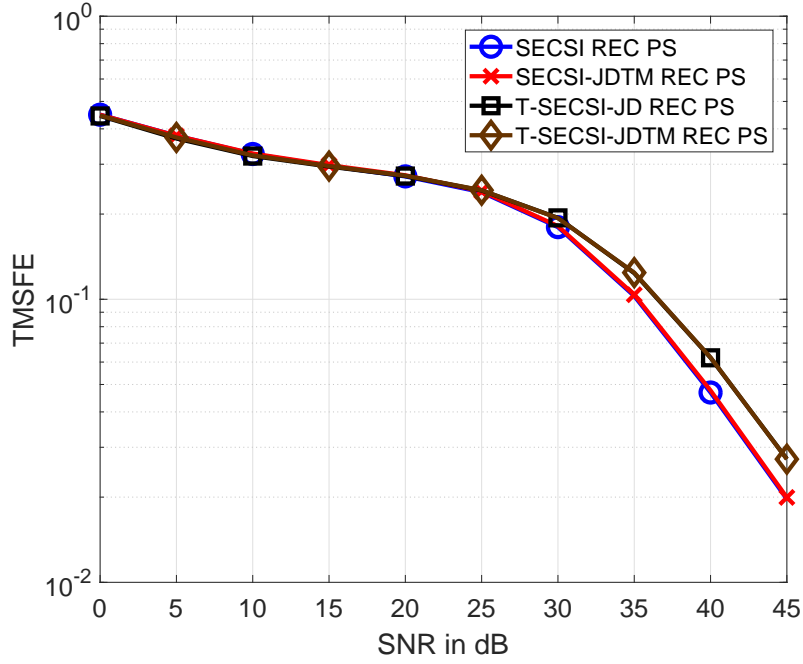


Figure 3.6.: TMSFE as a function of the SNR for a complex-valued tensor with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns.

The SECSI framework diagonalizes a set of M_n matrices jointly, where M_n is any of the tensor dimensions M_1 , M_2 , or M_3 . On the other hand, the T-SECSI framework diagonalizes a set of R matrices jointly, where R is the tensor rank. The reduced computational complexity becomes more pronounced with the increase of the tensor dimensions. The final solution is chosen in a subsequent step using the same heuristics as in the SECSI framework. Our simulation results show that if the tensor dimensions exceed the tensor rank, i.e., $M_n > R$, $\forall n = 1, 2, 3$ the T-SECSI framework has same performance as the SECSI framework. The T-SECSI framework is even more robust in critical scenarios. However, if the tensor dimensions are not larger than the tensor rank i.e., $M_n \leq R$, T-SECSI has a lower accuracy than SECSI. This variation of the performance accuracy based on the tensor dimensions can be explained by the performance analysis of the truncated HOSVD [BCS⁺16]. The authors of [BCS⁺16] show that the accuracy of the tensor representation by the truncated HOSVD depends directly on the tensor dimensions. The noise reduction is more effective for larger tensor dimensions. Hence, the truncated core tensor that is diagonalized via SMDs contains less noise. To this end, only a few slices are required to diagonalize the truncated core tensor when it is properly estimated. On the other hand, if the truncated core tensor is noise corrupted, the increased number of slices provides an advantage in terms of the accuracy. Therefore, we recommend the T-SECSI framework for the computation of the CP decomposition of low rank tensors with tensor dimensions larger than the tensor rank. It offers the same accuracy as the original SECSI framework with a reduced computational complexity.

3.3 Semi-Algebraic framework for the approximate CP decomposition via Non-Symmetric Simultaneous matrix diagonalization (NS-SECSI)

The algorithms for the computation of the CP decomposition based on an SMD including [DL05], [RH08], [LA11], [RH13a], and [RSH12] consider symmetric SMDs [FG06, LA10]. In this section, we propose a semi-algebraic framework for approximate CP decompositions via non-symmetric SMDs [NHT⁺16] to reduce the number of SMDs. This extension of the original SECSI framework we denote by NS-SECSI (Non-Symmetric SECSI). Moreover, we consider two different algorithms to calculate the non-symmetric SMDs, the TEDIA (Tensor DIagonalization) algorithm [TPC15] and an extended version of the IDIEM (Improved DIagonalization using Equivalent Matrices) algorithm [CB12], [CKM⁺14] that provides a closed-form solution for the non-symmetric SMD problem. In this work, we consider the computation of the CP decomposition for a three-way tensor. It is easy to generalize this concept to higher order tensors by combining the presented SECSI framework with generalized unfoldings as discussed in [RSH12].

The SECSI framework computes an approximate CP decomposition for a rank R tensor $\mathcal{X}_0 \in \mathbb{C}^{M_1 \times M_2 \times M_3}$. Note that the SECSI framework converts the CP decomposition into a HOSVD as a first step. In the following step, it diagonalizes the core tensor multiplied by one of the unitary matrices as shown in (3.4). In the final step, a final solution is selected from all of the available solutions. For the NS-SECSI framework as for the T-SECSI framework (see Section 3.2), we propose to diagonalize the truncated core tensor in equation (3.3) directly [NHT⁺16]. For the diagonalization along the 3-mode, NS-SECSI computes non-symmetric SMD of (3.9), whereas the original SECSI framework diagonalizes equation (3.4) based on symmetric SMDs. The NS-SECSI framework diagonalizes the truncated core tensor in a similar way as in the T-SECSI framework. However, NS-SECSI solves non-symmetric SMD problems, whereas T-SECSI solves symmetric SMD problems.

In order to obtain the set of matrices that we can use for the non-symmetric SMD, the truncated core tensor has to be sliced. Therefore, we multiply with a transpose of the pinning vector \mathbf{e}_r along the 3-mode. To this end, we define each of the corresponding slices as $\mathcal{S}_{(:, :, r)}^{[s]} = \mathcal{S}^{[s]} \times_3 \mathbf{e}_r^T$ and $\mathcal{T}_{3(:, :, r)} = \mathcal{T}_3 \times_3 \mathbf{e}_r^T$ for the left and right hand side of equation (3.9) and $r = 1, \dots, R$.

The described slicing of the truncated core tensor results in the following set of equations,

$$\mathbf{T}_1^{-1} \cdot \mathcal{S}_{(:, :, r)}^{[s]} \cdot \mathbf{T}_2^{-1} = \mathcal{T}_{3(:, :, r)} = \text{diag}(\mathbf{T}_{3(r, :)}), \quad r = 1, 2, \dots, R. \quad (3.12)$$

Equation (3.12) represents a non-symmetric SMD problem. Note that we have a set of R equations instead of the M_3 ($M_3 \geq R$) equations of the original SECSI framework, which reduces the computational complexity of the non-symmetric SMD. In this NS-SECSI framework, we use algorithms for the non-symmetric SMD, which are presented later in this section. Thereby, an estimate of the matrices \mathbf{T}_1 and \mathbf{T}_2 is obtained from the left-transform matrix and the right-transform matrix, respectively. The matrix \mathbf{T}_3 is calculated from the diagonal elements of the resulting tensor \mathcal{T}_3 .

Finally, from the knowledge of these three matrices, the factor matrices of the CP decomposition can be estimated, which is our final goal. From the link between the HOSVD and the CP decomposition in equation (3.2), we get

$$\hat{\mathbf{F}}_{1,\text{I}} = \mathbf{U}_1^{[s]} \cdot \mathbf{T}_1, \quad \hat{\mathbf{F}}_{2,\text{I}} = \mathbf{U}_2^{[s]} \cdot \mathbf{T}_2, \quad \text{and} \quad \hat{\mathbf{F}}_{3,\text{I}} = \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3.$$

The two additional tensor modes can be exploited such that two more sets of factor matrices are estimated. Accordingly, the core tensor is sliced along its 1-mode and 2-mode, and then diagonalized via non-symmetric SMDs. Therefore, we get three sets of estimated factor matrices $\hat{\mathbf{F}}_{1,\text{I}}, \hat{\mathbf{F}}_{1,\text{II}}, \hat{\mathbf{F}}_{1,\text{III}}, \hat{\mathbf{F}}_{2,\text{I}}, \hat{\mathbf{F}}_{2,\text{II}}, \hat{\mathbf{F}}_{2,\text{III}}, \hat{\mathbf{F}}_{3,\text{I}}, \hat{\mathbf{F}}_{3,\text{II}},$ and $\hat{\mathbf{F}}_{3,\text{III}}$. These sets of estimated factor matrices from a noisy observation \mathcal{X} are illustrated in Fig. 3.7.

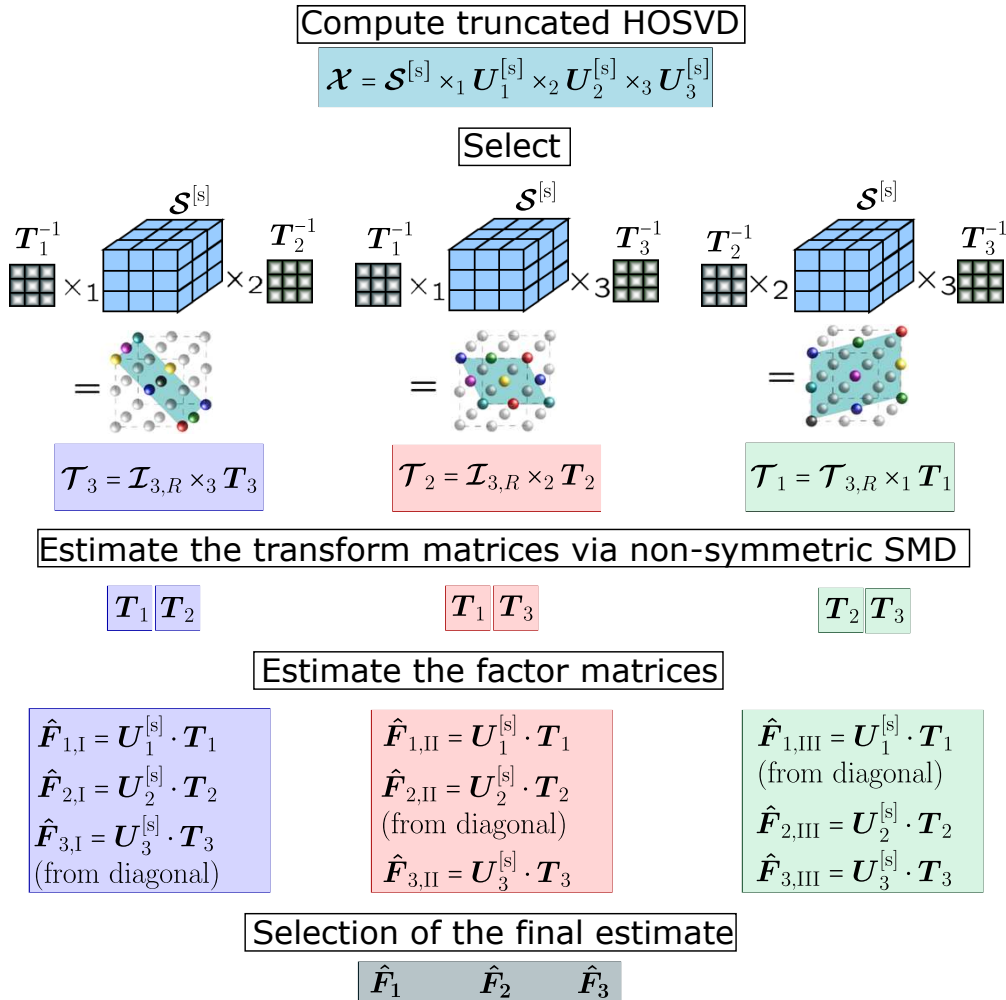


Figure 3.7.: The NS-SECSI framework for the computation of the approximate CP decomposition of a tensor $\mathcal{X} \in \mathbb{C}^{M_1 \times M_2 \times M_3}$.

From the N sets (for a N -way tensor) of estimated factor matrices different combinations can be selected, while searching for the best available solution. The heuristics for NS-SECSI are not

different than the SECSI heuristics [RH13a]. However, all together NS-SECSI has a reduced computational complexity compared to SECSI. This reduction is threefold. For an N -way tensor, there is a computational complexity reduction due to solving only N SMDs (as compared to $2N$ for SECSI), solving smaller SMD problems containing R matrices ($M_n \geq R, \forall n = 1, \dots, N$), and searching for the final solution over N sets of initial solutions ($2N$ for SECSI).

For the computation of the non-symmetric SMDs, we propose the algorithms TEDIA [TPC15] and NS-IDIEM (Non-Symmetric-Improved Diagonalization using Equivalent Matrices) [CKM⁺14]. The goal of TEDIA is to find non-orthogonal matrices $\mathbf{A}_L \in \mathbb{C}^{R \times R}$ and $\mathbf{A}_R \in \mathbb{C}^{R \times R}$ that diagonalize the set of matrices $\mathbf{M}_k \in \mathbb{C}^{R \times R}$, resulting in a set of diagonal matrices $\mathbf{D}_k \in \mathbb{C}^{R \times R}$, for $k = 1, 2, \dots, K$.

$$\mathbf{D}_k = \mathbf{A}_L^{-1} \cdot \mathbf{M}_k \cdot \mathbf{A}_R^{-1}, \quad \forall k = 1, 2, \dots, K. \quad (3.13)$$

Note that the matrices \mathbf{A}_L , \mathbf{A}_R , \mathbf{D}_k and \mathbf{M}_k , correspond to \mathbf{T}_1 , \mathbf{T}_2 , $\mathcal{T}_{3(.,.,r)}$ and $\mathcal{S}_{(.,.,r)}^{[s]}$ in equation (3.12), respectively. TEDIA does not try to minimize the off diagonal elements but rather to achieve a block-revealing condition, ideally leading to a diagonalized tensor. The algorithm is based on a search for elementary rotations that are applied to the matrices \mathbf{A}_L and \mathbf{A}_R and minimize the off-diagonal elements of \mathbf{M}_k based on a damped Gauss-Newton method. The TEDIA algorithm can be implemented in either a sequential or a parallel fashion and its main computational complexity comes from the different sweeps and the calculation of the Hessian matrix.

Although the IDIEM algorithm [CB12] was initially proposed for symmetric approximate diagonalizations, it can deal with a non-symmetric problem as well [CKM⁺14]. IDIEM provides an approximate closed-form solution for the minimization of the following so-called direct LS cost function

$$\sum_{k=1}^K \|\mathbf{M}_k - \mathbf{A}_L \mathbf{D}_k \mathbf{A}_R\|_{\mathbb{F}}^2, \quad (3.14)$$

where the matrices \mathbf{A}_L and \mathbf{A}_R of size $R \times R$ are the left and right-transform matrix, respectively. The inverse of these two matrices diagonalize the set of matrices $\mathbf{M}_k \in \mathbb{C}^{R \times R}$, resulting in a set of diagonal matrices $\mathbf{D}_k \in \mathbb{C}^{R \times R}$, for $k = 1, 2, \dots, K$. The NS-IDIEM algorithm computes in the first step the R most representative matrices \mathbf{R}_r from the K original target matrices \mathbf{M}_k in an LS sense. The matrix \mathbf{R}_r represents the r -th eigenvector (\mathbf{r}_r) of $\mathbf{M} = \sum_{k=1}^K \text{vec}(\mathbf{M}_k) \text{vec}(\mathbf{M}_k)^H$ by means of an inverse vector operation, i.e., $\mathbf{R}_r = \text{vec}^{-1}(\mathbf{r}_r)$. In the second step, a column-wise scrambling is performed such that a new set of R matrices is build from the r -th columns of the matrices \mathbf{R}_r , i.e., $\mathbf{R}'_r = [\mathbf{R}_1^{(r)}, \mathbf{R}_2^{(r)}, \dots, \mathbf{R}_R^{(r)}]$. In the third step, the left-transform matrix is estimated using the two most representative matrices $(\mathbf{V}_1, \mathbf{V}_2)$ from the R matrices \mathbf{R}'_r in an LS sense. The matrices $\mathbf{V}_1 = \text{vec}^{-1}(\mathbf{r}'_1)$ and $\mathbf{V}_2 = \text{vec}^{-1}(\mathbf{r}'_2)$ are built from the eigenvectors \mathbf{r}'_1 and \mathbf{r}'_2 of the matrix $\mathbf{R}' = \sum_{k=1}^K \text{vec}(\mathbf{R}'_k) \text{vec}(\mathbf{R}'_k)^H$ corresponding to the 2 largest eigenvalues. The left-transform matrix is then computed as $\mathbf{A}_L = \mathbf{V}_2 \mathbf{V}_1^{-1}$. Since the algorithm does not assume any explicit link between the two transform matrices, the right-transform matrix is simply obtained by using the rows of the R matrices \mathbf{R}_r instead of the columns in the second step. We propose to use this algorithm,

NS-IDIEM because it is not iterative and therefore very fast and computationally efficient. Its closed-form solution is a very practical choice for the non-symmetric SECSI framework. Hence, NS-SECSI based on the NS-IDIEM algorithm provides a closed-form solution for the computation of an approximate low-rank CP decomposition.

3.3.1 Simulation Results

In this section, we evaluate the performance of the proposed non-symmetric extension of the SECSI framework with its two implementations based on the TEDIA algorithm and the NS-IDIEM algorithm. We denote these two extensions by NS-SECSI-TEDIA and NS-SECSI-NS-IDIEM, respectively. Since the SECSI framework has already been compared to the state-of-the-art algorithms for various scenarios, we only compare our proposed framework to the original SECSI framework [RH13a,RH08]. Therefore, we provide an accuracy comparison and computational time comparison of NS-SECSI-TEDIA, NS-SECSI-NS-IDIEM, SECSI, and T-SECSI from Section 3.2 (we consider the two implementations of T-SECSI, T-SECSI-JD and T-SECSI-JDMT).

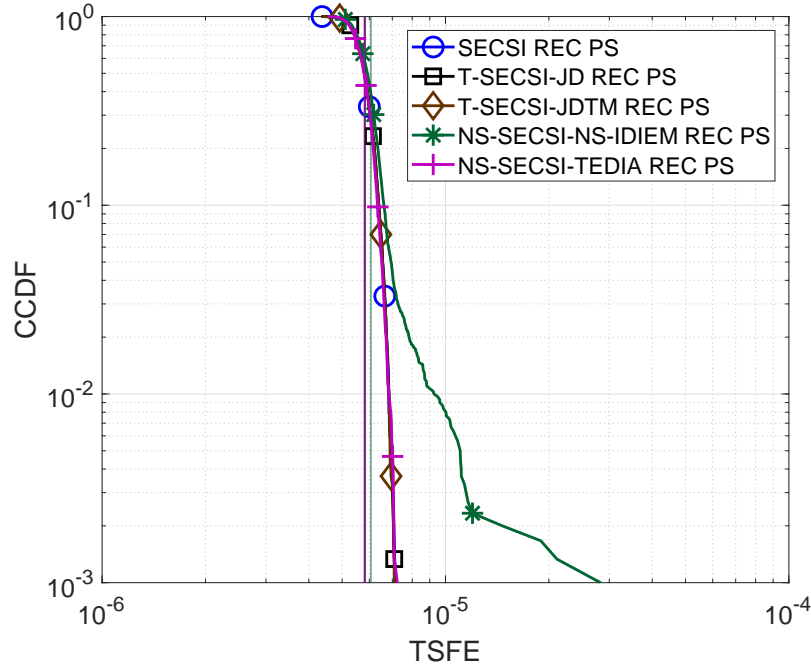


Figure 3.8.: CCDF of the TSFE for a real-valued tensor with dimensions $40 \times 40 \times 40$, tensor rank $R = 3$, and SNR = 25 dB.

For simulation purposes, tensors of size $M_1 \times M_2 \times M_3$ with tensor rank R have been designed according to the CP decomposition in equation (3.1). The factor matrices \mathbf{F}_1 , \mathbf{F}_2 , and \mathbf{F}_3 have i.i.d. zero mean Gaussian distributed random entries with variance one for real-valued tensors or ZMCSCG random entries with variance one for complex-valued tensors. Moreover, if we want the tensors to have correlated factor matrices, we add correlation via a correlation matrix as shown

in equation (3.10). Finally, we add i.i.d. zero mean Gaussian (or ZMCSCG for complex-valued tensors) noise with variance σ_N^2 to the synthetic low-rank data. The resulting SNR for the noisy tensor $\mathcal{X} = \mathcal{X}_0 + \mathcal{N}$ is $\text{SNR} = 10 \log_{10} (\|\mathcal{X}_0\|_H^2 / \|\mathcal{N}\|_H^2)$ in dB. We use the TSFE defined in equation (3.11) as an accuracy measure. The vertical lines correspond to the mean value of the TSFE.

In Fig. 3.8, we depict the CCDF of the TSFE for a real-valued tensor with uncorrelated factor matrices, for $\text{SNR} = 25$ dB and 3000 realizations. The tensor has dimensions $40 \times 40 \times 40$ and tensor rank $R = 3$. All algorithms in Fig. 3.8 use the REC PS heuristic. Notice that all of the algorithms are equally accurate. Only NS-SECSI-NS-IDIEM has a few additional outliers as compared to the remaining algorithms.

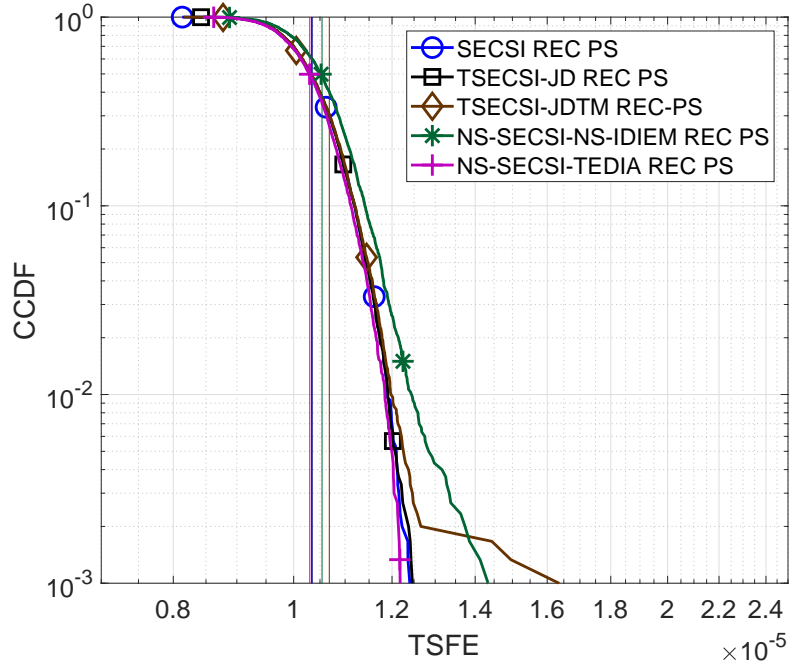


Figure 3.9.: CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and $\text{SNR} = 25$ dB.

In Fig. 3.9, we present the CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and $\text{SNR} = 25$ dB. The curves depicted in Fig. 3.9 result from 3000 realizations. Similar to the real-valued case depicted in Fig. 3.8, all algorithms have a similar accuracy in terms of the TSFE. The frameworks T-SECSI-JDMT and NS-SECSI-NS-IDIEM have several outliers as compared to the other algorithms. However, the difference in the mean TSFE is very small, considering the values on the horizontal axis. Note that in our simulation results, we have noticed outliers of T-SECSI-JDMT as compared to T-SECSI-JD only in the complex-valued cases.

Moreover, in Figs. 3.10 and 3.11 we consider a complex-valued tensor with dimension $30 \times 30 \times 30$ and tensor rank $R = 3$. The first factor matrix has mutually correlated columns with correlation factor $\rho_1 = 0.9$ (see equation (3.10)). In Fig. 3.10, we depict the CCDF of the TSFE for

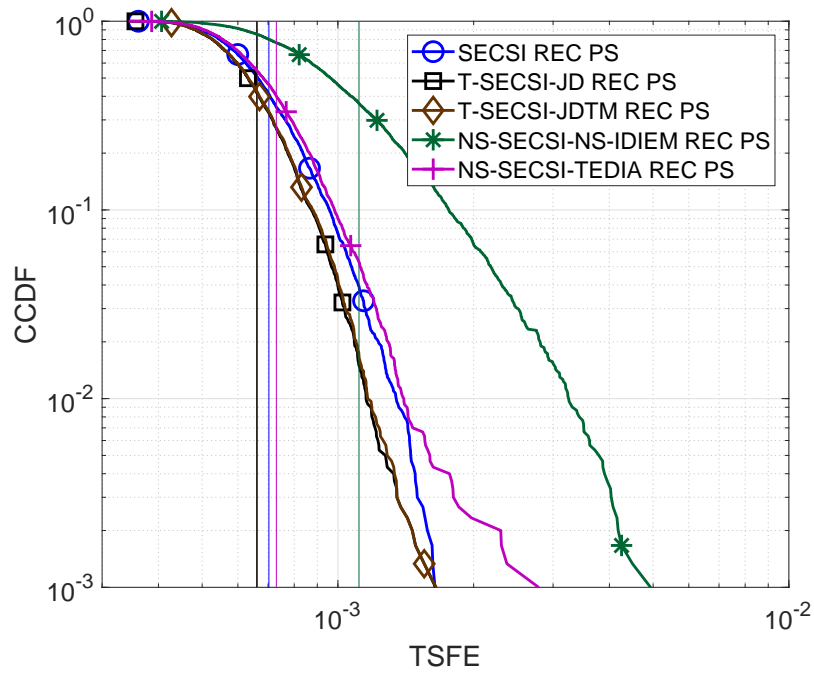


Figure 3.10.: CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 10 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.

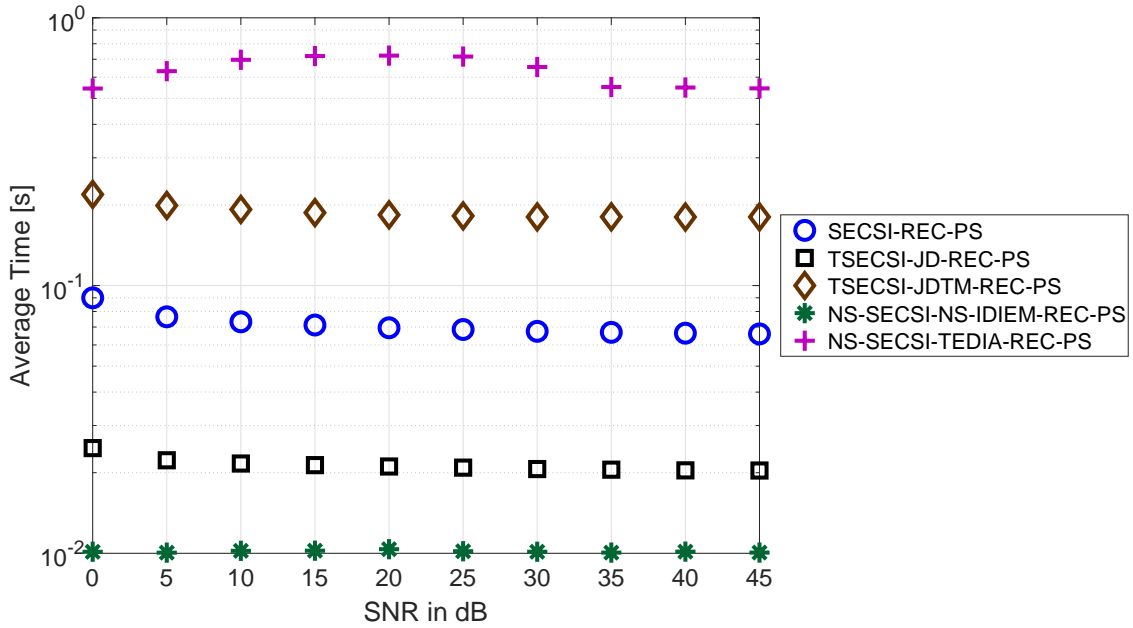


Figure 3.11.: Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $30 \times 30 \times 30$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.

SNR = 10 dB. Observe that T-SECSI has a higher accuracy as compared to SECSI even for small SNRs. NS-SECSI-TEDIA and NS-SECSI-NS-IDIEM have a lower accuracy than T-SECSI and SECSI. In Fig. 3.10, the reduced accuracy of NS-SECSI-NS-IDIEM is more evident. However, NS-SECSI-NS-IDIEM provides a closed-form solution that can be efficiently computed regardless of how large the tensor dimensions are. To confirm the efficiency of the computation of the CP decomposition using NS-SECSI-NS-IDIEM, we have investigated the average required time for the computation of the CP decomposition. Therefore, in Fig. 3.11 we depict the average time in seconds required for the computation of the CP decompositions as a function of the SNR. First, observe that there is no significant dependence between the averaged required time and the SNR. Next, NS-SECSI-NS-IDIEM outperforms the rest of the algorithms with respect to the computational time. On the other hand, NS-SECSI-TEDIA requires more computational time than all of the considered algorithms. Moreover, we observe a gain with respect to the computational time for T-SECSI-JD in comparison with SECSI. Both frameworks T-SECSI-JD and SECSI utilize the same JD algorithm [FG06]. Therefore, this gain is only due to the dimensionality reduction achieved by using T-SECSI-JD. Note that the computation time is not always a direct measure of the computational complexity. For instance, T-SECSI-JDTM has less computational complexity than SECSI as explained in Section 3.2, but it requires more computational time than SECSI. The computational time also depends on the implementation of the algorithms.

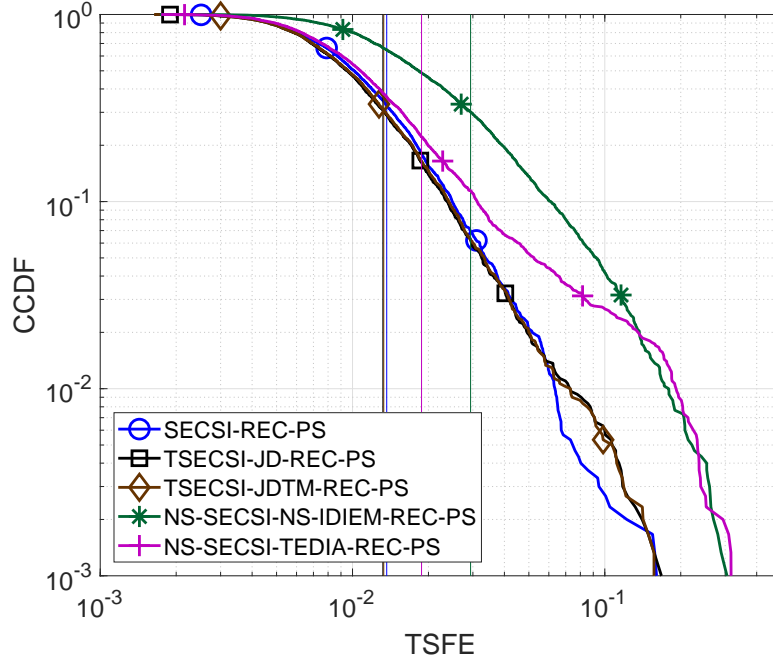


Figure 3.12.: CCDF of the TSFE for a complex-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 15 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.

Furthermore, in Figs. 3.12 and 3.13 we analyze the performance of the frameworks SECSI, T-SECSI, and NS-SECSI for a complex-valued tensor with dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$.

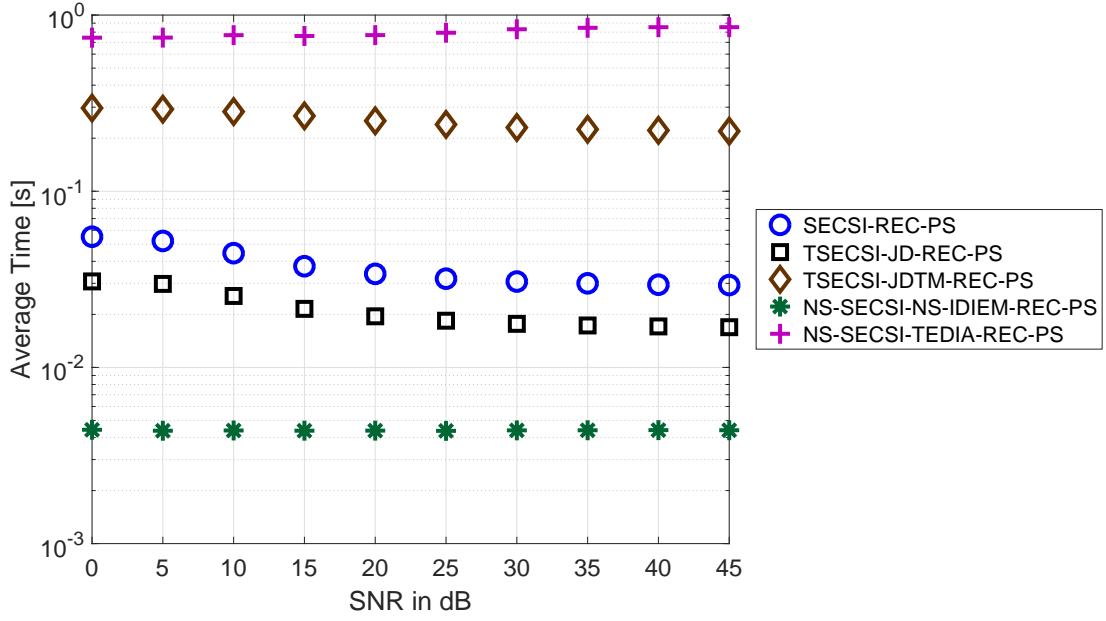


Figure 3.13.: Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.

The first factor matrix \mathbf{F}_1 has mutually correlated columns with correlation factor of $\rho_1 = 0.9$. Also, here, we observe that with NS-SECSI we sacrifice accuracy, but we reduce the computational complexity. Moreover, with NS-SECSI-NS-IDIEM we reduce the computational time required to compute the CP decomposition. Note that in our simulation results we have observed higher accuracy for real-valued tensors as compared to complex-valued tensors when decomposing these tensors using NS-SECSI-TEDIA. Further results are also available in [NHT⁺16].

Finally, in Fig. 3.14, we compare the different heuristics for the NS-SECSI framework. In Fig. 3.14, we depict the CCDF of the TSFE for a real-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 20 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of $\rho_1 = 0.9$. We consider 3000 realization of this scenario and the heuristics REC PS, BM, and RES. As for the original SECSI framework [RH13a], we observe the highest accuracy for BM that is the most computational expensive solution. Moreover, we observe the lowest accuracy for RES that is computationally much cheaper than BM. The heuristics REC PS offers high accuracy for an acceptable complexity.

3.3.2 Summary

In this section, we have presented an extension of the SECSI framework, by solving non-symmetric SMDs based on the TEDIA and the NS-IDIEM algorithm. The NS-SECSI-NS-IDIEM framework offers a very fast approximation for the CP decomposition with a reasonable accuracy. Notice

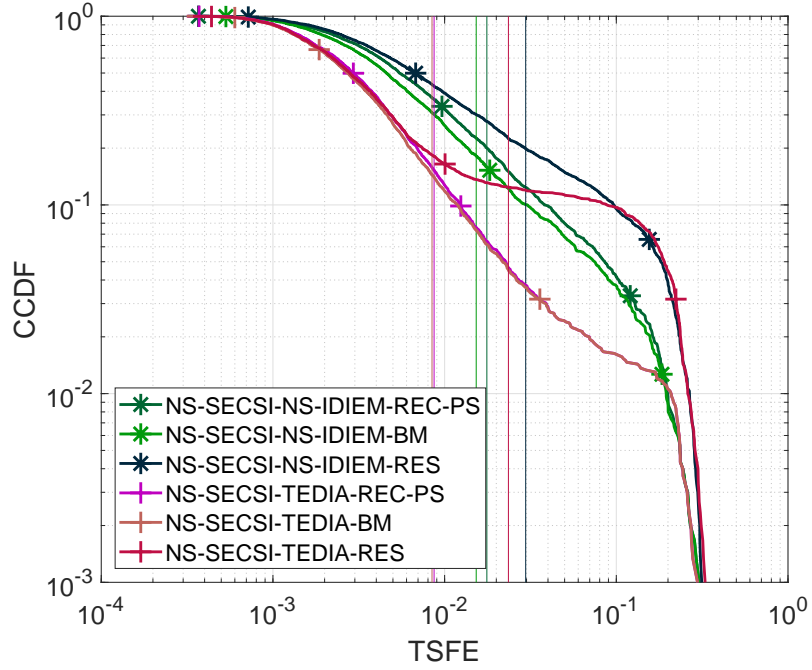


Figure 3.14.: CCDF of the TSFE for a real-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 20 dB. The first factor matrix \mathbf{F}_1 has mutually correlated columns with a correlation coefficient of 0.9.

that NS-SECSI-NS-IDIEM provides a closed-form solution for the CP decomposition, since the non-symmetric SMDs can be calculated in closed-form [CKM⁺14], [TPC15]. NS-SECSI-TEDIA is more accurate than NS-SECSI-NS-IDEM, in particular, it is more accurate for real-valued tensors. As for the T-SECSI framework, the accuracy of the NS-SECSI framework depends on the tensor dimensions. For small tensor dimensions (i.e., comparable to the tensor rank) the core tensor obtained from a truncated HOSVD is noisy [BCS⁺16]. Therefore, for small tensor dimensions, the diagonalization algorithms benefit from the larger number of slices in the original SECSI frameworks. In contrast to the original framework for an N -way tensor, NS-SECSI calculates N sets of non-symmetric SMDs instead of $2N$ sets of symmetric SMDs for a smaller number of matrices ($R \leq M_n, \forall n = 1, \dots, N$). The computational advantages provided by the truncations become more pronounced as the tensor size increases. Therefore, we recommend NS-SECSI-NS-IDIEM and the REC PS heuristic for the computation of the CP decomposition for tensors with large dimension $M_n > R, \forall n = 1, \dots, N$. Moreover, we recommend NS-SECSI-NS-IDIEM for applications where a fast and an efficient solution is required. Note that, we can increase the accuracy and thereby reduce the computational complexity of NS-SECSI-TEDIA if we use the transform matrices resulting from the NS-IDIEM algorithm to initialize the TEDIA algorithm.

3.4 Semi-Algebraic framework for the approximate CP decomposition via Simultaneous matrix diagonalization for Symmetric Tensors (S-SECSI)

A tensor is symmetric if its entries are invariable under the permutation of its modes [Kol15, Com14]. We can differentiate between a fully symmetric tensor and partially symmetric tensor depending on the number of modes that do not change under permutation. A 3-way tensor $\mathcal{X}_0 \in \mathbb{C}^{M \times M \times M_3}$ is partially symmetric (symmetric along the 1-mode and 2-mode) if [CGLM08, Kol15]

$$\mathcal{X}_{0(i,j,k)} = \mathcal{X}_{0(j,i,k)}, \forall i, j = 1, \dots, M, \quad k = 1, \dots, M_3.$$

On the other hand, a 3-way tensor $\mathcal{X}_0 \in \mathbb{C}^{M \times M \times M}$ is fully symmetric if

$$\mathcal{X}_{0(i,j,k)} = \mathcal{X}_{0(i,k,j)} = \mathcal{X}_{0(j,i,k)} = \mathcal{X}_{0(k,i,j)} = \mathcal{X}_{0(j,k,i)} = \mathcal{X}_{0(k,j,i)}, \forall i, j, k = 1, \dots, M.$$

A review of fully symmetric tensors and their uniqueness properties is provided in [CGLM08]. An extensive discussion and some results for the typical rank of symmetric tensors are available in [SDLF⁺17]. The typical rank of symmetric tensors is equal or smaller than the typical rank of non-symmetric tensors of the same size [SDLF⁺17].

Symmetric tensors can be associated to statistics, for instance, cumulant tensors that are derivatives of the characteristic function [McC87]. Also, higher-order derivatives of multivariate functions and homogeneous polynomials can be related with symmetric tensor [McC87, Com14]. The authors of [SDLF⁺17] review applications of symmetric tensors such as speech signal separation using an array of microphones, Gaussian mixture parameter estimation, and topic modeling applications. Blind source identification applications of symmetric tensors include the estimation of directional vectors without a prior knowledge of the array manifold [Car91]. Furthermore, the authors of [DLCC07] exploit symmetric tensors for blind source identification based on fourth order cumulants.

A partially symmetric tensor $\mathcal{X}_0 \in \mathbb{C}^{M \times M \times M_3}$ has partially symmetric CP decomposition. This decomposition is also known as INDSCAL (INDividual Differences in SCALing) [CC70]

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F} \times_2 \mathbf{F} \times_3 \mathbf{F}_3, \quad (3.15)$$

where $\mathbf{F} \in \mathbb{C}^{M \times R}$ and $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$ are the factor matrices corresponding to the symmetric CP decomposition. The authors of [CC70] propose an ALS algorithm for the computation of partially symmetric CP. The proposed algorithm computes all factors separately in the same fashion as the traditional ALS algorithm for the computation of the CP decomposition defined in (3.1). When the algorithm converges, the symmetric modes are set to be explicitly equal (i.e., $\mathbf{F}_1 = \mathbf{F}_2$ from equation (3.1)) and the remaining mode \mathbf{F}_3 is computed once again. Hence, this algorithm ignores the symmetry constraints until the last step. The advantages of ignoring the symmetry

constraints is that we can use the same algorithms for the computation of the CP decomposition. The authors of [Kol15] propose an algorithm that resolves the scaling ambiguity while converting a not explicitly symmetric CP into a symmetric CP. Therefore, for the computation of the symmetric CP decomposition, we can use any other algorithm for the computation of the CP decomposition (see Section 2.2.2) and then use the algorithm proposed in [Kol15] as a second (final) step. However, the challenging symmetry constraints can be easily incorporated in the SECSI framework. In this section we propose a S-SECSI (Symmetric SECSI) for the computation of the CP decomposition of symmetric and fully symmetric tensors. Note that the authors of [RH13a] already consider symmetric cases. The S-SECSI framework proposed here provides a closed-form solution based on IDIEM and NS-IDIEM. Therefore, it represents an extension of NS-SECSI-NS-IDIEM with symmetry constraints.

As in the previously presented SECSI extensions, here we derive the S-SECSI framework for 3-way tensors. An extension to N -way tensors is straightforward. Note that an extension using the generalized unfoldings similar to the SECSI-GU framework [RSH12] can also be considered. Moreover, for derivation simplicity we assume a noiseless case. In the presence of noise all following relations still hold approximately. Let us consider the CP decomposition defined in (3.15) of a symmetric tensor $\mathcal{X}_0 \in \mathbb{C}^{M \times M \times M_3}$, where the first and the second mode are equivalent. Following the derivation of the original SECSI framework [RH13a], we start by comparing the symmetric CP decomposition and the truncated HOSVD. Thereby, we assume that $M \geq R$ and $M_3 \geq R$ even though S-SECSI as the original SECSI framework can handle degenerate cases (see Section 3.1).

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F} \times_2 \mathbf{F} \times_3 \mathbf{F}_3 = \mathcal{S}^{[s]} \times_1 \mathbf{U}^{[s]} \times_2 \mathbf{U}^{[s]} \times_3 \mathbf{U}_3^{[s]}$$

The matrices $\mathbf{F} \in \mathbb{C}^{M \times R}$ and $\mathbf{F}_3 \in \mathbb{C}^{M_3 \times R}$ correspond to the symmetric CP decomposition with rank R . The tensor $\mathcal{S}^{[s]} \in \mathbb{C}^{R \times R \times R}$ and the matrices $\mathbf{U}^{[s]} \in \mathbb{C}^{M \times R}$ and $\mathbf{U}_3^{[s]} \in \mathbb{C}^{M_3 \times R}$ correspond to the truncated HOSVD. Similar to the derivations following equation (3.2), for the core tensor $\mathcal{S}^{[s]}$, we have

$$\mathcal{S}^{[s]} = \mathcal{I}_{3,R} \times_1 \mathbf{T} \times_2 \mathbf{T} \times_3 \mathbf{T}_3.$$

Hence, the invertible transform matrices \mathbf{T} and \mathbf{T}_3 diagonalize the truncated core tensor $\mathcal{S}^{[s]}$. By slicing the tensor $\mathcal{S}^{[s]}$ along the 3-mode, we get an SMD problem given by

$$\mathcal{T}_{3(.,.,r)} = \mathbf{T}^{-1} \mathcal{S}_{(.,.,r)}^{[s]} \mathbf{T}^{-T} = \text{diag}(\mathbf{T}_{(r,.)}), \forall r = 1, \dots, R,$$

where the tensor $\mathcal{T}_3 = \mathcal{I}_{3,R} \times_3 \mathbf{T}_3 \in \mathbb{C}^{R \times R \times R}$. The above relations correspond to symmetric SMD. To solve this symmetric SMD we recommend the IDIEM algorithm [CB12] in order to achieve a closed-form solution. The IDIEM algorithm provides a closed-form solution for the transform matrix \mathbf{T} . Furthermore, we compute the matrix \mathbf{T}_3 from the diagonal elements of the tensor \mathcal{T}_3 . Using the estimated transform matrices and the link between the CP decomposition and the HOSVD, we

estimate the first set of the factor matrices.

$$\mathbf{F}_I = \mathbf{U}^{[s]} \cdot \mathbf{T} \quad \mathbf{F}_{3,I} = \mathbf{U}_3^{[s]} \cdot \mathbf{T}_3$$

At this point, we have obtained only one set of estimates by diagonalizing the 3-mode slices of the truncated core tensor $\mathcal{S}^{[s]}$. In a similar fashion, we can diagonalize the remaining modes. Thus, for the diagonalization along the 2-mode and the 1-mode, we have

$$\mathcal{T}_2 = \mathcal{S}^{[s]} \times_1 \mathbf{T}^{-1} \times_3 \mathbf{T}_3^{-1} \Leftrightarrow \mathbf{T}^{-1} \mathcal{S}_{(.,.,r)}^{[s]} \mathbf{T}_3^{-T} = \text{diag}(\mathbf{T}_{(r,.)}), \forall r = 1, \dots, R, \text{ and} \quad (3.16)$$

$$\mathcal{T}_1 = \mathcal{S}^{[s]} \times_2 \mathbf{T}^{-1} \times_3 \mathbf{T}_3^{-1} \Leftrightarrow \mathbf{T}^{-1} \mathcal{S}_{(.,.,r)}^{[s]} \mathbf{T}_3^{-T} = \text{diag}(\mathbf{T}_{(r,.)}), \forall r = 1, \dots, R, \quad (3.17)$$

respectively. For the tensors $\mathcal{T}_2 \in \mathbb{C}^{R \times R \times R}$ and $\mathcal{T}_1 \in \mathbb{C}^{R \times R \times R}$, we have $\mathcal{T}_2 = \mathcal{I}_{3,R} \times_2 \mathbf{T}$ and $\mathcal{T}_1 = \mathcal{I}_{3,R} \times_1 \mathbf{T}$. Both equations (3.16) and (3.17) represent non-symmetric SMDs and in the noise-free case as well as the case of symmetric noise (that is usually the case in many applications) represent an identical diagonalization problem, i.e., provide identical estimates of the transform matrices. Therefore, we consider only one of these SMDs. For solving this non-symmetric SMD, we recommend the non-symmetric version of the IDIEM algorithm, NS-IDIEM [CKM⁺14]. Recall that NS-SECSI utilizes also the NS-IDIEM algorithm for the computation of the transform matrices that diagonalize the core tensor $\mathcal{S}^{[s]}$. The above non-symmetric SMD results into two sets of estimates of the factor matrices because the transform matrix \mathbf{T} is estimated twice. The first estimate of the matrix \mathbf{T} is obtained from the left-transform matrix and the second one is obtained from the resulting diagonal matrices (see equation (3.16)). Thereby, we obtain the matrices \mathbf{F}_{II} , \mathbf{F}_{III} , $\mathbf{F}_{3,II}$, and $\mathbf{F}_{3,III}$ from equation (3.16). The second estimate of the factor matrix \mathbf{F}_3 can be estimated via an LS fit. Hence, with S-SECSI for a 3-way tensor we solve two SMDs, but obtain three initial estimates of the factor matrices. In Fig. 3.15, we depict the S-SECSI framework for the computation of an approximate low-rank CP decomposition of a symmetric (along the 1-mode and 2-mode) tensor $\mathcal{X} \in \mathbb{C}^{M \times M \times M_3}$ that represents a noisy observation of a low-rank tensor rank with R .

Now, let us consider a symmetric CP decomposition of a fully symmetric tensor $\mathcal{X}_0 \in \mathbb{C}^{M \times M \times M}$ with rank R .

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F} \times \mathbf{F} \times \mathbf{F}$$

In this case, all three diagonalization problems $\mathcal{T}_1 = \mathcal{S}^{[s]} \times_2 \mathbf{T}^{-1} \times_3 \mathbf{T}^{-1}$, $\mathcal{T}_2 = \mathcal{S}^{[s]} \times_1 \mathbf{T}^{-1} \times_3 \mathbf{T}^{-1}$, and $\mathcal{T}_3 = \mathcal{S}^{[s]} \times_1 \mathbf{T}^{-1} \times_2 \mathbf{T}^{-1}$ are equivalent. The three diagonalization problems differ only in the index along which mode the diagonalization is performed, i.e., along the 1-mode, the 2-mode, or the 3-mode, respectively. Also, all three diagonalization problems represent a symmetric SMD. Thus, using the IDIEM algorithm we solve only one of these SMDs and obtain one set of estimates of the factor matrices. The visualization of S-SECSI for fully symmetric tensor is comparable to the first branch of S-SECSI for symmetric tensors depicted in Fig. 3.15. The S-SECSI framework for a

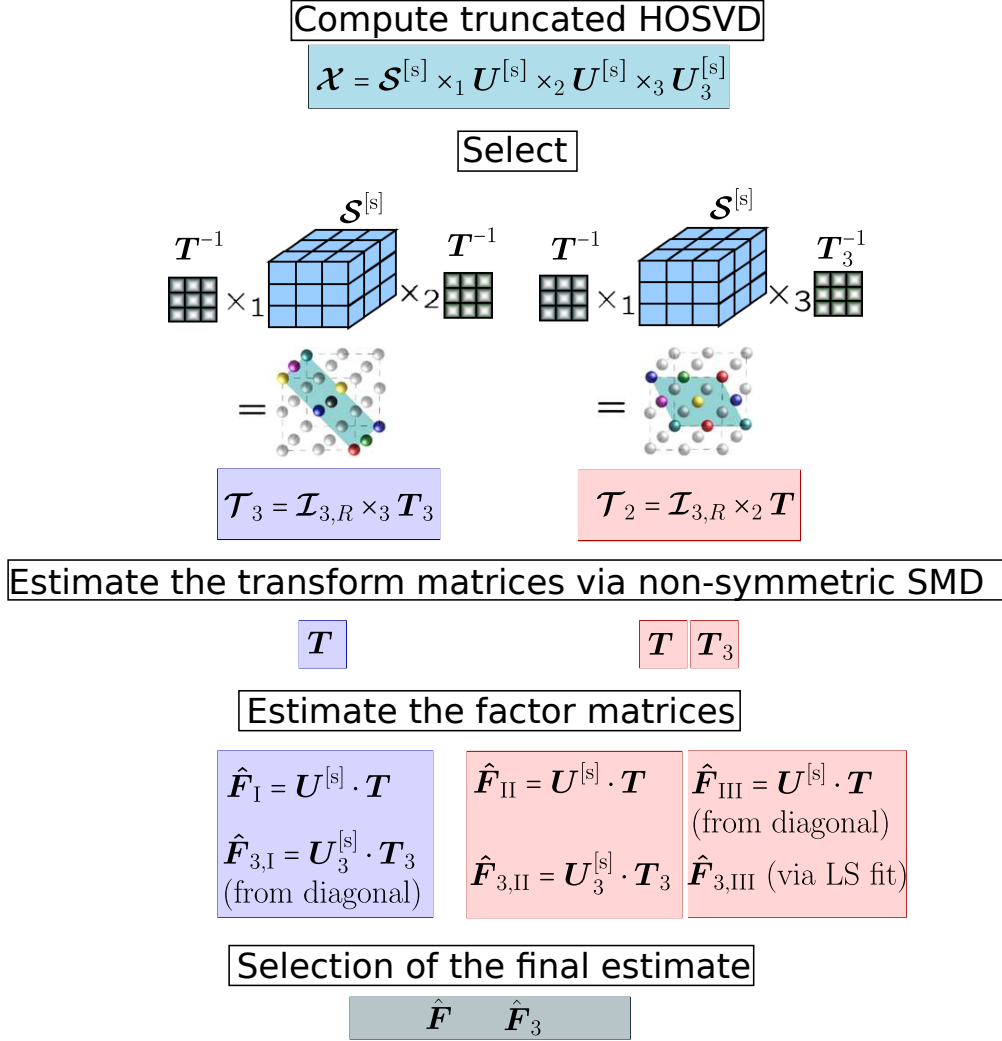


Figure 3.15.: The S-SECSI framework for the computation of the approximate CP decomposition of a symmetric tensor $\mathcal{X} \in \mathbb{C}^{M \times M \times M_3}$. The symmetry is with respect to the 1-mode and the 2-mode.

3-way tensor for fully symmetric tensor solves one SMD resulting in one initial set of estimates of the factor matrices.

As for SECSI, T-SECSI, and NS-SECSI, the final estimates of the factor matrices for S-SECSI of symmetric tensors (the fully symmetric case provides only one (final) solution) can be selected based on the different criteria, namely BM, REC PS, CON PS, and RES that are introduced at the beginning of this chapter [RH13a]. The S-SECSI framework exploits the closed-form algorithms IDIEM and NS-IDIEM to compute the SMDs. Therefore, the S-SECSI framework provides a closed-form solution for the approximate computation of the symmetric CP decomposition.

3.4.1 Simulation Results

To analyze the performance of the S-SECSI framework, we compare S-SECSI with NS-SECSI-NS-IDIEM (see Section 3.3) [NHT⁺16] and the original SECSI framework [RH13a]. Both frameworks NS-SECSI and SECSI ignore the symmetry constraints. In the comparisons, we include the original SECSI framework as a benchmark algorithm, because it has already been compared to other state-of-the-art algorithms in the past [RH08, RSH12, RH13a]. Moreover, we include NS-SECSI-NS-IDIEM in the comparisons because both extensions S-SECSI and NS-SECSI-NS-IDIEM exploit the same diagonalization algorithm, IDIEM [CB12]. Therefore, the comparison of S-SECSI and NS-SECSI-NS-IDIEM intends to show the importance of taking the symmetry constraints into account while computing the symmetric CP decomposition. We compare S-SECSI, NS-SECSI, and SECSI by means of the TSFE defined in (3.11) and the averaged required time for the computation of the CP decomposition. We indicate the heuristic used for the selection of the final solution in the legends.

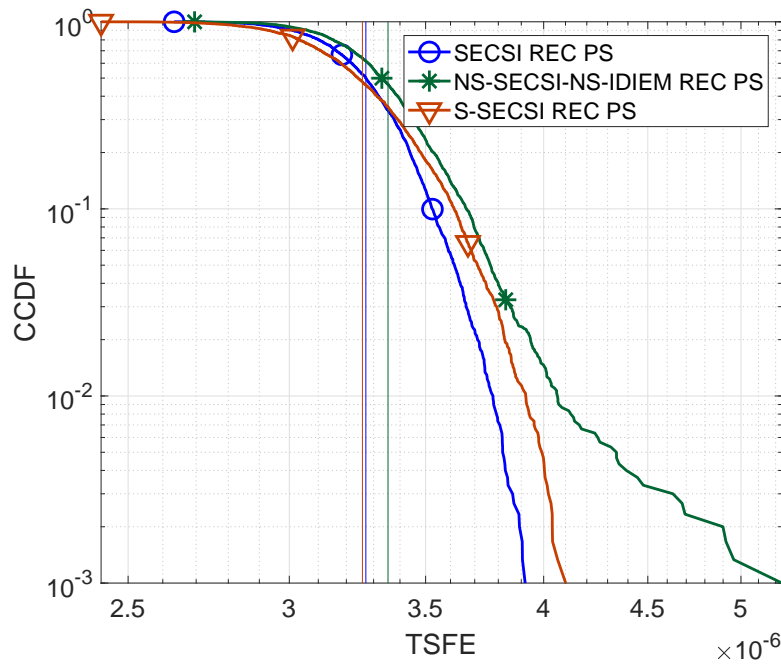


Figure 3.16.: CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode, i.e., $\mathbf{F}_1 = \mathbf{F}_2$ from equation (3.1).

For simulation purposes, we generate rank R tensors that have two symmetric modes according to equation (3.15). The factor matrices have either i.i.d. zero mean Gaussian random entries or ZMCSCG random entries depending on whether we generate real-valued or complex-valued tensor, respectively. Additional correlation between the columns of the factor matrices is added according to (3.10). A noisy observation is obtained by adding zero mean i.i.d. random Gaussian noise (or

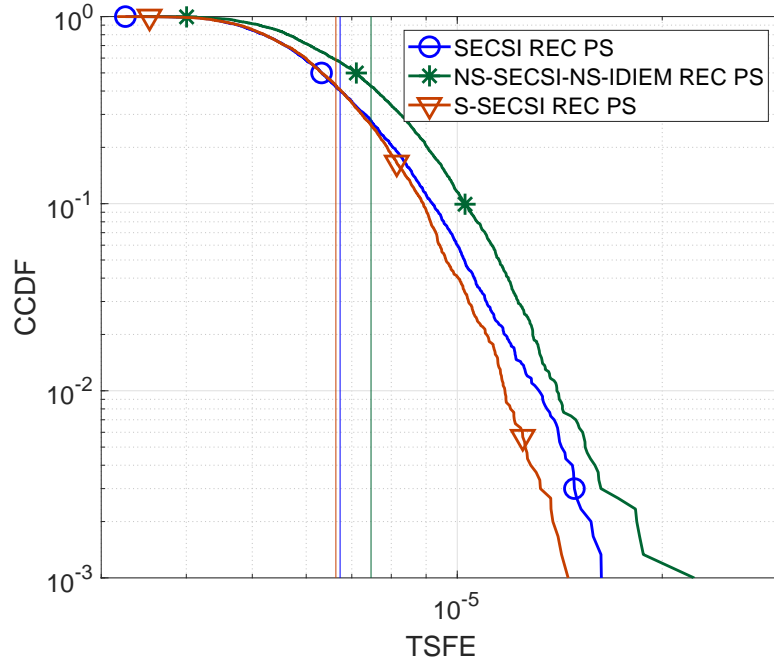


Figure 3.17.: CCDF of the TSFE for a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient 0.9.

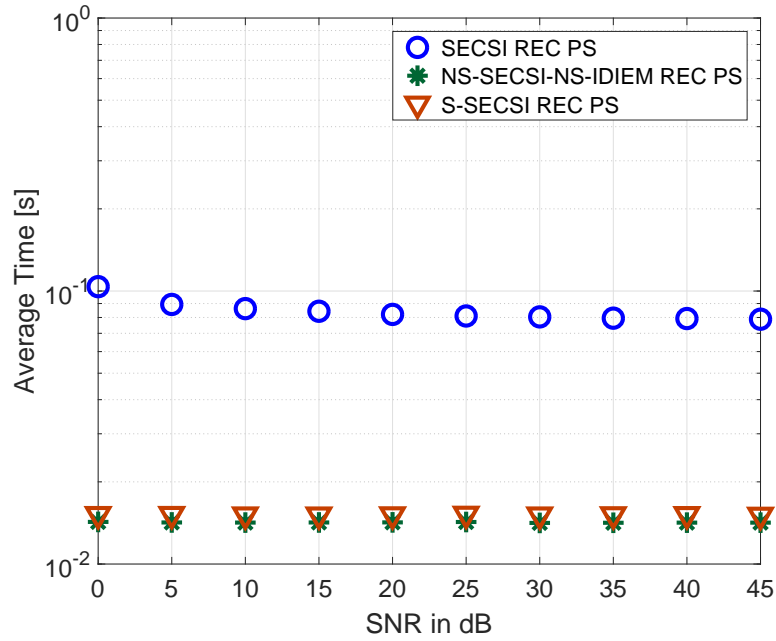


Figure 3.18.: Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $30 \times 30 \times 30$ and tensor rank $R = 3$. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.

ZMCSG noise for complex valued tensors) with variance σ_N . Therefore, the resulting SNR in dB is equal to $10 \log_{10} (\|\mathbf{x}_0\|_H^2 / \|\mathcal{N}\|_H^2)$.

In Fig. 3.16, we depict the CCDF of the TSFE for a complex-valued tensor and SNR = 30 dB. The tensor has dimensions $30 \times 30 \times 30$ and tensor rank $R = 3$. The averaged required time for the computation of the CP decomposition is 0.06 s, 0.0101 s, and 0.01 s for SECSI, NS-SECSI-NS-IDIEM, and S-SECSI, respectively. The advantage of considering the symmetry constraint is obvious in Fig. 3.16. S-SECSI has a higher accuracy than NS-SECSI-NS-IDIEM, while requiring the same time to compute the CP decomposition. S-SECSI has the same accuracy performance as SECSI. Hence, it compensates even the loss of accuracy due to the closed-form solution that was observed in the NS-SECSI-NS-IDIEM framework (see Section 3.3).

Next, we consider a complex-valued tensor with dimensions $30 \times 30 \times 30$, tensor rank $R = 3$. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation factor $\rho_3 = 0.9$. In Fig. 3.17, we depict the CCDF of the TSFE, for SNR = 30 dB. Similar to the uncorrelated scenario previously presented (see Fig. 3.16), we gain from the consideration of the symmetry constraints. This is reflected by the improved accuracy performance of S-SECSI as compared to NS-SECSI-NS-IDIEM. Moreover, Fig. 3.18 depicts the average required time for the computation of the CP decomposition for SNR values between 0 dB and 45 dB. With the S-SECSI framework we achieve the same accuracy as SECSI while requiring less computational time. Moreover, S-SECSI is a closed-form solution and it has less computational complexity than SECSI.

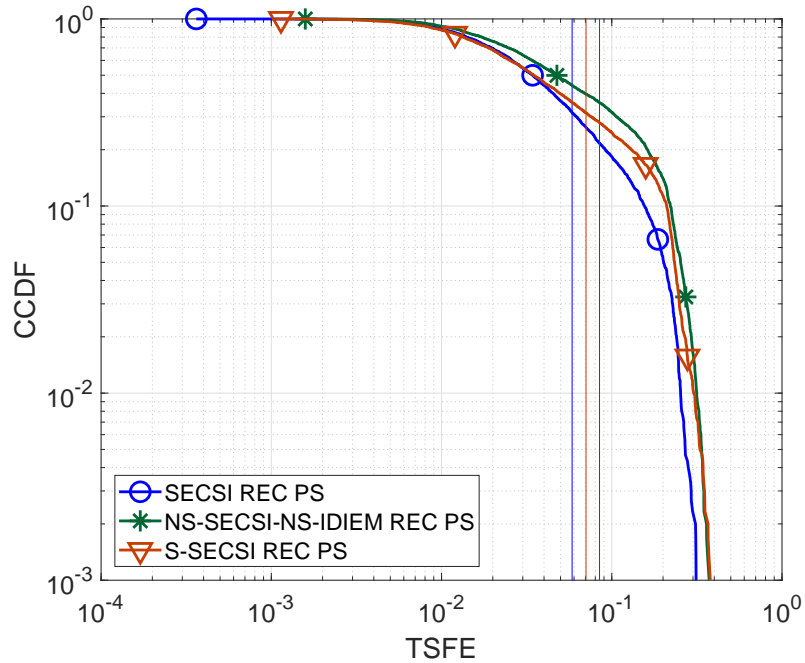


Figure 3.19.: CCDF of the TSFE for a real-valued tensor with dimensions $7 \times 7 \times 4$, tensor rank $R = 3$, and SNR = 35 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns.

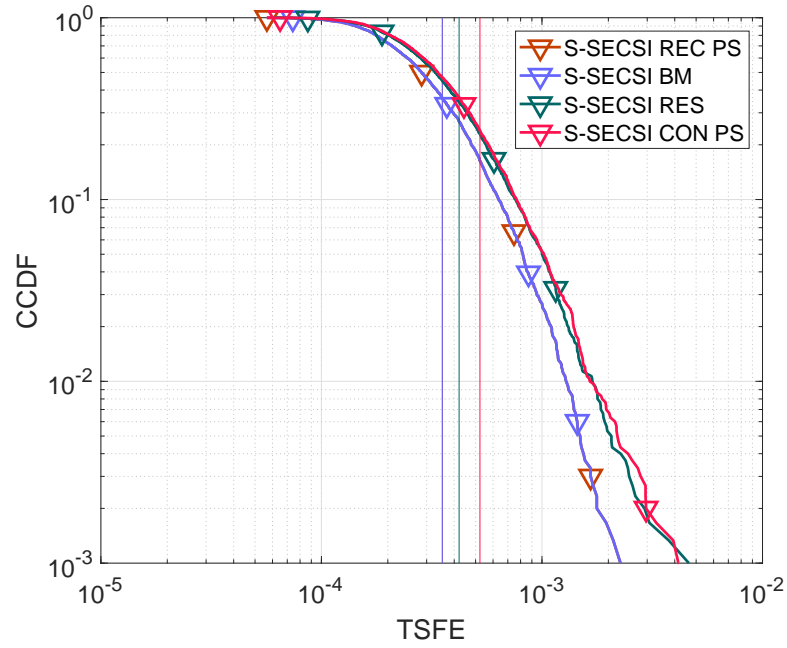


Figure 3.20.: CCDF of the TSFE for a complex-valued tensor with dimensions $8 \times 8 \times 8$, tensor rank $R = 3$, and SNR = 30 dB. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.

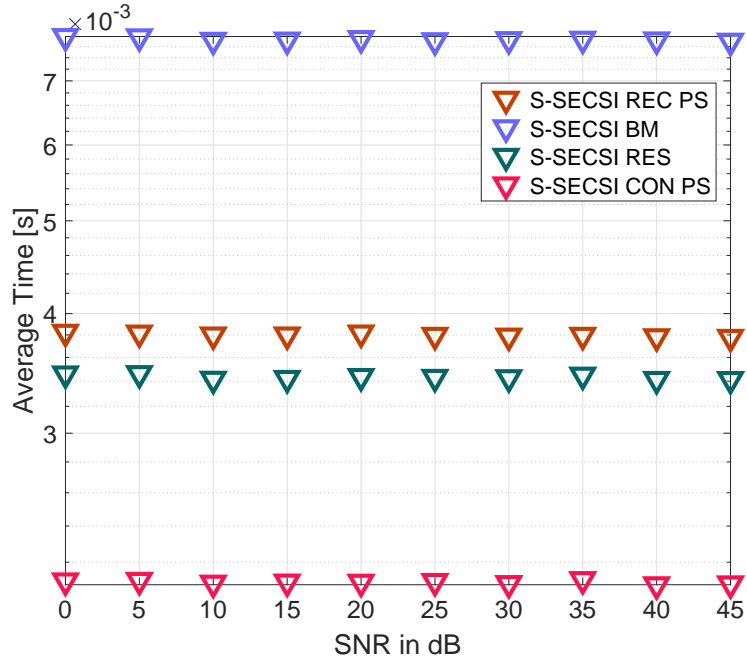


Figure 3.21.: Average required time in seconds for the computation of the CP decomposition of a complex-valued tensor with dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$. The symmetry is along the 1-mode and the 2-mode. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9.

Moreover, in Fig. 3.19 we consider a very difficult scenario. The synthetic real-valued tensor has dimensions $7 \times 7 \times 4$ and tensor rank $R = 3$. The third factor matrix is

$$\mathbf{F}_3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0.95 & 0.95 \\ 1 & 0.95 & 1 \\ 1 & 1 & 0.95 \end{bmatrix}.$$

Hence, the three components are highly correlated along the 3-mode. The CCDF of the TSFE is depicted in Fig. 3.19. All three algorithms use the REC PS heuristics and have a very similar performance. S-SECSI has a higher accuracy than NS-SECSI-NS-IDIEM even in this ill-posed scenario.

Furthermore, we study the performance of the different heuristics for S-SECSI. Therefore, we depict in Fig. 3.20 the CCDF of the TFSE for a complex-valued tensor and SNR = 30 dB. The tensor has dimensions $8 \times 8 \times 8$ and tensor rank $R = 3$. The third factor matrix \mathbf{F}_3 has mutually correlated columns with a correlation coefficient of 0.9. REC PS and BM have a similar performance in terms of the TFSE. The heuristic RES has lower accuracy than REC PS, but higher accuracy than CON PS. In Fig. 3.21, we illustrate the average time in seconds required to compute the CP decomposition for different SNRs. We observe a constant average required time for SNR between 0 dB and 45 dB. This is expected because S-SECSI diagonalizes a tensor of size $R \times R \times R$ in a closed-form fashion. As observed in [RH13a], where the different heuristics are proposed, we see that the CON PS requires the smallest amount of computational time followed by RES and REC PS. The time required to compute the CP decomposition using BM is always the longest. The computational time is consistent with the computational complexity of the different heuristics.

3.4.2 Summary

We propose a closed-form extension of the SECSI framework to the computation of the symmetric CP decomposition. The S-SECSI framework exploits the IDIEM algorithm and the NS-IDIEM algorithm for the computations of all possible initial estimates of the factor matrices. Note that for fully symmetric tensors only one set of estimates is computed. A final solution for symmetric tensors (not fully symmetric tensors) is chosen using the different heuristics BM, REC PS, RES, and CON PS. With the simulation results presented in this section we show the benefits of considering the symmetry constraints while computing the CP decomposition. In all considered scenarios, we observe that S-SECSI has a higher accuracy than NS-SECSI-NS-IDIEM in terms of the TSFE. NS-SECSI-NS-IDIEM also exploits the closed-form diagonalization algorithm NS-IDIEM. Therefore, NS-SECSI-NS-IDIEM and S-SECSI compute the CP decomposition in the same closed-form fashion. Only, NS-SECSI-NS-IDIEM ignores the symmetry constraints, whereas S-SECSI takes them into account.

