A Method for Completely Positive and Nonnegative Matrix Factorization

DISSERTATION

zur Erlangung des akademischen Grades eines
Doktor der Naturwissenschaften Dr. rer. nat.

vorgelegt am Fachbereich IV
der Universität Trier

von

M. Sc. Patrick Hermann Groetzner

Trier, 2018

Eingereicht am	11.06.2018
Disputation am	06.08.2018
Gutachter	Prof. Dr. Mirjam Dür
	Prof. Dr. Florian Jarre

Zusammenfassung

Viele nicht konvexe Optimierungsprobleme können als konvexes Problem über dem Kegel der vollständig positiven Matrizen reformuliert werden, sodass für diese Reformulierung lokale und globale Optima zusammenfallen und daher keine globalen Optimierungstechniken notwendig sind. Dies ist möglich, da die Komplexität des Problems nun vollständig in der Kegelnebenbedingung enthalten ist. Daher ist es nicht verwunderlich, dass die Überprüfung der Zugehörigkeit einer Matrix zum vollständig positiven Kegel NP-schwer ist, wie in [38] gezeigt. Als Hauptresultat dieser Arbeit werden wir sehen, wie algorithmisch ein Zertifikat generiert werden kann, welches für geeignete Startwerte verifiziert, dass eine gegebene Matrix vollständig positiv ist.

Dazu werden wir zunächst einige, den vollständig positiven Kegel betreffende Fakten sehen, die im Anschluss durch einige notwendige und teilweise hinreichende Bedingungen für eine vollständig positive Matrix ergänzt werden. Als fundamentale Definition gilt hier, dass eine Matrix $A \in \mathbb{R}^{n\times n}$ vollständig positiv ist, falls es eine Zerlegungsmatrix $B \in \mathbb{R}^{n\times r}$ gibt, die eintragsweise nichtnegativ ist und die Gleichung $A = BB^T$ erfüllt. Eine solche Zerlegung liefert daher immer ein Zertifikat, welches zeigt, dass die gegebene Matrix vollständig positiv ist. Basierend auf dieser Definition werden wir einige Fakten zu diesen Zerlegungen sehen, die nichtzuletzt auch für die praktischen Anwendungen relevant sind und daher durch diese motiviert werden können.

Basierend auf diesen Zerlegungen ist es zusätzlich möglich, weitere und teilweise neue Bedingungen für vollständig positive Matrizen abzuleiten. Hier ist es insbesondere notwendig mit einer passenden Startzerlegung der Matrix zu beginnen. Wie eine solche Zerlegung generiert werden kann, wird ebenfalls gezeigt. Hier werden wir insbesondere auf orthognale Matrizen als Werkzeug zurückgreifen. So ist es insgesamt möglich, das Problem der Verifizierung der Zugehörigkeit einer Matrix zum vollständig positiven Kegel auf ein Zulässigkeitsproblem zu reduzieren. Im Detail ist es dazu notwendig, eine Matrix im Schnitt eines polyedrischen Kegels und dem nichtnegativen Orthanten zu finden. Dabei werden wir auf die auf von Neumann (cf. [95]) zurückgehende Technik der alternierenden Projektionen zurückgreifen, um eine solche Matrix zu generieren.

Für dieses Verfahren wird eine kurze Einführung und Erläuterung der Anwendung auf verschiedene Typen von Mengen gegeben. Insbesondere werden anhand von geometrischen Eigenschaften bekannte Resultate bezüglich der Konvergenz des Verfahrens und deren Geschwindigkeit gezeigt. Erweitert man die Idee der alternierenden Projektionen auf mehr als zwei Mengen, so spricht man vom zyklischen Projektions-Verfahren. Auch für diesen Ansatz werden bekannte Resultate für Unterräume und allgemeine konvexe Mengen gezeigt. Des Weiteren wird ein neues Konvergenzresultat für die zyklische Projektion zwischen transversalen Mannigfaltigkeiten hergeleitet, welches auf den Resultaten für die alternierenden Projektionen auf Mannigfaltigkeiten in [70] basiert.

Insbesondere lässt sich die Methode der alternierenden Projektionen auf semialgebraische Mengen anwenden, wie in [42] gezeigt. Dieses Resultat werden wir nutzen, um einen ersten Algorith-

mus zur Generierung von Zerlegungen von vollständig positiven Matrizen herzuleiten. Für diesen Algorithmus ist es möglich, ein lokales Konvergenzresultat zu zeigen. Insbesondere greift dieser Algorithmus jedoch auf das wiederholte Lösen von sogenannten *second order cone* Problemen zurück. Diese sind zwar in polynomieller Zeit lösbar, aber immer noch vergleichsweise rechenintensiv.

Aus diesem Grund werden wir eine modifizierte Variante dieses Algorithmus sehen, die ohne diese speziellen Probleme auskommt. Hier verlieren wir zwar das lokale Konvergenzresultat, aber numerische Experimente zeigen, dass dieser Ansatz für nahezu alle getesteten Beispiele vollständig positiver Matrizen in sehr kurzer Zeit eine Zerlegung liefert.

Neben der Generierung von Zerlegungen für vollständig positive Matrizen können die gezeigten Methoden und Verfahren auch im Kontext der sogenannten Nichtnegativen Matrix Zerlegung angewandt werden. Hier werden wir sehen, dass für die symmetrische Variante dieser Zerlegung lediglich zusätzliche niedrig-Rang Nebenbedingungen bedacht und integriert werden müssen. Für den allgemeinen, nicht symmetrischen Fall hingegen können zwar die Ansätze der Verfahren zur Generierung von Zerlegungen für vollständig postitive Matrizen verwendet werden, müssen aber auf nicht-quadratische Ausgangsmatrizen erweitert werden. Hier werden wir sehen, dass orthogonale Matrizen nicht mehr das Werkzeug der Wahl sind und entsprechend ersetzt werden müssen. Des Weiteren ist es nicht mehr möglich auf den Ansatz der alternierenden Projektionen zurückzugreifen, da die dazu notwendigen Projektionen nicht mehr berechnet werden können. Nichtsdestotrotz ist es möglich, die Ideen des modifizierten Algorithmus für vollständig positive Matrizen auch in diesem Kontext zu verwenden. Sowohl für den symmetrischen, als auch für den allgemeinen Fall der nichtnegativen Matrixzerlegung, werden wir zahlreiche numerische Experimente sehen, die die Anwendbarkeit der in dieser Arbeit generierten Algorithmen auch in diesem Kontext untermauern.

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mir bei der Anfertigung dieser Dissertation unterstützend zur Seite gestanden haben.

Ein besonderer Dank gilt hier Frau Prof. Dr. Mirjam Dür und dies nicht nur für die Betreuung dieser Doktorarbeit. Durch ihr Vertrauen in mich und das damit verbundene Angebot bei ihr tätig zu werden, war es mir erst möglich, in das bearbeitete Themengebiet einzusteigen. Sie hatte mich in meinen mathematischen Interessen bestärkt und basierend auf diesen, offene Forschungsfragen formuliert, sodass mir der Zugang zu einem mir größtenteils noch unbekannten Forschungsgebiet leichtgemacht wurde. Während der gesamten Betreuungszeit waren ein klares Ziel und perfekte Organisation stetige Begleiter. Regelmäßige Treffen und fachliche Diskussionen waren ein Garant für eine produktive Arbeitsatmosphäre und haben diese Arbeit erst ermöglicht. Wann immer Fragen aufkamen oder Unklarheiten ein Vorankommen hemmten, hatte Frau Dür ein offenes Ohr und produktive Anregungen. Nicht zuletzt machten ausführliche Hilfestellungen bei der Formulierung mathematischer Inhalte und Feedback zu Vorträgen oder Ausarbeitungen eine Weiterentwicklung meinerseits möglich.

Zudem wurde mir insbesondere durch zahlreiche internationale Konferenzen die Möglichkeit der globalen Kontaktknüpfung eröffnet, sodass hier zukünftige Forschungskooperationen entstehen können. In unmittelbarer Zukunft freue ich mich weiterhin mit Frau Dür arbeiten zu können und weiß ihre Förderung und Unterstützung in jeder Phase meiner potentiellen wissenschaftlichen Karriere sehr zu schätzen.

Des Weiteren gilt mein Dank Herrn Prof. Dr. Florian Jarre für die Bereitschaft als Zweitgutachter für diese Dissertation zu fungieren, wie auch für die Möglichkeit, den Code seines in Zusammenarbeit mit Frau Prof. Dr. Katrin Schmallowsky entstandenen Ansatzes zu nutzen.

Außerdem möchte ich der *German-Israeli Foundation for Scientific Research and Development (GIF)* im Projekt Matrices (G-18-304.2/2011) sowie der *Deutschen Forschungsgemeinschaft (DFG)* und damit verbunden den Verantwortlichen des Graduiertenkollegs 2126 *Algorithmic Optimization (ALOP)* an der Universität Trier für die finanzielle Unterstützung während meiner Promotionszeit danken. Durch eine überzeugende Organisation, ein vielfältiges wissenschaftliches Programm und der Möglichkeit zur Teilnahme an internationalen Konferenzen, wurden mir auch als assoziiertes Mitglied im Graduiertenkolleg ALOP stets Optionen zur Weiterbildung und Kontaktknüpfung ermöglicht.

Ein weiterer Dank geht an das SIAM Student Chapter in Trier, welches durch seine zahlreichen wissenschaftlichen und nicht wissenschaftlichen Aktivitäten immer zu einer produktiven Arbeitsatmosphäre beigetragen hat.

Nicht zuletzt möchte ich mich bei meiner Mutter bedanken, die mir während meines gesamten Lebens immer mit Rat und Tat zur Seite stand. Sie hat mir insbesondere mein Studium ermöglicht, sodass ich die Mathematik und ihre Facetten für mich entdecken und somit den Weg der Promotion einschlagen konnte.

Ferner möchte ich den Korrekteuren dieser Arbeit für die hilfreichen Kommentare und Anregungen danken. Dies sind im Detail: Claudia Adams, Philipp Annen, Dana Becker, David Geulen, Simone Hesse, Daniel Hoffmann, Asim Nomani, Thorben Schlierkamp und Robin Schrecklinger.

Contents

1 Short Summary **1**

2 Introduction **3**

2.1 The Copositive and the Completely Positive Cone 3
2.2 Complexity and Theoretical Certificates for Complete Positivity 6
2.3 The Interior of the Completely Positive Cone 10
2.4 The cp-rank and the cp^+-rank for Completely Positive Matrices 12
2.5 Matrices of High cp-rank 15
2.6 The Boundary of the Completely Positive Cone 17
2.7 Conic Programming and Applications 18

3 Factorizations for Completely Positive Matrices **23**

3.1 Related Work 23
3.2 CP-Factorizations are not Unique 24
3.3 The Role of Orthogonal Matrices 25
3.4 Nearly Positive Matrices 33
3.5 Further Conditions for Complete Positivity 34
3.6 Generating Initial Factorizations of Arbitrary Order 35
3.7 Generating Factorizations for Matrices in the Interior via Maximization Problems 37

4 The Factorization Problem as a Nonconvex Feasibility Problem **41**

4.1 Feasibility Problems to Verify Complete Positivity 41
4.2 Feasibility Problems for Matrices in the Interior of the Completely Positive Cone 44

5 Alternating Projections **45**

5.1 Alternating Projections on Subspaces 45
5.2 Alternating Projections on Convex Sets 51
5.2.1 Cyclic Projections Among a Sequence of Convex Sets 54
5.2.2 Alternating Projections and the Angle Between Convex Sets 56
5.3 Alternating Projections on Manifolds 60
5.4 Cyclic Projections Among a Sequence of Manifolds 69
5.5 Alternating Projections on Closed Sets and on Semialgebraic Sets 75

6 Applying Alternating Projections to Construct $\mathcal{CP}$-Factorizations **79**

6.1 An Alternating Projections Approach for $\mathcal{CP}$-Factorizations 79
6.2 Modifying the Alternating Projections Method 82
6.3 Algorithms for Matrices in the Interior of the Completely Positive Cone 85

7 Numerical Results **89**
7.1 A Specifically Structured Example in Different Dimensions 89
7.2 The Influence of the Parameter r 91
7.3 A Low cp-rank Matrix Without Known Factorization 93
7.4 A Concrete Example for Algorithm 1 94
7.5 Algorithms 1 and 2 in Comparison 95
7.6 Column Replication Versus Appending Zero Columns 96
7.7 Performance of Algorithm 2 on the Boundary and in the Interior of $\mathcal{CP}_n$ 97
7.8 Other Difficult Instances 98
7.9 Randomly Generated Examples of Higher Order 99
7.10 Comparison with an Algorithm by Ding et al. 100
7.11 Comparison with a Method by Jarre and Schmallowsky 103
7.12 A Real Life Application in Statistics 106
7.13 Examples for Algorithms 4 and 5 107

8 Nonnegative Matrix Factorization **109**
8.1 Symmetric Nonnegative Matrix Factorization 109
8.1.1 Algorithms for Symmetric Nonnegative Matrix Factorization 110
8.1.2 Numerical Results for Symmetric Nonnegative Matrix Factorization . . . 113
8.2 General Nonnegative Matrix Factorization 118
8.2.1 Generalizing the Results to the Framework of Nonnegative Matrix Factorization 119
8.2.2 Exact Projection Algorithm for Nonnegative Matrix Factorization 126
8.2.3 Modified Algorithm for Nonnegative Matrix Factorization 127
8.2.4 Numerical Results for Nonnegative Matrix Factorization 130

9 Conclusion and Further Remarks **135**

Appendix: Singular Value Decomposition and Pseudoinverse Matrices **137**

List of Algorithms **141**

List of Figures **143**

List of Tables **145**

Nomenclature **147**

Bibliography **151**

Index **157**

1 Short Summary

A matrix $A \in \mathbb{R}^{n\times n}$ is called completely positive if there exists an entrywise nonnegative matrix B such that $A = BB^T$. These matrices can be used to obtain convex reformulations of for example nonconvex quadratic or combinatorial problems. According to [11], one of the main problems with completely positive matrices is checking whether a given matrix is completely positive. As shown in [38], this is NP-hard in general. So far, it is still an open question whether checking $A \in \mathcal{CP}_n$ is also in NP.

For a given matrix $A \in \mathcal{CP}_n$, it is nontrivial to find a cp-factorization $A = BB^T$ with $B \in \mathbb{R}_+^{n\times r}$ since this factorization would provide a certificate for the matrix to be completely positive. But this factorization is not only important for the membership to the completely positive cone, it can also be used to recover the solution of the underlying quadratic or combinatorial problem.

In addition, it is not a priori known how many columns r are necessary to generate a cp-factorization for the given matrix. The minimal possible number of columns is called the cp-rank of A and so far it is still an open question how to derive the cp-rank for a given matrix. Some facts on completely positive matrices and the cp-rank will be given throughout the following chapter and especially in Sections 2.2 and 2.4.

Moreover, in Chapter 6, we will see a factorization algorithm, which, for a given completely positive matrix A and a suitable starting point, computes the nonnegative factorization $A = BB^T$. The algorithm therefore returns a certificate for the matrix to be completely positive. As introduced in Chapter 3, the fundamental idea of the factorization algorithm is to start from an initial factorization $A = \tilde{B}\tilde{B}^T$, with $\tilde{B} \in \mathbb{R}^{n\times n}$ not necessarily entrywise nonnegative, and extend $\tilde{B}$ to a matrix $B \in \mathbb{R}^{n\times r}$, where r is greater than or equal to the cp-rank of A and $A = BB^T$. Then it is the goal to transform the initial factorization $A = BB^T$ into a cp-factorization.

This problem can be formulated as a nonconvex feasibility problem, as shown in Section 4.1, and solved by a method which is based on alternating projections, as proven in Chapter 6.

On the topic of alternating projections, a survey will be given in Chapter 5. Here we will see how to apply this technique to several types of sets like subspaces, convex sets, manifolds and semialgebraic sets. Furthermore, we will see some known facts on the convergence rate for alternating projections between these types of sets. Considering more than two sets yields the so called cyclic projections approach. Here some known facts for subspaces and convex sets will be shown. Moreover, we will see a new convergence result on cyclic projections among a sequence of manifolds in Section 5.4.

In the context of cp-factorizations, a local convergence result for the introduced algorithm will be given. This result is based on the convergence for alternating projections between semialgebraic sets in [42].

To obtain cp-facrorizations with this first method, it is necessary to solve a second order cone problem in every projection step, which is very costly. Therefore, in Section 6.2, we will see an additional heuristic extension, which improves the numerical performance of the algorithm. Extensive numerical tests in Chapter 7 will show that the factorization method is very fast in most instances. In addition, we will see how to derive a certificate for the matrix to be an element of the interior of the completely positive cone. The key aspects and results on deriving cp-factorizations can also be found in the submitted preprint article [50].

As a further application, this method can be extended to find a symmetric nonnegative matrix factorization, where we consider an additional low-rank constraint. Here again, the method to derive factorizations for completely positive matrices can be used, albeit with some further adjustments, introduced in Section 8.1. Moreover, we will see that even for the general case of deriving a nonnegative matrix factorization for a given matrix $A \in \mathbb{R}^{m \times n}$, the key aspects of the completely positive factorization approach can be used. To this end, it becomes necessary to extend the idea of finding a completely positive factorization such that it can be used for rectangular matrices. This yields an applicable algorithm for nonnegative matrix factorization in Section 8.2. Numerical results for this approach will suggest that the presented algorithms and techniques to obtain completely positive matrix factorizations can be extended to general nonnegative factorization problems.

The majority of the notation in this thesis will be standard notation. Nevertheless, for the reader's convenience, a Nomenclature is provided at the end of this thesis. Since the algorithmic approaches to obtain completely positive factorizations are related to the Moore-Penrose-inverse, and the nonnegative matrix factorization approach is based on an initial factorization calculated via singular value decomposition, some facts on these topics are collected in the Appendix. In addition, a Bibliography and an Index are also provided at the end of this thesis.

2 Introduction

To introduce the topic of completely positive matrices, the following chapter is organized as follows: First, we will see an introduction to the copositive and the completely positive matrix cone as subsets of the set of symmetric matrices, followed by some known facts on how to prove whether a given matrix is completely positive. Moreover, we will have a closer look at the interior of the completely positive cone and at its boundary. Especially for the latter, we will introduce the notation of the cp-rank and the cp$^+$-rank to obtain the sufficient number of columns for a completely positive factorization at the boundary or in the interior. In the end of this chapter, we will see some applications of the completely positive cone in the context of conic programming.

2.1 The Copositive and the Completely Positive Cone

In the context of completely positive optimization, we will consider only symmetric matrices of given order n. To this end, $\mathcal{S}_n$ will denote the set of $n \times n$ symmetric matrices. Having this, we can now introduce one of the fundamental properties of a matrix for this thesis.

Definition 2.1. *A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is called completely positive if there exists an entrywise nonnegative matrix $B \in \mathbb{R}^{n\times r}$ such that $A = BB^T$. The factorization $A = BB^T$ with B entrywise nonnegative will be called cp-factorization throughout this thesis.*

To mention the key properties of completely positive matrices, we further need the following definition.

Definition 2.2. *We call a convex cone K pointed if $K \cap (-K) = \{0\}$.*

Now the following properties hold, see for example the survey article of Dür [43] or the book of Berman and Shaked-Monderer [11]. Here $\mathbb{R}^n_+$ denotes the set of entrywise nonnegative vectors in $\mathbb{R}^n$.

Lemma 2.3. *The set of completely positive matrices,*

$$\mathcal{CP}_n := \left\{A \in \mathcal{S}_n \;\middle|\; A = BB^T, \text{ where } B \in \mathbb{R}^{n\times r}, B \geq 0\right\} = \operatorname{conv}\left\{xx^T \;\middle|\; x \in \mathbb{R}^n_+\right\},$$

is a closed, pointed, convex matrix cone with nonempty interior, whose extreme rays are the rank-1 matrices xx^T, where $x \in \mathbb{R}^n_+$.

In the following, we will have a closer look at the dual of the set $\mathcal{CP}_n$. For this, we denote by $\langle A, B\rangle := \operatorname{trace}(A^T B) = \sum_{i,j=1}^{n} a_{ij}b_{ij}$ the inner product of two matrices $A, B \in \mathbb{R}^{n\times n}$. Thus, we

can define the dual of a closed convex cone $K \subseteq \mathcal{S}_n$ as

$$K^* := \{X \in \mathcal{S}_n \mid \langle X, Y\rangle \geq 0 \text{ for all } Y \in K\}. \tag{1}$$

As shown in [9, Chapter 1, Section 2], we can additionally describe the interior of the dual cone as follows:

$$\operatorname{int}(K^*) = \{X \in \mathcal{S}_n \mid \langle X, Y\rangle > 0 \text{ for all } Y \in K \setminus \{0\}\}. \tag{2}$$

Considering the dual of the set of completely positive matrices gives rise to the set of so called copositive matrices. We call a symmetric matrix $A \in \mathbb{R}^{n\times n}$ copositive if the quadratic form $x^T Ax$ is nonnegative for all nonnegative vectors x. As shown in [11, Proposition 1.24], we have the following properties:

Lemma 2.4. *The set of copositive matrices,*

$$\mathcal{COP}_n := \left\{A \in \mathcal{S}_n \;\middle|\; x^T Ax \geq 0 \text{ for all } x \in \mathbb{R}^n_+\right\},$$

is a closed, pointed, convex matrix cone with nonempty interior.

To show that $\mathcal{CP}_n$ and $\mathcal{COP}_n$ are dual cones of each other, we need the following Lemma, for which the proof can be found in [11, Theorem 1.36].

Lemma 2.5. *S is a closed convex cone if and only if $S = S^{**}$.*

Now we can show the following duality, cf. [11, Theorem 2.3]:

Theorem 2.6. *$\mathcal{CP}_n$ and $\mathcal{COP}_n$ are dual cones in the space $\mathcal{S}_n$. This means $\mathcal{CP}_n^* = \mathcal{COP}_n$ and $\mathcal{COP}_n^* = \mathcal{CP}_n$.*

Proof. Consider a symmetric matrix $X \in \mathbb{R}^{n\times n}$. Then we have $X \in \mathcal{CP}_n^*$ if and only if $\langle X, A\rangle \geq 0$ for all $A \in \mathcal{CP}_n$. This is true if and only if $\operatorname{trace}(XA) \geq 0$ for all $A \in \mathcal{CP}_n$. Let $A = BB^T$ be a cp-factorization of A, with B of n rows and entrywise nonnegative, then $X \in \mathcal{CP}_n^*$ if and only if $\operatorname{trace}(XBB^T) = \operatorname{trace}(B^T XB) \geq 0$, if and only if $b^T Xb \geq 0$ for every $b \in \mathbb{R}^n_+$. This implies $X \in \mathcal{COP}_n$ such that $\mathcal{CP}_n^* = \mathcal{COP}_n$. On the other hand, $\mathcal{CP}_n$ is a closed convex cone according to Lemma 2.3, such that Lemma 2.5 implies $\mathcal{COP}_n^* = \mathcal{CP}_n^{**} = \mathcal{CP}_n$, completing the proof. □

To show that these matrix cones also have a close relation to other matrix cones, consider the following two definitions.

Definition 2.7. *The set $\mathcal{S}_n^+ := \{A \in \mathcal{S}_n \mid x^T Ax \geq 0 \text{ for all } x \in \mathbb{R}^n\} = \{A \in \mathcal{S}_n \mid A \succcurlyeq 0\}$ defines the cone of positive semidefinite matrices, where $A \succcurlyeq 0$ indicates that the matrix A is positive semidefinite.*

The set $\mathcal{N}_n := \{A \in \mathbb{R}^{n\times n} \mid A_{ij} \geq 0 \text{ for all } i, j = 1, \ldots, n\}$ defines the cone of entrywise nonnegative matrices. In short form, we will write $A \geq 0$ for $A_{ij} \geq 0$ for all $i, j = 1, \ldots, n$ throughout this thesis.

Then every $X \in \mathcal{S}_n^+$ is also copositive since $x^T Ax \geq 0$ for every $x \in \mathbb{R}^n$, implying

$$\mathcal{S}_n^+ \subseteq \mathcal{COP}_n.$$

On the other hand, considering $A \in \mathcal{N}_n$ gives $x^T Ax \geq 0$ for every nonnegative x such that $\mathcal{N}_n \subseteq \mathcal{COP}_n$. This shows for the Minkowski sum

$$\mathcal{S}_n^+ + \mathcal{N}_n \subseteq \mathcal{COP}_n.$$

For $n \leq 4$, we have $\mathcal{S}_n^+ + \mathcal{N}_n = \mathcal{COP}_n$, cf. [34], but to show that equality does not hold for $n \geq 5$, consider the so called Horn matrix, cf. [11, Example 1.30]:

$$H = \begin{pmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 & 1 \end{pmatrix}. \tag{3}$$

To show that H is copositive, write

$$\begin{aligned} x^T Hx &= (x_1 - x_2 + x_3 + x_4 - x_5)^2 + 4x_2x_4 + 4x_3(x_5 - x_4) \\ &= (x_1 - x_2 + x_3 - x_4 + x_5)^2 + 4x_2x_5 + 4x_1(x_4 - x_5). \end{aligned}$$

Here the first expression shows $x^T Hx \geq 0$ for nonnegative x and $x_5 \geq x_4$, whereas on the other hand, the second expression shows $x^T Hx \geq 0$ for nonnegative x and $x_5 < x_4$. Hence, $H \in \mathcal{COP}_5$. Moreover, H is neither positive semidefinite, nor entrywise nonnegative. It can be shown, cf. [53], that H is extreme for $\mathcal{COP}_5$ such that H can not be decomposed into $H = S + N$ with $S \in \mathcal{S}_5^+$ and $N \in \mathcal{N}_5$.

Symmetric matrices which are entrywise nonnegative and positive semidefinite at the same time are called doubly nonnegative matrices, see for example [11]. We denote the set of all such matrices by

$$\mathcal{DNN}_n := \mathcal{S}_n^+ \cap \mathcal{N}_n = \{A \in \mathcal{S}_n \mid A \geq 0 \text{ and } A \succcurlyeq 0\}.$$

Then, by duality, we get the property

$$\mathcal{CP}_n \subseteq \mathcal{DNN}_n. \tag{4}$$

If the order n is clear from the context and to simplify notation, we will drop the index n and simply write $\mathcal{CP}$, $\mathcal{COP}$ and $\mathcal{DNN}$ for the introduced sets.

To show that the necessary condition in (4) is not sufficient in general, consider the following doubly nonnegative matrix A_{DNN}, which is not completely positive, cf. [11, Example 2.9].

Example 2.8. *Consider the matrix*

$$A_{DNN} = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 & 6 \end{pmatrix},$$

then $A_{DNN} \in \mathcal{DNN}_5 \setminus \mathcal{CP}_5$.

To show that the matrix is not completely positive, we need certificates for a matrix to be completely positive. A selection of some certificates is given in the following section.

2.2 Complexity and Theoretical Certificates for Complete Positivity

It is easy to construct completely positive matrices, but one of the main problems in the theory of completely positive matrices is deciding if a given matrix is completely positive, as mentioned in the preface of [11].

In general, checking whether a given matrix is completely positive is NP-hard, as shown in [38, Theorem 5.3], but there exist several theoretical conditions for complete positivity of a matrix. For a comprehensible survey of these conditions, the reader is referred to [11, Chapter 2].

Nevertheless, we will see some of the conditions in the following part to keep this thesis self contained. In the following, we consider only symmetric square matrices $A, C \in \mathbb{R}^{n\times n}$.

A matrix remains completely positive under the following operations, cf. [11, Chapter 2.1].

Lemma 2.9. *(a) The sum $A + C$ of completely positive matrices A, C is completely positive.*

(b) The Kronecker product $A \otimes C$ of completely positive matrices A, C is completely positive.

(c) If A is completely positive and $D \in \mathbb{R}_+^{m\times n}$, then DAD^T is completely positive.

(d) If A is completely positive and $k \in \mathbb{N}$, then A^k is completely positive.

(e) If $P \in \mathbb{R}^{n\times n}$ is a permutation matrix, then A is completely positive if and only if P^TAP is completely positive.

(f) If $D \in \mathbb{R}^{n\times n}$ is diagonal matrix with positive diagonal entries, then A is completely positive if and only if DAD is completely positive.

Proof. (a) Let $A = B_1B_1^T$ with $B_1 \geq 0$ and $C = B_2B_2^T$ with $B_2 \geq 0$, then the matrix $B = [B_1|B_2]$ is entrywise nonnegative with $(A + C) = BB^T$, showing that the sum $A + C$ is completely positive.

(b) Let $A = B_1 B_1^T$ with $B_1 \geq 0$ and $C = B_2 B_2^T$ with $B_2 \geq 0$, then the matrix $B = B_1 \otimes B_2$ is entrywise nonnegative with $(A \otimes C) = BB^T$, showing that the Kronecker product is completely positive.

(c) Let $A = BB^T$ with $B \geq 0$. Then $DAD^T = (DB)(DB)^T$ is completely positive.

(d) If k is even, we have $k = 2l$ for $l \in \mathbb{N}$ such that $A^k = (A^l)^2 = (A^l)(A^l)^T$, where $A^l \in \mathbb{R}_+^{n \times n}$, proving that A^k is completely positive. If on the other hand k is odd, we have $k = 2l + 1$ for $l \in \mathbb{N}$ such that $A^k = A^{2l+1} = A^l A A^l$ and part (c) shows that A^k is completely positive.

(e) If A is completely positive, part (c) proves that $P^T AP$ is completely positive. For the reverse part on the other hand, let $P^T AP$ be completely positive and consider a cp-factorization $P^T AP = FF^T$ with $F \geq 0$. Then $PP^T = P^T P = I_n$ since P is orthogonal and we get

$$A = P(P^T AP)P^T = PFF^T P^T = (PF)(PF)^T.$$

Thus, A is completely positive since $PF \geq 0$ is entrywise nonnegative as a permutation of a nonnegative matrix.

(f) If A is completely positive, again part (c) shows that DAD^T is completely positive. If on the other hand DAD^T is completely positive, let $DAD^T = FF^T$ with $F \geq 0$. By definition we have $D = \mathrm{Diag}(D_{11}, \dots D_{nn})$ with $D_{ii} > 0$ for every i. Thus, $D \in \mathbb{R}_+^{n \times n}$ is nonsingular and we have $D^{-1} = \mathrm{Diag}(\frac{1}{D_{11}}, \dots, \frac{1}{D_{nn}})$ is entrywise nonnegative. Then

$$A = D^{-1}(DAD)D^{-1} = D^{-1}FF^T D^{-1} = (D^{-1}F)(D^{-1}F)^T,$$

proving that A is completely positive.

□

For these conditions, it is necessary to start with a completely positive matrix. The following conditions can be used to show that a given matrix is completely positive, cf. [11, Chapter 2.4].

Theorem 2.10. *Entrywise nonnegative, symmetric, diagonally dominant matrices are completely positive.*

Proof. Let $A \in \mathbb{R}^{n \times n}$ be entrywise nonnegative, symmetric and diagonally dominant and let

$$a_i := a_{ii} - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} \geq 0,$$

where a_{ij} denotes the specific entry of A. In addition, let $F_{ij} \in \mathbb{R}^{n \times n}$ have entry equal to 1 in positions $ii,\ ij,\ ji,\ jj$ and zero entries everywhere else. Thus, $F_{ij} = f_{ij} f_{ij}^T$, where $f_{ij} \in \mathbb{R}^n$ with entries $f_i = f_j = 1$ and $f_k = 0$ for every $k \neq i, j$, proving that F_{ij} is completely positive. Then we have

$$A = \sum_{1 \leq j < i \leq n} a_{ij} F_{ij} + \mathrm{Diag}(a_1, \dots, a_n).$$

Further, we can write

$$\operatorname{Diag}(a_1, \ldots, a_n) = \operatorname{Diag}(\sqrt{a_1}, \ldots, \sqrt{a_n}) \operatorname{Diag}(\sqrt{a_1}, \ldots, \sqrt{a_n})^T,$$

proving that $\operatorname{Diag}(a_1, \ldots, a_n)$ is completely positive. Since $\mathcal{CP}_n$ is a matrix cone, $a_{ij}F_{ij}$ is also completely positive and Lemma 2.9 (a) now proves $A \in \mathcal{CP}_n$. □

For the next certificate, we need the definition of a comparison matrix and an M-matrix. There exist several equivalent definitions for an M-matrix. For a survey on this topic, the reader is referred to [81]. We will use the following definition.

Definition 2.11. *(a) Let $A \in \mathbb{R}^{n\times n}$. The comparison matrix of A is denoted by $M(A)$ and is defined by*

$$M(A)_{ij} = \begin{cases} |a_{ij}|, & \text{if } i = j \\ -|a_{ij}|, & \text{if } i \neq j. \end{cases}$$

(b) A matrix $A \in \mathbb{R}^{n\times n}$ is called M-matrix if it can be written as $A = sI - B$, where $B \in \mathbb{R}^{n\times n}_+$ and $s \geq \rho(B)$. Here $\rho(B)$ denotes the spectral radius of B.

Now we can give our next theoretical condition for complete positivity, cf. [11, Theorem 2.6].

Theorem 2.12. *Let $A \in \mathbb{R}^{n\times n}$ be symmetric and entrywise nonnegative. Furthermore, let its comparison matrix $M(A)$ be positive semidefinite. Then A is completely positive.*

Proof. $M(A)$ is an M-matrix and therefore we can show, see for example [11, Theorem 1.16], that there exists a diagonal matrix D with positive diagonal entries such that $DM(A)D$ is diagonally dominant. The entries of $DM(A)D$ and DAD are equal in absolute value such that DAD is also diagonally dominant. Now Theorem 2.10 shows that DAD is completely positive. Finally, Lemma 2.9 (f) proves that A is completely positive. □

In addition, So and Xu proved the following simple sufficient condition for a doubly nonnegative matrix to be completely positive, see [88, Theorem 2.5].

Theorem 2.13. *Let $A \in \mathcal{DNN}_n$ with $\operatorname{rank}(A) = r$. Further let R_i denote the i-th row sum of A for every $i = 1, \ldots, n$ and let*

$$rR_i^2 \geq (r-1)A_{ii}(R_1 + \cdots + R_n) \qquad \text{for all } i = 1, \ldots, n,$$

where A_{ii} denotes the i-th diagonal entry of A. Then A is completely positive. Moreover, r is a sufficient number of columns for a cp-factorization.

We will have a closer look at the number of columns for a cp-factorization in Section 2.4. Theorem 2.13 provides a sufficient but not necessary condition for a doubly nonnegative matrix to be completely positive, which would help us to show that A_{DNN} in Example 2.8 is not completely positive. Nevertheless, we will use the result in Theorem 2.13 to show that a given matrix is completely positive in Section 7.3.

Furthermore, many certificates for complete positivity are based on graph theoretical aspects. To show that the matrix A_{DNN} in Example 2.8 is not completely positive, we will use one of them, cf. [11, Theorem 2.8]. Here a symmetric matrix A is called a matrix realization of a graph G if the off-diagonal entries A_{ij} are nonzero whenever the vertices i and j are connected by an edge in G.

Theorem 2.14. *Let G be a triangle-free graph and A be a nonnegative symmetric matrix realization of G. Then A is completely positive if and only if the comparison matrix $M(A)$ is positive semidefinite.*

Remark 2.15. *Coming back to Example 2.8, we see that the graph of A_{DNN} is triangle free, but its comparison matrix*

$$\begin{pmatrix} 1 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 6 \end{pmatrix}$$

is not positive semidefinite such that A_{DNN} is not completely positive.

So here we can find a contradiction, indicating that these results may be useful to discard the membership to the completely positive cone. For the conditions we considered so far, it seems unclear how they can be used algorithmically to obtain a certificate for an arbitrary matrix to be completely positive. Motivated by the definition of $\mathcal{CP}$, we also have the following characterizations:

Lemma 2.16. *$A \in \mathbb{R}^{n\times n}$ is completely positive if and only if one of the following conditions holds:*

(a) There exists an entrywise nonnegative matrix $B \in \mathbb{R}^{n\times r}$ such that $A = BB^T$.

(b) A has the following sum-representation

$$A = \sum_{i=1}^{r} b_i b_i^T, \text{ where } b_i \in \mathbb{R}^n_+ \text{ for every } i = 1, \ldots, r.$$

These two conditions are equivalent if b_i in (b) denotes the i-th column of B in (a). The representation in Lemma 2.16 (b) is called the *rank-1 representation* of A since it decomposes A into a sum of rank-1 matrices $b_i b_i^T$. So, finding a cp-factorization is a certificate for the matrix to be completely positive. We will use this fact in the following chapters to introduce an algorithmic method to show that a given matrix is completely positive. In addition, for small dimensions, we have the following certificate, which can also be used algorithmically.

Going back to (4), stating $\mathcal{CP}_n \subseteq \mathcal{DNN}_n$, it can be shown that additionally the reverse implication

$$\mathcal{DNN}_n \subseteq \mathcal{CP}_n \tag{5}$$

holds for $n \leq 4$, giving the following lemma, cf. [73].

Lemma 2.17. *Let $A \in \mathbb{R}^{n\times n}$ and $n \leq 4$. Then A is completely positive if and only if A is doubly nonnegative.*

So, for small dimensions, it is sufficient to verify that all entries are nonnegative and the matrix is positive semidefinite to get a certificate for complete positivity. But Example 2.8 shows that for $n \geq 5$, the if-part of Lemma 2.17 does not hold any more.

So far, we can not distinguish between matrices on the boundary of $\mathcal{CP}_n$ and in the interior of $\mathcal{CP}_n$. For this, we need a characterization of the interior of the completely positive cone. So we will have a closer look at the interior of the cone in the following section.

2.3 The Interior of the Completely Positive Cone

Especially in Section 2.7, it may become important to verify strict feasibility of a matrix and therefore to ensure the membership to the interior of the completely positive cone. The properties mentioned in the previous sections already provide a first characterization of the interior of the completely positive cone. More precisely, based on (4) and Lemma 2.17, we have

$$\begin{aligned} \mathcal{CP}_n &\subseteq \mathcal{DNN}_n \qquad \text{for every } n \in \mathbb{N} \text{ and} \\ \mathcal{CP}_n &= \mathcal{DNN}_n \qquad \text{for every } n \leq 4. \end{aligned}$$

Thus, we further get for $n \leq 4$:

$$\operatorname{int}(\mathcal{CP}_n) = \operatorname{int}(\mathcal{N}_n) \cap \operatorname{int}(\mathcal{S}_n^+) = \{A \in \mathcal{S}_n \mid A > 0 \text{ and } A \succ 0\}.$$

Here $\operatorname{int}(\mathcal{N}_n)$ describes the set of entrywise strictly positive matrices and $\operatorname{int}(\mathcal{S}_n^+)$ is the set of positive definite matrices.

For $n \geq 5$, it follows that $\operatorname{int}(\mathcal{CP}_n) \subseteq \operatorname{int}(\mathcal{N}_n) \cap \operatorname{int}(\mathcal{S}_n^+)$ but equality does not hold in general. For a concrete example of an entrywise nonnegative and positive definite matrix, which is an element of the boundary of the completely positive cone, see Example 2.43 in Section 2.6.

To give an explicit characterization of the interior of the completely positive cone for arbitrary order n, consider the following Theorem, cf. [44, Theorem 2.3].

Theorem 2.18. *For $B_1 \in \mathbb{R}^{n\times n}$ and $B_2 \in \mathbb{R}^{n\times r}$, let the notation $[B_1 \mid B_2]$ describe the matrix in $\mathbb{R}^{n\times(n+r)}$ whose columns are the columns of the matrix B_1 augmented with the columns of B_2. Then we have*

$$\operatorname{int}(\mathcal{CP}_n) = \{BB^T \mid B = [B_1 \mid B_2], \text{ where } B_1 > 0 \text{ is nonsingular, and } B_2 \geq 0\}. \tag{6}$$

Let $\mathbb{R}^n_{++}$ denote the set of entrywise strictly positive vectors in $\mathbb{R}^n$, then (6) can be rewritten in terms of the rank-1 representation of BB^T:

$$\operatorname{int}(\mathcal{CP}_n) = \left\{ \sum_{i=1}^{r} b_i b_i^T \;\middle|\; \begin{array}{c} r \geq n,\ b_i \in \mathbb{R}^n_+ \text{ for every } i, \\ b_i \in \mathbb{R}^n_{++} \text{ for every } i \leq n, \\ \operatorname{span}(b_1, \ldots, b_n) = \mathbb{R}^n \end{array} \right\}.$$

An improved characterization of the interior of the completely positive cone can be found in [36, Theorem 7.4] and reads as follows:

Theorem 2.19. *Consider the cone $\mathcal{CP}_n$ for $n \in \mathbb{N}$. Then we have*

$$\begin{aligned} \operatorname{int}(\mathcal{CP}_n) &= \left\{ \sum_{i=1}^{r} b_i b_i^T \;\middle|\; \begin{array}{c} b_i \in \mathbb{R}^n_{++} \text{ for every } i, \\ \operatorname{span}(b_1, \ldots, b_r) = \mathbb{R}^n \end{array} \right\} \\ &= \{BB^T \mid B > 0 \text{ and } \operatorname{rank}(B) = n\} \end{aligned} \tag{7}$$

and

$$\begin{aligned} \operatorname{int}(\mathcal{CP}_n) &= \left\{ \sum_{i=1}^{r} b_i b_i^T \;\middle|\; \begin{array}{c} b_1 \in \mathbb{R}^n_{++}, b_i \in \mathbb{R}^n_{+} \text{ for every } i, \\ \operatorname{span}(b_1, \ldots, b_r) = \mathbb{R}^n \end{array} \right\} \\ &= \{BB^T \mid \operatorname{rank}(B) = n,\ B = [a \mid \widehat{B}],\ \text{where } a \in \mathbb{R}^n_{++},\ \widehat{B} \geq 0\}. \end{aligned} \tag{8}$$

Here we can show, cf. [36, Lemma 7.13], that these two characterizations are equivalent to each other according to the following rotation argument. Equivalent in this case means that a rank-1 representation in equation (8) can be transformed to a rank-1 representation in equation (7) of Theorem 2.19.

Lemma 2.20. *Let $a \in \mathbb{R}^n_{++}$ and $b \in \mathbb{R}^n_{+}$. Then there exist vectors $c, d \in \mathbb{R}^n_{++}$ such that*

$$aa^T + bb^T = cc^T + dd^T.$$

Proof. Consider $\theta \in \mathbb{R}$ and define

$$\begin{aligned} a_\theta &:= a\cos(\theta) - b\sin(\theta), \\ b_\theta &:= b\cos(\theta) + a\sin(\theta). \end{aligned}$$

Then we observe that

$$aa^T + bb^T = a_\theta a_\theta^T + b_\theta b_\theta^T.$$

Since $a \in \mathbb{R}^n_{++}$ and $b \in \mathbb{R}^n_{+}$, we can pick $\theta > 0$ sufficiently small such that $a_\theta, b_\theta \in \mathbb{R}^n_{++}$. Now let $c = a_\theta$ and $d = b_\theta$, completing the proof. □

Remark 2.21. *This rotation argument can be seen as a special application of an orthogonal matrix to a given decomposition. We will use the idea of transforming decompositions in Chapter 3 to generate a certificate for a matrix to be completely positive.*

To characterize the membership of a matrix to the completely positive cone or to the interior of the cone, it is sufficient to find a certain rank-1 representation or a decomposition matrix, as shown in Lemma 2.16 and Theorem 2.19. Here the question is how many columns or how many rank-1 matrices do we need to prove the membership to $\mathcal{CP}_n$ or its interior. So the question arises whether we can say anything about the parameter r. We will consider this question in the following section.

2.4 The cp-rank and the cp$^+$-rank for Completely Positive Matrices

The factorization of a completely positive matrix $A \neq 0$ with $\operatorname{rank}(A) \geq 2$ is never unique. We illustrate this with an example by Dickinson, cf. [35].

Example 2.22. *Consider the matrix*

$$A := \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix}.$$

Then $A = B_i B_i^T$ for each of the following matrices:

$$B_1 := \begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}, \qquad B_2 := \begin{pmatrix} 3 & 3 & 0 & 0 \\ 3 & 0 & 3 & 0 \\ 3 & 0 & 0 & 3 \end{pmatrix}, \qquad B_3 := \begin{pmatrix} 3 & 3 & 0 \\ 3 & 0 & 3 \\ 0 & 3 & 3 \end{pmatrix}.$$

Observe that the factorizations $A = B_i B_i^T$ for $i = 1, 2$ prove that $A \in \operatorname{int}(\mathcal{CP}_n)$, whereas the factorization $A = B_3 B_3^T$ does not. Moreover, note that the number of columns of the factors B_i varies. This gives rise to the following definitions.

Definition 2.23. *Let $A \in \mathbb{R}^{n \times n}$. The cp-rank of A is defined as*

$$\operatorname{cpr}(A) := \inf\{r \in \mathbb{N} \mid \exists\, B \in \mathbb{R}^{n \times r},\ B \geq 0,\ A = BB^T\}.$$

The cp$^+$-rank of A is

$$\operatorname{cpr}^+(A) := \inf\{r \in \mathbb{N} \mid \exists\, B \in \mathbb{R}^{n \times r},\ B > 0,\ A = BB^T\}.$$

It is an open problem to compute the cp-rank or the cp$^+$-rank of a matrix, cf. [10]. Nevertheless, there are partial results on upper and lower bounds for the cp-rank of a completely positive matrix. The following results can be found in [11, Section 3.1].

Lemma 2.24. *Let $A, B \in \mathbb{R}^{n \times n}$ be completely positive. Then:*

(a) $\operatorname{cpr}(A + B) \leq \operatorname{cpr}(A) + \operatorname{cpr}(B)$.

(b) $\operatorname{cpr}(A) \geq \operatorname{rank}(A)$.

The relation to the rank in Lemma 2.24 (b) results in the question whether the cp-rank can be equal to the rank. This question is well studied in [11, Section 3.4]. We will see some of the results for small dimensions here.

Theorem 2.25. *If $A \in \mathbb{R}^{n \times n}$ is completely positive and $\operatorname{rank}(A) \leq 2$, then*

$$\operatorname{cpr}(A) = \operatorname{rank}(A).$$

Theorem 2.26. *If $A \in \mathbb{R}^{n\times n}$ is completely positive and $n \leq 3$, then*

$$\operatorname{cpr}(A) = \operatorname{rank}(A).$$

To show that the last theorem does not hold for $n = 4$, consider the following counterexample, cf. [11, Example 3.1].

Example 2.27. *The matrix*

$$A_{cp4} = \begin{pmatrix} 6 & 3 & 3 & 0 \\ 3 & 5 & 1 & 3 \\ 3 & 1 & 5 & 3 \\ 0 & 3 & 3 & 6 \end{pmatrix}$$

is of rank 3 *and has cp-rank equal to* 4*.*

To show that the matrix A_{cp4} is of cp-rank 4, we have a closer look at the support of a rank-1 representation of the matrix. Here $\operatorname{supp}(x) := \{i \in \{1,\dots,n\} \mid x_i \neq 0\}$ will denote the support of a vector $x \in \mathbb{R}^n$. If $A_{cp4} = \sum_{i=1}^{r} b_i b_i^T$ is a minimal rank-1 representation of A_{cp4}, then $1 \in \operatorname{supp}\{b_i\}$ for at least one $i \in \{1,\dots,4\}$. Without loss of generality, let $1 \in \operatorname{supp}(b_1)$. Assuming $1 \notin \operatorname{supp}(b_i)$ for every $i \neq 1$, we have

$$b_1 b_1^T = \begin{pmatrix} 6 & 3 & 3 & 0 \\ 3 & \frac{3}{2} & \frac{3}{2} & 0 \\ 3 & \frac{3}{2} & \frac{3}{2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

such that the entry $(A - b_1 b_1^T)_{23} < 0$. But since $(A - b_1 b_1^T) = \sum_{i=2}^{r} b_i b_i^T$ is completely positive, this leads to a contradiction. Hence, 1 must belong to at least two of the supports $\operatorname{supp}(b_i)$, $i = 1,\dots,r$. By symmetry, 4 also belongs to at least two of the supports $\operatorname{supp}(b_i)$, $i = 1,\dots,r$. Adding the fact that 1 and 4 can not belong to the same support (otherwise we would have $(A_{cp4})_{14} \neq 0$) eventually shows $r \geq 4$. This implies

$$\operatorname{cpr}(A_{cp4}) \geq 4 > 3 = \operatorname{rank}(A).$$

To show that the cp-rank is equal to 4, we will use the following result, cf. [11, Theorem 3.3].

Theorem 2.28. *Let $A \in \mathbb{R}^{n\times n}$ be completely positive and $n \leq 4$. Then*

$$\operatorname{cpr}(A) \leq n.$$

Remark 2.29. *Theorem 2.28 now shows* $\operatorname{cpr}(A_{cp4}) = 4$*.*

The following theorem can give an upper bound for the cp-rank in higher dimensions, depending on the rank of the matrix, cf. [11, Theorem 3.5].

Theorem 2.30. *Let $A \in \mathbb{R}^{n\times n}$ completely positive and $\operatorname{rank}(A) = k$, with $k \geq 2$. Then we have*

$$\operatorname{cpr}(A) \leq \frac{k(k+1)}{2} - 1.$$

If we consider the order of the given matrix instead of the rank, this naturally extends to the following upper bound for the cp-rank.

Theorem 2.31. *Let $A \in \mathbb{R}^{n\times n}$ completely positive. Then we have*

$$\operatorname{cpr}(A) \leq \frac{n(n+1)}{2} - 1. \tag{9}$$

Considering this result and Theorem 2.28, we notice that especially for lower dimensions, there exist much tighter upper bounds for the cp-rank than (9). Bomze, Dickinson and Still showed in [15, Theorem 5.1] the following tighter upper bound for the cp-rank and gave an upper bound for the cp$^+$-rank of a completely positive matrix in the interior of the cone. Here the analysis of the bounds on the cp-rank or the cp$^+$-rank are based on the results in [18] and [86].

Theorem 2.32. *We have:*

- *There exist matrices $A \in \operatorname{int}(\mathcal{CP}_n)$ for which $\operatorname{cpr}(A) \neq \operatorname{cpr}^+(A)$.*
- *For all $A \in \mathcal{CP}_n$, we have:*

$$\operatorname{cpr}(A) \leq \operatorname{cp}_n := \begin{cases} n & \text{for} \quad n \in \{2,3,4\} \\ \frac{1}{2}n(n+1) - 4 & \text{for} \quad n \geq 5. \end{cases}$$

- *For all $A \in \operatorname{int}(\mathcal{CP}_n)$, we have:*

$$\operatorname{cpr}^+(A) \leq \operatorname{cp}_n^+ := \begin{cases} n+1 & \text{for} \quad n \in \{2,3,4\} \\ \frac{1}{2}n(n+1) - 3 & \text{for} \quad n \geq 5. \end{cases}$$

For a matrix $A \in \mathcal{CP}_n \setminus \operatorname{int}(\mathcal{CP}_n)$, we have $\operatorname{cpr}^+(A) = \infty$. This motivates studying matrices on the boundary of $\mathcal{CP}_n$. Here the reader is referred to Section 2.6. On the other hand, we have the following characterization of the interior of the completely positive cone, cf. [15, Theorem 1.2].