3.5 Semi-Algebraic framework for the approximate CP decomposition via Simultaneous matrix diagonalization for Non-Negative Tensors (SECSI₊)

In some applications such as biomedical signal processing, image processing, and hyperspectral image processing the underlying components and the signal/data tensors are non-negative [CZPSI09, VCF⁺16] (see also Section 2.2.2). For these applications, we have to impose non-negativity constraints on the CP decomposition to ensure that the factor matrices have a reasonable physical interpretation.

The non-negative CP decomposition can be approximated using ADMM as proposed in [LS15]. The authors of [AALM16] exploit this concept for the computation of an SMD for positive semi-definite matrices. This algorithm is denoted by JEVD+ (Joint EigenValue Decomposition+), and it is also based on ADMM. In contrast to the ADMM algorithm proposed in [LS15], the JEVD+ algorithm imposes non-negativity constraints only on the resulting diagonal elements not on the transform matrices. Moreover, the authors of [AALM16] combine the JEVD+ algorithm with the DIAG algorithm [LA14] leading to the DIAG+ algorithm (DIrect AlGorithm for canonical polyadic decomposition+) that computes the CP decomposition of non-negative tensors. Furthermore, the authors of [CFC15] propose an ALS based algorithm for the computation of the CP decomposition for large non-negative tensors, where the fitting of the non-negative tensor is performed in a compressed domain. In this section, we propose an extension of the SECSI framework for the computation of an approximate CP decomposition for non-negative tensors. We denote this extension by SECSI₊.

The uniqueness properties of a non-negative CP decomposition are studied in [QCL16]. Some results regarding the rank of the CP decomposition with non-negativity constraints are provided in [CCG17].

Let us consider a non-negative tensor $\mathcal{X}_0 \in \mathbb{R}_+^{M_1 \times M_2 \times M_3}$ with rank R , where \mathbb{R}_+ denotes a set of non-negative real numbers. The tensor \mathcal{X}_0 has a non-negative CP decomposition given by

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_3, \quad (3.18)$$

where $\mathbf{F}_1 \in \mathbb{R}_+^{M_1 \times R}$, $\mathbf{F}_2 \in \mathbb{R}_+^{M_2 \times R}$, and $\mathbf{F}_3 \in \mathbb{R}_+^{M_3 \times R}$ are the factor matrices. Moreover, we assume that the tensor \mathcal{X}_0 has a NTD (Non-negative Tucker Decomposition) [CZPSI09]. The NTD is a Tucker decomposition with non-negativity constraints (for the definition of the Tucker decomposition see Section 2.2.1). For the NTD of the tensor \mathcal{X}_0 , we have

$$\mathcal{X}_0 = \mathcal{H} \times_1 \mathbf{G}_1 \times_2 \mathbf{G}_2 \times_3 \mathbf{G}_3,$$

where $\mathcal{H} \in \mathbb{R}_+^{R \times R \times R}$ is the core tensor and $\mathbf{G}_1 \in \mathbb{R}_+^{M_1 \times R}$, $\mathbf{G}_2 \in \mathbb{R}_+^{M_2 \times R}$, and $\mathbf{G}_3 \in \mathbb{R}_+^{M_3 \times R}$ are the loading matrices. Similar to the derivation of the SECSI framework [RH13a] described at the beginning of

this chapter, we assume that there exist invertible transform matrices, $\mathbf{T}_1 \in \mathbb{R}_+^{R \times R}$, $\mathbf{T}_2 \in \mathbb{R}_+^{R \times R}$, and $\mathbf{T}_3 \in \mathbb{R}_+^{R \times R}$ such that

$$\mathcal{X}_0 = \mathcal{I}_{3,R} \times_1 \underbrace{\mathbf{G}_1 \cdot \mathbf{T}_1}_{\mathbf{F}_1} \times_2 \underbrace{\mathbf{G}_2 \cdot \mathbf{T}_2}_{\mathbf{F}_2} \times_3 \underbrace{\mathbf{G}_3 \cdot \mathbf{T}_3}_{\mathbf{F}_3} = \mathcal{H} \times_1 \mathbf{G}_1 \times_2 \mathbf{G}_2 \times_3 \mathbf{G}_3. \quad (3.19)$$

Hence, for the tensor \mathcal{H} , we get $\mathcal{H} = \mathcal{I}_{3,R} \times_1 \mathbf{T}_1 \times_2 \mathbf{T}_2 \times_3 \mathbf{T}_3$. As in the original SECSI framework, after the multiplication of the tensor \mathcal{H} by \mathbf{G}_3 along the 3-mode, i.e., $\mathcal{H}_3 = \mathcal{H} \times_3 \mathbf{G}_3 \in \mathbb{R}_+^{R \times R \times M_3}$, we have

$$\mathcal{H}_3 = (\mathcal{I}_{3,R} \times_3 \mathbf{F}_3) \times_1 \mathbf{T}_1 \times_2 \mathbf{T}_2. \quad (3.20)$$

Equation (3.20) represents a non-symmetric SMD with positivity constraints on the transform matrices and the diagonalized tensor. The visualization of equation (3.20) is identical to the visualization of equation (3.4) depicted in Fig. 3.1. Note that equation (3.20) also represents a CP decomposition of a non-negative tensor that can be solved by the ADMM algorithm proposed in [LS15]. Alternatively, we propose to follow the derivation of the original SECSI framework and first to slice the tensor along the 3-mode, i.e.,

$$\mathcal{H}_{3(.,.,m_3)} = \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)}) \cdot \mathbf{T}_2^T \quad \forall m_3 = 1 \dots M_3.$$

Next, by eliminating one of the transform matrices from the right-hand side, we have

$$\begin{aligned} \mathcal{H}_{3(.,.,m_3)}^{\text{rhs}} &= \mathcal{H}_{3(.,.,m_3)} \cdot \mathcal{H}_{3(.,.,p)}^{-1} = \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)}) \cdot \mathbf{T}_2^T \cdot \mathbf{T}_2^{-T} \cdot \text{diag}(\mathbf{F}_{3(p,.)})^{-1} \cdot \mathbf{T}_1^{-1} \\ &= \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)} \oslash \mathbf{F}_{3(p,.)}) \cdot \mathbf{T}_1^{-1}. \end{aligned} \quad (3.21)$$

By eliminating one of the transform matrices from the left-hand side, we have

$$\begin{aligned} \mathcal{H}_{3(.,.,m_3)}^{\text{lhs}} &= \left(\mathcal{H}_{3(.,.,p)}^{-1} \cdot \mathcal{H}_{3(.,.,m_3)} \right)^T = \mathbf{T}_2 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)}) \cdot \mathbf{T}_1^T \cdot \mathbf{T}_1^{-T} \cdot \text{diag}(\mathbf{F}_{3(p,.)})^{-1} \cdot \mathbf{T}_2^{-1} \\ &= \mathbf{T}_2 \cdot \text{diag}(\mathbf{F}_{3(m_3,.)} \oslash \mathbf{F}_{3(p,.)}) \cdot \mathbf{T}_2^{-1}. \end{aligned} \quad (3.22)$$

For the selection of the pivoting slice $\mathcal{H}_{3(.,.,p)}$ we use the clever choice based on the conditioning number proposed in [RH13a], i.e., $p = \arg \min_{m_3=1, \dots, M_3} \text{cond}\{\mathcal{H}_{3(.,.,m_3)}\}$. The equations (3.21) and (3.22) represent two symmetric SMDs with non-negative constraints on the transform matrices and the diagonalized matrices. We propose to solve these constrained SMDs based on the ADMMD method similar to [LS15, AALM16]. We describe the proposed ADMMD₊ (Alternating Direction Method of Multipliers for non-negative simultaneous matrix Diagonalization) algorithm later in this section. Note that the authors of [AALM16] propose to solve similar SMDs as in equations (3.21) and (3.22). However, they do not impose non-negativity constraints on the transform matrices, but only on the resulting diagonal elements, i.e., the matrix \mathbf{F}_3 .

As explained at the beginning of this chapter and in [RH13a], by solving the constrained SMDs

in equations (3.21) and (3.22) we obtain the matrices, \mathbf{T}_1 , \mathbf{T}_2 , and \mathbf{F}_3 . Until now we have assumed a noiseless tensor \mathcal{X}_0 . However, in practice we have a noisy observation of the signal tensor. From this noisy observation, we can compute only an approximation of the factor matrices. Therefore, from equation (3.21) and (3.19), we obtain $\hat{\mathbf{F}}_1 = \hat{\mathbf{G}}_1 \cdot \hat{\mathbf{T}}_1$ using the transform matrix $\hat{\mathbf{T}}_1$ and $\hat{\mathbf{F}}_3$ from the resulting diagonal elements. Moreover, from equation (3.22) and (3.19), we obtain $\hat{\mathbf{F}}_2 = \hat{\mathbf{G}}_2 \cdot \hat{\mathbf{T}}_2$ using the transform matrix $\hat{\mathbf{T}}_2$ and $\hat{\mathbf{F}}_3$ from the resulting diagonal elements. The remaining factor matrices $\hat{\mathbf{F}}_1$ and $\hat{\mathbf{F}}_2$ can be then computed via a constrained LS fit [BdJ97]. Hence, we have obtained two sets of estimates of the factor matrices. Moreover, we can diagonalize the core tensor \mathcal{H} along the 1-mode and the 2-mode, after the multiplication by $\hat{\mathbf{G}}_1$ and $\hat{\mathbf{G}}_2$ along the corresponding mode exactly like it is proposed for the original SECSI framework [RH13a]. The diagonalization along the 1-mode and the 2-mode results into four additional sets of estimates of the factor matrices. Hence, we obtain six initial sets of estimates of the constrained factor matrices. These six estimates can be depicted via six parallel branches as for the SECSI framework in Fig. 3.2. The selection of the final solution can be done based on the different criteria BM, REC PS, RES, and CON PS defined at the beginning of this chapter and in [RH13a].

For the computation of the NTD (the first step of the SECSI₊ framework), we have chosen an algorithm that is based on block-coordinate decent [XY13] and available in the Tensor Toolbox from Sandia National Laboratories [BKS⁺12].

Alternating Direction Method of Multipliers for non-negative simultaneous matrix Diagonalization (ADMMD₊)

As previously mentioned, we propose the ADMMD₊ algorithm for the computation of a symmetric SMD with non-negative constraints given by

$$\mathcal{M} = \mathcal{I}_{3,R \times 1} \mathbf{A} \times_2 \mathbf{A}^{-T} \times_3 \mathbf{C} \Leftrightarrow \mathcal{M}_{(:, :, k)} = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k, :)}) \cdot \mathbf{A}^{-1} \quad \forall k = 1 \dots K,$$

where $\mathcal{M} \in \mathbb{R}_+^{R \times R \times K}$, $\mathbf{A} \in \mathbb{R}_+^{R \times R}$, and $\mathbf{C} \in \mathbb{R}_+^{K \times R}$. The goal of the ADMMD₊ algorithm is to diagonalize the 3-mode slices of the tensor \mathcal{M} (set of K matrices $\mathcal{M}_{(:, :, k)}$) jointly such that the transform matrix \mathbf{A} and the resulting diagonal elements that can be collected into a matrix \mathbf{C} do not contain negative elements. Therefore, we define the following cost function

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \Psi(\mathbf{A}, \mathbf{B}, \mathbf{C}) \quad \text{s.t.} \quad \mathbf{A} = \tilde{\mathbf{A}}, \mathbf{C} = \tilde{\mathbf{C}}, \\ \Psi(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \frac{1}{2} f(\mathbf{A}, \mathbf{B}, \mathbf{C}) + \frac{\alpha}{2} \|\mathbf{A}\mathbf{B}^T - \mathbf{I}_R\|_F^2 + g(\tilde{\mathbf{A}}) + g(\tilde{\mathbf{C}}) \end{aligned}$$

where $\mathbf{B} = \mathbf{A}^{-\text{T}}$ (recall that the superscript $^{-\text{T}}$ denotes matrix transposition and matrix inversion) and $\alpha \in [0, 1]$ is a penalty factor. Moreover,

$$f(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \left\| [\mathcal{M}]_{(1)} - \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^{\text{T}} \right\|_{\text{F}}^2 = \left\| [\mathcal{M}]_{(2)} - \mathbf{B}(\mathbf{C} \diamond \mathbf{A})^{\text{T}} \right\|_{\text{F}}^2 = \left\| [\mathcal{M}]_{(3)} - \mathbf{C}(\mathbf{B} \diamond \mathbf{A})^{\text{T}} \right\|_{\text{F}}^2 \quad (3.23)$$

and for a matrix $\mathbf{F} \in \mathbb{R}^{I \times J}$, we have

$$g(\tilde{\mathbf{F}}) = g(\tilde{\mathbf{F}}_{(i,j)}) = \begin{cases} 0, & \text{if } \tilde{\mathbf{F}}_{(i,j)} \geq 0 \\ \infty, & \text{otherwise} \end{cases}, \quad \forall i = 1, \dots, I, j = 1, \dots, J.$$

Next, for the augmented Lagrangian [LS15, AALM16], we get

$$\begin{aligned} \mathcal{L}_p(\mathbf{A}, \mathbf{B}, \mathbf{C}, \tilde{\mathbf{A}}, \tilde{\mathbf{C}}, \mathbf{\Lambda}_A, \mathbf{\Lambda}_C) &= \Psi(\mathbf{A}, \mathbf{B}, \mathbf{C}) + \|\mathbf{\Lambda}_A \odot (\mathbf{A} - \tilde{\mathbf{A}})\|_F^2 + \frac{\rho_A}{2} \|\mathbf{A} - \tilde{\mathbf{A}}\|_F^2 \\ &\quad + \|\mathbf{\Lambda}_C \odot (\mathbf{C} - \tilde{\mathbf{C}})\|_F^2 + \frac{\rho_C}{2} \|\mathbf{C} - \tilde{\mathbf{C}}\|_F^2, \end{aligned}$$

where the matrices $\mathbf{\Lambda}_A$ of size $R \times R$ and $\mathbf{\Lambda}_C$ of size $K \times R$ represent the Lagrangian multipliers. The values ρ_A and ρ_C are penalty factors. The ADMM is then given by [LS15]

$$\begin{aligned} \mathcal{L} &= \frac{1}{2} \arg \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} f(\mathbf{A}, \mathbf{B}, \mathbf{C}) + \frac{\alpha}{2} \|\mathbf{A}\mathbf{B}^{\text{T}} - \mathbf{I}_R\|_{\text{F}}^2 + \|\mathbf{\Lambda}_A \odot (\mathbf{A} - \tilde{\mathbf{A}})\|_F^2 + \frac{\rho_A}{2} \|\mathbf{A} - \tilde{\mathbf{A}}\|_F^2 \\ &\quad + \|\mathbf{\Lambda}_C \odot (\mathbf{C} - \tilde{\mathbf{C}})\|_F^2 + \frac{\rho_C}{2} \|\mathbf{C} - \tilde{\mathbf{C}}\|_F^2, \end{aligned} \quad (3.24)$$

The authors of [LS15] propose to solve the above Lagrangian using an alternating optimization scheme by substituting the different expressions given in equation (3.23). Hence, for the matrix \mathbf{A} , we get

$$\begin{aligned} \mathcal{L}_A &= \frac{1}{2} \arg \min_{\mathbf{A}} \left\| [\mathcal{M}]_{(1)} - \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^{\text{T}} \right\|_{\text{F}}^2 + \frac{\alpha}{2} \|\mathbf{A}\mathbf{B}^{\text{T}} - \mathbf{I}_R\|_{\text{F}}^2 + \|\mathbf{\Lambda}_A \odot (\mathbf{A} - \tilde{\mathbf{A}})\|_F^2 + \frac{\rho_A}{2} \|\mathbf{A} - \tilde{\mathbf{A}}\|_F^2 \\ &\quad + \|\mathbf{\Lambda}_C \odot (\mathbf{C} - \tilde{\mathbf{C}})\|_F^2 + \frac{\rho_C}{2} \|\mathbf{C} - \tilde{\mathbf{C}}\|_F^2, \end{aligned}$$

By computing the partial derivative with respect to \mathbf{A} and setting it to zero, we obtain the minimum of \mathcal{L}_A and with that the matrix \mathbf{A} .

$$\begin{aligned} -([\mathcal{M}]_{(1)} - \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^{\text{T}})(\mathbf{C} \diamond \mathbf{B}) + \alpha(\mathbf{A}\mathbf{B}^{\text{T}} - \mathbf{I}_R)\mathbf{B} + \mathbf{\Lambda}_A + \rho_A(\mathbf{A} - \tilde{\mathbf{A}}) &= 0 \\ -[\mathcal{M}]_{(1)}(\mathbf{C} \diamond \mathbf{B}) + \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^{\text{T}}(\mathbf{C} \diamond \mathbf{B}) + \alpha\mathbf{A}\mathbf{B}^{\text{T}}\mathbf{B} - \alpha\mathbf{B} + \mathbf{\Lambda}_A + \rho_A\mathbf{A} - \rho_A\tilde{\mathbf{A}} &= 0 \\ \mathbf{A}[(\mathbf{C} \diamond \mathbf{B})^{\text{T}}(\mathbf{C} \diamond \mathbf{B}) + \alpha\mathbf{B}^{\text{T}}\mathbf{B} + \rho_A] &= [\mathcal{M}]_{(1)}(\mathbf{C} \diamond \mathbf{B}) + \alpha\mathbf{B} - \mathbf{\Lambda}_A + \rho_A\tilde{\mathbf{A}} \\ \mathbf{A} &= [[\mathcal{M}]_{(1)}(\mathbf{C} \diamond \mathbf{B}) + \alpha\mathbf{B} - \mathbf{\Lambda}_A + \rho_A\tilde{\mathbf{A}}][(\mathbf{C} \diamond \mathbf{B})^{\text{T}}(\mathbf{C} \diamond \mathbf{B}) + \alpha\mathbf{B}^{\text{T}}\mathbf{B} + \rho_A\mathbf{I}_R]^{-1} \end{aligned} \quad (3.25)$$

Next, we substitute the second expression and the third expression from equation (3.23) into (3.24) and compute the partial derivatives with respect to \mathbf{B} and \mathbf{C} , respectively. After setting the partial

derivatives to zero, we obtain the matrices \mathbf{B} and \mathbf{C} .

$$\begin{aligned} & -[\mathcal{M}]_{(2)}(\mathbf{C} \diamond \mathbf{A}) + \mathbf{B}(\mathbf{C} \diamond \mathbf{A})^T(\mathbf{C} \diamond \mathbf{A}) + \alpha \mathbf{B} \mathbf{A}^T \mathbf{A} - \alpha \mathbf{A} = 0 \\ & \mathbf{B}[(\mathbf{C} \diamond \mathbf{A})^T(\mathbf{C} \diamond \mathbf{A}) + \alpha \mathbf{A}^T \mathbf{A}] = [\mathcal{M}]_{(2)}(\mathbf{C} \diamond \mathbf{A}) + \alpha \mathbf{A} \\ & \mathbf{B} = [[\mathcal{M}]_{(2)}(\mathbf{C} \diamond \mathbf{A}) + \alpha \mathbf{A}] [(\mathbf{C} \diamond \mathbf{A})^T(\mathbf{C} \diamond \mathbf{A}) + \alpha \mathbf{A}^T \mathbf{A}]^{-1} \end{aligned} \quad (3.26)$$

$$\begin{aligned} & -([\mathcal{M}]_{(3)} - \mathbf{C}(\mathbf{B} \diamond \mathbf{A})^T)(\mathbf{B} \diamond \mathbf{A}) + \mathbf{\Lambda}_C + \rho_C(\mathbf{C} - \tilde{\mathbf{C}}) = 0 \\ & -[\mathcal{M}]_{(3)}(\mathbf{B} \diamond \mathbf{A}) + \mathbf{C}(\mathbf{B} \diamond \mathbf{A})^T(\mathbf{B} \diamond \mathbf{A}) + \mathbf{\Lambda}_C + \rho_C \mathbf{C} - \rho_C \tilde{\mathbf{C}} = 0 \\ & \mathbf{C}[(\mathbf{B} \diamond \mathbf{A})^T(\mathbf{B} \diamond \mathbf{A}) + \rho_C \mathbf{I}_R] = ([\mathcal{M}]_{(3)}(\mathbf{B} \diamond \mathbf{A}) - \mathbf{\Lambda}_C + \rho_C \tilde{\mathbf{C}}) \\ & \mathbf{C} = [[\mathcal{M}]_{(3)}(\mathbf{B} \diamond \mathbf{A}) - \mathbf{\Lambda}_C + \rho_C \tilde{\mathbf{C}}] [(\mathbf{B} \diamond \mathbf{A})^T(\mathbf{B} \diamond \mathbf{A}) + \rho_C \mathbf{I}_R]^{-1} \end{aligned} \quad (3.27)$$

Moreover, from equation (3.24) we can compute the matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{C}}$ by computing the partial derivative of the argument with respect to $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{C}}$, respectively. By setting these partial derivatives to zero, we get

$$\tilde{\mathbf{A}} = \mathbf{A} + \frac{1}{\rho_A} \mathbf{\Lambda}_A \quad (3.28)$$

$$\tilde{\mathbf{C}} = \mathbf{C} + \frac{1}{\rho_C} \mathbf{\Lambda}_C \quad (3.29)$$

Finally, using the updates in equations (3.25)-(3.29), we can iterate in an ALS fashion until the algorithm converges. The convergence of this algorithm, its stopping criteria, and an optimal choice for the penalty factors are discussed in [LS15] (We use the results provided in [LS15, AALM16] in our implementation). Moreover, in each iteration the Lagrangian multipliers are updated according to $\mathbf{\Lambda}_A + \rho_A(\mathbf{A} - \tilde{\mathbf{A}})$ and $\mathbf{\Lambda}_C + \rho_C(\mathbf{C} - \tilde{\mathbf{C}})$. Note that the elements of the matrices \mathbf{A} and \mathbf{C} are not necessarily non-negative. They become non-negative upon convergence, or at least the negative values are very small. Note that the derivation shown here is very similar to [LS15, AALM16], however, we consider different constraints. The ADMM+ algorithm considers that all three factors differ and have non-negative elements [LS15]. On the other hand, the JEVD+ algorithm considers that $\mathbf{B} = \mathbf{A}^{-T}$, but only the factor \mathbf{C} has non-negative elements.

3.5.1 Simulation Results

In this section, we use simulation results to compare the proposed SECSI₊ framework with the ADMM+ algorithm [LS15]. Therefore, we generate synthetic tensors according to equation (3.18). The entries of the factor matrices are first drawn as i.i.d. zero mean Gaussian distributed random values. In a subsequent step, we take the absolute values of these entries, such that all elements of the factor matrices are non-negative. Moreover, we add i.i.d. zero mean Gaussian noise with variance σ_N^2 to the synthetic tensors. The resulting SNR of the noisy tensor is $\text{SNR} = 10 \log_{10}(\|\mathbf{x}_0\|_H^2 / \|\mathcal{N}\|_H^2)$ in dB, where the tensor \mathcal{N} represents the noise tensor.

We use the TSFE defined in equation (3.11) as an accuracy measure. We depict the CCDF of

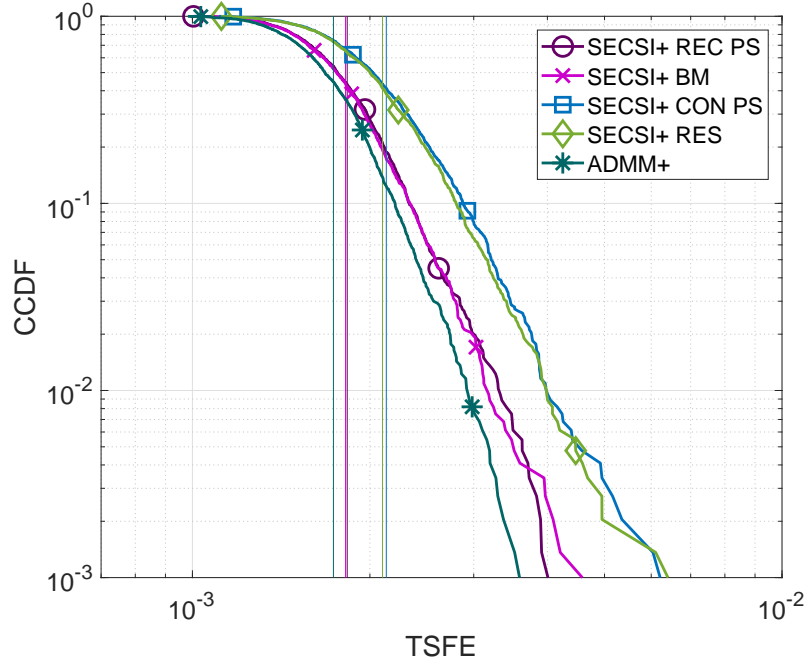


Figure 3.22.: CCDF of the TSFE for a real-valued tensor with dimensions $20 \times 20 \times 20$, tensor rank $R = 3$, and SNR = 10 dB.

the TSFE for real-valued tensor with dimensions $20 \times 20 \times 20$ and tensor rank $R = 3$ in Fig. 3.22. These results are representing 1500 realizations and correspond to an SNR = 10 dB. The vertical lines correspond to the mean values of the TSFE for each algorithm. In addition to the ADMM+ algorithm [LS15], in Fig. 3.22 we depict the TSFE of the SECSI+ framework with its different heuristics. The criteria for the selection of the final solution are indicated in the legend. Even though the SECSI+ framework (using REC PS, BM, or RES as a selection criterion) computes six initial estimates and chooses the best available solution, it has a lower accuracy than the ADMM+ algorithm. We have also analyzed the accuracy performance of the proposed ADMMD+ and the algorithm for the computation of the NTD [BKS⁺12] independently of the SECSI+ framework. The proposed ADMMD+ algorithm has a comparable accuracy to the JEVD+ algorithm [AALM16]. On the other hand, the algorithm for the computation of the NTD [BKS⁺12] has a limited accuracy even in noiseless cases. Hence, the compression of the tensor \mathcal{X} based on the NTD is not as effective as the compression based on the HOSVD in the SECSI framework. The authors of [CCG17] provide some similar results for the compression and noise suppression of non-negative tensors.

3.5.2 Summary

We propose an extension of the SECSI framework [RH13a] to the computation of the approximate CP decomposition of non-negative tensors denoted by SECSI+. The proposed SECSI+ framework utilizes the NTD for a tensor compression and noise suppression. In a subsequent step, we diagonalize the compressed core tensor based on constrained symmetric SMDs. For the computation of

these SMDs, we propose an ADMMD₊ algorithm based on ADMM [LS15]. SECSI₊ computes six initial estimates of the factor matrices for a 3-way tensor. In a final step using different heuristics, it selects the best available solution as a final solution. However, the SECSI₊ framework has no higher accuracy than other state-of-the-art algorithms. Therefore, to potentially increase the accuracy of SECSI₊, we propose the following ideas as a future work. First, we can consider other algorithms for the computation of the NTD in addition to the algorithm proposed in [BKS⁺12]. Moreover, we can compute a non-symmetric SMD from equation (3.20) using the ADMM₊ algorithm [LS15] resulting in three initial sets of estimates of the factor matrices (similar to the NS-SECSI framework from Section 3.3).

3.6 Semi-Algebraic framework for the approximate Coupled CP decomposition via Simultaneous matrix diagonalization (C-SECSI)

Several combined signal processing applications such as the joint processing of EEG and MEG data can benefit from coupled tensor decompositions, for instance, the coupled CP decomposition. The coupled CP decomposition jointly analyzes heterogeneous data sets or signals and identifies their shared underlying components. The facts that the heterogeneous signals can have a different nature make the coupled CP decomposition a very practical tool for signal analysis. This has already been shown in many applications such as array signal processing [SDL15, SDDL18, SDL17a, SDL17b], audio signal processing [ZCJW17], and biomedical signal processing [BCA12, ARS⁺13, PMS14, ABS15, RDGD⁺15, NKHH17, NLA⁺17, vEHDLvH17].

Assume two low rank 3-way tensors $\mathcal{X}_0^{(1)} \in \mathbb{C}^{M_1 \times M_2^{(1)} \times M_3^{(1)}}$ and $\mathcal{X}_0^{(2)} \in \mathbb{C}^{M_1 \times M_2^{(2)} \times M_3^{(2)}}$. Moreover, the two tensors have only one common mode and that is the 1-mode. These two tensors have a coupled CP decomposition defined as (see also Section 2.2.2)

$$\mathcal{X}_0^{(1)} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2^{(1)} \times_3 \mathbf{F}_3^{(1)} \quad (3.30)$$

$$\mathcal{X}_0^{(2)} = \mathcal{I}_{3,R} \times_1 \mathbf{F}_1 \times_2 \mathbf{F}_2^{(2)} \times_3 \mathbf{F}_3^{(2)}, \quad (3.31)$$

where, $\mathbf{F}_1 \in \mathbb{C}^{M_1 \times R}$, $\mathbf{F}_2^{(i)} \in \mathbb{C}^{M_2^{(i)} \times R}$ and $\mathbf{F}_3^{(i)} \in \mathbb{C}^{M_3^{(i)} \times R}$, $i = 1, 2$ are the factor matrices and R is the rank of both tensors. The coupled CP decomposition has even more relaxed uniqueness conditions as compared to the CP decomposition. Some uniqueness results for the coupled CP decomposition are available in [SDL15] and [ZCJW17]. The coupled CP decomposition is essentially unique under mild conditions, which means that the factor matrices (i.e., \mathbf{F}_1 , $\mathbf{F}_2^{(1)}$, $\mathbf{F}_2^{(2)}$, $\mathbf{F}_3^{(1)}$, and $\mathbf{F}_3^{(2)}$), can be identified up to a permutation and scaling ambiguity.

In order to compute the factors corresponding to the coupled CP decomposition, the existing algorithms for the computation of the CP decomposition have to be modified. For instance, the ALS algorithm can be simply extended to the C-ALS (Coupled-ALS) by taking into account that the common factor matrix can be computed jointly by means of concatenation. Similar to ALS,

the C-ALS also takes into account all unfoldings of the tensor and iteratively updates each of the factor matrices starting from a random initialization. The three unfoldings for the given 3-way tensors $\mathcal{X}_0^{(1)}$ and $\mathcal{X}_0^{(2)}$, that have the first factor matrix in common, are

$$[\mathcal{X}_0^{(i)}]_{(1)} = \mathbf{F}_1 \cdot (\mathbf{F}_3^{(i)} \diamond \mathbf{F}_2^{(i)})^T \quad [\mathcal{X}_0^{(i)}]_{(2)} = \mathbf{F}_2^{(i)} \cdot (\mathbf{F}_3^{(i)} \diamond \mathbf{F}_1)^T \quad [\mathcal{X}_0^{(i)}]_{(3)} = \mathbf{F}_3^{(i)} \cdot (\mathbf{F}_2 \diamond \mathbf{F}_1^{(i)})^T$$

where, $i = 1, 2$ denotes the two different tensors. From these unfoldings, the estimates of the factor matrices can be defined as follows. For the coupled mode (1-mode in our assumption), the u -th update of the corresponding factor matrix using the previous updates ($u - 1$) of the factor matrices is jointly calculated according to

$$\hat{\mathbf{F}}_{1,u} = \begin{bmatrix} [\mathcal{X}_0^{(1)}]_{(1)} & [\mathcal{X}_0^{(2)}]_{(1)} \end{bmatrix} \cdot \begin{bmatrix} \left(\hat{\mathbf{F}}_{3,u-1}^{(1)} \diamond \hat{\mathbf{F}}_{2,u-1}^{(1)} \right)^T & \left(\mathbf{F}_{3,u-1}^{(2)} \diamond \hat{\mathbf{F}}_{2,u-1}^{(2)} \right)^T \end{bmatrix}^+.$$

Note that the above equation could lead to an improved identifiability of the coupled CP decomposition. Hence, the coupled CP decomposition can have relaxed uniqueness conditions as compared to the CP decomposition. As previously mentioned, some uniqueness results for the coupled CP decomposition are available in [SDL15] and [ZCJW17]. Moreover, the u -th update for the other two factor matrices is given by

$$\begin{aligned} \hat{\mathbf{F}}_{2,u}^{(i)} &= [\mathcal{X}_0^{(i)}]_{(2)} \cdot \left(\left(\hat{\mathbf{F}}_{3,u-1}^{(i)} \diamond \hat{\mathbf{F}}_{1,u-1} \right)^T \right)^+ \\ \hat{\mathbf{F}}_{3,u}^{(i)} &= [\mathcal{X}_0^{(i)}]_{(3)} \cdot \left(\left(\hat{\mathbf{F}}_{2,u-1}^{(i)} \diamond \hat{\mathbf{F}}_{1,u-1} \right)^T \right)^+. \end{aligned}$$

The C-ALS algorithm is an iterative algorithm with no guarantee of convergence that requires stopping criteria. The stopping criteria are defined as follows. If the difference between the factor matrices of the previous and current update is smaller than a predefined error then the current factor matrices are the final estimate, or simply if the number of iterations exceeds a predefined maximum number of iterations the algorithm will stop. The C-ALS algorithm computes the factor matrices such that the coupled or common mode will always have the same estimate for the two tensors.

Another weighted version of coupled ALS using normalization is proposed in [FCC16] that can even support a hybrid and a noisy coupling. For the purpose of dimensionality reduction, a compression based on the truncated HOSVD can be incorporated with ALS [CFC16]. The ALS based algorithms are easy to implement, however, they have no convergence guarantee and can require many iterations. Alternatively, the coupled CP decomposition can be computed based on a line search. A line search based algorithm referred to as CCP-MINF (Coupled CP-Minimum Factors) is available in Tensorlab [VDS⁺16]. An NLS based algorithm for the computation of the coupled CP denoted by CCP-NLS (Coupled CP-Nonlinear Least-Squares) is also available in Tensorlab [VDS⁺16]. The CCP-NLS algorithm is an iterative algorithm that computes the update of the factor matrices based on Newton descent that includes linear approximation of the Hessian.

Furthermore, similar to the CP decomposition the coupled CP decomposition can be solved using semi-algebraic algorithms. A semi-algebraic computation of the coupled CP decomposition is proposed in [SDDL15]. However, the algorithm proposed in [SDDL15] considers only one of the full set of possible SMDs. The SECSI framework is an efficient tool for the calculation of the CP decomposition based on matrix diagonalizations (see Section 3.1) [RH13a]. Moreover, SECSI provides a semi-algebraic solution for the approximate CP decomposition even in ill-posed scenarios, e.g., if the columns of a factor matrix are highly correlated. Furthermore, the SECSI framework provides an adjustable complexity-accuracy trade-off. In this section, we present an extension of the SECSI framework to the efficient computation of an approximate coupled CP decomposition and show its advantages compared to other state-of-the-art algorithms [NH16]. This extension of the SECSI framework we denote by C-SECSI (Coupled SECSI). The C-SECSI framework proposed here uses the tensor structure to construct not only one but the full set of possible SMDs jointly for both tensors [NH16].

Another, tensor decomposition which is much easier to calculate is the HOSVD [DLDMV00b]. The HOSVD of the tensors $\mathcal{X}_0^{(1)}$ and $\mathcal{X}_0^{(2)}$ is given by (see also Section 2.2.1)

$$\begin{aligned}\mathcal{X}_0^{(1)} &= \mathcal{S}^{(1)} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2^{(1)} \times_3 \mathbf{U}_3^{(1)} \\ \mathcal{X}_0^{(2)} &= \mathcal{S}^{(2)} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2^{(2)} \times_3 \mathbf{U}_3^{(2)},\end{aligned}$$

where $\mathcal{S}^{(1)} \in \mathbb{C}^{M_1 \times M_2^{(1)} \times M_3^{(1)}}$ and $\mathcal{S}^{(2)} \in \mathbb{C}^{M_1 \times M_2^{(2)} \times M_3^{(2)}}$ are the core tensors. The matrices $\mathbf{U}_1 \in \mathbb{C}^{M_1 \times M_1}$, $\mathbf{U}_2^{(i)} \in \mathbb{C}^{M_2^{(i)} \times M_2^{(i)}}$, and $\mathbf{U}_3^{(i)} \in \mathbb{C}^{M_3^{(i)} \times M_3^{(i)}}$ ($i = 1, 2$), are unitary matrices.

Moreover, the coupled truncated HOSVD is defined as (see also Section 2.2.1)

$$\mathcal{X}_0^{(1)} = \mathcal{S}^{[s],(1)} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s],(1)} \times_3 \mathbf{U}_3^{[s],(1)} \quad (3.32)$$

$$\mathcal{X}_0^{(2)} = \mathcal{S}^{[s],(2)} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s],(2)} \times_3 \mathbf{U}_3^{[s],(2)}, \quad (3.33)$$

where $\mathcal{S}^{[s],(1)} \in \mathbb{C}^{R \times R \times R}$ and $\mathcal{S}^{[s],(2)} \in \mathbb{C}^{R \times R \times R}$ are the truncated core tensors and the loading matrices $\mathbf{U}_1^{[s]} \in \mathbb{C}^{M_1 \times R}$, $\mathbf{U}_2^{[s],(i)} \in \mathbb{C}^{M_2^{(i)} \times R}$, and $\mathbf{U}_3^{[s],(i)} \in \mathbb{C}^{M_3^{(i)} \times R}$ have unitary columns and span the column space of the n -mode unfolding of $\mathcal{X}_0^{(i)}$, for $n = 1, 2, 3$ and $i = 1, 2$, respectively. Note that the matrices $\mathbf{U}_1^{[s]}$ and \mathbf{F}_1 in (3.30) span the same column space of $[\mathcal{X}_0^{(1)}]_{(1)}$. Due to the fact that the tensors $\mathcal{X}_0^{(1)}$ and $\mathcal{X}_0^{(2)}$ have the factor matrix \mathbf{F}_1 in common the unitary matrix $\mathbf{U}_1^{[s]}$ spans the column space of $[\mathcal{X}_0^{(2)}]_{(1)}$ as well.

In practice, we can only observe a noise corrupted version of the low rank tensors, i.e., $\mathcal{X}^{(i)} = \mathcal{X}_0^{(i)} + \mathcal{N}^{(i)}$, where $\mathcal{N}^{(i)}$ contains uncorrelated zero mean circularly symmetric complex Gaussian noise. Hence, we have to calculate a rank R approximation of $\mathcal{X}^{(i)}$

$$\mathcal{X}^{(i)} \approx \mathcal{S}^{[s],(i)} \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s],(i)} \times_3 \mathbf{U}_3^{[s],(i)}. \quad (3.34)$$

Note that (3.34) holds exactly in the absence of noise and if R is the true rank of the tensor $\mathcal{X}^{(i)}$.

For the following derivations, we assume that this is true and hence write equalities. In the presence of noise, all following relations still hold approximately.

Next, we derive the C-SECSI framework for two tensors of order three denoted by $\mathcal{X}^{(i)}$, $i = 1, 2$, which have the first factor matrix in common. An extension to tensors of order N is possible. Moreover, an extension to multiple common modes is straightforward. Our goal is to jointly provide an estimate of the factor matrices for both tensors. The C-SECSI framework starts by computing the truncated HOSVD. Since the first factor matrix is common for both tensors, the column space of the corresponding 1-mode unfolding is calculated jointly, and independently for the rest of the modes (i.e., $n = 2, 3$) via the following SVDs (see also Section 2.2.1)

$$\begin{aligned} \begin{bmatrix} [\mathcal{X}^{(1)}]_{(1)} & [\mathcal{X}^{(2)}]_{(1)} \end{bmatrix} &= \mathbf{U}_1^{[s]} \cdot \boldsymbol{\Sigma}_1^{[s]} \cdot \mathbf{V}_1^{[s]H}, \\ [\mathcal{X}^{(i)}]_{(n)} &= \mathbf{U}_n^{[s],(i)} \cdot \boldsymbol{\Sigma}_n^{[s],(i)} \cdot \mathbf{V}_n^{[s],(i)H}, \quad n = 2, 3, \quad i = 1, 2. \end{aligned}$$

Inserting equations (3.32) and (3.33) into (3.30) and (3.31), we get

$$\mathcal{X}^{(i)} = \left(\mathcal{S}^{[s],(i)} \times_3 \mathbf{U}_3^{[s],(i)} \right) \times_1 \mathbf{U}_1^{[s]} \times_2 \mathbf{U}_2^{[s],(i)} \quad (3.35)$$

$$= \left(\mathcal{I}_{3,R} \times_3 \underbrace{(\mathbf{U}_3^{[s],(i)} \cdot \mathbf{T}_3^{(i)})}_{\mathbf{F}_3^{(i)}} \right) \times_1 \underbrace{(\mathbf{U}_1^{[s]} \cdot \mathbf{T}_1)}_{\mathbf{F}_1} \times_2 \underbrace{(\mathbf{U}_2^{[s],(i)} \cdot \mathbf{T}_2^{(i)})}_{\mathbf{F}_2^{(i)}} \quad (3.36)$$

The equations (3.35) and (3.36) represent the fundamental link between the HOSVD and the CP decomposition, and the coupling between the two tensors. The invertible matrices \mathbf{T}_1 , $\mathbf{T}_2^{(i)}$, and $\mathbf{T}_3^{(i)}$ of size $R \times R$ diagonalize the core tensors $\mathcal{S}^{[s],(i)}$, for $i = 1, 2$, respectively, as previously shown at the beginning of this chapter and in [RH08, RH13a]. Therefore, after multiplying equations (3.35) and (3.36) by $\times_1 \mathbf{U}_1^{[s]H} \times_2 \mathbf{U}_2^{[s],(i)H}$, we obtain the following tensors

$$\mathcal{S}_3^{(i)} = \mathcal{F}_3^{(i)} \times_1 \mathbf{T}_1 \times_2 \mathbf{T}_2^{(i)} \quad i = 1, 2, \quad (3.37)$$

where $\mathcal{S}_3^{(i)} = \mathcal{S}^{[s],(i)} \times_3 \mathbf{U}_3^{[s],(i)} \in \mathbb{C}^{R \times R \times M_3^{(i)}}$ and $\mathcal{F}_3^{(i)} = \mathcal{I}_{3,R} \times_3 \mathbf{F}_3^{(i)} \in \mathbb{C}^{R \times R \times M_3^{(i)}}$. Equation (3.37) is visualized in Fig. 3.23.

Equation (3.37) represents a non-symmetric SMD. Here, we recommend to diagonalize the core tensors via symmetric SMDs, for instance, using the JD algorithm proposed in [FG06]. The extension of the SECSI framework based on non-symmetric SMDs is presented in Section 3.3, [NHT⁺16]. However, instead of non-symmetric SMDs we recommend using symmetric SMDs so that the coupling between the two tensors can be better exploited. Therefore, we convert the non-symmetric SMD problem in equation (3.37) into two symmetric SMDs. In order to do so one of the transform matrices has to be eliminated. Hence, as shown in [RH08] and in Section 3.1 we multiply equation

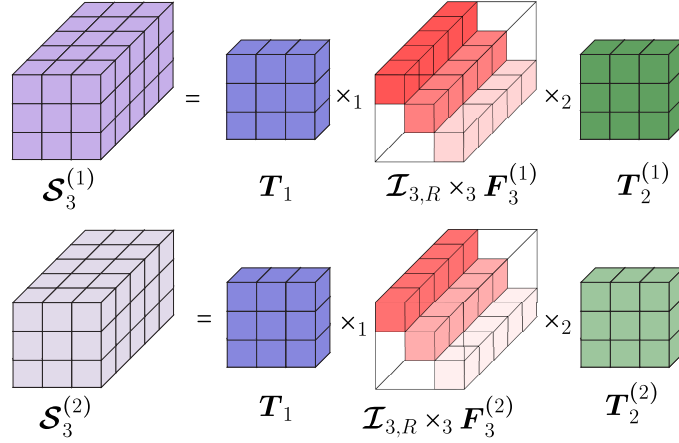


Figure 3.23.: Diagonalization of the tensors $\mathcal{S}_3^{(1)} \in \mathbb{C}^{R \times R \times M_3^{(1)}}$ and $\mathcal{S}_3^{(2)} \in \mathbb{C}^{R \times R \times M_3^{(2)}}$ along the 3-mode.

(3.37) by one pivoting slice from the right-hand and left-hand side, respectively.

$$\mathcal{S}_{3(\dots, k_i)}^{\text{rhs}, (i)} = \mathcal{S}_{3(\dots, k_i)}^{(i)} \cdot \mathcal{S}_{3(\dots, p_i)}^{(i)-1} = \mathbf{T}_1 \cdot \text{diag}(\mathbf{F}_{3(k_i, \cdot)}^{(i)} \oslash \mathbf{F}_{3(p_i, \cdot)}^{(i)}) \cdot \mathbf{T}_1^{-1} \quad (3.38)$$

$$\mathcal{S}_{3(\dots, k_i)}^{\text{lhs}, (i)} = \left(\mathcal{S}_{3(\dots, p_i)}^{(i)-1} \mathcal{S}_{3(\dots, k_i)}^{(i)} \right)^T = \mathbf{T}_2^{(i)} \cdot \text{diag}(\mathbf{F}_{3(k_i, \cdot)}^{(i)} \oslash \mathbf{F}_{3(p_i, \cdot)}^{(i)}) \cdot \mathbf{T}_2^{(i)-1} \quad (3.39)$$

where $\mathcal{S}_{3(\dots, k_i)}^{(i)}$ is the k_i -th slice of the tensor $\mathcal{S}_3^{(i)}$ and $k_i = 1, \dots, M_3^{(i)}$. Moreover, $\mathbf{F}_{3(k_i, \cdot)}^{(i)}$ represents the k_i -th row of the factor matrix $\mathbf{F}_3^{(i)}$. Furthermore, p_i can be any arbitrary pivoting slice, $p_i \in \{1, 2, \dots, M_3^{(i)}\}$. However, since this slice has to be inverted, the best choice is to choose the slice with the smallest conditioning number [RH13a]. Note that a different pivoting slice p_i can be chosen for the different tensors. Hence, equation (3.38) represents two symmetric SMDs, for each of the two tensors $\mathcal{S}_3^{(1)}$ and $\mathcal{S}_3^{(2)}$. Moreover, the two SMDs have the same transform matrices, which means that we can concatenate the two equations and solve one diagonalization problem instead. Hence,

$$\begin{Bmatrix} \mathcal{S}_{3(\dots, k_1)}^{\text{rhs}, (1)} \\ \mathcal{S}_{3(\dots, k_2)}^{\text{rhs}, (2)} \end{Bmatrix} = \mathbf{T}_1 \cdot \begin{Bmatrix} \text{diag}(\mathbf{F}_{3(k_1, \cdot)}^{(1)} \oslash \mathbf{F}_{3(p_1, \cdot)}^{(1)}) \\ \text{diag}(\mathbf{F}_{3(k_2, \cdot)}^{(2)} \oslash \mathbf{F}_{3(p_2, \cdot)}^{(2)}) \end{Bmatrix} \cdot \mathbf{T}_1^{-1} \quad (3.40)$$

is a coupled symmetric SMD, which allows as to diagonalize both core tensors jointly. From this coupled SMD, we can estimate the first factor matrix as $\hat{\mathbf{F}}_{1, \text{I}} = \mathbf{U}_1^{[\text{s}]} \cdot \mathbf{T}_1$ guaranteeing that even in a noisy scenario the common mode will have the same factor matrix estimate for the tensors $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$. Next, from the diagonal elements of the diagonalized tensor the factor matrices $\hat{\mathbf{F}}_{3, \text{I}}^{(1)}$ and $\hat{\mathbf{F}}_{3, \text{I}}^{(2)}$ are estimated [RH13a]. Finally, using the corresponding estimates of the other two factor matrices, the last factor matrices are estimated based on an LS solution $\hat{\mathbf{F}}_{2, \text{I}}^{(1)}$ and $\hat{\mathbf{F}}_{2, \text{I}}^{(2)}$.

Note that equation (3.39) does not depend on the common mode. Therefore, the two SMDs cannot be combined and they have to be solved separately. Similarly, to the coupled SMD, an estimate of the matrices $\hat{\mathbf{F}}_{2, \text{II}}^{(1)}$, $\hat{\mathbf{F}}_{2, \text{II}}^{(2)}$, $\hat{\mathbf{F}}_{3, \text{II}}^{(1)}$ and $\hat{\mathbf{F}}_{3, \text{II}}^{(2)}$ is provided from the transform matrix $\mathbf{T}_2^{(i)}$ and the resulting diagonal elements, respectively. The common factor matrix is then estimated

from the following joint LS problem

$$\hat{\mathbf{F}}_{1,\text{II}} = \begin{bmatrix} [\mathcal{X}^{(1)}]_{(1)} & [\mathcal{X}^{(2)}]_{(1)} \end{bmatrix} \cdot \begin{bmatrix} \left(\hat{\mathbf{F}}_{2,\text{II}}^{(1)} \diamond \hat{\mathbf{F}}_{3,\text{II}}^{(1)} \right)^T & \left(\hat{\mathbf{F}}_{2,\text{II}}^{(2)} \diamond \hat{\mathbf{F}}_{3,\text{II}}^{(2)} \right)^T \end{bmatrix}^+ \quad (3.41)$$

Up to this point, we have diagonalized the tensors along the 3-mode as depicted in Fig. 3.23, but the rest of the modes can also be used in order to obtain more estimates as explained in [RH13a]. Another two sets of estimates can be obtained by diagonalizing the tensors along the 2-mode based on the following SMDs

$$\begin{Bmatrix} \mathcal{S}_{2,(\cdot,k_1,\cdot)}^{\text{rhs},(1)} \\ \mathcal{S}_{2,(\cdot,k_2,\cdot)}^{\text{rhs},(2)} \end{Bmatrix} = \mathbf{T}_1 \cdot \begin{Bmatrix} \text{diag}(\mathbf{F}_{2(k_1,\cdot)}^{(1)} \oslash \mathbf{F}_{2(p_1,\cdot)}^{(1)}) \\ \text{diag}(\mathbf{F}_{2(k_2,\cdot)}^{(2)} \oslash \mathbf{F}_{2(p_2,\cdot)}^{(2)}) \end{Bmatrix} \cdot \mathbf{T}_1^{-1} \quad (3.42)$$

and

$$\mathcal{S}_{2,(\cdot,k_i,\cdot)}^{\text{lhs},(i)} = \mathbf{T}_3^{(i)} \cdot \text{diag}(\mathbf{F}_{2(k_i,\cdot)}^{(i)} \oslash \mathbf{F}_{2(p_i,\cdot)}^{(i)}) \cdot \mathbf{T}_3^{(i)-1}. \quad (3.43)$$

The estimates obtained from (3.42) are given by $\hat{\mathbf{F}}_{1,\text{III}} = \mathbf{U}_1^{[\text{s}]} \cdot \mathbf{T}_1$ from the transform matrix, $\hat{\mathbf{F}}_{2,\text{III}}^{(1)}$, $\hat{\mathbf{F}}_{2,\text{III}}^{(2)}$ from the diagonal elements of the diagonalized tensor, and $\hat{\mathbf{F}}_{3,\text{III}}^{(1)}$ and $\hat{\mathbf{F}}_{3,\text{III}}^{(2)}$ based on an LS solution using the corresponding estimates of the other two factor matrices. Moreover, from (3.43) the following estimates are obtained. The factor matrices $\hat{\mathbf{F}}_{3,\text{IV}}^{(1)} = \mathbf{U}_3^{[\text{s}],(i)} \cdot \mathbf{T}_3^{(i)}$ and $\hat{\mathbf{F}}_{3,\text{IV}}^{(2)} = \mathbf{U}_3^{[\text{s}],(i)} \cdot \mathbf{T}_3^{(i)}$ are obtained from the transform matrices. Furthermore, $\hat{\mathbf{F}}_{2,\text{IV}}^{(1)}$ and $\hat{\mathbf{F}}_{2,\text{IV}}^{(2)}$ are obtained from the diagonal elements of the diagonalized tensor and $\hat{\mathbf{F}}_{1,\text{IV}}$ is estimated based on the following joint LS problem.

$$\hat{\mathbf{F}}_{1,\text{IV}} = \begin{bmatrix} [\mathcal{X}^{(1)}]_{(1)} & [\mathcal{X}^{(2)}]_{(1)} \end{bmatrix} \cdot \begin{bmatrix} \left(\hat{\mathbf{F}}_{2,\text{IV}}^{(1)} \diamond \hat{\mathbf{F}}_{3,\text{IV}}^{(1)} \right)^T & \left(\hat{\mathbf{F}}_{2,\text{IV}}^{(2)} \diamond \hat{\mathbf{F}}_{3,\text{IV}}^{(2)} \right)^T \end{bmatrix}^+ \quad (3.44)$$

Finally, the following SMDs are defined for the tensor's diagonalization along the first mode.

$$\begin{aligned} \mathcal{S}_{1,(k_i,\cdot,\cdot)}^{\text{rhs},(i)} &= \mathbf{T}_2^{(i)} \cdot \text{diag}(\mathbf{F}_{1(k_i,\cdot)}^{(i)} \oslash \mathbf{F}_{1(p_i,\cdot)}^{(i)}) \cdot \mathbf{T}_2^{(i)-1} \\ \mathcal{S}_{1,(k_i,\cdot,\cdot)}^{\text{lhs},(i)} &= \mathbf{T}_3^{(i)} \cdot \text{diag}(\mathbf{F}_{1(k_i,\cdot)}^{(i)} \oslash \mathbf{F}_{1(p_i,\cdot)}^{(i)}) \cdot \mathbf{T}_3^{(i)-1} \end{aligned}$$

The coupled mode is in the diagonal elements of the diagonalized tensor, therefore a joint SMD cannot be calculated. From the four SMDs presented above (recall that $i = 1, 2$), four different estimates of the coupled mode are obtained. The estimates obtained from the diagonalization along the first mode and their origin are summarized in Table 3.1.

Transform Matrix	$\hat{\mathbf{F}}_{2,\text{V}}^{(i)}$	$\hat{\mathbf{F}}_{2,\text{VI}}^{(i)} = \hat{\mathbf{F}}_{2,\text{V}}^{(i)}$	$\hat{\mathbf{F}}_{3,\text{VII}}^{(i)}$	$\hat{\mathbf{F}}_{3,\text{VIII}}^{(i)} = \hat{\mathbf{F}}_{3,\text{VII}}^{(i)}$
Diagonalized Tensor	$\hat{\mathbf{F}}_{1,\text{V}}^{(i)}$	$\hat{\mathbf{F}}_{1,\text{VI}}^{(i)}$	$\hat{\mathbf{F}}_{1,\text{VII}}^{(i)}$	$\hat{\mathbf{F}}_{1,\text{VIII}}^{(i)}$
LS	$\hat{\mathbf{F}}_{3,\text{V}}^{(i)}$	$\hat{\mathbf{F}}_{3,\text{VI}}^{(i)}$	$\hat{\mathbf{F}}_{2,\text{VII}}^{(i)}$	$\hat{\mathbf{F}}_{2,\text{VIII}}^{(i)}$

Table 3.1.: Estimates of the factor matrices obtained from the diagonalization along the first mode.

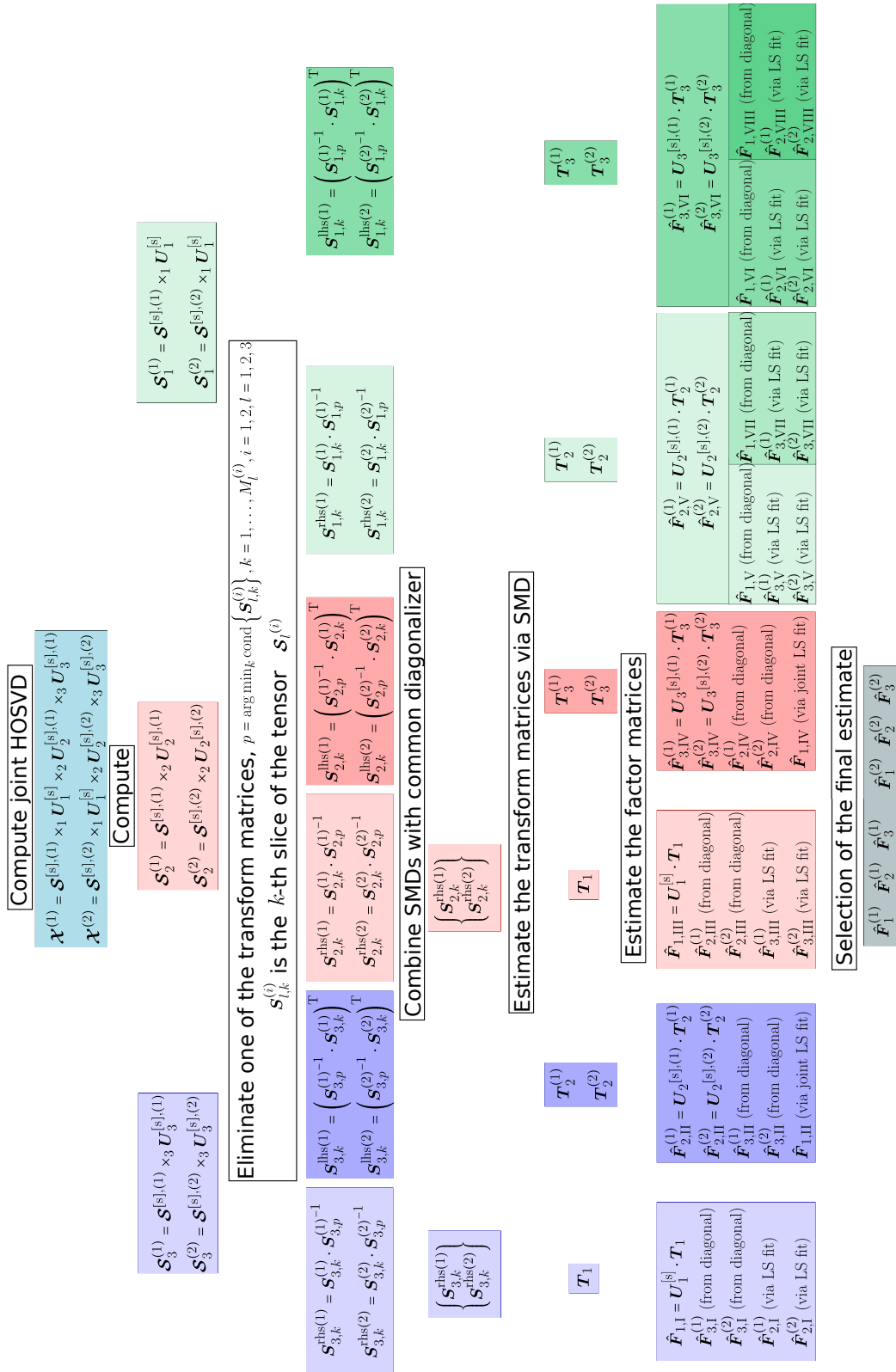


Figure 3.24.: The C-SECSI framework for the computation of the coupled CP decomposition of two tensors $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$ that have the 1-mode in common.

To summarize, the C-SECSI framework for two tensors of order three with N_c common modes, i.e., $N_c = 1, 2$, will result in $6 + 2N_c$ sets of estimates of the factor matrices. For the scenario that we have presented in this section, two tensors of order three with one mode in common, eight different initial sets of estimates are obtained with the C-SECSI framework. These eight different sets of estimates are depicted in Fig. 3.24. As a comparison, the original SECSI framework calculates six sets of estimates, [RH13a]. The two additional sets are obtained from the diagonalization along the coupled mode. The estimate of the common mode that results from the tensor $\mathcal{X}^{(1)}$ can be considered as a possible solution for the tensor $\mathcal{X}^{(2)}$ as well (see Table 3.1). However, when using the common factor matrix that is estimated from another tensor and for calculating the joint LS according to equations (3.41) and (3.44), the permutation and scaling ambiguity has to be taken into account. The estimates that are obtained from different SMDs have an arbitrary permutation, which can be eliminated via a comparison if one estimate is taken as a reference.

From the eight initial estimates of the factor matrices (see Fig. 3.24) the first four estimates of the common factor matrix (from $\hat{\mathbf{F}}_{1,I}$ to $\hat{\mathbf{F}}_{1,IV}$) are obtained either from the common transform matrices or via a joint LS fit. On the other hand, the last four estimates (from $\hat{\mathbf{F}}_{1,V}$ to $\hat{\mathbf{F}}_{1,VIII}$) are separately obtained from the diagonal elements of the diagonalized tensor. Therefore, the first four solutions are coupled, and the last four solutions are uncoupled. The final solution is then chosen for each of the tensors separately based on the chosen heuristics [NH16, RH13a]. The different heuristics BM, REC PS, and RES offer accuracy-complexity trade-off (see Section 3.1). Note that the heuristic CON if applied for C-SECSI offers no guarantee that a coupled solution will be computed. Therefore, the heuristic CON is not suitable for C-SECSI. In general, we recommend using the heuristic REC PS with which the final solution (one of the eight possible solutions I–VIII) is chosen based on the reconstruction error [RH13a], where the reconstruction error is calculated according to (3.8). Note that when $\mathcal{X}^{(i)}$ is a noisy observation, i.e. $\mathcal{X}^{(i)} = \mathcal{X}_0^{(i)} + \mathcal{N}^{(i)}$ we also refer to this error as a residual and denote it by RES.

Moreover, in [NKHH17] we propose a reliability measure for the C-SECSI framework that checks whether the same (coupled) solution is chosen for both tensors. Therefore, a reliability in percent-age,

$$\text{REL} = \left(1 - \frac{1}{2} \cdot \frac{\left\| \hat{\mathbf{F}}_1^{(2)} \cdot \mathbf{P} - \hat{\mathbf{F}}_1^{(1)} \right\|_F^2}{\left\| \hat{\mathbf{F}}_1^{(1)} \right\|_F^2} \right) \cdot 100\%, \quad (3.45)$$

is defined as a similarity measure of the final estimates of the common factor matrices. Here, \mathbf{P} is a permutation matrix of size $R \times R$ that resolves the permutation ambiguity of the CP decomposition. The matrices $\hat{\mathbf{F}}_1^{(1)}$ and $\hat{\mathbf{F}}_1^{(2)}$ are the final estimates of the common mode assigned to the tensors $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$, respectively. This reliability measure has a maximum if the final estimates result from a coupled solution and the assumed rank is correctly chosen. Therefore, the reliability can be used to control the tensor rank of the coupled approximate CP decomposition. Note that for tensor rank one the reliability is always 100%. This is due to the fact that for rank one tensors the

C-SECSI framework does not calculate any SMD. In this case, only one final estimate of the factor matrices is provided directly from the coupled truncated HOSVD.

3.6.1 Simulation Results

In this section, the proposed extension of SECSI for coupled CP decompositions, denoted by C-SECSI, is compared to the original SECSI framework [RH13a] and other state-of-the-art algorithms for the computation of coupled CP. These algorithms include C-ALS, CCP-NLS [VDS⁺16], and CCP-MINF [VDS⁺16]. For simulation purposes, we generate two different tensors with tensor rank R and first factor matrix in common according to equations (3.30) and (3.31), where the factor matrices \mathbf{F}_1 , $\mathbf{F}_2^{(i)}$, and $\mathbf{F}_3^{(i)}$ have i.i.d. zero mean Gaussian distributed random entries (or ZMCSCG distributed random entries for complex-valued tensors) with variance one. Moreover, for some simulation scenarios, we want the tensors to have correlated factor matrices. Therefore, we add correlation via a correlation matrix as shown in (3.10). Additionally, a zero-mean Gaussian distributed (or ZMCSCG for complex-valued tensors) noise with variance σ_N^2 is added resulting in SNR_1 and SNR_2 .

$$\text{SNR}_1 = 10 \log_{10} \left(\frac{\|\mathbf{x}_0^{(1)}\|_H^2}{\|\mathcal{N}^{(1)}\|_H^2} \right) [\text{dB}] \quad \text{SNR}_2 = 10 \log_{10} \left(\frac{\|\mathbf{x}_0^{(2)}\|_H^2}{\|\mathcal{N}^{(2)}\|_H^2} \right) [\text{dB}]$$

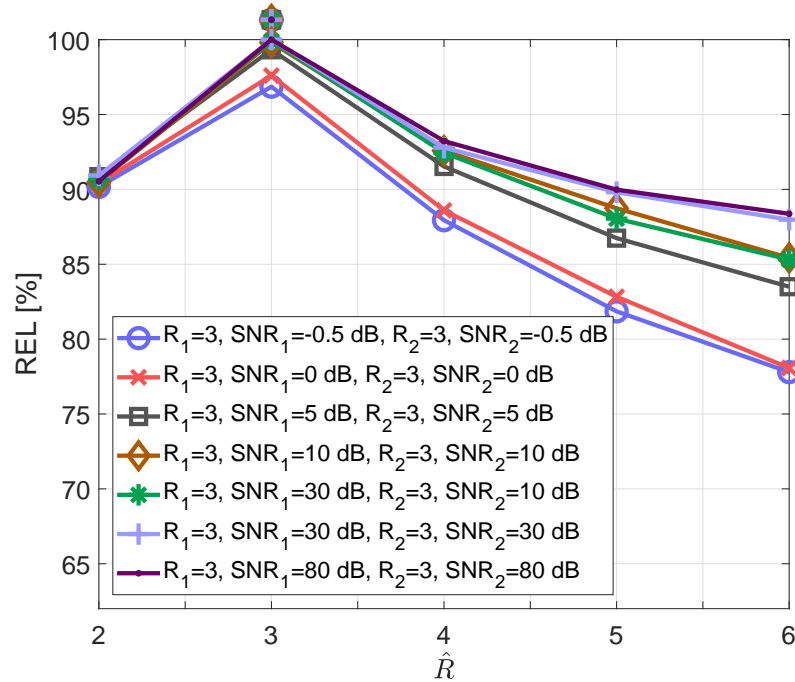


Figure 3.25.: Reliability as a function of the assumed rank \hat{R} for different SNRs.

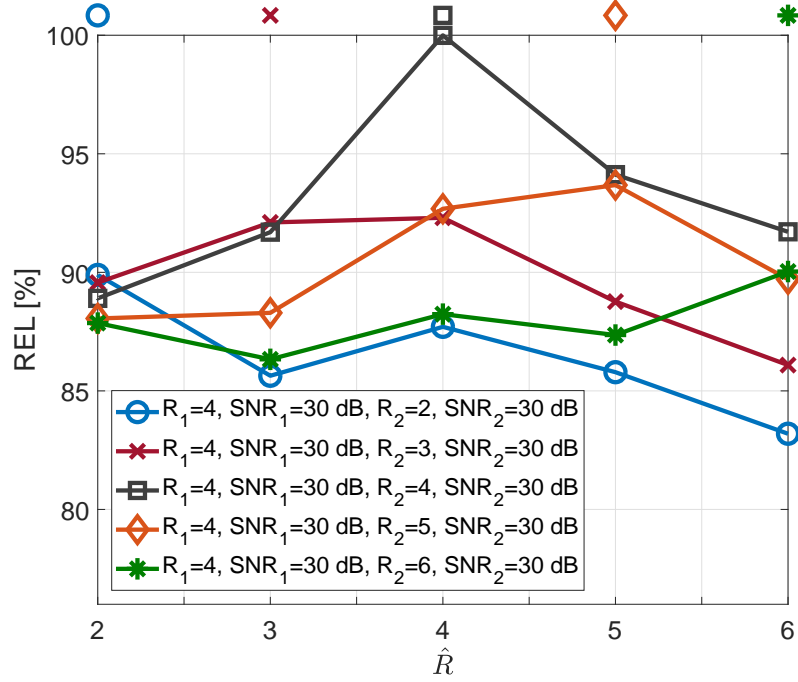


Figure 3.26.: Reliability as a function of the assumed rank \hat{R} for different ranks.

In Fig. 3.25, we visualize the typical reliability as a function of the assumed rank \hat{R} . These curves result from Monte Carlo simulations with 1000 realizations, for real-valued tensors with dimensions $8 \times 8 \times 8$. The true tensor rank and the corresponding SNRs are indicated in the legend in Figs. 3.25 and 3.26, whereas the assumed rank \hat{R} was varied from two to six. The true tensor rank for each curve is additionally indicated with a marker above the curves. In Fig. 3.25, it is obvious that we have maximum reliability when the assumed rank equals the exact tensor rank. Moreover, as expected the SNR influences the reliability measure due to the estimates dependency of the SNR. However, even in a low SNR regime (blue curve corresponding to $\text{SNR}_1 = \text{SNR}_2 = -0.5$ dB) the reliability is more than 95% when the assumed rank equals the true rank.

Moreover, in Fig. 3.26 we depict the reliability for two tensors that share different numbers of components. For instance, for the light blue curve (the first curve indicated in the legend), the first tensor has rank $R_1 = 4$, whereas the second tensor has rank $R_2 = 2$. This implies that the tensors share only two components and the first tensor with tensor rank $R_1 = 4$ has two additionally components. For the remaining curves, the first tensor has rank $R_1 = 4$, whereas the second tensor has rank $R_2 = 3$, $R_2 = 4$, $R_2 = 5$, and $R_2 = 6$, respectively. The ranks of the second tensor for each scenario are additionally indicated using a marker at the top of the Fig. 3.26. Details for the true tensor ranks and the SNRs are available also in the legend. In Fig. 3.26, we see that the reliability has local maxima for both ranks R_1 and R_2 .