Lemma 2.33. *For $A \in \mathcal{S}_n$, we have*

$$A \in \operatorname{int}(\mathcal{CP}_n) \iff \operatorname{cpr}^+(A) < \infty \text{ and } \operatorname{rank}(A) = n.$$

In addition, if we define the numbers

$$\begin{aligned} p_n &:= \max\{\operatorname{cpr}(A) \mid A \in \mathcal{CP}_n\} \quad \text{and} \\ p_n^+ &:= \max\{\operatorname{cpr}^+(A) \mid \operatorname{cpr}^+(A) < \infty\}, \end{aligned} \tag{10}$$

we get the following result, cf. [15, Theorem 5.1].

Lemma 2.34. *Let p_n and p_n^+ be as defined above. Then*

$$p_n \leq p_n^+ \leq p_n + 1.$$

Based on the definition of p_n, we further have the following result:

Remark 2.35. *It can be shown, see [18, Corollary 2.1], that asymptotically p_n is close to the upper bound cp_n in Theorem 2.32.*

Furthermore, Lemma 2.34 gives rise to the question when the cp-rank is equal to the cp$^+$-rank.

Remark 2.36. *Example 2.22 fulfills this property since $\mathrm{cpr}^+(A) = 3$ according to the factorization matrix B_1. On the other hand, we have $\mathrm{rank}(A) = 3$ and therefore, using Theorem 2.26, we also have $\mathrm{cpr}(A) = 3$.*

To show that this property holds generically within the completely positive cone, consider the following theorem, cf. [15, Corollary 6.8].

Theorem 2.37. *Consider $A \in \mathcal{CP}_n$. The following properties are generic within the completely positive cone:*

(a) Having infinitely many completely positive factorizations $A = BB^T$, where $B \in \mathbb{R}^{n \times \mathrm{cpr}(A)}$.

(b) The cp- and cp$^+$-ranks being equal.

So this means that the cp-rank and the cp$^+$-rank are equal almost everywhere, or in other words, the set

$$\{A \in \mathcal{CP}_n \mid \mathrm{cpr}(A) \neq \mathrm{cpr}^+(A)\}$$

is a set of measure zero.

In contrast to Theorem 2.28, we will consider matrices of high cp-rank in the following section, showing that the result in Theorem 2.28 does not hold for $n > 4$.

2.5 Matrices of High cp-rank

For concrete examples of matrices $A \in \mathcal{CP}_n$ with $n \geq 5$ and $\mathrm{cpr}(A) > n$, the reader is referred to the results of Bomze, Schachiner and Ulrich in [17]. Here the authors developed a method to generate examples for matrices of high cp-rank. We mention some of them in the context of this thesis. First, we consider $n = 7$ and the following example.

Example 2.38. *Consider the matrix*

$$A_{cp14} = \begin{pmatrix} 163 & 108 & 27 & 4 & 4 & 27 & 108 \\ 108 & 163 & 108 & 27 & 4 & 4 & 27 \\ 27 & 108 & 163 & 108 & 27 & 4 & 4 \\ 4 & 27 & 108 & 163 & 108 & 27 & 4 \\ 4 & 4 & 27 & 108 & 163 & 108 & 27 \\ 27 & 4 & 4 & 27 & 108 & 163 & 108 \\ 108 & 27 & 4 & 4 & 27 & 108 & 163 \end{pmatrix} \in \mathbb{R}^{7 \times 7}.$$

It is shown in [17] that $\mathrm{cpr}(A_{cp14}) = 14$ and the matrix is of full rank.

This is therefore an example, where the cp-rank is greater than the rank of the matrix and especially greater than the order of the matrix since the matrix is of full rank. Another example, now of order 8, is the following.

Example 2.39. *Consider the matrix*

$$A_{cp18} = \begin{pmatrix} 541 & 880 & 363 & 24 & 55 & 11 & 24 & 0 \\ 880 & 2007 & 1496 & 363 & 48 & 22 & 22 & 24 \\ 363 & 1496 & 2223 & 1452 & 363 & 24 & 22 & 11 \\ 24 & 363 & 1452 & 2325 & 1584 & 363 & 48 & 55 \\ 55 & 48 & 363 & 1584 & 2325 & 1452 & 363 & 24 \\ 11 & 22 & 24 & 363 & 1452 & 2223 & 1496 & 363 \\ 24 & 22 & 22 & 48 & 363 & 1496 & 2007 & 880 \\ 0 & 24 & 11 & 55 & 24 & 363 & 880 & 541 \end{pmatrix} \in \mathbb{R}^{8\times 8}.$$

It is shown in [17] that $\operatorname{cpr}(A_{cp18}) = 18$ *and the matrix is of full rank.*

In addition, Bomze, Schachinger and Ulrich provided the following concrete examples for $n = 9$ and $n = 11$, cf. [17].

Example 2.40. *Consider the matrix*

$$A_{cp26} = \begin{pmatrix} 2548 & 1628 & 363 & 60 & 55 & 55 & 60 & 363 & 1628 \\ 1628 & 2548 & 1628 & 363 & 60 & 55 & 55 & 60 & 363 \\ 363 & 1628 & 2483 & 1562 & 363 & 42 & 22 & 55 & 60 \\ 60 & 363 & 1562 & 2476 & 1628 & 363 & 42 & 55 & 55 \\ 55 & 60 & 363 & 1628 & 2548 & 1628 & 363 & 60 & 55 \\ 55 & 55 & 42 & 363 & 1628 & 2476 & 1562 & 363 & 60 \\ 60 & 55 & 22 & 42 & 363 & 1562 & 2483 & 1628 & 363 \\ 363 & 60 & 55 & 55 & 60 & 363 & 1628 & 2548 & 1628 \\ 1628 & 363 & 60 & 55 & 55 & 60 & 363 & 1628 & 2548 \end{pmatrix} \in \mathbb{R}^{9\times 9}.$$

It is shown in [17] that $\operatorname{cpr}(A_{cp26}) = 26$ *and the matrix is of full rank.*

Example 2.41. *Consider the matrix*

$$A_{cp32} = \frac{1}{441}\begin{pmatrix} 781 & 0 & 72 & 36 & 228 & 320 & 240 & 228 & 36 & 96 & 0 \\ 0 & 845 & 0 & 96 & 36 & 228 & 320 & 320 & 228 & 36 & 96 \\ 72 & 0 & 827 & 0 & 72 & 36 & 198 & 320 & 320 & 198 & 36 \\ 36 & 96 & 0 & 845 & 0 & 96 & 36 & 228 & 320 & 320 & 228 \\ 228 & 36 & 72 & 0 & 781 & 0 & 96 & 36 & 228 & 240 & 320 \\ 320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 & 36 & 228 & 320 \\ 240 & 320 & 198 & 36 & 96 & 0 & 745 & 0 & 96 & 36 & 228 \\ 228 & 320 & 320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 & 36 \\ 36 & 228 & 320 & 320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 \\ 96 & 36 & 198 & 320 & 240 & 228 & 36 & 96 & 0 & 745 & 0 \\ 0 & 96 & 36 & 228 & 320 & 320 & 228 & 36 & 96 & 0 & 845 \end{pmatrix} \in \mathbb{R}^{11\times 11}.$$

It is shown in [17] that $\operatorname{cpr}(A_{cp32}) = 32$ *and the matrix is of full rank.*

All these examples are artificially generated and so far there are no known cp-rank factorizations for these matrices. We will see later in the numerical results to the factorization algorithms that generating cp-factorizations for these matrices seems to be difficult. These matrices did not appear in applications but nevertheless they provide a counterexample for the Drew-Johnson-Loewy (DJL) conjecture, which can be found in [40] and reads as follows.

Remark 2.42. *Considering p_n as defined in* (10)*, the DJL conjecture suspects that*

$$p_n \leq \left\lfloor \frac{n^2}{4} \right\rfloor,$$

where $\lfloor x \rfloor$ describes the floor function evaluated in x. Thus, $\lfloor x \rfloor = \max\{m \in \mathbb{Z} \mid m \leq x\}$.

So for $n = 7$, Example 2.38 shows $\operatorname{cpr}(A_{cp14}) = 14 > 12 = \left\lfloor \frac{7^2}{4} \right\rfloor$ and for $n = 8$, Example 2.39 shows $\operatorname{cpr}(A_{cp18}) = 18 > 16 = \left\lfloor \frac{8^2}{4} \right\rfloor$. Moreover, the DJL conjecture is false for $n = 9$ since Example 2.40 proves $\operatorname{cpr}(A_{cp26}) = 26 > 20 = \left\lfloor \frac{9^2}{4} \right\rfloor$ and for $n = 11$ since Example 2.41 proves $\operatorname{cpr}(A_{cp32}) = 32 > 30 = \left\lfloor \frac{11^2}{4} \right\rfloor$.

For general counterexamples for the DJL-conjecture, the reader is referred to [18].

As mentioned in Section 2.4, we will have a closer look at the boundary of the completely positive cone. Thus, we consider the set of matrices with infinite cp^+-rank in the following section.

2.6 The Boundary of the Completely Positive Cone

The boundary of the completely positive cone is defined as

$$\begin{aligned} &\operatorname{bd}(\mathcal{CP}_n) = \mathcal{CP}_n \setminus \operatorname{int}(\mathcal{CP}_n) \quad \text{ or} \\ &\operatorname{bd}(\mathcal{CP}_n) = \{A \in \mathcal{CP}_n \mid \operatorname{cpr}^+(A) = \infty\}. \end{aligned}$$

If we consider the definition of the dual of a cone in (1) and of the interior of the dual cone in (2), we also have the following characterization for the boundary, based on $\mathcal{COP}_n$, as the dual cone of $\mathcal{CP}_n$.

$$\operatorname{bd}(\mathcal{CP}_n) = \{X \in \mathcal{S}_n \mid \langle X, Y \rangle = 0 \text{ for at least one } Y \in \mathcal{COP}_n \setminus \{0\}\}. \tag{11}$$

We will use this certificate to show that the following matrix is an element of the boundary of the completely positive cone, cf. [44, Example 2.2].

Example 2.43. *Consider the matrix*

$$A_{DS} = \begin{pmatrix} 8 & 5 & 1 & 1 & 5 \\ 5 & 8 & 5 & 1 & 1 \\ 1 & 5 & 8 & 5 & 1 \\ 1 & 1 & 5 & 8 & 5 \\ 5 & 1 & 1 & 5 & 8 \end{pmatrix}.$$

Then $A_{DS} \in \mathcal{CP}_5$ since $A = BB^T$ with

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 2 \\ 1 & 1 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 2 & 1 \end{pmatrix} \in \mathbb{R}_+^{5\times 10}$$

and $A_{DS} \notin \operatorname{int}(\mathcal{CP}_5)$ since there exists a copositive matrix H such that $\langle A, H\rangle = 0$. This matrix is the Horn matrix H in equation (3)*. Equation* (11) *then shows that the matrix A_{DS} is an element of the boundary of $\mathcal{CP}_5$.*

Remark 2.44. *Since the matrix A_{DS} is also positive definite and entrywise nonnegative, this gives a concrete example proving $\operatorname{int}(\mathcal{N}_n) \cap \operatorname{int}(\mathcal{S}_n^+) \not\subseteq \operatorname{int}(\mathcal{CP}_n)$ for $n \geq 5$, as mentioned in Section 2.3.*

By definition, we have $\mathcal{CP}_n = \operatorname{conv}\{xx^T \mid x \in \mathbb{R}_+^n\}$ and therefore the extreme rays of $\mathcal{CP}_n$ are the completely positive matrices of rank 1:

$$\operatorname{ext}(\mathcal{CP}_n) = \{xx^T \mid x \in \mathbb{R}_+^n\}.$$

In Chapter 7, we will see that the introduced methods to find factorizations for completely positive matrices also work well for some matrices at the boundary of $\mathcal{CP}_n$, but nevertheless for some instances at the boundary the method may fail numerically.

In the following section, we will see the key applications for completely positive and copositive matrices and how the factorizations of completely positive matrices become important for these applications.

2.7 Conic Programming and Applications

As mentioned in [43], the concept of copositivity was first introduced in 1952 by Motzkin in [78], followed by numerous publications dealing with copositivity and complete positivity. In optimization, the cones of completely positive and copositive matrices appeared in the 1990s and since then many papers work on completely positive or copositive reformulations of several types of nonconvex problems. Here the main idea is to overcome the lack of convexity since the reformulation allows us to avoid global optimization techniques, because for these reformulations, any optimum is global. Especially for reformulations of combinatorial problems, it becomes important to generate a completely positive factorization of the solution, in order to recover the solution of the underlying problem.

A general conic program is a linear optimization problem in matrix variables over a certain matrix cone $\mathcal{C}$. It can be written in the following form:

$$\begin{array}{ll} \min & \langle C, X\rangle \\ \text{s.t.} & \langle A_i, X\rangle = b_i \qquad (i = 1, \ldots, m) \\ & X \in \mathcal{C}. \end{array}$$

For the matrix cone $\mathcal{C}$, we can think of $\mathbb{R}^{n\times n}_+$, corresponding to linear programming, of $\mathcal{S}^+_n$, corresponding to semidefinite programming or of $\mathcal{COP}_n$ respectively $\mathcal{CP}_n$, corresponding to copositve or completely positive programming, repectively. In addition, another cone plays an important role for the methods presented in this thesis. The following definition can be found for example in [2].

Definition 2.45. *The second order cone of order n is defined as the set*

$$SOC_n = \left\{(x,t) \in \mathbb{R}^{n-1} \times \mathbb{R} \;\middle|\; \|x\| \leq t\right\}.$$

Here $\|\cdot\|$ denotes the Euclidean norm. This set is also known as the ice cream cone or as the Lorentz-cone.

A general second order cone problem, or short SOCP, can then be written as follows:

$$\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax \leq b \\ & x \in SOC. \end{array} \tag{SOCP}$$

In the following, we will focus on completely positive and copositive problems and their applications.

Completely positive matrices have received a lot of attention in the area of quadratic and binary optimization, as it has been shown that many combinatorial and nonconvex quadratic problems can be formulated as linear problems over $\mathcal{CP}_n$. For surveys on this area, the reader is referred to [13, 21, 43]. As a first example, consider the problem of computing the stability number α of a graph G on n nodes. A stable set is a set of vertices in a graph, where no two vertices are adjacent and the stability number is the cardinality of the largest possible stable set of the graph.

De Klerk and Pasechnik showed that α is the solution of a maximization problem over $\mathcal{CP}_n$, cf. [29]:

$$\alpha = \max\{\langle E, X\rangle \mid \langle A + I,\ X\rangle = 1, X \in \mathcal{CP}_n\}, \tag{12}$$

where A is the adjacency matrix of G and E is the all-ones matrix.

An optimal solution X^* of (12) also contains information about the maximal stable set: if X^* is of rank 1, i.e., $X^* = x^*(x^*)^T$, then $\operatorname{supp}(x^*)$ is the unique stable set in G. If $\operatorname{rank} X^* > 1$, then X^* can be factorized as $X^* = \sum_{i=1}^r x_i x_i^T$, and $\operatorname{supp}(x_i)$ is a maximal stable set for each i.

The completely positive reformulation of finding the maximum stable set in (12) motivated a reformulation of the following formulation by Motzin and Straus [76] for the clique number $\omega(G)$

of a graph G. The clique number of a graph is the cardinality of the largest possible subset of vertices, where every pair of vertices is adjacent. Such a subset is called a clique of G. Considering the complement graph $\bar{G}$ of G, cliques in G correspond to stable sets in $\bar{G}$. The clique problem was one of the 21 problems, which were shown to be NP-complete by Richard M. Karp in [63]. Motzkin and Straus [76] showed that

$$\frac{1}{\omega(G)} = \min\left\{x^T(E-A)x \;\middle|\; e^Tx = 1,\ x \geq 0\right\}. \tag{13}$$

Here A denotes the adjacency matrix of the graph, e is the all-ones-vector and E is again the all-ones-matrix. We now write the objective function as $x^T(E-A)x = \langle (E-A), xx^T\rangle$ and analogously the constraint $e^Tx = 1$ can be written as $\langle ee^T, xx^T\rangle = 1$. Then, since $E = ee^T$ and with $X := xx^T$ and equation (13), the clique number of a graph is the solution of the following convex problem:

$$\begin{aligned} \min\quad & \langle (E-A), X\rangle \\ \text{s.t.}\quad & \langle E, X\rangle = 1 \\ & X \in \mathcal{CP}_n. \end{aligned} \tag{14}$$

Here the complexity of the problem moved to the conic constraint and allows a convex reformulation such that any optimum will be global. To solve the reformulated problem, it is again necessary to verify the membership of a matrix to the completely positive cone. But we need not only a certificate for the matrix to be completely positive, we need a completely positive factorization of the matrix to recover the solution of the underlying problem. Going back to the stability number problem in equation (12), we can then provide the maximum stable set itself and not only its cardinality. This gives a first motivation to generate cp-factorizations in practical applications.

Furthermore, we will see that equation (13) is a special case of the standard quadratic problem, which motivates another important field for completely positive reformulations. As the following problems show, completely positive optimization is also closely connected to quadratic optimization. First, we will consider the general standard quadratic problem, shortly written as STQP:

$$\begin{aligned} \min\quad & x^TAx \\ \text{s.t.}\quad & e^Tx = 1 \\ & x \geq 0. \end{aligned} \tag{STQP}$$

A solution of this quadratic problem minimizes the not necessarily convex objective function and is an element of the standard simplex. Applying the same manipulations as used to obtain (14) now gives the following completely positive reformulation of the standard quadratic problem:

$$\begin{aligned} \min\quad & \langle A, X\rangle \\ \text{s.t.}\quad & \langle E, X\rangle = 1 \\ & X \in \mathcal{CP}_n. \end{aligned} \tag{15}$$

Here again, the objective function can be rewritten as $x^TAx = \langle A, xx^T\rangle$ and analogously the constraint $e^Tx = 1$ can be written as $\langle ee^T, xx^T\rangle = 1$. If we define $X := xx^T$ and E again

denotes the all-ones matrix, we get the above reformulation in (15). Based on (14), we get that solving the standard quadratic problem is also NP-hard. Moreover, the reformulation in (15) is again convex and the objective function is linear such that the optimum must be attained in an extremal point. Therefore, the optimum will be attained in a rank-one matrix xx^T with $x \geq 0$ and $e^T x = 1$ according to Lemma 2.3. This proves that (15) is an exact reformulation of the standard quadratic problem.

This approach does not only work for a single standard quadratic problem, but also for multiple STQP, where one considers the cartesian product of multiple simplices, cf. [16].

A further extension of the reformulations for quadratic problems was given by Burer in 2009. As shown in [21], it is possible to give a completely positive reformulation for any quadratic problem with linear and binary constraints. More precisely, it is possible to derive a completely positive reformulation of the following quadratic problem:

$$\begin{aligned} \min \quad & x^T Q x + 2c^T x \\ \text{s.t.} \quad & a_i^T x = b_i \quad (i = 1, \ldots, m) \\ & x \geq 0 \\ & x_j \in \{0, 1\} \quad (j \in B). \end{aligned} \tag{16}$$

Here B denotes some index subset for the binary variables. This problem can be equivalently formulated as the following completely positive problem:

$$\begin{aligned} \min \quad & \langle Q, X \rangle + 2c^T x \\ \text{s.t.} \quad & a_i^T x = b_i \quad (i = 1, \ldots, m) \\ & \langle a_i a_i^T, X \rangle = b_i^2 \quad (i = 1, \ldots, m) \\ & x_j = X_{jj} \quad (j \in B) \\ & \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \in \mathcal{CP}_n. \end{aligned} \tag{17}$$

Beside the so far considered quadratic problems, it is also possible to reformulate the following fractional quadratic problem, as shown in [82]. Here consider a copositive matrix A and additionally assume that $x^T A x = 0$ yields $x = 0$. Now the fractional quadratic problem reads as follows:

$$\begin{aligned} \max \quad & \tfrac{x^T Q x}{x^T A x} \\ \text{s.t.} \quad & e^T x = 1 \\ & x \geq 0, \end{aligned}$$

where no further assumptions on Q are necessary. This problem is then equivalent to the following problem, as shown in [82]:

$$\begin{aligned} \min \quad & \langle Q, X \rangle \\ \text{s.t.} \quad & \langle A, X \rangle = 1 \\ & X \in \mathcal{CP}_n. \end{aligned}$$

For further remarks on this result, see also [20].

Moreover, the need to factorize completely positive matrices arises in statistics in the area of multivariate extremes, cf. [27], where it is shown how tail dependence of a multivariate regularly-varying random vector can be summarized in a so called tail pairwise dependence matrix Σ of pairwise dependence metrics. This matrix Σ can be shown to be completely positive, and a non-negative factorization of it can be used to estimate probabilities of extreme events or to simulate realizations with pairwise dependence, summarized by Σ. So this application is again depending on the completely positive factorizations itself such that proving complete positivity is not sufficient. This therefore gives a further motivation for the question of how to derive a factorization for completely positive matrices. For numerical details on this application, the reader is referred to Section 7.12.

As recently shown, completely positive matrices are also related to quantum physics, cf. [94].

Overall, we saw that for numerous optimization problems a convex reformulation can be found using the completely positive cone. As already mentioned, the whole complexity moves to the cone constraint. Therefore, it is important to decide whether a given matrix is completely positive or not, especially in terms of solving these problem reformulations or in terms of identifying the stable set of vertices, for instance. Therefore, we will have a closer look at the factorizations of completely positive matrices in the following chapter. As mentioned in Lemma 2.16, they provide a certificate for the matrix to be completely positive.

3 Factorizations for Completely Positive Matrices

In this chapter, we will show theoretical results to obtain a factorization $A = BB^T$ with $B \geq 0$ for a given completely positive matrix A. This is not least motivated by the applications in Section 2.7. Finding such a factorization would not only help to recover optimal solutions in combinatorial optimization problems, it also provides a certificate for $A \in \mathcal{CP}_n$, according to Lemma 2.16. Proving the membership to the completely positive cone is necessary for the completely positive reformulations, introduced in the previous section. This is in general an NP-hard task, as mentioned in [38, Theorem 5.3]. In addition to proving the membership to the completely positive cone, finding a factorization $A = BB^T$ where B is entrywise strictly positive would even prove that $A \in \operatorname{int}(\mathcal{CP}_n)$, a property that may be useful to ensure strict feasibility for a completely positive problem reformulation.

This chapter is organized as follows: Since other authors have studied the completely positive factorization problem before, a short survey is given in the following section. Afterwards, we will focus on the properties of completely positive factorizations and how they are related among each other. Here we will see that orthogonal matrices play a major role. With the help of orthogonal matrices, it is then possible to derive theoretical certificates for a matrix to be completely positive. It is important for these results to start with an initial factorization of the given matrix, which is not entrywise nonnegative in general. Therefore, we will see how to obtain such a factorization of arbitrary order. Moreover, in the last part of this chapter, we will see that the certificates for the completely positive cone can also be specified to verify the membership to the interior of the cone.

3.1 Related Work

Factorization of matrices with special structures has been studied for a few decades – here the reader is especially referred to the references given in [11] and [37]. Dickinson and Dür [37] extend this work and give a factorization algorithm for acyclic matrices which works in linear time. Also Bomze [14] deals with special structures and shows how a factorization of an $n \times n$ matrix can be constructed if a factorization of an $(n-1) \times (n-1)$ principal submatrix is known. For a general input matrix A, Jarre and Schmallowsky [62] use a quadratic factorization heuristic to generate a sequence of matrices BB^T that, for a suitable starting point, eventually converges to A. Their algorithm works well for matrices of up to order 200×200. For a comparison of this approach to the methods introduced in this thesis, the reader is referred to Section 7.11. Nie [79] treats the completely positive factorization problem as a special case of a $\mathcal{A}$-truncated K-moment

problem. For this more general problem, Nie develops an algorithm based on solving a sequence of (numerically expensive) semidefinite optimization problems. Nie reports numerical experiments for the factorization of completely positive matrices up to order 8×8. Sponsel and Dür [89] develop an algorithm for the projection of a matrix onto the dual of $\mathcal{CP}_n$, which can also be used to compute completely positive factorizations. However, for reasonably big input matrices, the algorithm runs into memory problems. Finally, Anstreicher, Burer, and Dickinson [3] are developing a factorization algorithm based on the ellipsoid method.

In the following section, we will show that cp-factorizations are not unique in general.