When performing signal analysis using the CP decomposition, we are typically interested in the factor matrices, as their columns represent the signatures of the underlying components for the

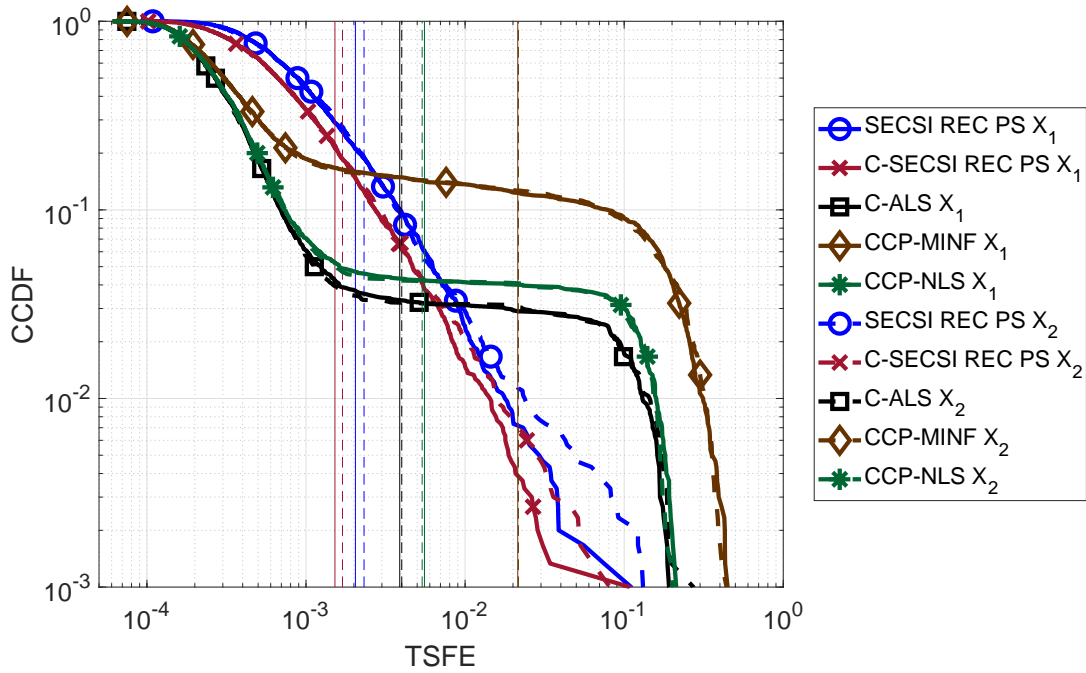


Figure 3.27.: CCDF of the TSFE for real-valued tensors with dimensions $40 \times 4 \times 10$, tensor ranks $R_1 = R_2 = 3$, factor matrices with mutually correlated columns designed as sine functions, and $\text{SNR}_1 = \text{SNR}_2 = 25$ dB.

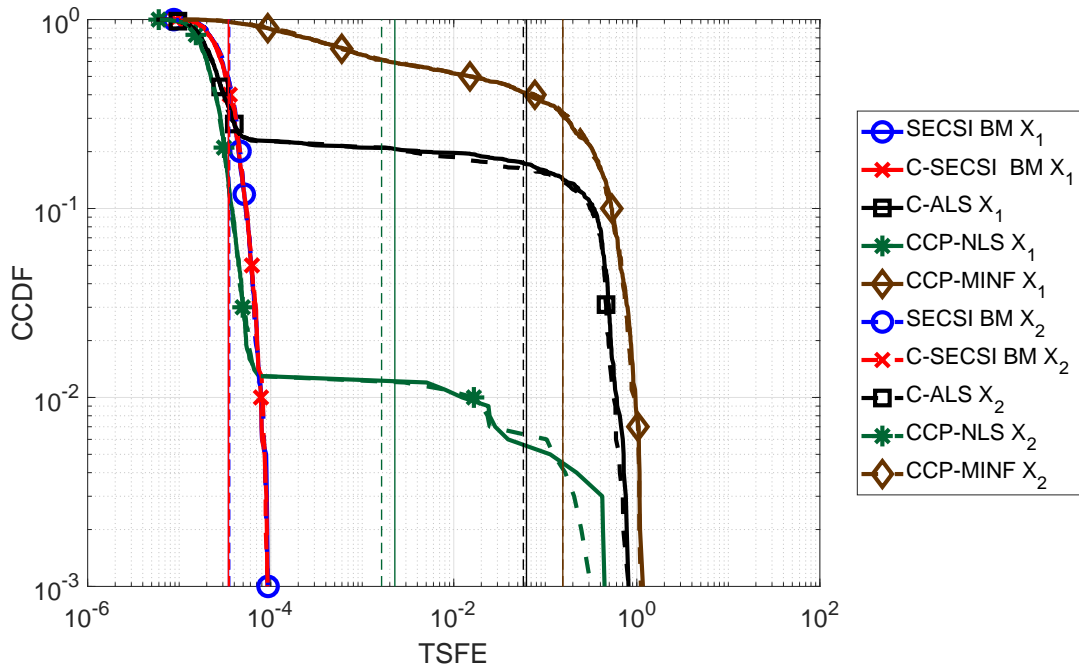


Figure 3.28.: CCDF of the TSFE for real-valued tensors with dimensions $80 \times 80 \times 80$, tensor ranks $R_1 = R_2 = 4$, and $\text{SNR}_1 = \text{SNR}_2 = 30$ dB. The first (common) factor matrix has mutually correlated columns with a correlation coefficient $\rho_1 = 0.98$.

corresponding dimension. Therefore, an important measure for algorithm comparison is the TSFE defined in (3.11). In Fig. 3.27, we compare the performance of the C-SECSI [NH16], SECSI [RH13a], C-ALS, CCP-NLS [VDS⁺16] and CCP-MINF [VDS⁺16] for two real-valued tensors of size $40 \times 4 \times 10$, $R_1 = R_2 = 3$, and first mode in common. The SECSI framework and the C-SECSI framework select the final solution based on the heuristic REC PS. The three signatures of the first factor matrix represent the first 40 samples of sine functions, $\sin(2\pi t f_1 + \frac{\pi}{3})$, $\sin(2\pi t f_2)e^{10t}$, and $\sin(2\pi t f_3)e^{-3t}$ with $f_1 = 10$ Hz, $f_2 = 20$ Hz, and $f_3 = 30$ Hz. The second and the third factor matrices were drawn from zero mean Gaussian random process with variance one. Moreover, the third factor matrices have collinear columns with a correlation coefficient $\rho_3 = 0.9$. The correlation is added according to equation (3.10). The CCDF of the TSFE for SNR equal to 25 dB is depicted in Fig. 3.27. The vertical lines represent the mean TSFE value for each curve. SECSI and C-SECSI do not have outliers even for such an ill-conditioned scenario in contrast to the other algorithms. Note that from all considered algorithms only SECSI is not a coupled algorithm and we are able to observe an accuracy gain of the C-SECSI framework compared to the uncoupled SECSI framework. The C-SECSI framework outperforms the rest of the algorithms in terms of mean the TSFE.

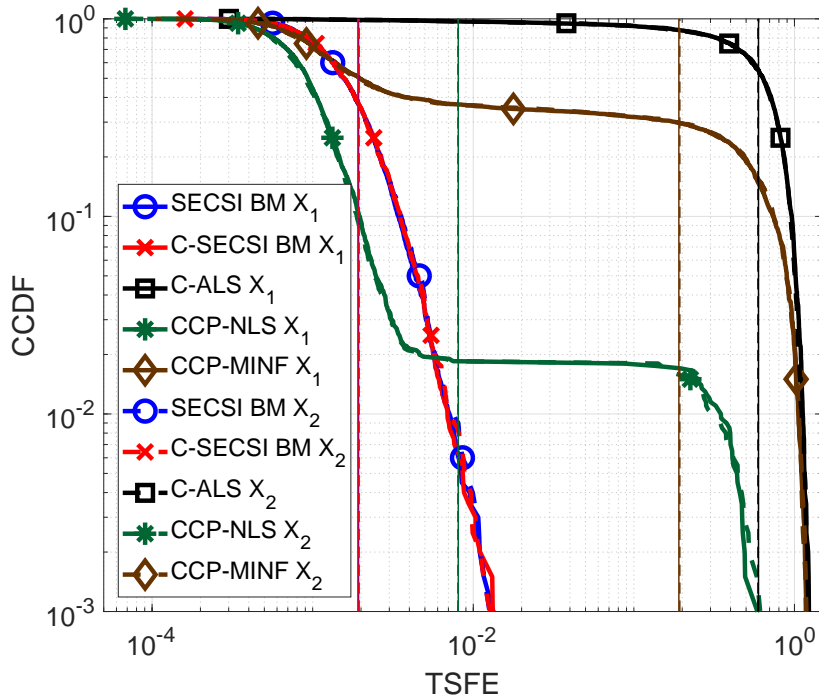


Figure 3.29.: CCDF of the TSFE for complex-valued tensors with dimensions $4 \times 8 \times 7$, tensor rank $R_1 = R_2 = 3$, and $\text{SNR}_1 = \text{SNR}_2 = 45$ dB. The first (common) factor matrix has mutually correlated columns.

Next, we compare the performance of the six algorithms for two real-valued tensors, \mathcal{X}_1 and \mathcal{X}_2 of size $80 \times 80 \times 80$. The two tensors have the first factor matrix in common, and additionally the common factor matrix has collinear columns with a correlation factor $\rho_1 = 0.98$. The CCDF of the

TSFE for $\text{SNR}_1 = \text{SNR}_2 = 30$ dB is presented in Fig. 3.28. In Fig. 3.28, we observe that C-SECSI outperforms the rest of the algorithms, however it has the same accuracy as the original SECSI framework. Here, both frameworks SECSI and C-SECSI use the BM criterion to choose the final solution.

In Fig. 3.29, the CCDF of the TSFE is presented for two tensors of size $4 \times 8 \times 7$ with a common 1-mode. For this scenario, the common factor matrix is chosen as

$$\mathbf{F}_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0.95 & 0.95 \\ 1 & 0.95 & 1 \\ 1 & 1 & 0.95 \end{bmatrix}.$$

This factor matrix is ill-conditioned and has highly correlated columns, and the CP decomposition containing this factor matrix is very difficult to calculate. From the Fig. 3.29 it is noticeable that C-ALS fails in most of the attempts to decompose the given tensors. However, the SECSI framework and the C-SECSI framework using BM are still able to decompose the tensors. Moreover, C-SECSI shows a better performance than CCP-MINF and CCP-NLS.

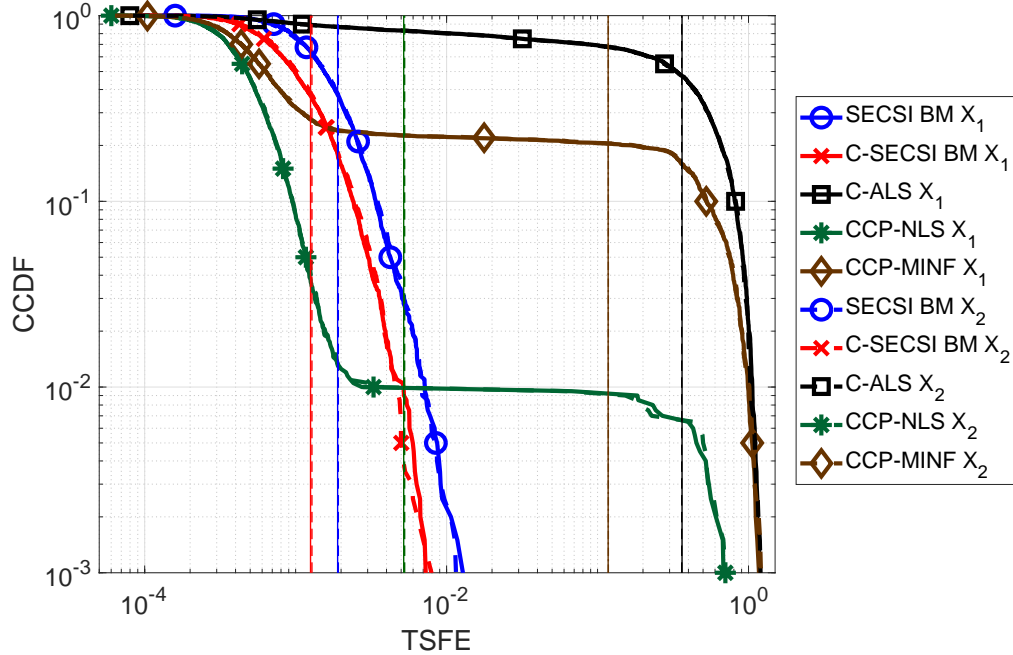


Figure 3.30.: CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 8 \times 4$, tensor ranks $R_1 = R_2 = 3$, and $\text{SNR}_1 = \text{SNR}_2 = 45$ dB. The third factor matrices have mutually correlated columns.

Similarly, in Fig. 3.30 we compare the performance of the above discussed algorithms for an

ill-conditioned scenario, where the third factor matrices are chosen as $\mathbf{F}_3^{(1)}$ and $\mathbf{F}_3^{(2)}$ from (3.46).

$$\mathbf{F}_3^{(1)} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0.95 & 0.95 \\ 1 & 0.95 & 1 \\ 1 & 1 & 0.95 \end{bmatrix} \quad \mathbf{F}_3^{(2)} = \begin{bmatrix} 0.95 & 1 & 0.95 \\ 1 & 1 & 1 \\ 0.95 & 1 & 1 \\ 1 & 1 & 0.95 \end{bmatrix} \quad (3.46)$$

The two tensors are complex-valued with dimensions $7 \times 8 \times 4$ and they have the first mode in common. The CCDF of the TSFE for $\text{SNR}_1 = \text{SNR}_2 = 45$ dB is depicted in Fig. 3.30. In this scenario, where the ill-conditioned factor matrix is not the common mode, we are able to observe an accuracy gain compared to the uncoupled SECSI framework.

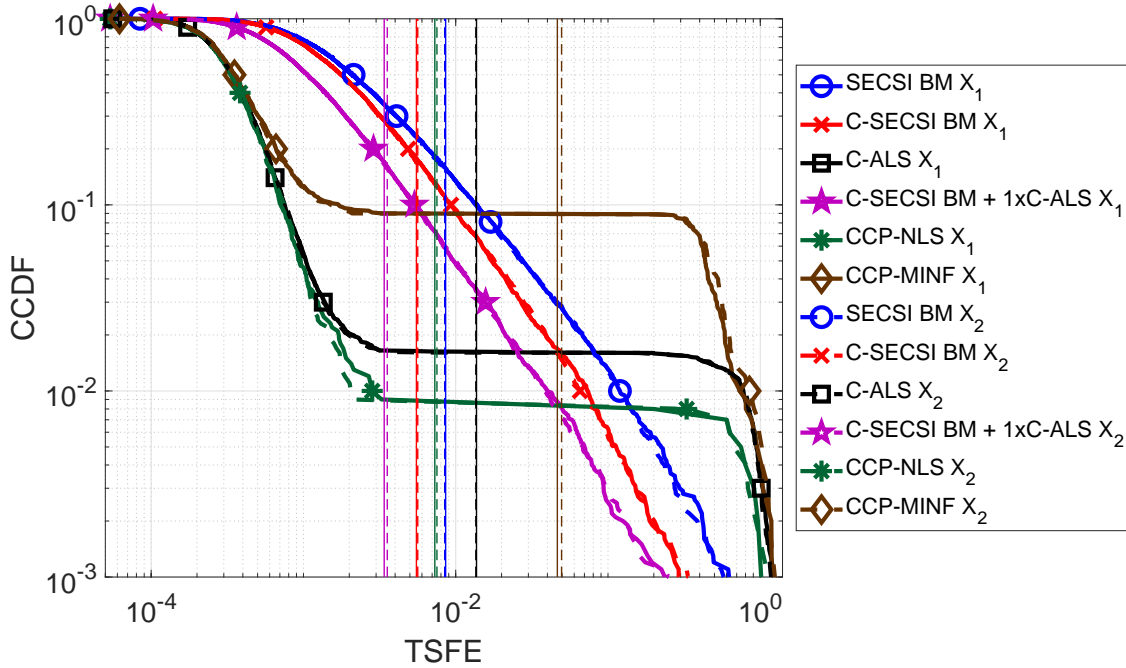


Figure 3.31.: CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 3 \times 4$, tensor ranks $R_1 = R_2 = 4$, and $\text{SNR}_1 = \text{SNR}_2 = 30$ dB. Both tensors are degenerate with respect to the 2-modes, i.e., $R_1 > M_2^{(1)}$ and $R_2 > M_2^{(2)}$.

Moreover, since the SECSI framework is able to estimate the factor matrices even in a degenerate case, when the rank of the tensor exceeds the tensor size in at least one of the modes, we have also simulated such a scenario. The tensors are of size $7 \times 3 \times 4$ with ranks $R_1 = R_2 = 4$, hence the two tensors are degenerate in mode two, but still have the first factor matrix in common. The CCDF of the TSFE for $\text{SNR}_1 = \text{SNR}_2 = 30$ dB is visualized in Fig. 3.31. Moreover, in this figure we show the performance of the C-SECSI framework plus one C-ALS iteration, denoted by C-SECSI BM + 1xC-ALS. In this case, the C-SECSI framework outperforms the SECSI framework. If they converge, the C-ALS, CCP-NLS, and CCP-MINF provide a more accurate estimate, but in some cases they do not converge at all. Therefore, the mean error is larger than for the C-SECSI

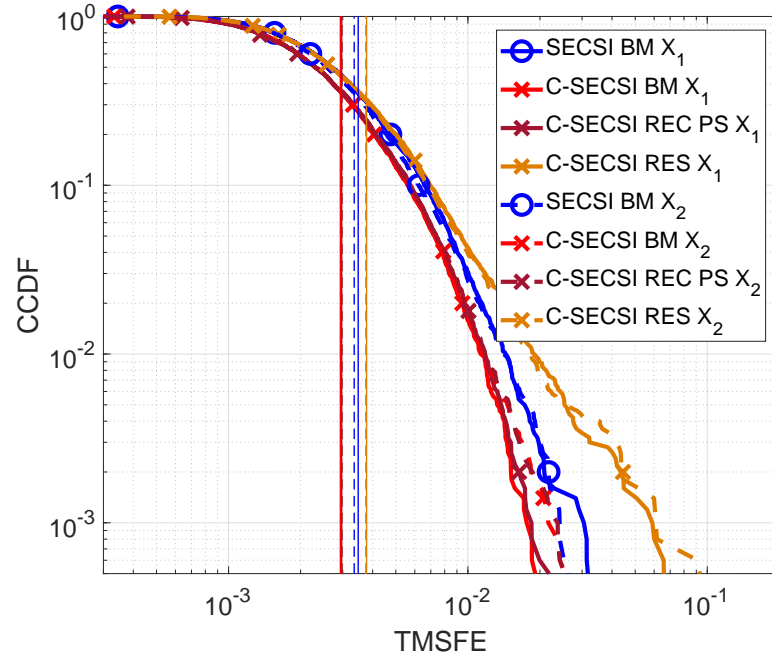


Figure 3.32.: CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 7 \times 7$, tensor ranks $R_1 = R_2 = 3$, $\text{SNR}_1 = \text{SNR}_2 = 30$ dB, and the factor matrices have mutually correlated columns with correlation coefficients $\rho_1 = 0.1$, $\rho_2 = 0.9$, and $\rho_3 = 0.1$.

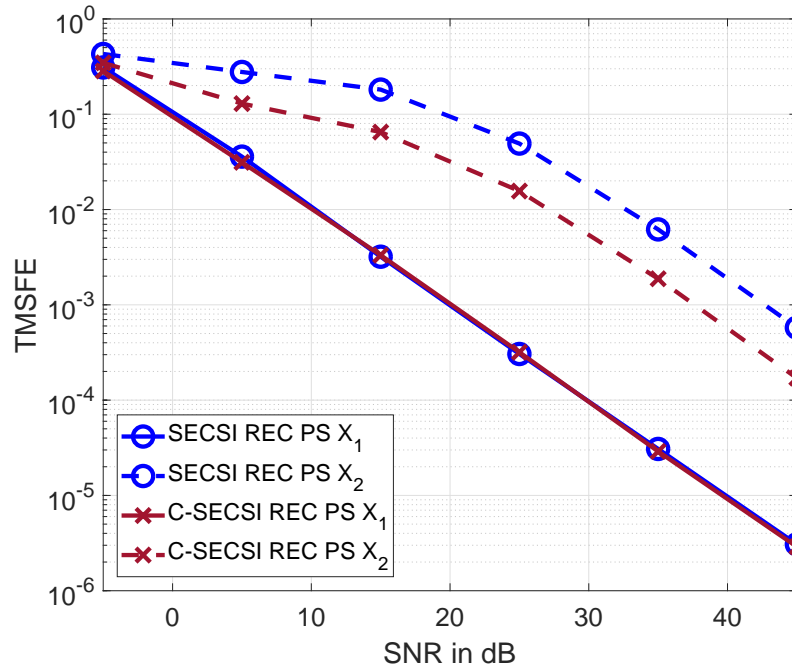


Figure 3.33.: TMSFE as a function of the SNR for complex-valued tensors with dimensions $4 \times 8 \times 7$, tensor ranks $R_1 = R_2 = 3$, where the second tensor has third factor matrix with mutually correlated columns with correlation coefficient of 0.98.

framework. Furthermore, in Fig. 3.31 we observe that already a single iteration of C-ALS improves the accuracy of the C-SESCI framework additionally.

In Fig. 3.32, we compare the accuracy of the different heuristics. Therefore, we depict in Fig. 3.32 the CCDF of the TSFE for complex-valued tensors with dimensions $7 \times 7 \times 7$, tensor ranks $R_1 = R_2 = 3$, and $\text{SNR}_1 = \text{SNR}_2 = 30$ dB. Moreover, the factor matrices have mutually correlated columns with correlation coefficients $\rho_1 = 0.1$, $\rho_2 = 0.9$, and $\rho_3 = 0.1$, for both tensors. According to Fig. 3.32, the BM and REC PS have comparable accuracy in terms of TSFE, but REC PS has lower computational complexity. The heuristic RES has the lowest computational complexity and the lowest accuracy.

Moreover, in Fig. 3.33 we depict the TMSFE (i.e., mean TFSE) for different SNRs. Both tensors \mathcal{X}_1 and \mathcal{X}_2 with common first mode have dimensions $40 \times 4 \times 10$ and tensor ranks $R_1 = R_2 = 3$. However, only the third factor matrix of the second tensor \mathcal{X}_2 has mutually correlated columns, with a correlation coefficient of 0.98. This highly correlated factor matrix causes the tensor \mathcal{X}_2 to be ill-conditioned. The Fig. 3.33 shows that using the coupled algorithm improves the estimate accuracy of the ill-conditioned tensor without corrupting the good conditioned tensor.

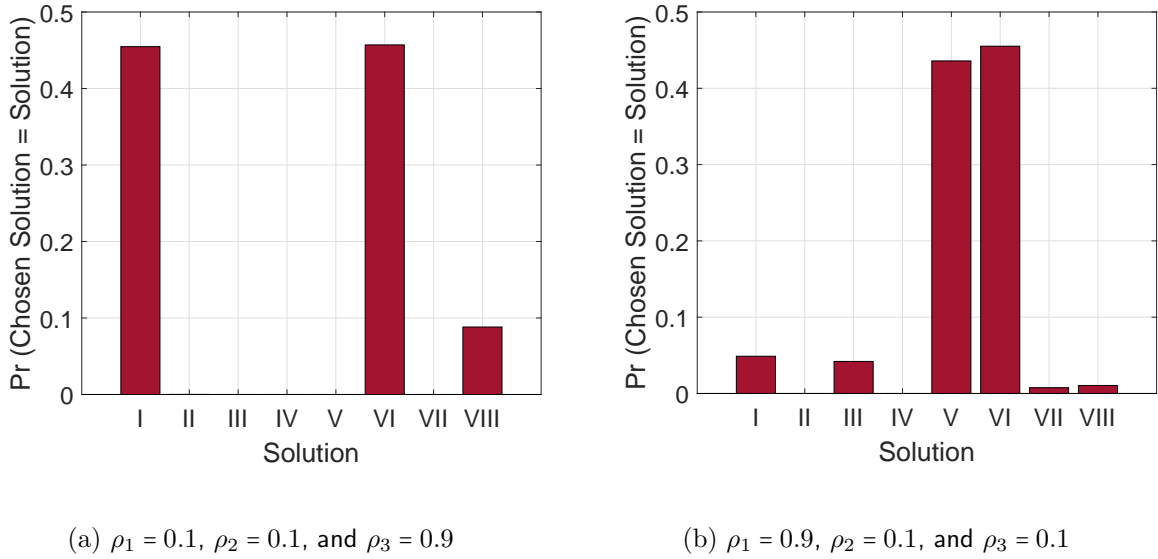


Figure 3.34.: Probability of selecting the final estimate for the C-SESCI framework using the REC PS heuristic. The decomposed complex-valued tensors have dimensions $4 \times 7 \times 8$, common 1-mode, ranks $R_1 = R_2 = 3$, $\text{SNR}_1 = \text{SNR}_2 = 30$ dB, and correlated factor matrices with correlation coefficients ρ_1 , ρ_2 , and ρ_3 .

As observed in Figs. 3.27, 3.30-3.33 when the common matrix is not ill-conditioned, i.e., its columns are not mutually correlated, C-SESCI has a higher accuracy than SECSI. On the other hand, Figs. 3.28 and 3.29 show that C-SESCI is as accurate as SECSI in terms of TSFE when the common matrix is ill-conditioned. In order to investigate the reason which leads to the improved accuracy, we study the REC PS heuristics and selection of the final solution. Recall that C-SESCI computes eight sets of initial estimates of the factor matrices that are depicted in Fig. 3.24 and

accordingly numbered I-VIII. In Fig. 3.34, we depict the histogram of the probability that one of the I-VIII solutions is chosen as a final solution, for the tensor \mathcal{X}_1 . Moreover, we consider two scenarios illustrated in Figs. 3.34a and 3.34b. For both scenarios, we consider complex-tensors with dimensions $4 \times 7 \times 8$, common 1-mode, ranks $R_1 = R_2 = 3$, and $\text{SNR}_1 = \text{SNR}_2 = 30$ dB. The tensors in the first scenario depicted in Fig. 3.34a have correlated factor matrices with correlation coefficients $\rho_1 = 0.1$, $\rho_2 = 0.1$, and $\rho_3 = 0.9$. The tensors in the second scenario depicted in Fig. 3.34b have correlated factor matrices with correlation coefficients $\rho_1 = 0.9$, $\rho_2 = 0.1$, and $\rho_3 = 0.1$. Hence, Fig. 3.34a depicts the probability of choosing the final solution when the common mode is not ill-conditioned, whereas Fig. 3.34b depicts the probability of choosing the final solution when the common mode is ill-conditioned. The solutions I, VI and VII are chosen with the highest probability when the 3-mode is ill-conditioned. Comparing to Fig. 3.24, these most frequently chosen solutions are corresponding to the initial solutions that have either the ill-conditioned factor matrix or the coupled one in the diagonal elements. Solution I is the coupled solution that leads to an improved accuracy. On the other hand, in Fig. 3.34b the solutions V and VI are chosen as final solution with the highest probability. Hence, the most frequently chosen solutions are when the common factor matrix is in the diagonal elements, that is at the same time the ill-conditioned matrix. These most frequently chosen solutions are not coupled. Therefore, the performance is equivalent to the original SECSI framework for scenarios when the common mode has correlated columns.

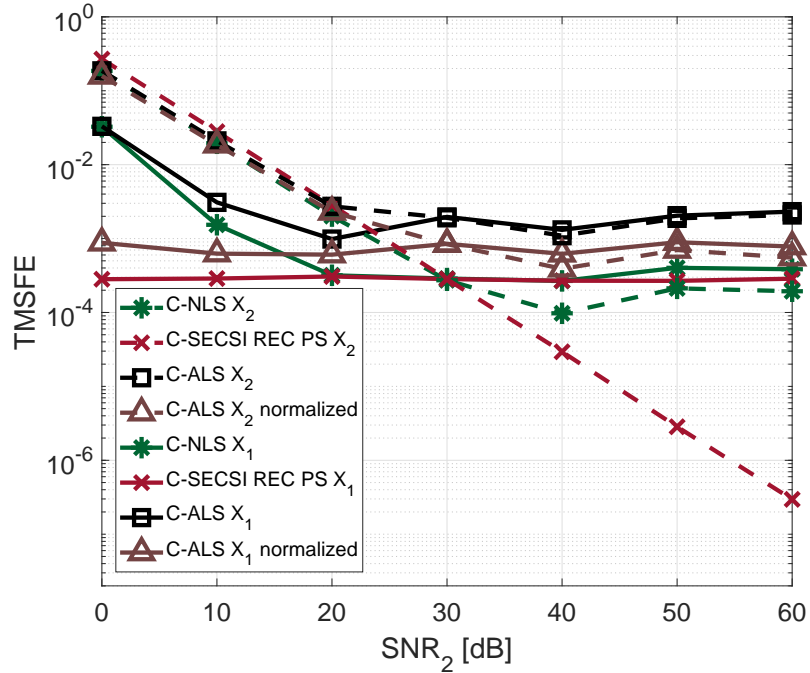


Figure 3.35.: TMSFE as a function of the SNR_2 for complex-valued tensors with dimensions $3 \times 8 \times 7$, tensor ranks $R_1 = R_2 = 3$, and $\text{SNR}_1 = 30$ dB.

Furthermore, in Fig. 3.35 we show that the C-SECSI framework unlike other algorithms can jointly decompose coupled tensors even if they are affected by noise with different variance. The

tensors \mathcal{X}_1 and \mathcal{X}_2 with common first mode have dimensions $3 \times 8 \times 7$, and tensor ranks $R_1 = R_2 = 3$. The factor matrices have complex valued values drawn from ZMCSCG random process with variance one. The first tensor has a constant SNR_1 of 30 dB, while the SNR_2 is varied from 0 to 60 dB. These results are averaged over 3000 realizations. The C-ALS normalized denotes the C-ALS algorithm with additional normalization with respect to the different noise variances. Note that in order to perform the normalization the noise variance has to be priorly estimated or known. For the results presented in Fig. 3.35, the noise variance was estimated. Further details on the importance of normalization and compression with truncated HOSVD for ALS are available in [CFC16]. However, normalization with respect to the noise variance is not required when computing the coupled CP decomposition using the C-SECSI framework. Therefore, C-SECSI is a very suitable and robust tool for applications when the coupled tensors have a different nature and different SNRs.

3.6.2 Summary

In this section, we present the C-SECSI framework for the efficient computation of an approximate coupled CP decomposition. For 3-way tensors, the C-SECSI framework results in $6 + 2N_c$ initial sets of estimates of the factor matrices, where $N_c = 1, 2$ is the number of common modes. The final estimate can be selected based on different heuristics that lead to different complexity-accuracy trade-offs of the C-SECSI framework. C-SECSI computes the coupled CP decomposition under the constraint that one of the modes is coupled, but it still computes uncoupled estimates as well. This is very practical for the analysis of biomedical data, where the coupling is assumed, but not yet proven. Moreover, for comparing the independently chosen final estimates we have defined the coupling reliability. With simulations we show that the reliability can be used to control the rank of the coupled decomposition. This is a very important feature of C-SECSI because the rank estimate is a very challenging problem, especially for noisy measurement signals. Moreover, we have compared the C-SECSI framework with the original SECSI framework as well as with other state-of-the-art algorithms and shown that it outperforms these algorithms. The C-SECSI framework has a higher accuracy in ill-conditioned scenarios such as computing the coupled CP decomposition with factors with correlated columns. Even more, we observe an accuracy gain as compared to the traditional SECSI framework proposed in [RH13a]. This accuracy gain originates from the coupled solutions. Another advantage of the C-SECSI framework is that it can decompose tensors that are corrupted by noise with different variances without any additional normalization or estimation of the SNRs. Future work includes extensions to coupled matrix-tensor decompositions and coupling of more than two tensors that are straightforward. The C-SECSI framework has been published in [NH16, NKHH17].

3.7 Conclusions

The SECSI framework [RH08,RSH12,RH13a] is a robust semi-algebraic framework for the computation of an approximate low-rank CP decomposition. It computes all possible symmetric SMDs that lead to six initial estimates of the factor matrices, for a 3-way tensor. The final estimate is then selected based on different heuristics as discussed in [RH13a], that lead to different complexity-accuracy trade-off of the SECSI framework. In this chapter, we have proposed five extensions of the SECSI framework that reduce the computational complexity of the original framework or introduce constraints to the factor matrices. These extensions include T-SECSI (Truncated-SECSI), NS-SECSI (Non-Symmetric-SECSI), S-SECSI (Symmetric-SECSI/SECSI for symmetric tensors), SECSI₊ (Non-negative SECSI/SECSI for non-negative tensors), and C-SECSI (Coupled-SECSI/SECSI for coupled tensors). The T-SECSI framework has lower computational complexity than SECSI because it diagonalizes a compressed core tensor for size $R \times R \times R$, where R is the tensor rank. This gain is more pronounced as the tensor dimensions increase. Therefore, we recommend the T-SECSI framework for the computation of the CP decomposition for tensors with large dimensions. The NS-SECSI framework exploits non-symmetric SMDs for the computation of the CP decomposition. We consider two algorithms for the computation of the non-symmetric SMDs the TEDIA [TPC15] and the NS-IDIEM [CB12,CKM⁺14] algorithms. The NS-SECSI-NS-IDIEM framework is a closed-form (algebraic) solution for the computation of an approximate low-rank CP decomposition. Therefore, it has even lower computational complexity than the T-SECSI. Moreover, NS-SECSI computes only three initial sets of estimates of the factor matrices that additionally reduces the computational complexity of the selection of the final solution. The NS-SECSI-NS-IDIEM framework has lower accuracy than the original SECSI framework, for tensors with dimensions comparable to the tensor rank. Therefore, we recommend NS-SECSI-NS-IDIEM for the computation of an approximate CP decomposition for large tensors or for applications in which the CP decomposition has to be approximated very fast and very efficiently. The S-SECSI framework is a closed-form solution for the computation of the CP decomposition of symmetric tensors or fully symmetric tensors. In a way, it represents an extension of the NS-SECSI-NS-IDIEM framework because it exploits the same closed form solution for the computation of the SMDs, namely the algorithms IDIEM and NS-IDIEM [CB12,CKM⁺14]. Therefore, it computes a symmetric CP very fast in a closed form fashion. The computational advantages provided by the truncation (included in T-SECSI and NS-SECSI-TEDIA) and the closed form solution (included in NS-SECSI-NS-IDIEM and S-SECSI) are more pronounced as the tensor dimensions increase. The SECSI₊ framework computes an approximated CP decomposition of non-negative tensors under the constraint that the factor matrices are also non-negative. We propose to compress the non-negative tensor based on NTD (Non-negative Tucker Decomposition) and then to compute a symmetric SMDs with non-negative constraints. Therefore, we also propose an ADMMD₊ diagonalization algorithm for the computation of SMDs with non-negative constraints. However, this C-SECSI framework has no advantages over the state-of-the-art algorithms. In order to potentially increase its accuracy, we propose to further investigate the proposed solution, to consider other NTD algo-

rithms, and to consider non-symmetric SMDs as proposed in the NS-SECSI framework. Finally, the C-SECSI framework computes the coupled CP decomposition in a robust semi-algebraic fashion. The C-SECSI framework outperforms the state-of-the-art algorithms especially in ill-conditioned scenarios. It does not require prior normalization of the tensors even if they are corrupted by noise with different variances. Moreover, we propose a reliability that controls the rank of the coupled tensor decomposition. In the future, it is possible to extend the C-SECSI framework to coupled matrix-tensor decompositions and to coupled CP for more than two tensors. A closed-form solution of C-SECSI can also be obtained based on the IDIEM [CB12] algorithm. Note that, in the future the SECSI framework and the C-SECSI framework can be extended by considering sparse tensors and tensors with missing entries by utilizing the algorithm proposed in [YFLZ16] as a first step instead of the truncated HOSVD. The NS-SECSI framework and the C-SECSI framework have already been published in [NHT⁺16, NKHH17] and [NH16], respectively.

Chapter 4

PARATUCK2 and PARAFAC2 via constrained CP model

In this chapter, we consider the PARATUCK2 and the PARAFAC2 decomposition. The PARATUCK2 decomposition represents a mixture of two basic decompositions, the CP decomposition and the HOSVD. Therefore, it offers more flexibility than the CP decomposition. The PARAFAC2 decomposition is a generalization of the PARAFAC/CP tensor decomposition. This tensor decomposition represents a set of coupled matrix decompositions with one mode in common, i.e., one of the components varies along the set of matrices (tensor slices), whereas the second component stays constant. Compared to CP, PARAFAC2 allows a variation of the two-mode factors over the three-mode dimension and thus has an increased flexibility. Both tensor decompositions PARATUCK2 and PARAFAC2 have many applications including the analysis of chromatographic data, data clustering, modeling of wireless communication systems, and biomedical signal processing [dAFX13, CHGH18] (see also Sections 2.2.4, 2.2.5, and 2.3). Typically, these two decompositions are described by means of a slice-wise multiplication between two tensors. In this chapter, we develop new tensor models for the PARATUCK2 decomposition and the PARAFAC2 decomposition using the generalization of the slice-wise multiplication based on the tensor contraction operator proposed in Section 2.1.4. In Section 4.1 and Section 4.2, we express the PARATUCK2 decomposition and the PARAFAC2 decomposition in terms of these new tensor models resulting in constrained CP models. Moreover, we show that these new models open efficient ways for the computation of these decompositions. For instance, we show that this new description leads to an efficient single loop algorithm to compute the PARAFAC2 decomposition. The novel tensor models proposed in this chapter describe the noiseless tensor. The algorithms that estimate the noiseless tensor from noise corrupted measurements depend on the noise model (see also equation (2.34)). Here, we assume Gaussian noise. Therefore, the algorithm proposed in this chapter estimates the low-rank tensor from noisy observations in an LS sense. However, the tensor models proposed here can also be used as a starting point in the derivation of other more advanced algorithms.

4.1 PARATUCK2

The PARATUCK2 decomposition [HL96] is a very flexible decomposition that is usually described using slice-wise or element-wise notation. However, these descriptions do not reveal the explicit tensor structure. In Section 2.2.4, we have reviewed the PARATUCK2 decomposition, its uniqueness

properties, and its applications.

We visualize the slice-wise notation from equation (2.46) of the PARATUCK2 decomposition in Fig. 4.1.

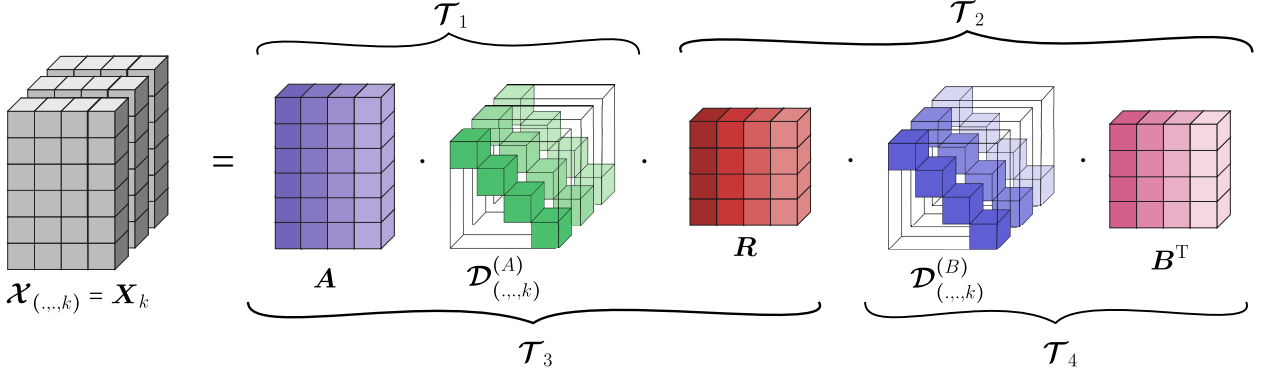


Figure 4.1.: Slice-wise visualization of the PARATUCK2 decomposition for a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ with slices $\mathcal{X}_{(.,.,k)} = \mathbf{A} \cdot \mathcal{D}_{(.,.,k)}^{(A)} \cdot \mathbf{R} \cdot \mathcal{D}_{(.,.,k)}^{(B)} \cdot \mathbf{B}^T$.

Using an element-wise notation, the PARATUCK2 tensor decomposition defined in Section 2.2.4 equation (2.46) of a tensor $\mathcal{X} \in \mathbb{C}^{I \times T \times K}$ is defined as [HL96]

$$\mathcal{X}_{(i,t,k)} = \sum_{j=1}^J \sum_{p=1}^P a_{i,j} d_{j,k}^{(A)} r_{j,p} d_{p,k}^{(B)} b_{t,p}.$$

Let us now define the matrices $\mathbf{A} \in \mathbb{C}^{I \times J}$, $\mathbf{B} \in \mathbb{C}^{T \times P}$, $\mathbf{R} \in \mathbb{C}^{J \times P}$, $\mathbf{U} \in \mathbb{C}^{J \times K}$, and $\mathbf{V} \in \mathbb{C}^{P \times K}$ with elements, $\mathbf{A}_{(i,j)} = a_{i,j}$, $\mathbf{B}_{(t,p)} = b_{t,p}$, $\mathbf{R}_{(j,p)} = r_{j,p}$, $\mathbf{U}_{(j,k)} = d_{j,k}^{(A)}$, and $\mathbf{V}_{(p,k)} = d_{p,k}^{(B)}$, respectively. Moreover, we define the tensors $\mathcal{D}^{(A)} = \mathcal{I}_{3,J} \times_3 \mathbf{U}^T$ and $\mathcal{D}^{(B)} = \mathcal{I}_{3,P} \times_3 \mathbf{V}^T$.

Derivation of the structure illustrated at the top of Fig. 4.1

Next, we define $\mathcal{T}_1 = \mathcal{I}_{3,J} \times_1 \mathbf{A} \times_3 \mathbf{U}^T$ and $\mathcal{T}_2 = \mathcal{I}_{3,P} \times_1 \mathbf{R} \times_2 \mathbf{B} \times_3 \mathbf{V}^T$ as illustrated in Fig. 4.1. This definition follows directly from the element-wise notation and the above defined matrices. Considering the tensors $\mathcal{T}_1 \in \mathbb{C}^{I \times J \times K}$ and $\mathcal{T}_2 \in \mathbb{C}^{J \times T \times K}$, the tensor \mathcal{X} can be expressed as

$$\mathcal{X}_{(.,.,k)} = \mathcal{T}_{1(.,.,k)} \cdot \mathcal{T}_{2(.,.,k)}.$$

Hence, PARATUCK2 can be interpreted as a slice-wise multiplication between the two tensors \mathcal{T}_1 and \mathcal{T}_2 . In Section 2.1.4, we have proposed an alternative representation of the slice-wise multiplication based on generalized tensor contraction. More precisely, we propose to substitute the slice-wise multiplication by a double contraction. The first contraction is defined along the modes that are multiplied and the second contraction is defined along the slices $k = 1, \dots, K$ (the dimension that is unaffected). Moreover, as shown in Section 2.1.4, one of the two tensors should

be diagonalized beforehand along the unaffected dimension. Hence, for the tensor \mathcal{X} , we have

$$\mathcal{X}' = \mathcal{T}_{1,D} \bullet_{2,3}^{1,3} \mathcal{T}_2 \in \mathbb{C}^{I \times K \times T} \quad \text{or} \quad \mathcal{X} = \mathcal{T}_1 \bullet_{2,3}^{1,3} \mathcal{T}_{2,D} \in \mathbb{C}^{I \times T \times K}, \quad (4.1)$$

where $\mathcal{T}_{1,D} \in \mathbb{C}^{I \times J \times K \times K}$ and $\mathcal{T}_{2,D} \in \mathbb{C}^{J \times T \times K \times K}$ are the diagonalized tensors with non-zero elements $\mathcal{T}_{1,D}(i,j,k,k) = \mathcal{T}_1(i,j,k)$ and $\mathcal{T}_{2,D}(j,t,k,k) = \mathcal{T}_2(j,t,k)$. The diagonal structure of these tensors can be expressed by the means of the Khatri-Rao product between an identity matrix and a generalized unfolding as shown in Table 2.1. Hence, we have $[\mathcal{T}_{1,D}]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [\mathcal{T}_1]_{([1,2],3)}$ and $[\mathcal{T}_{2,D}]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [\mathcal{T}_2]_{([1,2],3)}$. Note that the two tensors \mathcal{X}' and \mathcal{X} in (4.1) contain the same elements, but have permuted dimensions due to the definition of the contraction operator in equation (2.9). The different permutation can be easily resolved for the final results by permuting the modes of the resulting tensor.

First, let us derive the resulting tensor structure of the tensor \mathcal{X} beginning from $\mathcal{X}' = \mathcal{T}_{1,D} \bullet_{2,3}^{1,3} \mathcal{T}_2 \in \mathbb{C}^{I \times K \times T}$. By substituting the structure of \mathcal{T}_1 into the unfolding of the diagonalized tensor, we get

$$[\mathcal{T}_{1,D}]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [\mathcal{T}_1]_{([1,2],3)} = \mathbf{I}_K \diamond [(\mathbf{I}_J \diamond \mathbf{A}) \cdot \mathbf{U}].$$

Next, by considering the tensor structure of $\mathcal{T}_{1,D}$, it can be shown that the tensor $\mathcal{T}_{1,D}$ satisfies a BTD. In Appendix B.1, we show that the diagonalized tensor satisfies a BTD, if the tensor that is diagonalized has a CP structure. The result used here can be directly deduced from the results presented in Appendix B.1 taking into account that $\mathcal{T}_1 = \mathcal{I}_{3,J} \times_1 \mathbf{A} \times_3 \mathbf{U}^T$. Hence, for the tensor $\mathcal{T}_{1,D}$, we get

$$\mathcal{T}_{1,D} = \sum_{k=1}^K (\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,J}) \times_1 \mathbf{A} \times_2 \mathbf{I}_J \times_3 (\mathbf{e}_k \otimes \mathbf{U}_{(:,k)}^T) \times_4 \mathbf{e}_k. \quad (4.2)$$

The tensor $(\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,J})$ has dimensions $J \times J \times J \times 1$ and the operator \otimes corresponds to the tensor Kronecker product that was introduced in Section 2.1.1 (see Fig. 2.4). Also, as shown in Section 2.2.3 (see Fig. 2.15) the sum can be replaced by a block diagonal core and factor matrices partitioned accordingly. Moreover, the block diagonal structure of the core tensor built from the K tensors $(\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,J})$ in equation (4.2) can be defined as $(\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}) \in \mathbb{C}^{JK \times JK \times JK \times K}$. Similarly, the structure of the partitioned matrix $\begin{bmatrix} \mathbf{A} & \mathbf{A} & \dots & \mathbf{A} \end{bmatrix}$ can be represented by $\mathbf{1}_K^T \otimes \mathbf{A}$. For the partitioned matrix $\begin{bmatrix} (\mathbf{e}_1 \otimes \mathbf{U}_{(:,1)}^T) & (\mathbf{e}_2 \otimes \mathbf{U}_{(:,2)}^T) & \dots & (\mathbf{e}_K \otimes \mathbf{U}_{(:,K)}^T) \end{bmatrix}$ we can show that it equals $(\mathbf{I}_K \diamond \mathbf{U})^T$ (see Fig. B.2 in Appendix B.1). Using all of the matrices defined above and the core tensor, for the tensor $\mathcal{T}_{1,D}$, we have

$$\mathcal{T}_{1,D} = (\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}) \times_1 (\mathbf{1}_K^T \otimes \mathbf{A}) \times_2 (\mathbf{1}_K^T \otimes \mathbf{I}_J) \times_3 (\mathbf{I}_K \diamond \mathbf{U})^T \times_4 \mathbf{I}_K.$$

Utilizing the structure of both tensors $\mathcal{T}_{1,D}$ and \mathcal{T}_2 , the double contraction between these two tensors can be rewritten using the generalized unfoldings. According to the property of the generalized

unfoldings in equation (2.10), we get

$$\begin{aligned}
[\mathcal{X}']_{([1,2],3)} &= [\mathcal{T}_{1,D}]_{([1,4],[2,3])} \cdot [\mathcal{T}_2]_{([1,3],2)} \\
&= \underbrace{[\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{A}] \cdot [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}]_{([1,4],[2,3])} \cdot [(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{I}_J)]^T}_{*} \cdot [\mathbf{V}^T \diamond \mathbf{R}] \cdot \mathbf{B}^T
\end{aligned} \tag{4.3}$$

After reordering and consideration of the sparse structure, similar to equation (2.22), for the first part of equation (4.3) denoted by $*$, we have

$$\begin{aligned}
&[\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{A}] \cdot [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}]_{([1,4],[2,3])} \cdot [(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{I}_J)]^T = \\
&\left[[\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{I}_J] \cdot [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}]_{([1,4],[2,3])} \cdot [(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{A})]^T \right]^T = \\
&[\mathbf{I}_{JK} \diamond \mathbf{I}_{JK}]^T \cdot [(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{A})]^T = [(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{A})] [\mathbf{I}_{JK} \diamond \mathbf{I}_{JK}] = \\
&[(\mathbf{I}_K \diamond \mathbf{U})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{A})].
\end{aligned}$$

In Appendix B.2, we show the structure of the selection matrix $\mathbf{J}_{JK} = \mathbf{I}_{JK} \diamond \mathbf{I}_{JK}$. In Appendix B.2, we also show that this selection matrix equals $\mathbf{J}_{JK} = [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,J}]_{([2,3],[1,4])} \cdot [\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{I}_J]^T$. By substituting the last results into the tensor unfolding $[\mathcal{X}']_{([1,2],3)}$, we get

$$[\mathcal{X}']_{([1,2],3)} = \left[(\mathbf{I}_K \diamond \mathbf{U})^T \otimes \underbrace{(\mathbf{1}_K^T \otimes \mathbf{A})}_{**} \right] \cdot [\mathbf{V}^T \diamond \mathbf{R}] \cdot \mathbf{B}^T. \tag{4.4}$$

This tensor unfolding represents an unfolding of a constrained CP model that is degenerate in all modes $\mathcal{X}' = \mathcal{I}_{3,JK} \times_1 (\mathbf{1}_K^T \otimes \mathbf{A}) \times_2 (\mathbf{I}_K \diamond \mathbf{U})^T \times_3 \mathbf{B}(\mathbf{V}^T \diamond \mathbf{R})^T \in \mathbb{C}^{I \times K \times T}$. Moreover, using the properties of the Kronecker product (c.f. Section 2.1.2, equations (2.12)-(2.22)), for part of equation (4.4) denoted by $**$, we have

$$\mathbf{1}_K^T \otimes \mathbf{A} = \mathbf{1} \cdot \mathbf{1}_K^T \otimes \mathbf{A} \cdot \mathbf{I}_J = (\mathbf{1} \otimes \mathbf{A})(\mathbf{1}_K^T \otimes \mathbf{I}_J) = \mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_J).$$

Taking into account this property and that we can correct for the permuted dimensions \mathcal{X}' by simple permutation of the modes, the PARATUCK2 decomposition in equation (2.46) is equivalent to

$$\mathcal{X} = \mathcal{I}_{3,JK} \times_1 \mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_J) \times_2 \mathbf{B}(\mathbf{V}^T \diamond \mathbf{R})^T \times_3 (\mathbf{I}_K \diamond \mathbf{U})^T \in \mathbb{C}^{I \times T \times K}. \tag{4.5}$$

Derivation of the structure illustrated at the bottom of Fig. 4.1

Moreover, we show an alternative derivation of the PARATUCK2 data model. Thus, let us define $\mathcal{T}_3 = \mathcal{I}_{3,J} \times_1 \mathbf{A} \times_2 \mathbf{R}^T \times_3 \mathbf{U}^T \in \mathbb{C}^{I \times P \times K}$ and $\mathcal{T}_4 = \mathcal{I}_{3,P} \times_2 \mathbf{B} \times_3 \mathbf{V}^T \in \mathbb{C}^{P \times T \times K}$ as illustrated at the

bottom of Fig. 4.1. In this case, for the PARATUCK2 tensor, we get

$$\mathcal{X} = \mathcal{T}_3 \bullet_{2,3}^{1,3} \mathcal{T}_{4,D} \in \mathbb{C}^{I \times T \times K}, \quad (4.6)$$

where $[\mathcal{T}_{4,D}]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [(\mathbf{B} \diamond \mathbf{I}_P) \cdot \mathbf{V}]$. Similar to the first case, the tensor $\mathcal{T}_{4,D} \in \mathbb{C}^{P \times T \times K \times K}$ is modeled according to the BTD (see Appendix B.1).

$$\mathcal{T}_{4,D} = (\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,P}) \times_1 (\mathbf{1}_K^T \otimes \mathbf{I}_P) \times_2 (\mathbf{1}_K^T \otimes \mathbf{B}) \times_3 (\mathbf{I}_K \diamond \mathbf{V})^T \times_4 \mathbf{I}_K$$

From the structure of tensors \mathcal{T}_3 and $\mathcal{T}_{4,D}$, the unfolding of (4.6) leads to

$$\begin{aligned} [\mathcal{X}]_{(1,[2,3])} &= [\mathcal{T}_3]_{(1,[2,3])} [\mathcal{T}_{4,D}]_{([1,3],[2,4])} \\ &= \mathbf{A} \cdot [\mathbf{U}^T \diamond \mathbf{R}^T]^T \cdot [(\mathbf{I}_K \diamond \mathbf{V})^T \otimes (\mathbf{1}_K^T \otimes \mathbf{I}_P)] \cdot [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,P}]_{([1,3],[2,4])} \cdot \\ &\quad [\mathbf{I}_K \diamond (\mathbf{1}_K^T \otimes \mathbf{B})]^T. \end{aligned}$$

As in the derivation shown before (the derivation beginning from $\mathcal{X}' = \mathcal{T}_{1,D} \bullet_{2,3}^{1,3} \mathcal{T}_2$), we get

$$[\mathcal{X}]_{(1,[2,3])} = \mathbf{A} \cdot (\mathbf{U}^T \diamond \mathbf{R}^T)^T [(\mathbf{I}_K \diamond \mathbf{V})^T \diamond (\mathbf{1}_K^T \otimes \mathbf{B})]^T,$$

which represents an unfolding of a degenerate constrained CP model.

$$\mathcal{X} = \mathcal{I}_{3,PK} \times_1 \mathbf{A} \cdot (\mathbf{U}^T \diamond \mathbf{R}^T)^T \times_2 \mathbf{B} (\mathbf{1}_K^T \otimes \mathbf{I}_P) \times_3 (\mathbf{I}_K \diamond \mathbf{V})^T \in \mathbb{C}^{I \times T \times K} \quad (4.7)$$

Both equations (4.5) and (4.7) represent a constrained CP model of the PARATUCK2 decomposition with rank JK and PK , respectively. As previously mentioned, PARATUCK2 defines the mutual relationship between two different sets of latent components of rank J and P that vary along the K slices. There are obvious similarities and differences between the two versions of the derived constrained CP models in equations (4.5) and (4.7). The first version in equation (4.5) isolates the matrices \mathbf{A} and \mathbf{U} in separate modes. On the other hand, the second version in (4.7) isolates the matrices \mathbf{B} and \mathbf{V} in separate modes. The remaining modes themselves correspond to unfoldings of CP models of ranks P and J , for (4.5) and (4.7), respectively. The link between the PARATUCK2 decomposition and the CP decomposition is indicated also in [FdA14b], where it is shown that PARATUCK2 is equivalent to constrained CP with rank JP . Note that we have not exploit all possible combinations of representing PARATUCK2 based on generalized contractions. For instance, the development of $\mathcal{X} = \mathcal{T}_1 \bullet_{2,3}^{1,3} \mathcal{T}_{2,D} \in \mathbb{C}^{I \times T \times K}$ from equation (4.1) might lead to another constrained CP based representation of the PARATUCK2 model. The constrained CP decomposition is also called CONFAC (CONstrained FACTor) decomposition in [dAFM08], which is a special case of the PARALIND (PARAllel profiles with LINear Dependences) model [BHSL09], [SdA10]. The PARALIND model is equivalent to the PARATUCK2 decomposition if $\mathcal{D}_{(.,.,k)}^{(A)} = \mathbf{I}_J, \forall k = 1, \dots, K$ or $\mathbf{U} = \mathbf{1}_{J \times K}$. As shown in [SdA10], CONFAC/PARALIND models enjoy uniqueness (or partial uniqueness) under mild conditions, depending on their linear dependence

structure. Therefore, these new tensor models can be used to study the uniqueness properties of the PARATUCK2 decomposition based on the available results of general CP models (see Section 2.2.2). An extension of PARATUCK2 to more than three dimensions is possible following the above shown derivation.

Unlike CP, for the computation of the PARATUCK2 decomposition there exist almost no algorithms. The only ALS based algorithm is proposed in [Bro98]. This algorithm is based on the algorithm for the computation of DEDICOM [Kie93] as explained in Section 2.2.4. Another adaptation of the same algorithm is proposed in [BHS07] for the PARALIND decomposition. Recently, the authors in [CSH18] have proposed an extension of the ALS algorithm proposed in [Bro98] by introducing positivity constraints on all factor matrices. Exploiting equations (4.5) and (4.7), it is possible to derive an alternative ALS based algorithm for the PARATUCK2 decomposition. For instance, from equation (4.5) we can compute the following factor matrices

$$\begin{aligned} \mathbf{A}^T &= \left[\left[(\mathbf{I}_K \diamond \mathbf{U})^T \diamond \mathbf{B}(\mathbf{V}^T \diamond \mathbf{R})^T \right] (\mathbf{1}_K \otimes \mathbf{I}_J) \right]^+ [\mathcal{X}]_{([2,3],1)} \\ \mathbf{U}_{(:,k)} &= \left[\mathcal{P}_{1(.,.,k)}^T \diamond \mathbf{A} \right]^+ \text{vec}(\mathcal{X}_{(.,.,k)}), \quad \forall k = 1, \dots, K, \end{aligned}$$

where $\mathcal{P} = \mathcal{I}_{3,P \times 1} \mathbf{R} \times_2 \mathbf{B} \times_3 \mathbf{V}^T \in \mathbb{C}^{J \times T \times K}$. Moreover, the remaining matrices can then be computed from the unfoldings of \mathcal{P} assuming that the matrices \mathbf{A} and \mathbf{U} are known.

4.1.1 Summary

In this section, we use the alternative representation of the slice-wise multiplication via generalized tensor contraction proposed in Section 2.1.4. Based on this generalization, we derive two explicit tensor models of the PARATUCK2 decomposition. Both models represent constrained CP decompositions. These models capture the entire PARATUCK2 structure and can be used to further study the uniqueness properties of the PARATUCK2 decomposition. Moreover, these models can be used to develop new algorithms for the computation of the PARATUCK2 decomposition from noise corrupted observations.

4.2 PARAFAC2

The PARAFAC2 decomposition of a tensor $\mathcal{X}' \in \mathbb{R}^{I \times J \times K}$ is defined in a slice-wise fashion as depicted in Fig. 4.2. We have reviewed the PARAFAC2 decomposition, discussed its uniqueness properties and applications in Section 2.2.5.

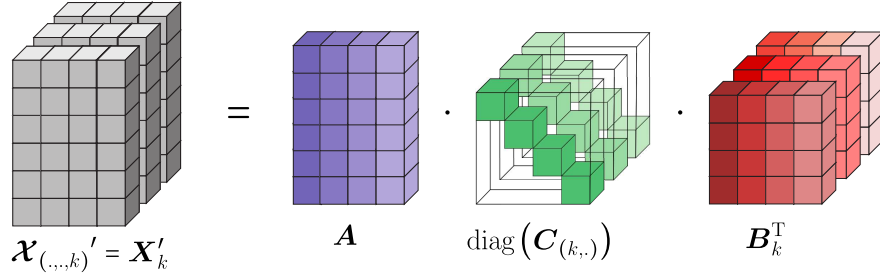


Figure 4.2.: Slice-wise visualization of the PARAFAC2 decomposition for a tensor $\mathcal{X}' \in \mathbb{R}^{I \times J \times K}$ with slices $\mathbf{X}'_k = \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k, \cdot)}) \cdot \mathbf{B}_k^T$, $\forall k = 1, \dots, K$.

PARAFAC2 via constrained CP model

Let us consider the PARAFAC2 decomposition of a tensor $\mathcal{X}' \in \mathbb{R}^{I \times J \times K}$ defined in Section 2.2.5 equation (2.47). First, we define

$$\mathcal{X}'_{(:, :, k)} = \mathbf{A} \cdot \mathbf{C}_{(:, :, k)} \cdot \mathbf{B}_{(:, :, k)} \in \mathbb{R}^{I \times J \times K}, \quad (4.8)$$

where $\mathcal{X}'_{(:, :, k)} = \mathbf{X}_k \in \mathbb{R}^{I \times J}$, $\mathbf{C} = \mathcal{I}_{3,R} \times_3 \mathbf{C} \in \mathbb{R}^{R \times R \times K}$ ($\mathbf{C} \in \mathbb{R}^{K \times R}$), and $\mathbf{B}_{(:, :, k)} = \mathbf{B}_k^T$ ($\mathbf{B} \in \mathbb{R}^{R \times J \times K}$, $\mathbf{B} \in \mathbb{R}^{R \times J}$). Using the Harshman constraint defined in Section 2.2.5 and defining $\mathcal{V}_{(:, :, k)} = \mathbf{V}_k$ ($\mathcal{V} \in \mathbb{R}^{R \times J \times K}$), we get

$$\begin{aligned} \mathbf{B} &= \mathcal{V} \times_1 \mathbf{F}^T, \\ \mathbf{B} \bullet_{2,3}^{2,3} \mathbf{B} &= \mathbf{K} \mathbf{F}^T \mathbf{F} \in \mathbb{R}^{R \times R}, \text{ and} \\ \mathcal{V} \bullet_{2,3}^{2,3} \mathcal{V} &= \mathbf{K} \mathbf{I}_R \in \mathbb{R}^{R \times R}. \end{aligned}$$

Using the generalized tensor contraction, we can rewrite the PARAFAC2 decomposition of the tensor \mathcal{X} (permuted version of the tensor \mathcal{X}') as

$$\mathcal{X} = (\mathcal{D}_C \times_1 \mathbf{A}) \bullet_{2,4}^{1,3} (\mathcal{V} \times_1 \mathbf{F}^T) \in \mathbb{R}^{I \times K \times J}, \quad (4.9)$$

where $\mathcal{D}_C \in \mathbb{R}^{R \times R \times K \times K}$ has only the following non-zero elements $\mathcal{D}_{C(r,r,k,k)} = \mathbf{C}_{(r,r,k)}$. According to Table 2.1 for the generalized unfolding of the diagonalized tensor, we have $[\mathcal{D}_C]_{([1,2,4],[3])} = \mathbf{I}_K \diamond [\mathbf{C}]_{([1,2],[3])}$. Note that the tensor \mathcal{X} has permuted dimensions as compared to (2.47) and (4.8) due to the definition of the contraction operator. The diagonalized tensor $\mathcal{D}_C \in \mathbb{C}^{R \times R \times K \times K}$ has also a BTD structure given by

$$\mathcal{D}_C = \sum_{k=1}^K (\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,R}) \times_3 ((\mathbf{e}_k \otimes \mathbf{C}_{(k, \cdot)})) \times_4 \mathbf{e}_k.$$

We derive the BTD structure of the diagonalized tensor in Appendix B.1, where we assume that the tensor to be diagonalized has a general CP structure. The result used in this section is easily deduced from the general result presented in Appendix B.1. Replacing the sum by a block diagonal

tensor and partitioned factor matrices, we get

$$\begin{aligned}\mathcal{D}_C = & \text{blkdiag}((\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,R}))_{k=1}^K \times_1 [\mathbf{I}_R, \dots, \mathbf{I}_R] \times_2 [\mathbf{I}_R, \dots, \mathbf{I}_R] \\ & \times_3 [(e_1 \otimes \mathbf{C}_{(1,\cdot)}), \dots, (e_K \otimes \mathbf{C}_{(K,\cdot)})] \times_4 \mathbf{I}_K\end{aligned}$$

Next, as shown in Appendix B.1, we can express the block diagonal structure of the core tensor and partitioned factor matrices by means of the Kronecker product. Hence, we have

$$\mathcal{D}_C = (\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,R}) \times_1 (\mathbf{1}_K^T \otimes \mathbf{I}_R) \times_2 (\mathbf{1}_K^T \otimes \mathbf{I}_R) \times_3 ((\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T).$$

Using equation (4.9), the property $[\mathcal{X}]_{([1,2],3)} = [\mathcal{D}_C \times_1 \mathbf{A}]_{([1,3],[2,4])} [\mathcal{B}]_{([1,3],2)}$, and the structure of the tensor \mathcal{D}_C , we get

$$\begin{aligned}[\mathcal{X}]_{([1,2],3)} = & \left[((\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T) \otimes (\mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_R)) \right] \cdot \\ & [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,R}]_{([1,3],[2,4])} \cdot (\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{I}_R)^T \cdot \\ & (\mathbf{I}_K \otimes \mathbf{F}^T) \cdot [\mathcal{V}]_{([1,3],2)}.\end{aligned}$$

Note that the matrix $[\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,R}]_{([1,3],[2,4])} \cdot (\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{I}_R)^T = \mathbf{I}_{RK} \diamond \mathbf{I}_{RK}$ is a selection matrix that converts the Kronecker product into a Khatri-Rao product (c.f. equation (2.28) and Appendix B.2). Using this property and the unfolding of a CP decomposition, we have

$$[\mathcal{X}]_{([1,2],3)} = \left[((\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T) \otimes (\mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_R)) \right] \cdot (\mathbf{I}_K \otimes \mathbf{F}^T) \cdot [\mathcal{V}]_{([1,3],2)}.$$

Hence, the tensor $\mathcal{X} \in \mathbb{C}^{I \times K \times T}$ can be expressed in the following CP format

$$\begin{aligned}\mathcal{X} = & \mathcal{I}_{3,RK} \times_1 \mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_R) \times_2 ((\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T) \times_3 ((\mathbf{I}_K \otimes \mathbf{F}^T) \cdot [\mathcal{V}]_{([1,3],2)})^T \\ = & \mathcal{I}_{3,RK} \times_1 \bar{\mathbf{A}} \times_2 \bar{\mathbf{C}} \times_3 \bar{\mathbf{B}},\end{aligned}\tag{4.10}$$

where $\bar{\mathbf{A}} = \mathbf{A}(\mathbf{1}_K^T \otimes \mathbf{I}_R) \in \mathbb{R}^{I \times RK}$, $\bar{\mathbf{C}} = (\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T \in \mathbb{R}^{K \times RK}$, and $\bar{\mathbf{B}} = [\mathcal{B}]_{(2,[1,3])} = ((\mathbf{I}_K \otimes \mathbf{F}^T) \cdot [\mathcal{V}]_{([1,3],2)})^T \in \mathbb{R}^{J \times RK}$. Equation (4.10) shows that PARAFAC2 is equivalent to a constrained CP decomposition [FdA14b], which is degenerate in all three modes. This decomposition is also referred to in the literature as the CONFAC decomposition [dAFM08]. It is a special case of the PARALIND model [BHSL09], [SdA10] as discussed in Section 4.1. The CONFAC model enjoys uniqueness (or partial uniqueness) under mild conditions depending on their linear dependence structure [SdA10].

Computation of the PARAFAC2 decomposition

The computation of the PARAFAC2 decomposition can be performed based on indirect fitting algorithms and direct fitting algorithms. The indirect fitting approach originally proposed in [Kie93]

tries to fit the cross product $\mathbf{X}_k \cdot \mathbf{X}_k^T$ instead of \mathbf{X}_k from equation (2.47). The cross product

$$\begin{aligned}\mathbf{X}_k \cdot \mathbf{X}_k^T &= \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{B}_k^T \cdot \mathbf{B}_k \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{A}^T \\ &= \mathbf{A} \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{H} \cdot \text{diag}(\mathbf{C}_{(k,.)}) \cdot \mathbf{A}^T\end{aligned}$$

with $\mathbf{H} = \mathbf{F}^T \mathbf{F} \in \mathbb{R}^{R \times R}$ is equivalent to a symmetric PARATUCK2 decomposition [HL96] that is also known as DEDICOM. Using a generalized tensor contraction, we have shown in Section 4.1 that the PARATUCK2 decomposition is also equivalent to a constrained CP. Similarities between the CP decomposition and the PARATUCK2 decomposition are also presented in [FdA14b]. On the other hand, the authors of [KTBB99] propose a direct fitting algorithm for the computation of the PARAFAC2 decomposition. This direct fitting approach is initialized with an initial guess of \mathbf{A} , \mathbf{C} , and \mathbf{F} and it consists of two loops. In the outer loop, using the solution of the OPP (Orthogonal Procrustes Problem) \mathbf{V} is estimated [Sch99]. Utilizing the orthogonality of the 3-mode slices of \mathbf{V} , we can convert the PARAFAC2 decomposition into a CP model. In the inner loop, we fit this CP model by estimating the factor matrices \mathbf{A} , \mathbf{C} , and \mathbf{F} based on ALS. Instead of ALS, for the inner loop we can use other methods for fitting the CP model such as line search or SMD [CH19]. The author of [Wei15] proposes a direct fitting algorithm for the computation of the PARAFAC2 decomposition, where the inner loop uses an SMD. In addition, the authors of [KTBB99] show that the direct fitting approach is more efficient than the indirect approach. Therefore, in this section we propose a single loop direct fitting algorithm for PARAFAC2 that has been derived via generalized tensor contractions [NCdAH18]. The proposed single loop algorithm requires fewer iterations than to the algorithms with two loops.

Assume that the matrices \mathbf{A} , \mathbf{C} , and \mathbf{F} are known. From the PARAFAC2 tensor model defined in equation (4.10) we can estimate \mathbf{V} , in an LS sense, as

$$[\mathbf{V}]_{([1,3],2)} = ((\bar{\mathbf{C}} \diamond \bar{\mathbf{A}}) \cdot (\mathbf{I}_K \otimes \mathbf{F}^T))^+ \cdot [\mathbf{X}]_{([1,2],3)}.$$

This LS estimate does not take into account the orthogonality constraints of the tensor \mathbf{V} . Therefore, it is not applicable in this case. However, it is applicable if the Harshman constraint is not considered. Alternatively, we can estimate the unfolding of the tensor \mathbf{B}

$$\bar{\mathbf{B}} = [\mathbf{B}]_{(2,[1,3])} = [\mathbf{X}]_{(3,[1,2])} \cdot ((\bar{\mathbf{C}} \diamond \bar{\mathbf{A}})^T)^+,$$

followed by an estimate of \mathbf{V} via a solution of the OPP using

$$[\mathbf{B}]_{(1,[2,3])} = \mathbf{F}^T \cdot [\mathbf{V}]_{(1,[2,3])}.$$

Simulation results, though, have shown that this approach is less accurate in the noisy case than solving directly the OPP. Therefore, we propose to estimate the tensor \mathbf{V} via OPP using directly

equation (4.9). Using the orthogonality properties of \mathbf{V} , it can be shown that

$$\tilde{\mathbf{X}} = \mathbf{X} \bullet_{2,3}^{2,3} \mathbf{D}_V = \mathbf{I}_{3,R} \times_1 \mathbf{A} \times_2 \mathbf{F}^T \times_3 \mathbf{C} \quad (4.11)$$

has a CP structure, where $\mathbf{D}_V \in \mathbb{R}^{R \times J \times K \times K}$ and $[\mathbf{D}_V]_{([1,2,4],[3])} = \mathbf{I}_K \diamond [\mathbf{V}]_{([1,2],[3])}$ according to Table 2.1. As shown in [Sch99] the best orthogonal estimate of the 3-mode slices of \mathbf{V} is provided from $\mathbf{Q} = \mathbf{X} \bullet_{1,3}^{1,3} \mathbf{D}_{\tilde{X}}$, where $[\mathbf{D}_{\tilde{X}}]_{([1,2,4],[3])} = \mathbf{I}_K \diamond [\tilde{\mathbf{X}}]_{([1,2],[3])}$. Then, \mathbf{V} is obtained from

$$\mathbf{V}_{(:,k)} = \left(\mathbf{Q}_{(:,k)}^T \cdot \mathbf{Q}_{(:,k)} \right)^{-\frac{1}{2}} \mathbf{Q}_{(:,k)}^T, \quad \forall k = 1, \dots, K.$$

Next, an estimate of the matrix \mathbf{A} is obtained from the unfolding $[\mathbf{X}]_{(1,[2,3])}$.

$$\mathbf{A} = [\mathbf{X}]_{(1,[2,3])} \cdot \left((\mathbf{1}_K^T \otimes \mathbf{I}_R) \cdot (\bar{\mathbf{B}} \diamond \bar{\mathbf{C}})^T \right)^+$$

Utilizing the orthogonality of \mathbf{V} , an estimate of \mathbf{F} follows from equation (4.11).

$$\mathbf{F} = [\tilde{\mathbf{X}}]_{(2,[1,3])} \cdot \left((\mathbf{C} \diamond \mathbf{A})^T \right)^+$$

The tensor \mathbf{B} is then computed by $\mathbf{B} = \mathbf{V} \times_1 \mathbf{F}^T$. Finally, \mathbf{C} is estimated based on the unfolding $[\mathbf{X}]_{(2,[3,1])}$ of equation (4.10)

$$[\mathbf{X}]_{(2,[3,1])} = \left((\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \text{vec}(\mathbf{C}^T)^T \right) \cdot (\bar{\mathbf{A}} \diamond \bar{\mathbf{B}})^T.$$

This unfolding represents also an unfolding of a 4-way CP tensor. Hence

$$\mathbf{X} = \mathbf{I}_{4,RK} \times_1 \bar{\mathbf{A}} \times_2 (\mathbf{I}_K \otimes \mathbf{1}_R^T) \times_3 \bar{\mathbf{B}} \times_4 \text{vec}(\mathbf{C}^T)^T,$$

which leads to

$$\text{vec}(\mathbf{C}^T) = (\bar{\mathbf{B}} \diamond (\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \bar{\mathbf{A}})^+ \cdot \text{vec}(\mathbf{X}).$$

Based on these estimates, we propose an ALS based direct fitting algorithm with only one loop for the computation of the PARAFAC2 decomposition. The proposed algorithm is initialized with a initial values of \mathbf{A} chosen based on the SVD of $[\mathbf{X}]_{(1,[2,3])}$, \mathbf{C} is chosen randomly, and \mathbf{F} as identity matrix. The algorithm is stopped if it exceeds the predefined maximum number of iterations, reaches a predefined minimum of the cost function $e_r = \|\hat{\mathbf{X}} - \mathbf{X}\|_H^2 / \|\mathbf{X}\|_H^2$, where $\hat{\mathbf{X}}$ is the reconstructed tensor after the decomposition, or the error e_r has not significantly changed within two consecutive iterations. The proposed algorithm that we denote by ‘‘P2-ALS’’ is summarized in Algorithm 4.1.

Algorithm 4.1: PARAFAC2 via ALS (“P2-ALS”)

Data: Given the initial matrices \mathbf{A} based on SVD, \mathbf{C} randomly chosen, and $\mathbf{F} = \mathbf{I}_R$.
while *does not exceed the maximum number of iterations, does not reach a predefined minimum, or the error of the cost function has not changed within two consecutive iterations* **do**
 Estimate \mathbf{V} by solving the OPP from $\mathcal{X} = (\mathcal{D}_C \times_1 \mathbf{A}) \bullet_{2,4}^{1,3} (\mathbf{V} \times_1 \mathbf{F}^T)$ as explained after equation (4.11).
 Update $\mathbf{A} = [\mathcal{X}]_{(1,[2,3])} \cdot ((\mathbf{1}_K^T \otimes \mathbf{I}_R) \cdot (\bar{\mathbf{B}} \diamond \bar{\mathbf{C}})^T)^+.$
 Utilizing the orthogonality of \mathbf{V} , update $\mathbf{F} = [\tilde{\mathcal{X}}]_{(2,[1,3])} \cdot ((\mathbf{C} \diamond \mathbf{A})^T)^+$, where $\tilde{\mathcal{X}} = \mathcal{X} \bullet_{2,3}^{2,3} \mathcal{D}_V = \mathcal{I}_{3,R} \times_1 \mathbf{A} \times_2 \mathbf{F}^T \times_3 \mathbf{C}$ and $[\mathcal{D}_V]_{([1,2,4],[3])} = \mathbf{I}_K \diamond [\mathbf{V}]_{([1,2],[3])}.$
 Compute $\mathbf{B} = \mathbf{V} \times_1 \mathbf{F}^T.$
 Update \mathbf{C} via
 $\text{vec}(\mathbf{C}^T) = (\bar{\mathbf{B}} \diamond (\mathbf{I}_K \otimes \mathbf{1}_R^T) \diamond \bar{\mathbf{A}})^+ \cdot \text{vec}(\mathcal{X}).$
end
Result: \mathbf{A} , \mathbf{C} , \mathbf{F} , and \mathbf{V}

4.2.1 Simulation Results

In this section, we provide comparisons between the direct fitting approaches for the computation of the PARAFAC2 decomposition based on simulation results. The single loop ALS based algorithm proposed in Section 4.2 and in [NCdAH18] is denoted as “P2-ALS”. The ALS algorithm [KTBB99] that consists of two ALS loops, an inner and an outer loop is denoted by “P2-ALS, two loops”. Finally, “P2-SMD” [Wei15] denotes the algorithm similar to the “P2-ALS, two loops” algorithm, but the second loop utilizes a single SMD similar to [RH08, RH13a]. A distinctive difference between the “P2-SMD” and the SECSI framework [RH13a] is that the “P2-SMD” solves only one SMD instead of all possible SMDs.

For the purpose of comparing the algorithms, “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD” we generate synthetic data according to equation (2.47). The elements of the matrices, \mathbf{A} , \mathbf{C} , and \mathbf{F} are randomly drawn from a zero mean Gaussian distribution with variance one. The tensor \mathbf{V} is also randomly generated, such that each slice has orthogonal rows. Next, we add zero mean additive white Gaussian noise with variance σ_N^2 to the generated signal tensor, $\mathcal{X} = \mathcal{X}_0 + \mathcal{N}$. The tensors \mathcal{X}_0 , \mathcal{N} , and \mathcal{X} represent the noiseless signal tensor, the noise tensor, and the noisy tensor respectively. Therefore, the instantaneous SNR equals $10 \log_{10} (\|\mathcal{X}_0\|_H^2 / \|\mathcal{N}\|_H^2).$

As an accuracy measure, we use the SRE (Squared Reconstruction Error)

$$\text{SRE} = \frac{\|\hat{\mathcal{X}} - \mathcal{X}_0\|_H^2}{\|\mathcal{X}_0\|_H^2}, \quad (4.12)$$

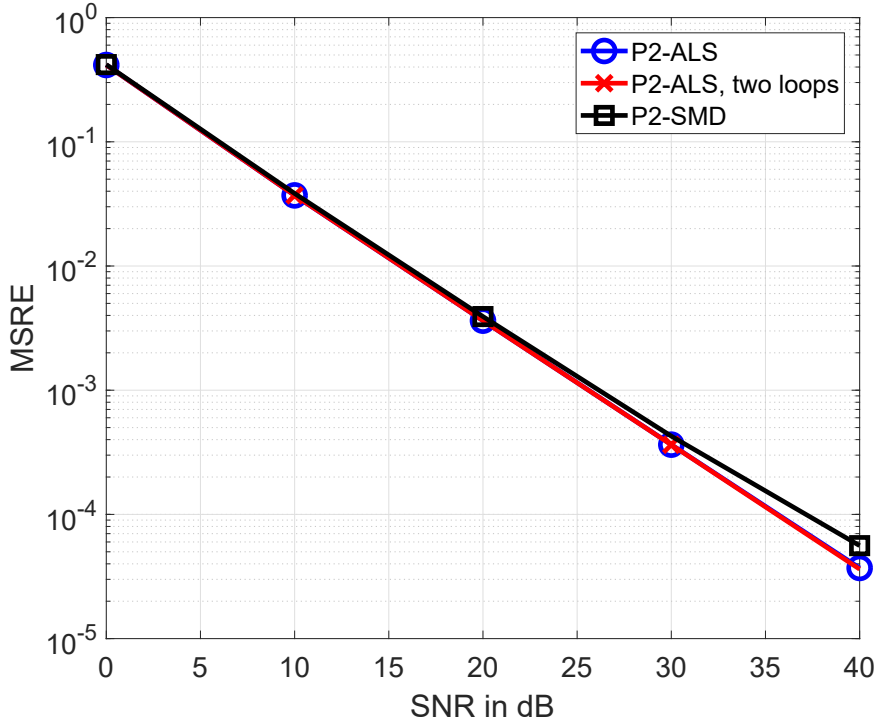


Figure 4.3.: MSRE as a function of the SNR = 0, ..., 40 dB for a real-valued tensor with dimensions $8 \times 10 \times 12$, and 3 components. The results are averaged over 2000 realizations. All algorithms are initialized with the same initial factor matrices.

where $\hat{\mathcal{X}}$ is the reconstructed tensor using the estimated factor matrices. All algorithms are initialized with the same initial factor matrices. The maximum number of iterations for all algorithms is set to 2000 iterations as discussed in the last section. Moreover, the maximum number of iterations for the inner loop, of the algorithm “P2-ALS, two loops” and “P2-SMD” is set to 5 and 50 iterations, respectively. The algorithms are stopped if they reach the maximum number of iterations, if they reach the minimum error of the cost function equal to 10^{-7} , or if the difference of the error between two consecutive iterations is smaller than 10^{-7} .

In Fig. 4.3, we illustrate the MSRE (Mean Squared Reconstruction Error) as a function of the SNR. Here, the MSRE is the SRE from equation (4.12) averaged over 2000 realizations. A real-valued tensor with dimensions $8 \times 10 \times 12$ and $R = 3$ components is generated. The accuracy of the three algorithms is very similar with the exception of the “P2-SMD” for high SNRs, where its accuracy is slightly lower as compared to “P2-ALS” and “P2-ALS, two loops”.

Moreover, in order to provide a more detailed comparison in Fig. 4.4, we depict the CCDF of the SRE for SNR = 20 dB. These results are the same 2000 realizations which average is presented in Fig. 4.3. Here, the CCDF depicts each of the 2000 SREs corresponding to the different realizations. The vertical lines represent the mean values for each curve. These mean values are identical with the mean SREs depicted in the Fig. 4.3 for SNR = 20 dB. Fig. 4.4 confirms the similar accuracy for the “P2-ALS” and “P2-ALS, two loops”. On the other side, the “P2-SMD” shows a slightly

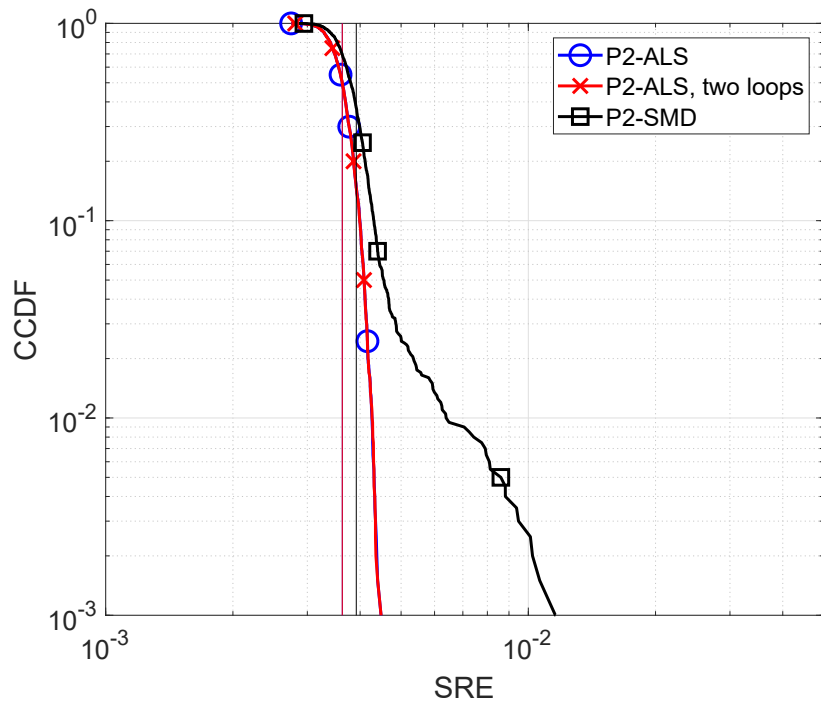


Figure 4.4.: CCDF of the SRE for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and SNR = 20 dB. All algorithms are initialized with the same initial factor matrices. The vertical lines correspond to the mean values also depicted in Fig. 4.3.

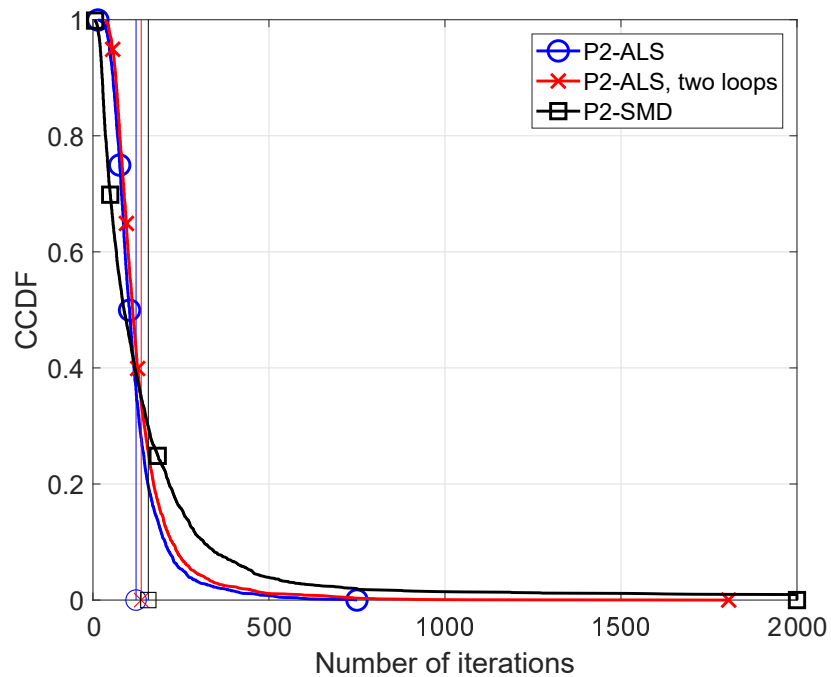


Figure 4.5.: CCDF of the number of iterations for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and SNR = 20 dB. All algorithms are initialized with the same initial factor matrices. The vertical lines represent the average number of iterations equal to 122, 137, and 157 for “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD”, respectively.

lower accuracy that is not obvious in Fig. 4.3. Also, the “P2-SMD” algorithm has more outliers than to the ALS based algorithms which is visible in Fig. 4.4. Additionally, in Fig. 4.5 we visualize the CCDF of the number of iterations corresponding to SREs from Fig. 4.4. Note that in this figure we include only the number of iterations required for the outer loop not the inner loop. The maximum number of iterations for the inner loop, for the algorithm “P2-ALS, two loops” and “P2-SMD” is set to 5 and 50 iterations, respectively. The vertical lines represent the average number of iterations, that is equal to 122, 137, and 157 for “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD”, respectively. According to Fig. 4.5, the “P2-ALS” algorithm requires fewer iterations than “P2-ALS” with two loops and “P2-SMD”.

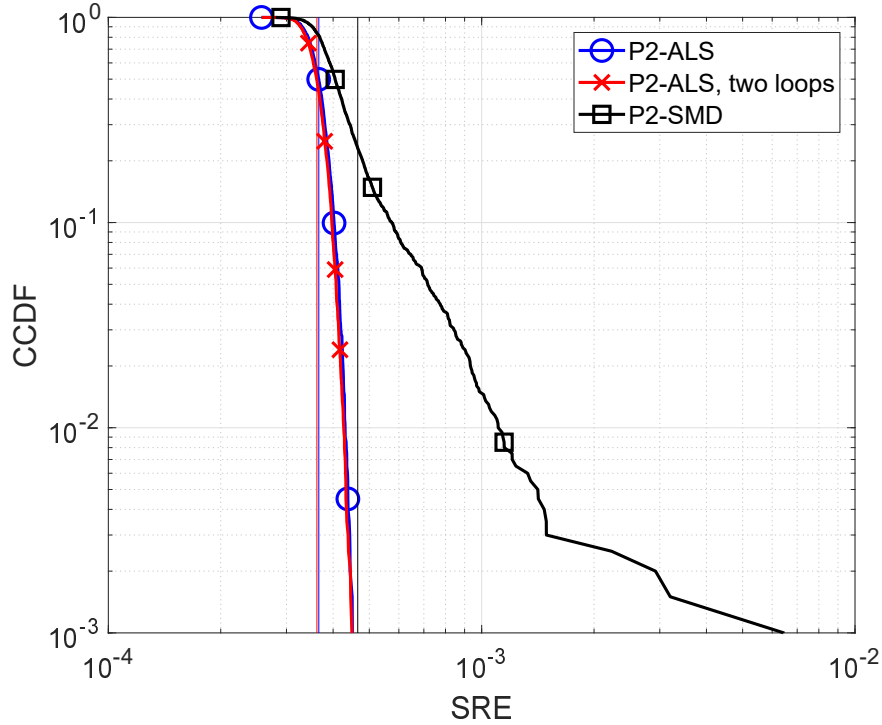


Figure 4.6.: CCDF of the SRE for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and $\text{SNR} = 30$ dB. The matrix \mathbf{C} has collinear columns with a correlation coefficient 0.8. All algorithms are initialized with the same initial matrices. The vertical lines represent the mean values.

Furthermore, in Figs. 4.6 and 4.7 we depict the CCDF of the SRE and the number of iterations, respectively, for a real-valued tensor with correlation. The decomposed tensor consists of $R = 3$ components and it has dimensions $8 \times 10 \times 12$. As previously mentioned, its factor matrices \mathbf{A} , \mathbf{C} , and \mathbf{F} are drawn from a zero mean Gaussian distribution with variance one. However, the matrix \mathbf{C} has correlated columns with a correlation coefficient of 0.8. We add correlation using a correlation matrix $\mathbf{R}(\rho_c) \in \mathbb{R}^{R \times R}$

$$\begin{aligned} \mathbf{C} &\leftarrow \mathbf{C} \cdot \mathbf{R}(\rho_c) \\ \mathbf{R}(\rho_c) &= (1 - \rho_c) \cdot \mathbf{I}_{R \times R} + \frac{\rho_c}{R} \cdot \mathbf{1}_{R \times R}, \end{aligned}$$

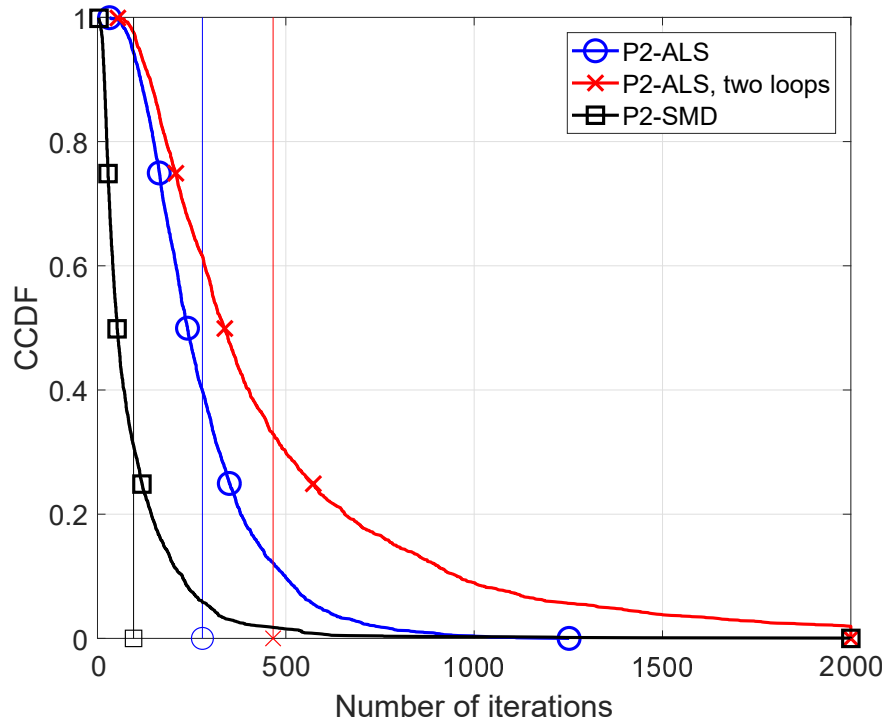


Figure 4.7.: CCDF of the number of iterations for a real-valued tensor with dimensions $8 \times 10 \times 12$, 3 components, and SNR = 30 dB. The matrix \mathbf{C} has mutually correlated columns with a correlation coefficient of 0.8. All algorithms are initialized with the same initial matrices. The vertical lines represent the average number of iterations equal to 278, 466, and 95 for “P2-ALS”, “P2-ALS, two loops”, and “P2-SMD”, respectively.

where ρ_c is the correlation factor corresponding to the matrix \mathbf{C} . The SREs presented in Fig. 4.6 correspond to an SNR of 30 dB and 2000 realizations. In Fig. 4.7, we depict the CCDF of the number of iterations for these 2000 realizations. As previously mentioned, all algorithms are initialized with the same initial matrices. For the “P2-ALS with two loops” and “P2-SMD” we only count the number of iterations of the outer loop, not of the inner one. This results in an average number of iterations equal to 278, 466, and 95 for “P2-ALS”, “P2-ALS with two loops”, and “P2-SMD”, respectively. The algorithms “P2-ALS” and “P2-ALS, two loops” have the same accuracy even though the “P2-ALS” requires fewer iterations. According to Fig. 4.7 the “P2-SMD” requires the lowest number of iterations for the outer loop, but the inner loop was set to a maximum number of 50 iterations and it has a lower accuracy.

To summarize, “P2-ALS” and “P2-ALS, two loops” have similar accuracy. The “P2-SMD” has a lower accuracy than the ALS based algorithms. The accuracy of the “P2-SMD” algorithm can be improved if all possible SMDs are taken into account as it was proposed in [RH13a] for the computation of the CP decomposition [CH19]. The “P2-ALS” requires fewer number of iterations than “P2-SMD” and “P2-ALS, two loops”.

4.2.2 Summary

In this section, we exploit a novel tensor representation derived using the generalized contraction between two tensors for a slice-wise (matrix) multiplication. This representation leads to a new tensor model that reveals the complete tensor structure of the PARAFAC2 tensor decomposition. We show that PARAFAC2 is equivalent to a constrained, degenerate CP model. Unitizing this model, we derive a direct fitting, single loop ALS algorithm (“P2-ALS”). This “P2-ALS” algorithm has the same accuracy, but requires fewer iterations than the state-of-art direct fitting ALS algorithm with two loops.

4.3 Conclusions

In this chapter, we consider the PARATUCK2 decomposition and the PARAFAC2 decomposition. For both decompositions, we first show that they represent a slice-wise multiplication between two tensors. We then exploit the novel representations of the slice-wise multiplication proposed in Section 2.1.4 based on the generalized tensor contraction. By substituting the individual structure of the tensors involved in the contraction, we derive new tensor models for both decompositions PARATUCK2 and PARAFAC2. For the PARATUCK2 decomposition we derive two alternatives of a constrained CP model that can be used to study the uniqueness of the PARATUCK2 decomposition and to develop new algorithms for its computation. The novel tensor representation leads also, to a constrained CP model of the PARAFAC2 decomposition that can be used to study its uniqueness properties. Moreover, we exploit this novel PARAFAC2 model to derive an efficient single loop ALS algorithm for the computation of the PARAFAC2 decomposition that requires fewer iteration than the state-of-the-art algorithms. This algorithm has already been published in [NCdAH18].

Chapter 5

Application of Tensor Algebra to Wireless Communication Systems

Traditionally, an OFDM system is described on a subcarrier-wise basis, i.e., using a slice-wise notation, if we consider that each tensor slice represents a different subcarrier. In this chapter, we exploit the generalized tensor contraction operator to model wireless communication systems. The generalized tensor contraction as introduced in Section 2.1.1 defines an inner product between two tensors with compatible dimensions. Here, we show that this tensor operator is useful to model multi-carrier MIMO communication systems. In Section 5.1.1, we show that the MIMO-OFDM received signal can be modeled by means of the tensor contraction operator. This tensor model is obtained without requiring additional spreading and provides a new, compact, and flexible formulation of a MIMO-OFDM system. Moreover, exploiting it at the receiver side facilitates the design of several types of receivers based on iterative LS or recursive LS [NHdA18]. In Section 5.1.2, we extend this novel model to MIMO-OFDM systems with Khatri-Rao coded symbols. In this application, the transmit signal tensor contains Khatri-Rao coded symbols that can be modeled using the CP decomposition. This new tensor based model facilitates the design of a receiver based on the LSKRF that jointly estimates the channel and the data symbols [NHdA17]. Moreover, in Section 5.1.3 we show that the spectral efficiency of the Khatri-Rao coded MIMO-OFDM can be increased by introducing "random coding" such that the "coding matrix" also contains random symbols. In Section 5.1.3, we derive a tensor model for the proposed randomly coded MIMO-OFDM using the generalized tensor contraction and two types of receivers. Furthermore, in Section 5.2 we show that our tensor model based on generalized contraction can also be extended to MIMO-GFDM systems. Based on this derived model, we design an iterative ALS receiver that simultaneously estimates the channel and the transmitted data in MIMO-GFDM systems.

5.1 Orthogonal Frequency Division Multiplexing (OFDM)

OFDM is the most widely used multi-carrier technique in current wireless communication systems. It is robust in multipath propagation environments and has a simple and efficient implementation [HYW⁺09], [FB11]. Using the FFT (Fast Fourier Transform), the complete frequency band is divided into smaller frequency subcarriers. The use of the cyclic prefix mitigates the ISI (Inter-Symbol Interference) and the ICI (Inter-Carrier Interference). Typically, the OFDM receiver is implemented in the frequency domain based on a ZF (Zero Forcing) filter. More advanced solu-

tions are proposed in [SFFM99]. Optimal training and channel estimation for OFDM systems are proposed in [BLM03], [HYSH06].

Tensor based signal processing offers an improved identifiability, uniqueness, and more efficient denoising compared to matrix based techniques. In [dAFX13], a MIMO multi-carrier system is modeled using tensor algebra and the PARATUCK2 tensor decomposition resulting in a novel space, time, and frequency coding structure. Similarly, in [dAF13b], trilinear coding in space, time, and frequency is proposed for MIMO-OFDM systems based on the CP tensor decomposition. By exploiting tensor models, semi-blind receivers have been introduced for multi-carrier communications systems in [FdA14a] and [LdCSdA13]. All these publications use additional spreading that leads to a significantly reduced spectral efficiency to create the tensor structure. Moreover, previous publications on tensor models for multi-carrier communications systems [dAFX13], [dAF13b], [LdCSdA13], and [FdA14a] do not exploit the channel correlation between the adjacent subcarriers. All these publications rely on the subcarrier-wise description of the MIMO-OFDM system. In this thesis, we present a tensor structure of the received signal that includes the frequency (subcarrier) mode.

In Section 5.1.1, we present the contraction between an uncoded signal tensor and a channel tensor for OFDM systems, yielding the same spectral efficiency as matrix based approaches (since no additional spreading is used) [NHdA18]. Since we do not use the Khatri-Rao coding structure as in [NHdA17], we do not get the corresponding CP structure for the transmit signal tensor. By exploiting this new tensor structure, we can reshape it into the factorization of a sum of Khatri-Rao products. This problem can be solved by means of iterative and recursive least squares originally proposed for blind source separation.

Moreover, in Section 5.1.2, an application of the generalized contraction operator to Khatri-Rao coded MIMO-OFDM systems is presented [NHdA17]. Due to the Khatri-Rao coding, we can use a CP model for the signal tensor. Khatri-Rao space-time coding was introduced in [SB02]. Later, it was extended in [dAF13a] to Khatri-Rao space-time-frequency coding. In contrast to the state of the art, we exploit the structure of the channel and the contraction properties using the transmit signal tensor and the known coding matrix to propose a receiver based on the LS-KRF in Section 5.1.2. Furthermore, we reduce the number of required pilot symbols by exploiting the correlation of the channel in the frequency domain. In Section 5.1.3, we propose "random coding" for MIMO-OFDM systems. We propose to keep the CP structure of the Khatri-Rao coded transmit signal [NHdA17], but the "coding matrix" contains random data symbols. Thus, the proposed randomly coded MIMO-OFDM system has higher spectral efficiency than Khatri-Rao coded MIMO-OFDM system. By exploiting the derived tensor structure of the received signal, we also, derive two types of receivers for randomly coded MIMO-OFDM systems in Section 5.1.3.

5.1.1 MIMO-OFDM via generalized tensor contraction

We assume a MIMO-OFDM system with M_T transmit and M_R receive antennas. One OFDM block consists of N samples, which equals the DFT (Discrete Fourier Transform) length, using the assumption that all N subcarriers are used for data transmission. If guard subcarriers are used, i.e., not all subcarriers are used for data transmission, the number of OFDM samples is smaller than the DFT length. All signals and equations used for the following derivation are in the frequency domain. Moreover, N is the number of subcarriers and K denotes the number of transmitted frames. The received signal in the frequency domain $\tilde{\mathbf{Y}} \in \mathbb{C}^{N \times M_R \times K}$ after the removal of the cyclic prefix is defined by means of the contraction operator

$$\tilde{\mathbf{Y}} = \tilde{\mathcal{H}} \bullet_{2,4}^{1,2} \tilde{\mathbf{S}} + \tilde{\mathcal{N}} = \tilde{\mathbf{Y}}_0 + \tilde{\mathcal{N}}. \quad (5.1)$$

We use \sim to distinguish the frequency domain from the time domain, i.e., $\tilde{\mathbf{Y}} = \mathbf{Y} \times_1 \mathbf{F}_N$, where $\mathbf{F}_N \in \mathbb{C}^{N \times N}$ is the DFT matrix (defined in Appendix A.2 equation (A.5)) and \mathbf{Y} is the received signal in time domain. The transmit signal tensor is denoted as $\tilde{\mathbf{S}} \in \mathbb{C}^{N \times M_T \times K}$ and $\tilde{\mathcal{N}} \in \mathbb{C}^{N \times M_R \times K}$ represents the additive white Gaussian noise in the frequency domain. The tensor $\tilde{\mathbf{Y}}_0 \in \mathbb{C}^{N \times M_R \times K}$ represents the noiseless received signal in frequency domain after the removal of the cyclic prefix. The frequency-selective propagation channel is represented by a channel tensor $\tilde{\mathcal{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ as we propose in [NHdA17] and can be explained as follows.

Channel tensor

A frequency-selective channel has an impulse response $\mathbf{h}_L^{(m_R, m_T)} \in \mathbb{C}^{L \times 1}$, for each receive-transmit antenna pair, (m_R, m_T) , and a maximum of L taps. After the removal of the cyclic prefix, the channel matrix in the frequency domain is a diagonal matrix for each receive-transmit antenna pair, $\tilde{\mathbf{H}}^{(m_R, m_T)} = \text{diag}(\mathbf{F}_{N \times L} \cdot \mathbf{h}_L^{(m_R, m_T)}) \in \mathbb{C}^{N \times N}$ [BLM03, HYSH06]. Here, the matrix $\mathbf{F}_{N \times L} \in \mathbb{C}^{N \times L}$ contains the first L columns of the DFT matrix of size $N \times N$ (see also Appendix A.2 equation (A.5)). Collecting all of the channel matrices in a 4-way channel tensor $\tilde{\mathcal{H}}$, we get

$$\tilde{\mathcal{H}}_{(.,., m_R, m_T)} = \text{diag}(\mathbf{F}_{N \times L} \cdot \mathbf{h}_L^{(m_R, m_T)}) = \text{diag}(\tilde{\mathbf{h}}^{(m_R, m_T)}). \quad (5.2)$$

For each receive-transmit antenna pair the channel transfer matrix is a diagonal matrix that is represented by the corresponding slice of the tensor $\tilde{\mathcal{H}}$ as shown in (5.2). The vector $\tilde{\mathbf{h}}^{(m_R, m_T)} \in \mathbb{C}^{N \times 1}$ contains the frequency domain channel coefficients. An example of a MIMO system with $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas and the corresponding channel vectors are depicted in Fig. 5.1. We assume that the channel stays constant during the K frames. Note that only in case of cyclic prefix OFDM the channel tensor in the frequency domain contains diagonal matrices for each receive-transmit antenna pair. In a general multi-carrier system, the frequency domain channel matrix is not necessarily diagonal. However, equation (5.1) is still satisfied which means our model remains valid for non-orthogonal multi-carrier systems.

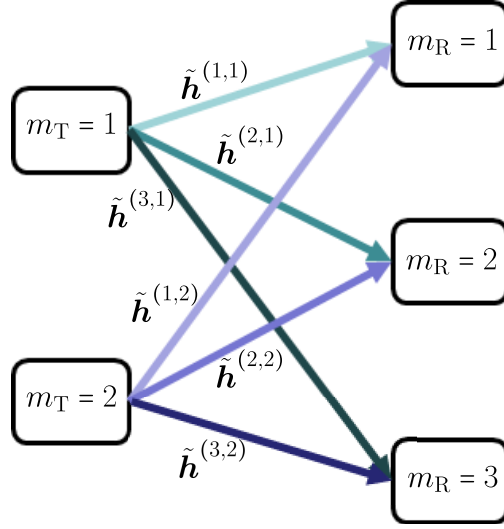


Figure 5.1.: A MIMO system with $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas.

In equation (5.2), we have defined the channel tensor. However, we have not revealed the explicit tensor structure, yet. In order to do so, let us first assume that all channel transfer matrices for the m_T -th transmit and all receive antennas are collected in a diagonal tensor, $\tilde{\mathcal{H}}_R^{(m_T)} \in \mathbb{C}^{N \times N \times M_R}$.

$$\tilde{\mathcal{H}}_{R(\dots, m_R)}^{(m_T)} = \text{diag}\left(\tilde{\mathbf{h}}^{(m_R, m_T)}\right) \quad m_R = 1, \dots, M_R, m_T = 1, \dots, M_T \quad (5.3)$$

Based on this diagonal structure, the tensor $\tilde{\mathcal{H}}_R^{(m_T)}$ has the following CP decomposition

$$\tilde{\mathcal{H}}_R^{(m_T)} = \mathcal{I}_{3, N \times 1} \mathbf{I}_N \times_2 \mathbf{I}_N \times_3 \tilde{\mathbf{H}}_R^{(m_T)}, \quad (5.4)$$

where $\tilde{\mathbf{H}}_R^{(m_T)} = \begin{bmatrix} \tilde{\mathbf{h}}^{(1, m_T)} & \tilde{\mathbf{h}}^{(2, m_T)} & \dots & \tilde{\mathbf{h}}^{(M_R, m_T)} \end{bmatrix}^T \in \mathbb{C}^{M_R \times N}$.

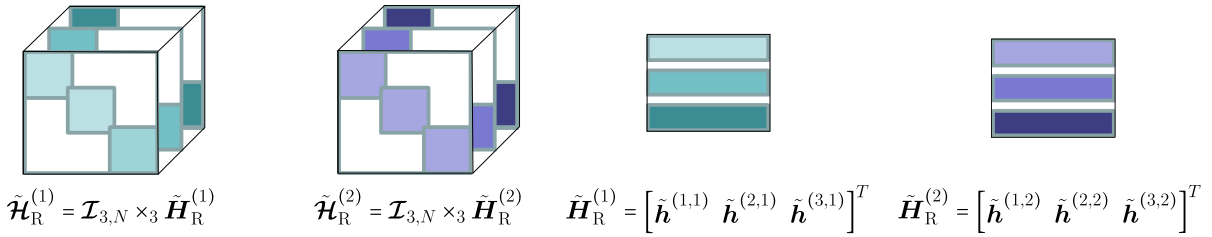


Figure 5.2.: Visualization of the channel tensors $\tilde{\mathcal{H}}_R^{(1)}$ and $\tilde{\mathcal{H}}_R^{(2)}$ as well as the channel matrices $\tilde{\mathbf{H}}_R^{(1)}$ and $\tilde{\mathbf{H}}_R^{(2)}$ for a MIMO system with $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas. The MIMO system is depicted in Fig. 5.1.

In Fig. 5.2, we depict the structure of the channel tensors $\tilde{\mathcal{H}}_R^{(1)}$ and $\tilde{\mathcal{H}}_R^{(2)}$ as well as the channel matrices $\tilde{\mathbf{H}}_R^{(1)}$ and $\tilde{\mathbf{H}}_R^{(2)}$ for the MIMO system given in Fig. 5.1. This exemplary MIMO system has $M_T = 2$ transmit antennas and $M_R = 3$ receive antennas. The complete 4-way channel tensor, defined in equation (5.2) can be obtained by concatenating the $\tilde{\mathcal{H}}_R^{(m_T)}$ tensors along the fourth

dimension (the concatenation between two tensors is defined in Fig. 2.6). Hence, the 4-way channel tensor $\tilde{\mathcal{H}}$ can be expressed as

$$\begin{aligned}\tilde{\mathcal{H}} &= \left[\tilde{\mathcal{H}}_{\text{R}}^{(1)} \sqcup_4 \tilde{\mathcal{H}}_{\text{R}}^{(2)} \sqcup_4 \dots \sqcup_4 \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \right] \\ &= \left[\tilde{\mathcal{H}}_{\text{R}}^{(1)} \sqcup_4 \tilde{\mathcal{H}}_{\text{R}}^{(2)} \sqcup_4 \dots \sqcup_4 \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \right] \times_4 \mathbf{I}_{M_{\text{T}}} \\ &= \sum_{m_{\text{T}}=1}^{M_{\text{T}}} \tilde{\mathcal{H}}_{\text{R}}^{(m_{\text{T}})} \circ \mathbf{e}_{m_{\text{T}}} = \sum_{m_{\text{T}}=1}^{M_{\text{T}}} \mathcal{D} \times_1 \mathbf{I}_N \times_2 \mathbf{I}_N \times_3 \tilde{\mathbf{H}}_{\text{R}}^{(m_{\text{T}})} \times_4 \mathbf{e}_{m_{\text{T}}}.\end{aligned}\quad (5.5)$$

Note that $\tilde{\mathcal{H}}$ satisfies a very special BTD, where $\mathcal{D}_{(\dots,1)} = \mathbf{I}_{3,N} \in \mathbb{R}^{N \times N \times N \times 1}$ ($\mathcal{D} = \mathbf{I}_{4,1} \otimes \mathbf{I}_{3,N}$) and $\mathbf{e}_{m_{\text{T}}} \in \mathbb{R}^{M_{\text{T}} \times 1}$ is a pinning vector. We prove the BTD structure of the channel tensor $\tilde{\mathcal{H}}$ in Appendix B.3. In Appendix B.3, we also show that the $([1,3],[2,4])$ generalized unfolding of the channel tensor can be expressed as

$$[\tilde{\mathcal{H}}]_{([1,3],[2,4])} = \tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N) \in \mathbb{C}^{NM_{\text{R}} \times NM_{\text{T}}}, \quad (5.6)$$

where $\tilde{\mathbf{H}} \in \mathbb{C}^{M_{\text{R}} \times NM_{\text{T}}}$ is a matrix containing all non-zero elements of the tensor $\tilde{\mathcal{H}}$ and it is defined as,

$$\tilde{\mathbf{H}} = \left[\begin{array}{c|c|c|c} \tilde{\mathbf{h}}^{(1,1)\text{T}} & \tilde{\mathbf{h}}^{(1,2)\text{T}} & \dots & \tilde{\mathbf{h}}^{(1,M_{\text{T}})\text{T}} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{h}}^{(M_{\text{R}},1)\text{T}} & \tilde{\mathbf{h}}^{(M_{\text{R}},2)\text{T}} & \dots & \tilde{\mathbf{h}}^{(M_{\text{R}},M_{\text{T}})\text{T}} \end{array} \right] = \left[\tilde{\mathbf{H}}_{\text{R}}^{(1)} \quad \tilde{\mathbf{H}}_{\text{R}}^{(2)} \quad \dots \quad \tilde{\mathbf{H}}_{\text{R}}^{(M_{\text{T}})} \right] \in \mathbb{C}^{M_{\text{R}} \times NM_{\text{T}}}. \quad (5.7)$$

Fig. 5.3 depicts the structure of the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$ for a MIMO-OFDM system with parameters $M_{\text{T}} = 2$, $M_{\text{R}} = 3$, and $N = 3$.

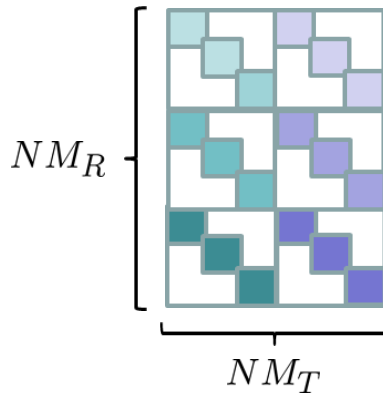


Figure 5.3.: Visualization of the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$ for a MIMO-OFDM system with parameters $M_{\text{T}} = 2$, $M_{\text{R}} = 3$, and $N = 3$.

Data transmission

The signal tensor $\tilde{\mathbf{S}}$ in equation (5.1) contains all symbols in the frequency domain that are transmitted on N subcarriers, M_T transmit antennas, and K frames. For notational simplicity, we define the following block matrix $\tilde{\mathbf{S}}$ as the transpose of the 3-mode unfolding of $\tilde{\mathbf{S}}$

$$\tilde{\mathbf{S}} = \tilde{\mathbf{S}}_{([1,2],[3])}^T = \begin{bmatrix} \tilde{\mathbf{S}}^{(1)} & \tilde{\mathbf{S}}^{(2)} & \dots & \tilde{\mathbf{S}}^{(M_T)} \end{bmatrix} \in \mathbb{C}^{K \times NM_T}, \quad (5.8)$$

where $\tilde{\mathbf{S}}^{(m_T)} \in \mathbb{C}^{K \times N}$ contains the symbols transmitted via the m_T -th antenna.

Moreover, we assume that the symbol matrix consists of data and pilot symbols, $\tilde{\mathbf{S}} = \tilde{\mathbf{S}}_d + \tilde{\mathbf{S}}_p$. The matrices $\tilde{\mathbf{S}}_d$ and $\tilde{\mathbf{S}}_p$ represent the data symbols and the pilot symbols, respectively. The matrix $\tilde{\mathbf{S}}_d$ contains zeros at the positions of the pilot symbols. Accordingly, the matrix $\tilde{\mathbf{S}}_p$ contains non-zero elements only at the pilot positions. Typically, there are three ways of arranging the pilot symbol within the OFDM blocks (block, comb, and lattice-type) [HYW⁺09]. We assume a comb-type arrangement of the pilot symbols with equidistant positions in the time and the frequency domain, for each antenna. The spacing in the time domain is denoted by ΔK . Moreover, we send pilots only with subcarrier spacing of ΔF between two pilot symbols. In Fig. 5.4, we show the positions of the pilot symbols and the data symbols for the first antenna, where $M_T = 2$, $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$. Furthermore, there are positions where neither pilot symbols or data symbols are allowed to be transmitted. These positions are marked in black color in Fig. 5.4, and are reserved for the pilot symbols corresponding to the remaining antennas. In Fig. 5.5, we illustrate the distribution of the pilot symbols for $M_T = 2$ transmit antennas with respect to the subcarriers, for $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$. This results in $M_T \lfloor \frac{N}{\Delta F} \rfloor$ pilot symbols per frame. In comparison, other publications such as [dAFX13], [dAF13b], [LdCSdA13], and [FdA14a] use NM_T pilot symbols per frame. By exploiting the channel correlation among adjacent subcarriers, a reduced number of pilot symbols can be used for channel estimation.

Receiver design

Using the property of the generalized unfoldings in equation (2.10), the received signal in equation (5.1) becomes

$$[\tilde{\mathbf{Y}}]_{([1,2],[3])} = [\tilde{\mathbf{H}}]_{([1,3],[2,4])} \tilde{\mathbf{S}}_{([1,2],[3])} + [\tilde{\mathbf{N}}]_{([1,2],[3])} \in \mathbb{C}^{NM_R \times K}. \quad (5.9)$$

Next, by substituting the corresponding tensor unfoldings in equation (5.9), we get

$$[\tilde{\mathbf{Y}}]_{([1,2],[3])} = (\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)) \cdot \tilde{\mathbf{S}}^T + [\tilde{\mathbf{N}}]_{([1,2],[3])}. \quad (5.10)$$

The above equation satisfies an unfolding of a noisy observation of a low-rank tensor with a CP structure. By applying an inverse unfolding for the received signal in the frequency domain after

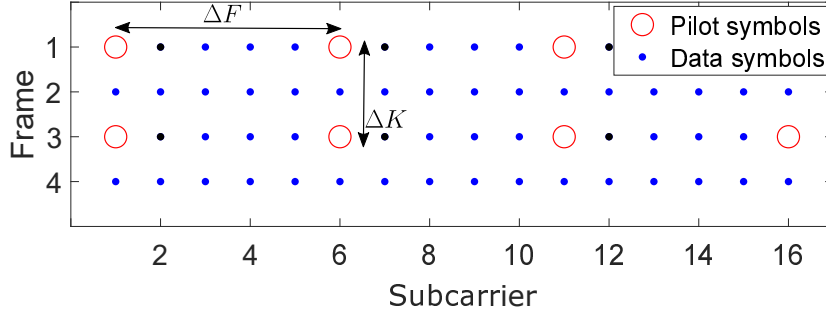


Figure 5.4.: Visualization of the arrangement of the pilot symbols and the data symbols for the first antenna with respect to the frame number and the subcarrier number for a MIMO-OFDM system with parameters $M_T = 2$, $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$.

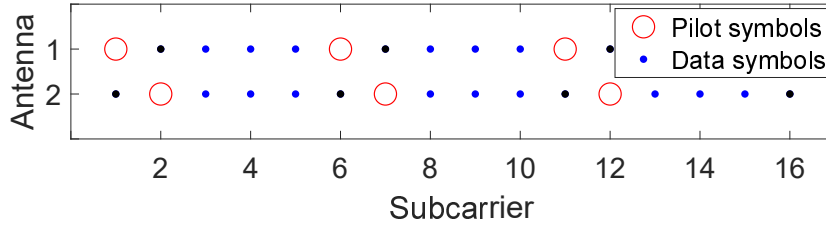


Figure 5.5.: Visualization of the arrangement of the pilot symbols and the data symbols in the first frame with respect to the antenna number and the subcarrier number for a MIMO-OFDM system with parameters $M_T = 2$, $K = 4$, $F = 16$, $\Delta K = 2$, and $\Delta F = 5$.

the removal of the cyclic prefix, we get

$$\tilde{\mathbf{Y}} = \mathbf{I}_{3, NM_T} \times_1 (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \times_2 \tilde{\mathbf{H}} \times_3 \tilde{\mathbf{S}} + \tilde{\mathbf{N}} \in \mathbb{C}^{N \times M_R \times K}. \quad (5.11)$$

Our goal is to jointly estimate the channel and the symbols, i.e., $\tilde{\mathbf{H}}$ and $\tilde{\mathbf{S}}$ in equation (5.11). Note that all factor matrices are flat resulting in a degenerate CP model in all three modes. Therefore, it is difficult to estimate the channel and the symbols by simply fitting a CP model of the received signal tensor in (5.11).

Using the prior knowledge of the pilot symbols and their positions, the channel in the frequency domain can be estimated. Naturally, the channel is estimated only at those subcarrier positions where the pilot symbols are located. Afterwards, an interpolation is applied to get the complete channel estimate. Moreover, as shown in [BLM03, HYSH06] the channel can be first estimated in the time domain and then transformed into the frequency domain. Either way, this leads to a pilot based channel estimate that we denote as $\hat{\tilde{\mathbf{H}}}_p$, or $\tilde{\mathcal{H}}_p$ ¹. The pilot based channel estimate is then used to estimate the data symbols. In the remainder of this section, we discuss different ways to estimate the symbols. We use the pilot based channel estimate to initialize the proposed algorithms.

Traditionally, the estimate of the symbols is obtained in the frequency domain with a ZF receiver. In this case, the symbols are calculated by inverting the channel matrix for each subcarrier indi-

¹In our simulations, we use the pilot based channel estimate obtained in the time domain.

vidually. This ZF receiver using the above defined tensor notation is summarized in Algorithm 5.1.

Algorithm 5.1: ZF receiver

```

initialization  $\tilde{\mathcal{H}}_p$ 
for  $n = 1 : N$  do
     $\hat{\mathcal{S}}_{(n,...)} \approx \hat{\mathcal{H}}_{p(n,n,...)}^+ \tilde{\mathcal{Y}}_{(n,...)}$ 
end
Result:  $\hat{\mathcal{S}}$ 

```

Alternatively, if we compute the 1-mode unfolding of the tensor $\tilde{\mathcal{Y}}$ in equation (5.11), we get

$$[\tilde{\mathcal{Y}}]_{([1],[2,3])} = (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \cdot (\tilde{\mathcal{S}} \diamond \tilde{\mathcal{H}})^T + [\tilde{\mathcal{N}}]_{([1],[3,2])} \in \mathbb{C}^{N \times M_R K}.$$

Taking into account the structure of the matrices $(\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \in \mathbb{R}^{N \times N M_T}$, $\tilde{\mathcal{H}}$ in (5.7), and $\tilde{\mathcal{S}}$ in (5.8), the above unfolding becomes

$$[\tilde{\mathcal{Y}}]_{([1],[2,3])} = \sum_{m_T=1}^{M_T} \left(\tilde{\mathcal{S}}^{(m_T)} \diamond \tilde{\mathcal{H}}_R^{(m_T)} \right)^T + [\tilde{\mathcal{N}}]_{([1],[3,2])}.$$

After transposition and omitting the noise term, we get

$$[\tilde{\mathcal{Y}}]_{([2,3],[1])} \approx \sum_{m_T=1}^{M_T} \left(\tilde{\mathcal{S}}^{(m_T)} \diamond \tilde{\mathcal{H}}_R^{(m_T)} \right) \in \mathbb{C}^{M_R K \times N}.$$

This sum of Khatri-Rao products can be resolved in a column-wise fashion. Let $\tilde{\mathbf{y}}_n \in \mathbb{C}^{M_R K \times 1}$ denote the n -th column of $[\tilde{\mathcal{Y}}]_{([2,3],[1])} \in \mathbb{C}^{M_R K \times N}$. After reshaping this vector into matrix $\tilde{\mathbf{Y}}_n \in \mathbb{C}^{M_R \times K}$, such that $\tilde{\mathbf{y}}_n = \text{vec}(\tilde{\mathbf{Y}}_n)$, it is easy to see that this matrix satisfies

$$\tilde{\mathbf{Y}}_n \approx \tilde{\mathcal{H}}_n \cdot \tilde{\mathcal{S}}_n, \quad (5.12)$$

where $\tilde{\mathcal{H}}_n$ and $\tilde{\mathcal{S}}_n$ are the n -th slices of $\tilde{\mathcal{H}}_{(n,n,...)} \in \mathbb{C}^{M_R \times M_T}$ and $\tilde{\mathcal{S}}_{(n,...)} \in \mathbb{C}^{M_T \times K}$, respectively. Note that $\tilde{\mathbf{Y}}_n$ is the n -th slice of $\tilde{\mathcal{Y}}_{(n,...)}$. Using the pseudo-inverse of the channel, we get the traditional ZF receiver as summarized in Algorithm 5.1. This shows that our model in (5.11) is a very general model based on which we can derive even the traditional receivers.

Moreover, the channel and the symbols on each subcarrier can be estimated by means of iterative or recursive LS algorithms. Similar algorithms were proposed in [TVP94] and [TVP96] for blind source separation on a single subcarrier. We have extended the four algorithms presented in [TVP96] to our application.

The algorithms ILSP (Iterative Least-Squares with Projection) and ILSE (Iterative Least-Squares with Enumeration) summarized in Algorithm 5.2 and Algorithm 5.3, respectively, are iterative algorithms based on LS. Both algorithms are initialized with the pilot based channel estimate,

Algorithm 5.2: Iterative Least-Squares with Projection (ILSP)

```

initialization  $\tilde{\mathcal{H}}_p$ , maxIteration, minErr
for  $n = 1 : N$  do
    set  $i = 1$ ,  $e = \infty$ 
    while  $i < \text{maxIteration}$  or  $e < \text{minErr}$  do
         $\tilde{\mathbf{S}}_n^{(i)} = (\tilde{\mathbf{H}}_n^{(i-1)\text{H}} \tilde{\mathbf{H}}_n^{(i-1)})^{-1} \tilde{\mathbf{H}}_n^{(i-1)\text{H}} \tilde{\mathbf{Y}}_n$ 
         $\tilde{\mathbf{S}}_n^{(i)} = \text{proj}(\tilde{\mathbf{S}}_n^{(i)})$ 
        if  $\text{rank}(\tilde{\mathbf{S}}_n^{(i)}) = M_T$  then
             $\tilde{\mathbf{H}}_n^{(i)} = \tilde{\mathbf{Y}}_n \tilde{\mathbf{S}}_n^{(i)\text{H}} (\tilde{\mathbf{S}}_n^{(i)} \tilde{\mathbf{S}}_n^{(i)\text{H}})^{-1}$ 
        else
             $\tilde{\mathbf{H}}_n^{(i)} = \tilde{\mathbf{H}}_n^{(i-1)}$ 
        end
         $i = i + 1$ ,  $e = \|\tilde{\mathbf{H}}_n^{(i-1)} - \tilde{\mathbf{H}}_n^{(i)}\|_F^2$ 
    end
end
Result:  $\tilde{\mathcal{S}}$  and  $\tilde{\mathcal{H}}$ 

```

Algorithm 5.3: Iterative Least-Squares with Enumeration (ILSE)

```

initialization  $\tilde{\mathcal{H}}_p$ , maxIteration, minErr
for  $n = 1 : N$  do
    set  $i = 1$ ,  $e = \infty$ 
    while  $i < \text{maxIteration}$  or  $e < \text{minErr}$  do
        for  $k = 1 : K$  do
             $\hat{\mathbf{s}} = \arg \min_{\mathbf{s}^{(j)} \in \Omega} \|\tilde{\mathbf{Y}}_{n(.,k)} - \tilde{\mathbf{H}}_n^{(i-1)} \mathbf{s}^{(j)}\|$ ,  $\forall j = 1, \dots, M_o^{M_T}$ 
             $\tilde{\mathbf{S}}_{n(.,k)}^{(i)} = \hat{\mathbf{s}}$ 
        end
        if  $\text{rank}(\tilde{\mathbf{S}}_n^{(i)}) = M_T$  then
             $\tilde{\mathbf{H}}_n^{(i)} = \tilde{\mathbf{Y}}_n \tilde{\mathbf{S}}_n^{(i)\text{H}} (\tilde{\mathbf{S}}_n^{(i)} \tilde{\mathbf{S}}_n^{(i)\text{H}})^{-1}$ 
        else
             $\tilde{\mathbf{H}}_n^{(i)} = \tilde{\mathbf{H}}_n^{(i-1)}$ 
        end
         $i = i + 1$ ,  $e = \|\tilde{\mathbf{H}}_n^{(i-1)} - \tilde{\mathbf{H}}_n^{(i)}\|_F^2$ 
    end
end
Result:  $\tilde{\mathcal{S}}$  and  $\tilde{\mathcal{H}}$ 

```

the maximum number of iterations (maxIteration), and the minimum error difference between two consecutive updates (minErr). The ILSP algorithm is essentially an iterative version of the ZF algorithm, where in each iteration the estimated symbols are projected onto the finite alphabet Ω of the transmitted symbols. This finite alphabet depends on the modulation type and the modulation

order M_o . Details regarding the convergence for different finite alphabets are discussed in [TVP96]. To estimate the symbols, we compute a pseudo-inverse of the channel which leads to the condition $M_R \geq M_T$. On the other hand, the ILSE algorithm does not require this condition as it estimates the symbols based on enumeration. Equation (5.13) represents the enumeration or the exhaustive search over the final alphabet of symbols Ω .

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}^{(j)} \in \Omega} \|\tilde{\mathbf{Y}}_{n(:,k)} - \tilde{\mathbf{H}}_n \mathbf{s}^{(j)}\|, \quad j = 1, \dots, M_o^{M_T} \quad (5.13)$$

Both algorithms update the channel only if it is possible, i.e., if the rank of the symbol matrix $\tilde{\mathbf{S}}_n \in \mathbb{C}^{M_T \times K}$ is M_T , $K \geq M_T$. Note that this is not possible for all values of M_T , K , and for all patterns of random data symbols from a finite distribution.

Algorithm 5.4: Recursive Least-Squares with Projection (RLSP)

```

initialization  $\tilde{\mathbf{H}}_p, 0 \leq \alpha \leq 1$ 
for  $n = 1 : N$  do
     $\tilde{\tilde{\mathbf{S}}}_n = (\tilde{\mathbf{H}}_n^H \tilde{\mathbf{H}}_n)^{-1} \tilde{\mathbf{H}}_n^H \tilde{\mathbf{Y}}_n$ 
     $\tilde{\mathbf{S}}_n = \text{proj}(\tilde{\tilde{\mathbf{S}}}_n)$ 
    set  $\mathbf{P}^{(0)} = \mathbf{I}_{M_T}, \tilde{\mathbf{H}}_n^{(0)} = \tilde{\mathbf{H}}_n, \alpha = 1$ 
    for  $k = 1 : K$  do
         $\mathbf{s} = \tilde{\mathbf{S}}_{n(:,k)}$ 
         $\tilde{\mathbf{H}}_n^{(k)} = \tilde{\mathbf{H}}_n^{(k-1)} + \frac{(\tilde{\mathbf{Y}}_{n(:,k)} - \tilde{\mathbf{H}}_n^{(k-1)} \mathbf{s})}{\alpha + \mathbf{s}^H \mathbf{P}'^{(k-1)} \mathbf{s}} \mathbf{s}^H \mathbf{P}'^{(k-1)}$ 
         $\mathbf{P}'^{(k)} = \frac{1}{\alpha} \left( \mathbf{P}'^{(k-1)} - \frac{\mathbf{P}'^{(k-1)} \mathbf{s} \mathbf{s}^H \mathbf{P}'^{(k-1)}}{\alpha + \mathbf{s}^H \mathbf{P}'^{(k-1)} \mathbf{s}} \right)$ 
    end
end
Result:  $\tilde{\mathbf{S}}$  and  $\tilde{\mathbf{H}}$ 

```

The remaining two algorithms, namely RLSP (Recursive Least-Squares with Projections) and RSLE (Recursive Least-Squares with Enumeration) are recursive implementations of ILSP and ISLE, respectively. We summarize these two algorithms in Algorithm 5.4 and Algorithm 5.5, respectively. In both algorithms, the channel is estimated based on RLS (Recursive Least Squares), where α is the weighting coefficient and \mathbf{P}' denotes the inverse correlation matrix. Due to the computation of the pseudo-inverse of the channel matrix for the algorithm RLSP, $M_R \geq M_T$ should hold. On the other hand, the RLSE algorithm is suitable for any values of M_T , M_R , and K .

ILSP has the same computational complexity as traditional ZF receivers with the added complexity of the additional iterations if the symbol matrix has full rank. The ILSE algorithm does not compute a pseudo-inverse of the channel matrix. However, its computational complexity comes from the enumeration and it depends on the number of antennas and the modulation order. The recursive algorithms, RLSP and RLSE require a finite number of iterations that is equal to NK . The RLSP algorithm still computes the pseudo-inverse of the channel, however, RLSE does not

Algorithm 5.5: Recursive Least-Squares with Enumeration (RLSE)

```

initialization  $\tilde{\mathcal{H}}_p$ ,  $0 \leq \alpha \leq 1$ 
for  $n = 1 : N$  do
    set  $\mathbf{P}^{(0)} = \mathbf{I}_{M_T}$ ,  $\tilde{\mathbf{H}}_n^{(0)} = \tilde{\mathbf{H}}_n$ ,  $\alpha = 1$ 
    for  $k = 1 : K$  do
         $\hat{\mathbf{s}} = \arg \min_{\mathbf{s}^{(j)} \in \Omega} \|\tilde{\mathbf{Y}}_{n(:,k)} - \tilde{\mathbf{H}}_n^{(k-1)} \mathbf{s}^{(j)}\|$ ,  $\forall j = 1, \dots, M_o^{M_T}$ 
         $\tilde{\mathbf{S}}_{n(:,k)} = \hat{\mathbf{s}}$ 
         $\tilde{\mathbf{H}}_n^{(k)} = \tilde{\mathbf{H}}_n^{(k-1)} + \frac{(\tilde{\mathbf{Y}}_{n(:,k)} - \tilde{\mathbf{H}}_n^{(k-1)} \hat{\mathbf{s}})}{\alpha + \hat{\mathbf{s}}^H \mathbf{P}'^{(k-1)} \hat{\mathbf{s}}} \hat{\mathbf{s}}^H \mathbf{P}'^{(k-1)}$ 
         $\mathbf{P}'^{(k)} = \frac{1}{\alpha} \left( \mathbf{P}'^{(k-1)} - \frac{\mathbf{P}'^{(k-1)} \hat{\mathbf{s}} \hat{\mathbf{s}}^H \mathbf{P}'^{(k-1)}}{\alpha + \hat{\mathbf{s}}^H \mathbf{P}'^{(k-1)} \hat{\mathbf{s}}} \right)$ 
    end
end
Result:  $\tilde{\mathbf{S}}$  and  $\tilde{\mathcal{H}}$ 
    
```

compute any matrix pseudo-inverse. The computational complexity of the RLSE algorithm comes from the enumeration (exhaustive search).

5.1.1.1 Simulation Results

We compare the performance of the five algorithms, ZF, ILSP, ILSE, RLSP, and RLSE (i.e., Algorithms 5.1-5.5) using Monte Carlo simulations and 5000 realizations. First, we consider a 2×2 OFDM system, with K frames, and $N = 128$ subcarriers. The pilot symbols are transmitted on every third subcarrier such that $\Delta F = 3$ and only during the first frame, i.e., $\Delta K = K$. Using these pilots we obtain a pilot based channel estimate with which we initialize all of the algorithms. The transmitted data symbols are independent and modulated using 4-QAM (Quadrature Amplitude Modulation). The frequency selective propagation channel is modeled according to the 3GPP (3rd Generation Partnership Project) Pedestrian A channel (Ped A) [ITU97]. The duration of the cyclic prefix is 32 samples and the weighting factor $\alpha = 1$, for the recursive LS. The maximum number of iterations for the iterative algorithms is set to 7. In Fig. 5.6 and 5.7, we depict the SER (Symbol Error Rate) as a function of the E_b/N_0 (energy per bit/ noise power spectral density) in dB for $K = 2$ and $K = 8$, respectively. Both algorithms based on enumeration, ILSE and RLSE outperform the rest of the algorithms. The performance of ILSP and RLSP is similar to the ZF performance and it depends on the number of frames. As shown in Fig. 5.7, increasing the number of frames leads to a slightly better SER than using a ZF receiver. Note that the transmitted data symbols are independent and randomly drawn with no guarantee that the matrices $\tilde{\mathbf{S}}_n$ are of rank M_T . Therefore, in many cases the number of iterations is equal to one. In all of the other simulated cases, the algorithms converge after 3 iterations.

As in [TVP96], we also observe that the iterative algorithms have a better performance than the recursive ones for an increased number of frames. However, the recursive algorithms, RSLP and

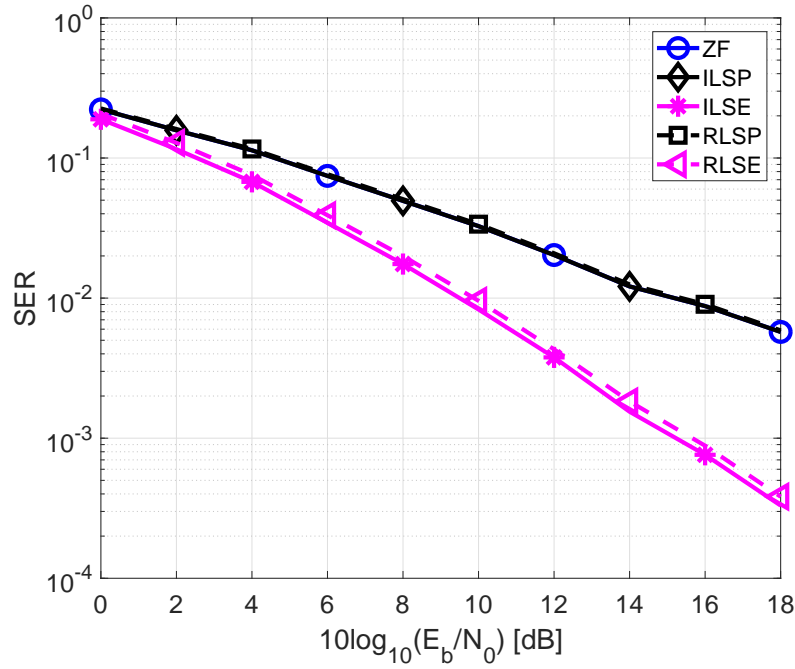


Figure 5.6.: SER for a 2×2 OFDM system with parameters $\Delta K = 2, \Delta F = 3, N = 128, K = 2$.

RLSE, require less computational complexity than the iterative ones, ILSP and ILSE. Moreover, for the same simulation parameters as in Fig. 5.7, but taking into account only 100 realizations and $E_b/N_0 = 10$ dB we depict the computational time required for each algorithm in Table 5.1. The ILSP algorithm requires the smallest amount of time, because additional iterations will not be computed when the symbol matrix does not have a full row rank. The RLSP algorithm requires the longest time as it performs iterations and computes a pseudo-inverse of the channel matrix. The RLSE algorithm has a smaller computational complexity than ILSE and requires a moderate amount of time.

Algorithm	ILSP	ILSE	RLSP	RLSE
Total Time [s]	2.815	6.910	7.962	4.810

Table 5.1.: Computational time required for each algorithm.

Furthermore, in Fig. 5.8, we show the SER as a function of E_b/N_0 in dB for 4×4 and 4×2 MIMO-OFDM systems, respectively. Here, we compare only the recursive algorithms with respect to the ZF receiver, as they are less complex and have a comparable performance as the respective iterative versions. The RLSE outperforms the rest of the algorithms and it is capable of estimating the data symbols even if $M_T > M_R$ without additional spreading as in [NHdA17].