3.2 CP-Factorizations are not Unique

Let us recall Example 2.22 and the factorizations therein: Consider the matrix

$$A := \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix}.$$

Then $A = B_i B_i^T$ for each of the following matrices:

$$B_1 := \begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}, \qquad B_2 := \begin{pmatrix} 3 & 3 & 0 & 0 \\ 3 & 0 & 3 & 0 \\ 3 & 0 & 0 & 3 \end{pmatrix}, \qquad B_3 := \begin{pmatrix} 3 & 3 & 0 \\ 3 & 0 & 3 \\ 0 & 3 & 3 \end{pmatrix}. \tag{18}$$

For the same matrix A, we can consider another factorization of the same order as B_1 and B_3, based on the eigendecomposition of A. To be more precise, let

$$A = \underbrace{\begin{pmatrix} -0.4010 & 0.7112 & \frac{1}{\sqrt{3}} \\ 0.8165 & -0.0083 & \frac{1}{\sqrt{3}} \\ -0.4154 & -0.7029 & \frac{1}{\sqrt{3}} \end{pmatrix}}_{\text{eigenvectors}} \cdot \underbrace{\begin{pmatrix} 9 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 36 \end{pmatrix}}_{\text{eigenvalues}} \cdot \underbrace{\begin{pmatrix} -0.4010 & 0.7112 & \frac{1}{\sqrt{3}} \\ 0.8165 & -0.0083 & \frac{1}{\sqrt{3}} \\ -0.4154 & -0.7029 & \frac{1}{\sqrt{3}} \end{pmatrix}^T}_{\text{eigenvectors}}$$

be the eigendecomposition of A. Then we define

$$\begin{aligned} B_4 &:= \begin{pmatrix} -0.4010 & 0.7112 & \frac{1}{\sqrt{3}} \\ 0.8165 & -0.0083 & \frac{1}{\sqrt{3}} \\ -0.4154 & -0.7029 & \frac{1}{\sqrt{3}} \end{pmatrix} \cdot \begin{pmatrix} \sqrt{9} & 0 & 0 \\ 0 & \sqrt{9} & 0 \\ 0 & 0 & \sqrt{36} \end{pmatrix} \\ &= \begin{pmatrix} -1.2030 & 2.1337 & 3.4641 \\ 2.4494 & -0.0250 & 3.4641 \\ -1.2463 & -2.1087 & 3.4641 \end{pmatrix}, \end{aligned} \tag{19}$$

such that $B_4 B_4^T = A$.

Thus, if we would have only $B_4B_4^T$ as a factorization of A, we would not be able to deduce the membership of A to the completely positive cone since B_4 is not entrywise nonnegative. So the questions arises whether we can transform a given factorization to a different factorization. And especially, whether is it possible to transform any factorization of a given completely positive matrix A into a completely positive factorization. To answer this question, we will have a closer look at orthogonal matrices in the following section.

3.3 The Role of Orthogonal Matrices

Recall that a matrix $Q \in \mathbb{R}^{r\times r}$ is called orthogonal if $QQ^T = I_r$. The set of orthogonal matrices is a smooth manifold, also called the Stiefel manifold, see for example [1]. These matrices will be used to transform any given factorization of a completely positive matrix into a cp-factorization. We therefore consider the following definition.

Definition 3.1. *We denote by $\mathcal{O}_r$ the set of $r \times r$ orthogonal matrices, and we introduce*

$$\mathcal{O}_r^+ := \{Q \in \mathcal{O}_r \mid \det Q = 1\} \quad \textit{and} \quad \mathcal{O}_r^- := \{Q \in \mathcal{O}_r \mid \det Q = -1\}.$$

The first set is the set of rotation matrices, the latter is the set of reflection matrices.

Clearly, $\mathcal{O}_r = \mathcal{O}_r^+ \cup \mathcal{O}_r^-$ and hence $\mathcal{O}_r$ is nonconnected. It is well known that $\mathcal{O}_r$ is compact, as shown in the following Lemma.

Lemma 3.2. *The set $\mathcal{O}_r$ is a compact subset of $\mathbb{R}^{r\times r}$.*

Proof. Consider the mapping $f : \mathbb{R}^{r\times r} \to \mathbb{R}^{r\times r}, A \mapsto AA^T$. Then f is continuous and we have $\mathcal{O}_r = f^{-1}(I_r)$, where I_r denotes the $r \times r$ identity matrix. As a preimage of a singleton, $\mathcal{O}_r$ is closed. Since $\|Qx\| = \|x\|$ for each $Q \in \mathcal{O}_r$ and $x \in \mathbb{R}^r$, we have that $\mathcal{O}_r$ is also bounded. The theorem of Heine-Borel now proves that $\mathcal{O}_r$ is a compact set. □

The set $\mathcal{O}_r$ is not convex since for any $Q \in \mathcal{O}_r$ we have $-Q \in \mathcal{O}_r$, but on the other hand, $\frac{1}{2}Q + \frac{1}{2}(-Q) = 0 \notin \mathcal{O}_r$. However, using the well known Schur complement theorem, cf. [28, Theorem A.9], we can characterize the convex hull of the set of orthogonal matrices. To prove the Schur complement theorem, we need the following lemma, cf. [57, Section 1.1].

Lemma 3.3. *Let $A \in \mathbb{R}^{n\times n}$ be an arbitrary matrix and $Q \in \mathbb{R}^{n\times n}$ be a nonsingular matrix. Then we have:*

$$A \succeq 0 \iff QAQ^T \succeq 0,$$

where $A \succeq 0$ again indicates A being positive semidefinite.

Proof. Let $x \in \mathbb{R}^n$ be an arbitrary vector. Then A is positive semidefinite if and only if $x^TAx \geq 0$. With $y := (Q^{-T}x) \in \mathbb{R}^n$, this is equivalent to

$$0 \leq x^TQ^{-1}QAQ^TQ^{-T}x = (Q^{-T}x)^TQAQ^T(Q^{-T}x) = y^T(QAQ^T)y,$$

which is true if and only if QAQ^T is positive semidefinite. □

Theorem 3.4. *Consider the block matrix*

$$M = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix},$$

where A is positive definite and C is symmetric. Then the matrix

$$C - B^T A^{-1} B$$

is called the Schur complement of A in M. Moreover, the following are equivalent:

(a) $M \succeq 0$.

(b) $C - B^T A^{-1} B \succeq 0$.

Here $A \succeq 0$ again indicates that a given matrix A is positive semidefinite.

Proof. Let $D = -A^{-1}B$ and consider the matrix product

$$\begin{pmatrix} I & 0 \\ D^T & I \end{pmatrix} \begin{pmatrix} A & B \\ B^T & C \end{pmatrix} \begin{pmatrix} I & D \\ 0 & I \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & C - B^T A^{-1} B \end{pmatrix}.$$

Since

$$\det \begin{pmatrix} I & 0 \\ D^T & I \end{pmatrix} = 1,$$

Lemma 3.3 shows that

$$M \succeq 0 \quad \Leftrightarrow \quad \begin{pmatrix} I & 0 \\ D^T & I \end{pmatrix} \begin{pmatrix} A & B \\ B^T & C \end{pmatrix} \begin{pmatrix} I & D \\ 0 & I \end{pmatrix} \succeq 0 \quad \Leftrightarrow \quad \begin{pmatrix} A & 0 \\ 0 & C - B^T A^{-1} B \end{pmatrix} \succeq 0.$$

Due to the fact that a block diagonal matrix is positive (semi)definite if and only if its diagonal blocks are positive (semi)definite, the proof is complete. □

With the help of Theorem 3.4, we can prove the following equality.

Lemma 3.5. *We have*

$$\left\{ Q \in \mathbb{R}^{r \times r} \;\middle|\; QQ^T \preceq I_r \right\} = \left\{ Q \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} I_r & Q^T \\ Q & I_r \end{pmatrix} \succeq 0 \right\},$$

where I_r is again the $r \times r$ identity matrix.

Proof. Let

$$Q \in \left\{ Q \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} I_r & Q^T \\ Q & I_r \end{pmatrix} \succeq 0 \right\},$$

then by applying Theorem 3.4, we get the following equivalent statement since the identity matrix is positive definite and symmetric:

$$I_r - QI_r^{-1}Q^T \succeq 0 \Longleftrightarrow QQ^T \preceq I_r,$$

completing the proof. □

These sets will give a characterization of the convex hull of the set of orthogonal matrices. To see this, we consider the following characterization of $\operatorname{conv} \mathcal{O}_r$, as shown in [84, Proposition 4.8].

Lemma 3.6. *The set* $\operatorname{conv} \mathcal{O}_r$ *has the following representation:*

$$\operatorname{conv} \mathcal{O}_r = \left\{ X \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} 0 & X \\ X^T & 0 \end{pmatrix} \preceq I_{2r} \right\}, \tag{20}$$

where I_{2r} *denotes the identity matrix of order* $2r \times 2r$.

Proof. As in [84], we will prove both inclusions separately. First, consider an arbitrary matrix $Q \in \mathcal{O}_r$. Since $Q^TQ = I_r$, it follows that

$$\begin{pmatrix} I_r & -Q \\ -Q^T & I_r \end{pmatrix} = \begin{pmatrix} I_r \\ -Q^T \end{pmatrix} \begin{pmatrix} I_r & -Q \end{pmatrix} \succeq 0,$$

and by rearranging we see that Q is an element of the right hand side of equation (20). Since the right hand side is a convex set, it follows that

$$\operatorname{conv} \mathcal{O}_r \subseteq \left\{ X \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} 0 & X \\ X^T & 0 \end{pmatrix} \preceq I_{2r} \right\}.$$

For the reverse inclusion, assume that X is an element of the right hand side of equation (20) and consider its singular value decomposition. For a short review and some basic facts on the singular value decomposition, the reader is referred to the Appendix of this thesis. Thus, there exists a diagonal matrix Σ containing the singular values of X and orthogonal matrices $U, V \in \mathcal{O}_r$ such that $X = U\Sigma V^T$. Consider the orthogonal matrix

$$\begin{pmatrix} U^T & 0 \\ 0 & V^T \end{pmatrix}.$$

Then

$$\begin{pmatrix} U^T & 0 \\ 0 & V^T \end{pmatrix} \begin{pmatrix} 0 & X \\ X^T & 0 \end{pmatrix} \begin{pmatrix} U^T & 0 \\ 0 & V^T \end{pmatrix}^T = \begin{pmatrix} 0 & \Sigma \\ \Sigma & 0 \end{pmatrix}$$

and using Lemma 3.3, we see that

$$\begin{pmatrix} 0 & X \\ X^T & 0 \end{pmatrix} \preceq I_{2r} \Longleftrightarrow \begin{pmatrix} 0 & \Sigma \\ \Sigma & 0 \end{pmatrix} \preceq I_{2r},$$

which is equivalent to $-1 \leq \Sigma_{ii} \leq 1$ for $i \in \{1, \ldots, r\}$. It follows that $\Sigma \in \mathcal{D}_r \cap \operatorname{conv} \mathcal{O}_r$, where $\mathcal{D}_r$ denotes the set of (square) diagonal matrices of order r. Thus, $X = U\Sigma V^T \in \operatorname{conv} \mathcal{O}_r$. □

Now we can show that Lemma 3.5 provides two description of $\operatorname{conv} \mathcal{O}_r$:

Lemma 3.7. *We have*

$$\operatorname{conv} \mathcal{O}_r = \left\{ Q \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} I_r & Q^T \\ Q & I_r \end{pmatrix} \succeq 0 \right\}.$$

Proof. First, we observe that

$$\begin{pmatrix} I_r & Q^T \\ Q & I_r \end{pmatrix} \succeq 0 \quad \Leftrightarrow \quad \left(\begin{pmatrix} I_r & 0 \\ 0 & I_r \end{pmatrix} + \begin{pmatrix} 0 & Q^T \\ Q & 0 \end{pmatrix} \right) \succeq 0 \quad \Leftrightarrow \quad I_{2r} \succeq \begin{pmatrix} 0 & -Q^T \\ -Q & 0 \end{pmatrix}.$$

Thus, Lemma 3.6 yields:

$$Q \in \left\{ Q \in \mathbb{R}^{r \times r} \;\middle|\; \begin{pmatrix} I_r & Q^T \\ Q & I_r \end{pmatrix} \succeq 0 \right\} \Leftrightarrow I_{2r} \succeq \begin{pmatrix} 0 & -Q^T \\ -Q & 0 \end{pmatrix} \Leftrightarrow -Q^T \in \operatorname{conv} \mathcal{O}_r \Leftrightarrow Q \in \operatorname{conv} \mathcal{O}_r,$$

concluding the proof. □

The convex hull of the set of rotation matrices, $\operatorname{conv} \mathcal{O}_r^+$, can also be described by semidefiniteness constraints. To obtain the result by Saunderson et al. (cf. [84]), it is necessary to consider the following matrices.

Definition 3.8. *We will consider matrices $A_{ij} \in \mathcal{S}_{2^{r-1}}$ for $1 \leq i, j \leq r$. They are described as*

$$A_{ij} = -P_{\mathit{Saund}}^T \Lambda_i \Omega_j P_{\mathit{Saund}},$$

where Λ_i and Ω_j for $i, j = 1, \ldots, r$ are the $2^r \times 2^r$ skew-symmetric matrices defined as

$$\Lambda_i = \overbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}^{i-1 \text{ times}} \otimes \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \otimes \overbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}^{r-i \text{ times}}$$

$$\Omega_j = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}_{j-1 \text{ times}} \otimes \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \otimes \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}_{r-j \text{ times}}$$

and $P_{\mathit{Saund}} \in \mathbb{R}^{2^r \times 2^{r-1}}$ is the following matrix:

$$P_{\mathit{Saund}} = \frac{1}{2} \begin{pmatrix} I_{2^{r-1}} + \overbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}^{r-1 \text{ times}} \\ I_{2^{r-1}} - \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}_{r-1 \text{ times}} \end{pmatrix}.$$

As mentioned in [84], with $P_{\mathit{Saund}}^T M P_{\mathit{Saund}}$ we select a $2^{r-1} \times 2^{r-1}$ principal submatrix of an arbitrary matrix M. Moreover, since Λ_i and Ω_j are skew-symmetric and commute for every $1 \leq i, j \leq r$, the matrices A_{ij} are symmetric. Since Λ_i and Ω_j are signed permutation matrices, so is $-\Lambda_i \Omega_j$ such that all the entries of A_{ij} are elements of $\{-1, 0, 1\}$ for every $1 \leq i, j \leq r$.

With this definition, we can now give the representation of $\operatorname{conv} \mathcal{O}_r^+$ in the following Theorem, cf. [84, Theorem 1.3].

Theorem 3.9. *The convex hull* $\operatorname{conv} \mathcal{O}_r^+$ *can be described as follows:*

$$\operatorname{conv} \mathcal{O}_r^+ = \left\{ X \in \mathbb{R}^{r\times r} \;\middle|\; \begin{pmatrix} 0 & X \\ X^T & 0 \end{pmatrix} \preceq I_{2r},\ \sum_{i,j=1}^{r} A_{ij}(RX)_{ij} \preceq (r-2)I_{2^{r-1}} \right\},$$

where $R = \operatorname{Diag}(1, 1, \ldots, 1, -1) \in \mathbb{R}^{r\times r}$ *and the* A_{ij} *are the matrices introduced in Definition 3.8. For* $r \in \{2, 3\}$*, this simplifies to*

$$\operatorname{conv} \mathcal{O}_2^+ = \left\{ \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \in \mathbb{R}^{2\times 2} \;\middle|\; \begin{pmatrix} 1+c & s \\ s & 1-c \end{pmatrix} \succeq 0 \right\}$$

and

$$\begin{aligned} \operatorname{conv} \mathcal{O}_3^+ &= \left\{ X \in \mathbb{R}^{3\times 3} \;\middle|\; \sum_{i,j=1}^{3} A_{ij}(RX)_{ij} \preceq I_4 \right\} \\ &= \left\{ X \in \mathbb{R}^{3\times 3} \;\middle|\; \begin{pmatrix} 1-X_{11}-X_{22}+X_{33} & X_{13}+X_{31} & X_{12}-X_{21} & X_{23}+X_{32} \\ X_{13}+X_{31} & 1+X_{11}-X_{22}-X_{33} & X_{23}-X_{32} & X_{12}+X_{21} \\ X_{12}-X_{21} & X_{23}-X_{32} & 1+X_{11}+X_{22}+X_{33} & X_{31}-X_{13} \\ X_{23}+X_{32} & X_{12}+X_{21} & X_{31}-X_{13} & 1-X_{11}+X_{22}-X_{33} \end{pmatrix} \succeq 0 \right\}. \end{aligned}$$

This result is based on the equation

$$\operatorname{conv} \mathcal{O}_r^+ = \operatorname{conv} \mathcal{O}_r \cap (r-2)(\mathcal{O}_r^-)^\circ,$$

where $(\mathcal{O}_r^-)^\circ$ represents the polar of the set of reflection matrices. Theorem 3.9 can therefore be proven with the help of Lemma 3.6. The result can be easily extended to $\operatorname{conv} \mathcal{O}_r^-$, but as shown in Section 4.1, it is not necessary to analyse both components of $\mathcal{O}_r$ separately in our context. We will use the representation of the set of orthogonal matrices or reflection matrices to obtain a certificate for complete positivity in Chapter 6. On the other hand, we will use a further property of the set of orthogonal matrices to introduce a numerical method to show complete positivity in terms of semialgebraic sets. Expanding the condition $QQ^T = I_r$ into r^2 quadratic equations shows that $\mathcal{O}_r$ is a semialgebraic set, as we will see later in Section 6.1.

Hitherto, the connection between factorizations of complete positive matrices and orthogonal matrices is still missing. The following technical lemma will be used to prove a fundamental connection.

Lemma 3.10. *Let* $B, C \in \mathbb{R}^{n\times r}$ *with* $BB^T = CC^T$*. For* $i = 1, \ldots, n$*, let* B_i *resp.* $(BB^T)_i$ *denote the* i*-th rows of* B *resp.* BB^T*. And in the same way, for* $i = 1, \ldots, n$*, let* C_i *resp.* $(CC^T)_i$ *denote the* i*-th rows of* C *resp.* CC^T*. Further let* $R(B) \subseteq \mathbb{R}^r$ *resp.* $R(C) \subseteq \mathbb{R}^r$ *denote the subspaces spanned by the rows of* B *and* C*, respectively. Then:*

(a)

$$B_n = \sum_{i=1}^{n-1} \lambda_i B_i \quad \textit{if and only if} \quad \left(BB^T\right)_n = \sum_{i=1}^{n-1} \lambda_i \left(BB^T\right)_i, \tag{21}$$

with the same scalar values λ_i ($i = 1, \ldots, n$) in both equations.

(b) There exists a linear map $\varphi : R(B) \to R(C)$ such that $\varphi(B_i) = C_i$ for all $i = 1, \ldots, n$.

Proof. (a) We will prove both directions separately. To show that the right hand side in (21) is necessary, we assume that the left hand side holds and we consider the last row $\left(BB^T\right)_n$ of BB^T. Thus, we have

$$(BB^T)_n = B_n B^T = \left(\sum_{i=1}^{n-1} \lambda_i B_i\right) B^T = \sum_{i=1}^{n-1} \lambda_i B_i B^T = \sum_{i=1}^{n-1} \lambda_i \left(BB^T\right)_i,$$

such that the equation on the right hand side in (21) holds.

Conversely, assume that the equation on the right hand side of (21) holds. Since $\left(BB^T\right)_{ij} = B_i B_j^T$ for every $i, j = 1, \ldots, n$, we have for any entry $\left(BB^T\right)_{nj}$ of the row $(BB^T)_n$:

$$B_n B_j^T = \left(BB^T\right)_{nj} = \sum_{i=1}^{n-1} \lambda_i \left(BB^T\right)_{ij} = \sum_{i=1}^{n-1} \lambda_i B_i B_j^T$$

for every $j = 1, \ldots, n$. This gives

$$\left(B_n - \sum_{i=1}^{n-1} \lambda_i B_i\right) B_j^T = 0 \text{ for every } j = 1, \ldots, n.$$

This means that

$$z := \left(B_n - \sum_{i=1}^{n-1} \lambda_i B_i\right) \in \operatorname{span}\{B_1, \ldots, B_n\}^\perp$$

since the inner product of z with any row B_j is zero. At the same time, $z \in \operatorname{span}\{B_1, \ldots, B_n\}$ by construction. Consequently, $z = 0$, which now proves the equality on the left handside in (21).

(b) First, we observe that $\operatorname{rank}(B) = \operatorname{rank}\left(BB^T\right) = \operatorname{rank}\left(CC^T\right) = \operatorname{rank}(C)$ since the equality $BB^T = CC^T$ holds. If the matrices B and C are of full row-rank, the result in (b) is obvious. Now we assume that $\operatorname{rank}(B) = n - 1$ and without loss of generality, we assume that the last row B_n of B is linearly dependent on the rows $B_1, \ldots, B_{n-1}$. Thus, we have

$$B_n = \sum_{i=1}^{n-1} \lambda_i B_i \tag{22}$$

for some scalar values $\lambda_1, \ldots, \lambda_{n-1}$. Let φ denote the unique linear function with

$$\varphi(B_i) = C_i \quad \text{for all } i = 1, \ldots, n-1.$$

It remains to show that $\varphi(B_n) = C_n$. Applying part (a) to equation (22) shows

$$(BB^T)_n = \sum_{i=1}^{n-1} \lambda_i (BB^T)_i = \sum_{i=1}^{n-1} \lambda_i (CC^T)_i,$$

where the last equality holds since $BB^T = CC^T$. Again with part (a), now applied to the matrix CC^T in the last equation, we get

$$C_n = \sum_{i=1}^{n-1} \lambda_i C_i,$$

with the same scalar values $\lambda_i, \ldots, \lambda_{n-1}$. Finally, we get

$$\varphi(B_n) = \varphi\left(\sum_{i=1}^{n-1} \lambda_i B_i\right) = \sum_{i=1}^{n-1} \lambda_i \varphi(B_i) = \sum_{i=1}^{n-1} \lambda_i C_i = C_n,$$

concluding the case $\operatorname{rank}(B) = n - 1$.

If $\operatorname{rank}(B) < n - 1$, we can apply the same technique such that the linear map φ exists for any $\operatorname{rank}(B) \in \{1, \ldots, n\}$. This eventually proves part (b).

□

The next lemma is well known and can be found for instance in [96, Lemma 1]. The lemma will be fundamental for the main algorithm in Chapter 6 and illustrates how different factorizations of a matrix are related. It bridges the gap between the theory of orthogonal matrices and cp-factorizations.

Lemma 3.11. *Let $B, C \in \mathbb{R}^{n \times r}$. Then $BB^T = CC^T$ if and only if there exists $Q \in \mathcal{O}_r$ with $BQ = C$.*

Proof. The if part is obvious. To see the reverse, observe that B and C are of equal rank since $\operatorname{rank}(B) = \operatorname{rank}(BB^T) = \operatorname{rank}(CC^T) = \operatorname{rank}(C)$. Let B_i resp. C_i $(i = 1, \ldots, n)$ denote the rows of B resp. C, and let $R(B) \subseteq \mathbb{R}^r$ resp. $R(C) \subseteq \mathbb{R}^r$ denote the subspaces spanned by the rows of B and C, respectively. Due to Lemma 3.10 (b), there exists a linear map $\varphi : R(B) \to R(C)$ such that $\varphi(B_i) = C_i$ for all $i = 1, \ldots, n$.

Moreover, the equality $BB^T = CC^T$ entails that

$$\langle B_i, B_j \rangle = \langle \varphi(B_i), \varphi(B_j) \rangle \quad \text{for all } i, j \in \{1, \ldots, n\},$$

so φ is an isometry. Extending φ from $R(B)$ to an isometry on $\mathbb{R}^r$ gives the desired matrix Q. □

Lemma 3.11 therefore shows that two cp-factorizations for the same matrix are connected via an orthogonal matrix.

Getting back to Example 2.22 and the factorizations in (18) of the matrix

$$A = \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix},$$

Lemma 3.11 gives rise to the following equations.

$$\underbrace{\begin{pmatrix} 3 & 3 & 0 \\ 3 & 0 & 3 \\ 0 & 3 & 3 \end{pmatrix}}_{B_3} \cdot \underbrace{\begin{pmatrix} \frac{2}{3} & \frac{2}{3} & -\frac{1}{3} \\ \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} \\ -\frac{1}{3} & \frac{2}{3} & \frac{2}{3} \end{pmatrix}}_{Q} = \underbrace{\begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}}_{B_1}$$

Hence, we can transform one cp-factorization into a second one, provided the two factorization matrices are of the same order. Even for factorizations of the matrix, which are not entrywise nonnegative, like B_4 in (19), we can apply Lemma 3.11. This yields the following equation.

$$\underbrace{\begin{pmatrix} -1.2030 & 2.1337 & 3.4641 \\ 2.4494 & -0.0250 & 3.4641 \\ -1.2463 & -2.1087 & 3.4641 \end{pmatrix}}_{B_4} \cdot \underbrace{\begin{pmatrix} -0.4010 & 0.7112 & 0.5774 \\ 0.8165 & -0.0083 & 0.5774 \\ -0.4154 & -0.7029 & 0.5774 \end{pmatrix}}_{Q} = \underbrace{\begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}}_{B_1}$$

Remark 3.12. *We can transform any factorization $A = BB^T$, with $B \in \mathbb{R}^{n\times r}$ not necessarily entrywise nonnegative, of a completely positive matrix A with $\operatorname{cpr}(A) \leq r$ into a cp-factorization using an orthogonal matrix.*

As another application of Lemma 3.11, consider a decomposition $A = BB^T$, where B has one or multiple nonpositive columns and all the other columns are entrywise nonnegative. Then this is still a decomposition showing $A \in \mathcal{CP}_n$ according to the following argument:

Corollary 3.13. *Let $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $B_j \leq 0$ for some column indices $j \in J \subseteq \{1, \ldots, r\}$ and $B_i \geq 0$ for all $i \notin J$. Then there exists an orthogonal matrix $Q \in \mathbb{R}^{r\times r}$ such that $BQ \geq 0$. This proves $A \in \mathcal{CP}_n$.*

Proof. Let $Q \in \mathbb{R}^{r\times r}$ with $Q_{il} = 0$ for all $i \neq l$, $Q_{ii} = 1$ or all $i \notin J$ and $Q_{jj} = -1$ for all $j \in J$. Then $BQ \geq 0$ since the columns of BQ fulfill $(BQ)_i = B_i$ for all $i \notin J$ and $(BQ)_j = -B_j \geq 0$ for all $j \in J$. In addition, we have

$$(BQ)(BQ)^T = B\underbrace{QQ^T}_{I_n}B^T = BB^T = A$$

since Q is an orthogonal matrix. This completes the proof. □

Thus, we have the following additional characterization of completely positive matrices:

Corollary 3.14. *The set of completely positive matrices is equal to the following set:*

$$\mathcal{CP}_n = \{A \in \mathbb{R}^{n\times n} \mid A = BB^T, \text{ where } B \in \mathbb{R}^{n\times r},\ B_i \geq 0 \text{ or } B_i \leq 0 \text{ for every column } B_i\}.$$

Considering not only a single orthogonal matrix but a sequence of orthogonal matrices gives rise to the definition of a nearly positive matrix. We will have a closer look at these matrices in the following section, which will lead us to a sufficient condition for a matrix to be an element of the interior of the completely positive cone.