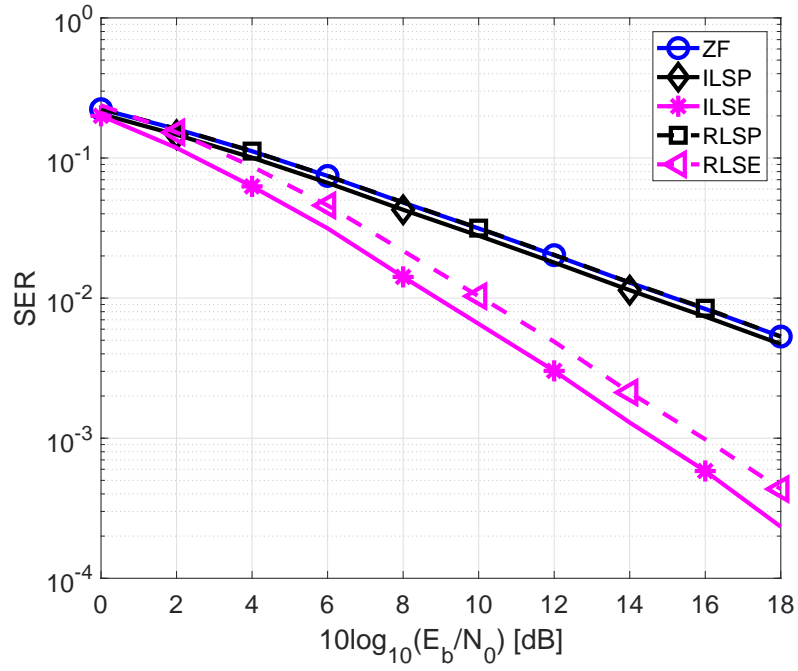


Figure 5.7.: SER for a 2×2 OFDM system with parameters $\Delta K = 8, \Delta F = 3, N = 128, K = 8$.

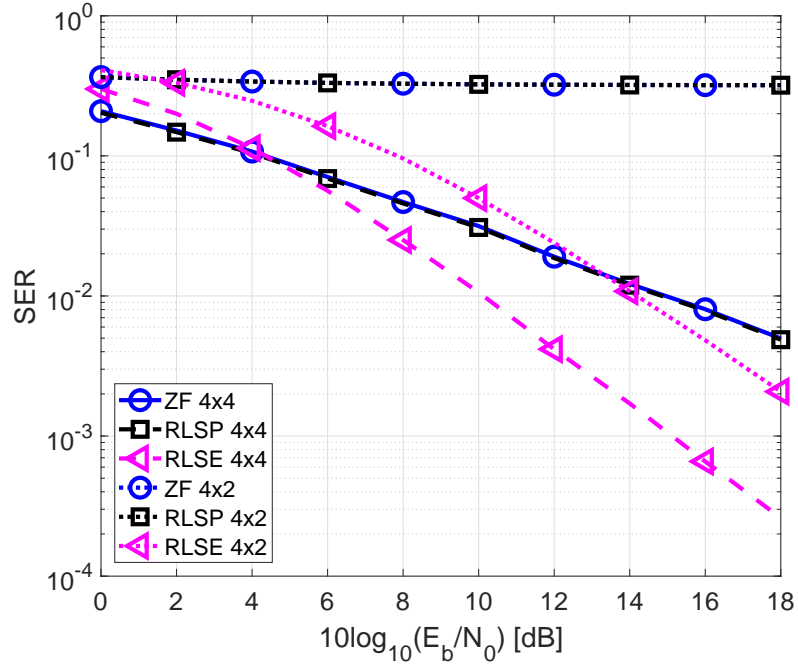


Figure 5.8.: SER for an OFDM system with parameters $\Delta K = 2, \Delta F = 4, N = 512, K = 2$, and $M_T \times M_R$ antennas depicted in the legend.

5.1.1.2 Summary

In this section, we have presented a tensor model for MIMO-OFDM systems based on the generalized tensor contraction operator. The derivation of this model facilitates the design of several types

of receivers based on iterative and recursive LS algorithms. We have compared these algorithms with and without enumeration with the traditional ZF receiver. ILSP and RLSP show a similar performance as the ZF algorithm. Due to the projection, the algorithms based on projection require only a few iterations to converge. The other two algorithms, ISLE and RLSE based on enumeration, outperform the rest of the algorithms at the cost of an increased complexity. Both recursive algorithms, RLSE and RLSP have less computational complexity as compared to their iterative versions ILSE and ILSP, respectively. The RLSE algorithm does not perform matrix inversion. Therefore, the RLSE algorithm is suitable for any configuration setup. It is capable of estimating the data symbols even for $M_T > M_R$ without additional spreading. In the future, recursive algorithms can be used to exploit the correlation of the channel tensor, especially in time varying scenarios. Moreover, the system can be modified such that only specific codewords leading to rank M_T symbol matrices are used. This transmit strategy would guarantee that each symbol matrix is invertible to improve the channel estimates. In addition, we can exploit coding strategies for the transmission of the OFDM symbols over MIMO system, such as Khatri-Rao codes. Finally, it is worth mentioning that our generalized tensor contraction formalism presented here is very general and can be extended to any other multi-carrier system, such as GFDM or FBMC leading to tensor based improvement of these multi-carrier systems. In the following sections, we present the extension of this model to Khatri-Rao coded MIMO-OFDM systems, randomly coded MIMO-OFDM systems, and MIMO-GFDM systems.

5.1.2 Khatri-Rao Coded MIMO-OFDM via generalized tensor contraction

In this section, we model a Khatri-Rao coded MIMO-OFDM communication system as a generalized tensor contraction between a channel and a signal tensor. This generalized tensor contraction is essentially equivalent to the model in (5.1). However, we assume that the signal tensor contains Khatri-Rao coded symbols.

As in the previously presented MIMO-OFDM model without coding (see Section 5.1.1), we assume a MIMO-OFDM communication system with M_T transmit and M_R receive antennas. One OFDM block consists of N samples, which equals the DFT length. Moreover, all N subcarriers are used for data transmission. Furthermore, we assume a frequency-selective channel model that stays constant over the transmission of P frames. In contrast to the model presented in the previous section (Section 5.1.1), here, we assume that the P frames are divided into K groups of Q blocks (Q corresponds to the spreading factor), $P = K \cdot Q$.

Accordingly, the received signal in the frequency domain is given by

$$\tilde{\mathcal{Y}} = \tilde{\mathcal{H}} \bullet_{2,4}^{1,2} \tilde{\mathcal{X}} + \tilde{\mathcal{N}} = \tilde{\mathcal{Y}}_0 + \tilde{\mathcal{N}} \in \mathbb{C}^{N \times M_R \times K \times Q}, \quad (5.14)$$

where $\tilde{\mathcal{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ is the channel tensor and $\tilde{\mathcal{X}} \in \mathbb{C}^{N \times M_T \times K \times Q}$ is the signal tensor. The tensor $\tilde{\mathcal{N}} \in \mathbb{C}^{N \times M_R \times K \times Q}$ contains additive white Gaussian noise and $\tilde{\mathcal{Y}}_0 \in \mathbb{C}^{N \times M_R \times K \times Q}$ is the noiseless received signal.

Channel tensor

In this section, we use the model of the channel tensor $\tilde{\mathcal{H}}$ defined in equation (5.5). Moreover, we have defined the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$ in equation (5.6). Using a permutation matrix, it can be shown that the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[4,2])}$ of the channel is equal to

$$[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = \bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T), \quad (5.15)$$

where

$$\bar{\mathbf{H}} = \underbrace{\begin{bmatrix} \tilde{\mathbf{H}}_R^{(1)} & \dots & \tilde{\mathbf{H}}_R^{(M_T)} \end{bmatrix}}_{\tilde{\mathbf{H}}} \cdot \mathbf{P} \in \mathbb{C}^{M_R \times M_T N}.$$

The permutation matrix $\mathbf{P} \in \mathbb{R}^{NM_T \times M_T N}$ reorders the columns such that the faster increasing index is M_T instead of N and it is defined as $[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = [\tilde{\mathcal{H}}]_{([1,3],[2,4])} \cdot \mathbf{P}$. Recognize that the matrices $\tilde{\mathbf{H}} \in \mathbb{C}^{M_R \times NM_T}$ and $\tilde{\mathbf{H}}_R^{(m_T)} \in \mathbb{C}^{M_R \times N}$ are previously defined in equation (5.7). The structure of the 4-way channel tensor in the frequency domain $\tilde{\mathcal{H}}$ and its unfoldings are derived in Appendix B.3.

Data transmission

We can impose a CP structure on the transmit signal tensor, if we assume Khatri-Rao coded symbols [SB02, dAF13a]. The coding is proportional to the number of transmit antennas if we use a spreading factor $Q = M_T$, for each subcarrier $n = 1, 2, \dots, N$. Hence, the generalized unfolding of the signal tensor is

$$[\tilde{\mathcal{X}}]_{([2,1],[4,3])} = \begin{bmatrix} \tilde{\mathbf{S}}_1 \diamond \mathbf{C}_1 & \tilde{\mathbf{S}}_2 \diamond \mathbf{C}_2 & \dots & \tilde{\mathbf{S}}_N \diamond \mathbf{C}_N \end{bmatrix}^T = \mathbf{I}_{M_T N} (\bar{\mathbf{S}} \diamond \bar{\mathbf{C}})^T, \quad (5.16)$$

where the matrix $\tilde{\mathbf{S}}_n \in \mathbb{C}^{K \times M_T}$ contains modulated data symbols and $\mathbf{C}_n \in \mathbb{C}^{Q \times M_T}$ is a Vandermonde coding matrix as defined in [SB02]. The matrices $\bar{\mathbf{S}} = \begin{bmatrix} \tilde{\mathbf{S}}_1 & \dots & \tilde{\mathbf{S}}_N \end{bmatrix} \in \mathbb{C}^{K \times M_T N}$ and $\bar{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_1 & \dots & \mathbf{C}_N \end{bmatrix} \in \mathbb{C}^{Q \times M_T N}$ contain all symbol and coding matrices for each subcarrier, respectively. Note that $\bar{\mathbf{S}} = \tilde{\mathbf{S}} \cdot \mathbf{P}$, where the matrix $\tilde{\mathbf{S}}$ is defined in equation (5.8) and $\mathbf{P} \in \mathbb{R}^{NM_T \times M_T N}$ is the above mentioned permutation matrix that reorders the columns such that the faster increasing index is M_T instead of N . Moreover, we assume that $\tilde{\mathbf{S}}$ contains pilot symbols as explained after equation (5.8). As shown in [SB02] and as directly follows from (5.16), the tensor $[\tilde{\mathcal{X}}]_{([2,1],3,4)}$ satisfies the following CP decomposition²

$$[\tilde{\mathcal{X}}]_{([2,1],3,4)} = \mathcal{I}_{3, M_T N} \times_1 \mathbf{I}_{M_T N} \times_2 \bar{\mathbf{S}} \times_3 \bar{\mathbf{C}}.$$

² For the definition of the 4-way signal tensor we need to define a selective Kronecker product between two tensors, where only selected modes are involved in the Kronecker product.

The structure of the tensor $[\tilde{\mathcal{X}}]_{([2,1],3,4)}$ resulting from the Khatri-Rao coding is depicted in Fig. 5.9, where $\Theta \in \mathbb{C}^{M_T N \times M_T N}$ is the precoding matrix and $\mathbf{A} \in \mathbb{C}^{M_T N \times K}$ is the symbol matrix. The precoding matrix Θ suitable for QAM constellations with modulation order M_o is given by [SB02]

$$\Theta = \frac{1}{\sqrt{M_o}} \mathbf{F}'_{M_o} \text{diag}(1, \alpha, \dots, \alpha^{M_o-1}),$$

where \mathbf{F}'_{M_o} is an inverse DFT matrix and $\alpha = e^{\frac{j2\pi}{4M_o}}$. Moreover, as previously mentioned, the coding matrices for each subcarrier \mathbf{C}_n ($n = 1, \dots, N$) are chosen to be a Vandermonde matrix with elements $\mathbf{C}_{n(q, m_T)} = e^{j2\pi(m_T-1)(q-1)/M_T}$ [SB02]. Hence, the coding matrices have full row rank, as required for achieving maximum diversity gain [SB02].

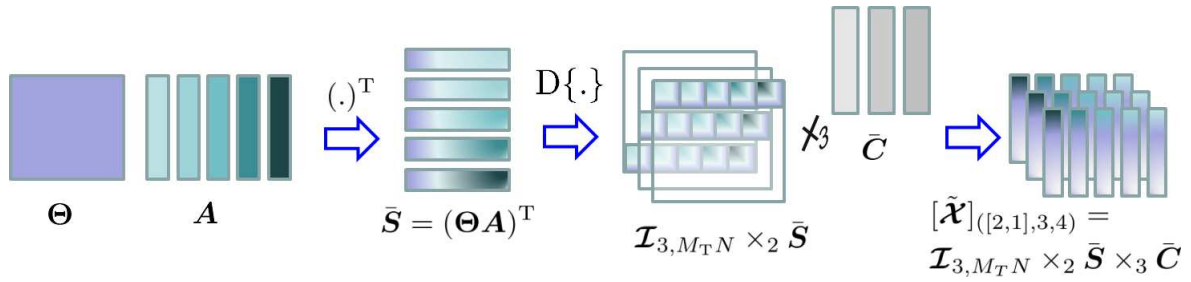


Figure 5.9.: Visualization of the Khatri-Rao coding.

Receiver Design

Using equations (2.10), (2.11), and (5.14), the noiseless received signal can be expressed as

$$[\tilde{\mathcal{Y}}_0]_{([1,2],[4,3])} = [\tilde{\mathcal{H}}]_{([1,3],[4,2])} \cdot [\tilde{\mathcal{X}}]_{([2,1],[4,3])}.$$

Inserting the corresponding unfoldings of the channel and the signal tensor, in equation (5.15) and (5.16), respectively, the noiseless received signal in the frequency domain is given by

$$[\tilde{\mathcal{Y}}_0]_{([1,2],[4,3])} = (\bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)) \cdot (\bar{\mathbf{S}} \diamond \bar{\mathbf{C}})^T.$$

The above equation represents an unfolding of a 4-way tensor with a CP structure. Therefore, the noiseless received signal tensor can be expressed as

$$\tilde{\mathcal{Y}}_0 = \mathcal{I}_{4, M_T N} \times_1 (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \times_2 \bar{\mathbf{H}} \times_3 \bar{\mathbf{S}} \times_4 \bar{\mathbf{C}} \in \mathbb{C}^{N \times M_R \times K \times Q}. \quad (5.17)$$

Equation (5.17) represents the received signal in the frequency domain, for all N subcarriers, M_R receive antennas, and P frames after the removal of the cyclic prefix. Depending on the available a priori knowledge at the receiver side, channel estimation, symbol estimation, or joint channel and symbol estimation can be performed.

Let us compare the MIMO-OFDM tensor model and the Khatri-Rao coded MIMO-OFDM tensor

model in equations (5.11) and (5.17), respectively. First, the factor matrices in these equations have different index ordering. In equation (5.11), the faster increasing index is N , whereas in equation (5.17) the faster increasing index is M_T along the columns of the factor matrices. We use \sim and $-$ to distinguish the different index orderings of the factor matrices. Recall that we have defined a permutation matrix \mathbf{P} that considers the reordering of the columns of the factor matrices. Moreover, equation (5.17) has an additional tensor dimension (the 4-mode) corresponding to the coding technique and the spreading factor Q . Furthermore, taking into account the permutation matrix \mathbf{P} , we get equation (5.11) from equation (5.17) for $Q = 1$ and $\bar{\mathbf{C}} = \mathbf{1}_{M_T N}^T$ (i.e., no coding and the spreading factor equals one).

Using equation (5.17), the channel and the data symbols can be jointly estimated from the $([1, 4], [3, 2])$ generalized unfolding of the noise corrupted received signal

$$[\tilde{\mathbf{Y}}]_{([1,4],[3,2])} \approx (\bar{\mathbf{C}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)) \cdot (\bar{\mathbf{H}} \diamond \bar{\mathbf{S}})^T.$$

Under the assumption that $Q = M_T$, $(\bar{\mathbf{C}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)) \in \mathbb{C}^{NQ \times M_T N}$ is a block diagonal, left invertible matrix and known at the receiver, and using the properties of the coding matrices defined in [SB02], i.e., $\mathbf{C}_n^H \mathbf{C}_n = M_T \mathbf{I}_{M_T}$, we have

$$\bar{\mathbf{Y}} \triangleq \frac{1}{M_T} (\bar{\mathbf{C}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^H \cdot [\tilde{\mathbf{Y}}]_{([1,4],[3,2])} \approx (\bar{\mathbf{H}} \diamond \bar{\mathbf{S}})^T.$$

After transposition, $\bar{\mathbf{Y}}^T \approx \bar{\mathbf{H}} \diamond \bar{\mathbf{S}}$ can be approximated by the Khatri-Rao product between the channel and the data symbols. Therefore, the channel and the data symbols can be jointly estimated based on the LSKRF as in Algorithm 2.2.

Using the LSKRF, the matrices $\bar{\mathbf{H}}$ and $\bar{\mathbf{S}}$ can be identified up to one complex scaling factor ambiguity per column. Hence, the estimated matrices are equal to

$$\hat{\bar{\mathbf{H}}} = \bar{\mathbf{H}} \cdot \mathbf{\Lambda}, \quad (5.18)$$

$$\hat{\bar{\mathbf{S}}} = \bar{\mathbf{S}} \cdot \mathbf{\Lambda}^{-1}, \quad (5.19)$$

where $\mathbf{\Lambda} \in \mathbb{C}^{M_T N \times M_T N}$ is a diagonal matrix with diagonal elements equal to the $M_T N$ complex scaling ambiguities. The simplest way to resolve the scaling ambiguity is by using one row of the matrix $\bar{\mathbf{S}} \in \mathbb{C}^{K \times M_T N}$. This corresponds to $M_T N$ pilot symbols, one pilot symbol per transmit antenna and subcarrier. Since traditional MIMO-OFDM communication systems use less pilot symbols than $M_T N$, we propose to use the same amount of pilot symbols and exploit the channel correlation between adjacent subcarriers in order to estimate the scaling matrix. As illustrated in Figs. 5.4 and 5.5, we transmit pilot symbols on positions with equidistant spacing in the frequency and the time domain. The spacing in the frequency domain is denoted by ΔF and the spacing in the time domain by ΔK . With the prior knowledge of the pilot symbols and their positions, we can obtain an initial channel estimate as in traditional MIMO-OFDM systems (see Section 5.1.1). We denote this pilot based channel estimate by $\tilde{\mathbf{H}}_p$. The pilot based channel estimate is then used

to estimate the scaling ambiguity $\mathbf{\Lambda}$ in equation (5.18) as

$$\hat{\mathbf{\Lambda}} = \text{diag} \left(\frac{1}{M_R} \sum_{m_R=1}^{M_R} \hat{\mathbf{H}}_{(m_R, \cdot)} \oslash \hat{\mathbf{H}}_{p(m_R, \cdot)} \right).$$

By multiplying the solution of the LSKRF with the diagonal matrix $\hat{\mathbf{\Lambda}}$, the scaling ambiguity in equation (5.19) is resolved and the data symbols can be demodulated. Note that the proposed Khatri-Rao receiver estimates the channel and the symbols in a semi-blind fashion. First, the channel and the symbols are jointly estimated without any a priori information. The pilot based channel estimate is then used to resolve the scaling ambiguity affecting the columns of $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$. Therefore, the optimal length and repetition of the piloting sequences are identical as for the traditional OFDM systems. We summarize the steps of the proposed Khatri-Rao (KR) receiver in Algorithm 5.6.

Algorithm 5.6: Khatri-Rao (KR) receiver

initialization $\tilde{\mathbf{H}}_p$ and $\bar{\mathbf{C}}$

Compute $\bar{\mathbf{Y}} = \frac{1}{M_T} (\bar{\mathbf{C}} \oslash (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^H \cdot [\tilde{\mathbf{Y}}]_{([1,4],[3,2])}$.

Compute the LSKRF of $\bar{\mathbf{Y}}^T$ using Algorithm 2.2 that results in $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$.

Compute the scaling matrix $\hat{\mathbf{\Lambda}} = \text{diag} \left(\frac{1}{M_R} \sum_{m_R=1}^{M_R} \hat{\mathbf{H}}_{(m_R, \cdot)} \oslash \hat{\mathbf{H}}_{p(m_R, \cdot)} \right)$. The matrix $\hat{\mathbf{H}}_{p(m_R, \cdot)}$ is defined as in equation (5.15) using the channel tensor $\tilde{\mathbf{H}}_p$ that results from the pilot based channel estimation.

Resolve the scaling ambiguity $\bar{\mathbf{H}} = \hat{\mathbf{H}} \cdot \hat{\mathbf{\Lambda}}^{-1}$ and $\bar{\mathbf{S}} = \hat{\mathbf{S}} \cdot \hat{\mathbf{\Lambda}}$.

Result: $\bar{\mathbf{S}}$ and $\bar{\mathbf{H}}$

Furthermore, the channel estimate resulting from the KR receiver can be used for channel tracking in future transmission frames if the channel has not changed drastically. If the channel estimate is used for tracking, it could be improved by means of an additional LS estimate from $[\tilde{\mathbf{Y}}]_{([2,4,1],[3])}$ with the knowledge of the estimated and projected symbols onto the finite alphabet Ω , i.e., $Q(\bar{\mathbf{S}}) = \text{proj}(\bar{\mathbf{S}})$. The finite alphabet Ω depends on the modulation type and the modulation order M_o .

$$\hat{\mathbf{H}}_{LS}^T = ((\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \oslash \bar{\mathbf{C}} \oslash Q(\bar{\mathbf{S}}))^+ \cdot [\tilde{\mathbf{Y}}]_{([2,4,1],[3])}$$

However, we can also use this improved channel estimation to improve the performance of the KR receiver. Using this updated channel estimate an improved estimate of the diagonal scaling matrix $\hat{\mathbf{\Lambda}}$ can be calculated and with that an enhanced estimate of the symbols, $\hat{\mathbf{S}}_{LS}$, using equation (5.19). Note that, instead of just one LS estimate of the channel and the symbols the performance can be even more enhanced with additional iterations leading to an iterative receiver. Note that the symbol matrix $\hat{\mathbf{S}}_{LS}$ can be estimated in the least squares sense from the 3-mode unfolding of

equation (5.17), but the estimation of $\hat{\mathbf{\Lambda}}$ is computationally cheaper. The KR receiver with its enhancement via LS is summarized in Algorithm 5.7.

Algorithm 5.7: Khatri-Rao receiver and its enhancement via Least-Squares (KR+LS)

initialization $\tilde{\mathcal{H}}_p$ and $\bar{\mathbf{C}}$

Compute $\bar{\mathbf{Y}} = \frac{1}{M_T} (\bar{\mathbf{C}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^H \cdot [\tilde{\mathbf{Y}}]_{([1,4],[3,2])}$.

Compute the LSKRF of $\bar{\mathbf{Y}}^T$ using Algorithm 2.2 that results in $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$.

Compute the scaling matrix $\hat{\mathbf{\Lambda}} = \text{diag} \left(\frac{1}{M_R} \sum_{m_R=1}^{M_R} \hat{\mathbf{H}}_{(m_R, \cdot)} \oslash \hat{\mathbf{H}}_{p(m_R, \cdot)} \right)$. The matrix $\hat{\mathbf{H}}_{p(m_R, \cdot)}$ is defined as in equation (5.15) using the channel tensor $\tilde{\mathcal{H}}_p$ that results from the pilot based channel estimation.

Resolve the scaling ambiguity $\bar{\mathbf{H}} = \hat{\mathbf{H}} \cdot \hat{\mathbf{\Lambda}}^{-1}$ and $\bar{\mathbf{S}} = \hat{\mathbf{S}} \cdot \hat{\mathbf{\Lambda}}$.

Project the symbols onto the finite alphabet Ω , i.e., $Q(\bar{\mathbf{S}}) = \text{proj}(\bar{\mathbf{S}})$.

Compute an enhancement of the channel estimate

$$\hat{\mathbf{H}}_{\text{LS}}^T = ((\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \diamond \bar{\mathbf{C}} \diamond Q(\bar{\mathbf{S}}))^+ \cdot [\tilde{\mathbf{Y}}]_{([2,4,1],[3])}.$$

Improve the estimate of the diagonal scaling matrix

$$\hat{\mathbf{\Lambda}}_{\text{LS}} = \text{diag} \left(\frac{1}{M_R} \sum_{m_R=1}^{M_R} \hat{\mathbf{H}}_{(m_R, \cdot)} \oslash \hat{\mathbf{H}}_{\text{LS}(m_R, \cdot)} \right).$$

Compute an enhancement of the estimate of the symbols $\hat{\mathbf{S}}_{\text{LS}} = \bar{\mathbf{S}} \cdot \hat{\mathbf{\Lambda}}_{\text{LS}}$.

Result: $\hat{\mathbf{S}}_{\text{LS}}$ and $\hat{\mathbf{H}}_{\text{LS}}$

5.1.2.1 Simulation Results

Based on Monte-Carlo simulations, we compare the performance of the traditional frequency domain zero forcing (ZF) receiver (see Algorithm 5.1), the proposed Khatri-Rao (KR) receiver (see Algorithm 5.6) and the proposed Khatri-Rao receiver with one additional LS iteration (see Algorithm 5.7). In the simulations, 5000 realizations and Ped A channel [ITU97] are employed.

In Fig. 5.10, we depict the SER as a function of E_b/N_0 for different numbers of transmitted blocks. Thereby, we consider a MIMO system with the following parameters $N = 128$, $Q = 2$, $M_T = 2$, $M_R = 2$, $\Delta K = K$, $\Delta F = 4$ and different numbers of blocks K (the number of blocks is indicated in the legend). Note that, the KR and the KR+LS receivers benefit from the increased number of frames as the channel has been kept constant during the $P = Q \cdot K$ frames. Moreover, as the number of frames increases, the advantages of the enhancement via LS become more pronounced.

Moreover, the SER comparison for $N = 128$, $Q = M_T$, $K = 2$, $\Delta K = 2$, $\Delta F = 4$, and different

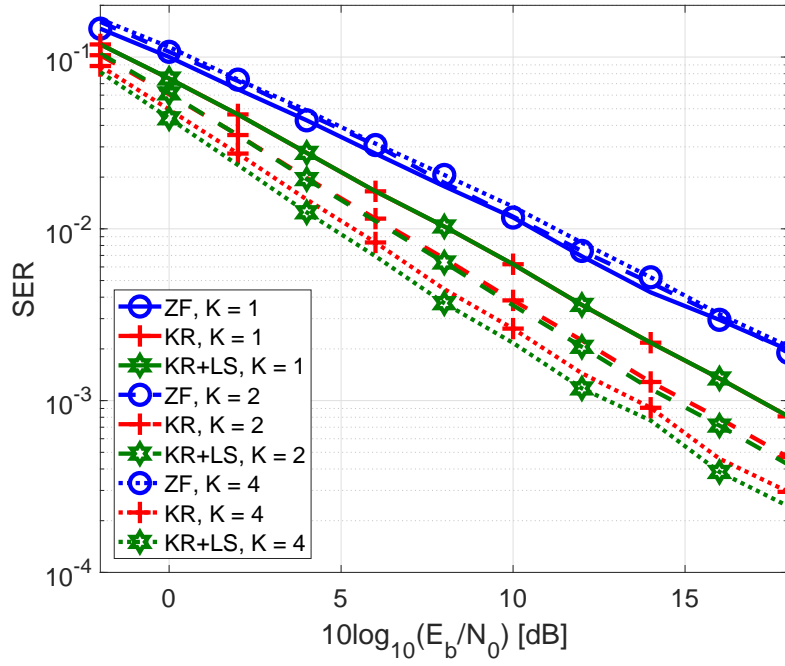


Figure 5.10.: SER comparison for different numbers of transmitted blocks.

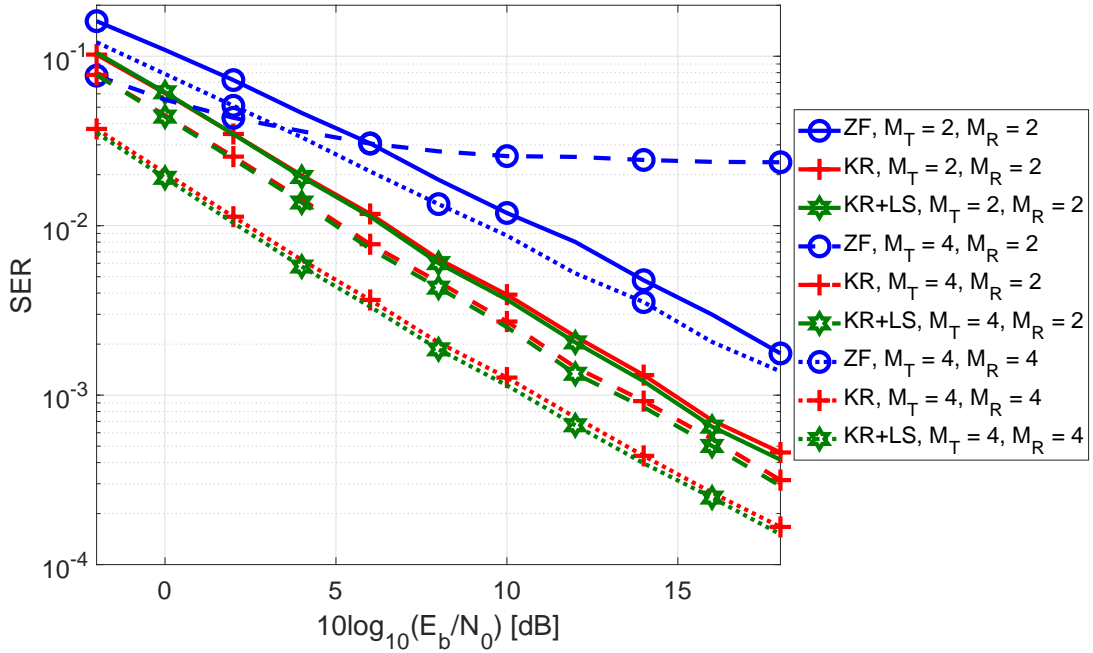


Figure 5.11.: SER comparison for different numbers of transmit and receive antennas.

numbers of antennas is depicted in Fig. 5.11. The KR and KR-LS receivers benefit from an increased number of transmit antennas due to the increased spreading factor, $Q = M_T$. The performance enhancement with the additional LS estimate plays a role only for $K > 2$. However, the KR receiver has a better performance than ZF even without the LS enhancement in terms of the SER.

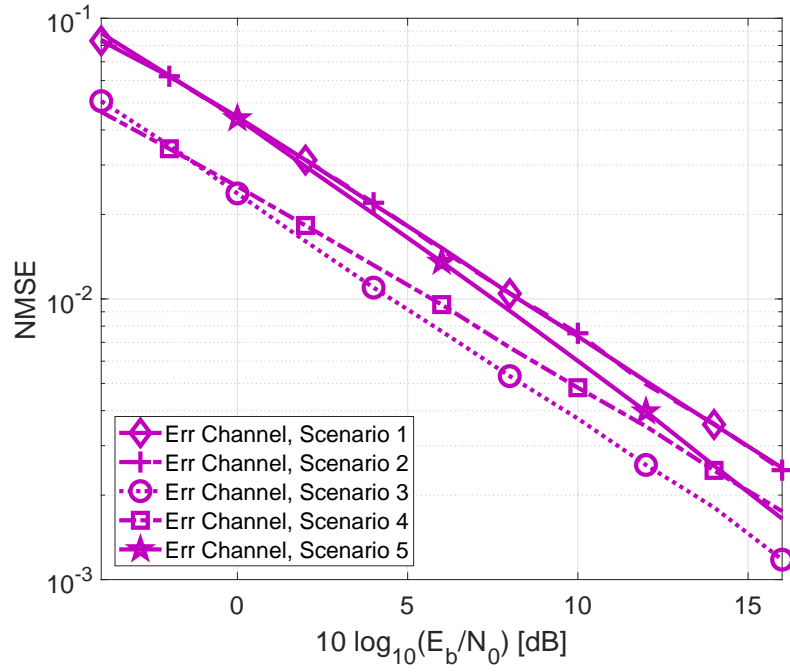


Figure 5.12.: NMSE of the channel matrix for different scenarios after the LSKRF

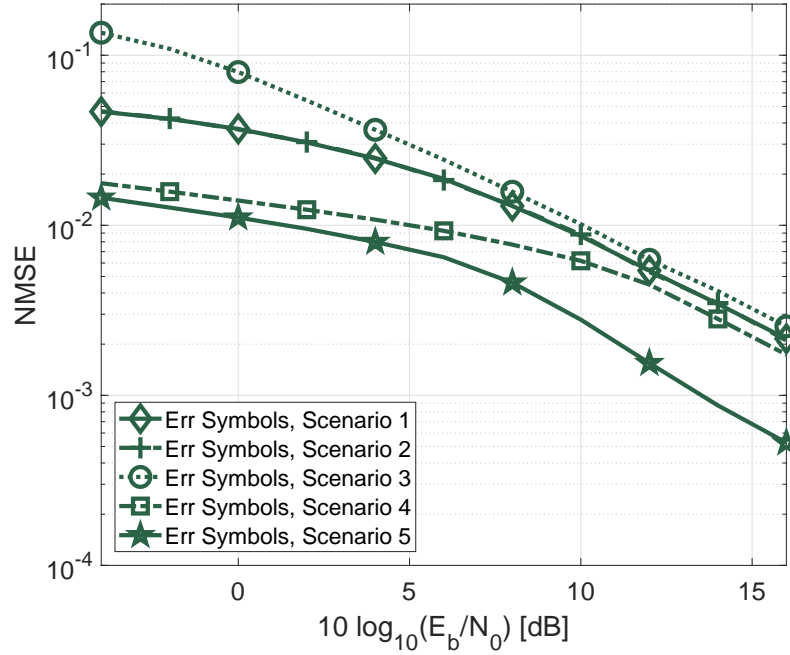


Figure 5.13.: NMSE of the symbol matrix for different scenarios after the LSKRF

Next, we consider the scenarios listed in Table 5.2. In Figs. 5.12 and 5.13, we evaluate the performance of the KR receiver for the five different scenarios defined in Table 5.2. Figs. 5.12 and 5.13 illustrate the estimation error of the channel matrix and the symbol matrix after the LSKRF, respectively. We assume that the scaling matrix is known for both Figs. 5.12 and 5.13.

	M_T	M_R	K	Q	ΔF	ΔK	N
Scenario 1	2	2	2	2	4	2	128
Scenario 2	2	2	2	2	8	2	128
Scenario 3	2	2	4	2	4	4	128
Scenario 4	4	2	2	4	4	2	128
Scenario 5	4	4	2	4	4	2	128

Table 5.2.: Parameters for Figs. 5.12, 5.13, and 5.14

The NMSE (Normalized Mean Squared Error) of the channel matrix is calculated as

$$\text{NMSE} = \frac{\|\hat{\mathbf{H}} - \bar{\mathbf{H}}\|_F^2}{\|\bar{\mathbf{H}}\|_F^2}.$$

In Fig. 5.12, we see that the channel estimate provided by the LSKRF becomes more accurate with the increase of the number of slices K and/or the increase of the spreading factor Q . Hence, larger $P = K \cdot Q$ leads to more accurate estimate of the channel matrix $\hat{\mathbf{H}}$. Fig. 5.13 visualizes the NMSE of the symbol matrix that is computed according to $\|\hat{\mathbf{S}} - \bar{\mathbf{S}}\|_F^2 / \|\bar{\mathbf{S}}\|_F^2$. In contrast to the estimate of the channel matrix, the estimate of the symbol matrix after the LSKRF benefits from the increased number of antennas (receive or transmit antennas).

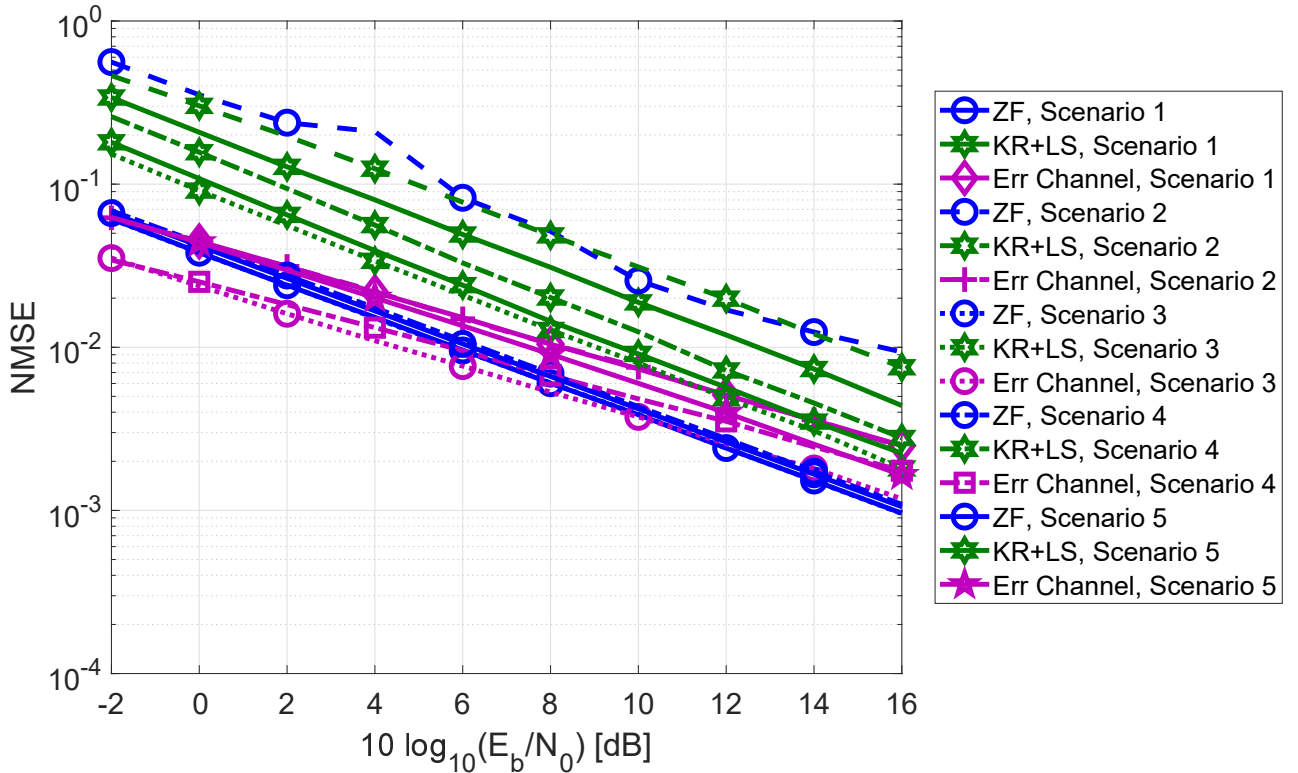


Figure 5.14.: Channel estimation error for different scenarios.

Moreover, in Fig. 5.14, we show the NMSE of the channel estimate for the five scenarios in

Table 5.2. In both figures (Figs. 5.12 and 5.14), the curves in magenta are identical. We can see that the channel estimate based on the LSKRF assuming a perfect estimate of the scaling ambiguity has a better performance than the ZF solution in the low SNR regime. From both figures, we can confirm that the previous findings for the SER and for the channel estimation error are true. The channel estimate is more accurate as the number of transmit blocks K increases or with increased spreading factor Q . On the other hand, the channel estimate using conventional techniques, for instance, [BLM03, HYSH06] strictly depends on the amount of pilot symbols. In order to resolve the scaling ambiguity of the estimated symbol matrix, the channel estimate is required. Therefore, the SER is influenced by both, the accuracy of the pilot based channel estimate and the accuracy of the estimate of the symbols using LSKRF.

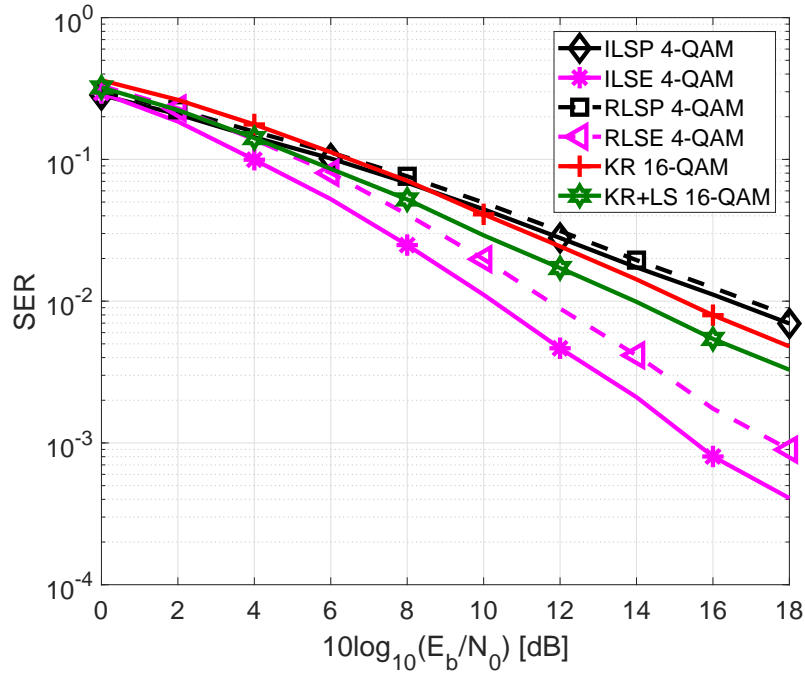


Figure 5.15.: SER for 2×2 OFDM and KR coded OFDM systems, $N = 128$ and $\Delta F = 10$. The OFDM system has the following parameters $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. The KR coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 16-QAM. Hence, both systems transmit 2 Bits/Symbol.

Finally, we compare the receivers proposed in Section 5.1.1 (ILSP, ILSE, RLSP, and RLSE) for a MIMO-OFDM system with the receivers proposed in this section for Khatri-Rao coded MIMO-OFDM system. We assume that both systems have $N = 128$ subcarriers, $M_T = 2$ transmit antennas, $M_R = 2$ receive antennas. Moreover, for both systems we assume that $\Delta F = 10$ is the subcarrier spacing between two pilot symbols in the frequency domain and $\Delta K = K$ is the spacing between two pilot symbols in the time domain. The OFDM system has the following parameters $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. The KR coded OFDM system has the

following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 16-QAM. Hence, we transmit 2 Bits/Symbol with both systems. In Fig. 5.15, we depict the SERs for these two systems. It is obvious that the algorithms ILSE and RLSE from Section 5.1.1 outperform the rest of the algorithms. However, recall that these algorithms are based on enumeration, i.e., they are based on exhaustive search and therefore have much higher computational complexity than the rest of the algorithms. This computation complexity increases with the dimensionality of the system, for instance, increased numbers of antennas or subcarriers. The KR receiver has similar accuracy to the ILSP and the RLSP algorithms that improves with the increased SNR. The KR+LS receiver outperforms the ILSP algorithm, the ILSE algorithm, and the KR algorithm in terms of SER. Recall that the KR coded OFDM model in equation (5.17) has more structure than the OFDM model in equation (5.11) due to the coding. The KR algorithm and KR-LS algorithm exploit this structure to estimate the channel and the symbols. Note that the KR-LS algorithm computes an improved estimate of the scaling matrix. Therefore, KR-LS leads to lower SER than the algorithms ILSP, ILSE, and KR.

5.1.2.2 Summary

In this section, we have shown that the generalized contraction operator can be used to model KR coded MIMO-OFDM communication systems. Using the properties of the contraction and by imposing a CP structure on the signal tensor with Khatri-Rao coding, we derive a novel tensor model for the received signal. By exploiting this model, we propose a receiver for MIMO-OFDM based on the LSKRF. Even though the proposed KR receiver requires the same amount of training symbols as traditional OFDM techniques, it has an improved performance in terms of the SER. Moreover, we propose an enhancement of the KR receiver by means of an additional LS iteration (KR+LS). Moreover, we compare the proposed receiver and its enhancement via LS to the receivers for the uncoded OFDM proposed in Section 5.1.1 and show that the KR coded OFDM system benefits from the additional tensor structure. Among the perspectives for future work, we should consider the use of optimally designed orthogonal pilot sequences, which should enhance the performance of the proposed KR receiver. Moreover, the KR receiver can be extended to an iterative receiver that performs several LS iterations not just one as in the proposed KR+LS receiver. Furthermore, to increase the spectral efficiency of the Khatri-Rao coded MIMO-OFDM system we can consider a system where the coding matrices have random data symbols as entries instead of the Vandermonde structure proposed in [SB02]. This extension of the Khatri-Rao coded MIMO-OFDM system is presented in the following section of this chapter.

5.1.3 Randomly Coded MIMO-OFDM via generalized tensor contraction

In the previous section (Section 5.1.2), we have proposed a tensor model for KR coded MIMO-OFDM systems that introduces an additional CP structure to the signal tensor. Moreover, we have shown the benefits of the additional CP structure of the signal tensor (as compared to the projec-

tion based receivers for MIMO-OFDM systems without coding that have moderate computational complexity, see Section 5.1.1) by proposing a KR receiver based on the LSKRF that outperforms the iterative receivers based on projection for MIMO-OFDM. Recall that the receivers based on enumeration are significantly more computationally expensive than the algorithms based on projection. In Section 5.1.2, the additional CP structure of the signal tensor is achieved by means of a Khatri-Rao coding. However, using the Khatri-Rao coding, we add additional spreading that reduces the spectral efficiency of the system. Therefore, in this section we propose to keep the CP structure of the signal tensor proposed in Section 5.1.2, but to introduce random coding. We introduce the random coding, by using a "coding matrix" that contains data symbols (i.e., the coding matrix $\bar{\mathbf{C}}$ in (5.16) contains also data symbols).

As in Section 5.1.2, the received signal in the frequency domain after the removal of the cyclic prefix is given by

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{H}} \bullet_{4,2}^{2,1} \tilde{\mathbf{X}} + \tilde{\mathbf{N}} = \tilde{\mathbf{Y}}_0 + \tilde{\mathbf{N}} \in \mathbb{C}^{N \times M_R \times K \times Q}, \quad (5.20)$$

where $\tilde{\mathbf{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ is the channel tensor and $\tilde{\mathbf{X}} \in \mathbb{C}^{N \times M_T \times K \times Q}$ is the signal tensor. The tensor $\tilde{\mathbf{N}}$ contains additive white Gaussian noise and $\tilde{\mathbf{Y}}_0$ is the noiseless received signal. As for the KR coded MIMO-OFDM system, we transmit $P = KQ$ frames that are divided into K groups of Q blocks ("spreading factor"). The number of subcarriers is N , and M_R and M_T denote the number of receive and transmit antennas, respectively.

Channel tensor

We model the channel tensor $\tilde{\mathbf{H}}$ according to equation (5.5). Details regarding this model are also provided in Appendix B.3. In this section, we use the generalized unfolding $[\tilde{\mathbf{H}}]_{([1,3],[4,2])} = \bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)$ that is defined in (5.15).

Data Transmission

As previously mentioned, we impose a CP structure on the signal tensor $\tilde{\mathbf{X}}$ similar to the Khatri-Rao coding proposed in Section 5.1.2. For the generalized unfolding $([2,1],[4,3])$ of the signal tensor, we have

$$[\tilde{\mathbf{X}}]_{([2,1],[4,3])} = \begin{bmatrix} \bar{\mathbf{S}}_1 \diamond \mathbf{C}'_1 & \bar{\mathbf{S}}_2 \diamond \mathbf{C}'_2 & \dots & \bar{\mathbf{S}}_N \diamond \mathbf{C}'_N \end{bmatrix}^T = \mathbf{I}_{M_T N} (\bar{\mathbf{S}} \diamond \bar{\mathbf{C}}')^T, \quad (5.21)$$

where the matrix $\bar{\mathbf{S}}_n \in \mathbb{C}^{K \times M_T}$ contains modulated data symbols. In contrast to the Khatri-Rao coding in Section 5.1.2, here, we assume that the first row of the matrix $\mathbf{C}'_n \in \mathbb{C}^{Q \times M_T}$ contains only ones, whereas the remaining $Q - 1$ rows contain modulated data symbols. Hence, the "coding matrix" (the matrix \mathbf{C}_n in (5.16) represents the coding matrix) contains also random entries. We refer to this transmission technique as random coding. Moreover, the matrices $\bar{\mathbf{S}} = \begin{bmatrix} \bar{\mathbf{S}}_1 & \dots & \bar{\mathbf{S}}_N \end{bmatrix} \in$

$\mathbb{C}^{K \times M_T N}$ and $\bar{\mathbf{C}}' = [\mathbf{C}'_1 \ \dots \ \mathbf{C}'_N] \in \mathbb{C}^{Q \times M_T N}$ contain all symbol and random coding matrices for each subcarrier, respectively. Note that $\bar{\mathbf{S}}$ is defined as in Section 5.1.2, i.e., $\bar{\mathbf{S}} = \tilde{\mathbf{S}} \cdot \mathbf{P}$, where the matrix $\tilde{\mathbf{S}}$ is defined in equation (5.8) and $\mathbf{P} \in \mathbb{R}^{N M_T \times M_T N}$ is the permutation matrix that reorders the columns such that the faster increasing index is M_T instead of N . Moreover, we assume that $\tilde{\mathbf{S}}$ contains pilot symbols as explained after equation (5.8). As shown in [SB02] and as directly follows from (5.21), the tensor $[\tilde{\mathbf{X}}]_{([2,1],3,4)}$ satisfies the following CP decomposition ³

$$[\tilde{\mathbf{X}}]_{([2,1],3,4)} = \mathcal{I}_{3, M_T N} \times_1 \mathbf{I}_{M_T N} \times_2 \bar{\mathbf{S}} \times_3 \bar{\mathbf{C}}'.$$

Receiver Design

Using equations (2.10) and (5.20), for the noiseless received signal, we get

$$[\tilde{\mathbf{Y}}_0]_{([1,2],[4,3])} = [\tilde{\mathbf{H}}]_{([1,3],[4,2])} \cdot [\tilde{\mathbf{X}}]_{([2,1],[4,3])}. \quad (5.22)$$

Inserting the corresponding unfoldings of the channel tensor and the signal tensor, i.e., inserting (5.15) and (5.21) into (5.22), we get

$$[\tilde{\mathbf{Y}}_0]_{([1,2],[4,3])} = (\bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)) \cdot (\bar{\mathbf{S}} \diamond \bar{\mathbf{C}}')^T.$$

The above equation represents an unfolding of a 4-way tensor with a CP structure. Therefore, it can be expressed as

$$\tilde{\mathbf{Y}}_0 = \mathcal{I}_{4, M_T N} \times_1 (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \times_2 \bar{\mathbf{H}} \times_3 \bar{\mathbf{S}} \times_4 \bar{\mathbf{C}}' \in \mathbb{C}^{N \times M_R \times K \times Q}. \quad (5.23)$$

Equation (5.23) represents the noiseless received signal in the frequency domain for all N subcarriers, M_R receive antennas, and P frames after the removal of the cyclic prefix for MIMO-OFDM system with RC (Random Coding). Note that the CP decomposition in (5.23) is degenerate in all four modes.

Depending on the available a priori knowledge at the receiver side, channel estimation, symbol estimation, or joint channel and symbol estimation can be performed. For instance, from the 3-mode unfolding of the tensor \mathbf{Y}_0 in (5.23), we can obtain

$$\bar{\mathbf{S}} = [\mathbf{Y}_0]_{(3)} \cdot \left[(\bar{\mathbf{C}}' \diamond \bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^T \right]^+, \quad (5.24)$$

provided that $M_R Q \geq M_T$. Moreover, from the 4-mode unfolding and 2-mode unfolding of tensor

³ For the definition of the 4-way signal tensor, we need to define a selective Kronecker product between two tensors, where only selected modes are involved in the Kronecker product.

\mathcal{Y}_0 in (5.23), we can obtain $\bar{\mathbf{C}}'$ and $\bar{\mathbf{H}}$, respectively.

$$\bar{\mathbf{C}}' = [\mathcal{Y}_0]_{(4)} \cdot \left[(\bar{\mathbf{S}} \diamond \bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^T \right]^+ \quad (5.25)$$

$$\bar{\mathbf{H}} = [\mathcal{Y}_0]_{(2)} \cdot \left[(\bar{\mathbf{C}}' \diamond \bar{\mathbf{S}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^T \right]^+ \quad (5.26)$$

Note that we can compute $\bar{\mathbf{C}}'$ via a pseudo-inverse if $M_R K \geq M_T$.

For noisy observations such as (5.20), the equations (5.24)-(5.26) hold approximately. In this case, we can use the equations (5.24)-(5.26) to estimate the symbols and the channel in an ALS fashion. However, there is no guarantee of convergence if we initialize the ALS algorithm randomly. Therefore, we propose to use the pilot based channel estimate $\bar{\mathbf{H}}_p$ to obtain initial estimates of the matrices $\bar{\mathbf{S}}$ and $\bar{\mathbf{C}}'$ based on LSKRF. This pilot based channel estimated is obtained from the pilot symbols in $\bar{\mathbf{S}}$ and the first row of $\bar{\mathbf{C}}'$ that has entries equal to one. From the $([3,4], [1,2])$ generalized unfolding of the noisy observation \mathcal{Y} , we get

$$[\tilde{\mathcal{Y}}]_{([3,4],[1,2])} \approx [\bar{\mathbf{C}}' \diamond \bar{\mathbf{S}}] \cdot [\bar{\mathbf{H}}_p \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)]^T.$$

Algorithm 5.8: Random Coding-Khatri-Rao (RC-KR) receiver

initialization $\bar{\mathbf{H}}_p$

Compute $\bar{\mathbf{Y}} = [\mathcal{Y}]_{([3,4],[1,2])} \cdot \left[(\bar{\mathbf{H}}_p \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^T \right]^+.$

Compute the LSKRF of $\bar{\mathbf{Y}}$ using the Algorithm 2.2 that results in $\hat{\bar{\mathbf{C}}}'$ and $\hat{\bar{\mathbf{S}}}$.

Compute the scaling matrix $\hat{\mathbf{\Lambda}} = \text{diag}(\hat{\bar{\mathbf{C}}}'_{(1,\cdot)} \oslash \bar{\mathbf{C}}'_{(1,\cdot)})$. (The first row of the matrix $\bar{\mathbf{C}}'$ contains only ones.)

Resolve the scaling ambiguity $\bar{\mathbf{C}}' = \hat{\bar{\mathbf{C}}}' \cdot \hat{\mathbf{\Lambda}}^{-1}$ and $\bar{\mathbf{S}} = \hat{\bar{\mathbf{S}}} \cdot \hat{\mathbf{\Lambda}}$.

Result: $\bar{\mathbf{S}}$ and $\bar{\mathbf{C}}'$

Given $\bar{\mathbf{H}}_p$ and $M_R \geq M_T$, from $[\tilde{\mathcal{Y}}]_{([3,4],[1,2])} \cdot \left[(\bar{\mathbf{H}}_p \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T))^T \right]^+ \approx [\bar{\mathbf{C}}' \diamond \bar{\mathbf{S}}]$ based on LSKRF, we obtain $\hat{\bar{\mathbf{S}}}$ and $\hat{\bar{\mathbf{C}}}'$. However, the matrices $\hat{\bar{\mathbf{S}}}$ and $\hat{\bar{\mathbf{C}}}'$ are estimated up to one complex scaling ambiguity per column. We exploit the first row of the matrix $\bar{\mathbf{C}}'$ to estimate this ambiguity (recall that the elements of the first row of the matrix $\bar{\mathbf{C}}'$ are set to one). After resolving the scaling ambiguity, we propose to iterate between the equations (5.24)-(5.26) to enhance the accuracy of the receiver. Hence, we propose two receivers RC-KR (Random Coding-Khatri-Rao) and RC-KR+ALS (Random Coding-Khatri-Rao+Alternating Least-Squares) for randomly coded MIMO-OFDM systems. These two algorithms are summarized in Algorithm 5.8 and Algorithm 5.9, respectively. The RC-KR receiver exploits the LSKRF to compute an estimate of the symbol matrices $\bar{\mathbf{S}}$ and $\bar{\mathbf{C}}'$, assuming that $M_R \geq M_T$, the first row on the matrix $\bar{\mathbf{C}}'$ contains only ones, and a

Algorithm 5.9: Random Coding-Khatri-Rao + ALS (RC-KR+ALS) receiver

initialization $\bar{\mathbf{H}}_p$

Compute $\bar{\mathbf{Y}} = [\mathbf{Y}]_{([3,4],[1,2])} \cdot \left[\left(\bar{\mathbf{H}}_p \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \right)^T \right]^+$.

Compute the LSKRF of $\bar{\mathbf{Y}}$ using the Algorithm 2.2 that results in $\hat{\bar{\mathbf{C}}}'$ and $\hat{\bar{\mathbf{S}}}$.

Compute the scaling matrix $\hat{\mathbf{\Lambda}} = \text{diag} \left(\hat{\bar{\mathbf{C}}}'_{(1, \cdot)} \oslash \bar{\mathbf{C}}'_{(1, \cdot)} \right)$. (The first row of the matrix $\bar{\mathbf{C}}'$ contains only ones.)

Resolve the scaling ambiguity $\hat{\bar{\mathbf{C}}}' = \hat{\bar{\mathbf{C}}}' \cdot \hat{\mathbf{\Lambda}}^{-1}$ and $\hat{\bar{\mathbf{S}}} = \hat{\bar{\mathbf{S}}} \cdot \hat{\mathbf{\Lambda}}$.

while *does not exceed the maximum number of iterations, does not reach a predefined minimum, or the error of the cost function has not changed within two consecutive iterations* **do**

if $\text{rank} \left(\left[\bar{\mathbf{C}}' \oslash \bar{\mathbf{S}} \oslash (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \right]^T \right) = M_T N$ **then**
 Update $\hat{\mathbf{H}} = [\mathbf{Y}_0]_{(2)} \cdot \left[\left(\hat{\bar{\mathbf{C}}}' \oslash \hat{\bar{\mathbf{S}}} \oslash (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \right)^T \right]^+$
else
 keep the previous estimate of $\hat{\mathbf{H}}$
end

Update $\hat{\bar{\mathbf{C}}}' = [\mathbf{Y}_0]_{(4)} \cdot \left[\left(\hat{\bar{\mathbf{S}}} \oslash \hat{\mathbf{H}} \oslash (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \right)^T \right]^+$.

Project $\hat{\bar{\mathbf{C}}}' = \text{proj} \left(\hat{\bar{\mathbf{C}}}' \right)$ onto the finite alphabet Ω .

Update $\hat{\bar{\mathbf{S}}} = [\mathbf{Y}_0]_{(3)} \cdot \left[\left(\hat{\bar{\mathbf{C}}}' \oslash \hat{\mathbf{H}} \oslash (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \right)^T \right]^+$.

Project $Q(\hat{\bar{\mathbf{S}}}) = \text{proj} \left(\hat{\bar{\mathbf{S}}} \right)$ onto the finite alphabet Ω .

end

Result: $\hat{\bar{\mathbf{S}}}$, $\hat{\bar{\mathbf{C}}}'$, and $\hat{\mathbf{H}}$

pilot based channel estimate $\bar{\mathbf{H}}_p$ is already available. The initial steps of the RC-KR+ALS receiver are equivalent to the RC-KR receiver. In the following steps using the RC-KR+ALS receiver, we estimate the channel matrix and both symbol matrices in an ALS fashion. Therefore, the RC-KR+ALS receiver exploits LSKRF to initialize the ALS algorithm. The ALS algorithm is stopped if it exceeds the number of iterations that is set to 5, reaches a predefined minimum of the cost function $\left\| \tilde{\mathbf{Y}} - \mathbf{I}_{4, M_T N \times 1} (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T) \times_2 \hat{\mathbf{H}} \times_3 \hat{\bar{\mathbf{S}}} \times_4 \hat{\bar{\mathbf{C}}}' \right\|_{\text{H}}^2 / \left\| \tilde{\mathbf{Y}} \right\|_{\text{H}}^2$, or if the error of the cost function has not changed within two consecutive iteration.

5.1.3.1 Simulation Results

In this section, we evaluate the performance of the proposed RC-KR and RC-KR+ALS receivers for randomly coded MIMO-OFDM systems using Monte-Carlo simulations. We consider 2×2 systems,

with $N = 128$ subcarriers, $P = KQ$ frames. Moreover, the spacing between two pilot symbols in the time domain and in the frequency domain is denoted by ΔK and ΔF , respectively. The frequency selective propagation channel is modeled according to the 3GPP Pedestrian A channel [ITU97]. The duration of the cyclic prefix is 32 samples. In the simulations, we use 5000 realizations.

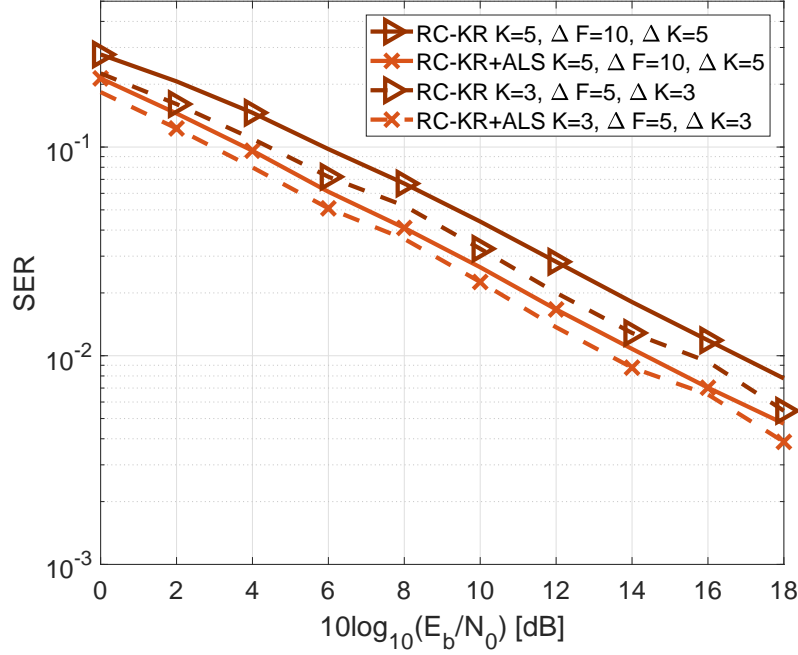


Figure 5.16.: SER for a 2×2 randomly coded OFDM system with parameters $N = 128$, $Q = 2$, K , ΔK , ΔF , and the symbols are modulated using 4-QAM. The parameters K , ΔK and ΔF are indicated in the legend.

In Fig. 5.16, we provide an SER comparison for two scenarios. For both scenarios, we assume 2×2 randomly coded OFDM system, $Q = 2$, and the symbols are modulated using 4-QAM modulation. Moreover, $K = 5$, $\Delta F = 10$, and $\Delta K = 5$, for the first scenario, whereas for the second scenario $K = 3$, $\Delta F = 5$, and $\Delta K = 3$. Hence, in the first scenario we estimate more symbols than in the second scenario, using less pilot symbols. As expected, we achieve a lower SER if more pilot symbols are used because they lead to a more accurate initial pilot based channel estimate. Moreover, in Fig. 5.16, we see that the RC-KR+ALS receiver outperforms the RC-KR receiver. Thus, we benefit from the additional iterations and from exploiting the complete tensor structure. In contrast to RC-KR, RC-KR+ALS also estimates the channel matrix. Furthermore, the accuracy gain of the RC-KR+ALS receiver is more pronounced if we initialize the RC-KR+ALS with a less accurate pilot based channel estimate (the gain is more pronounced for the solid lines than for the dashed lines in Fig. 5.16).

Moreover, in Fig. 5.17, we depict the SER comparison for a 2×2 OFDM system (see Section 5.1.1), Khatri-Rao coded OFDM system (see Section 5.1.2), and randomly coded OFDM system. More precisely, we compare the ILSP receiver, the RLSP receiver, the KR receiver, the KR-LS receiver,

the RC-KR receiver, and the RC-KR+ALS receiver summarized in Algorithms 5.2, 5.4, 5.6, 5.7, 5.8, and 5.9, respectively. In order to assure a fair comparison the following parameters that lead to similar spectral efficiency are selected. The KR coded OFDM system has the following parameters $N = 128$, $\Delta F = 10$, $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$, and the symbols are modulated using 16-QAM. The RC coded OFDM system has the following parameters $N = 128$, $\Delta F = 10$, $K = 5$, $\Delta K = 4$, $Q = 2$, $P = KQ = 10$, and the symbols are modulated using 4-QAM. Finally, the OFDM system has the following parameters $N = 128$, $\Delta F = 10$, $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. Hence, all systems have equal number of pilots and transmit 2 Bits/Symbol (not excluding the pilot symbols). In Fig. 5.17, we see that the RC-KR receiver has the same performance as the ILSP algorithm, even though the second one is an iterative algorithm. The RC-KR+ALS algorithm outperforms the rest of the algorithms especially in the low SNR regime. The KR and KR-LS receivers for KR coded OFDM have different slopes than the uncoded OFDM and the randomly coded OFDM. Note that with the Khatri-Rao coding we achieve maximal diversity [SB02]. Moreover, we observe that larger tensor dimensions lead to larger tensor gain.

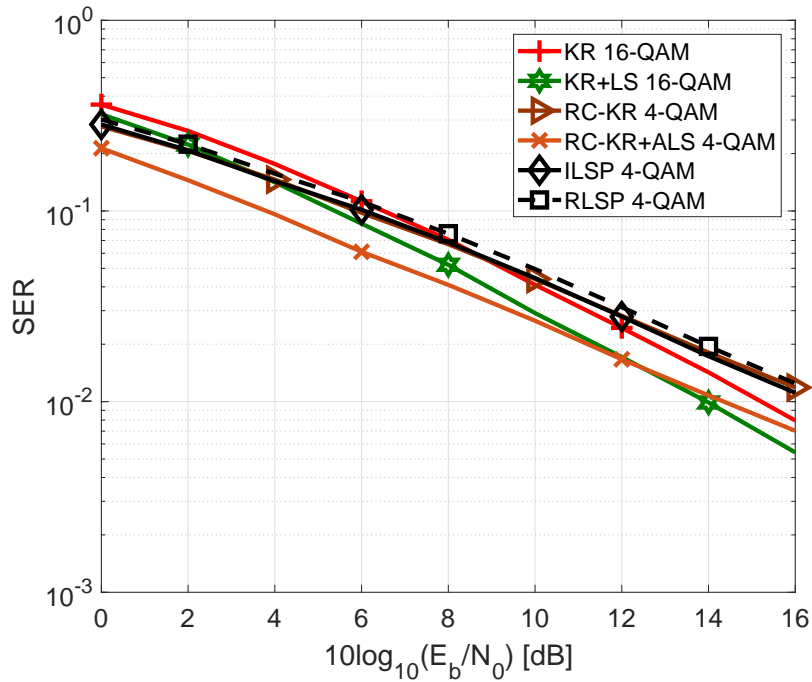


Figure 5.17.: SER for 2×2 KR coded OFDM, randomly coded OFDM, and traditional OFDM systems for $N = 128$ and $\Delta F = 10$. The KR coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 16-QAM. The randomly coded OFDM system has the following parameters $K = 5$, $\Delta K = 5$, $Q = 2$, $P = KQ = 10$ and the symbols are modulated using 4-QAM. The OFDM system has the following parameters $K = 10$, $\Delta K = 10$, and the symbols are modulated using 4-QAM. Therefore, all systems transmit 2 Bits/Symbol.

5.1.3.2 Summary

In this section, we have proposed a randomly coded transmission technique for MIMO-OFDM systems. This transmission technique imposes a CP structure on the signal tensor similar to the Khatri-Rao coded MIMO-OFDM (proposed in Section 5.1.2), but it has higher spectral efficiency. We achieve this higher spectral efficiency such that the "coding matrix" contains random data symbols. Moreover, we show that the received signal for randomly coded MIMO-OFDM can also be expressed in terms of a generalized tensor contraction between the channel tensor and the signal tensor. Exploiting the structure of the resulting receive signal tensor, we propose two receivers RC-KR and RC-KR+ALS for randomly coded MIMO-OFDM systems. The RC-KR receiver estimates the symbol matrices based on the LSKRF using a pilot based channel estimate. However, to resolve the scaling ambiguity that arises from the LSKRF, the first row of the random "coding matrix" has to be known (we set its values to ones). The RC-KR+ALS receiver is an iterative receiver that estimates the symbol matrices and the channel matrix in an ALS fashion. This iterative algorithm requires only a few iterations to converge due to the projection of the estimated symbol matrices on the finite alphabet of modulated symbols. Note that we initialize the RC-KR+ALS receiver with the estimates of the symbol matrices that are obtained using the RC-KR receiver. Therefore, both receivers assume that $M_R \geq M_T$, the first row of the random "coding matrix" is known and there is a pilot based channel estimate available beforehand. The proposed RC-KR+ALS outperforms the iterative receivers for MIMO-OFDM because it exploits the additional tensor structure of the signal tensor. Furthermore, the system can be modified such that both symbol matrices contain symbols from different constellations. This will lead to a resulting signal tensor with more diverse entries. These diverse entries will potentially increase the rank of the matrices that will provide more accurate estimates via a pseudo-inverse. In the future, we should investigate the optimal combinations of modulation order and modulation type for different SNRs, for both symbol matrices.

5.2 Generalized Frequency Division Multiplexing (GFDM)

GFDM is one of the multi-carrier transmission techniques considered as an alternative to OFDM for beyond 5G wireless communication systems. GFDM is a flexible multi-carrier scheme that spreads the data symbols in a time-frequency block [MMG⁺14]. Compared to OFDM, in GFDM each subcarrier is additionally filtered with a circular pulse shaping filter. OFDM requires a significant signaling overhead due to its strict synchronization requirements, which is a major drawback for the application scenarios being considered for beyond 5G systems. In contrast to OFDM, GFDM has less stringent synchronization requirements. On the other hand, GFDM introduces ISI due to the fact that (unlike OFDM) not all symbols are transmitted on orthogonal subcarriers. Therefore, especially for frequency selective channels, ISI cancellation has to be included, as presented in [AMGG13].