3.4 Nearly Positive Matrices

As mentioned in Remark 3.12, we can use orthogonal matrices to transform an arbitrary factorization $A = BB^T$ of a completely positive matrix A into a cp-factorization as long as the number of columns in B is greater than or equal to $\operatorname{cpr}(A)$. If we consider not only a single orthogonal matrix but a certain sequence of such matrices, we have the following definition, cf. [85].

Definition 3.15. *A matrix $B \in \mathbb{R}^{n\times r}$ is called nearly positive if there exists a sequence of matrices $(Q_l)_{l\in\mathbb{N}} \in \mathcal{O}_r$ such that*

$$\lim_{l\to\infty} Q_l = I_r \text{ and } BQ_l > 0 \text{ for all } l \in \mathbb{N}.$$

Then we have $\{B \in \mathbb{R}^{n\times r} \mid B \text{ nearly positive}\} \subseteq \{B \in \mathbb{R}^{n\times r} \mid B \geq 0\}$ according to the following argument:

Lemma 3.16. *Every nearly positive matrix is entrywise nonnegative.*

Proof. We have

$$0 \leq \lim_{l\to\infty} \underbrace{BQ_l}_{>0} = B \lim_{l\to\infty} Q_l = B \cdot I = B$$

and therefore $B \geq 0$. □

But the reverse subset implication does not hold, as the following lemma shows.

Lemma 3.17. *We have $\{B \in \mathbb{R}^{n\times r} \mid B \text{ nearly positive}\} \not\supseteq \{B \in \mathbb{R}^{n\times r} \mid B \geq 0\}$.*

Proof. For every matrix $A \in \mathcal{CP}_n \setminus \operatorname{int}(\mathcal{CP}_n)$ with $\operatorname{rank}(A) = n$ (see for examle A_{DS} in Example 2.43), there exists a matrix $B \in \mathbb{R}^{n\times \operatorname{cpr}(A)}$ with $B \geq 0$ and $A = BB^T$, but there does not exist a matrix $C \in \mathbb{R}^{n\times \operatorname{cpr}(A)}$ with $A = CC^T$ and $C > 0$. B is therefore entrywise nonnegative but not nearly positive, completing the proof. □

A simple necessary condition for an entrywise nonnegative matrix B to be nearly positive is the following, cf. [85, Proposition 2.3].

Lemma 3.18. *If $B \in \mathbb{R}^{n\times r}$ is nearly positive, then $BB^T > 0$.*

Proof. Consider $Q \in \mathcal{O}_r$ such that $BQ > 0$. Then $BB^T = BQQ^TB^T = (BQ)(BQ)^T > 0$. □

The main idea of nearly positive matrices is to slightly perturb nonnegative matrices into the interior of the nonnegative orthant. In the context of the interior of the completely positive cone, as described in Section 2.3, they can also be used in the following lemma, cf. [85, Proposition 7.3].

Lemma 3.19. *Let $B \in \mathbb{R}^{n\times r}$ be entrywise nonnegative and of full row-rank. Further assume that B is nearly positive. Then $BB^T \in \operatorname{int}(\mathcal{CP}_n)$.*

Proof. Since B is nearly positive, there exists an orthogonal matrix Q such that $BQ > 0$. Then $BB^T = (BQ)(BQ)^T$ and (BQ) is of full row rank, proving BB^T is of full rank and therefore we have $BB^T \in \operatorname{int}(\mathcal{CP}_n)$. □

Thus, the interior of the completely positive cone has the following inner approximation:

$$\left\{BB^T \;\middle|\; B \geq 0 \text{ nearly positive and } \operatorname{rank}(B) = n\right\} \subseteq \operatorname{int}(\mathcal{CP}_n).$$

Furthermore, a generalized version of this approach yields a certificate for complete positivity, as mentioned in the following section.

3.5 Further Conditions for Complete Positivity

A further theoretical condition for $A \in \mathcal{CP}_n$ can be deduced from Definition 3.15 and reads as follows:

Lemma 3.20. *Consider a matrix $A \in \mathbb{R}^{n\times n}$ with its factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$. If B is nearly positive, then $A \in \mathcal{CP}_n$.*

Proof. If B is nearly positive, there exists an orthogonal matrix $Q \in \mathbb{R}^{r\times r}$ such that $BQ > 0$. Since $A = (BQ)(BQ)^T$, we have $A \in \mathcal{CP}_n$. □

Here a strong condition has to hold to verify complete positivity. This lemma is based on an entrywise strictly positive factorization such that we can not apply this result to some matrices at the boundary of the completely positive cone. Thus, the condition in Lemma 3.20 is not necessary to show that a matrix is completely positive. For a necessary and sufficient condition for complete positivity, Lemma 3.11 gives rise to the following statement.

Lemma 3.21. *Let $A \in \mathbb{R}^{n\times n}$ with its factorization $A = BB^T$, where $B \in \mathbb{R}^{n\times r}$. Further assume that $r \geq \operatorname{cpr}(A)$. Then $A \in \mathcal{CP}_n$ if and only if there exists $Q \in \mathcal{O}_r$ such that $BQ \geq 0$.*

Proof. For the if part, let $Q \in \mathcal{O}_r$ such that $BQ \geq 0$. Then $A = (BQ)(BQ)^T$ is a cp-factorization proving $A \in \mathcal{CP}_n$. For the reverse part on the other hand, let $A \in \mathcal{CP}_n$. Since $r \geq \operatorname{cpr}(A)$, there exists a cp-factorization $A = CC^T$ with $C \in \mathbb{R}^{n\times r}$ and $C \geq 0$. Lemma 3.11 gives an orthogonal matrix $Q \in \mathbb{R}^{r\times r}$ such that $BQ = C \geq 0$, which completes the proof. □

For this lemma, it is necessary to start with a sufficient number of columns r in the given initial factorization $A = BB^T$. In the next section, we will see that it is always possible to generate such an initial factorization.

3.6 Generating Initial Factorizations of Arbitrary Order

We will now analyse how to generate initial factorizations $A = BB^T$ with variying numbers of columns in B. The following results can also be found in the submitted article [50].

To obtain an initial factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times n}$, we can for instance use the Cholesky decomposition $A = LL^T$, where L is a lower triangular matrix, or the eigenvalue decomposition $A = V\Sigma V^T$, by setting $B := V\Sigma^{\frac{1}{2}}$. In both cases, the factorization matrix is square but not necessarily entrywise nonnegative. To generate a factorization with $r > n$ columns, we will use a different approach. To ensure a correct choice of r, we note the following remark.

Remark 3.22. *Since it is known that* $\operatorname{cpr}(A)$ *can be considerably larger than* n *and in general it is not possible to compute* $\operatorname{cpr}(A)$*, we will use the upper bound* cp_n *for the cp-rank, as introduced in Lemma 2.32, for our desired number of columns in our initial factorization. So we set* $r = \operatorname{cp}_n$*, ensuring* $r \geq \operatorname{cpr}(A)$*, such that the number of columns in our initial factorization* B *will be sufficient to generate a cp-factorization.*

To obtain an initial factorization of order $n \times cp_n$, we will use the following two approaches: Consider an initial factorization $A = \tilde{B}\tilde{B}^T$ with $\tilde{B} \in \mathbb{R}^{n\times n}$ and assume that $\operatorname{cpr}(A)$ is unknown, but $\operatorname{cpr}(A) \leq \operatorname{cp}_n$. One way to construct an $n \times \operatorname{cp}_n$-matrix $\widehat{B}$ with $A = \widehat{B}\widehat{B}^T$ is to append $k := \operatorname{cp}_n - n$ zero columns to $\tilde{B}$, i.e.,

$$\widehat{B} := [\tilde{B}, 0_{n\times k}]. \tag{23}$$

Numerically, it turns out that using the following replication approach is more promising than using $\widehat{B}$ as our initial factorization, see Section 7.6 below.

Lemma 3.23. *Consider an initial factorization* $A = \tilde{B}\tilde{B}^T$ *with* $\tilde{B} \in \mathbb{R}^{n\times n}$*. Let* $\tilde{B}_j$ *denote the* j*-th column of* $\tilde{B}$*, and assume without loss of generality that* $\tilde{B}_n$ *is the column with the least number of negative entries. Now we decompose* $\tilde{B}_n$ *into* $m := cp_n - n + 1$ *columns to obtain*

$$B := \left[\tilde{B}_1, \ldots, \tilde{B}_{n-1}, \underbrace{\tfrac{1}{\sqrt{m}}\tilde{B}_n, \tfrac{1}{\sqrt{m}}\tilde{B}_n, \ldots, \tfrac{1}{\sqrt{m}}\tilde{B}_n}_{m \text{ columns}}\right] \in \mathbb{R}^{n\times \operatorname{cp}_n}. \tag{24}$$

Then $BB^T = A$*.*

Proof. Let B_j denote the j-th column of B. Then we get

$$\begin{aligned} BB^T &= \sum_{j=1}^{\operatorname{cp}_n} B_j B_j^T = \sum_{j=1}^{n-1} \tilde{B}_j\tilde{B}_j^T + \sum_{j=n}^{\operatorname{cp}_n} \left(\frac{1}{\sqrt{m}}\tilde{B}_n\right)\left(\frac{1}{\sqrt{m}}\tilde{B}_n\right)^T \\ &= \sum_{j=1}^{n-1} \tilde{B}_j\tilde{B}_j^T + \frac{1}{m}(cp_n - n + 1)\tilde{B}_n\tilde{B}_n^T = \sum_{j=1}^{n} \tilde{B}_j\tilde{B}_j^T, \end{aligned}$$

where the last equality follows by the definition of m. Since $\tilde{B}\tilde{B} = A$, the proof is complete. □

Remark 3.24. *For this theoretical result, the column can be arbitrarily chosen. Here it is recommended to pick the column with the least number of negative entries. In some cases, there exists a strictly positive column. If A is positive definite for example, then the first column of the Cholesky decomposition is entrywise strictly positive.*

For a concrete example of this approach, consider the matrices B_2 and B_3 in Example 2.22. Then picking the last column of B_3 for the replication as introduced in equation (24) gives:

$$B_3 = \begin{pmatrix} 3 & 3 & \mathbf{0} \\ 3 & 0 & \mathbf{3} \\ 0 & 3 & \mathbf{3} \end{pmatrix} \Longrightarrow \underbrace{\begin{pmatrix} 3 & 3 & \mathbf{0} & \mathbf{0} \\ 3 & 0 & \frac{\mathbf{3}}{\sqrt{\mathbf{2}}} & \frac{\mathbf{3}}{\sqrt{\mathbf{2}}} \\ 0 & 3 & \frac{\mathbf{3}}{\sqrt{\mathbf{2}}} & \frac{\mathbf{3}}{\sqrt{\mathbf{2}}} \end{pmatrix}}_{=:B},$$

such that $B_3 B_3^T = BB^T$. The generated matrix B and the matrix B_2 in Example 2.22 are of the same order such that we can apply Lemma 3.11 to guarantee the existence of an orthogonal matrix $Q \in \mathbb{R}^{4\times 4}$, which yields

$$BQ = B_2 \Longleftrightarrow \underbrace{\begin{pmatrix} 3 & 3 & 0 & 0 \\ 3 & 0 & 3/\sqrt{2} & 3/\sqrt{2} \\ 0 & 3 & 3/\sqrt{2} & 3/\sqrt{2} \end{pmatrix}}_{B\in\mathbb{R}^{3\times 4}} \cdot \underbrace{\begin{pmatrix} 1/2 & 1/2 & 1/2 & -1/2 \\ 1/2 & 1/2 & -1/2 & 1/2 \\ 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \end{pmatrix}}_{Q\in\mathbb{R}^{4\times 4}} = \underbrace{\begin{pmatrix} 3 & 3 & 0 & 0 \\ 3 & 0 & 3 & 0 \\ 3 & 0 & 0 & 3 \end{pmatrix}}_{B_2\in\mathbb{R}^{3\times 4}}.$$

This shows that if we apply Lemma 3.23 in advance, we can transform two different factorizations into one another, even though they are of different order. Therefore, consider the factorization matrix with the least number of columns and pick the column with the least number of negative entries and replicate this column until the number of columns in the second factorization matrix is reached. Then Lemma 3.11 returns the orthogonal transformation matrix. We will use this method, combined with the results in Lemma 3.21, to introduce a new algorithmic method to check complete positivity in Chapter 6. The main idea of this approach is given in the following remark.

Remark 3.25. *Let $A \in \mathbb{R}^{n\times n}$ and consider an initial factorization $A = BB^T$ (resp. $A = \widehat{B}\widehat{B}^T$), generated using Lemma 3.23 (resp. the method in equation* (23)*) such that $B \in \mathbb{R}^{n\times \mathrm{cp}_n}$ (resp. $\widehat{B} \in \mathbb{R}^{n\times \mathrm{cp}_n}$). Lemmas 2.32 and 3.11 now prove that $A \in \mathcal{CP}_n$ if and only if there exists $Q \in \mathcal{O}_{\mathrm{cp}_n}$ such that $BQ \geq 0$ (resp. $\widehat{B}Q \geq 0$).*

This gives rise to the following observation:

Remark 3.26. *Let $A \in \mathcal{CP}_n$. Then it is possible to obtain a cp-factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ for any $r \in [\mathrm{cpr}(A), \mathrm{cp}_n^+]$, even for any $r \geq \mathrm{cpr}(A)$.*

In addition, based on Lemma 3.21, it is possible to derive an optimization problem, whose optimal value can verify whether a given matrix is an element of the interior of the completely positive cone. The details of this approach are given in the following section.

3.7 Generating Factorizations for Matrices in the Interior via Maximization Problems

In this section, we will introduce a certificate which can be used to prove the membership of a given nonsingular matrix to the interior of the completely positive cone. As shown in Theorem 2.19, for $A \in \mathbb{R}^{n\times n}$ nonsingular, it is sufficient to generate a factorization $A = BB^T$, where $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$ such that $B \geq 0$ and $B_i > 0$ for at least one column B_i, in order to show $A \in \operatorname{int}(\mathcal{CP}_n)$.

Inspired by Lemma 3.21, the idea of the following approach is to start with an arbitrary initial factorization $A = \tilde{B}\tilde{B}^T$, where $\tilde{B} \in \mathbb{R}^{n\times n}$ is not necessarily nonnegative, and to construct from that a factorization $A = BB^T$, where $B \in \mathbb{R}^{n\times r}$ is again not necessarily entrywise nonnegative, but $r \geq \operatorname{cpr}^+(A)$. Here again, the method introduced in Lemma 3.23 can be used. But since the cp$^+$-rank can be larger than the cp-rank, as shown in Section 2.4, it may become necessary to increase the number of columns in the initial factorization to cp_n^+, the upper bound for the cp$^+$-rank given in Lemma 2.32. For this, we will use cp_n^+ instead of cp_n for the number of columns in Lemma 3.23 for the results in this section, in case the exact cp$^+$-rank is unknown.

Now let $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$ and consider the following problems:

$$\begin{aligned} \max \quad & \varepsilon \\ \text{s.t.} \quad & (BQ)_{ij} \geq \begin{cases} \varepsilon, & j = 1,\ i = 1,\dots,n \\ 0, & j \neq 1,\ i = 1,\dots,n \end{cases} \\ & Q \in \mathcal{O}_r \end{aligned} \tag{25}$$

and

$$\begin{aligned} \max \quad & \varepsilon \\ \text{s.t.} \quad & BQ \geq \varepsilon E_{n\times r} \\ & Q \in \mathcal{O}_r, \end{aligned} \tag{26}$$

where $E_{n\times r}$ again denotes the all-ones matrix, in this case of order $n \times r$. Then the following lemma holds.

Lemma 3.27. *Let $A = BB^T$ as considered above. Then $A \in \operatorname{int}(\mathcal{CP}_n)$ if and only if A has full rank and the optimal value ε^* of problem* (25) *or* (26) *is strictly positive.*

Proof. First, we will show that a strictly positive optimal value in either of the problems is sufficient to show $A \in \operatorname{int}(\mathcal{CP}_n)$. For this, let $\varepsilon^* > 0$ for problem (25) and assume that A is of full rank. Then we get for the optimal solution Q^* of (25)

$$BQ^* \geq \varepsilon^* \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix},$$

such that $\operatorname{rank}(BQ^*) = n$, $BQ^* \geq 0$ and the first column $(BQ^*)_1 > 0$. In addition, we have

$A = (BQ^*)(BQ^*)^T$ since $Q^* \in \mathcal{O}_r$. With Theorem 2.19 and especially equation (8) therein, we finally get $A \in \operatorname{int}(\mathcal{CP}_n)$.

If on the other hand $\varepsilon^* > 0$ for problem (26) and A is again of full rank, we get for the optimal solution Q^* of (26)

$$BQ^* \geq \varepsilon^* E_{n\times r} > 0.$$

Again, we have $A = (BQ^*)(BQ^*)^T$ since $Q^* \in \mathcal{O}_r$. Thus, Theorem 2.19 and especially equation (7) therein shows $A \in \operatorname{int}(\mathcal{CP}_n)$.

For the reverse part, we assume that $A \in \operatorname{int}(\mathcal{CP}_n)$. Then A has full rank according to Theorem 2.19 and there exists a matrix C, written columnwise as $[C_1, \ldots, C_r]$, such that $A = CC^T$, $C \geq 0$ and $C_1 > 0$. Due to Lemma 3.11, there exists an orthogonal matrix $Q \in \mathcal{O}_r$ such that $BQ = C$. This Q is feasible for (25), so the optimal value fulfills

$$\varepsilon^* \geq \min_{i\in\{1,\ldots,n\}} C_{i1} > 0.$$

On the other hand, according to Theorem 2.19, there also exists a matrix $D \in \mathbb{R}^{n\times r}$ such that $A = DD^T$ and $D > 0$. With the help of Lemma 3.11, there exists another orthogonal matrix $Q_2 \in \mathcal{O}_r$ such that $BQ_2 = D$. This Q_2 is feasible for (26), so the optimal value fulfills

$$\varepsilon^* \geq \min_{\substack{i\in\{1,\ldots,n\},\\ j\in\{1,\ldots,r\}}} D_{ij} > 0$$

and the proof is complete. □

So, this lemma allows us to check membership to the completely positive cone and its interior by solving a maximization problem. Since the set of orthogonal matrices is not a convex set, as shown in Section 3.3, the problems (25) and (26) are not convex. Thus, these problems are hard to solve and Lemma 3.27 clearly provides only a theoretical result.

A first idea to overcome the lack of convexity would be to convexify the orthogonality constraint and therefore to optimize over the convex hull of the set of orthogonal matrices, which was given in Lemma 3.7. This gives rise to the following problem:

$$\begin{aligned} \max \quad & \varepsilon \\ \text{s.t.} \quad & (BQ)_{ij} \geq \begin{cases} \varepsilon, & j = 1,\ i = 1,\ldots,n \\ 0, & j \neq 1,\ i = 1,\ldots,n \end{cases} \\ & \begin{pmatrix} I & Q^T \\ Q & I \end{pmatrix} \succeq 0. \end{aligned} \tag{27}$$

As the following example shows, the results in Lemma 3.27 will not hold for problem (27).

Example 3.28. *Consider the matrix*

$$A = \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix}$$

as given in Example 2.22 and its initial factorization, cf. equation (19)*:*

$$A = B_4 B_4^T \text{ with } B_4 = \begin{pmatrix} -1.2030 & 2.1337 & 3.4641 \\ 2.4494 & -0.0250 & 3.4641 \\ -1.2463 & -2.1087 & 3.4641 \end{pmatrix}.$$

Since we already know that $\mathrm{cpr}^+(A) = 3$ *(see Remark 2.36), it is not necessary to replicate columns to obtain a tractable initial factorization. To solve the problem* (27)*, we use Matlab and especially we apply SDP solvers like SDPT3, cf. [92] or [93]. We will again use these solver for the numerical experiments in Chapter 7. Now solving problem* (27) *with initial factorization matrix* B_4 *returns the following matrix* Q *as an optimal solution:*

$$Q = \begin{pmatrix} 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 \\ 1.0000 & 0.0001 & 0.0001 \end{pmatrix}.$$

Then we have

$$B_4 Q = \begin{pmatrix} 3.4641 & 0.0002 & 0.0002 \\ 3.4641 & 0.0002 & 0.0002 \\ 3.4641 & 0.0002 & 0.0002 \end{pmatrix},$$

but obviously $Q \notin \mathcal{O}_3$ *such that*

$$(B_4 Q)(B_4 Q)^T = \begin{pmatrix} 12.0000 & 12.0000 & 12.0000 \\ 12.0000 & 12.0000 & 12.0000 \\ 12.0000 & 12.0000 & 12.0000 \end{pmatrix} \neq A.$$

So finally, even though the optimal value $\varepsilon^* = 3.4641 > 0$ *and* Q *is optimal for* (27)*, we can not verify whether* A *is an element of the interior of the completely positive cone by solving problem* (27)*.*

Thus, the convexified version of problem (25) is solvable, but does not give any certificate for complete positivity, as long as the resulting matrix Q is not orthogonal. Thus, the result in Lemma 3.27 can not be extended to the convex problem in (27).

But the method shown in this section now motivates the idea to verify complete positivity by solving a certain feasibility problem. We will therefore modify the problem in (26). Here the details are given in the following chapter.

4 The Factorization Problem as a Nonconvex Feasibility Problem

This chapter is organized as follows: First, we will see how to prove complete positivity based on certain nonconvex feasibility problems. In the second part, we will show how these feasibility problems can be generalized in order to prove that the given matrix is even an element of the interior of the completely positive cone.

4.1 Feasibility Problems to Verify Complete Positivity

From now on, we will assume that we are given an initial factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$, where either $r = \operatorname{cpr}(A)$ if this quantity happens to be known, or otherwise we use the bound from Lemma 2.32 and set $r = \operatorname{cp}_n$. The problem of finding a completely positive factorization of A can then be formulated as the following feasibility problem, cf. [50, Equation (3)]:

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq 0 \\ & Q \in \mathcal{O}_r. \end{aligned} \tag{28}$$

Since a factorization is always a certificate for the matrix to be completely positive, we can derive the following result.

Theorem 4.1. *Let $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}(A)$. Moreover, consider the problem* (28)*. Then $A \in \mathcal{CP}_n$ if and only if* (28) *is feasible.*

Proof. For the if part, let $Q \in \mathcal{O}_r$ such that $BQ \geq 0$. Then $A = (BQ)(BQ)^T$ is a cp-factorization such that $A \in \mathcal{CP}_n$. For the reverse part, Lemma 3.21 returns a matrix Q which is feasible for (28), concluding the proof. □

In addition, consider the following problem, which yields another certificate for complete positivity.

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq 0 \\ & Q \in \mathcal{O}_r^+. \end{aligned} \tag{29}$$

Then the problems (28) and (29) are equivalent according to the following argument:

Lemma 4.2. *Problem* (28) *is feasible if and only if problem* (29) *is.*

Proof. The if part is obvious. For the reverse part, we consider two cases. If problem (28) returns an orthogonal matrix $Q \in \mathcal{O}_r^+$, this matrix Q is also feasible for (29). If on the other hand problem (28) returns a matrix $Q^- \in \mathcal{O}_r^-$, we can multiply Q^- with a permutation matrix $P \in \mathcal{O}_r^-$, which permutes two columns. Then the matrix $Q^-P =: Q \in \mathcal{O}_r^+$ is also feasible for (29). This concludes the proof. □

This shows that it is not necessary to consider the set $\mathcal{O}_r^-$ separately, as mentioned in Section 3.3. Unfortunately, neither $\mathcal{O}_r$ nor $\mathcal{O}_r^+$ are convex sets such that neither problem (28) nor (29) is a convex problem. Thus, they are hard to solve and again, a first idea to obtain a solvable problem would be to consider the convex hulls of $\mathcal{O}_r$ or $\mathcal{O}_r^+$. As shown in Lemma 3.7, it is possible to use a certain semidefinite relaxation of the orthogonality constraint such that convexifying (28) yields the semidefinite feasibility problem

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq 0 \\ & \begin{pmatrix} I & Q^T \\ Q & I \end{pmatrix} \succeq 0. \end{aligned} \tag{30}$$

Problem (29) can be convexified using the technique by Saunderson et al. (cf. [84]), as shown in Theorem 3.9. This again leads to a semidefinite feasibility problem, however one of considerably larger size.

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq 0 \\ & \begin{pmatrix} 0 & Q \\ Q^T & 0 \end{pmatrix} \preceq I_{2r} \\ & \sum_{i,j=1}^{r} A_{ij}(RQ)_{ij} \preceq (r-2)I_{2^{r-1}}, \end{aligned} \tag{31}$$

where $R = \operatorname{Diag}(1, 1, \ldots, 1, -1)$ and A_{ij} are the matrices introduced in Definition 3.8.

Remark 4.3. *It should be noted that* (30) *is always feasible (take $Q = 0$) and can therefore not be used to refute the membership of A to the completely positive cone.*

On the other hand, problem (31) can be used to refute the membership to the completely positive cone, as the following lemma shows.