Tensor algebra efficiently describes multi-dimensional signals, preserves their structure, and provides an improved identifiability. Moreover, in the past, communication systems have been modeled using tensor algebra and often showed a tensor gain compared to matrix based receivers. For instance, tensors and tensor decompositions have been used to describe various communication systems as discussed in [dAFX13, SGB00, FdA14a]. In these works, wireless communication systems are modelled using the PARATUCK2 or the generalized PARATUCK2 decomposition. Similarly, in [NCH⁺17], we show that the GFDM transmit signal can also be defined as a PARATUCK2 model. Furthermore, in Section 5.1 and in [NHdA17, NHdA18], we propose a model for MIMO-OFDM systems based on the generalized contraction operator that can be extended to any multi-carrier technique. In this section, we present the extension of this model to MIMO-GFDM systems. We also exploit this model for the design of a very simple iterative receiver that has better performance than ZF receivers.

The GFDM modulated signal is given by [MMG⁺14]

$$x_n = \sum_{r=1}^R \sum_{m=1}^M d_{r,m} p_{r,n} g_{m,n}, \quad \forall n = 1, \dots, N, \quad (5.27)$$

where M is the number of complex time subsymbols to be transmitted on R subcarriers, $N = R \cdot M$ is the block length in the time domain, and $d_{r,m}$ are the complex modulated data symbols. Note that the notation used in this section is different than the notation in Section 5.1 dedicated to OFDM systems. The data symbols are filtered with the filter coefficients $g_{m,n}$ (for example, a root raised cosine filter) and are accordingly shifted to the corresponding subcarrier $p_{r,n} = \exp(j2\pi \frac{r}{N}n)$ as explained in [MMG⁺14].

Next, we assume a MIMO-GFDM system with M_T transmit and M_R receive antennas [EMZF16, CNA⁺16]. The GFDM modulated signal for each transmit antenna $m_T = 1, \dots, M_T$ and for each sample $n = 1, \dots, N$ is defined as

$$x_{n,m_T} = \sum_{r=1}^R \sum_{m=1}^M d_{r,m,m_T} p_{r,n} g_{m,n}, \quad (5.28)$$

where d_{r,m,m_T} are the complex GFDM modulated data symbols for each transmit antenna m_T . Note that the GFDM modulated signal can be represented by a matrix $\mathbf{X} \in \mathbb{C}^{N \times M_T}$ with elements $\mathbf{X}_{(m,m_T)} = x_{n,m_T}$. Moreover, using the DFT, we can transform the transmitted signal \mathbf{X} into the frequency domain, i.e., $\tilde{\mathbf{X}} = \mathbf{F}_N \cdot \mathbf{X}$, where \mathbf{F}_N is a DFT matrix of size $N \times N$ defined in (A.5).

Note that the model proposed in equation (5.1) is a general model that describes the received signal for multi-carrier MIMO systems. Hence, for the received signal in the frequency domain after the removal of the cyclic prefix, we have

$$\tilde{\mathbf{Y}} = \tilde{\mathcal{H}} \bullet_{2,4}^{1,2} \tilde{\mathbf{X}} + \tilde{\mathbf{N}} = \tilde{\mathbf{Y}}_0 + \tilde{\mathbf{N}}. \quad (5.29)$$

The matrices, $\tilde{\mathbf{Y}} \in \mathbb{C}^{N \times M_R}$, $\tilde{\mathbf{Y}}_0 \in \mathbb{C}^{N \times M_R}$, and $\tilde{\mathbf{N}} \in \mathbb{C}^{N \times M_R}$, represent the noisy received signal, the

noiseless received signal, and the noise matrix, respectively.

Channel tensor

We model the frequency selective channel using the 4-way channel tensor $\tilde{\mathcal{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ defined in (5.2). We derive the structure of the 4-way channel tensor in the frequency domain and its unfoldings in Appendix B.3. Hence, the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$ is given by

$$[\tilde{\mathcal{H}}]_{([1,3],[2,4])} = \tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \in \mathbb{C}^{NM_R \times NM_T}, \quad (5.30)$$

where the matrix $\tilde{\mathbf{H}} \in \mathbb{C}^{M_R \times NM_T}$ as defined in (5.7) contains all channel coefficients in the frequency domain. Moreover, the matrix $\mathbf{H} \in \mathbb{C}^{M_R \times LM_T}$ represents the channel matrix in the time domain. Hence, the matrix \mathbf{H} contains all channel impulse responses $\mathbf{h}_L^{(m_R, m_T)} \in \mathbb{C}^{L \times 1}$, for $m_R = 1, \dots, M_R$, $m_T = 1, \dots, M_T$, and L is the number of channel taps as explained in Section 5.1.1.

$$\mathbf{H} = \left[\begin{array}{c|c|c|c} \mathbf{h}_L^{(1,1)T} & \mathbf{h}_L^{(1,2)T} & \dots & \mathbf{h}_L^{(1,M_T)T} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{h}_L^{(M_R,1)T} & \mathbf{h}_L^{(M_R,2)T} & \dots & \mathbf{h}_L^{(M_R,M_T)T} \end{array} \right] \quad (5.31)$$

The relationship between the channel matrix in the time domain and the frequency domain is given by

$$\tilde{\mathbf{H}}^T = (\mathbf{I}_{M_T} \otimes \mathbf{F}_{N \times L}) \mathbf{H}^T, \quad (5.32)$$

where the matrix $\mathbf{F}_{N \times L} \in \mathbb{C}^{N \times L}$ contains the first L columns of the DFT matrix \mathbf{F}_N .

Data Transmission

Recall that the GFDM modulated signal for each transmit antenna $m_T = 1, \dots, M_T$ and for each sample $n = 1, \dots, N$ is defined as

$$x_{n,m_T} = \sum_{r=1}^R \sum_{m=1}^M d_{r,m,m_T} p_{r,n} g_{m,n}, \quad (5.33)$$

where d_{r,m,m_T} are the complex modulated data symbols for each transmit antenna m_T . Let us define the data tensor $\mathcal{D} \in \mathbb{C}^{R \times M \times M_T}$ that contains the elements $\mathcal{D}_{(r,m,m_T)} = d_{r,m,m_T}$. Moreover, we define the matrices $\mathbf{P} \in \mathbb{C}^{R \times N}$ with elements $\mathbf{P}_{(r,n)} = p_{r,n}$, $\mathbf{G} \in \mathbb{C}^{M \times N}$ with elements $\mathbf{G}_{(m,n)} = g_{m,n}$, and $\mathbf{X} \in \mathbb{C}^{N \times M_T}$ with elements $\mathbf{X}_{(n,m_T)} = x_{n,m_T}$. The elements of these matrices are defined at the beginning of this section. Hence, the matrices \mathbf{P} , \mathbf{G} , and \mathbf{X} contain the subcarriers, the filter coefficients, and the GFDM modulated symbols, respectively. From equation (5.33), we define the

tensor $\mathcal{T} \in \mathbb{C}^{R \times N \times M_T}$ as

$$\mathcal{T}_{(r,n,m_T)} = \sum_{m=1}^M d_{r,m,m_T} g_{m,n} \Leftrightarrow \mathcal{T} = \mathcal{D} \times_2 \mathbf{G}^T.$$

Using the defined tensor \mathcal{T} , the matrices \mathbf{X} and \mathbf{P} , and the definition of the contraction operator in (2.9), for equation (5.28), we get

$$\mathbf{X} = \mathcal{D}_P \bullet_{1,2}^{1,2} \mathcal{T},$$

where $\mathcal{D}_P = \mathcal{I}_{3,N} \times_1 \mathbf{P} \in \mathbb{C}^{R \times N \times N}$. Inserting the structure of the tensors \mathcal{D}_P and \mathcal{T} into the above equation, the transmit signal becomes

$$\mathbf{X} = (\mathcal{I}_{3,N} \times_1 \mathbf{P}) \bullet_{1,2}^{1,2} (\mathcal{D} \times_2 \mathbf{G}^T).$$

Next, using the property (2.10), we have

$$\begin{aligned} \mathbf{X} &= [\mathcal{I}_{3,N} \times_1 \mathbf{P}]_{(3,[1,2])} \cdot [\mathcal{D} \times_2 \mathbf{G}^T]_{([1,2],3)} \\ &= (\mathbf{I}_N \diamond \mathbf{P})^T \cdot (\mathbf{G}^T \otimes \mathbf{I}_R) \cdot [\mathcal{D}]_{([1,2],3)} \end{aligned}$$

Transposing the above equation and using property (2.18), we have

$$\begin{aligned} \mathbf{X}^T &= [\mathcal{D}]_{(3,[1,2])} \cdot (\mathbf{G}^T \otimes \mathbf{I}_R)^T \cdot (\mathbf{I}_N \diamond \mathbf{P}) \\ &= [\mathcal{D}]_{(3,[1,2])} \cdot (\mathbf{G} \otimes \mathbf{I}_R) \cdot (\mathbf{I}_N \diamond \mathbf{P}) \\ &= [\mathcal{D}]_{(3,[1,2])} \cdot (\mathbf{G} \diamond \mathbf{P}) \end{aligned}$$

Finally, by transposing \mathbf{X}^T , we get

$$\mathbf{X} = (\mathbf{G} \diamond \mathbf{P})^T \cdot [\mathcal{D}]_{([1,2],3)} \in \mathbb{C}^{N \times M_T}. \quad (5.34)$$

Hence, the GFDM modulation matrix denoted by \mathbf{A} in [AMGG13, MMG⁺14, EMZF16, CNA⁺16] equals $\mathbf{A} = (\mathbf{G} \diamond \mathbf{P})^T$ [NCH⁺17].

The transmit signal in the frequency domain is then given by

$$\tilde{\mathbf{X}} = \mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T \cdot [\mathcal{D}]_{([1,2],3)} \in \mathbb{C}^{N \times M_T}. \quad (5.35)$$

Moreover, we assume that the data tensor consists of data and pilot symbols, $\mathcal{D} = \mathcal{D}_d + \mathcal{D}_p$. The unfoldings \mathcal{D}_d and \mathcal{D}_p represent the data symbols and the pilot symbols, respectively. We assume that the pilot symbols are placed on equidistant positions in the frequency domain, for each antenna. This spacing in the frequency domain is denoted by ΔF . Moreover, in order to separate the different transmit antennas, the pilot symbols corresponding to the different antennas should be orthogonal to each other. Therefore, assuming two transmit antennas, the pilot positions

dedicated to antenna one are forbidden as pilot positions for the antenna two or to transmit data. We explain the piloting sequences in Section 5.1.1 and we visualize them in Fig. 5.5. Note that in Section 5.1.1 we transmit K frames, whereas here, we transmit only one frame, i.e., $K = 1$.

Receiver Design

Using the property (2.10), for the noiseless part of the received signal in the frequency domain given in (5.29), we get

$$\underbrace{[\tilde{\mathbf{Y}}_0]_{([1,2],0)}}_{\text{vec}(\tilde{\mathbf{Y}}_0)} = [\tilde{\mathcal{H}}]_{([1,3],[2,4])} \cdot \underbrace{[\tilde{\mathbf{X}}]_{([1,2],0)}}_{\text{vec}(\tilde{\mathbf{X}})}. \quad (5.36)$$

After substituting equation (5.30) into equation (5.36), we have

$$\text{vec}(\tilde{\mathbf{Y}}_0) = [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \text{vec}(\tilde{\mathbf{X}}). \quad (5.37)$$

The above equation of the received signal in the frequency domain represents a CP model that is degenerate in all modes, and it is given by

$$\tilde{\mathbf{Y}}_0 = \mathcal{I}_{3, NM_T} \times_1 (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \times_2 \tilde{\mathbf{H}} \times_3 \text{vec}(\tilde{\mathbf{X}})^T \in \mathbb{C}^{N \times M_R}. \quad (5.38)$$

Equation (5.38) has a similar structure as the OFDM received signal in (5.11) if we consider only one frame ($K = 1$ in (5.11)). Moreover, in equation (5.38) the symbols are GFDM modulated. Furthermore, equation (5.38) can be easily extended to multiple frames ($k = 1, \dots, K$) by concatenating multiple rows of GFDM modulated symbols along the 3-mode, i.e.,

$$\begin{bmatrix} \text{vec}(\tilde{\mathbf{X}}_1)^T \\ \vdots \\ \text{vec}(\tilde{\mathbf{X}}_K)^T \end{bmatrix}.$$

For the received signal containing K frames, we get

$$\tilde{\mathcal{Y}}_0 = \mathcal{I}_{3, NM_T} \times_1 (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \times_2 \tilde{\mathbf{H}} \times_3 \begin{bmatrix} \text{vec}(\tilde{\mathbf{X}}_1)^T \\ \vdots \\ \text{vec}(\tilde{\mathbf{X}}_K)^T \end{bmatrix} \in \mathbb{C}^{N \times M_R \times K}.$$

We can estimate the GFDM modulated symbols from equation (5.38) using a ZF filter as described in Algorithm 5.1. Note that the ZF receiver requires a prior pilot based channel estimate $\tilde{\mathcal{H}}_p$. The pilot based channel estimate is obtained from the pilot symbols within the tensor \mathcal{D} as explained beforehand and in Section 5.1.1. Additionally, we demodulate the data symbols based

on a ZF filter according to equation (5.34), i.e., $[(\mathbf{G} \diamond \mathbf{P})^T]^+ \cdot \mathbf{X} = [\mathcal{D}]_{([1,2],3)}$ [MDK⁺17]. Another alternative for the demodulation of the GFDM symbols is a matched filter [MDK⁺17]. However, the ZF filter is more accurate than the matched filter. Therefore, we use the ZF receiver to demodulate the GFDM symbols as a benchmark.

Moreover, we propose an iterative receiver that is initialized with the pilot based channel estimate $\tilde{\mathbf{H}}_p$. Note that this pilot based channel estimate is also used to initialize the ZF algorithm (see Algorithm 5.1). Considering the noisy observation of the received signal in equation (5.29) and the structure of the noiseless tensor in (5.38), we can estimate the transmitted data symbols in an LS sense. Recall that equation (5.37) represents the $[\tilde{\mathbf{Y}}_0]_{([1,2],0)}$ unfolding of the noiseless tensor. Similarly, for the noisy received signal, we have

$$\text{vec}(\tilde{\mathbf{Y}}) = [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \text{vec}(\tilde{\mathbf{X}}) + \text{vec}(\tilde{\mathbf{N}}). \quad (5.39)$$

Taking into account that $\tilde{\mathbf{X}}$ is a GFDM modulated signal in the frequency domain with a structure given in (5.35) and the property (2.24), the noisy received signal is given by

$$\begin{aligned} \text{vec}(\tilde{\mathbf{Y}}) &= [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \text{vec}(\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T \cdot [\mathcal{D}]_{([1,2],3)}) + \text{vec}(\tilde{\mathbf{N}}) \\ &= [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \text{vec}(\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T \cdot [\mathcal{D}]_{([1,2],3)} \cdot \mathbf{I}_{M_T}) + \text{vec}(\tilde{\mathbf{N}}) \\ &= \underbrace{[\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot [\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T)]}_{*} \cdot \text{vec}([\mathcal{D}]_{([1,2],3)}) + \text{vec}(\tilde{\mathbf{N}}). \end{aligned} \quad (5.40)$$

Recall that the GFDM modulated symbols are not orthogonal since they are spread over R subcarriers and M time subsymbols. Therefore, the block diagonal channel matrix in the part denoted by $*$ in equation (5.40) is additionally multiplied by $\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T)$ as compared to OFDM. Fig. 5.18 depicts the structure of the matrix denoted by $*$ in equation (5.40) for a MIMO-GFDM system with parameters $M_T = 2$, $M_R = 3$, and $N = 3$. Recognize that in contrast to the block diagonal structure of the channel matrix depicted in Fig. 5.3 for MIMO-OFDM systems, the matrix in Fig. 5.18 is a full matrix.

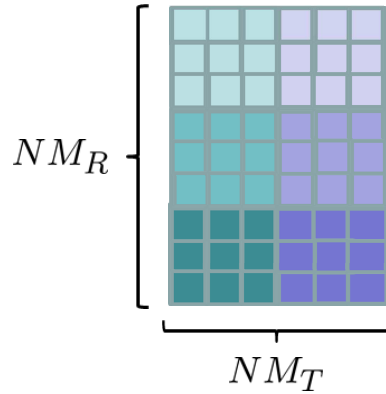


Figure 5.18.: Visualization of the matrix $[\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot [\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T)]$ for a MIMO-GFDM system with parameters $M_T = 2$, $M_R = 3$, and $N = 3$.

Using the assumption $M_R \geq M_T$, for equation (5.40), we get the LS estimate of the data symbols according to

$$\text{vec}([\hat{\mathcal{D}}]_{([1,2],3)}) = [[\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot [\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P}))^T]]^+ \cdot \text{vec}(\tilde{\mathbf{Y}}).$$

Next, we project the estimated symbols on the finite alphabet Ω that depends on the modulation order and the modulation type. Afterwards, we exploit these projected symbols to compute an improved estimate of the channel tensor $\tilde{\mathcal{H}}$. For the 2-mode unfolding of the noisy received signal $\tilde{\mathbf{Y}}$ (the transpose of $\tilde{\mathbf{Y}}$), we have

$$[\tilde{\mathbf{Y}}]_{(2,[1,3])} = \tilde{\mathbf{Y}}^T = \tilde{\mathbf{H}} \cdot [\text{vec}(\tilde{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)]^T + \tilde{\mathbf{N}}^T \quad (5.41)$$

However, the structure of the matrix $[\text{vec}(\tilde{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)]^T \in \mathbb{C}^{NM_T \times N}$ does not allow us to compute an LS estimate of the channel in the frequency domain. The authors of [EMZF16] also show this. We can compute an improved estimate of the channel in the frequency domain if we transmit multiple frames $K \geq M_T$. Moreover, we can compute the estimate of the channel in the time domain using equation (5.32). Here, we propose to compute the channel estimate in the time domain because in the time domain there are less unknowns that should be estimated (in the time domain, we estimate $L \ll N$ unknowns per receive-transmit antenna pair, where L denotes the number of channel taps and N denotes the number of samples/channel coefficients in the frequency domain). Therefore, we consider equation (5.32) and the transpose of equation (5.41) that is given by

$$\begin{aligned} \tilde{\mathbf{Y}} &= [\text{vec}(\tilde{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \tilde{\mathbf{H}}^T + \tilde{\mathbf{N}} \\ &= [\text{vec}(\tilde{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot (\mathbf{I}_{M_T} \otimes \mathbf{F}_{N \times L}) \cdot \mathbf{H}^T + \tilde{\mathbf{N}}, \end{aligned} \quad (5.42)$$

where the matrix $\mathbf{F}_{N \times L} \in \mathbb{C}^{N \times L}$ contains the first L columns of the DFT matrix \mathbf{F}_N and L denotes the number of channel taps. Moreover, the matrix $\mathbf{H} \in \mathbb{C}^{M_R \times LM_T}$ represents the channel matrix in the time domain as previously defined in equation (5.31). From equation (5.42), we can compute an LS estimate of the matrix \mathbf{H} , i.e.,

$$\hat{\mathbf{H}}^T = \tilde{\mathbf{Y}} [[\text{vec}(\tilde{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot (\mathbf{I}_{M_T} \otimes \mathbf{F}_{N \times L})]^+$$

The proposed ALS algorithm for the joint channel estimation and symbols estimation for GFDM systems is summarized in Algorithm 5.10. This ALS algorithm is stopped if it exceeds the number of iterations that is set to 5, reaches a predefined minimum of the cost function given by $\|\tilde{\mathbf{Y}} - \tilde{\mathcal{H}} \bullet_{2,4}^{1,2} \tilde{\mathbf{X}}\|_{\text{F}}^2 / \|\tilde{\mathbf{Y}}\|_{\text{F}}^2$, or if the error of the cost function has not changed within two consecutive iteration. Note that we have proposed this receiver in [NCH⁺17], however, here it has been derived by means of the generalized tensor contractions.

Algorithm 5.10: Alternating Least-Squares (ALS) receiver

```

initialization  $\tilde{\mathcal{H}}_p$ 
while does not exceed the maximum number of iterations, does not reach a predefined
minimum, or the error of the cost function has not changed within two consecutive
iterations do
    Update  $\text{vec}([\hat{\mathcal{D}}]_{([1,2],3)}) = [[\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot [\mathbf{I}_{M_T} \otimes (\mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P}))^T]]^+ \cdot \text{vec}(\tilde{\mathbf{Y}})$ .
    Project  $\hat{\mathcal{D}} = \text{proj}(\hat{\mathcal{D}})$  onto the finite alphabet  $\Omega$ .

    Compute  $\hat{\mathbf{X}} = \mathbf{F}_N \cdot (\mathbf{G} \diamond \mathbf{P})^T \cdot [\hat{\mathcal{D}}]_{([1,2],3)}$ .
    Update  $\hat{\mathbf{H}}^T = \mathbf{Y} \left[ \left[ \text{vec}(\hat{\mathbf{X}}) \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \right] \cdot (\mathbf{I}_{M_T} \otimes \mathbf{F}_{N \times L}) \right]^+$ .
    Compute the channel in the frequency domain  $\hat{\mathbf{H}} = \hat{\mathbf{H}} \cdot (\mathbf{I}_{M_T} \otimes \mathbf{F}_{N \times L})^T$ .
end
Result:  $\hat{\mathcal{D}}$  and  $\hat{\mathcal{H}}$ 

```

5.2.1 Simulation Results

For simulation purposes, we have considered a 2×2 GFDM system with a frequency selective channel, more precisely the 3GPP Pedestrian A channel (Ped A) [ITU97]. For this simulations results, we use 3000 realizations. The data symbols are modulated using 4-QAM or 16-QAM. The modulated data symbols are transmitted on 32 subcarriers ($R = 32$) with subcarrier spacing of 240 kHz and 15 subsymbols ($M = 15$). The filter is a root raised cosine with roll off factor 0.3 and the duration of the cyclic prefix is 32 samples. The performance of the iterative receiver proposed in this section is compared in terms of the SER with a ZF receiver (we use Algorithm 5.1 to estimate the GFDM modulated symbols followed by a ZF filter to demodulate the data symbols). Moreover, we assume a perfect synchronization and no coding. We assume pilot symbols that are distributed on equidistant positions denoted by ΔF . The ALS algorithm is stopped if the error of the cost function is smaller than 10^{-4} , if the error difference in two consecutive iterations is smaller than 10^{-4} , or the number of iterations exceeds five.

First, we compare the proposed ALS receiver and the ZF receiver for different amounts of pilot symbols $\Delta F = 20$ and $\Delta F = 40$. The results are presented in Fig. 5.19. The iterative receiver outperforms the ZF receiver. As expected, a larger amount of pilot symbols leads to a more accurate pilot based channel estimate and therefore, to a lower SER. Moreover, the gain of the additional iterations becomes more pronounced with an increase of the SNR and if the initial pilot based channel estimate is less accurate.

Next, we compare the ALS receiver and the ZF receiver for different modulation orders. In Fig. 5.20, we depict the SERs for GFDM systems using 4-QAM and 16-QAM to modulate the data symbols. The total transmit power in both cases is equal. Hence, the SER is higher if we modulate the data symbols using 16-QAM because the minimum Euclidean distance between the symbols is smaller.

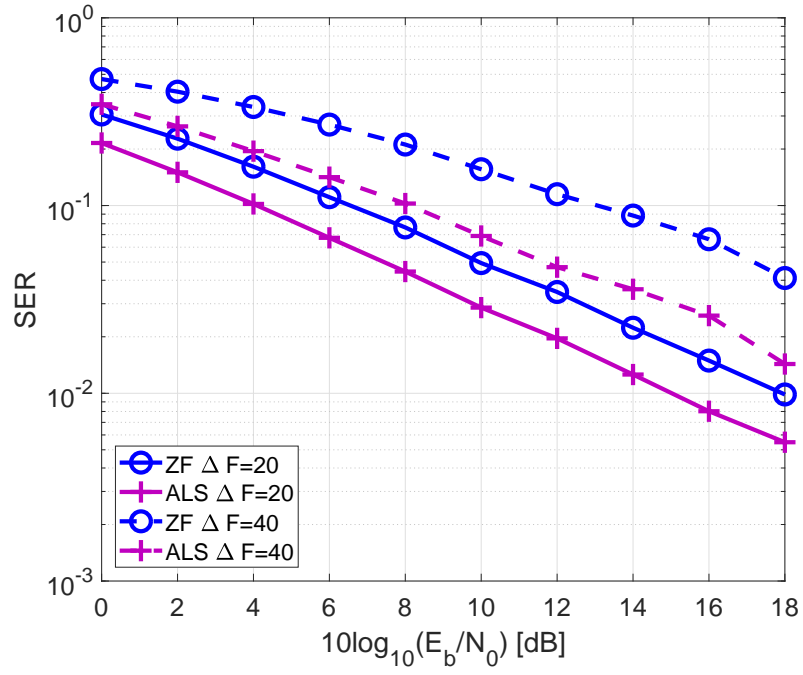


Figure 5.19.: Comparison of the SER for different pilot positions spacings in the frequency domain.

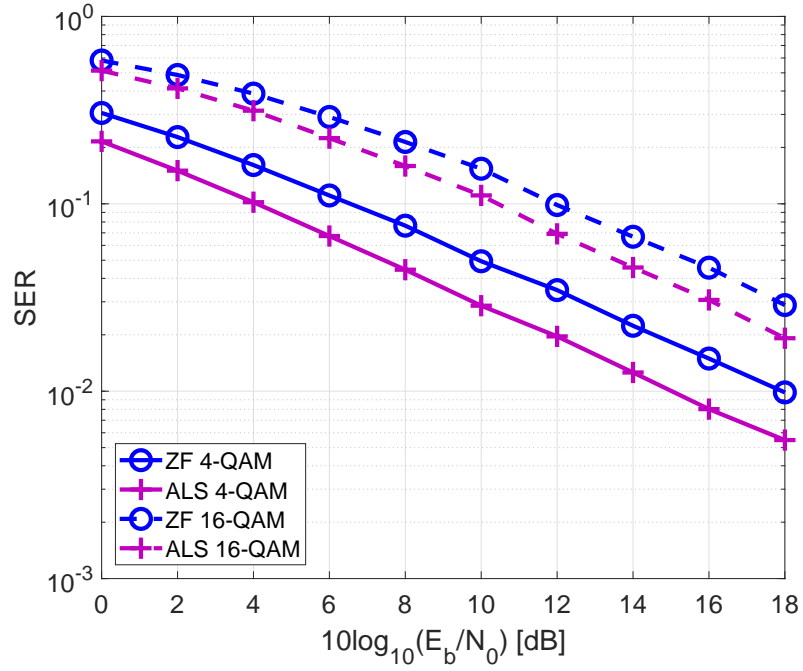


Figure 5.20.: Comparison of the SER for different modulation order.

5.2.2 Summary

In this section, we have first shown that the GFDM transmit signal can also be modeled based on the generalized tensor contraction. Thereby, we show that the tensor contraction is a very practical

operator. Using this model of the transmit signal, we first provide more insides on the structure of the GFDM modulation matrix. Moreover, we show that the received signal tensor for MIMO-GFDM can also be described by a generalized tensor contraction between the transmit signal and the channel tensor in the frequency domain. The extension of our model to a system with several frames is straightforward and leads to a higher tensor gain. Moreover, our model provides new opportunities for the GFDM filter bank, such as finding the best pilot sequences while studying the structure of the channel matrix under the assumption that the subcarriers are not orthogonal, investigating more general GFDM systems when not all subcarriers or subsymbols are used for data transmission and investigating new closed form solutions by adding coding or random coding. Furthermore, we have presented a simple iterative ALS receiver. By comparison with a ZF receiver, we show that the proposed iterative receiver is able to estimate the channel impulse response and the data symbols within only a few iterations. The fast convergences of the algorithm is due to the projection of the estimated symbols onto the finite alphabet that depends on the modulation type and the modulation order.

5.3 Conclusions

In this chapter, we have first presented a tensor model for MIMO-OFDM systems using the generalized tensor contraction operator between a channel tensor and a transmit signal tensor. This model is a very general and flexible way of describing the received signal in MIMO-OFDM systems for all subcarriers jointly. We have also proposed a new representation of the channel tensor using a 4-way tensor with a special BTD structure. The resulting tensor model of the received signal enables the design of the traditional ZF receiver and facilitates the design of two iterative LS receivers based on projections and enumeration, respectively. Moreover, we have also proposed recursive versions of the two iterative receivers ILSP and ILSE denoted by RLSP and RLSE, respectively. The algorithms based on enumeration outperform the rest of the algorithms at the cost of an increased complexity. The RLSE algorithm is suitable for estimating the channel and the symbols for any configuration setup without additional coding. The algorithms based on projections have a better performance than the ZF receiver if the number of transmitted frames is large enough. Moreover, the accuracy of the ILSP algorithm depends on the rank of the transmitted symbol matrices. Therefore, its performance in terms of the SER depends on the chosen modulation order and the modulation scheme. Hence, the system can be modified such that only specific code words are used. Moreover, the recursive algorithms can be modified such that they exploit the channel correlation in time varying scenarios. Note that we already exploit the correlation of the channel among adjacent subcarriers that leads to a reduced number of pilot symbols as compared to other tensor models. The aforementioned model and results have been published in [NHdA18].

Next, we have used Khatri-Rao coding for the transmission of the OFDM symbols leading to Khatri-Rao coded MIMO-OFDM systems. The generalized tensor model using the contraction operator has been extended to the Khatri-Rao coded MIMO-OFDM system in a straightforward

way. In this case, the transmit signal tensor has a CP structure. By exploiting the overall tensor model, we propose a receiver based on the LSKRF. This receiver requires the same amount of training symbols as traditional OFDM techniques, but it has an improved performance in terms of the SER. Hence, we benefit from the additional tensor structure of the transmitted signal to achieve a tensor gain. In addition, we propose to even more improve the performance of this receiver by means of an additional LS iteration. In the future, we can consider not just one additional LS iteration, but several iterations leading to an ALS based receiver initialized using the LSKRF. We should also consider the design of optimal orthogonal pilot sequences specific to the KR receiver. The tensor model and the proposed receiver for Khatri-Rao coded MIMO-OFDM systems have been published in [NHdA17]. Note that the Khatri-Rao coding strategy has a reduced spectral efficiency than the uncoded MIMO-OFDM system. Therefore, we propose an alternative transmission technique where the "coding matrices" contain random data symbols. Thereby, this transmission technique also imposes a CP structure on the transmit signal tensor. Using the resulting received signal tensor, we propose two receivers for randomly coded MIMO-OFDM systems. The first proposed receiver RC-KR estimates the symbol matrices based on the LSKRF. The second proposed RC-KR+ALS receiver is an ALS algorithm initialized with the estimates of the symbol matrices using the RC-KR receiver. The proposed RC-KR+ALS algorithm outperforms the iterative receivers for MIMO-OFDM because it exploits the additional tensor structure of the signal tensor. Unlike the receivers for Khatri-Rao coded MIMO-OFDM, both receivers for the randomly coded MIMO-OFDM assume that $M_R \geq M_T$. In the future, we can consider a recursive LS instead of LS in order to relax this condition. However, the randomly coded system has a higher spectral efficiency than the Khatri-Rao coded system. In the future, the system can be modified such that both symbol matrices contain symbols from different constellations and/or different modulation orders. This will lead to a resulting transmit signal tensor with diverse entries and potentially improved performance for the receivers in terms of the SER. We should investigate which combinations of modulation orders and modulation types are suitable for different SNRs.

Moreover, in this chapter, we have shown that our general model using tensor contraction for MIMO-OFDM systems can be extended to MIMO-GFDM systems in a straightforward fashion. Thus, we have shown the flexibility and importance of this model. In the case of MIMO-GFDM systems, the transmit signal tensor can also be expressed in terms of the generalized tensor contraction. Therefore, we also use the properties of the tensor contraction operator to provide more insights into the structure of the GFDM modulation matrix. Based on the overall received signal, we have proposed an ALS receiver for MIMO-GFDM systems. Note that for the MIMO-GFDM system, in this chapter we have assumed only one frame ($K = 1$). However, the extension to multiple frames is straightforward and leads to an additional tensor gain. Using the proposed model, we can study the structure of the resulting channel tensor for MIMO-GFDM systems, where the subcarriers are not orthogonal. Moreover, our model provides new opportunities for the GFDM systems, such as finding the best pilot sequences, investigating more general GFDM systems when not all carriers or subsymbols are used for data transmission, and investigating new closed form solutions for the

receiver by adding coding or random coding. The presented iterative ALS receiver has already been published in [NCH⁺17]. However, we have derived the ALS receiver in [NCH⁺17] based on the PARATUCK2 decomposition. In contrast to the derivation based on the PARATUCK2 decomposition in [NCH⁺17], the solution presented in this thesis based on the generalized tensor contraction is more elegant, shorter, and practical, as well as it provides more physical insights.

In the future, the general tensor model for multi-carrier systems proposed in this chapter can be extended to other multi-carrier techniques such as UFMC and FBMC. Even more, this model can be straightforwardly extended to relay-assisted systems and multi-user systems.

Chapter 6

Application of Tensor Algebra to Biomedical Signal Processing

EEG (ElectroEncephaloGraphy) and MEG (MagnetoEncephaloGraphy) are methods for recording the brain activity. We use these methods to investigate the function and organization of the human brain. They, respectively, measure the magnetic flux and the electric potential at the head surface. The EEG and MEG signals are typically multi-dimensional, and their dimensions correspond to time, space (channels), participant, and experimental condition. Moreover, EEG and MEG signals used in biomedical studies are typically acquired simultaneously. Thus, these signals simultaneously capture aspects of the same electric activity and therefore can be coupled.

The multi-dimensional signals can be decomposed into rank one components according to the CP decomposition [KB09]. The authors of [RH08] and [RH13a] propose the SECSI framework for the efficient computation of the approximated CP decomposition (see Section 3.1). Moreover, many combined signal processing applications benefit from a coupled analysis based on the coupled CP decomposition [SDL17a, ZCJW17, BCA12, ARS⁺13, ABS15, RDGD⁺15]. The coupled CP decomposition jointly decomposes heterogeneous tensors that have at least one factor matrix in common. A detailed analysis of the computation of the coupled CP decomposition based on ALS is presented in [FCC16, CFC16], where it is shown that tensors have to be normalized before the computation of the coupled CP decomposition based on ALS if the coupled tensors contain noise with different noise variances. We propose the coupled SECSI (C-SECSI) framework in Section 3.6 and in [NH16] as an extension of the SECSI framework [RH08, RH13a] for the computation of the coupled CP decomposition. The C-SECSI framework efficiently approximates the coupled CP decomposition of two noisy tensors that have at least one mode in common even in ill-posed scenarios, e.g., if the columns of the factor matrices are highly correlated. Moreover, the C-SECSI framework offers adjustable complexity-accuracy trade-offs and efficiently decomposes tensors with different noise variances without performance degradation. Furthermore, in Section 3.6 and in [NKHH17] a reliability measure for the C-SECSI framework is proposed. This reliability test allows us to control the rank of the coupled CP decomposition.

In this chapter, we present two applications of the C-SECSI framework proposed in Section 3.6 for the joint analysis of EEG and MEG signals. In the first section of this chapter, we provide a validation of the photic driving effect using C-SECSI [NKHH17]. In the second section, we present a validation of a controlled experiment based on a joint EEG-MEG signal decomposition in order

to show the effects of skull defects on the measurement signals [NLA⁺17].

6.1 Validation of the photic driving effect based on the coupled CP decomposition

In the past, it has been shown that the human brain activity contains several typical frequencies (oscillators) [Wol99, Her01]. These frequencies include the alpha frequency, the beta frequency, the theta frequency, and the gamma frequency. The alpha frequency is typically in the range between 7.5 Hz and 12.5 Hz, the beta frequency is in the range between 12.5 Hz and 30 Hz, the theta frequency is in the range between 4 Hz and 7.5 Hz, and the gamma frequency is in the range between 30 Hz and 40 Hz [Wol99, Her01]. However, these ranges/bands are not strict as they vary from individual to individual. Moreover, the author of [Wol99] shows that the power and synchronization of some of these frequencies varies with the state of the individual volunteer. For instance, the awakesness state of the volunteer, the age of the volunteer, the health of the volunteer, and other environmental factors.

An IPS (Intermittent Photic Stimulation) is a stimulation of the brain by repetitive light flashes that can induce oscillations in the human brain. This effect is called the PD (Photic Driving) effect. IPS can cause a frequency entrainment that is indicated by the synchronization of the individual brain rhythm with the photic stimulation frequency. The PD effect is widely used to assess effects of medicaments and for diagnosis. The studies of the PD effect provide evidence for the frequency selectivity of the neural oscillator network in the human brain [dS91, NKH16]. The authors of [LPdA09] use the PD effect for the investigation of neurophysiological mechanisms underlying autistic symptoms. Moreover, in [KPVldS02], the PD effect of epileptic patients is investigated on the basis of simultaneously recorded EEG and MEG signals. The first investigation of frequency entrainment using simultaneously recorded EEG and MEG signals during the IPS with frequency, which is adapted to the individual alpha rhythm is performed in [SLJ⁺06]. Furthermore, the authors of [SSK⁺16] analyze a rod-driven PD effect and show that strong alpha resonance phenomena exist for a rodinput at stimulation frequencies around the individual alpha rhythm and the first subharmonic (note that the first subharmonic corresponds to a frequency in the theta band, i.e., theta rhythm). Moreover, based on a spectral analysis, the authors of [SLJ⁺06] show an entrainment of the alpha frequency. Furthermore, the frequency entrainment is analyzed in [SSK⁺16] using both time analysis and frequency analysis. Also, a combined time-frequency analysis is utilized in [WGP⁺11] to study the frequency entrainment.

In this section, we consider the same study as in [SSK⁺16]. Here, we present a combined time-frequency-channel analysis by utilizing tensor decompositions. We evaluate numerically and visually all resulting components for each volunteer and stimulation frequency.

6.1.1 Measured EEG-MEG Signals

In order to investigate the synchronization effect of the alpha rhythm, an experiment has been conducted on twelve healthy volunteers at the Biomagnetic Center of the University Hospital in Jena, Germany. The study has been approved by the Ethics Committee of the Faculty of Medicine of the Friedrich-Schiller-University Jena. Within this work we number the different volunteers from 1 to 12, for distinguishing reasons. In the course of the experiment, the volunteers have been exposed to a flickering light. The stimulus has been transmitted using optical fibers from light emitting diodes outside the recording room. Light diffusers at approximately 10 cm in front of the volunteer's eyes have provided a luminance of 0.0003 cd/m^2 . Throughout the exposure, the eyes of the volunteers have been closed. Their brain activity has been recorded simultaneously with EEG and MEG. The EEG and MEG signals have been measured using 128 channels and 306 channels, respectively.

In the first step of the experiment, the individual alpha rhythm has been measured during 60 s of resting MEG. The individual alpha frequencies (f_α) have been then calculated by means of the averaged Fourier transform from the occipital MEG channels. The resulting alpha frequencies in Hz for each volunteer are listed in Table 6.1.

Volunteer	1	2	3	4	5	6	7	8	9	10	11	12
f_α	9.6	10.7	10.4	10.8	10.7	10.8	7.5	10.8	11.0	10.7	12.2	10.3

Table 6.1.: Individual alpha frequency f_α in Hz for each volunteer.

As an IPS, twenty different stimulation frequencies with irregular step size have been used, i.e., the f_s frequencies listed in Table 6.2. Each stimulation frequency has been performed in 30 stimulation trains, each consisting of 40 periods with a pulse/cycle duration of 0.5. Therefore, the measured signals contain only odd harmonics. Between each train there has been a resting period of 3 s. From one frequency block to the next one, there has been a resting period of 30 seconds. To avoid an ordering effect, the sequential arrangement of the stimulation frequencies has been chosen randomly. Further details regarding the experiment are available in [SSK⁺16]. Moreover,

Condition	1	2	3	4	5	6	7	8	9	10
f_α ratio	0.40	0.45	0.50	0.55	0.60	0.70	0.80	0.90	0.95	1.00
Condition	11	12	13	14	15	16	17	18	19	20
f_α ratio	1.05	1.10	1.30	1.60	1.90	1.95	2.00	2.05	2.10	2.30

Table 6.2.: Stimulation frequencies in f_α rations for the particular conditions.

the authors of [SSK⁺16] report no evidence of entrainment for stimulation frequencies larger than $1.30 \cdot f_\alpha$. Therefore, we analyze the measurement results only for the first thirteen stimulation frequencies, i.e., from $0.40 \cdot f_\alpha$ until $1.30 \cdot f_\alpha$.

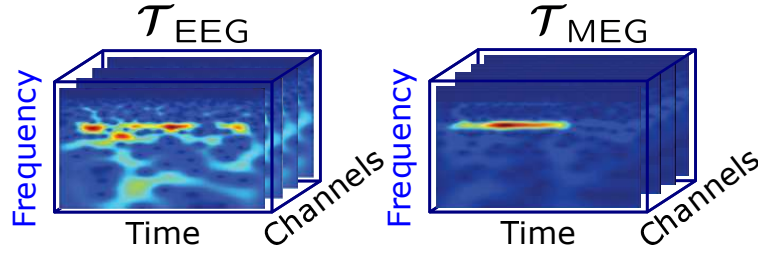


Figure 6.1.: Visualization of the EEG and MEG tensors per volunteer and stimulation frequency

6.1.2 Joint EEG-MEG Signal Decomposition

In our analysis, we first average the MEG and EEG signals for each stimulation frequency and we exclude all faulty EEG and MEG channels. In addition to the usual preprocessing, a complex Morlet wavelet decomposition is used to obtain an estimate of the time-frequency distribution of the EEG and MEG signals for each channel and stimulation frequency. The wavelet coefficients between 3.77 Hz and 15.15 Hz are selected for the further analysis, thereby including only the alpha band and theta band in this analysis. The complex wavelet coefficients for each of the EEG and MEG channels are then arranged as frontal slices in 3-way tensors as depicted in Fig. 6.1. As a result, we have different complex-valued tensor with dimensions frequency \times time \times channels for each stimulation frequency, measurement type (EEG or MEG), and volunteer. The frequency mode and the time mode correspond to the discretized values resulting from the wavelet transform. Therefore, the frequency mode contains two hundred discrete frequency values from 3.77 Hz until 15.15 Hz. The time mode, however, varies from around 5000 ms up to 20000 ms depending on the corresponding stimulation frequency. Furthermore, the channels mode corresponds to the numbers of EEG and the MEG channels, which can also vary from volunteer to volunteer and condition. This is because as previously mentioned, in this analysis taken into account are only the channels that do not contain artifacts and are perfectly intact, meaning that the sensors corresponding to those channels had have perfect connection during the measurement (i.e., the faulty channels are excluded from this analysis).

Next, the EEG and MEG signal tensors are jointly analyzed using the C-SECSI framework, for each volunteer and stimulation frequency, respectively. The coupled CP decomposition has been originally computed for different ranks. However, the reliability and residual have indicated that the tensor rank is overestimated for values equal to or larger than three [NKHH17]. Therefore, in this thesis, we present the result for tensor rank $\hat{R} = 2$. Moreover, it is assumed that the frequency mode is common for both the EEG and MEG signal tensors. Before the computation of the coupled CP decomposition, each of the tensors is normalized to norm one, i.e.,

$$\frac{\mathcal{T}_{\text{EEG}}}{\|\mathcal{T}_{\text{EEG}}\|_{\text{H}}} \quad \text{and} \quad \frac{\mathcal{T}_{\text{MEG}}}{\|\mathcal{T}_{\text{MEG}}\|_{\text{H}}}.$$

This normalization of the tensors is due to the different amplitude scales and units of the EEG and MEG tensors values (fT and μV).

6.1.3 Analysis

The joint EEG-MEG signal decomposition based on the coupled CP decomposition using the C-SECSI framework (see Section 3.6) results in three factor matrices for the EEG and three factor matrices for the MEG signal tensor (see equations (3.30) and (3.31)), per volunteer and per stimulation frequency. Each factor matrix, consist of two columns corresponding to the two components, due to the assumed rank $\hat{R} = 2$.

The resulting components are then visually and numerically analyzed. The visual analysis corresponds to the analysis of the field-maps in order to identify the recordings and components that contain a successfully evoked response of the visual cortex. The measured MEG and EEG signals are converted to field-maps by calculating the RMS (Root Mean Square) of the values for each channel. All field-maps are accordingly labeled using three categories namely (1) if they contain only clear visual response patterns, (2) contain visual response patterns and some other activity, or (3) contain no visual response patterns. The labeling of the field-maps has been performed by three experienced professionals (Stephan Lau (SL), Uwe Graichen (UG) and Daniel Strohmeier (DS)). Their individual labeling results have been cross-referenced and unified by a majority vote. This labeling of the field-maps assures that the obtained components and frequencies are not artificial. Also, the labeling confirms a visual response of the photic driving effect in the occipital region in every data set as compared to other state-of-the-art studies where often this has not been done.

The numerical analysis includes search of the maxima in the frequency components, comparing the weights of the components, and computing the reliability and the residual. To the maxima of the frequency signatures we refer to as the obtained frequencies. Selected for further analysis, however, are only frequencies which field-map show only clear visual response (category (1)). These frequencies are then further rearranged into two groups. The first group represents the recruited frequencies, whose frequency is very close to the stimulation frequency and/or its second harmonic with a maximum difference of $\pm 0.05f_s$ and $\pm 0.05 \cdot 2 \cdot f_s$, respectively. Similar to our work, the higher order harmonics are also analyzed and confirmed in [Her01]. The second group, i.e., the not-recruited group contains all remaining obtained frequencies. Moreover, all frequencies are expressed in fractions of the individual alpha frequency of the volunteer to account for the inter-individual differences. Furthermore, the weights represent the power of each component. The reliability is defined in (3.45) as a similarity measure of the final estimates of the common factor matrices. To this end, the residual is computed according to

$$\text{RES} = \frac{\|\hat{\mathcal{X}} - \mathcal{X}\|_{\text{H}}^2}{\|\mathcal{X}\|_{\text{H}}^2}, \quad (6.1)$$

where \mathcal{X} is the noisy tensor to be decomposed using the C-SECSI framework, and $\hat{\mathcal{X}}$ is the reconstructed tensor after the decomposition. Details for the computation of the coupled CP decomposition via C-SECSI, the reliability, and the residual are provided in Section 3.6.

6.1.4 Results

Statistics from the labeling of the field-maps

The labeling of the visual response topographies in the RMS field-maps show that in 89.4% of the measured EEG signals and in 73.5% of the measured MEG signals some visual response of the photic driving is present (i.e., category(2)). After the decomposition, in at least one of the components there is some visual response in 96.03% and 72.85%, for EEG and MEG, respectively. Clear photic driving visual response (i.e., category(1)) is observed in 41.72% of the MEG signals before and 62.91% after the decomposition. Clear photic driving visual response (i.e., category(1)) is observed in 72.18% of the EEG signals before and 84.77% after the decomposition. This shows the benefit of using tensor decomposition for extracting the signal components, rather than analyzing directly the measured signals. Moreover, from the 84.77% of the EEG results with visual response, after the decomposition 49% correspond to the recruited frequencies, and from the 62.91% of the MEG results with visual response, after the decomposition 31.79% correspond to the recruited frequencies. Altogether visual pattern is more often present in the EEG signals than in the MEG signals. However, a larger percentage of the MEG frequencies with a visual pattern are recruited (31.79% of 62.91%) as compared to EEG (49% of 84.77%).

Visualization of the estimated components

Recall that the joint EEG-MEG signal decomposition based on the coupled CP decomposition results in three factor matrices for the EEG signal tensor and three factor matrices for the MEG signal tensor (c.f. equations (3.30) and (3.31)), per volunteer and per stimulation frequency. Each factor matrix consists of two columns corresponding to the two components.

In Fig. 6.2, we visualize the estimated factor matrices from the coupled CP decomposition for volunteer 1 and stimulation frequency $1.1f_\alpha$. Additionally to the estimated components, in Fig. 6.2 depicted are the field-maps of the RMS of the measured signal values per channel for both MEG and EEG signals (first column). These RMS field-maps represent the power distribution of the measured signals before the decompositions. Next, depicted are the field-maps for the channel signatures for component one and two, respectively (second and third column). The frequency and the time signatures are depicted in Fig. 6.2 as a function of the frequency in Hz and as a function of the time in s, respectively. The topographic visualization of the channel signatures for both components depicts the occipital area where the photic driving effect is expected to occur. Comparing the resulting frequency signatures for EEG and MEG, it is obvious that they are not identical. Note that it has been assumed that the frequency mode is the common mode. In this example, with the C-SECSI framework we are able to extract two underlying components. Both components occurred in the occipital area and are represented by two different frequencies. The component 2 is common for the EEG and the MEG signals. On the other hand, component 1 has a different but similar frequency (not-recruited) that is closer to the individual alpha frequency.

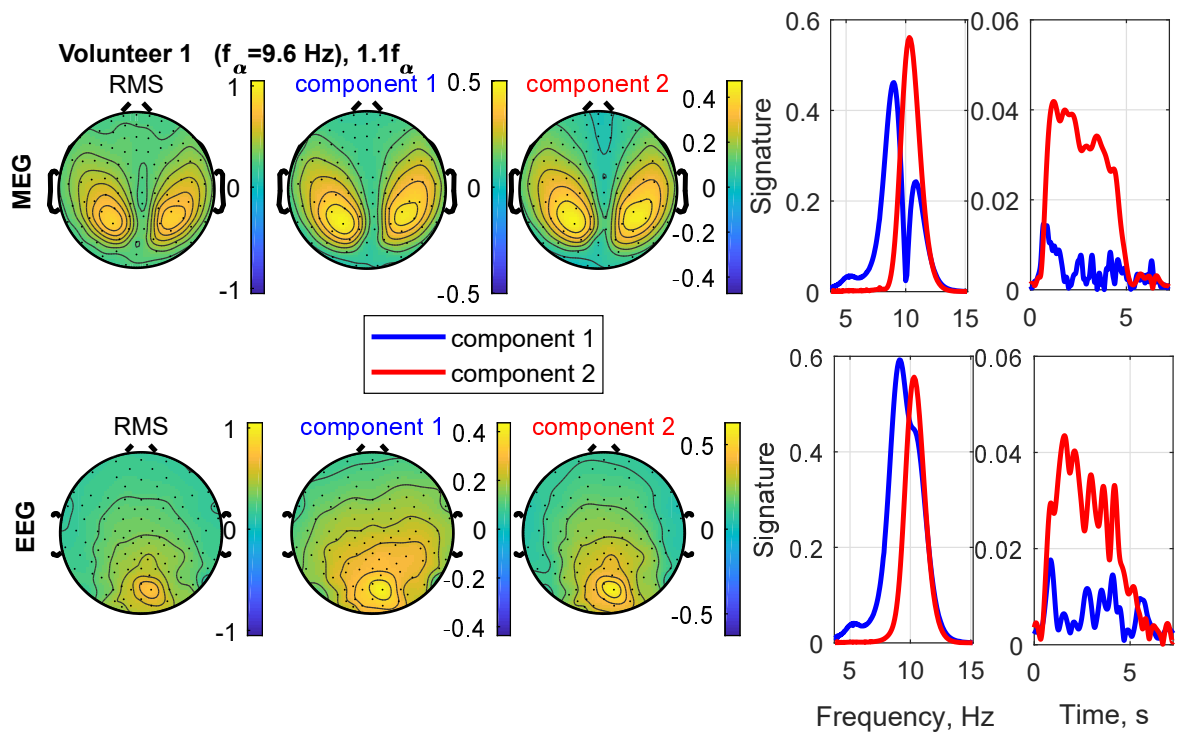


Figure 6.2.: RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $1.1f_\alpha$.

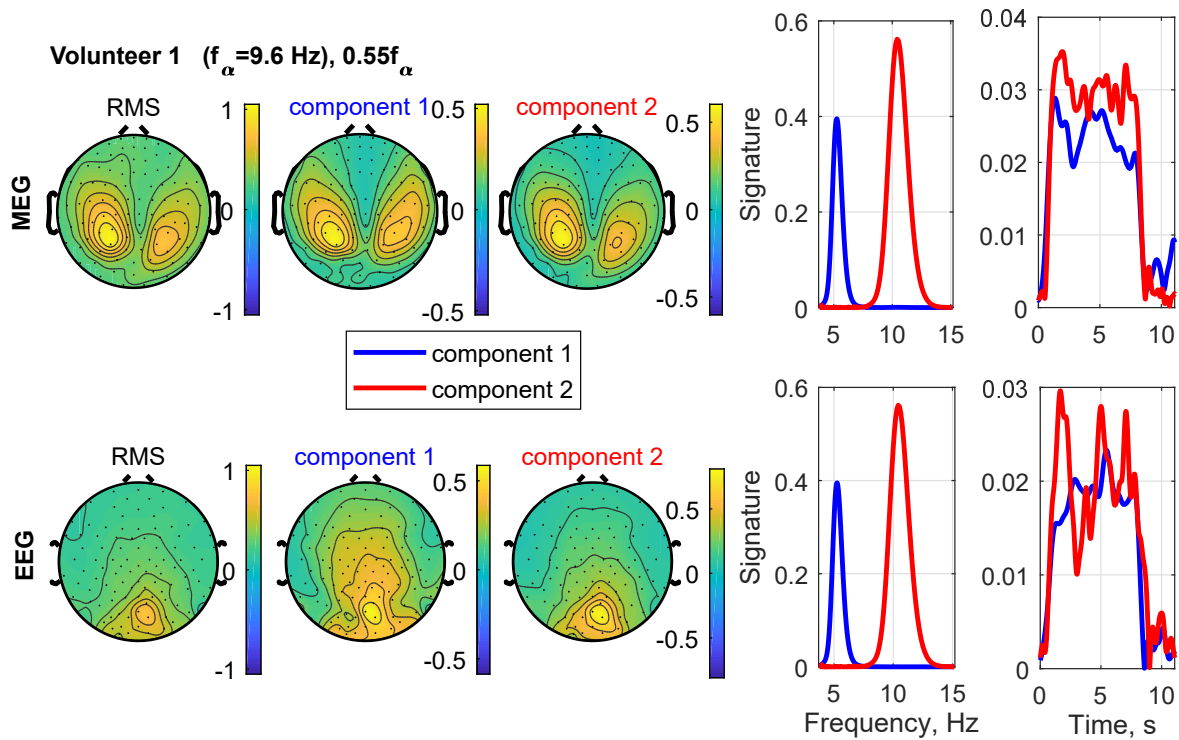


Figure 6.3.: RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $0.55f_\alpha$.

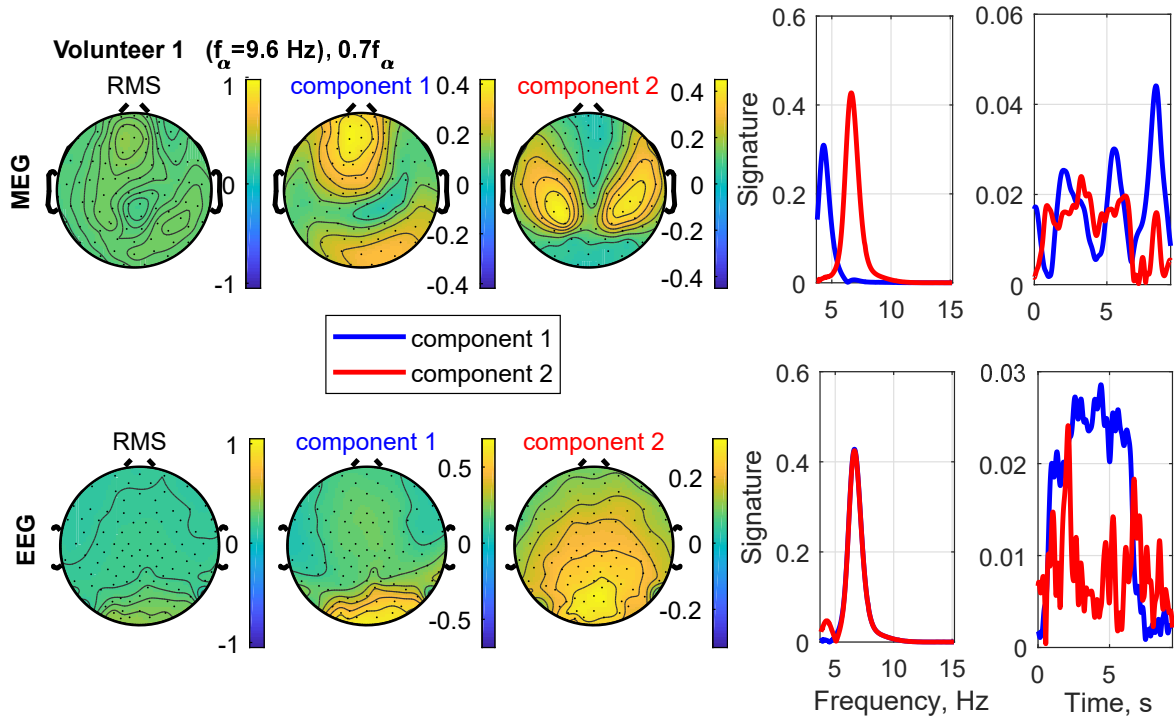


Figure 6.4.: RMS, channel, frequency, and time signatures for volunteer 1, and stimulation frequency $0.7f_{\alpha}$.

Similar, the RMS, channel, frequency, and time signatures for volunteer 1 and stimulation frequency $0.55f_{\alpha}$ are depicted in Fig. 6.3. Note that this stimulation frequency is in the range of frequencies corresponding to the theta band. The two components show visual response in the occipital region, and both components have comparable amplitudes over the time course. Here, extracted are two underlying components, and they are represented by two different frequencies, around $0.5f_{\alpha}$ and f_{α} (i.e., theta frequency and alpha frequency).

Moreover, the RMS, channel, frequency, and time signatures for volunteer 1 and stimulation frequency $0.7f_{\alpha}$ are depicted in Fig. 6.4. In contrast to the previous two examples, the field-map of the measured EEG signal contains only clear visual response pattern, but has a lower power. Moreover, the measured MEG signal contains no visual response pattern. Despite this, the tensor decomposition is able to extract two components of which only component 2 is shared between the EEG and the MEG signals. The MEG component 1, however, does not contain any visual response. The EEG component 2 contains visual response and some other activity. Both resulting EEG components are corresponding to the theta band.

Analysis of the obtained frequencies

Moreover, we analyze the obtained frequencies (the maxima of the frequency signatures) corresponding to field-maps that contain clear visual response patterns (category (1)). As previously mentioned in Section 6.1.3, we differentiate between two groups of obtained frequencies, recruited

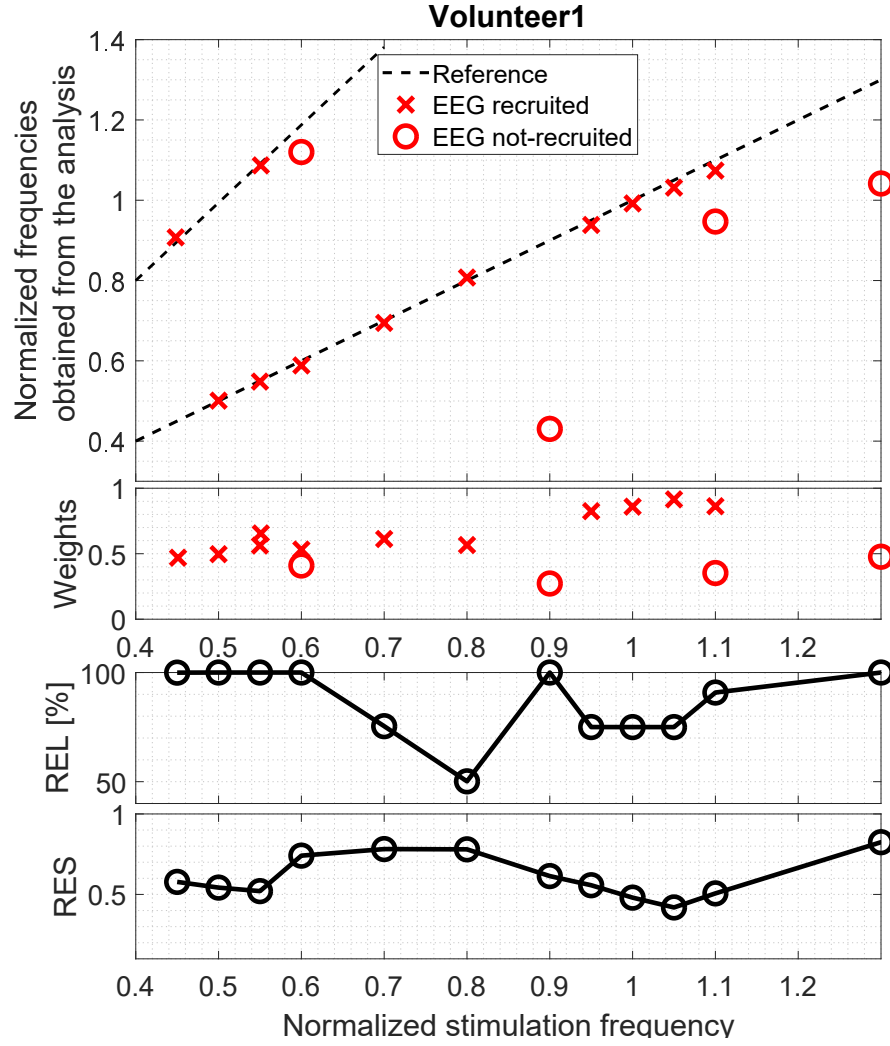


Figure 6.5.: Normalized frequencies obtained from the analysis, weights, reliability, and residual corresponding to the EEG signals.

and not-recruited frequencies.

The normalized frequencies obtained from the analysis, for EEG and MEG are visualized using scatter plots in Fig. 6.5 and Fig. 6.6, respectively. Figs. 6.5 and 6.6 correspond to volunteer 1. In addition to the scatter plots, in these figures, we depict the weights of the components, the reliability, and the residual for each stimulation frequency. Moreover, we depict two reference curves representing the obtained frequencies equal to the stimulation frequency f_s and $2f_s$. Figs. 6.5 and 6.6 show that the recruited frequencies are equivalent to the stimulation frequency. The not-recruited frequencies correspond to $0.5f_\alpha$ and f_α frequencies. Note that the frequency $0.5f_\alpha = 4.8$ Hz is in the theta band. The recruited frequencies are mainly more dominant than the not-recruited frequencies that can be seen from the weights of these recruited and not-recruited frequencies. For stimulation frequencies around $0.5f_\alpha$, the EEG and MEG signals have two components (one component in the theta band and another component in the alpha band) in common. On the other hand, for the rest

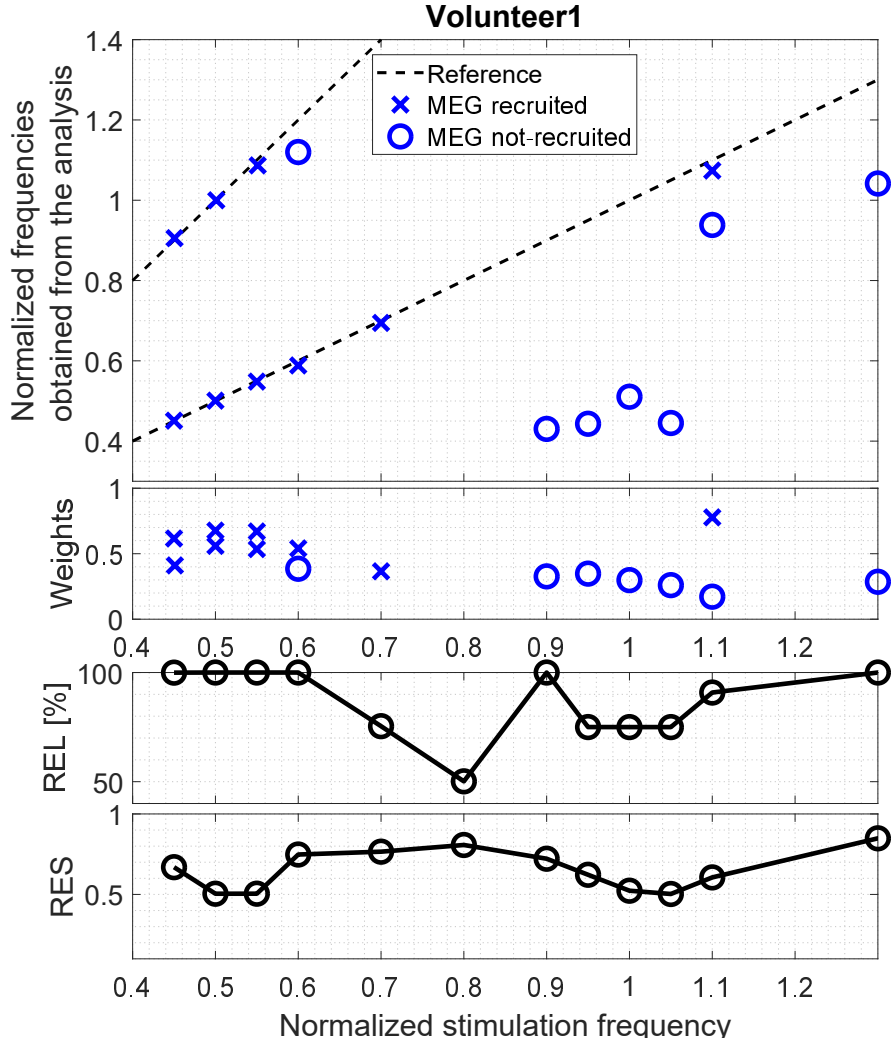


Figure 6.6.: Normalized frequencies obtained from the analysis, weights, reliability, and residual corresponding to the MEG signals.

of the stimulation frequencies, the signals are represented by only one component.

Next, we analyze the obtained frequencies jointly for all volunteers. The scatter plot of the normalized frequencies obtained from the EEG frequency signatures for all volunteers is depicted in Fig. 6.7. The scatter plot of the normalized frequencies obtained from the MEG frequency signatures for all volunteers is depicted in Fig. 6.8. In addition to the scatter plots, Figs. 6.7 and 6.8 contain violin plots of the weights of the components, violin plots of the reliability, and violin plots of the residual for each stimulation frequency. The violin plots are an alternative to the box plots, that depict the median and the estimate of the distribution of the available data [HN98]. In Figs. 6.7 and 6.8, we depict two reference curves representing the obtained frequencies equal to the stimulation frequency f_s and $2f_s$. The recruited EEG and MEG frequencies follow these reference curves. The median of the not-recruited frequencies, however, for stimulation frequencies between $0.4f_\alpha - 1.0f_\alpha$ is around $(0.4 \pm 0.1)f_\alpha$, and for stimulation frequencies between $1.0f_\alpha - 1.3f_\alpha$

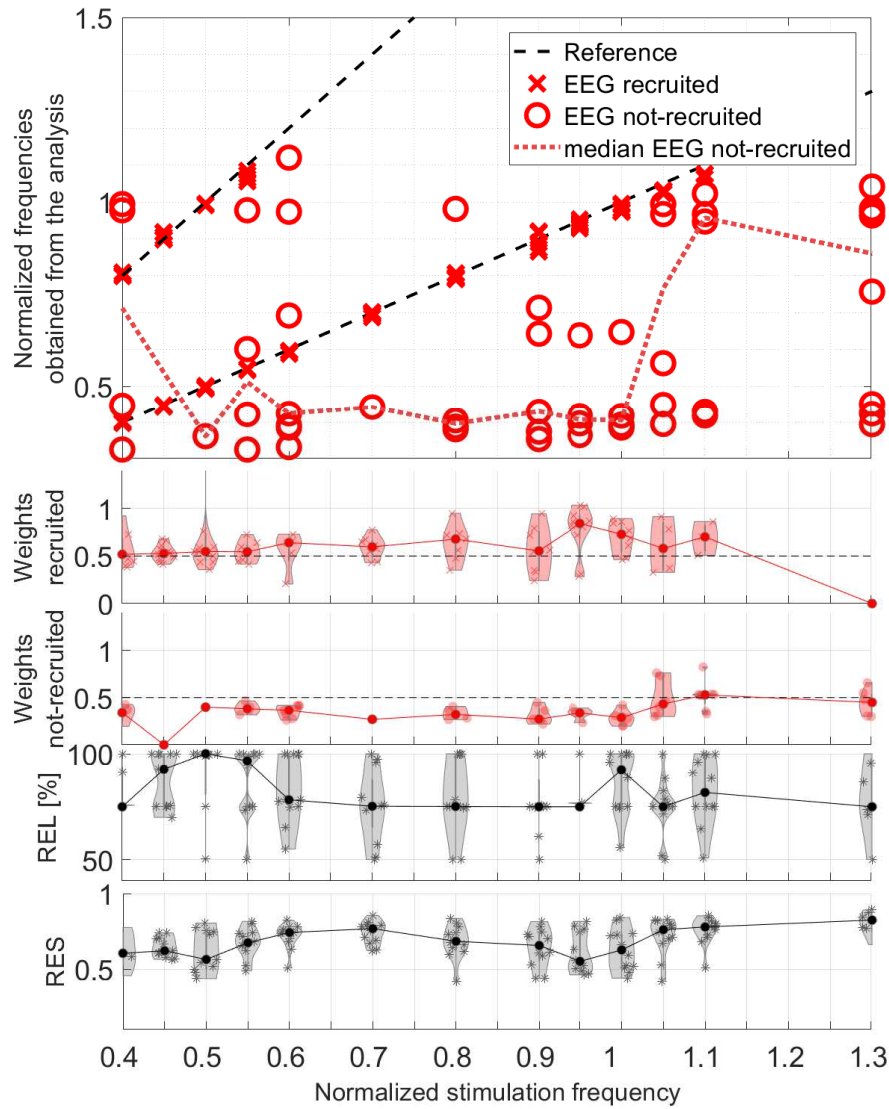


Figure 6.7.: Scatter plot of the normalized frequencies obtained from the analysis for EEG, violin plots of the weights of the recruited and not-recruited components, reliability, and residual as a function of the normalized stimulation frequency.

it increases towards $(1 \pm 0.1)f_\alpha$. The violin plots of the weights in Fig. 6.7 and Fig. 6.8 show that the recruited frequencies are mainly more dominant than the not-recruited frequencies. The reliability and the residual show that for stimulation frequencies around $0.5f_\alpha$ and f_α the EEG and MEG signals have two components in common. Note that if two components are active, one component has a frequency in the theta band and the other one has a frequency in the alpha band. For the rest of the stimulation frequencies, the EEG and MEG signals are represented by only one common component.

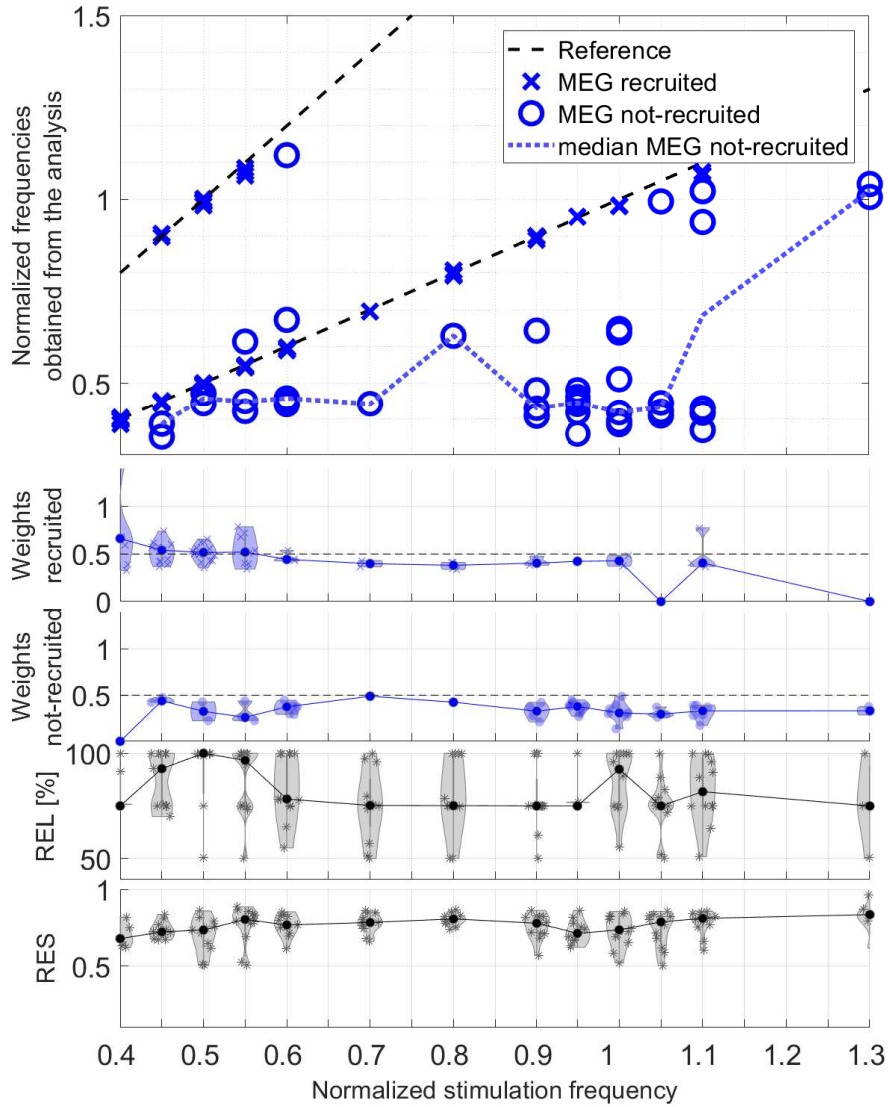


Figure 6.8.: Scatter plot of the normalized frequencies obtained from the analysis for MEG, violin plots of the weights of the recruited and not-recruited components, reliability, and residual as a function of the normalized stimulation frequency.

6.1.5 Discussion and Summary

Coupled CP decomposition and C-SECSI

We have applied a coupled CP decomposition on simultaneously recorded EEG-MEG signals for differentiating cortical oscillators during photic driving. The coupled CP decomposition allows us to decompose these multi-dimensional heterogeneous signals into their underlying components. The parallel underlying components are extracted while preserving the original multi-dimensional structure of the signals (frequency \times time \times channels, see Fig. 6.1) under the assumption that they have the frequency mode in common. This is possible because the frequency entrainment has similar behavior in the frequency domain for both EEG and MEG signals as shown in [SLJ⁺06, SSK⁺16,

NKHH17]. Moreover, the benefits of coupled decompositions have been shown in many other biomedical signal progressing applications such as [BCA12, ARS⁺13, PMS14, ABS15, RDGD⁺15, NKHH17, NLA⁺17, vEHDLvH17]. We use the C-SECSI framework to compute the coupled CP decomposition (see Section 3.6). The C-SECSI framework for 3-way tensors with one mode in common computes eight initial estimates, four of which are coupled solutions and four are uncoupled solutions. The final estimate is then chosen based on the minimum reconstruction error for both tensors independently. Therefore, C-SECSI computes the coupled CP decomposition under the constraint that one of the modes is coupled, but it still computes uncoupled estimates. This is very piratical for the analysis of biomedical data, were the coupling is assumed, but not yet proven. Moreover, for comparing the independently chosen final estimates we define a coupling reliability in equation (3.45). As shown in Section 3.6, the reliability can be used to control the rank of the coupled decomposition. We present similar results in [NKHH17]. The C-SECSI framework has a higher accuracy in ill-conditioned scenarios such as computing the coupled CP decomposition of tensors with collinear factors. Furthermore, another advantage of the C-SECSI framework is that it can decompose tensors corrupted by noise with different variances without any additional normalization or estimation of the SNRs.

Analysis of the signal components

In this section, we consider simultaneously recorded EEG-MEG signals resulting from a photic driving study. Details regarding the experiment are described in Section 6.1.1 and in [SSK⁺16]. In this experiment, as IPS are used high density frequencies, which are portions of the individual alpha frequency. The authors of [SLJ⁺06] and [SSK⁺16] show that frequency entrainment is more likely to occur if the stimulation frequency is a portion of the individual alpha frequency. A frequency domain analysis is used in [SLJ⁺06] in order to show the entrainment of the alpha frequency. The frequency entrainment is analyzed using both time and frequency analysis in [SSK⁺16]. Also, the authors of [WGP⁺11] show that a combined time-frequency analysis is important while studying the frequency entrainment. Here, we present a combined time-frequency-channel analysis by utilizing the coupled CP tensor decomposition. We study all of the components resulting from the decomposition, for twelve volunteers and 13 stimulation frequencies. By analyzing the frequency signatures, we observe that for some of the volunteers the estimated alpha frequency varies with ± 1 Hz as compared to the estimated one during the measured resting state before the stimulation. Resembling findings are reported in [SLJ⁺06] and [SSK⁺16]. Moreover, we observe that EEG and MEG signals do not always have two frequency components in common. For instance, for volunteer 1, stimulation frequencies $1.1f_\alpha$ and $0.7f_\alpha$ EEG and MEG have only one frequency component in common, whereas for stimulation frequency $0.55f_\alpha$ they have two components in common (one component corresponds to the theta band and another one to the alpha band). We also observe in [NKHH17] and [NLA⁺17] that EEG and MEG not always share the same components. In the estimated time signatures, we can mainly recognize three cases, one when both components have an obvious plateau, second, only one of the components has a plateau, and third, when there is an on-off switching between the

two components. A detailed analysis of the time signatures is not included in this work. However, a time domain investigation for frequency entrainment for the same experiment is presented in [SSK⁺16] and [WGP⁺11]. Here, we focus on the channel signatures and if they contain visual response pattern in the occipital area or not. Using tensor decomposition, we are able to extract meaningful components even if the RMS of the measured signal contains no clear visual response pattern. Such an example is illustrated in Fig. 6.4, where for volunteer 1 and stimulation frequency $0.7f_\alpha$, the measured EEG signal contains some visual pattern, but the MEG signal contains no visual pattern. The tensor decompositions, however, results in two meaningful components for EEG and one for MEG corresponding to the entrained frequency. The visual response pattern is studied for the purpose of photic driving effect at stimulation frequency f_α and the on-off-response for the stimulation with f_α in [SSK⁺16]. Despite that, it is not reported if visual response pattern is observed for all entrained frequencies. The authors of [HGS⁺11] perform a time-analysis of the stimuli and they as well have presented the field-maps for different stimulation frequency, but an entrainment is accounted even if there is no visual response. In this work, we take into account only the results for which a clear visual response pattern of the photic driving effect is present. The labeling of all field-maps has been performed by three experienced professionals (SL, UG, DS).

Analysis of the obtained frequencies

Further, we analyze which of the obtained frequencies are entrained and contain clear visual response pattern. We denote these frequencies as recruited frequencies (defined in section 6.1.3). The authors of [SLJ⁺06, WGP⁺11, HGS⁺11] and [SSK⁺16] show that the entrainment is more effective in the MEG measurement rather the EEG measurement as it is in our case. Nonetheless, it should not be neglected that our results show that a visual pattern of the phonic driving effect is existent in considerably less cases of MEG data rather than EEG. We have found entrainment in frequencies equal to the stimulation frequency f_s and its harmonic $2f_s$. Higher harmonics of the entrainment are also reported in [Her01]. Moreover, we have found no entrainment after $1.1f_\alpha$ same as in [SSK⁺16], where it is argued that above 15 Hz, photic stimuli cannot be differentiated any more due to slow processing. Note that the preprocessing of the measurement signals included filtering with a Butterworth band-pass filter with bandwidth 2-30 Hz. With that an entrainment of higher harmonics or gamma oscillators cannot be expected in our results. In addition to the recruited frequencies at the stimulation frequencies and their harmonic, we have observed a presence of a theta and/or alpha frequency as a second not-recruited component. The presence of the alpha frequency when stimulating with other frequencies (not the alpha frequency) is also reported in [Her01]. The presence of an additional pick around the theta frequency band for significant number of volunteers is reported in [LSSd01]. The authors of [KDRP96] and [Wol99] observe a coexistent correlation between the alpha frequency and the theta frequency. Moreover, the alpha frequency and theta frequency interchangeably have more power depending on the age or awakened status of the volunteers [Wol99]. However, to the best of our knowledge the presence of the theta oscillator has been determined as an additional component during photic driving only in [MPCB93].

The coexistence of several components corresponding to the different oscillators, theta, alpha and gamma is shown in [MPCB93] using principal component analysis. In our analysis, we use rank two decomposition. Therefore, we can show a coexistence of two oscillators (alpha and theta). The theta or alpha oscillators are present in all decomposition results when there is no entrainment, but visual response in the field-maps has either the recruited frequency or the not-recruited theta/alpha frequency. Moreover, our numerical analysis based on the reliably (REL) and the residual (RES) are in accordance with the labeling of the field-maps. Therefore, we propose that labeling of the field-maps should be considered in future analyses as a confirmation of the photic driving effect.

6.2 Validation of a controlled experiment based on Joint EEG-MEG signal decomposition using the coupled SECSI framework

Recall that EEG and MEG measure the electric potential and the magnetic flux density, respectively, at the head surface that is generated by the electric currents of neuronal activity inside the brain. The head tissues act as a volume conductor that influences the measured signals. MEG and EEG signals can be measured simultaneously and complement each other's information content. Therefore, a coupled analysis of EEG-MEG data using C-SECSI (see Section 3.6) could be of great benefit (see also Section 6.1). A conducting skull defect in the weakly conducting skull, e.g., after a surgery, is a volume conductor condition that can strongly influence MEG and EEG signals in characteristic ways [CGC79, LTS⁺10, LFH14, LGF⁺16]. Therefore, EEG-MEG recordings above skull defects are a suitable case for validating signal decomposition algorithms.

In this section, our objective is to decompose simultaneously recorded MEG and EEG signals above intact skull and above two conducting skull defects in a controlled experimental setup using C-SECSI in order to determine how skull defects are reflected in the tensor decomposition. The C-SECSI framework is presented in Section 3.6.

6.2.1 Measured EEG-MEG Signals

In a previous study [LFH14], a miniaturized electric dipole has been implanted in vivo into a rabbit's brain and connected to a 20 Hz sinusoidal constant-current source. Simultaneous recording using 64-channel EEG and 16-channel MEG has been conducted, first without defect (WOD) and then with two skull defects (WD). Skull defects have been filled with agar gels, which had been formulated to have a time-stable conductivity of approximately 1.0 S/m. The dipole has been moved under the skull defects, and measurements have been taken at regularly spaced points (i.e., dipole positions). The signals have been sampled using a sampling frequency of 1 kHz and band-pass filtered (15-25 Hz). Approximately 300 consecutive trials have been measured, which later have been averaged. All MEG and EEG recordings, respectively, have been co-registered and resampled to a common set of virtual channels. We arrange these measurements in two 4-way tensors \mathcal{T}_{MEG} and \mathcal{T}_{EEG} with dimensions channel \times time point \times defect state (WOD = 1, WD = 2) \times dipole position. A

visualization of the experimental setup from the study [LFH14] is presented in Fig. 6.9.

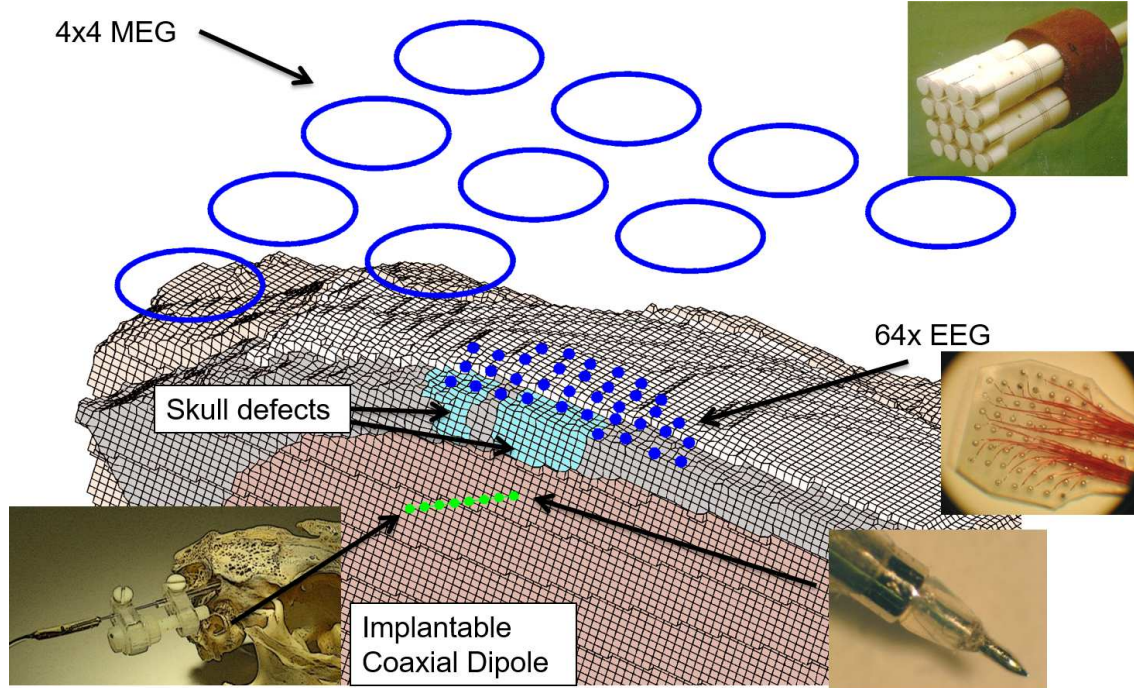


Figure 6.9.: Visualization of the experimental setup for the skull defect study in [LFH14].

6.2.2 Joint EEG-MEG Signal Decomposition

In a subsequent step, we construct the following tensors.

$$\begin{aligned} \mathcal{T}_{MEG,WOD} &= \mathcal{T}_{MEG(.,.,1,.)} & \mathcal{T}_{EEG,WOD} &= \mathcal{T}_{EEG(.,.,1,.)} \\ \mathcal{T}_{MEG,WD} &= \mathcal{T}_{MEG(.,.,2,.)} & \mathcal{T}_{EEG,WD} &= \mathcal{T}_{EEG(.,.,2,.)} \end{aligned}$$

All of the tensors are normalized to norm one, for instance $\mathcal{T}_{EEG,WOD}/\|\mathcal{T}_{EEG,WOD}\|_H$. The indices MEG and EEG represent the MEG and EEG signal tensors, respectively. The tensors $\mathcal{T}_{EEG,WOD}$ and $\mathcal{T}_{EEG,WD}$, represent the EEG signal tensors for all available positions, without skull defect (WOD) and with skull defect (WD), respectively. Accordingly, the tensors $\mathcal{T}_{MEG,WOD}$ and $\mathcal{T}_{MEG,WD}$, represent the MEG signal tensor without and with skull defect, respectively, for all available positions. Therefore, the tensors $\mathcal{T}_{EEG,WOD}$, $\mathcal{T}_{EEG,WD}$, $\mathcal{T}_{MEG,WOD}$, and $\mathcal{T}_{MEG,WD}$ are 3-way tensors with dimensions channel \times time point \times dipole position.

For all of the above defined tensors, a coupled EEG-MEG CP decomposition based on the C-SECSI framework (see Section 3.6) is computed for different tensor ranks, $R = 1, 2, 3$.

$$\begin{aligned} \mathcal{T}_{EEG} &= \mathcal{I}_{3,R} \times_1 \mathbf{F}_{1,EEG} \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_{3,EEG} \\ \mathcal{T}_{MEG} &= \mathcal{I}_{3,R} \times_1 \mathbf{F}_{1,MEG} \times_2 \mathbf{F}_2 \times_3 \mathbf{F}_{3,MEG} \end{aligned}$$

The coupled CP decomposition is computed for the two pairs (WOD and WD) of EEG-MEG signal tensors assuming that the time is the common mode. The EEG and MEG signals have been simultaneously recorded for the same input signal in the time domain. Therefore, the coupled EEG-MEG signal decomposition is justified.

6.2.3 Results

Rank and Residual

For the numerical evaluation, we use the metrics residual and reliability introduced in Section 3.6. The residual is calculated according to equation (6.1). The reliability in percentage is defined in equation (3.45). Small residual after fitting the coupled CP decomposition indicates that the given tensor is correctly approximate via the estimated components. The reliability indicates how similar are the estimated factor matrices corresponding to the common mode. The residual for each of the MEG and EEG signal estimates using the C-SECSI framework are depicted in Table 6.3. The residual decreases as the assumed rank increases indicating that the rank of the tensors is larger than one. Moreover, in Table 6.3 the reliability for each of the coupled CP decompositions is depicted. Based on these errors, the MEG and EEG signals with defects have one identical component in the time mode and additional not common components. The MEG and the EEG signal tensors without defects share one component in the time mode. Considering the results in Table 6.3, more components can be extracted for the MEG signal with skull defect then for the rest of the signals.

To present our results, we depict some of the factor matrices resulting from the tensor decompositions. In Fig. 6.10, we illustrate the components of the MEG signal without skull defect as a result of the coupled EEG-MEG signal decomposition for tensor rank $R = 1, 2, 3$. By analyzing the signature of the positions in this figure, we can see that for $R = 3$ each of the components is related to one of the positions, i.e., position 12, positions 14, and position 16. The components of the MEG signal with skull defect as a result of the coupled EEG-MEG signal decomposition for tensor rank $R = 1, 2, 3$ are depicted in Fig. 6.11. Based on the signature of the dipole position it

	$R = 1$	$R = 2$	$R = 3$
Residual for the MEG signal tensors			
MEG WD	0.029	0.007	0.001
MEG WOD	0.128	$8.5 \cdot 10^{-4}$	$3.1 \cdot 10^{-9}$
Residual for the EEG signal tensors			
EEG WD	0.065	0.0034	$6.9 \cdot 10^{-4}$
EEG WOD	0.335	0.004	0.013
Reliability of the coupled CP decomposition			
WD (EEG-MEG)	100 %	99.9994 %	99.9973 %
WOD (EEG-MEG)	100 %	92.5 %	90.3 %

Table 6.3.: Residual for the MEG and EEG signal tensors and reliability of the coupled CP decomposition.

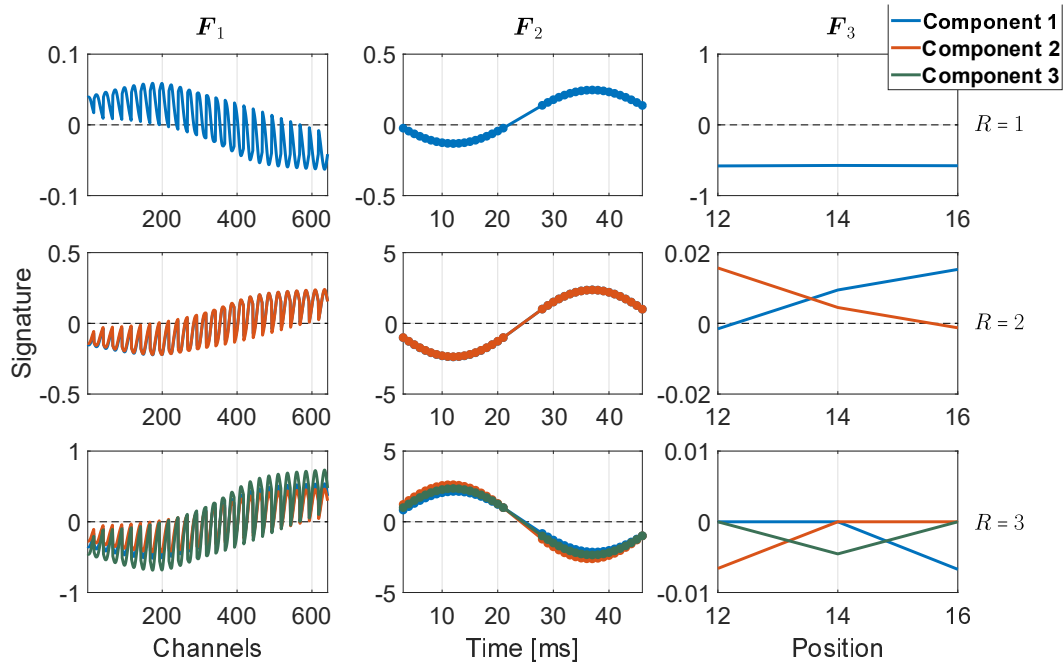


Figure 6.10.: Components of the MEG signal tensor without a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.

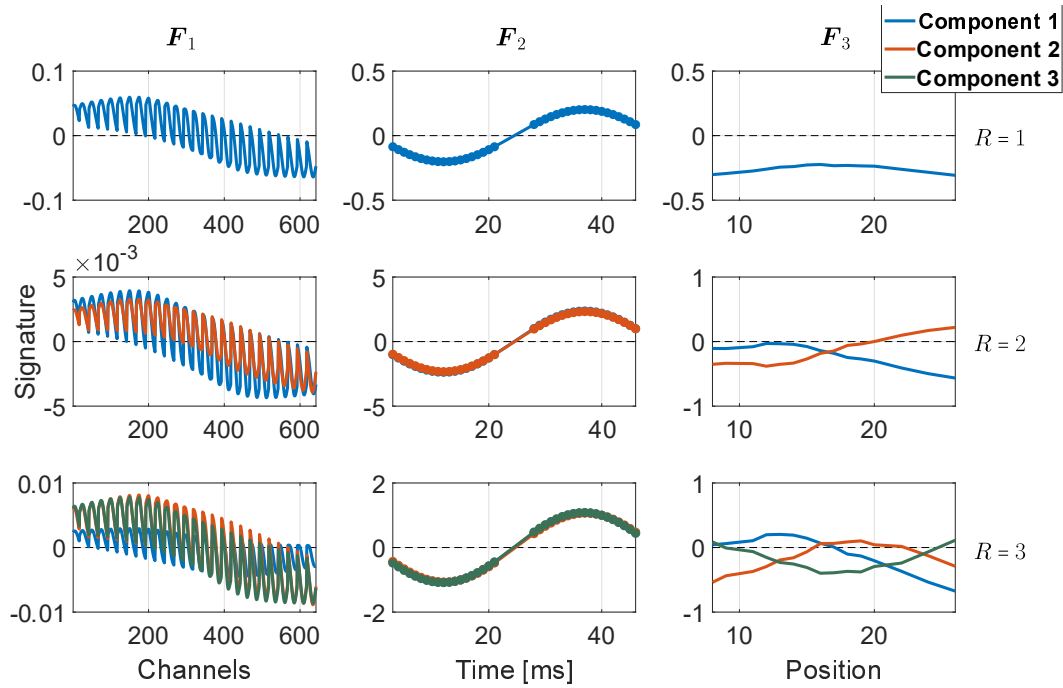


Figure 6.11.: Components of the MEG signal tensor with a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.

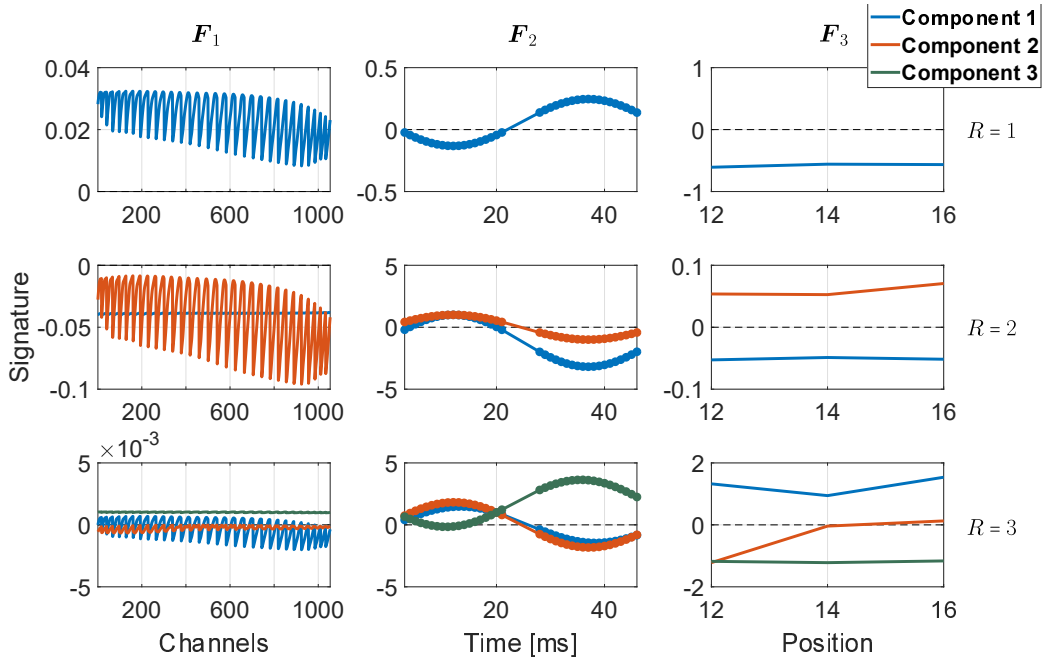


Figure 6.12.: Components of the EEG signal tensor without a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 1, component 2, and component 3, respectively.

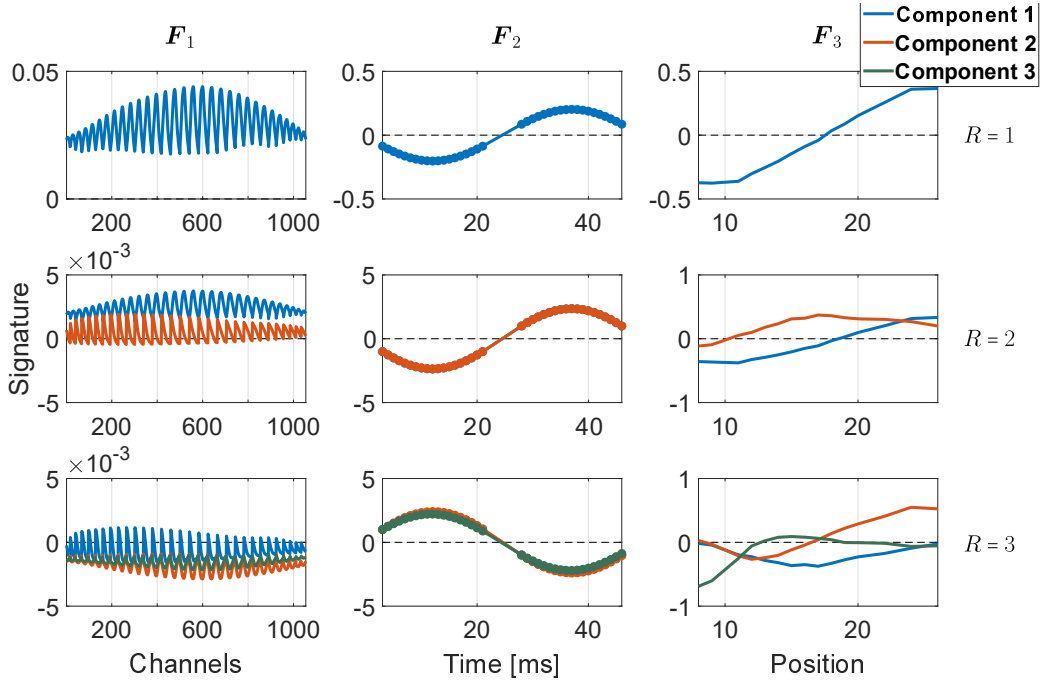


Figure 6.13.: Components of the EEG signal tensor with a skull defect resulting from the coupled EEG-MEG, CP decomposition for tensor ranks $R = 1, 2, 3$ (first, second, and third row, respectively). The curves in blue, red, and green represent component 2, and component 3, respectively.

is noticeable that the changes in the signature of the channels correspond to the position changes. Therefore, for the MEG signal tensors with defect, more signal components can be extracted, as it was expected based on the residual and reliability in Table 6.3.

Moreover, in Figs. 6.12 and 6.13, we illustrate the components for the EEG signal without and with skull defect resulting from the EEG-MEG coupled CP decomposition for rank $R = 1, 2, 3$, respectively. In contrast to the MEG signals without defect, for the EEG signals without defect, the extracted components do not represent only one position, but a linear combination of them. For the EEG signal with defect (Fig. 6.13) similar to MEG, the three components represent the dipole positions under, before, and after the skull defect.

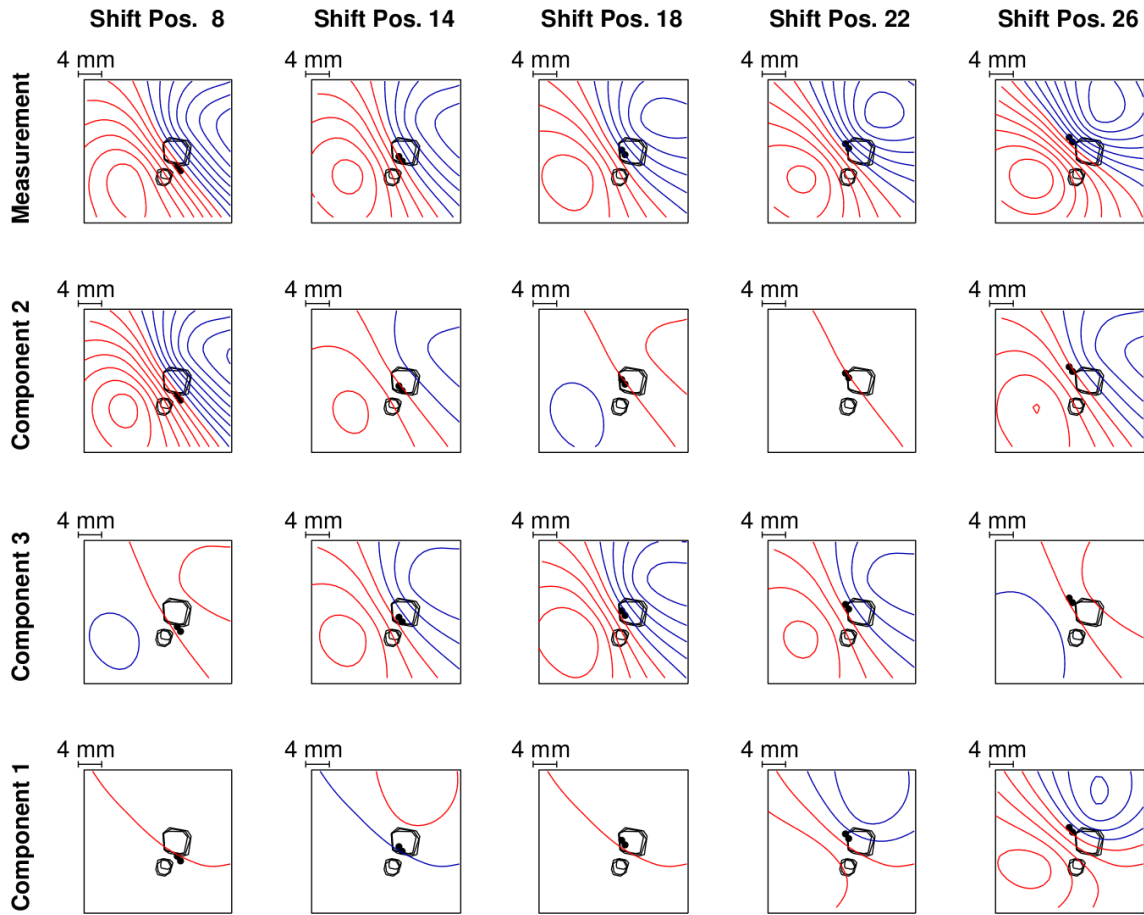


Figure 6.14.: Measured MEG signals above two skull defects (row 1) and components of rank 3 coupled decomposition (rows 2-4) shown at selected source positions (columns) at the first peak in the time dimension. The iso-line increment is uniform within measurements and within components, respectively. The components are arranged in a meaningful sequence. The dipolar source is indicated with a black bar with two spheres marking the poles. Skull defects are marked by closed black lines indicating the inner, middle and outer boundaries of the defects.

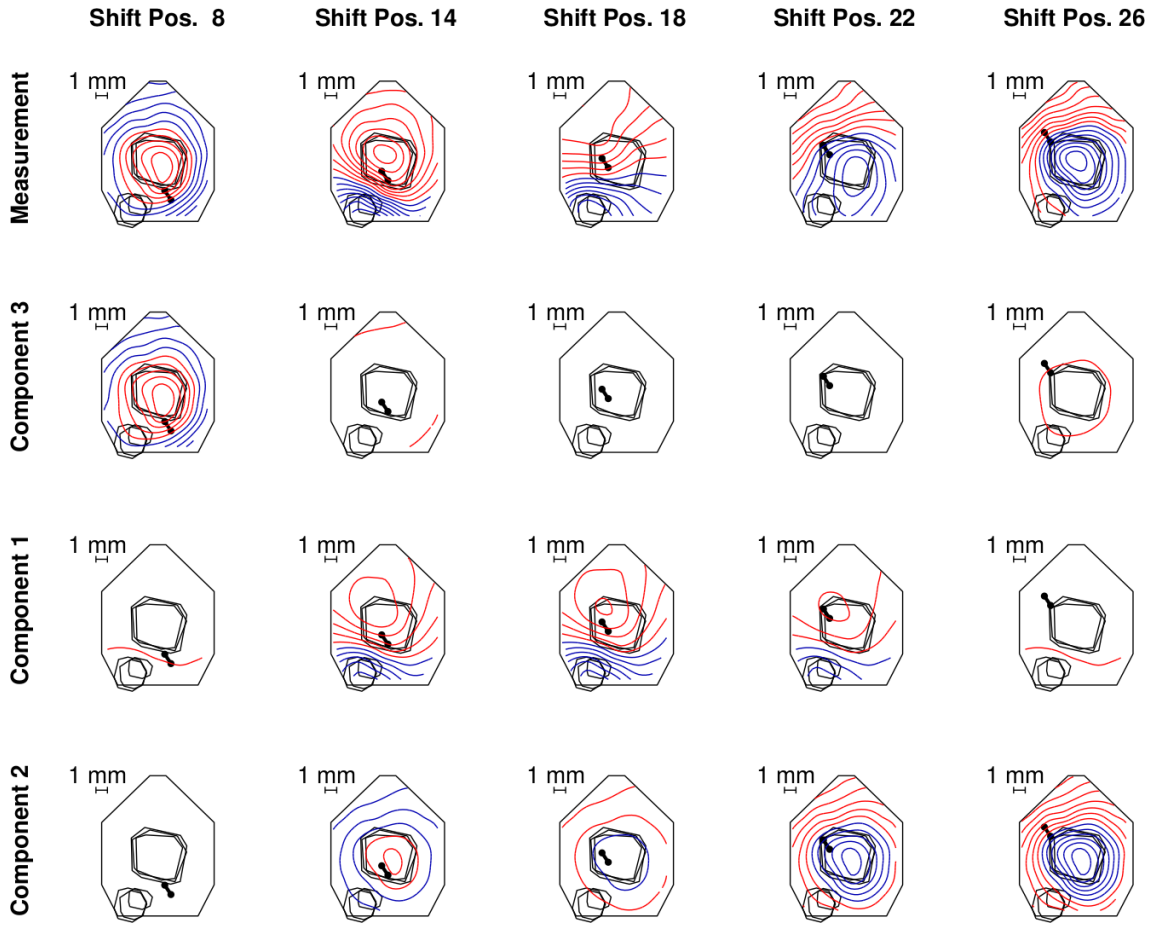


Figure 6.15.: Measured EEG signals above two skull defects (row 1) and components of rank 3 coupled decomposition (rows 2-4) shown at selected source positions (columns) at the first peak in the time dimension. Formatting and markings equivalent to Fig. 6.14.

MEG Signal Components

Fig. 6.14 shows the rank 3 components computed based on the coupled decomposition of the MEG signals above the two skull defects. The measured MEG signals (top row) of the tangential source experience changes in position, orientation, and amplitude [LFH14]. Component 2 (second row) of the tensor decomposition reflects primarily the MEG signals at the lowest source positions. Component 3 (third row) models the mid-range of source positions. The highest source positions are represented by a combination of component 1 (row 4) and component 2. The gradual transition of component amplitudes is also reflected in the lower right diagram of Fig. 6.11. The rank 2 decomposition in row 2 of Fig. 6.11 models the measurements with one or two components, but the difference in source position is not as well differentiated as with rank 3. With rank 1 (see row 1 in Fig. 6.11), the one component represents the mid-range source positions best with errors increasing towards low and high source positions. Hence, the best available representation of the MEG signal with defect is using three components.

Without skull defects, the coupled decomposition is based on three available measurements (pos.

12, 14, and 16). Therefore, the rank 3 result trivially represents each measurement with one component, which can be seen in the lower right diagram of Fig. 6.10. Assuming rank 2, the two components represent primarily the lowest and highest source position, respectively, whereas position 14 is represented by both components (see Fig. 6.10). Using rank 1, the single component represents the middle source position best with errors increasing towards low and high positions (see Fig. 6.10).

EEG Signal Components

Fig. 6.15 shows the components from the rank 3 coupled decomposition of the EEG signals above the two skull defects. The measured EEG signals (top row) of the tangential source experience a reversal of polarity above defect 1 depending on which pole is closer to it [LFH14]. When the source is approximately central under the defect 1, the overall topography is bipolar, but with distortions above that skull defect. Component 3 represents the defect-related monopolar signal at the lowest positions of both defects. Component 1 represents the source positions close to the center of the defect and bipolar aspects related to defect 2. Component 2 captures the monopolar signal increase above defect 1 at middle and high positions. The strength of the components transitions across the position range. This can also be observed in the position signatures in Fig. 6.13.

Without skull defects, a rank 3 coupled decomposition represents the measured EEG signals primarily with one component only (see Fig. 6.12 where the positions are not clearly separated from one another). This matches the fact that the three source positions only span 0.7 mm along the shift axis and, therefore, are very similar if the skull is intact. However, the decomposition has separated the common signal from an instance of noise in the measurement at position 12 (see component 2 row 3 in Fig. 6.12). Assuming rank 2 or rank 1, none of the positions are separated.

6.2.4 Discussion and Summary

The decomposition of the measured EEG and MEG signals using C-SECSI produces meaningful components with respect to the experimental setup. The main mode to be decomposed is the source position, along which the signal changes due to the presence of skull defects. Based on the experimental results [LFH14], the range of source positions can be broadly divided into three cases, the positions under defect 1 and the positions on either side of defect 1. This is reflected in the three components of the rank 3 decomposition of MEG and EEG signals, respectively. The gradual transition between the cases along the source position series is modeled by the combination of two components. The defect-related EEG signals are up to 10 times as strong as the intact skull EEG signals, whereas the amplitude difference in the MEG signals is only approximately 24 % [LFH14]. Consequently, the components of the EEG signals reflect mostly defect-related aspects and the MEG signal components reflect changes in source position jointly with the defect-related changes.

The measurements without defects show a difference in rank between EEG and MEG signals.

The EEG measurements are primarily represented by one component only (rank 1), whereas the simultaneously measured MEG signals are estimated with three components (rank 3). This may be due to the stronger topographic difference of the MEG signals between source positions as well as the higher topographic complexity in this experimental setup.

The experimental setup with skull defects involves even more than three conceptual components. For example, defect 1 and defect 2 may be differentiated and the intact skull signal component could be isolated in the defect measurements. However, the available data samples do not span these dimensions with sufficiently many data points. Consequently, the rank estimation and decomposition identify only the three components in the data.

Using the C-SECSI framework, coupled EEG-MEG signals above intact skull and above two conducting skull defects have been decomposed in order to determine the influence of the skull defects in the tensor decomposition. Meaningful components have been successfully extracted representing the three characteristic signal topographies for the source position. The C-SECSI framework is a very promising method for blind source separation, signal decomposition within the source reconstruction workflow, and for signal artifact extraction. The multimodal integration of EEG and MEG signals through their coupling can improve the localization accuracy in clinical diagnostics, pre-surgical planning, and functional mapping of the human brain.

6.3 Conclusions

In this chapter, we focus on applications of the coupled CP decomposition in biomedical signal processing. Under the assumption that the frequency mode is common for EEG and MEG signals, we provide a validation of the photic driving effect using the C-SECSI framework. Moreover, we show that the model order estimation for coupled tensors can be controlled using the reliability of the C-SECSI framework. Our analysis results show that the EEG and MEG tensors do not necessarily have the same tensor rank. The frequency entrainment is more evident after the decompositions than before the decompositions. When there was no evident frequency entrainment, a frequency in the alpha band or in the theta band becomes noticeable. In the future, we should consider to extend the analysis to wider frequency band e.g., from 1 Hz to 40 Hz in order to investigate the existence of other brain oscillators.

Furthermore, we perform a coupled analysis of EEG-MEG signals above skull defects resulting for a controlled experiment based on the C-SECSI framework. Also, in this application, meaningful components are successfully extracted under the assumption that time domain is common for EEG and MEG signals. These components represent the three characteristic signal topographies of the source positions that include before, above, and after skull defects. Note that the common mode (time) contains highly correlated columns resulting in an ill-posed scenario. Even though, we have been able to extract meaningful components using the C-SECSI framework, we should consider to analyze these signals in the future using coupled BTM decompositions in rank- $(L, L, 1)$

terms and compare the results.

As shown in this chapter, the coupled CP decomposition computed using the C-SECSI framework is a robust method for the unsupervised extraction and separation of meaningful components from multi-dimensional biomedical measurement signals. By considering and utilizing the tensor structure, the signal features are effectively extracted from the measured observations. Therefore, this approach may improve our insight into the brain's function and organization. Consequently, leading to an improved treatment outcome of brain disorders, for example, by locating epileptic network nodes in the epileptic brain.

Chapter 7

Conclusions and Future Work

This thesis focuses on two application areas namely signal processing for wireless communication system and biomedical signal processing. The observed/measured signals in these two applications are multi-dimensional. Their dimensions typically correspond to time, frequency, antennas (space), and users as well as time, channels (space), modality (electroencephalography, magnetoencephalography, electrocardiography), participant (volunteer), and experimental condition, respectively. With the goal of preserving the multi-dimensional structure, we use tensor based methods to model and analyze these signals. In this thesis, we develop new theoretical aspects, efficient algorithms for the computation of tensor decompositions, and new flexible models for multi-carrier wireless communication systems. As shown in this thesis, these aspects lead to an improved receiver design for multi-carrier wireless communication systems and efficient extraction of meaningful signal features in biomedical signal processing. Moreover, the novel aspects and results presented in this thesis open new research directions in these fields.

Chapter 2 is dedicated to the fundamental concepts of tensor algebra, tensor decompositions, and applications of tensor algebra. Moreover, we present our contribution to the fundamental concepts of tensor algebra for the generalized tensor contraction operator in Section 2.1.4. In particular, we propose an alternative representation of the element-wise multiplication and slice-wise multiplication between two arrays using the generalized tensor contraction. In contrast to the element-wise or slice-wise representations, this novel representation facilitates the derivation of the explicit tensor structure and all corresponding tensor unfoldings as shown in Chapter 4 and Chapter 5 of this thesis. In the future, we should study the uniqueness properties of the overall tensor after the contraction that may lead to new identifiability results.

Chapter 3 is devoted to the efficient computation of the CP decomposition and the coupled CP decomposition that is the basis of many signal/data analysis applications. The SECSI framework [RH08,RSH12,RH13a] is a robust and efficient framework for the computation of an approximate low-rank CP decomposition. It computes all possible symmetric SMDs that lead to six initial estimates of the factor matrices, for a 3-way tensor. The final estimate is then selected based on different heuristics as discussed in [RH13a] that lead to different complexity-accuracy trade-offs in the SECSI framework. In Chapter 3, we propose five extensions of the SECSI framework that reduce the computational complexity of the original framework or introduce constraints to the factor matrices. In this thesis, we show the derivations for 3-way tensors. However, the presented results can be easily extended to N -way tensors. The extensions of the SECSI framework include T-SECSI

(Truncated-SECSI), NS-SECSI (Non-Symmetric-SECSI), S-SECSI (Symmetric-SECSI/SECSI for symmetric tensors), SECSI₊ (Non-negative SECSI/SECSI for non-negative tensors), and C-SECSI (Coupled-SECSI/SECSI for coupled tensors). The T-SECSI framework and the NS-SECSI framework have a lower computational complexity than SECSI because they diagonalize a compressed core tensor for size $R \times R \times R$, where R is the tensor rank. This gain is more pronounced as the dimensions of the low-rank tensor increase. In addition, the NS-SECSI framework exploits non-symmetric SMDs for the computation of the CP decomposition. The NS-SECSI-NS-IDIEM framework represents a closed-form (algebraic) solution for the computation of the approximate low-rank CP decomposition. Recall that previous publications propose closed-form solutions for the CP decomposition only in special cases (for tensors with rank two, and tensors with two slices) [RH13a], whereas we propose a closed-form solution for the general case. Moreover, NS-SECSI computes only three initial sets of estimates of the factor matrices that additionally reduces the computational complexity of the selection of the final solution. Therefore, NS-SECSI has even lower computational complexity than T-SECSI. Hence, we recommend the closed-form NS-SECSI framework for the computation of the CP decomposition for low-rank tensors with large dimensions. The S-SECSI framework provides a closed-form solution for the computation of the CP decomposition of symmetric tensors or fully symmetric tensors. The SECSI₊ framework computes an approximation of the CP decomposition for non-negative tensors under the constraint that the factor matrices are non-negative. In the first step of the SECSI₊ framework, we compress the non-negative tensor based on NTD, and then we compute symmetric SMDs with non-negativity constraints. To this end, we also propose an ADMMD₊ diagonalization algorithm for the computation of SMDs with non-negativity constraints. However, this SECSI₊ framework has no advantages over the state-of-the-art algorithms. In the future, we should consider to investigate the compression step further, to consider other NTD algorithms than the one proposed in [BKS⁺12], and to consider non-symmetric SMDs as proposed in the NS-SECSI framework. It can also be considered the derivation of SECSI₊ for tensors where only some of the factor matrices have non-negativity constraints. Furthermore, the C-SECSI framework computes the coupled CP decomposition in a robust semi-algebraic fashion. It outperforms the state-of-the-art algorithms especially in ill-conditioned scenarios. In addition, it does not require prior normalization of the tensors even if they are corrupted by noise with different variances. In combination with the C-SECSI framework, we propose a reliability measure that controls the rank of the coupled tensor decomposition. The C-SECSI framework is utilized in Chapter 6 for the joint analysis of EEG and MEG signals. In the future, it is possible to extend the C-SECSI framework to coupled matrix-tensor decompositions and to coupled CP for more than two tensors. A closed-form solution of C-SECSI can also be obtained based on the IDIEM [CB12] algorithm. Note that, it is possible to further extend the SECSI framework and the C-SECSI framework by considering sparse tensors and tensors with missing entries if in the first step we use an appropriate compression technique that can handle missing entries such as [YFLZ16] instead of the truncated HOSVD. The NS-SECSI framework and the C-SECSI framework have already been published in [NHT⁺16, NH16] and [NKHH17], respectively.

In addition to the CP decomposition, we consider the PARATUCK2 decomposition and the

PARAFAC2 decomposition in Chapter 4. We first show that these decompositions can be interpreted as a slice-wise multiplication between two tensors. Using the novel representations of the slice-wise multiplication proposed in Section 2.1.4 based on the generalized tensor contraction and by substituting the individual structure of the tensors involved in the contraction, we derive new tensor models for both decompositions PARATUCK2 and PARAFAC2. For the PARATUCK2 decomposition, we derive two alternatives of a constrained CP model that can be used to study the uniqueness of the PARATUCK2 decomposition. Note that we have not yet derived all overall models that result from different combinations of the tensors involved in the contraction. In the future, we can derive and exploit these models as well. The overall tensor models can also be used to develop new efficient algorithms for the computation of the PARATUCK2 decomposition. In addition, it is possible to extend these models to the generalized PARATUCK decompositions for N -way ($N > 3$) tensors [FdA14b]. The novel tensor representation also leads to a constrained CP model of the PARAFAC2 decomposition that can be used to study its uniqueness properties. We exploit this novel representation of the PARAFAC2 model to derive an efficient single loop ALS algorithm for the computation of the PARAFAC2 decomposition that requires fewer iteration than the state-of-the-art algorithms. This algorithm has already been published in [NCdAH18]. Note that the proposed algorithm estimates the low-rank tensor from a noisy observation in an LS sense. However, the tensor models proposed in this thesis can be used as a starting point in the derivation of other algorithms. The derivation of these models for the PARATUCK2 and the PARAFAC2 decomposition opens up interesting research areas. For instance, in our derivations we often encounter a Kronecker product between two tensors. Therefore, we should inquire into the properties of the Kronecker product between two tensors. In the future, these properties can help us to theoretically derive new structures. To this end, the derived overall tensor structure recurrently fits a constrained CP model or a constrained BTD model. Hence, it is of a great importance to study the uniqueness properties of these models. These uniqueness properties may consequently lead to novel identifiability results. For instance, we observe at the end of Appendix B.1 that by studying the uniqueness properties of the constrained BTD decomposition, we can derive identifiability results concerning a matrix factorization.

Chapter 5 focuses on the modeling of multi-carrier wireless communication systems using tensor algebra. We first present a tensor model for MIMO-OFDM systems based on the generalized contraction between a channel and a transmit signal tensor. This model is a general and flexible way of describing the received signal in MIMO-OFDM systems for all subcarriers jointly. Moreover, together with this model we propose a new representation of the channel tensor using a 4-way tensor with a special BTD structure. Using the general tensor model and the properties of the contraction operator, we are able to derive an explicit tensor model of the overall received signal that facilitates the design of several iterative and recursive receivers for MIMO-OFDM systems. The accuracy of the ILSP algorithm depends on the rank of the transmitted symbol matrices. Hence, the system can be modified such that only specific code words are used in the future. Moreover, the recursive algorithms can be modified such that they exploit the temporal channel correlation in time varying scenarios. Note that the proposed algorithms already exploit the correlation of the channel among

adjacent subcarriers that leads to a reduced number of pilot symbols as compared to other tensor models. Next, we show that the generalized tensor model using the contraction operator can straightforwardly be extended to Khatri-Rao coded MIMO-OFDM systems. In this model, we exploit the Khatri-Rao coding strategy that imposes a CP structure on the transmit signal tensor. The overall tensor model leads to a receiver based on LSKRF that can be more improved by means of an additional LS iteration. Moreover, we can consider not just one additional LS iteration, but several iterations leading to an ALS based receiver initialized using the LSKRF. In the future, we can consider the design of optimal orthogonal pilot sequences specific to the proposed KR receiver. Recognize that the Khatri-Rao coding strategy has a reduced spectral efficiency than the uncoded MIMO-OFDM system. Therefore, we propose an alternative transmission technique where the "coding matrices" contain random data symbols. Thereby, this transmission technique also imposes a CP structure on the transmit signal tensor while having higher spectral efficiency than the Khatri-Rao coding technique. We propose two types of receivers for randomly coded MIMO-OFDM systems based on the overall structure of the resulting receive signal tensor denoted by RC-KR and RC-KR+ALS. The RC-KR receiver estimates the symbol matrices based on LSKRF, whereas the RC-KR+ALS receiver uses the LSKRF estimates to initialize an ALS algorithm to further enhance the accuracy of the receiver. In contrast to the receivers for Khatri-Rao coded MIMO-OFDM, the receivers for randomly coded MIMO-OFDM systems assume that $M_R \geq M_T$. To relax this condition, we can consider recursive LS instead of LS to estimate the symbol matrices. In the future, the system can be modified such that both symbol matrices contain symbols from different constellations and/or different modulation order. This will lead to a resulting transmit signal tensor with diverse entries and potentially improved performance for the receivers in terms of SER. The aforementioned models for MIMO-OFDM and Khatri-Rao coded MIMO-OFDM systems and their corresponding receivers have been published in [NHdA18] and [NHdA17], respectively.

Moreover, in Chapter 5 we show that the proposed general model using tensor contraction for MIMO-OFDM systems can be straightforwardly extended to MIMO-GFDM systems. The transmit signal tensor in the case of MIMO-GFDM systems can also be expressed in terms of generalized tensor contraction. Therefore, we also use the properties of the tensor contraction to provide more insides of the structure of the GFDM modulation matrix. Using on the resulting received signal, we derive an ALS receiver for MIMO-GFDM systems. Note that the number of frames in this model is set to one ($K = 1$). However, the extension of the model to multiple frames is straightforward and leads to an additional tensor gain. Using the proposed model, we can study the structure of the resulting channel tensor for MIMO-GFDM systems, where the subcarriers are not orthogonal. Moreover, the proposed model opens up new opportunities for the GFDM systems, such as finding the best pilot sequences, investigating more general GFDM systems when not all carriers or subsymbols are used for data transmission, and investigating new closed form solutions by adding coding or random coding. The presented iterative ALS receiver has already been published in [NCH⁺17] where the ALS receiver is derived based on the PARATUCK2 decomposition. Recall that many references dedicated to relay-assisted MIMO communication systems either assume a single carrier system or use subcarrier-wise description of the system. The proposed general model

based on tensor contraction can be simply extended to systems with multiple hops by contracting the received signal of the first hop with the channel tensor of the following hop. Therefore, the tensor model proposed in this thesis provides an inspiring way of exploiting the benefits of tensor contraction in relay-assisted multi-carrier systems. Even more, the proposed tensor model can be extended to multi-user system in a straightforward fashion. Furthermore, the general tensor model for multi-carrier systems proposed in Chapter 5 can be extended to other multi-carrier techniques such as UFMC and FBMC. Another promising perspective is to assume a low-rank structure of the channel tensor, which will be a more realistic assumption for millimeter wave MIMO-OFDM systems. Exploiting the generalized tensor contraction model and the additional structure of the low-rank channel tensor would lead to new blind receivers.

The last chapter of this thesis is dedicated to applications of the coupled CP decomposition in biomedical signal processing. First, we analyze the photic driving effect using the C-SECSI framework under the assumption that the frequency mode is common for the EEG and the MEG signals. This analysis shows that using the reliability measure of the C-SECSI framework, the model order of the coupled tensor decompositions can be controlled. Moreover, our analysis results show that the EEG and MEG tensors do not necessarily have equal tensor ranks. We are also able to observe that the frequency entrainment is more evident after the decompositions than before the decompositions. However, when there was no evident frequency entrainment, a frequency in the alpha band or in the theta band becomes noticeable. In the future, we should consider to extend the analysis to wider frequency band e.g., from 1 Hz to 40 Hz in order to investigate the existence of other brain oscillators. The analysis presented in this thesis is for frequencies between 3.77 Hz and 15.15 Hz, i.e., in the theta band and alpha band. Note that the analysis presented here is restricted to 3-way tensors. However, from the available signals it is possible to construct even 5-way tensors (frequency \times time \times channels \times stimulation frequency \times volunteer). Therefore, we should consider analyzing these 5-way tensors directly and comparing the results to the analysis presented here. Moreover, we perform a coupled analysis of EEG-MEG signals above skull defects resulting for a controlled experiment based on the C-SECSI framework. In this application, meaningful components are also successfully extracted under the assumption that time domain is common for the EEG and the MEG signals. The extracted components correspond to the three characteristic signal topographies of the source positions that include before, above, and after skull defects. Note that in this application, the columns of the common mode (time) are highly correlated resulting in an ill-posed scenario. Despite this, using the C-SECSI framework, we were able to extract meaningful components. However, we can analyze in the future these signals using coupled BTD decompositions in rank- $(L, L, 1)$ terms and compare the results. In both applications, we observe that the EEG and MEG tensors do not necessarily share the same number of components. Therefore, flexible coupled models are more suitable for biomedical signal processing rather than exact coupled models. Hence, we require robust algorithms for the computation of the CP decomposition such as the C-SECSI framework. Recall that C-SECSI computes several coupled and uncoupled estimates for the factor matrices, thus it is able to handle flexible coupled models. As shown in this chapter, the coupled CP decomposition computed using

the C-SECSI framework is a robust method for the unsupervised extraction and separation of meaningful components from multi-dimensional biomedical measurement signals. By considering and utilizing the tensor structure, the EEG and MEG signal features are effectively extracted from the measured observations. Hence, this approach may improve our insight into the brain's function and organization.

Appendices

Appendix A

Acronyms, Symbols, and Notation

A.1 Acronyms

3GPP	3rd Generation Partnership Project
ADMM	Alternating Direction Method of Multipliers
ADMMD+	Alternating Direction Method of Multipliers for non-negative simultaneous matrix Diagonalization
ALMS	Alternating Least Mean Squares
ALS	Alternating Least-Squares
BM	Best Matching
BSUM	Block Successive Upper bound Minimization
BTD	Block Term Decomposition
C-ALS	Coupled Alternating Least-Squares
CAND	CANonical Decomposition
CANDECOMP	CANonical DECOMPosition
CCDF	Complementary Cumulative Distribution Function
CCP-MINF	Coupled Canonical Polyadic - MINimum Factors
CCP-NLS	Coupled Canonical Polyadic - Nonlinear Least Squares
CDMA	Code-Division Multiple Access
CG	Conjugate Gradient
CON PS	CONditioning criterion Paired Solutions
CONFAC	CONstrained FACtor
CP	Canonical Polyadic
CR-bound	Cramér-Rao-bound
CRI-bound	Cramér-Rao-Induced-bound
C-SECSI	Semi-Algebraic framework for the approximate Coupled CP decomposition via Simultaneous matrix diagonalization
DEDICOM	DEcomposition into DIrectional COMponents
DFT	Discrete Fourier Transform
DIAG	DIRECT ALGORITHM for canonical polyadic decomposition
DIAG+	DIRECT ALGORITHM for canonical polyadic decomposition+
EEG	ElectroEncephaloGraphy

FBMC	Filtered Bank MultiCarrier
FFT	Fast Fourier Transform
FT	Fourier Transform
GD	Gradient Descent
GFDM	Generalized Frequency Division Multiplexing
HOGSVD	Higher-Order Generalized Singular Value Decomposition
HOOI	Higher-Order Orthogonal Iteration
HOSVD	Higher-Order Singular Value Decomposition
ICI	Inter-Carrier Interference
IDIEM	Improved DIagonalization using Equivalent Matrices
IDSCAL	INDividual Differences in SCALing
ILSE	Iterative Least Squares with Enumeration
ILSP	Iterative Least Squares with Projection
IPS	Intermittent Photic Simulation
ISI	Inter-Symbol Interference
JD	Joint eigenvalue Decomposition
JDTM	Joint Diagonalization algorithm based on Targeting hyperbolic Matrices
JET	Joint Eigenvalue decomposition algorithm based on Triangular matrices
JEVD+	Joint EigenValue Decomposition+
KR	Khatri-Rao
KR+LS	Khatri-Rao receiver and its enhancement via Least-Squares
LMMSE	Linear Minimum Mean Square Error
LS	Least-Squares
LSKF	Least-Squares Kronecker Factorization
LSKRF	Least-Squares Khatri-Rao Factorization
MDL	Minimum Description Length
MEET	Modified Eigenvalues Estimator for Tucker rank determination
MEG	MagnetoEncephaloGraphy
MIMO	Multiple-Input Multiple-Output
MMSE	Minimum Mean Squared Error
MRI	Magnetic Resonance Imaging
MSRE	Mean Squared Reconstruction Error
NLS	Nonlinear Least-Squares
NMSE	Normalized Mean Squared Error
NS-IDIEM	Non-Symmetric-Improved DIagonalization using Equivalent Matrices
NS-SECSI	Semi-Algebraic framework for the approximate Canonical Polyadic (CP) decomposition via Non-Symmetric SIMultaneous matrix diagonalization
NTF	Non-negative Tensor Factorization
OFDM	Orthogonal Frequency Division Multiplexing

OPP	Orthogonal Procrustes Problem
PARAFAC	PARAllel FACtors
PARAFAC2	PARAllel FACtors2
PARALIND	PARAllel profiles with LINear Dependences
PARATUCK2	PARAfac and TUCKer2
PD	Photic Driving
Ped A	Pedestrian A
QAM	Quadrature Amplitude Modulation
RC	Random Coding
RC-KR	Random Coding-Khatri-Rao
RC-KR+ALS	Random Coding-Khatri-Rao + Alternating Least-Squares
REC PS	REConstruction criterion Paired Solutions
REL	RELIability
RES	RESidual
RLSE	Recursive Least-Squares with Enumeration
RLSP	Recursive Least-Squares with Projection
RMS	Root Mean Square
RNA	RiboNucleic Acid
SALT	Semi-ALgebraic Tensor decomposition
SECSI	Semi-Algebraic framework for approximate CP decomposition via SImultaneous matrix diagonalization
SECSI ₊	Semi-Algebraic framework for the approximate CP decomposition via SImultaneous matrix diagonalization for non-negative tensors
SECSI-GU	SECSI-Generalized Unfoldings
SER	Symbol Error Rate
SGD	Stochastic Gradient Descent
SMD	Simultaneous Matrix Decomposition
SNR	Signal to Noise Ratio
SRE	Squared Reconstruction Error
S-SECSI	Semi-Algebraic framework for the approximate Canonical Polyadic (CP) decomposition via SImultaneous matrix diagonalization for symmetric tensors /Symmetric SECSI
SVD	Singular Value Decomposition
TEDIA	TENSOR DIAgonalization
TMSFE	Total Mean Squared Factor Error
T-SECSI	Truncated Semi-Algebraic framework for the approximate CP decomposition via SImultaneous matrix diagonalization
TSFE	Total Squared Factor Error
UFMS	Universal Filtered MultiCarrier

WD	With Defect
WOD	WithOut Defect
ZF	Zero Forcing
ZMCSCG	Zero Mean Circularly Symmetric Complex Gaussian

A.2 Symbols and Notation

j	Imaginary unit, $\sqrt{-1}$
e	Euler number
π	Pi
.....
\mathbb{R}	Set of real numbers
\mathbb{C}	Set of complex numbers
\mathbb{R}_+	Set of non-negative real numbers
.....
a, b, c	Scalars
$\mathbf{a}, \mathbf{b}, \mathbf{c}$	Column vectors
$\mathbf{A}, \mathbf{B}, \mathbf{C}$	Matrices
$\mathcal{A}, \mathcal{B}, \mathcal{C}$	Tensors
.....
$\mathbf{a}_{(i)}$	The i -th element of the column vector $\mathbf{a} \in \mathbb{C}^I$
$\mathbf{A}_{(i,j)}$	The (i,j) -th element of the matrix $\mathbf{A} \in \mathbb{C}^{I \times J}$
$\mathcal{A}_{(i,j,k)}$	The (i,j,k) -th element of the tensor $\mathcal{A} \in \mathbb{C}^{I \times J \times K}$
.....
$\mathbf{e}_{I,i}, \mathbf{e}_i$	$\mathbf{e}_{I,i} \in \mathbb{R}^{I \times 1}$ is a pinning vector of all zeros and one at the i -th position. When not ambiguous the subscript denoting the vector length can be skipped, i.e., \mathbf{e}_i .
$\mathbf{0}_{M \times N}$	Matrix of zeros of size $M \times N$
$\mathbf{1}_{M \times N}$	Matrix of ones of size $M \times N$
\mathbf{I}_M	Identity matrix of size $M \times M$
$\mathcal{I}_{N,R}$	Identity N -way tensor with dimensions $R \times R \times \dots \times R$
.....
$\arg(x)$	Argument (phase) of a complex variable x
$\text{diag}(\cdot)$	Transforms a vector into a square diagonal matrix. Moreover, the operator $\text{diag}(\cdot)$ extracts the main diagonal of a square matrix and places its elements into a vector, when the argument is a matrix.
$\text{blkdiag}(\mathbf{A}_n)_{n=1}^N$	Transforms the matrices $\mathbf{A}_1, \dots, \mathbf{A}_N$ into a block diagonal matrix. The operator $\text{blkdiag}(\cdot)$ is defined as
$\text{blkdiag}(\mathbf{A}_n)_{n=1}^N = \text{blkdiag}(\mathbf{A}_1, \dots, \mathbf{A}_N) = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_N \end{bmatrix} \quad (\text{A.1})$	
$\text{vec}(\cdot)$	Transforms a matrix or a tensor into a column vector. The stacking is

	performed in an increasing order, i.e., begins with first index, then proceed to second index, third index, etc.
$\text{unvec}_{M \times N}(\cdot)$	Transforms a vector into a matrix/tensor of indicated size, i.e., $M \times N$. It is the inverse operation of the $\text{vec}(\cdot)$ operator.
$\text{rank}(\cdot)$	Rank of a matrix or a tensor
$\text{n-rank}(\cdot)$	$\text{n-rank}(\cdot)$ (multi-linear rank) of a tensor
$\text{k-rank}(\cdot)$	Kruskal rank of a matrix
$\min(\cdot)$	Minimum
$\max(\cdot)$	Maximum
$\mathbb{E}\{\cdot\}$	Expected value, expectation operator
.....
$\ \cdot\ _2$	Euclidean (two-) norm
$\ \cdot\ _F$	Frobenius norm
$\ \cdot\ _H$	Higher-Order norm
.....
z^*	Complex conjugate of z
$(\cdot)^T$	Matrix transpose
$(\cdot)^H$	Hermitian transpose
$(\cdot)^{-1}$	Matrix inverse
$(\cdot)^+$	Moore-Penrose pseudo-inverse [Moo20, Pen55] of a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ which we can compute via <ul style="list-style-type: none"> • $\mathbf{A}^+ = (\mathbf{A}^H \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^H$ for a full column rank matrix • $\mathbf{A}^+ = \mathbf{A}^H \cdot (\mathbf{A} \cdot \mathbf{A}^H)^{-1}$ for a full row rank matrix • based on the economy size SVD of $\mathbf{A} = \mathbf{U}^{[s]} \mathbf{\Sigma}^{[s]} \mathbf{V}^{[s]H}$, $\mathbf{A}^+ = \mathbf{V}^{[s]} \mathbf{\Sigma}^{[s]-1} \mathbf{U}^{[s]H}$
$(\cdot)^{-T}$	Matrix transpose and inverse
$(\cdot)^{-H}$	Hermitian transpose and inverse
.....
\triangleq	Definition
\approx	Approximate
.....
$\langle \cdot \rangle$	Scalar/Inner Product. Scalar product between two tensors $\mathbf{A} \in \mathbb{C}^{M_1 \times M_2 \times \dots \times M_N}$ and $\mathbf{B} \in \mathbb{C}^{M_1 \times M_2 \times \dots \times M_N}$ is defined as $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{m_1=1}^{M_1} \sum_{m_2=1}^{M_2} \dots \sum_{m_N=1}^{M_N} \mathbf{A}^*_{(m_1, m_2, \dots, m_N)} \mathbf{B}_{(m_1, m_2, \dots, m_N)} = \mathbf{A}^{\bullet 1, 2, \dots, N}_{1, 2, \dots, N} \mathbf{B} \quad (\text{A.2})$
\circ	Outer product

\otimes	Kronecker product between $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{P \times Q}$ defined as $\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{1,1} \cdot \mathbf{B} & a_{1,2} \cdot \mathbf{B} & \dots & a_{1,N} \cdot \mathbf{B} \\ a_{2,1} \cdot \mathbf{B} & a_{2,2} \cdot \mathbf{B} & \dots & a_{2,N} \cdot \mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M,1} \cdot \mathbf{B} & a_{M,2} \cdot \mathbf{B} & \dots & a_{M,N} \cdot \mathbf{B} \end{bmatrix} \in \mathbb{C}^{PM \times QN} \quad (\text{A.3})$
\diamond	Similarly, Kronecker product between two tensors $\mathcal{A} \in \mathbb{C}^{M \times N \times L}$ and $\mathcal{B} \in \mathbb{C}^{P \times Q \times R}$ is defined as $\mathcal{K} = \mathcal{A} \otimes \mathcal{B} \in \mathbb{C}^{MP \times NQ \times LR}$ (Section 2.1.1). Khatri-Rao (column-wise Kronecker) product between $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{P \times N}$ defined as $\mathbf{A} \diamond \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_2 \otimes \mathbf{b}_2 \quad \dots \quad \mathbf{a}_N \otimes \mathbf{b}_N] \in \mathbb{C}^{PM \times N} \quad (\text{A.4})$
\boxtimes_R	If the matrices \mathbf{A} and \mathbf{B} are partitioned in R blocks as $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_R]$ and $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_R]$, a partition-wise Kronecker product between these matrices is defined as $\mathbf{A} \boxtimes_R \mathbf{B} = [\mathbf{A}_1 \otimes \mathbf{B}_1, \dots, \mathbf{A}_R \otimes \mathbf{B}_R]$. Note that if R equals the number of columns of \mathbf{A} and \mathbf{B} (i.e., if the partitioning is in a column-wise fashion), the partition-wise Kronecker product equals the Khatri-Rao product.
\odot	Hadamard (element-wise) product between two vectors/matrices/tensors of equal dimensions, for instance, $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{M \times N}$.
\oslash	Inverse Hadamard product (element-wise division) between two vectors/matrices/tensors of equal dimensions, for instance, $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{M \times N}$.
$\mathcal{A} \times_n \mathbf{U}$	n -mode product between a tensor \mathcal{A} and a matrix \mathbf{U} (Section 2.1.1).
$\mathcal{A} \times_{n=1}^N \mathbf{U}_n$	Repeated n -mode products along the modes $n = 1, \dots, N$, shorthand notation for $\mathcal{A} \times_1 \mathbf{U}_1 \dots \times_N \mathbf{U}_N$ (Section 2.1.1)
$[\mathcal{A}]_{\sqcup_n} \mathcal{B}$	Tensor concatenation along the n -mode of the tensors \mathcal{A} and \mathcal{B} (Section 2.1.1)
$[\mathcal{A}]_{(n)}$	n -mode unfolding of tensor \mathcal{A} (Section 2.1.1).
$[\mathcal{A}]_{([\alpha^{(1)}, \alpha^{(2)}])}$	Generalized matrix unfolding of an N -way tensor \mathcal{A} . The set of indices is divided into P and $N - P$ non-overlapping subsets, $\alpha^{(1)} = [\alpha_1 \dots \alpha_P]$ and $\alpha^{(2)} = [\alpha_{P+1} \dots \alpha_N]$ (Section 2.1.1).
$\mathcal{A} \bullet_n^m \mathcal{B}$	Tensor contraction between tensors $\mathcal{A} \in \mathbb{C}^{I_1 \times \dots \times I_N}$ and $\mathcal{C} \in \mathbb{C}^{J_1 \times \dots \times J_M}$ that it is represented as an inner product of the n -mode of \mathcal{A} with the m -mode of \mathcal{B} , provided that $I_n = J_m$ (Section 2.1.1).
$\mathcal{A} \bullet_{n,k}^{m,l} \mathcal{B}$	Double tensor contraction. The n -mode and k -mode of \mathcal{A} is contracted with the m -mode and l -mode of \mathcal{B} , respectively (Section 2.1.1).
$(\cdot)^{[s]}$	Denotes the signal component

$(\cdot)^{[n]}$	Denotes the noise component
$(\cdot)^{\text{rhs}}$	The superscript rhs denotes right-hand side
$(\cdot)^{\text{lhs}}$	The superscript lhs denotes left-hand side
\hat{x}	Denotes an estimate of x

\mathbf{F}_N Denotes a DFT matrix of size $N \times N$

$$\mathbf{F}_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & w_N & \dots & w_N^{(N-1)} \\ 1 & w_N^2 & \dots & w_N^{2(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w_N^{N-1} & \dots & w_N^{(N-1)^2} \end{bmatrix}, \quad \text{where } w_N = e^{-j\frac{2\pi}{N}} \quad (\text{A.5})$$

Moreover, a matrix $\mathbf{F}_{N \times L} \in \mathbb{C}^{N \times L}$ contains the first L columns of the DFT matrix \mathbf{F}_N .