Lemma 4.4. *If problem* (31) *is infeasible, then this certifies that the input matrix A is not completely positive.*

Proof. We prove this result by contradiction. For this, we assume that $A \in \mathcal{CP}_n$, but there does not exist a matrix $Q \in \operatorname{conv} \mathcal{O}_r^+$ such that $BQ \geq 0$ entrywise. Thus, according to Lemma 4.2, there does not exist a $Q \in \mathcal{O}_r$ such that $BQ \geq 0$. Since $A \in \mathcal{CP}_n$, Remark 3.26 gives a factorization $A = CC^T$ with $C \in \mathbb{R}^{n \times r}$, $r = \operatorname{cp}_n$ and $C \geq 0$. Since $B \in \mathbb{R}^{n \times r}$, Lemma 3.11 gives an orthogonal matrix Q such that $BQ = C \geq 0$. This yields a contradiction, completing the proof. □

Nevertheless, both problems (30) and (31) can give a certificate for complete positivity. They can be solved by applying SDP solvers, using interior point methods.

Lemma 4.5. *If solving either of the convexified problems happens to provide a $Q \in \mathcal{O}_r$ (resp. $Q \in \mathcal{O}_r^+$), then we have derived a completely positive factorization of A.*

Proof. Let $Q \in \mathcal{O}_r$ (resp. $Q \in \mathcal{O}_r^+$). Then we have $A = (BQ)(BQ)^T$ since Q is orthogonal, and $BQ \geq 0$ since Q solves the problem (30), or problem (31), respectively. □

Unfortunately, in numerical tests with randomly generated completely positive input matrices A, we have always observed the third case: the convexified problems were feasible, but the resulting matrix Q that could be obtained was not orthogonal. In this case, nothing can be inferred about A. The following example corroborates this observation.

Example 4.6. *Consider the matrix*

$$A = \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix}$$

as given in Example 2.22 and its initial factorization, cf. equation (19)*:*

$$A = B_4 B_4^T \text{ with } B_4 = \begin{pmatrix} -1.2030 & 2.1337 & 3.4641 \\ 2.4494 & -0.0250 & 3.4641 \\ -1.2463 & -2.1087 & 3.4641 \end{pmatrix}.$$

Since $\mathrm{cp}_3 = 3$ *and we already know that* $\mathrm{cpr}(A) = 3$ *(see Remark 2.36), it is not necessary to replicate columns for our initial factorization. Problem* (31) *then returns the following matrix as a feasible solution.*

$$Q = \begin{pmatrix} -0.0004 & -0.0004 & -0.0004 \\ -0.0006 & -0.0006 & -0.0006 \\ 0.4821 & 0.4825 & 0.4817 \end{pmatrix}.$$

Indeed, this matrix is feasible for (31) *and we have*

$$B_4 Q = \begin{pmatrix} 1.6692 & 1.6705 & 1.6678 \\ 1.6690 & 1.6703 & 1.6677 \\ 1.6718 & 1.6732 & 1.6704 \end{pmatrix} \geq 0.$$

But since Q is not an orthogonal matrix, this will not give a cp-factorization. We get

$$(B_4 Q)(B_4 Q)^T = \begin{pmatrix} 8.3582 & 8.3575 & 8.3716 \\ 8.3575 & 8.3568 & 8.3709 \\ 8.3716 & 8.3709 & 8.3851 \end{pmatrix} \neq A.$$

Consequently, convexifying problems (28) or (29) does not produce any useful insight, so we need to pursue a different approach. In Chapter 5, we will see an introduction to the alternating projections method, which will yield an applicable algorithm to obtain a cp-factorization of a completely positive matrix and therefore a certificate for complete positivity. This result will be

based on problem (28). Before that, we will derive a similar feasibility problem as in (28) to prove membership of a matrix to the interior of the completely positive cone.

4.2 Feasibility Problems for Matrices in the Interior of the Completely Positive Cone

In this section, we will assume that we are given an initial factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$, where either $r = \operatorname{cpr}^+(A)$ if this quantity happens to be known, or otherwise we use the upper bound from Lemma 2.32 and set $r = \operatorname{cp}_n^+$.

Based on (25) and (26), we consider the following feasibility problems for some $\varepsilon > 0$:

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & (BQ)_{ij} \geq \begin{cases} \varepsilon, & j = 1,\ i = 1,\dots,n \\ 0, & j \neq 1,\ i = 1,\dots,n \end{cases} \\ & Q \in \mathcal{O}_r \end{aligned} \tag{32}$$

and

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq \varepsilon E_{n\times r} \ . \\ & Q \in \mathcal{O}_r \end{aligned} \tag{33}$$

Similar to Theorem 4.1, we get the following result:

Theorem 4.7. *Let $\varepsilon > 0$ be small enough and $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$. Further let A be of full rank and consider the problems* (32) *and* (33)*. Then $A \in \operatorname{int}(\mathcal{CP}_n)$ if and only if* (32) *or* (33) *is feasible.*

Proof. For the if part let $Q \in \mathcal{O}_r$ such that $BQ \geq \varepsilon E_{n\times r}$ or

$$BQ \geq \begin{cases} \varepsilon, & j = 1,\ i = 1,\dots,n \\ 0, & j \neq 1,\ i = 1,\dots,n. \end{cases}$$

Then $A = (BQ)(BQ)^T$ is a cp-factorization showing $A \in \operatorname{int}(\mathcal{CP}_n)$ according to Theorem 2.19. For the reverse part, assume that $A \in \operatorname{int}(\mathcal{CP}_n)$. Thus, there exists a factorization $A = CC^T$, where $C \in \mathbb{R}^{n\times r}$ is entrywise strictly positive. Since B and C are of the same order, Lemma 3.11 provides an orthogonal matrix $Q \in \mathcal{O}_r$ such that $BQ = C$. The matrix Q is then feasible for both problems (32) and (33) if $\varepsilon > 0$ is chosen small enough. This completes the proof. □

But again, the problems (32) and (33) are not convex and therefore they are hard to solve. Similar to the approaches in (30) or (31), it would be possible to convexify these problems using the convex hull of the set of orthogonal matrices or rotation matrices. Unfortunately, this leads to the same drawback as shown in Example 4.6. So, to solve the feasibility problems (28), (29), (32) and (33), we need to pursue a different approach. To obtain a tractable method to solve the feasibility problems, we will use the method of alternating projections. In the following chapter, we give a survey of this method for several types of sets.

5 Alternating Projections

In this chapter, we give an outline of the alternating projections method for several types of sets. For an introduction to this topic, the reader is referred to [6], [31] and [51]. Moreover, we will have a look at an extension of this method, considering more than two sets. This will lead us to the so called cyclic projections approach. In this context, some known facts for subspaces and convex sets will be given. Considering a sequence of manifolds, we will also prove a convergence result for the cyclic projections method in this setting. In the end of this chapter, we will have a closer look at how to obtain an element in the intersection of two semialgebraic sets, based on the results in [41]. The latter will be used to solve the feasibility problems mentioned in the previous chapter.

But first, we consider a suitable definition for the projection onto a closed subset M of a Hilbert space H. Throughout this chapter, $\|\cdot\|$ will denote the norm induced by the scalar product of H.

Definition 5.1. *Let H be a Hilbert space and let $M \subseteq H$ be a closed subset. The projection of a point $x \in H$ onto M will be denoted by $P_M(x)$ and is the best approximation to x from M, i.e.*

$$\|x - P_M(x)\| = \min_{y \in M} \|x - y\|.$$

Here it should be mentioned that the projection of x onto M is not necessarily unique. For example, consider the boundary of the closed unit ball $M := B_1(0) \setminus \operatorname{int}(B_1(0))$ in $\mathbb{R}^n$ and $x = 0$, then $P_M(0) = M$.

We will use the method of alternating projections to obtain points in the intersection of two or more sets. Considering the case of two linear subspaces of a Hilbert space motivates the first approach on alternating projections by von Neumann, which was first published in 1933 (cf. [95]) and can also be found in [6].

5.1 Alternating Projections on Subspaces

Let A, B be closed linear subspaces of a Hilbert space H and consider the task of finding a point in the intersection $A \cap B$. We will use the following sequence to obtain a point in the intersection of the subspaces algorithmically, starting from an arbitrary point $x \in H$. This definition can be found for example in [6].

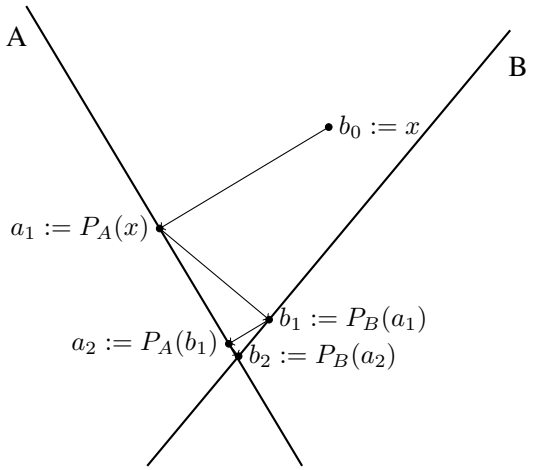
Definition 5.2. *Let A, B be closed linear subspaces in a Hilbert space H. Given a starting point $x \in H$, we define:*

$$b_0 := x, \quad a_n := P_A(b_{n-1}), \quad b_n := P_B(a_n) \quad \textit{for every} \quad n \geq 1,$$

based on the projection introduced in Definition 5.1. The sequences $(a_n)_{n\in\mathbb{N}}$ *and* $(b_n)_{n\in\mathbb{N}}$ *are called von Neumann sequences and the sequence* $(b_0, a_1, b_1, a_2, b_2, \dots)$ *is called the alternating von Neumann sequence.*

Geometrically, this can be interpreted as follows: To find the best approximation to x in $A \cap B$, we first project the starting point x orthogonally onto the set A. The resulting element in A is then orthogonally projected onto the set B. This point is again projected onto A and so on, until a point in the intersection $A \cap B$ is reached. Note, by the way, that $A \cap B \neq \emptyset$: since A, B are linear subspaces, we have $0 \in A \cap B$. As mentioned in [6], this approach has advantages whenever the projections onto A and B are easier to calculate than the projection onto the intersection $A \cap B$.

Figure 5.1: Alternating projections for two linear subspaces in $\mathbb{R}^2$



This procedure is illustrated in Figure 5.1. In this figure, we can see that for the given starting point x, we set $b_0 := x$ and start the procedure with an orthogonal projection of b_0 onto A. We then continue with alternating projections onto B and A. Eventually, the limits of the von Neumann sequences provide an element of the intersection $A \cap B$, which is a singleton in this special case. This visualizes the idea of alternating projections for closed linear subspaces in the 2-dimensional space. The approach extends naturally to higher dimensional spaces as well.

Based on Definition 5.2, we will mention the main result for alternating projections for two closed linear subspaces, which was first given by von Neumann in [95] and can also be found in [6] and [31]. For a proof of this result, see also [24, Theorem 5.1.5].

Theorem 5.3. *Let* A, B *be closed linear subspaces in a Hilbert space* H*. Then the von Neumann sequences and the alternating von Neumann sequence converge to* $P_{A\cap B}(x)$ *in norm.*

Thus, the mentioned sequences converge to the projection of the starting point x onto the intersection. More precisely, we can write this result as

$$\lim_{n\to\infty} \|a_n - P_{A\cap B}(x)\| = 0.$$

This is a very strong result, which can not be equivalently extended to other types of sets, as we will see later in this chapter.

Remark 5.4. *As Figure 5.1 illustrates for the alternating projections between subspaces, the convergence result does not require any constraints on the starting point. We therefore have a global convergence result.*

Hitherto, we considered only two closed linear subspaces and their intersection in a Hilbert space H. The following result shows that this idea can also be extended to multiple closed linear subspaces in H. Here it becomes necessary to project onto the subspaces in a certain order and to stick to this order for the entire process. Therefore, this method is called cyclic projections method. This generalized approach was introduced by Halperin in [54] and can also be found in [5], [24] and [31].

Theorem 5.5. *Let $V_1, \dots, V_k$ be closed linear subspaces of a Hilbert space H. Further define $V := \bigcap_{i=1}^{k} V_i$ and consider an arbitrary starting point $x \in H$. Then*

$$\lim_{n\to\infty} \|(P_{V_k} P_{V_{k-1}} \dots P_{V_1})^n x - P_V x\| = 0.$$

So, we have a global convergence result for the cyclic projections approach for a finite sequence of subspaces. Again, this method converges in norm to the projection of the starting point onto the intersection of the subspaces. Note again that $V \neq \emptyset$ since $0 \in V$.

Remark 5.6. *Geometrically this process can be interpreted as follows: We start with a projection of the starting point onto the first subspace V_1 and the thus generated point is projected onto the second subspace V_2. In general, the point in subspace V_i is then projected onto the next subspace V_{i+1} until V_k is reached. The point in V_k is then again projected onto the set V_1 and a new cycle starts. This method terminates whenever a fixed point of this cycle is reached. More formally, we can write the cyclic projections approach for a given starting point $x \in H$ as*

$$v_1^1 := P_{V_1}(x)\ ,\ v_2^1 := P_{V_2}(v_1^1)\ , \dots ,\ v_k^1 := P_{V_k}(v_{k-1}^1)$$

such that

$$v_k^1 = (P_{V_k} P_{V_{k-1}} \dots P_{V_1})(x),$$

closing the first cycle. The second cycle then reads as

$$v_1^2 := P_{V_1}(v_k^1)\ ,\ v_2^2 := P_{V_2}(v_1^2)\ , \dots ,\ v_k^2 := P_{V_k}(v_{k-1}^2)$$

such that

$$v_k^2 = (P_{V_k} P_{V_{k-1}} \dots P_{V_1})^2(x).$$

For the n-th cycle, this yields

$$v_k^n = (P_{V_k} P_{V_{k-1}} \dots P_{V_1})^n(x).$$

Theorem 5.5 therefore shows that this procedure converges to a fixed point, i.e., a point z with

$$(P_{V_k} P_{V_{k-1}} \dots P_{V_1})(z) = z.$$

Convergence in norm, as shown in Theorems 5.3 and 5.5, does not give any information about the convergence rate. In the following, we will see that it is possible to determine the convergence rate of the alternating projections approach for two or more subspaces. For these results, the reader is referred for example to [4, Section 1 and 2] or [31, Section 6]. The key aspects are also collected in the following results. Here, the convergence rate will depend on the angle between the subspaces and therefore we consider the definition of the so called Friedrichs angle first.

Definition 5.7. *Let A, B be closed linear subspaces of a Hilbert space H. Let $B_H := \{x \in H \mid \|x\| \leq 1\}$ denote the unit ball in H. The Friedrichs angle $\alpha(A, B)$ between A and B is the angle in $[0, \frac{\pi}{2}]$ whose cosine is given by*

$$\cos \alpha(A, B) := \sup \left\{ |\langle x, y \rangle| \;\middle|\; x \in A \cap (A \cap B)^\perp \cap B_H \text{ and } y \in B \cap (A \cap B)^\perp \cap B_H \right\}.$$

Deutsch showed in [32, Lemma 9.5] that the Friedrichs angle can also be written in the following way, where $\|\cdot\|_{\text{op}}$ denotes the operator norm.

Lemma 5.8. *For the Friedrichs angle in Definition 5.7, we get*

$$\cos \alpha(A, B) = \left\| P_{B \cap (A \cap B)^\perp} P_{A \cap (A \cap B)^\perp} \right\|_{\text{op}} = \|P_B P_A - P_{A \cap B}\|_{\text{op}}.$$

Proof. We have

$$\begin{aligned}
\cos \alpha(A, B) &= \sup \left\{ |\langle x_1, x_2 \rangle| \,\middle|\, x_1 \in A \cap (A \cap B)^\perp \cap B_H \text{ and } x_2 \in B \cap (A \cap B)^\perp \cap B_H \right\} \\
&= \sup \left\{ |\langle P_{A \cap (A \cap B)^\perp} x_1, P_{B \cap (A \cap B)^\perp} x_2 \rangle| \,\middle|\, \|x_1\| \leq 1,\ \|x_2\| \leq 1 \right\} \\
&= \sup \left\{ |\langle P_{B \cap (A \cap B)^\perp} P_{A \cap (A \cap B)^\perp} x_1, x_2 \rangle| \,\middle|\, \|x_1\| \leq 1,\ \|x_2\| \leq 1 \right\} \\
&= \sup \left\{ \frac{\left\| P_{B \cap (A \cap B)^\perp} P_{A \cap (A \cap B)^\perp}(x) \right\|}{\|x\|} \,\middle|\, \|x\| \neq 0 \right\} \\
&= \left\| P_{B \cap (A \cap B)^\perp} P_{A \cap (A \cap B)^\perp} \right\|_{\text{op}}.
\end{aligned}$$

This proves the first equality. Moreover, it can be shown that

$$\left\| P_{B \cap (A \cap B)^\perp} P_{A \cap (A \cap B)^\perp} \right\|_{\text{op}} = \left\| P_B P_A P_{(A \cap B)^\perp} \right\|_{\text{op}},$$

cf. [32, Lemma 9.5 (7)]. This gives

$$\begin{aligned}
\cos \alpha(A, B) &= \left\| P_B P_A P_{(A \cap B)^\perp} \right\|_{\text{op}} = \|P_B P_A (I - P_{A \cap B})\|_{\text{op}} \\
&= \|P_B P_A - P_B P_A P_{A \cap B}\|_{\text{op}} = \|P_B P_A - P_{A \cap B}\|_{\text{op}},
\end{aligned}$$

showing the second equality. □

Based on the definition of the Friedrichs angle, Kayalaar and Weinert showed the following result, cf. [64, Theorem 2].

Theorem 5.9. *Let A, B be closed linear subspaces of a Hilbert space H, define $c := \cos\alpha(A,B)$ and let $x \in H$. Then*

$$\|(P_B P_A)^n x - P_{A\cap B}x\| = c^{2n-1}\|x\| \quad \text{for any } n \geq 1.$$

This means that for the alternating von Neumann sequence, we have convergence as fast as geometrical progression, provided that $c < 1$, which is true if and only if the Friedrichs angle between A and B is positive.

To determine the convergence rate of the cyclic projections approach for an arbitrary finite number of subspaces $V_1, \ldots, V_k$, we consider the subspaces V_i and $\bigcap_{j=i+1}^{k} V_j$ for every $i \leq k-1$. This gives the following result by Smith, Solomon and Wagner, cf. [87]. This result can also be found in [31, Theorem 6.4].

Theorem 5.10. *Let $V_1, \ldots, V_k$ be closed linear subspaces of a Hilbert space H. Further let $V := \bigcap_{j=1}^{k} V_j$. Then for each $x \in H$, we have*

$$\|(P_{V_k} P_{V_{k-1}} \ldots P_{V_1})^n x - P_V x\| \leq c^n \|x\|,$$

where

$$c = \left(1 - \prod_{i=1}^{k-1} \sin^2\alpha\left(V_i, \bigcap_{j=i+1}^{k} V_j\right)\right)^{\frac{1}{2}}.$$

Remark 5.11. *For this c in Theorem 5.10, we have $c \in [0,1]$. And $c < 1$ if for every $i \in \{1, \ldots, n\}$ the following inequality holds:*

$$\alpha\left(V_i, \bigcap_{j=i+1}^{k} V_j\right) > 0.$$

On the other hand, it is possible to generalize the Friedrichs angle to more than two subspaces, giving the following definition, cf. [4, Definition 2.1]:

Definition 5.12. *Let $V_1, \ldots, V_k$ be closed linear subspaces of a Hilbert space H. Further let $V := \bigcap_{j=1}^{k} V_j$. The Friedrichs number $c(V_1, \ldots, V_k)$ associated to the k subspaces is defined as*

$$c(V_1, \ldots, V_k) := \sup\left\{\frac{2}{k-1}\frac{\sum_{i<j} Re\langle v_i, v_j\rangle}{\|v_1\|^2 + \cdots + \|v_k\|^2} \;\middle|\; v_i \in V_i \cap V^{\perp},\ \|v_1\|^2 + \cdots + \|v_k\|^2 \neq 0\right\}.$$

Here Definition 5.12 and Definition 5.7 coincide for the case $k = 2$, as the following lemma shows.

Lemma 5.13. *Let V_1, V_2 be two closed linear subspaces of a Hilbert space H. Further let $c(V_1, V_2)$ be as defined in Definition 5.12 and let $\cos\alpha(V_1, V_2)$ be as introduced in Definition 5.7. Then $c(V_1, V_2) = \cos\alpha(V_1, V_2)$.*

Proof. Due to Definition 5.12, we have

$$\begin{aligned} c(V_1, V_2) &= \sup\left\{ \frac{2\,\mathrm{Re}\langle v_1, v_2\rangle}{\|v_1\|^2 + \|v_2\|^2} \;\middle|\; v_i \in V_i \cap (V_1 \cap V_2)^\perp,\ \|v_1\|^2 + \|v_2\|^2 \neq 0 \right\} \\ &= \sup\left\{ \frac{2\,\mathrm{Re}\langle v_1, v_2\rangle}{\|v_1\|^2 + \|v_2\|^2} \;\middle|\; v_i \in V_i \cap (V_1 \cap V_2)^\perp,\ \|v_1\|^2 = \|v_2\|^2 = 1 \right\} \\ &= \sup\left\{ \mathrm{Re}\langle v_1, v_2\rangle \;\middle|\; v_i \in V_i \cap (V_1 \cap V_2)^\perp,\ \|v_1\|^2 = \|v_2\|^2 = 1 \right\}. \end{aligned}$$

Now let $\lambda := \frac{\overline{\langle v_1, v_2\rangle}}{|\langle v_1, v_2\rangle|} \in \mathbb{C}$, such that $|\lambda| = 1$ and

$$\langle v_1, \lambda v_2\rangle = \lambda \langle v_1, v_2\rangle = \frac{\overline{\langle v_1, v_2\rangle}}{|\langle v_1, v_2\rangle|} \langle v_1, v_2\rangle = \frac{|\langle v_1, v_2\rangle|^2}{|\langle v_1, v_2\rangle|} = |\langle v_1, v_2\rangle| \in \mathbb{R}.$$

This yields $\mathrm{Re}\langle v_1, \lambda v_2\rangle = \langle v_1, \lambda v_2\rangle = |\langle v_1, v_2\rangle|$ and since V_2 is a subspace, we have $\lambda v_2 \in V_2$ with $\|\lambda v_2\| = \|v_2\|$. Furthermore, for any $x \in (V_1 \cap V_2)$ with $\langle x, v_2\rangle = 0$, we have $\langle x, \lambda v_2\rangle = \lambda\langle x, v_2\rangle = 0$ such that $\lambda v_2 \in (V_1 \cap V_2)^\perp$. Finally, we get with the above reformulation of $c(V_1, V_2)$:

$$\begin{aligned} c(V_1, V_2) &= \sup\left\{ |\langle v_1, v_2\rangle| \;\middle|\; v_i \in V_i \cap (V_1 \cap V_2)^\perp,\ \|v_1\|^2 = \|v_2\|^2 = 1 \right\} \\ &= \sup\left\{ |\langle v_1, v_2\rangle| \;\middle|\; v_i \in V_i \cap (V_1 \cap V_2)^\perp,\ \|v_1\|^2 \leq 1,\ \|v_2\|^2 \leq 1 \right\} \\ &= \cos\alpha(V_1, V_2). \end{aligned}$$

□

Moreover, as mentioned in [4], we always have $c(V_1, \ldots, V_k) \in [0, 1]$. Furthermore, the following result concerning the convergence rate for more than two subspaces holds, cf. [4, Theorem 2.4].

Theorem 5.14. *Let $V_1, \ldots, V_k$ be closed linear subspaces of a Hilbert space H. Further let $V := \bigcap_{j=1}^{k} V_j$. Suppose that $c := c(V_1, \ldots, V_k) < 1$. Then we have*

$$\|(P_{V_k} P_{V_{k-1}} \ldots P_{V_1})^n - P_V\|_{op} \leq \left[1 - \left(\frac{1-c}{4k}\right)^2\right]^{\frac{n}{2}} \quad \textit{for any } n \geq 1.$$

So, in this section, we saw that alternating and cyclic projections between subspaces converge globally in norm and we can give a convergence rate whenever the cosine of the Friedrichs angle or the Friedrichs number is bounded away from one. In the following section, we will have a closer look at alternating projections between general convex sets.

5.2 Alternating Projections on Convex Sets

In this section, we assume A, B to be closed convex sets of a Hilbert space H. We are looking for a point in the intersection $A \cap B$, in case $A \cap B$ is nonempty. Otherwise, we are looking for a pair of points $(a, b) \in A \times B$ attaining the minimal distance, as described in the following definition.

Definition 5.15. *Let A, B be closed sets of Hilbert space H. Then we define the minimal distance between A and B as*

$$d(A, B) := \inf \{\|a - b\| \mid a \in A \text{ and } b \in B\}.$$

If $A \cap B \neq \emptyset$, we have $d(A, B) = 0$.

To obtain a point in the intersection $A \cap B$ (or a pair of points attaining $d(A, B)$), we will use the von Neumann sequences as introduced in Definition 5.2. Considering only two sets first, an early result by Cheney and Goldstein from 1959 should be mentioned, cf. [25, Theorem 4].

Theorem 5.16. *Let A and B be closed convex sets in a Hilbert space H and let $x \in H$ be an arbitrary starting point. Moreover, consider the von Neumann sequence $(b_0, b_1, \dots)$, provided by the operator $P_B(P_A)$, such that $b_0 = x$ and $b_{i+1} = (P_B(P_A))(b_i) = (P_B(P_A))^i(b_0)$ for every $i \geq 1$. Then convergence of $(P_B(P_A))^n$ to a fixed point of $P_B(P_A)$ (that is $(P_B(P_A))(z) = z$) is ensured if one of the following conditions holds:*

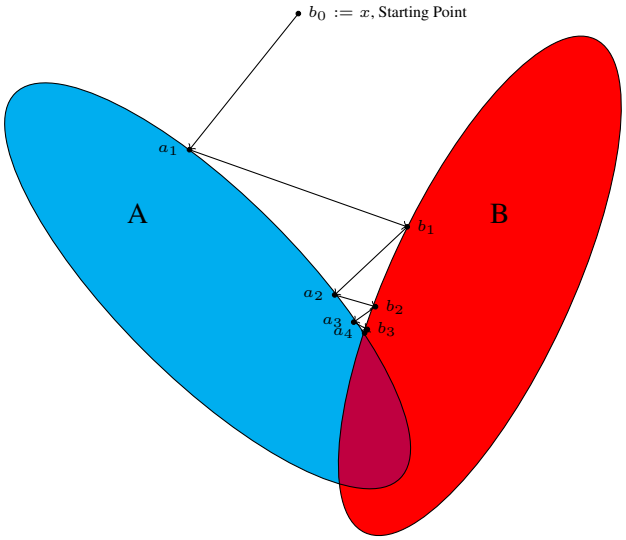
(a) A or B is a compact set.

(b) A or B is a finite dimensional set and $d(A, B)$ is attained.

This theorem provides convergence independent from the starting point for finite dimensional Hilbert spaces H, but does not give any hints concerning the convergence rate. In addition, the result holds even for convex sets which do not intersect. Nevertheless, Theorem 5.16 only shows convergence for alternating projections between exactly two convex sets. Figure 5.2 gives an example for two intersecting convex sets in $\mathbb{R}^2$ and visualizes the results of Theorem 5.16 in a concrete setting.

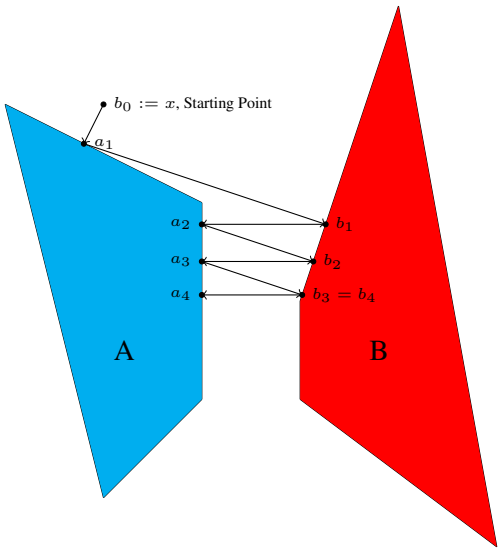
In Figure 5.2, we consider an arbitrarily chosen starting point x. To start the alternating projections approach, we first project x orthogonally onto the set A. The thus generated point is projected orthogonally onto the set B. We continue this procedure until a fixed point of the projection $P_B P_A$ is reached. In this case, the limit z of the alternating projections approach is very close to the point a_4 and fulfills $P_B P_A(z) = z$, such that in the limit we obtained a point in the intersection $A \cap B$.

Figure 5.2: Alternating projections for two convex sets in $\mathbb{R}^2$



Since Theorem 5.16 also holds for convex sets A and B with $A \cap B = \emptyset$, we can as well visualize the convergence result in this setting, again for a concrete example in $\mathbb{R}^2$. Figure 5.3 shows the alternating projections for two polyhedra in $\mathbb{R}^2$.

Figure 5.3: Alternating projections for two nonintersecting convex sets in $\mathbb{R}^2$



Again, we start with an arbitrarily chosen starting point x and first project x orthogonally onto the set A. The thus generated point is then projected onto the set B and we continue as described before. In this concrete example, we have $b_4 = P_B(a_4) = P_B P_A(b_3) = b_3$ and therefore we reached a fixed point of $P_B P_A$ (in this case, in a finite number of steps). This example fulfills the assumptions in Theorem 5.16 and therefore illustrates the convergence of the von Neumann sequences to a fixed point of $P_B P_A$ and thus the results in the theorem for this concrete example.

The points a_4, b_3 not only give a fixed point of the projection, they also attain the minimal

distance between A and B. Hence, we have

$$\|a_4 - b_3\| = d(A, B).$$

This is ensured by the following results, which are based on [7, Section 2 and 4]. The key aspect can also be found in [6, Fact 1.2].

Lemma 5.17. *Let H be a Hilbert space and consider closed convex subsets A, B of H and a starting point $x \in H$. In addition, let $E := \{a \in A \mid \|a - \bar{b}\| = d(A, B)$ for some $\bar{b} \in B\}$ and $F := \{b \in B \mid \|\bar{a} - b\| = d(A, B)$ for some $\bar{a} \in A\}$, the set of points in A (resp. B) that are nearest to B (resp. A). Then:*

(a) $E = \mathrm{Fix}(P_A P_B)$ (resp. $F = \mathrm{Fix}(P_B P_A)$), the set of fixed points of the Projection $P_A P_B$ (resp. $P_B P_A$).

(b) The value $d(A, B)$ is attained if and only if E, F are nonempty.

(c) Consider the displacement vector $v = P_{\mathrm{cl}(B-A)}(0)$ and let E and F be nonempty. Then $\|v\| = d(A, b)$. Moreover, we have $E + v = F$ and $F = (A + v) \cap B$.

Having this, we can give the main result in this setting, cf. [6, Fact 1.2] and [7, Theorem 4.8].

Theorem 5.18. *Let H be a Hilbert space and consider closed convex subsets A, B of H and a starting point $x \in H$. Assume that $d(A, B)$ is attained and consider the von Neumann sequences as introduced in Definition 5.2. Then we get:*

(a) Let $v := P_{\mathrm{cl}(B-A)}(0)$ be the displacement vector introduced in Lemma 5.17. If A or B is compact, then we get

$$\lim_{n\to\infty} \|a_n - e^*\| = 0 \text{ and } \lim_{n\to\infty} \|b_n - (e^* + v)\| = 0$$

for some $e^* \in E$. Moreover, we have $f^* := e^* + v \in F$, again due to Lemma 5.17.

(b) Without a compactness assumption, only weak convergence can be ensured.

This theorem now shows that the von Neumann sequences in Figure 5.3 converge to a pair of points ($e^* = b_3$ and $f^* = a_4$) attaining $d(A, B)$. In addition, we have the following corollary.

Corollary 5.19. *If $A \cap B \neq \emptyset$, then $E = F = A \cap B$.*

Proof. If $A \cap B \neq \emptyset$, then the value $d(A, B)$ is attained for every point in $A \cap B$. Hence, by definition, we have $E = A \cap B = F$. □

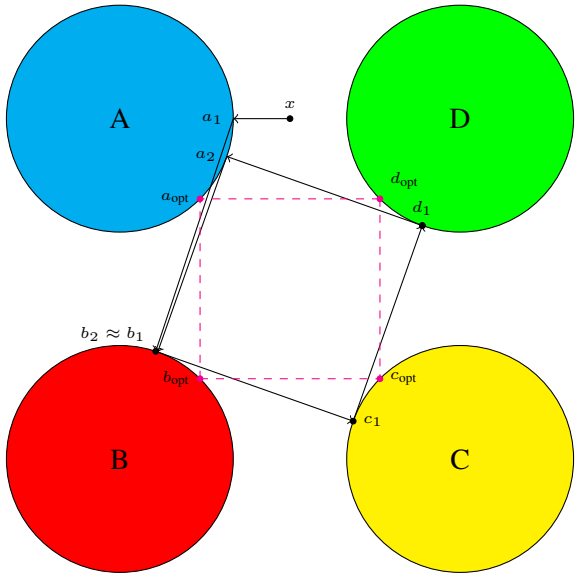
Thus, Figure 5.2 can also be seen as an illustration of the convergence result in Theorem 5.18.

Like in Section 5.1, the question arises whether it is possible to apply this method to more than two convex sets. For this, it is necessary to project in a certain cyclic order, similar to the approach mentioned in 5.6, but now for a sequence of convex sets.

5.2.1 Cyclic Projections Among a Sequence of Convex Sets

Before we will analyse the cyclic projections approach for a sequence of convex sets, we will see that it becomes necessary to assume that the convex sets have nonempty intersection. To this end, we consider the following example in $\mathbb{R}^2$.

Figure 5.4: Cyclic projections approach for four noninterseting convex sets in $\mathbb{R}^2$



For the cyclic projections approach between convex sets, we will use the cyclic projections method introduced in Remark 5.6, where the subspaces V_i are simply replaced by the closed convex sets A_i.

In Figure 5.4, the cyclic projections approach is visualized for four convex sets, where the order of projection is alphabetical. More precisely, we have

$$A = B_1([1,4]),\ B = B_1([1,1]),\ C = B_1([4,1]) \text{ and } D = B_1([4,4]),$$

where $B_r(x)$ denotes the closed ball in $\mathbb{R}^2$ of radius r with center equal to the vector $x \in \mathbb{R}^2$. The cyclic projections approach begins with the starting point x and its orthogonal projection onto the closed convex set A. This generates the point a_1, which is then projected orthogonally onto the set B, returning b_1. Next, we project b_1 onto the set C, keeping the alphabetical order. This generates c_1, which is projected onto D as the last projection of the first cycle. The second cycle starts with the projection of d_1 onto A again. This returns the point a_2. If we now project a_2 orthogonally onto B, we get the point b_2, which is very close to our first iterate b_1 in B. From now on, every iterate in each set will be very close to the previous iterate in this set such that in the limit we reach a fixed point of the cycle $P_D P_C P_B P_A$.

Considering only two closed convex sets, Theorem 5.18 shows that under the given assumptions, the von Neumann sequences converge to a pair of points attaining the minimal distance between the two convex sets. As Figure 5.4 shows, this can not be generalized to more than two sets.

More precisely, considering the points a_2, b_1, c_1, d_1 and the convex hull $\operatorname{conv}\{a_2, b_1, c_1, d_1\}$, it can be seen that there exists a square of smaller volume, where every vertex is an element of one of the sets. This square is illustrated as the convex hull $\operatorname{conv}\{a_{\text{opt}}, b_{\text{opt}}, c_{\text{opt}}, d_{\text{opt}}\}$. Moreover, the points $a_{\text{opt}}, b_{\text{opt}}, c_{\text{opt}}, d_{\text{opt}}$ can never be a limit of a cyclic projections approach since for example $c_{\text{opt}} \neq P_C(b_{\text{opt}})$. Therefore, the points a_2, b_1, c_1, d_1, which are very close to the limits of the cyclic projections approach in each set, do not attain the minimal distance among the sets A, B, C, D in any reasonable sense. For example, if we consider $(c_1, d_1) \in C \times D$ and $(c_{\text{opt}}, d_{\text{opt}}) \in C \times D$, we have $\|c_{\text{opt}} - d_{\text{opt}}\| < \|c_1 - d_1\|$. This also holds for any other pairwise distance between two elements of $\{a_2, b_1, c_1, d_1\}$ or $\{a_{\text{opt}}, b_{\text{opt}}, c_{\text{opt}}, d_{\text{opt}}\}$.

Motivated by this counterexample, we will from now on assume that $\bigcap_{i=1}^{k} A_i \neq \emptyset$. Under this assumption, it is possible to derive a convergence result for the cyclic projections method on a sequence of convex sets, as shown in the following result, see for example [26, Theorem 3.2].

Theorem 5.20. *Let $A_1, \ldots, A_k$ be an ordered sequence of closed and convex subsets of a Hilbert space H. Further let $\bigcap_{i=1}^{k} A_i \neq \emptyset$. Then for every starting point $x \in H$, we consider the cyclic projections method as shown in Remark 5.6. Then the following holds:*

(a) The cyclic projections method converges weakly to a point $a \in \bigcap_{i=1}^{k} A_i$.

(b) The cyclic projections method converges in norm to a point $a \in \bigcap_{i=1}^{k} A_i$ if one of the sets A_i is boundedly compact. (This means that the intersection of A_i and an arbitrary closed ball is compact).

Here it should be mentioned that a closed convex set is boundedly compact if and only if it is locally compact, as for example mentioned in [65]. Since every compact set is also locally compact, we have the following corollary.

Corollary 5.21. *Under the same assumptions as in Theorem 5.20, the cyclic projections method converges in norm to a point $a \in \bigcap_{i=1}^{k} A_i$ if one of the sets A_i is compact.*

For all these results, the convergence is again independent from the starting point and therefore global convergence is ensured.

Additional sufficient conditions for convergence in norm of the iterates of the cyclic projections approach are summed up in the following theorem, based on the results by Gubin et al. in [51, Theorem 1].

Theorem 5.22. *Let $A_1, \ldots, A_k$ be an ordered sequence of closed and convex subsets of a Hilbert space H. Further let $\bigcap_{i=1}^{k} A_i \neq \emptyset$ and let any of the following conditions be satisfied:*

(a) $A_j \cap \operatorname{int}\left(\bigcap_{\substack{i=1 \\ i \neq j}}^{k} A_i\right) \neq \emptyset$ for every $j \in \{1, \ldots, k\}$.

(b) H is finite dimensional.

(c) The sets A_i are all halfspaces, that is $A_i = \{x \in H \mid \langle c_i, x \rangle \leq \beta_i\}$ for every $i \in \{1, \ldots, k\}$.

Then for any starting point $x \in H$, the cyclic projections method converges in norm to a point $a \in \bigcap_{i=1}^{k} A_i$.

Here part (b) gives a generalization of Theorem 5.16 and part (c) can be seen as a generalization of the result in Theorem 5.5 to halfspaces.

So far, we have a certificate for convergence in norm or weak convergence, but we do not have a result on the convergence rate. Similar to the results in Section 5.1, we will introduce a certain angle between convex sets, giving an explicit convergence rate for alternating or cyclic projections on intersecting convex sets.

5.2.2 Alternating Projections and the Angle Between Convex Sets

We start by defining the concept for two convex sets. After that, we consider the case of more than two sets. For the case of two intersecting convex sets in a Hilbert space H, we will analyse a certain angle between these two sets. The definiton of the angle and further definitions in this section are based on the reults by Deutsch and Hundal in [33]. As in [33], we assume here that H is a real Hilbert space.

First, we introduce the ε-polar cone as defined in [33, Definition 3.1].

Definition 5.23. *Let A be a closed convex set A in a Hilbert space H, and let $\varepsilon > 0$. Then the ε-polar cone of A is defined as*

$$A^{\circ,\varepsilon} := \operatorname{cone}\{x - P_A(x) \mid x \in B_\varepsilon(0)\},$$

where $B_\varepsilon(0)$ denotes the closed ball of radius ε with center equal to $0 \in H$.

The set $A^{\circ,\varepsilon}$ is a convex cone and generalizes the so called polar cone, which is defined as

$$A^\circ := \{x \in H \mid \langle x, y\rangle \leq 0 \text{ for all } y \in A\}.$$

For a convex cone K and with equation (1), we therefore have

$$K^\circ = -K^*,$$

where K^* denotes the dual cone of K. Taking an arbitrary $y \in H$, the ε-polar cone can be generalized in the following way, cf. [33, Lemma 3.2]:

$$(A-y)^{\circ,\varepsilon} = \operatorname{cone}\{x - P_A(x) \mid x \in B_\varepsilon(y)\}. \tag{34}$$

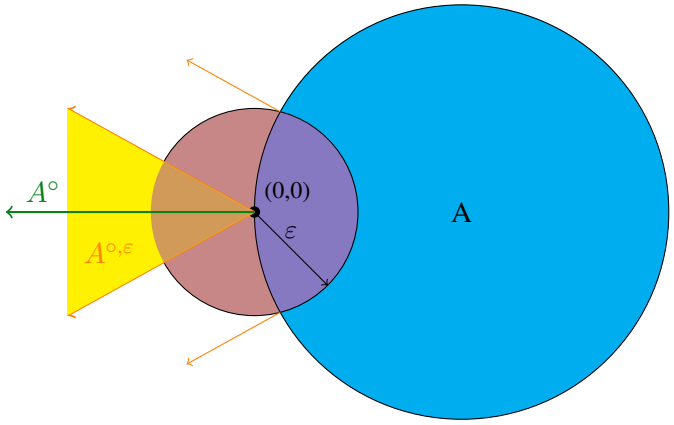
Further, if $y \in \operatorname{int}(A)$, we have for $\varepsilon > 0$ sufficiently small, that

$$(A-y)^{\circ,\varepsilon} = \{0\} = (A-y)^\circ.$$

The property $(A-y)^{\circ,\varepsilon} \supseteq (A-y)^\circ$ holds for any $y \in A$ (cf. [33, Theorem 3.3]), but in general we have $(A-y)^{\circ,\varepsilon} \neq (A-y)^\circ$, as Figure 5.5 shows.

In this figure, we consider a two dimensional example, where $A = B_1([1,0])$ is the blue set and $y = [0,0]$ is the origin. Then the polar cone A° is the green ray starting in the origin and

Figure 5.5: The polar cone and ε-polar cone are different in general



pointing into direction $[-1, 0]$ since for any $x = \lambda \cdot [-1, 0]$, with $\lambda \geq 0$, and any $y \in A$, we have $\langle x, y\rangle \leq 0$.

On the other hand, to determine the ε-polar cone $A^{\circ,\varepsilon}$, we consider a ball around the origin with radius equal to ε. Now we consider all points $x \in B_\varepsilon([0, 0])$ and directions $(x - P_A(x))$ as any possible vector in $A^{\circ,\varepsilon}$, where $P_A(x)$ is the projection of x onto the set A. The extreme rays of these directions are located at the intersection of the boundaries of $B_1([1, 0])$ and $B_\varepsilon([0, 0])$ and they are marked in orange. Any conic combination of these two extreme vectors can be attained by a vector $(x - P_A(x))$ and therefore the conic hull of these vectors is the ε-polar cone. The set $A^{\circ,\varepsilon}$ itself is marked in yellow and contains all possible rays in between the extremal directions.

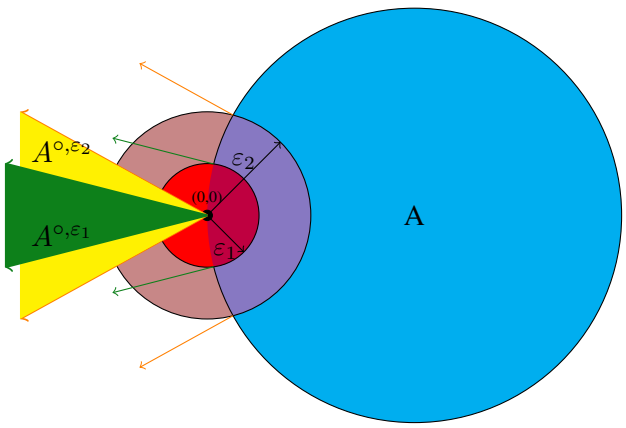
Clearly, A° is one of these rays such that $A^\circ \subseteq A^{\circ,\varepsilon}$, but any other ray in $A^{\circ,\varepsilon}$ is not an element of A°, proving $(A - y)^{\circ,\varepsilon} \neq (A - y)^\circ$ in general.

In addition, if we consider various values for $\varepsilon > 0$, the following monotonicity holds, cf. [33, Lemma 3.2 (i)]:

$$A^{\circ,\varepsilon_1} \subseteq A^{\circ,\varepsilon_2} \text{ for any } \varepsilon_2 \geq \varepsilon_1 > 0.$$

This result is illustrated in Figure 5.6, again for the two dimensional example in Figure 5.5.

Figure 5.6: ε-polar cones for different values of ε



Here the smaller red ball with radius ε_1 corresponds to the smaller cone, marked in green. On the other hand, the greater value ε_2 corresponds to the larger cone, which is marked in yellow.

Nevertheless, in some cases the ε-polar cone and the polar cone are equal, as the following lemma shows, cf. [33, Corollaries 3.4 and 3.5].

Lemma 5.24. *Let $A_i := \{x \in H \mid \langle x, a_i\rangle \leq \alpha_i\}$ be a halfspace with $a_i \in H \setminus \{0\}$ and $\alpha_i \in \mathbb{R}$ for every $i \in \{i, \dots, k\}$. For the polyhedron $P = \bigcap_{i=1}^k A_i$ and $y \in P$, let $\varepsilon(y) > 0$ be sufficiently small. Then we have*

$$(P - y)^{\circ,\varepsilon} = \operatorname{cone}\{a_i \mid i \in I(y)\} = (P - y)^\circ,$$

where $I(y) := \{i \in \{1, \dots, k\} \mid \langle y, a_i\rangle = \alpha_1\}$ denotes the set of active indices for y relative to P.

In addition, let K be a closed convex cone, A a closed affine set, $\varepsilon > 0$ and $y \in A$. Then the following holds:

$$K^{\circ,\varepsilon} = K^\circ \quad \text{and} \quad (A - y)^{\circ,\varepsilon} = (A - y)^\circ.$$

We will now use the definition of an ε-polar cone to define the angle between convex sets, cf. [33, Definition 4.1].

Definition 5.25. *Let $A_1,\ A_2$ be closed convex sets with $0 \in A_1 \cap A_2$. Further let $\varepsilon \geq 0$. Then for $i \in \{1, 2\}$, the i-th ε-angle between the ordered pair A_1 and A_2 is the angle in the interval $[0, \frac{\pi}{2}]$ whose cosine $c_i(A_1, A_2; \varepsilon)$ is given by*

$$c_i(A_1, A_2; \varepsilon) := \sup\left\{ \frac{\left\| P_{A_2 \cap \overline{(A_1^{\circ,\varepsilon} + A_2^{\circ,\varepsilon})}} P_{A_1 \cap \overline{(A_1^{\circ,\varepsilon} + A_2^{\circ,\varepsilon})}}(x) \right\|}{\|x\|} \;\middle|\; x \in A_i \cap \overline{(A_1^{\circ,\varepsilon} + A_2^{\circ,\varepsilon})},\ \|x\| = \varepsilon \right\}.$$

If $\varepsilon = 0$ or $A_i \cap \overline{(A_1^{\circ,\varepsilon} + A_2^{\circ,\varepsilon})} \cap B_\varepsilon(0) = \emptyset$, we define $c_i(A_1, A_2; \varepsilon) = 0$.

If A_1 and A_2 are closed linear subspaces, Definition 5.25 and Definition 5.7 coincide, as the following lemma shows, cf. [33, Lemma 4.2]. In particular, for closed linear subspaces, the ε-angles are independent of ε.

Lemma 5.26. *Let A_1 and A_2 be closed linear subspaces of a Hilbert space H. Further let $\varepsilon > 0$. Then $c_1(A_1, A_2; \varepsilon) = \cos \alpha(A_1, A_2)$.*

Proof. For subspaces A_1, A_2 and $\varepsilon > 0$, we have

$$(A_1 \cap A_2)^\perp = \overline{A_1^\perp + A_2^\perp} = \overline{A_1^\circ + A_2^\circ} = \overline{A_1^{\circ,\varepsilon} + A_2^{\circ,\varepsilon}},$$

where the last equation follows by Lemma 5.24. Now let B_H denote the unit ball in H. Then

Lemma 5.8 yields:

$$\begin{aligned}
&\cos\alpha(A_1, A_2)\\
&= \sup\left\{ \frac{\left\| P_{A_2\cap(A_1\cap A_2)^\perp} P_{A_1\cap(A_1\cap A_2)^\perp}(x)\right\|}{\|x\|} \;\middle|\; \|x\| = 1 \right\}\\
&= \sup\left\{ \frac{\left\| P_{A_2\cap(A_1\cap A_2)^\perp} P_{A_1\cap(A_1\cap A_2)^\perp}(x)\right\|}{\|x\|} \;\middle|\; \|x\| = \varepsilon \right\}\\
&= \sup\left\{ \frac{\left\| P_{A_2\cap(A_1\cap A_2)^\perp} P_{A_1\cap(A_1\cap A_2)^\perp}(x)\right\|}{\|x\|} \;\middle|\; x \in A_1\cap(A_1\cap A_2)^\perp,\ \|x\| = \varepsilon \right\}\\
&= \sup\left\{ \frac{\left\| P_{A_2\cap\overline{(A_1^{\circ,\varepsilon}+A_2^{\circ,\varepsilon})}} P_{A_1\cap\overline{(A_1^{\circ,\varepsilon}+A_2^{\circ,\varepsilon})}}(x)\right\|}{\|x\|} \;\middle|\; x \in A_1\cap\overline{(A_1^{\circ,\varepsilon}+A_2^{\circ,\varepsilon})},\ \|x\| = \varepsilon \right\}\\
&= c_1(A_1, A_2; \varepsilon).
\end{aligned}$$

□

Similar to the Friedrichs angle, it is possible to generalize the ε-angle to more than two closed convex sets. We therefore consider the following definition, cf. [33, Definition 4.3].

Definition 5.27. *Let $A_1, A_2, \ldots, A_k$ be closed convex sets with $0 \in \bigcap_{i=1}^k A_i$ and let $\varepsilon \geq 0$. For $i = 1$ or $i = k$, the i-th ε-angle of the ordered collection $A_1, A_2, \ldots, A_k$ is the angle in $[0, \frac{\pi}{2}]$ whose cosine is defined as*

$$c_i(A_1, \ldots, A_k; \varepsilon) := \sup\left\{ \frac{\left\| P_{A_k\cap A^\varepsilon} P_{A_{k-1}\cap A^\varepsilon} \ldots P_{A_1\cap A^\varepsilon}(x)\right\|}{\|x\|} \;\middle|\; x \in A_i \cap A^\varepsilon \cap B_\varepsilon(0) \right\},$$

where $A^\varepsilon := \overline{\sum_{i=1}^k A_i^{\circ,\varepsilon}}$. If $\varepsilon = 0$ or $A_i \cap A^\varepsilon \cap B_\varepsilon(0) = \emptyset$, we define $c_i(A_1, A_2, \ldots, A_k; \varepsilon) = 0$.