$u(t) \circ \bullet U(f)$ Denotes Fourier transform, where $u(t)$ and $U(f)$ represents the signal in the time and the frequency domain, respectively, or a DFT for discrete signals.

$\tilde{s}, \tilde{\mathbf{S}}, \tilde{\mathcal{S}}$ We use \sim to distinguish the frequency domain from the time domain signal vector, matrix, or tensor, i.e., $s \circ \bullet \tilde{s}$, $\mathbf{S} \circ \bullet \tilde{\mathbf{S}}$, or $\mathcal{S} \circ \bullet \tilde{\mathcal{S}}$

Appendix B

Proofs and derivations

B.1 Derivation of the explicit tensor structure of a diagonalized tensor

In Section 2.1.4, we show that an element-wise or a slice-wise multiplication between two arrays can be expressed as a generalized tensor contraction, where one of the arrays is diagonalized along the unaffected dimension. Next, the diagonalized array can be expressed in terms of its generalized unfolding as a Khatri-Rao product between an identity matrix and a generalized unfolding of the tensor to be diagonalized. The link between the diagonalized array structures and their generalized unfoldings is given in Table 2.1. Here, we derive the explicit tensor structure of a diagonalized tensor $\mathcal{T}_D \in \mathbb{C}^{I \times J \times K \times K}$ with non-zero elements $\mathcal{T}_{D(i,j,k,k)} = \mathcal{T}_{(i,j,k)}$. Moreover, we assume that the tensor to be diagonalized has a CP structure $\mathcal{T} = \mathcal{I}_{3,R} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \in \mathbb{C}^{I \times J \times K}$, where $\mathbf{A} \in \mathbb{C}^{I \times R}$, $\mathbf{B} \in \mathbb{C}^{J \times R}$, $\mathbf{C} \in \mathbb{C}^{K \times R}$, and R is the tensor rank. According to Table 2.1, we have

$$[\mathcal{T}_D]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [\mathcal{T}]_{([1,2],[3])}.$$

By substituting the assumed CP structure of the tensor \mathcal{T} into the above unfolding, we get

$$[\mathcal{T}_D]_{([1,2,3],[4])} = \mathbf{I}_K \diamond [(\mathbf{B} \diamond \mathbf{A})\mathbf{C}^T].$$

However, the above equation does not reveal the tensor structure. Therefore, let us consider the non-zero elements of the diagonalized tensor $\mathcal{T}_{D(i,j,k,k)} = \mathcal{T}_{(i,j,k)}$ and its diagonal slices $\mathcal{T}_{D(i,j,.,.)} = \text{diag}(\mathcal{T}_{(i,j,.,.)})$. We can also express the tensor \mathcal{T}_D by the means of concatenation along the 4-mode

$$\mathcal{T}_D = [\mathcal{T}_1 \quad \sqcup_4 \quad \mathcal{T}_2 \quad \sqcup_4 \quad \dots \quad \mathcal{T}_K], \quad (\text{B.1})$$

where the tensors $\mathcal{T}_k = \mathcal{T}_{D(.,.,.,k)} \in \mathbb{C}^{I \times J \times K}$. From the CP structure of the tensor \mathcal{T} , we get $\mathcal{T}_k = \mathcal{I}_{3,R} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 (\mathbf{e}_k \otimes \mathbf{C}_{(k,.)})$ taking the k -th row of \mathbf{C} . The pinning vector \mathbf{e}_k represents the k -th column of an identity matrix \mathbf{I}_K . We visualize the structure of the matrices \mathbf{C} and $(\mathbf{e}_k \otimes \mathbf{C}_{(k,.)})$ as well as the tensor $\mathcal{I}_{3,R} \times_3 (\mathbf{e}_k \otimes \mathbf{C}_{(k,.)})$, for $K = 3$, $R = 2$, and $k = 1$ in Fig. B.1.

Now, we can substitute these \mathcal{T}_k tensors in equation (B.1). Moreover, the equation (B.1) can be multiplied along the 4-mode with an identity matrix without changing the result.

$$\mathcal{T}_D = [\mathcal{T}_1 \quad \sqcup_4 \quad \mathcal{T}_2 \quad \sqcup_4 \quad \dots \quad \mathcal{T}_K] \times_4 \mathbf{I}_K = [\mathcal{T}_1 \quad \sqcup_4 \quad \mathcal{T}_2 \quad \sqcup_4 \quad \dots \quad \mathcal{T}_K] \times_4 [\mathbf{e}_1 \quad \mathbf{e}_2 \quad \dots \quad \mathbf{e}_K]$$

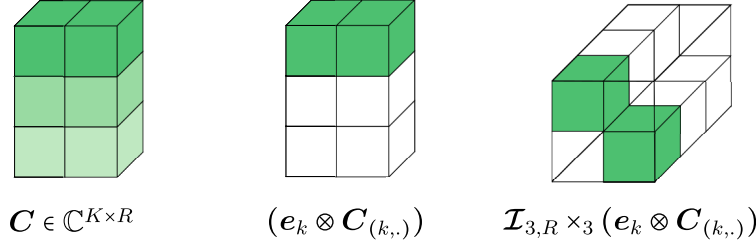


Figure B.1.: Visualization of the structure of the matrices C and $(e_k \otimes C_{(k,.)})$ and the tensor $\mathcal{I}_{3,R} \times_3 (e_k \otimes C_{(k,.)})$, for $K = 3$, $R = 2$, and $k = 1$.

Next, this concatenation in (B.1) can be described as

$$\mathcal{T}_D = \sum_{k=1}^K \mathcal{T}_k \circ e_k = \sum_{k=1}^K (\mathcal{I}_{3,R} \times_1 A \times_2 B \times_3 (e_k \otimes C_{(k,.)})) \circ e_k$$

Here, the outer product can be represented by a 4-mode product, if we add a singleton dimension as a fourth dimension to the core identity tensors, i.e.,

$$\mathcal{T}_D = \sum_{k=1}^K (\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,R}) \times_1 A \times_2 B \times_3 (e_k \otimes C_{(k,.)}) \times_4 e_k. \quad (\text{B.2})$$

The tensor $\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,R} \in \mathbb{R}^{R \times R \times R \times 1}$ is essentially an identity tensor, only its fourth dimension is a singleton dimension. We add the singleton dimension by means of a Kronecker product with a 4-way tensor $\mathcal{I}_{4,1} \in \mathbb{R}^{1 \times 1 \times 1 \times 1}$. Naturally, it is not necessary to explicitly add the singleton dimension, but it simplifies the rest of our derivation. The tensor in (B.2) corresponds to a special BTD of K rank- $(R, R, R, 1)$ terms. Moreover, the terms differ only in the 3-mode and the 4-mode.

In Section 2.2.3, we show that the BTD can be rewritten in a block diagonal structure. According to Fig. 2.15, we have

$$\begin{aligned} \mathcal{T}_D = \text{blkdiag}(\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,R})_{k=1}^K \times_1 \begin{bmatrix} A & \dots & A \end{bmatrix} \times_2 \begin{bmatrix} B & \dots & B \end{bmatrix} \\ \times_3 \begin{bmatrix} e_1 \otimes C_{(1,.)} & \dots & e_K \otimes C_{(K,.)} \end{bmatrix} \times_4 \begin{bmatrix} e_1 & \dots & e_K \end{bmatrix}. \end{aligned}$$

In Fig. B.2, we depict the structure of the matrix $\begin{bmatrix} e_1 \otimes C_{(1,.)} & \dots & e_K \otimes C_{(K,.)} \end{bmatrix} = (\mathbf{I}_K \diamond C^T)^T$, where the matrix C is depicted in Fig. B.1.

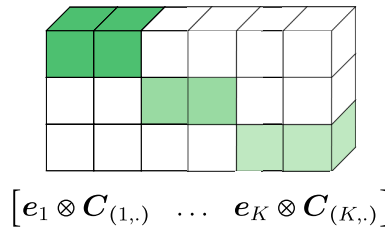


Figure B.2.: Visualization of the structure of the matrix $(\mathbf{I}_K \diamond C^T)^T$, for $K = 3$ and $R = 2$.

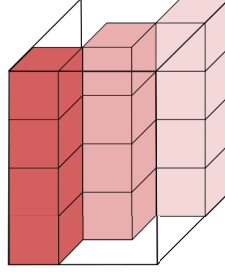
Finally, considering the block diagonal structure of the core tensors and the factor matrices, we obtain the tensor structure of the diagonalized tensor

$$\mathcal{T}_D = (\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,R}) \times_1 (\mathbf{1}_K^T \otimes \mathbf{A}) \times_2 (\mathbf{1}_K^T \otimes \mathbf{B}) \times_3 (\mathbf{I}_K \diamond \mathbf{C}^T)^T \times_4 \mathbf{I}_K,$$

where $\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,R} \in \mathbb{R}^{RK \times RK \times RK \times K}$. To depict the structure of the diagonalized tensor \mathcal{T}_D we should reduce its dimensionality to a 3-way tensor. Therefore, we assume that the matrix to be diagonalized is given by $\mathbf{T} = \mathcal{I}_{2,R} \times_1 \mathbf{B} \times_2 \mathbf{C} \in \mathbb{C}^{J \times K}$ ($\mathbf{T} = \mathbf{B} \cdot \mathbf{C}^T$), where $\mathbf{B} \in \mathbb{C}^{J \times R}$ and $\mathbf{C} \in \mathbb{C}^{K \times R}$. For the corresponding diagonalized tensor, we get

$$\mathcal{T}_D = (\mathcal{I}_{3,K} \otimes \mathcal{I}_{2,R}) \times_1 (\mathbf{1}_K^T \otimes \mathbf{B}) \times_2 (\mathbf{I}_K \diamond \mathbf{C}^T)^T \times_3 \mathbf{I}_K \in \mathbb{C}^{J \times K \times K}, \quad (\text{B.3})$$

where $\mathcal{I}_{3,K} \otimes \mathcal{I}_{2,R} \in \mathbb{R}^{RK \times RK \times K}$. We visualize the diagonal structure of this tensor in Fig. B.3, for $J = 4$, $K = 3$, and $R = 2$.



$$\begin{aligned} \mathcal{T}_D &= \mathcal{I}_{3,K} \times_1 \mathbf{B} \mathbf{C}^T \\ \mathcal{T}_D &\in \mathbb{C}^{J \times K \times K} \end{aligned}$$

Figure B.3.: Visualization of the structure of the tensor $\mathcal{T}_D \in \mathbb{C}^{J \times K \times K}$ in (B.3), for $J = 4$, $K = 3$, and $R = 2$.

Note that the tensor \mathcal{T}_D in equation (B.3) also has a CP structure given by $\mathcal{T}_D = \mathcal{I}_{3,K} \times_1 \mathbf{B} \mathbf{C}^T$. In contrast to this CP structure, the constrained BTD structure in equation (B.3) separates the two matrices \mathbf{B} and \mathbf{C} in two modes. Therefore, by studying the uniqueness properties of the constrained BTD, we can derive results for the identifiability of the matrix factorization $\mathbf{B} \mathbf{C}^T$.

B.2 Structure of the selection matrix

Often in our derivations, we use the selection matrix $\mathbf{J}_{MK} \in \mathbb{R}^{MKMK \times MK}$ that transforms a Kronecker product of two matrices into a Khatri-Rao product of two matrices as shown in (2.29). According to (2.28), this selection matrix can be represented by a Khatri-Rao product between the matrices \mathbf{I}_{MK} and \mathbf{I}_{MK} or by a transpose of any unfolding of the identity tensor $\mathcal{I}_{3,MK} \in \mathbb{R}^{MK \times MK \times MK}$.

$$\mathbf{J}_{MK} = \mathbf{I}_{MK} \diamond \mathbf{I}_{MK} = [\mathcal{I}_{3,MK}]_{(1)}^T = [\mathcal{I}_{3,MK}]_{(2)}^T = [\mathcal{I}_{3,MK}]_{(3)}^T$$

For $M = 2$ and $K = 2$, the selection matrix \mathbf{J}_{MK} has the following elements

$$\mathbf{J}_{MK} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T.$$

Moreover, we get the very same selection matrix \mathbf{J}_{MK} from the following multiplication of two matrices.

$$\mathbf{J}_{MK} = [\mathcal{I}_{4,K} \otimes \mathcal{I}_{3,M}]_{([2,3],[1,4])} \cdot [\mathbf{I}_K \otimes \mathbf{1}_K^T \otimes \mathbf{I}_M]^T \quad (\text{B.4})$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

B.3 Derivation of the 4-way channel tensor in the frequency domain and its unfoldings

Let us assume a MIMO-OFDM system with M_T transmit antennas and M_R receive antennas. Such a system is depicted in Fig. 5.1, for $M_T = 2$ and $M_R = 3$. As shown in Section 5.1.1, we can define a 4-way channel tensor $\tilde{\mathcal{H}} \in \mathbb{C}^{N \times N \times M_R \times M_T}$ in equation (5.2) by concatenating the channel tensors for each transmit antenna, i.e., $\tilde{\mathcal{H}}_R^{(m_T)} \in \mathbb{C}^{N \times N \times M_R}$ along the 4-mode. The tensors $\tilde{\mathcal{H}}_R^{(m_T)} \in \mathbb{C}^{N \times N \times M_R}$ contain the channel vectors for the m_T -th transmit antenna and all receive antennas as defined in equation (5.3), for $m_T = 1, \dots, M_T$. Recall that these tensors have a CP structure, i.e., $\tilde{\mathcal{H}}_R^{(m_T)} = \mathcal{I}_{3,N} \times_3 \tilde{\mathbf{H}}_R^{(m_T)}$, for $m_T = 1, \dots, M_T$. The matrices $\tilde{\mathbf{H}}_R^{(m_T)}$ ($m_T = 1, \dots, M_T$) are defined in equation (5.4). The tensors $\tilde{\mathcal{H}}_R^{(m_T)}$ and the matrices $\tilde{\mathbf{H}}_R^{(m_T)}$ are depicted in Fig. 5.2.

Hence, the 4-way channel tensor is

$$\begin{aligned}\tilde{\mathcal{H}} &= \left[\tilde{\mathcal{H}}_{\text{R}}^{(1)} \quad \sqcup_4 \quad \tilde{\mathcal{H}}_{\text{R}}^{(2)} \quad \sqcup_4 \quad \dots \quad \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \right] \\ &= \left[\tilde{\mathcal{H}}_{\text{R}}^{(1)} \quad \sqcup_4 \quad \tilde{\mathcal{H}}_{\text{R}}^{(2)} \quad \sqcup_4 \quad \dots \quad \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \right] \times_4 \mathbf{I}_{M_{\text{T}}}\end{aligned}$$

We can rewrite this concatenation by means of an outer product with a pinning vector $\mathbf{e}_{m_{\text{T}}}$. Moreover, if we substitute the CP structure of the tensor $\tilde{\mathcal{H}}_{\text{R}}^{(m_{\text{T}})}$, we get

$$\tilde{\mathcal{H}} = \sum_{m_{\text{T}}=1}^{M_{\text{T}}} \tilde{\mathcal{H}}_{\text{R}}^{(m_{\text{T}})} \circ \mathbf{e}_{m_{\text{T}}} = \sum_{m_{\text{T}}=1}^{M_{\text{T}}} \left(\mathcal{I}_{3,N} \times_1 \mathbf{I}_N \times_2 \mathbf{I}_N \times_3 \tilde{\mathcal{H}}_{\text{R}}^{(m_{\text{T}})} \right) \circ \mathbf{e}_{m_{\text{T}}}.$$

Now, replacing the outer product by an n -mode product, we have

$$\tilde{\mathcal{H}} = \sum_{m_{\text{T}}=1}^{M_{\text{T}}} \mathcal{D} \times_1 \mathbf{I}_N \times_2 \mathbf{I}_N \times_3 \tilde{\mathcal{H}}_{\text{R}}^{(m_{\text{T}})} \times_4 \mathbf{e}_{m_{\text{T}}}, \quad (\text{B.5})$$

where $\mathcal{D}_{(\cdot,\cdot,\cdot,1)} = \mathcal{I}_{3,N}$. Note that the tensor $\mathcal{D} \in \mathbb{R}^{N \times N \times N \times 1}$ is a 4-way tensor, but its 4-mode is a singleton dimension. We can define this tensor in terms of a Kronecker product, which yields $\mathcal{D} = \mathcal{I}_{4,1} \otimes \mathcal{I}_{3,N}$. Equation (B.5) represents a very special BTD, where the block terms are equivalent in all modes, but the 3-mode and the 4-mode. As shown in Fig. 2.15, we can replace the sum in (B.5) with a block diagonal core tensor and factor matrices partitioned accordingly.

$$\begin{aligned}\tilde{\mathcal{H}} &= \text{blkdiag}(\mathcal{I}_{4,1} \otimes \mathcal{I}_{3,N})_{m_{\text{T}}=1}^{M_{\text{T}}} \times_1 \begin{bmatrix} \mathbf{I}_N & \dots & \mathbf{I}_N \end{bmatrix} \times_2 \begin{bmatrix} \mathbf{I}_N & \dots & \mathbf{I}_N \end{bmatrix} \\ &\quad \times_3 \begin{bmatrix} \tilde{\mathcal{H}}_{\text{R}}^{(1)} & \dots & \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \end{bmatrix} \times_4 \begin{bmatrix} \mathbf{e}_1 & \dots & \mathbf{e}_{M_{\text{T}}} \end{bmatrix}\end{aligned}$$

Next, we rewrite the block diagonal structure and the partitioned factor matrices using Kronecker products

$$\tilde{\mathcal{H}} = (\mathcal{I}_{4,M_{\text{T}}} \otimes \mathcal{I}_{3,N}) \times_1 (\mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N) \times_2 (\mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N) \times_3 \underbrace{\begin{bmatrix} \tilde{\mathcal{H}}_{\text{R}}^{(1)} & \dots & \tilde{\mathcal{H}}_{\text{R}}^{(M_{\text{T}})} \end{bmatrix}}_{\tilde{\mathcal{H}}} \times_4 \mathbf{I}_{M_{\text{T}}}. \quad (\text{B.6})$$

This last equation explicitly reveals the structure of the channel tensor $\tilde{\mathcal{H}}$. Exploiting this structure, we can define any of the tensor unfoldings.

For the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[2,4])}$, from equation (B.6), we get

$$[\tilde{\mathcal{H}}]_{([1,3],[2,4])} = [\tilde{\mathcal{H}} \otimes (\mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N)] \underbrace{[\mathcal{I}_{4,M_{\text{T}}} \otimes \mathcal{I}_{3,N}]_{([1,3],[2,4])} [\mathbf{I}_{M_{\text{T}}} \otimes \mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N]}_{*} \quad (\text{B.7})$$

Considering the results from Appendix B.2, we have

$$[\mathcal{I}_{4,M_{\text{T}}} \otimes \mathcal{I}_{3,N}]_{([1,3],[2,4])} [\mathbf{I}_{M_{\text{T}}} \otimes \mathbf{1}_{M_{\text{T}}}^{\text{T}} \otimes \mathbf{I}_N] = \mathbf{I}_{NM_{\text{T}}} \diamond \mathbf{I}_{NM_{\text{T}}}$$

for the $*$ part in equation (B.7). Recognize that $\mathbf{I}_{NM_T} \diamond \mathbf{I}_{NM_T} = \mathbf{J}_{NM_T}$ is the selection matrix that converts a Kronecker product into a Khatri-Rao as shown in property (2.29). Using this property, equation (B.7) becomes

$$[\tilde{\mathcal{H}}]_{([1,3],[2,4])} = \tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N). \quad (\text{B.8})$$

Moreover, the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[4,2])}$ can also be derived directly from equation (B.6). However, to simplify the final result is not straightforward because N is the faster rising index along the columns of the factor matrix $\tilde{\mathbf{H}}$ in equation (B.6). On the other hand, M_T varies faster than N along the columns in the generalized unfolding $[\tilde{\mathcal{H}}]_{([1,3],[4,2])}$. Therefore, we derive this generalized unfolding by means of a permutation matrix $\mathbf{P} \in \mathbb{R}^{N_{M_T} \times M_T N}$. The permutation matrix \mathbf{P} reorders the columns such that the faster increasing index is M_T instead of N and is defined as $[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = [\tilde{\mathcal{H}}]_{([1,3],[2,4])} \cdot \mathbf{P}$. Hence,

$$[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \mathbf{P}. \quad (\text{B.9})$$

Considering that the permutation matrix \mathbf{P} reorders the columns in equation (B.9) and the Khatri-Rao product is a column-wise operator (Khatri-Rao product is column-wise Kronecker product), the following equality holds

$$[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = [\tilde{\mathbf{H}} \diamond (\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N)] \cdot \mathbf{P} = [\tilde{\mathbf{H}} \cdot \mathbf{P}] \diamond [(\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \cdot \mathbf{P}].$$

The permutation matrix for $M_T = 2$ and $N = 3$ is given by

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Finally, using $(\mathbf{1}_{M_T}^T \otimes \mathbf{I}_N) \cdot \mathbf{P} = (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T)$ and defining $\bar{\mathbf{H}} = \tilde{\mathbf{H}} \cdot \mathbf{P}$, for the generalized unfolding $([1,3],[4,2])$, we get

$$[\tilde{\mathcal{H}}]_{([1,3],[4,2])} = \bar{\mathbf{H}} \diamond (\mathbf{I}_N \otimes \mathbf{1}_{M_T}^T).$$

Bibliography

Publications and Technical Documents as Co- or First Author

- [CNA⁺16] S. Cheema, K. Naskovska, M. Attar, B. Zafar, and M. Haardt, “Performance comparison of space time block codes for different 5G air interface proposals,” in *Proc. of the 20th International ITG Workshop on Smart Antennas (WSA)*, March 2016, pp. 229–235.
- [CNH⁺18] Y. Cheng, K. Naskovska, M. Haardt, T. Götz, and J. Haueisen, “A new coupled PARAFAC2 decomposition for joint processing of somatosensory evoked magnetic fields and somatosensory evoked electrical potentials,” in *Proc. 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018.
- [NCdAH18] K. Naskovska, Y. Cheng, A. L. F. de Almeida, and M. Haardt, “Efficient computation of the PARAFAC2 decomposition via generalized tensor contractions,” in *Proc. 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018.
- [NCH⁺17] K. Naskovska, S. A. Cheema, M. Haardt, B. Valeev, and Y. Evdokimov, “Iterative GFDM receiver based on the PARATUCK2 tensor decomposition,” in *Proc. 21-st International ITG Workshop on Smart Antennas (WSA)*, 2017.
- [NH16] K. Naskovska and M. Haardt, “Extension of the semi-algebraic framework for approximate CP decompositions via simultaneous matrix diagonalization to the efficient calculation of coupled CP decompositions,” in *Proc. 50th Asilomar Conference on Signals, Systems, and Computers, (Pacific Grove, CA)*, 2016, pp. 1728–1732.
- [NHdA17] K. Naskovska, M. Haardt, and A. L. F. de Almeida, “Generalized tensor contraction with application to Khatri-Rao Coded MIMO OFDM systems,” in *Proc. IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 286–290.
- [NHdA18] —, “Generalized Tensor Contractions for an Improved Receiver design in MIMO-OFDM Systems,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018, pp. 3186–3190.
- [NHT⁺16] K. Naskovska, M. Haardt, P. Tichavsky, G. Chabriel, and J. Barrere, “Extension of the semi-algebraic framework for approximate CP decomposition via non-symmetric simultaneous matrix diagonalization,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 2971–2975.
- [NKHH17] K. Naskovska, A. A. Korobkov, M. Haardt, and J. Haueisen, “Analysis of the Photoc Driving Effect via joint EEG and MEG data processing based on the coupled CP decomposition,” in *Proc. 25th European Signal Processing Conference (EUSIPCO)*, 2017.
- [NLA⁺17] K. Naskovska, S. Lau, A. Aboughazala, M. Haardt, and J. Haueisen, “Joint MEG-EEG signal decomposition using the coupled SECSI framework: Validation on a

controlled experiment,” in *Proc. IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 286–290.

- [ZNH14] J. Zhang, K. Naskovska, and M. Haardt, “Tensor-based channel estimation for non-regenerative two-way relaying networks with multiple relays,” in *Proc. 48th Asilomar Conference Signals, Systems, and Computers*, 2014, pp. 591–595.
- [ZNNH15] J. Zhang, A. Nimr, K. Naskovska, and M. Haardt, “Enhanced tensor based semi-blind estimation algorithm for relay-assisted MIMO systems,” *Latent Variable Analysis and Signal Separation, Volume 9237 of the series Lecture Notes in Computer Science, in Proceedings of the 12-th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA) (E. Vincent, A. Yeredor, Z. Koldovsky, and P. Tichavsky, eds.)*, pp. 64 – 72, 2015.
-

References by Other Authors

- [AALM16] R. André, L. Albera, X. Luciani, and E. Moreau, “On JEVD of semi-definite positive matrices and CPD of nonnegative tensors,” in *Proc. of Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 2016.
- [ABS15] E. Acar, R. Bro, and A. K. Smilde, “Data Fusion in Metabolomics Using Coupled Matrix and Tensor Factorizations,” in *Proc. of the IEEE*, vol. 103, no. 9, 2015, pp. 1602–1620.
- [AKD11] E. Acar, T. G. Kolda, and D. M. Dunlavy, “All-at-once Optimization for Coupled Matrix and Tensor Factorizations,” in *Proc. Mining and Learning with Graphs*, August 2011. [Online]. Available: <http://arxiv.org/pdf/1105.3422v1>
- [AKDM11] E. Acar, T. G. Kolda, D. M. Dunlavy, and M. Mørup, “Scalable tensor factorizations with missing data,” *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41–56, 2011.
- [AMGG13] B. M. Alves, L. L. Mendes, D. A. Guimaeres, and I. S. Gaspar, “Performance of GFDM over frequency-selective channels,” in *Proc. International Workshop on Telecommunications*, May 2013.
- [ARS⁺13] E. Acar, M. A. Rasmussen, F. Savorani, T. Næs, and R. Bro, “Understanding data fusion within the framework of coupled matrix and tensor factorizations,” *Chemometrics and Intelligent Laboratory Systems*, vol. 129, pp. 53–63, 2013.
- [BAC⁺14] H. Becker, L. Albera, P. Comon, M. Haardt, G. Birot, F. Wendling, M. Gavaret, C. G. Bnar, and I. Merlet, “EEG extended source localization: tensor-based vs. conventional methods,” *NeuroImage*, vol. 96, pp. 143–157, 2014.
- [BAK99] R. Bro, C. A. Andersson, and H. A. L. Kiers, “PARAFAC2—Part II. Modeling chromatographic data with retention time shifts,” *Journal of Chemometrics*, vol. 13, no. 3-4, pp. 295–309, 1999.
- [BCA12] H. Becker, P. Comon, and L. Albera, “Tensor-based preprocessing of combined EEG/MEG data,” in *Proc. 20-th European Signal Processing Conference (EU-SIPCO)*, October 2012.
- [BCS⁺16] E. R. Balda, S. A. Cheema, J. Steinwandt, M. Haardt, A. Weiss, and A. Yeredor,
-

- “First-Order Perturbation Analysis of Low-Rank Tensor Approximations Based on the Truncated HOSVD,” in *Proc. of 50th Asilomar Conference Signals, Systems, and Computers*, 2016, pp. 1723–1727.
- [BCW⁺17] E. R. Balda, S. A. Cheema, A. Weiss, A. Yeredor, and M. Haardt, “Perturbation analysis of joint eigenvalue decomposition algorithms,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 3101–3105.
- [BdJ97] R. Bro and S. de Jong, “A fast non-negativity-constrained least squares algorithm,” *Journal of Chemometrics*, vol. 11, no. 5, pp. 393–401, 1997.
- [Bel60] R. Bellman, *Introduction To Matrix Analysis*. New York: McGraw-Hill, 1960.
- [BHS07] R. Bro, R. A. Harshman, and N. D. Sidiropoulos, “A new model for multi-way rank-deficient data,” *Journal of Chemometrics*, 2007.
- [BHSL09] R. Bro, R. A. Harshman, N. D. Sidiropoulos, and M. E. Lundy, “Modeling multiway data with linearly dependent loadings,” *Journal of Chemometrics*, vol. 23, no. 7, pp. 324–340, 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, no. 5, pp. 274–286, 2003.
- [BKS⁺12] B. W. Bader, T. G. Kolda, J. Sun, D. M. Acar, Evrim Dunlavy, E. C. Chi, and J. Mayo, “Tensor Toolbox version 2.6, Sandia National Laboratories,” 2012. [Online]. Available: <http://www.sandia.gov/~tgkolda/TensorToolbox/>
- [BLM03] I. Barhumi, G. Leus, and M. Moonen, “Optimal training design for MIMO OFDM systems in mobile wireless channels,” *IEEE Transactions Signal Processing*, vol. 5, pp. 1615–1624, 2003.
- [BMG13] J. A. Bazerque, G. Mateos, and G. B. Giannakis, “Rank Regularization and Bayesian Inference for Tensor Completion and Extrapolation,” *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5689–5703, 2013.
- [Bre78] J. Brewer, “Kronecker products and matrix calculus in system theory,” *IEEE Transactions on Circuits and Systems*, vol. 25, no. 9, pp. 772–781, 1978.
- [Bro97] R. Bro, “PARAFAC. Tutorial and applications,” *Chemometrics and Intelligent Laboratory Systems*, vol. 38, no. 2, pp. 149–171, 1997.
- [Bro98] —, “Multi-way Analysis in the Food Industry Models, Algorithms, and Applications,” *PhD thesis, University of Amsterdam*, 1998.
- [BSG99] R. Bro, N. D. Sidiropoulos, and G. B. Giannakis, “A fast least squares algorithm for separating trilinear mixtures,” in *Proc. International Workshop on Independent Component Analysis and Blind Signal Separation*, January 1999, pp. 11–15.
- [Car91] J.-F. Cardoso, “Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors,” in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1991, pp. 3109–3112.
- [CB12] G. Chabriel and J. Barrere, “A direct algorithm for nonorthogonal approximate joint diagonalization,” *IEEE Transactions on Signal Processing*, vol. 60, pp. 39–47, January 2012.
- [CBC⁺17] S. A. Cheema, E. R. Balda, Y. Cheng, M. Haardt, A. Weiss, and A. Yeredor,

-
- “First-Order Perturbation Analysis of the SECSI Framework for the Approximate CP Decomposition of 3-D Noise-Corrupted Low-Rank Tensors,” 2017. [Online]. Available: <http://arxiv.org/pdf/1710.06693v1>
- [CBW⁺17] S. A. Cheema, E. R. Balda, A. Weiss, A. Yeredor, and M. Haardt, “Analytical performance analysis of the semi-algebraic framework for approximate CP decompositions via simultaneous matrix diagonalizations (SECSI),” in *Proc. 51st Asilomar Conference on Signals, Systems, and Computers*, October 2017, pp. 703–707.
- [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, no. 3, pp. 283–319, 1970.
- [CC13a] C. Caiafa and A. Cichocki, “Computing Sparse Representations of Multidimensional Signals Using Kronecker Bases,” *Neural Computation*, vol. 25, pp. 186–220, January 2013.
- [CC13b] —, “Multidimensional Compressed Sensing and thier Applications,” *WIREs Data Mining and Knowledge Discovery*, vol. 3, pp. 355–280, October 2013.
- [CCG17] J. Cohen, P. Comon, and N. Gillis, “Some theory on Non-negative Tucker Decomposition,” in *Proc. 13th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA 2017)*, 2017.
- [CCH⁺18] Y. Cheng, S. A. Cheema, M. Haardt, A. Weiss, and A. Yeredor, “First-order perturbation analysis of SECSI with generalized unfolding,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 3196–3200.
- [CFC15] J. E. Cohen, R. C. Farias, and P. Comon, “Fast Decomposition of Large Nonnegative Tensors,” *IEEE Signal Processing Letters*, vol. 22, no. 7, pp. 862–866, 2015.
- [CFC16] —, “Joint tensor compression for coupled canonical polyadic decompositions,” in *Proc. 24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 2285–2289.
- [CG18] J. E. Cohen and N. Gillis, “Dictionary-based Tensor Canonical Polyadic Decomposition,” 2018. [Online]. Available: <http://arxiv.org/pdf/1704.00541v1>
- [CGC79] W. A. Cobb, R. J. Guiloff, and J. Cast, “Breach rhythm. The EEG related to skull defects,” *Electroencephalography and Clinical Neurophysiology*, vol. 47, pp. 251–271, 1979.
- [CGLM08] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain, “Symmetric Tensors and Symmetric Tensor Rank,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1254–1279, 2008.
- [CH13] Y. Cheng and M. Haardt, “Robust multi-dimensional model order estimation in the presence of brief sensor failures,” in *Proc. 18-th International Conference on Digital Signal Processing (DSP)*, July 2013, pp. 412–416.
- [CH19] —, “Enhanced direct fitting algorithms for PARAFAC2 with algebraic ingredients,” *IEEE Signal Processing Letters*, 2019.
- [CHGH18] Y. Cheng, M. Haardt, T. Götz, and J. Haueisen, “Using PARAFAC2 for multi-way component analysis of somatosensory evoked magnetic fiels and somatosensory evoked potentials,” in *Proc. IEEE 10th Sensor Array and Multichannel Signal Pro-*
-

- cessing Workshop (SAM), 2018.
- [Cic14] A. Cichocki, “Era of big data processing: A new approach via tensor networks and tensor decompositions,” *arXiv:1403.2048 [cs.ET]*, 2014.
- [CKM⁺14] G. Chabriel, M. Kleinstaubert, E. Moreau, H. Shen, P. Tichavsky, and A. Yeredor, “Joint matrices decompositions and blind source separation: A survey of methods, identification, and applications,” *IEEE Signal Processing Magazine*, vol. 31, pp. 34–43, May 2014.
- [CL11] P. Comon and L.-H. Lim, “Sparse Representations and Low-Rank Tensor Approximation,” *hal-00703494*, 2011.
- [CLdA09] P. Comon, X. Luciani, and A. L. F. de Almeida, “Tensor decompositions, alternating least squares and other tales,” *Journal of Chemometrics*, vol. 23, no. 7-8, pp. 393–405, 2009.
- [CLK⁺15] F. Cong, Q. Lin, L. Kuang, X. Gong, P. Astikainen, and T. Ristaniemi, “Tensor decomposition of EEG signals: A brief review,” *Journal of Neuroscience Methods*, 2015.
- [CMP⁺15] A. Cichocki, D. Mandic, A. Phan, C. Caiafa, G. Zhou, Q. Zhao, and L. De Lathauwer, “Tensor decompositions for signal processing applications: From two-way to multiway,” *IEEE Signal Processing Magazine*, vol. 32, pp. 145–163, 2015.
- [Com14] P. Comon, “Tensors: A brief introduction,” *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 44–53, 2014.
- [Cra46] H. Cramer, *Mathematical Methods of Statistics*. Princeton University Press, 1946.
- [CRKH14] Y. Cheng, F. Roemer, O. Khatib, and M. Haardt, “Tensor subspace Tracking via Kronecker structured projections (TeTraKron) for time-varying multidimensional harmonic retrieval,” *EURASIP Journal on Advances in Signal Processing*, vol. 123, pp. 1687–6180, 2014.
- [CSH18] J. Charlier, R. State, and J. Hilger, “Non-negative Paratuck2 Tensor Decomposition Combined to LSTM Network for Smart Contracts Profiling,” in *Proc. IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2018, pp. 74–81.
- [CZPSI09] A. Cichocki, R. Zdunek, A. H. Phan, and A. Shun-Ichi, *Nonnegative Matrix and Tensor Factorizations*. John Wiley and Sons, Ltd, 2009.
- [dAF13a] A. L. F. de Almeida and G. Favier, “Double Khatri-Rao Space-Time-Frequency coding using semi-blind PARAFAC based receiver,” *IEEE Signal Processing Letters*, vol. 20, pp. 471–474, 2013.
- [dAF13b] —, “Unified tensor model for space-frequency spreading-multiplexing (SFSM) MIMO communication systems,” *EURASIP Journal on Advances in Signal Processing*, vol. 48, 2013.
- [dAFM08] A. L. F. de Almeida, G. Favier, and J. C. M. Mota, “A Constrained Factor Decomposition With Application to MIMO Antenna Systems,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2429–2442, 2008.
- [dAFX13] A. L. F. de Almeida, G. Favier, and L. R. Ximenes, “Space-time-frequency (STF) MIMO communication systems with blind receiver based on a generalized PARATUCK2 model,” *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp.

-
- 1895–1909, 2013.
- [dCFR18] M. N. da Costa, G. Favier, and J. M. T. Romano, “Tensor modelling of MIMO communication systems with performance analysis and Kronecker receivers,” *Signal Processing*, vol. 58, pp. 304–316, Apr. 2018.
- [dCHRDG07] 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 Conference on Signals, Systems, and Computers*, November 2007, pp. 412–416.
- [dCRHdS13] J. P. C. L. da Costa, F. Roemer, M. Haardt, and R. T. de Sousa, “Multi-dimensional model order selection,” *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 26, 2013.
- [DDL13a] I. Domanov and L. De Lathauwer, “On the Uniqueness of the Canonical Polyadic Decomposition of third-order tensors — Part I: Basic results and uniqueness of one factor matrix,” *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 855–875, 2013.
- [DDL13b] —, “On the Uniqueness of the Canonical Polyadic Decomposition of third-order tensors — Part II: Uniqueness of the overall decomposition,” *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 876–903, 2013. [Online]. Available: <http://arxiv.org/pdf/1301.4603v1>
- [DL05] L. De Lathauwer, “Parallel factor analysis by means of simultaneous matrix decompositions,” in *Proc. First IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2005)*, December 2005, pp. 125–128.
- [DL08a] —, “Decompositions of a Higher-Order Tensor in Block Terms—Part I: Lemmas for Partitioned Matrices,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1022–1032, 2008.
- [DL08b] —, “Decompositions of a higher-order tensor in block terms-part II: Definitions and uniqueness,” *SIAM Journal Matrix Analysis Applications (SIMAX)*, vol. 30, pp. 1033–1066, 2008.
- [DLCC07] L. De Lathauwer, J. Castaing, and J.-F. Cardoso, “Fourth-Order Cumulant-Based Blind Identification of Underdetermined Mixtures,” *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2965–2973, 2007.
- [DLDMV00a] L. De Lathauwer, B. De Moor, and J. Vandewalle, “On the best rank-1 and rank- (R_1, R_2, R_3) approximation of higherorder tensors,” *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.
- [DLDMV00b] —, “A Multilinear Singular Value Decomposition,” *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [DLN08] L. De Lathauwer and D. Nion, “Decompositions of a Higher-Order Tensor in Block Terms—Part III: Alternating Least Squares Algorithms,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1067–1083, 2008.
- [dS91] F. L. da Silva, “Neural mechanisms underlying brain waves: from neural membranes to networks,” *Linguistics*, vol. 29, no. 6, 1991.
- [dSCdA15a] A. P. da Silva, P. Comon, and A. L. F. de Almeida, “An iterative deflation algorithm for exact CP tensor decomposition,” in *Proc. International Conference on Acoustics*,
-

- Speech and Signal Processing (ICASSP)*, 2015, pp. 3961–3965.
- [dSCdA15b] ———, “Rank-1 Tensor Approximation Methods and Application to Deflation,” 2015. [Online]. Available: <http://arxiv.org/pdf/1508.05273v1>
- [EMZF16] S. Ehsanfar, M. Matthe, D. Zhang, and G. Fettweis, “A study of pilot-aided channel estimation in MIMO-GFDM systems,” in *Proc. 20th International ITG Workshop on Smart Antennas (WSA)*, March 2016.
- [FB11] B. Farhang-Boroujeny, “OFDM versus filter bank multicarrier,” *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 92 – 112, 2011.
- [FCC16] R. C. Farias, J. E. Cohen, and P. Comon, “Exploring multimodal data fusion through joint decompositions with flexible couplings,” *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4830–4844, 2016. [Online]. Available: <http://arxiv.org/pdf/1505.07717v3>
- [FdA14a] G. Favier and A. L. F. de Almeida, “Tensor space-time-frequency coding with semi-blind receivers for MIMO wireless communication systems,” *IEEE Transactions on Signal Processing*, vol. 62, no. 22, pp. 5987–6002, 2014.
- [FdA14b] ———, “Overview of constrained PARAFAC models,” *EURASIP Journal on Advances in Signal Processing*, no. 1, 2014.
- [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)*, May 2006, pp. 1137–1140.
- [FGC18] R. C. Farias, J. H. D. M. Goulart, and P. Comon, “Coherence constrained alternating least squares,” in *Proc. 26th European Signal Processing Conference (EUSIPCO)*, September 2018, pp. 613–617.
- [FMD15] A. Farhang, N. Marchetti, and L. Doyle, “Low complexity transceiver design for GFDM,” *arXiv:1501.02940 [cs.IT]*, 2015.
- [HAK⁺17] X. Han, L. Albera, A. Kachenoura, L. Senhadji, and H. Z. Shu, “Low Rank Canonical Polyadic Decomposition of Tensors Based on Group Sparsity,” in *Proc. 25th European Signal Processing Conference (EUSIPCO)*, 2017.
- [Har72] R. A. Harshman, “PARAFAC2: Mathematical and Technical Notes,” *CLA Working Papers in Phonetics (University Microfilms, Ann Arbor, Michigan, No. 10,085)*, pp. 30–44, 1972.
- [Har78] ———, “Models for analysis of asymmetrical relationships among n objects or stimuli,” in *First Joint Meeting of the Psychometric Society and the Society for Mathematical Psychology, McMaster University, Hamilton, Ontario*, 1978.
- [HCS⁺14] B. Hunyadi, D. Camps, L. Sorber, W. van Paesschen, M. D. Vos, S. van Huffel, and L. de Lathauwer, “Block term decomposition for modelling epileptic seizures,” *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, p. 540, 2014.
- [Her01] C. S. Herrmann, “Human EEG responses to 1-100Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena,” *Experimental Brain Research*, vol. 137, no. 3-4, pp. 346–353, 2001.
- [HGS⁺11] A. Halbleib, M. Gratkowski, K. Schwab, C. Ligges, H. Witte, and J. Haueisen, “Topographic Analysis of Engagement and Disengagement of Neural Oscillators in Photic Driving: A Combined Electroencephalogram/Magnetoencephalogram

-
- Study,” *Journal of Clinical Neurophysiology*, vol. 29, no. 1, pp. 33–41, 2011.
- [HL96] R. A. Harshman and M. E. Lundy, “Uniqueness proof for a family of models sharing features of Tucker’s three-mode factor analysis and PARAFAC/CANDECOMP,” *Psychometrika*, vol. 61, no. 1, pp. 133–154, 1996.
- [HN98] J. L. Hintze and R. D. Nelson, “Violin plots: A box plot-density trace synergism,” *The American Statistician*, vol. 52, no. 2, pp. 181–184, 1998.
- [HRDG08] 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 Transactions on Signal Processing*, vol. 56, pp. 3198–3213, 2008.
- [HYSH06] D. Hu, L. Yang, Y. Shi, and L. He, “Optimal pilot sequence design for channel estimation in MIMO OFDM systems,” *IEEE Communications Letters*, vol. 10, pp. 1–3, 2006.
- [HYW⁺09] T. Hwang, C. Yang, G. Wu, S. Li, and G. Ye Li, “OFDM and Its Wireless Applications: A Survey,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1673–1694, 2009.
- [ITU97] ITU-R Recommendation M.1225, “Guidelines for evaluation of radio transmission technologies for IMT-2000,” Recommendation, 1997.
- [KB09] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM Review*, vol. 51, pp. 455–500, 2009.
- [KCdA17] E. Kofidis, C. Chatzichristos, and A. L. F. de Almeida, “Joint Channel Estimation / Data Detection in MIMO-FBMC/OQAM Systems - A Tensor-Based Approach,” in *Proc. 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 445–449.
- [KDRP96] W. Klimesch, M. Doppelmayr, H. Russegger, and T. Pachinger, “Theta band power in the human scalp eeg and the encoding of new information,” *Neuro Report*, vol. 7, pp. 1235–1240, 1996.
- [Kie93] H. A. Kiers, “An alternating least squares algorithm for PARAFAC2 and three-way DEDICOM,” *Computational Statistics & Data Analysis*, vol. 16, no. 1, pp. 103–118, 1993.
- [Kol15] T. G. Kolda, “Numerical optimization for symmetric tensor decomposition,” *Mathematical Programming*, vol. 151, no. 1, pp. 225–248, 2015.
- [KPVLDs02] S. Kalitzin, J. Parra, D. N. Velis Lopes, and F. H. da Silva, “Enhancement of phase clustering in the EEG/MEG gamma frequency band anticipates transitions to paroxysmal epileptic form activity in epileptic patients with known visual sensitivity,” *IEEE Transactions Biomedical Engineering*, vol. 49, pp. 1279 – 1286, 2002.
- [Kru77] J. B. Kruskal, “Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics,” *Linear Algebra and Its Applications*, vol. 18, pp. 95–138, 1977.
- [Kru89] —, “Rank, decomposition, and uniqueness for 3-way and N-way arrays,” *Multitway data analysis published by North-Holland Publishing Co. Amsterdam, The Netherlands*, pp. 7–18, 1989.
- [KTBB99] H. A. L. Kiers, J. M. F. Ten Berge, and R. Bro, “PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model,” *Journal of Chemometrics*, vol. 13, no.
-

- 3-4, pp. 275–294, 1999.
- [LA10] X. Luciani and L. Albera, “Joint Eigenvalue Decomposition using Polar Matrix Factorization,” in *Proc. 4-th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, 2010, pp. 612–619.
- [LA11] —, “Semi-algebraic canonical decomposition of multi-way arrays and joint eigenvalue decomposition,” in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2011, pp. 4104–4107.
- [LA14] —, “Canonical Polyadic Decomposition based on joint eigenvalue decomposition,” *Chemometrics and Intelligent Laboratory Systems*, vol. 132, pp. 152–167, 2014.
- [LC10] L.-H. Lim and P. Comon, “Multiarray Signal Processing: Tensor decomposition meets compressed sensing,” *Comptes Rendus Mécanique*, vol. 338, no. 6, pp. 311–320, 2010. [Online]. Available: <http://arxiv.org/pdf/1002.4935v3>
- [LdCS⁺16] K. Liu, J. P. C. L. da Costa, H. C. So, L. Huang, and J. Yea, “Detection of Number of Components in CANDECOMP/PARAFAC models via Information Theoretic Criterion,” *Digital Signal Processing*, vol. 51, pp. 110–123, 2016.
- [LdCSdA13] K. Liu, J. P. C. L. da Costa, H. C. So, and A. L. F. de Almeida, “Semi-blind receivers for joint symbol and channel estimation in space-time-frequency MIMO-OFDM systems,” *IEEE Transactions Signal Processing*, vol. 61, no. 21, pp. 5444–5457, 2013.
- [LFH14] S. Lau, L. Flemming, and J. Haueisen, “Magnetoencephalography signals are influenced by skull defects,” *Clinical Neurophysiology*, vol. 125, pp. 1653 – 1662, 2014.
- [LGF⁺16] S. Lau, D. Guellmar, L. Flemming, D. B. Grayden, M. J. Cook, C. H. Wolters, and J. Haueisen, “Skull defects in finite element head models for source reconstruction from magnetoencephalography signals,” *Frontiers in Neuroscience*, vol. 10, article 141, 2016.
- [Liu99] S. Liu, “Matrix results on the Khatri-Rao and Tracy-Singh products,” *Linear Algebra and its Applications*, vol. 289, pp. 267–277, 1999.
- [LPdA09] V. V. Lazarev, A. Pontes, and L. de Azevedo, “EEG photic driving: Right-hemisphere reactivity deficit in childhood autism. a pilot study,” *International Journal of Psychophysiology*, vol. 71, pp. 177 – 183, 2009.
- [LS01] X. Liu and N. D. Sidiropoulos, “Cramer-Rao lower bounds for low-rank decomposition of multidimensional arrays,” *IEEE Transactions on Signal Processing*, vol. 49, no. 9, pp. 2074–2086, 2001.
- [LS15] A. P. Liavas and N. D. Sidiropoulos, “Parallel Algorithms for Constrained Tensor Factorization via the Alternating Direction Method of Multipliers,” *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5450–5463, 2015. [Online]. Available: <http://arxiv.org/pdf/1409.2383v2>
- [LSSd01] V. V. Lazarev, D. M. Simpson, B. M. Schubsky, and L. C. de Azevedo, “Photic driving in the electroencephalogram of children and adolescents: harmonic structure and relation to the resting state,” *Brazilian Journal of Medical and Biological Research*, vol. 34, pp. 1573–1584, 2001.
- [LT08] S. Liu and G. Trenkler, “Hadamard, Khatri-Rao, Kronecker and other matrix products,” *International Journal of Information and Systems Sciences*, vol. 4, no. 1, pp.

-
- 160–177, 2008.
- [LTS⁺10] J. W. Lee, N. Tanaka, H. Shiraishi, T. A. Milligan, B. A. Dworetzky, S. Khoshbin, S. M. Stufflebeam, and E. Bromfield, “Evaluation of postoperative sharp waveforms through EEG and magnetoencephalography,” *Journal of Clinical Neurophysiology*, vol. 27, pp. 7–11, 2010.
- [McC87] P. McCullagh, *Tensor Methods in Statistics*. Chapman and Hall London, 1987.
- [MDK⁺17] N. Michailow, R. Datta, S. Krone, M. Lentmaier, and G. Fettweis, “Generalized Frequency Division Multiplexing: A Flexible Multi-Carrier Modulation Scheme for 5th Generation Cellular Networks,” *Mobile Communications*, 2017.
- [MMG⁺14] N. Michailow, M. Matthe, I. Gaspar, A. Caldevilla, L. Mendes, A. Festag, and G. Fettweis, “Generalized frequency division multiplexing for 5th generation cellular networks,” *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3045–3061, 2014.
- [MMG15] M. Mardani, G. Mateos, and G. B. Giannakis, “Subspace Learning and Imputation for Streaming Big Data Matrices and Tensors,” *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2663–2677, 2015. [Online]. Available: <http://arxiv.org/pdf/1404.4667v1>
- [Moo20] E. H. Moore, “On the reciprocal of the general algebraic matrix,” *Bulletin of the American Mathematical Society*, vol. 26, pp. 394–395, 1920.
- [MPCB93] G. L. Mangan, O. Pellett, I. M. Colrain, and T. C. Bates, “Photic driving and personality,” *Personality and Individual Differences*, vol. 15, no. 3, pp. 329–340, 1993.
- [MSK09] X. Ma, D. Schonfeld, and A. Khokhar, “Dynamic updating and downdating matrix SVD and tensor HOSVD for adaptive indexing and retrieval of motion trajectories,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 1129–1132.
- [NAMLT16] V.-D. Nguyen, K. Abed-Meraim, and N. Linh-Trung, “Fast adaptive PARAFAC decomposition algorithm with linear complexity,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 6235–6239.
- [NBR⁺14] M. Niknazar, H. Becker, B. Rivet, C. Jutten, and P. Comon, “Blind source separation of underdetermined mixtures of event-related sources,” *Signal Processing*, vol. 101, pp. 52–64, 2014.
- [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.
- [NKH16] A. Notbohm, J. Kurths, and C. Herrmann, “Modification of brain oscillations via rhythmic light stimulation provides evidence for entrainment but not for superposition of event-related responses,” *Frontiers in Human Neuroscience*, vol. 10, 2016.
- [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, 2009.
- [Pen55] R. Penrose, “A generalized inverse for matrices,” *Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, 1955.
-

-
- [PF15] E. E. Papalexakis and C. Faloutsos, “Fast efficient and scalable Core Consistency Diagnostic for the PARAFAC decomposition for big sparse tensors,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5441–5445.
 - [PFS12] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos, “ParCube: Sparse Parallelizable Tensor Decompositions,” in *Machine Learning and Knowledge Discovery in Databases*, P. A. Flach, T. De Bie, and N. Cristianini, Eds. Springer Berlin Heidelberg, 2012, pp. 521–536.
 - [PHT⁺18] I. Podkurkov, L. Hamidullina, E. Traikov, M. Haardt, and A. Nadeev, “Tensor-based near-field localization in bistatic MIMO radar systems,” in *Proc. 22-nd International ITG Workshop on Smart Antennas (WSA)*, 2018.
 - [PMS14] E. E. Papalexakis, T. M. Mitchell, and N. D. Sidiropoulos, “Turbo-SMT Accelerating Coupled Sparse Matrix-Tensor,” in *Proc. of the 2014 SIAM International Conference on Data Mining*, 2014.
 - [PNCZ⁺16] A. I. Perez-Neira, M. Caus, R. Zakaria, D. Le Ruyet, E. Kofidis, M. Haardt, X. Mestre, and Y. Cheng, “MIMO signal processing in Offset-QAM based filter bank multicarrier systems,” *IEEE Transactions on Signal Processing*, vol. 64, pp. 5733–5762, 2016.
 - [Pon10] S. P. Ponnappalli, “Higher-Order Generalized Singular Value Decomposition: Comparative Mathematical Framework with Applications to Genomic Signal Processing,” *Doctoral Thesis, The University of Texas at Austin*, 2010.
 - [PP08] K. B. Petersen and M. S. Pedersen, “The Matrix Cookbook,” <http://matrixcookbook.com> (version 2012), 2008.
 - [PSvA11] S. P. Ponnappalli, M. A. Saunders, C. F. van Loan, and O. Alter, “A higher-order generalized singular value decomposition for comparison of global mRNA expression from multiple organisms,” *PloS one*, vol. 6, no. 12, p. e28072, 2011.
 - [PTC13] A. H. Phan, P. Tichavský, and A. Cichocki, “Low Complexity Damped Gauss-Newton Algorithms for CANDECOMP/PARAFAC,” *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 1, pp. 126–147, 2013. [Online]. Available: <http://arxiv.org/pdf/1205.2584v2>
 - [QBLH06] A. Quinlan, J.-P. Barbot, P. Larzabal, and M. Haardt, “Model Order Selection for Short Data: An Exponential Fitting Test (EFT),” *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 2006.
 - [QCL16] Y. Qi, P. Comon, and L.-H. Lim, “Uniqueness of Nonnegative Tensor Approximations,” *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 2170–2183, 2016. [Online]. Available: <http://arxiv.org/pdf/1410.8129v4>
 - [RCH08] M. Rajih, P. Comon, and R. Harshman, “Enhanced line search: A novel method to accelerate PARAFAC,” *SIAM Journal on Matrix Analysis Applications*, vol. 30, pp. 1148–1171, September 2008.
 - [RdAZ16] L. N. Ribeiro, A. L. F. de Almeida, and V. Zarzoso, “Enhanced block term decomposition for atrial activity extraction in atrial fibrillation ECG,” in *2016 IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*. IEEE, 2016, pp. 1–5.
 - [RDGD⁺15] B. Rivet, M. Duda, A. Guérin-Dugué, C. Jutten, and P. Comon, “Multimodal
-

-
- approach to estimate the ocular movements during EEG recordings: a coupled tensor factorization method,” in *Proc. 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015.
- [RH08] F. Roemer and M. Haardt, “A closed-form solution for parallel factor (PARAFAC) analysis,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2008, pp. 2365–2368.
- [RH09a] —, “Structured least squares (SLS) based enhancements of Tensor-Based channel estimation (TENCE) for Two-Way relaying with multiple antennas,” in *Proc. International ITG Workshop on Smart Antennas (WSA)*, February 2009.
- [RH09b] —, “Tensor-Based channel estimation (TENCE) for Two-Way relaying with multiple antennas and spatial reuse,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009, pp. 3641–3644.
- [RH10] —, “Tensor-Based Channel Estimation (TENCE) and Iterative Refinements for Two-Way Relaying With Multiple Antennas and Spatial Reuse,” *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5720–5735, 2010.
- [RH13a] —, “A SEmi-algebraic framework for approximate CP decomposition via Simultaneous matrix diagonalization (SECSI),” *Signal Processing*, vol. 93, pp. 2722–2738, September 2013.
- [RH13b] —, “Tensor subspace tracking via Kronecker structured projections (TeTraKron),” in *Proc. IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec. 2013.
- [Ris78] J. Rissanen, “Modeling by shortest data description,” *Automatica*, pp. 465–471, 1978.
- [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, 2012.
- [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. of the 46th Asilomar Conference on Signals, Systems, and Computers*, November 2012, pp. 2023–2027.
- [SB02] N. D. Sidiropoulos and R. S. Budampati, “Khatri-Rao Space-Time codes,” *IEEE Transactions on Signal Processing*, vol. 50, no. 10, pp. 2396–2407, 2002.
- [Sch99] P. H. Schönemann, “A generalized solution of the orthogonal procrustes problem,” *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1999.
- [SdA10] A. Stegeman and A. L. F. de Almeida, “Uniqueness conditions for constrained three-way factor decompositions with linearly dependent loadings,” *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 3, pp. 1469–1490, 2010.
- [SdAH17] B. Sokal, A. L. F. de Almeida, and M. Haardt, “Rank-One Tensor Modeling Approach to Joint Channel and Symbol Estimation in Two-Hop MIMO Relaying Systems,” in *Proc. XXXV Simpsio Brasileiro de Telecomunicacoes e Prosessamento de Sinais (SBrT 2017)*, 2017.
- [SDDL15] M. Sorensen, I. Domanov, and L. De Lathauwer, “Coupled Canonical Polyadic de-
-

- compositions and (coupled) decompositions in multilinear rank- $(L_{r,n}, L_{r,n}, 1)$ terms — part II: Algorithms,” *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 2, pp. 496–522, 2015.
- [SDDL18] —, “Coupled canonical polyadic decompositions and multiple shift-invariance in array processing,” *IEEE Transactions on Signal Processing*, vol. 66, 2018.
- [SDL15] M. Sorensen and L. De Lathauwer, “Coupled Canonical Polyadic Decompositions and (Coupled) Decompositions in Multilinear Rank- $(L_r, n, L_r, n, 1)$ Terms—Part I: Uniqueness,” *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 2, pp. 496–522, 2015.
- [SDL17a] —, “Multidimensional Harmonic Retrieval via Coupled Canonical Polyadic Decomposition—Part I: Model and Identifiability,” *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 517–527, 2017.
- [SDL17b] —, “Multidimensional Harmonic Retrieval via Coupled Canonical Polyadic Decomposition—Part II: Algorithm and Multirate Sampling,” *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 528–539, 2017.
- [SDF⁺17] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor Decomposition for Signal Processing and Machine Learning,” *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, 2017. [Online]. Available: <http://arxiv.org/pdf/1607.01668v2>
- [SFFM99] M. Speth, S. A. Fechtel, G. Fock, and H. Meyr, “Optimum receiver design for wireless broad-band systems using OFDM. I,” *IEEE Transactions on Communications*, vol. 47, no. 11, pp. 1668–1677, 1999.
- [SGB00] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, “Blind PARAFAC receivers for DS-CDMA systems,” *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 810–823, 2000.
- [SLJ⁺06] K. Schwab, C. Ligges, T. Jungmann, B. Hilgenfeld, J. Haueisen, and H. Witte, “Alpha entrainment in human electroencephalogram and magnetoencephalogram recordings,” *Neuroreport*, vol. 17, pp. 1829–1833, 2006.
- [SPF14] N. D. Sidiropoulos, E. E. Papalexakis, and C. Faloutsos, “PARallel RANdomly COM-Pressed Cubes (PARACOMP),” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.
- [SSK⁺16] C. Salchow, D. Strohmeier, S. Klee, D. Jannek, K. Schiecke, H. Witte, A. Nehorai, and J. Haueisen, “Rod Driven Frequency Entrainment and Resonance Phenomena,” *Frontiers in Human Neuroscience*, vol. 10, p. 413, 2016.
- [SvBDL13] L. Sorber, M. van Barel, and L. De Lathauwer, “Optimization-Based Algorithms for Tensor Decompositions: Canonical Polyadic Decomposition, Decomposition in Rank- $(L_r, L_r, 1)$ Terms, and a New Generalization,” *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 695–720, 2013.
- [SvBDL15] —, “Structured Data Fusion,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, pp. 586–600, 2015.
- [SWC14] F. Schaich, T. Wild, and Y. Chen, “Waveform contenders for 5G - suitability for short packet and low latency transmissions,” in *Proc. of 79th IEEE Vehicular Technology Conference (VTC Spring)*, 2014, pp. 1–5.

-
- [TPC15] P. Tichavsky, A. Phan, and A. S. Cichocki, “Two-sided diagonalization of order-three tensors,” in *Proc. 23rd European Signal Processing Conference (EUSIPCO)*, September 2015, pp. 998–1002.
- [TPK13] P. Tichavsky, A. H. Phan, and Z. Koldovsky, “Cramér-Rao-Induced Bounds for CANDECOMP/PARAFAC Tensor Decomposition,” *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp. 1986–1997, 2013.
- [Tuc63] L. R. Tucker, “Implications of factor analysis of three-way matrices for measurement of change,” in *Problems in Measuring Change*, C. W. Harris, ed., University of Wisconsin Press, vol. 14, no. 5, pp. 122–137, 1963.
- [TVP94] S. Talwar, M. Viberg, and A. Paulraj, “Blind estimation of multiple co-channel digital signals using an antenna array,” *IEEE Signal Processing Letters*, vol. 1, no. 2, pp. 29–31, 1994.
- [TVP96] —, “Blind separation of synchronous co-channel digital signals using an antenna array. I. Algorithms,” *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1184–1197, 1996.
- [VCF⁺16] M. A. Veganzones, J. E. Cohen, R. C. Farias, J. Chanussot, and P. Comon, “Non-negative Tensor CP Decomposition of Hyperspectral Data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 2577–2588, 2016.
- [VDS⁺16] N. Vervliet, O. Debals, L. Sorber, M. van Barel, and L. De Lathauwer, “Tensorlab 3.0,” March 2016.
- [vEHDLvH17] S. van Eyndhoven, B. Hunyadi, L. De Lathauwer, and S. van Huffel, “Flexible fusion of electroencephalography and functional magnetic resonance imaging: Revealing neural-hemodynamic coupling through structured matrix-tensor factorization,” in *Proc. 25th European Signal Processing Conference (EUSIPCO)*, August 2017, pp. 26–30.
- [VLP93] C. Van Loan and N. Pitsianis, “Approximation with Kronecker products,” *Linear Algebra for Large-Scale and Real-Time Applications*, pp. 293–314, 1993.
- [VLP97] C. F. Van Loan and N. Pitsianis, *Approximation with Kronecker Products, from Linear Algebra for Large Scale and Real-Time Applications*, M. Moonen, G. Golub, and B. De Moor, Eds. Kluwer Academic Pub., Dordrecht, Netherlands, 1997.
- [Wei15] M. Weis, “Multi-Dimensional Signal Decomposition Techniques for the Analysis EEG Data,” *Doctoral Thesis, Ilmenau University of Technology*, 2015.
- [WGP⁺11] M. Wacker, M. Galicki, P. Putsche, T. Milde, K. Schwab, J. Haueisen, C. Ligges, and H. Witte, “A time-variant processing approach for the analysis of alpha and gamma MEG oscillations during flicker stimulus generated entrainment,” *IEEE transactions on Biomedical engineering*, vol. 58, no. 11, pp. 3069–3077, 2011.
- [WJG⁺10] 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 Signal Processing Conference (EUSIPCO)*, 2010, pp. 696–700.
- [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. 32-th International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2010.

- [Wol99] Wolfgang Klimesch, “EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis,” *Brain Research Reviews*, pp. 169–195, 1999.
- [XFdAS14] L. R. Ximenes, G. Favier, A. L. F. de Almeida, and Y. C. B. Silva, “PARAFAC-PARATUCK semi-blind receivers for two-hop cooperative MIMO relay systems,” *IEEE Transactions on Signal Processing*, vol. 62, pp. 3604 – 3615, 2014.
- [XY13] Y. Xu and W. Yin, “A block cordinated descent method for regulized multiconvex optimization with applications to nonnegative tensor factorization and completion,” *SIAM Journal on imaging sciences*, vol. 6, no. 3, pp. 1758–1789, 2013.
- [YFLZ16] L. Yang, J. Fang, H. Li, and B. Zeng, “An Iterative Reweighted Method for Tucker Decomposition of Incomplete Multiway Tensors,” *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4817–4829, 2016. [Online]. Available: <http://arxiv.org/pdf/1511.04695v1>
- [YLC17] T. Yokota, N. Lee, and A. Cichocki, “Robust Multilinear Tensor Rank Estimation Using Higher Order Singular Value Decomposition and Information Criteria,” *IEEE Transactions on Signal Processing*, vol. 65, no. 5, pp. 1196–1206, 2017.
- [ZCJW17] L. Zou, X. Chen, X. Ji, and Z. J. Wang, “Underdetermined Joint Blind Source Separation of Multiple Datasets,” *IEEE Access*, vol. 5, pp. 7474–7487, 2017.