Remark 5.28. *The assumptions $0 \in A_1 \cap A_2$ in Definition 5.25 or $0 \in \bigcap_{i=1}^k A_i$ in Definition 5.27 are equivalent to the assumptions $A_1 \cap A_2 \neq \emptyset$ or $\bigcap_{i=1}^k A_i \neq \emptyset$: If 0 is no element of the intersection we take $y \in \bigcap_{i=1}^k A_i$ and consider $A_i - y$ instead of A_i for every i.*

Now we can give a convergence rate for cyclic (and alternating) projections approach on closed convex subsets. Here the reader is referred to [33, Theorem 4.6].

Theorem 5.29. *Let $A_1, A_2, \ldots, A_k$ be closed convex subsets of a Hilbert space H such that $\bigcap_{i=1}^k A_i \neq \emptyset$, and let $x \in H$. Further consider the cyclic projections approach as introduced in*

Remark 5.6 and let $a \in \bigcap_{i=1}^{k} A_i$ be its (weak) limit, given by Theorem 5.20. Then for every $n \geq 1$:

$$\begin{aligned}\|(P_{A_k}P_{A_{k-1}}\dots P_{A_1})^n(x) - a\| &\leq c_{k,n-1}\|(P_{A_k}P_{A_{k-1}}\dots P_{A_1})^{n-1}(x) - a\| \\ &\leq c_{1,1}\left(\prod_{j=1}^{n-1} c_{k,j}\right)\|x - a\|,\end{aligned}$$

where

$$c_{1,1} = c_1(A_1 - a, A_2 - a, \dots, A_k - a; \|P_{A_1}(x) - a\|),$$

and for $j = 1, \dots, n-1$, we define

$$c_{k,j} := c_k(A_1 - a, A_2 - a, \dots, A_k - a; \|x_k^j - a\|) \text{ with } x_k^j = (P_{A_k}P_{A_{k-1}}\dots P_{A_1})^j(x).$$

This theorem gives an explicit convergence rate depending on the ε-angle between the convex sets. To sum up the results in Theorem 5.14 and Theorem 5.29, we have a linear convergence rate for cyclic or alternating projections onto subspaces or onto closed convex sets, depending on the generalized angle between the sets. For the generalized angle for subspaces, we used the Friedrichs-number in Definition 5.12, and for closed convex sets, we used the ε-angle as defined in Definition 5.27. So far, a global convergence result could be shown for cyclic projections onto subspaces, as well as for cyclic projections onto closed convex sets. Hitherto, all sets we considered were convex subsets of a Hilbert space H. In the following section, we will have a closer look at manifolds, which are not convex in general. It will be shown that alternating projections can also be applied to these nonconvex sets, albeit we will lose the global convergence in general.

5.3 Alternating Projections on Manifolds

In this section, we will show that the alternating projections method can also be applied to nonconvex sets. More precisely, we will see a convergence result for alternating projections between transversally intersecting manifolds. The definitions and propositions shown here are based on the results by Lewis and Malick, which can be found in [70]. First, we will give a short definition of a smooth manifold, cf. [83, Section 6.C].

Definition 5.30. *We say that a subset M of a Euclidean space $\mathbb{E}$ is a C^k manifold of codimension d around $\bar{x} \in M$ if there exists an open set $U \subseteq \mathbb{E}$ with $\bar{x} \in U$ such that the following equation holds:*

$$M \cap U = \{x \in U \mid F(x) = 0\},$$

where $F : U \to \mathbb{R}^d$ is a C^k function with surjective derivative throughout U. Here k denotes the degree of smoothness of the manifold M or the function F.

Since sets that are defined this way are not convex any more, we are looking for a best approximation instead of a unique projection in general. Let M be a smooth manifold in a Euclidean

space $\mathbb{E}$. Motivated by Definition 5.1, we denote the best approximation to $x \in \mathbb{E}$ from M by

$$P_M(x) := \operatorname{argmin}\{\|x - y\| \mid y \in M\}. \tag{35}$$

Throughout this section, we will consider the tangent space and its orthogonal complement, which are defined as follows:

Definition 5.31. *Let M be a C^k-manifold around a point $x \in M$. The tangent space to M at x is then defined as*

$$T_M(x) := \ker(\nabla F(x)),$$

where $\ker(f(x))$ denotes the kernel of a function f. And the normal space to M at x is defined as

$$N_M(x) := T_M(x)^{\perp}.$$

For the results in this section, it is necessary to concretise the intersection of two manifolds in $\mathbb{E}$. We therefore introduce the concept of transversality as in [70, Definition 2.1].

Definition 5.32. *Let $M, N \subseteq \mathbb{E}$ be C^k-manifolds around a point $x \in M \cap N$. Then M and N intersect transversally in x if*

$$T_M(x) + T_N(x) = \mathbb{E},$$

where $T_M(x)$ is the tangent space to M at x from Definition 5.31 and the sum of the tangent spaces is the Minkowski sum. We say that two manifolds are transverse if they intersect transversally.

For transverse manifolds, we know that the intersection is again a manifold and in any point of the intersection, the tangent space to the intersection is explicitly given, as the following lemma shows.

Lemma 5.33. *Let $M, N \subseteq \mathbb{E}$ be transverse C^k-manifolds. Then:*

(a) $M \cap N$ itself is a C^k manifold with codimension $\operatorname{codim}(M \cap N) = \operatorname{codim}(M) + \operatorname{codim}(N)$.

(b) For any point $x \in M \cap N$, we have $T_{M\cap N}(x) = T_M(x) \cap T_N(x)$.

Proof. (a) This result can be found in [52, page 30].

(b) First, note that since $M \cap N \subseteq M$ and $M \cap N \subseteq N$, we have for any $x \in M \cap N$ that

$$T_{M\cap N}(x) \subseteq T_M(x) \cap T_N(x).$$

Moreover, part (a) yields

$$\dim(M \cap N) = \dim(M) + \dim(N) - \dim(\mathbb{E}).$$

Let $x \in M \cap N$, then this gives

$$\dim(T_{M\cap N}(x)) = \dim(T_M(x)) + \dim(T_N(x)) - \dim(\mathbb{E}).$$

On the other hand, since M and N are transverse, we know that

$$\begin{aligned}\dim(T_M(x) \cap T_N(x)) &= \dim(T_M(x)) + \dim(T_N(x)) - \dim(T_M(x) + T_N(x)) \\ &= \dim(T_M(x)) + \dim(T_N(x)) - \dim(\mathbb{E}).\end{aligned}$$

Altogether, $T_{M\cap N}(x) \subseteq T_M(x) \cap T_N(x)$ and both vectorspaces are of the same dimension, hence they must be equal. This completes the proof.

□

Figure 5.7: Transversality for manifolds. An example in $\mathbb{R}^2$

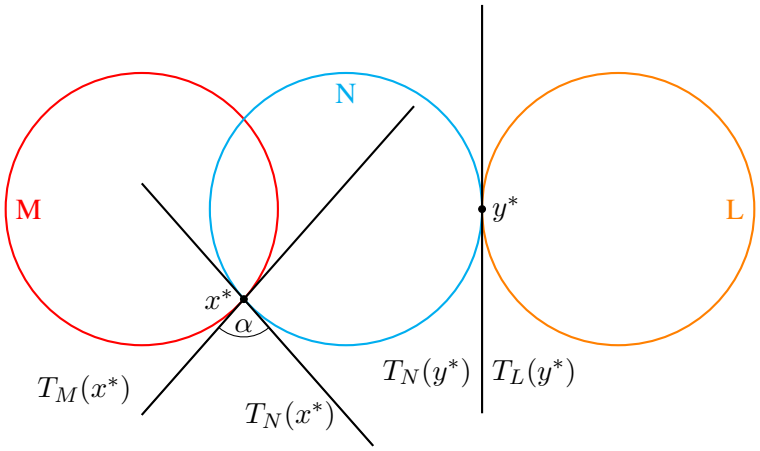


Figure 5.7 shows the difference between transverse manifolds and nontransverse manifolds around an intersection point in $\mathbb{R}^2$. In this concrete example, we have $M = \mathrm{bd}(B_1([-1.5, 0])$, $N = \mathrm{bd}(B_1([0, 0]))$ and $L = \mathrm{bd}(B_1([2, 0]))$. The point x^* represents a point in the intersection of the transverse manifolds M and N. To see that these two manifolds are transverse, consider the tangent space to M at x^* and the tangent space to N at x^*. They only have one point in common and every $x \in \mathbb{R}^2$ can be represented as a sum $x_{T_N(x^*)} + x_{T_M(x^*)}$ with $x_{T_M(x^*)} \in T_M(x^*)$ and $x_{T_N(x^*)} \in T_N(x^*)$. On the other hand, the point y^* represents the intersection of the manifolds N and L. Here, the manifolds do not intersect transversally since the tangent space to N at y^* and the tangent space to L at y^* are equal, such that for example x^* can not be expressed as a sum of elements in this linear subspaces.

In order to apply alternating projections on manifolds, we need to guarantee the existence of $P_M(x)$ in equation (35) for any x which is close enough to M. Here the following lemma holds, cf. [70, Lemma 2.1].

Lemma 5.34. *Let $M \subseteq \mathbb{E}$ be a C^k-manifold around a point $x \in M$ with $k \geq 2$. Then for any point $\bar{x} \in B_\varepsilon(x)$, with $\varepsilon > 0$ small enough, the projection operator P_M is well defined and there exists a unique projection $P_M(\bar{x})$ on M. In addition, the function P_M is of class C^{k-1} around x with derivative*

$$\nabla P_M(x) = P_{T_M(x)}.$$

We therefore have a unique best approximation on M of any point $\bar{x}$ close enough to M. On the other hand, we will extend the definition of a Friedrichs angle from Definition 5.7 to the context of manifolds, again to obtain a convergence rate, this time for the alternating projections between manifolds. For the following definition, see also [70, Definition 3.1].

Definition 5.35. *Consider two manifolds M and N in the Euclidean space $\mathbb{E}$ around a point $x \in M \cap N$. The angle between M and N at x is the angle in $[0, \frac{\pi}{2}]$ whose cosine $c(M, N; x)$ is given by*

$$c(M, N; x) := \cos \alpha \left(T_M(x), T_N(x) \right),$$

where $\alpha \left(T_M(x), T_N(x) \right)$ is the Friedrichs angle, introduced in Definition 5.7, since $T_M(x)$ and $T_N(x)$ are subspaces. This value is well defined unless one tangent space is a subspace of the other. In this case, we define $c(M, N; x) = 0$.

Moreover, we also have the following property, as mentioned in [70]:

Lemma 5.36. *Let A, B be subspaces of a Euclidean space $\mathbb{E}$ and let $B_{\mathbb{E}} := \{x \in \mathbb{E} \mid \|x\| \leq 1\}$ denote the unit ball in $\mathbb{E}$. Then the supremum*

$$\cos \alpha(A, B) = \sup \left\{ |\langle x, y \rangle| \;\middle|\; x \in A \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}} \text{ and } y \in B \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}} \right\}$$

is attained and we always have $\cos \alpha(A, B) < 1$.

Proof. The compactness of $A \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}}$ and $B \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}}$ ensures that the supremum is attained. Moreover, we will prove the inequality $\cos \alpha(A, B) < 1$ by contradiction. To this end, assume that $\cos \alpha(A, B) = 1$, i.e.,

$$\max \left\{ |\langle x, y \rangle| \;\middle|\; x \in A \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}} \text{ and } y \in B \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}} \right\} = 1.$$

Thus, there exists $x \in A \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}}$ and $y \in B \cap (A \cap B)^{\perp} \cap B_{\mathbb{E}}$ such that by the Cauchy-Schwarz-Inequality we have

$$1 = |\langle x, y \rangle| \leq \|x\| \, \|y\| \leq 1.$$

This shows that $\alpha x = y$ for some scalar value α. Thus, we have $y \in A \cap B$ since A is a subspace. Moreover, we have $y \in (A \cap B)^{\perp}$ by construction. Hence, $y = 0$ and we get $|\langle x, y \rangle| = 0$, which is a contradiction. □

The angle between manifolds is also visualized in Figure 5.7. For the transverse manifolds M and N and $x^* \in M \cap N$, the angle is marked as α between the tangent spaces $T_M(x^*)$ and $T_N(x^*)$. Considering the manifolds L and N and the point $y^* \in N \cap L$, the tangent space $T_N(y^*)$ is equal to the tangent space $T_L(y^*)$ and therefore $c(N, L; y^*) = 0$ by definition.

Remark 5.37. *The angle between manifolds in Definition 5.35 depends on x in general. If M and N are subspaces, the angle becomes independent from x and this yields*

$$c(M, N; x) = \cos \alpha(M, N) \quad \textit{for every } x \in M \cap N.$$

In this case, Definition 5.7 and Definition 5.35 coincide.

Since $T_M(x)$ and $T_N(x)$ are subspaces, we also have the following characterization of the angle between manifolds, where $\|\cdot\|_{\text{op}}$ again denotes the operator norm.

Lemma 5.38. *Let M and N be two manifolds in the Euclidean space $\mathbb{E}$ around a point $x \in M \cap N$. Then we have*

$$c(M, N; x) = \left\| P_{T_M(x)} P_{T_N(x)} - P_{T_M(x) \cap T_N(x)} \right\|_{\text{op}}.$$

If we additionally assume that the manifolds M and N intersect transversally in x, we further have

$$c(M, N; x) = \left\| P_{T_M(x)} P_{T_N(x)} - P_{T_{M \cap N}(x)} \right\|_{\text{op}}.$$

Proof. The first statement follows from Lemma 5.8 and the second part follows from Lemma 5.33. □

In addition, the angle, seen as a function in x, is smooth according to the following lemma, cf. [70, Lemma 3.4].

Lemma 5.39. *Let M and N be two transverse C^k manifolds around a point $\overline{x} \in M \cap N$ with $k \geq 2$. Then the function*

$$c(M, N; \cdot) : \ M \cap N \to \ [0, 1], \ x \mapsto c(M, N; x)$$

is of class C^{k-1} around $\overline{x}$.

With the definition of an angle between two manifolds at a common point, we can now give the following asymptotical improvement of the iterates of the alternating projections approach in Definition 5.2 for manifolds. This result and its proof can be found in [70, Theorem 4.2].

Theorem 5.40. *Let M and N be two transverse C^2 manifolds around a point $\overline{x} \in M \cap N$. Then we have:*

$$\limsup_{x \to \overline{x}, x \notin M \cap N} \frac{\|P_M P_N(x) - P_{M \cap N}(x)\|}{\|x - P_{M \cap N}(x)\|} \leq c(M, N; \overline{x}).$$

Proof. Due to Lemma 5.34, there exists $\varepsilon > 0$ such that the operators P_M, P_N and $P_{M \cap N}$ are well defined and of class C^1 in the neighbourhood $B_\varepsilon(\overline{x})$. To make sure that the fraction in the result is well defined, we need to show that $P_M P_N$ is also well defined. So consider $x \in B_{\frac{\varepsilon}{2}}(\overline{x})$ such that

$$\|\overline{x} - P_N(x)\| = \|\overline{x} - x + x - P_N(x)\| \leq \|\overline{x} - x\| + \|x - P_N(x)\| \leq 2\|x - \overline{x}\| \leq \varepsilon,$$

which shows $P_N(x) \in B_\varepsilon(\overline{x})$. Thus, $P_M P_N$ is also well defined and of class C^1 on $B_{\frac{\varepsilon}{2}}(\overline{x})$. Hence, the fraction is well defined.

Now consider an arbitrary sequence $(x_r)_{r \in \mathbb{N}}$ in $B_{\frac{\varepsilon}{2}}(\overline{x}) \setminus (M \cap N)$ tending to $\overline{x}$. We will write $\overline{x}_r := P_{M \cap N}(x_r)$ to shorten notation. Thus, we have

$$P_M P_N(x_r) - \overline{x}_r = P_M P_N(x_r) - P_M P_N(\overline{x}_r). \tag{36}$$

Since $M \cap N$ is again a C^2 manifold (cf. Lemma 5.33), Lemma 5.34 shows that the operator $P_{M\cap N}$ is continuous and it follows

$$\lim_{r\to\infty} \overline{x}_r = \overline{x},$$

such that equation (36) can be written in the following way, using continuous differentiability.

$$P_M P_N(x_r) - \overline{x}_r = \nabla(P_M P_N)(\overline{x}_r)(x_r - \overline{x}_r) + o\left(\|x_r - \overline{x}_r\|\right). \tag{37}$$

By the chain rule and Lemma 5.34, we have

$$\nabla(P_M P_N)(\overline{x}_r) = P_{T_M(\overline{x}_r)} P_{T_N(\overline{x}_r)}. \tag{38}$$

According to the transversality assumption and since $(x_r - \overline{x}_r) \in N_{M\cap N}(\overline{x}_r) = T_{M\cap N}(\overline{x}_r)^\perp$, we get

$$P_{T_M(\overline{x}_r)\cap T_N(\overline{x}_r)}(x_r - \overline{x}_r) = P_{T_{M\cap N}(\overline{x}_r)}(x_r - \overline{x}_r) = 0.$$

Thus, we can write

$$P_{T_M(\overline{x}_r)} P_{T_N(\overline{x}_r)}(x_r - \overline{x}_r) = (P_{T_M(\overline{x}_r)} P_{T_N(\overline{x}_r)} - P_{T_M(\overline{x}_r)\cap T_N(\overline{x}_r)})(x_r - \overline{x}_r). \tag{39}$$

Combining equations (37), (38) and (39) gives

$$\frac{\|P_M P_N(x_r) - \overline{x}_r\|}{\|x_r - \overline{x}_r\|} \leq \left\|P_{T_M(\overline{x}_r)} P_{T_N(\overline{x}_r)} - P_{T_M(\overline{x}_r)\cap T_N(\overline{x}_r)}\right\|_{\mathrm{op}} + o(1).$$

By Lemma 5.38, this simplifies to

$$\frac{\|P_M P_N(x_r) - \overline{x}_r\|}{\|x_r - \overline{x}_r\|} \leq c(M, N; \overline{x}_r) + o(1).$$

Taking the limsup in this inequality and by Lemma 5.39, we get

$$\limsup_{x_r\to\overline{x}, x_r\notin M\cap N} \frac{\|P_M P_N(x_r) - \overline{x}_r\|}{\|x_r - \overline{x}_r\|} \leq c(M, N; \overline{x}),$$

concluding the proof. □

With the help of Theorem 5.9, the result in Theorem 5.40 can be generalized in the following way:

Corollary 5.41. *Let M and N be two transverse C^2 manifolds around a point $\overline{x} \in M \cap N$. Then we have for every $n \geq 1$:*

$$\limsup_{x\to\overline{x}, x\notin M\cap N} \frac{\|(P_M P_N)^n(x) - P_{M\cap N}(x)\|}{\|x - P_{M\cap N}(x)\|} \leq c(M, N; \overline{x})^{2n-1}.$$

This gives rise to the following remark.

Remark 5.42. *Under the assumptions of Theorem 5.40, we have that for every $c > c(M, N; \overline{x})$, there exists a radius $\varepsilon > 0$ such that for all $x \in B_\varepsilon(\overline{x})$, we have*

$$\|P_M P_N(x) - P_{M\cap N}(x)\| \leq c\|x - P_{M\cap N}(x)\|. \tag{40}$$

Having this, we can now give the main result for alternating projections between two transverse manifolds, cf. [70, Theorem 4.3].

Theorem 5.43. *Let $\mathbb{E}$ be a Euclidean space and let M and N be two transverse manifolds around a point $\overline{x} \in M \cap N$. Further let the starting point $x_0 \in \mathbb{E}$ be close enough to $\overline{x}$. Then the method of alternating projections and its sequence $(x_k)_{k\in\mathbb{N}}$, with $x_{k+1} = P_M P_N(x_k)$ for every $k \geq 0$, is well defined and $d(\{x_k\}, M \cap N)$ decreases Q-linearly to zero for $k \to \infty$. More precisely, given any $1 > c > c(M, N; \overline{x})$ and x_0 close enough to $\overline{x}$, the iterates satisfy*

$$d(\{x_{k+1}\}, M \cap N) \leq c \cdot d\left(\{x_k\}, M \cap N\right) \quad \text{for every } \ k \geq 0,$$

where $d(A, B)$ is the distance between two sets A and B, as in Definition 5.15. Furthermore, the iterates x_k converge linearly to a point $x^* \in M \cap N$. That is, for some constant $a > 0$, we have:

$$\|x_k - x^*\| \leq ac^k \quad \text{for every } \ k \geq 0.$$

Proof. Due to Lemma 5.36, we know that $c(M, N; \overline{x}) < 1$. Now choose c such that $c(M, N; \overline{x}) < c < 1$ and $\varepsilon > 0$ such that (40) is satisfied. Set $\delta = (1 - c)\frac{\varepsilon}{4} > 0$ and consider any starting point $x \in B_\delta(\overline{x})$. We will prove the theorem in two parts and use the same notation as in the proof of [70, Theorem 4.3].

For the first part, we will show by induction that the sequence $(x_k)_{k\in\mathbb{N}}$ is well defined and both x_k and its projection $\overline{x}_k := P_{M\cap N}(x_k)$ are elements of $B_\varepsilon(\overline{x})$ and satisfy the following system of inequalities for every $k \geq 0$.

$$\|x_k - \overline{x}_{k-1}\| \leq \delta c^k, \tag{H_1}$$

$$\|x_k - \overline{x}_k\| \leq \delta c^k, \tag{H_2}$$

$$\|\overline{x}_k - \overline{x}_{k-1}\| \leq 2\delta c^k, \tag{H_3}$$

$$\|\overline{x}_k - \overline{x}\| \leq 2\left(\sum_{i=0}^{k} c^i\right)\delta, \tag{H_4}$$

$$\|x_k - \overline{x}\| \leq 2\left(\sum_{i=0}^{k} c^i\right)\delta. \tag{H_5}$$

Let us define $\overline{x}_{-1} := \overline{x}_0$. Since $\|x_0 - \overline{x}_0\| \leq \|x_0 - \overline{x}\| \leq \delta$, inequalities (H_1), (H_2), (H_3) and (H_5) are satisfied in the case $k = 0$. For (H_4) with $k = 0$, we have

$$\|\overline{x}_0 - \overline{x}\| \leq \|\overline{x}_0 - x_0\| + \|x_0 - \overline{x}\| \leq 2\delta.$$

So the inequalities (H_1) to (H_5) hold for $k = 0$ and $x_0, \overline{x}_0 \in B_\varepsilon(\overline{x})$. Now assume that these inequalities hold for an arbitrary $k \geq 0$ and further assume that $x_k, \overline{x}_k \in B_\varepsilon(\overline{x})$. We will prove that these conditions also hold for k replaced by $k+1$.

If $x_k \in M \cap N$, there is nothing to prove. On the other hand, if $x_k \notin M \cap N$, we have $x_k \in B_\varepsilon(\overline{x})$. Hence, $P_M P_N(x_k) = x_{k+1}$ is well defined and the inequality

$$\|x_{k+1} - \overline{x}_k\| \leq c\|x_k - \overline{x}_k\|$$

holds according to Remark 5.42. This gives

$$d(\{x_{k+1}\}, M \cap N) \leq \|x_{k+1} - \overline{x}_k\| \leq c\|x_k - \overline{x}_k\| = c \cdot d(\{x_k\}, M \cap N). \tag{41}$$

To show (H_1) for k replaced by $k+1$, we use inequality (H_2) for k and inequality (41). This yields

$$\|x_{k+1} - \overline{x}_k\| \leq c\|x_k - \overline{x}_k\| \leq \delta c^{k+1}. \tag{42}$$

To show (H_2) for $k+1$, we already have $\|x_{k+1} - \overline{x}_{k+1}\| \leq \|x_{k+1} - \overline{x}_k\|$ by definition of the projection $\overline{x}_{k+1}$. Thus, inequality (42) gives

$$\|x_{k+1} - \overline{x}_{k+1}\| \leq \delta c^{k+1}. \tag{43}$$

From inequalities (42) and (43), we can obtain (H_3) for $k+1$:

$$\|\overline{x}_{k+1} - \overline{x}_k\| \leq \|\overline{x}_{k+1} - x_{k+1}\| + \|x_{k+1} - \overline{x}_k\| \leq 2\delta c^{k+1}. \tag{44}$$

Since $\|\overline{x}_{k+1} - \overline{x}\| \leq \|\overline{x}_{k+1} - \overline{x}_k\| + \|\overline{x}_k - \overline{x}\|$, inequalities (44) and (H_4) for k yield (H_4) for $k+1$:

$$\|\overline{x}_{k+1} - \overline{x}\| \leq 2\delta c^{k+1} + 2\left(\sum_{i=0}^{k} c^i\right)\delta \leq 2\left(\sum_{i=0}^{k+1} c^i\right)\delta. \tag{45}$$

Similarly, we have $\|x_{k+1} - \overline{x}\| \leq \|x_{k+1} - \overline{x}_k\| + \|\overline{x}_k - \overline{x}\|$ and inequalities (42) and (H_4) give inequality (H_5) for $k+1$:

$$\|x_{k+1} - \overline{x}\| \leq \delta c^{k+1} + 2\delta \sum_{i=0}^{k} c^i \leq 2\delta \sum_{i=0}^{k+1} c^i. \tag{46}$$

So the inequalities (H_1) to (H_5) also hold for $k+1$. Moreover, since

$$(1-c)\sum_{i=0}^{k} c^i = \sum_{i=0}^{k}(c^i - c^{i+1}) = 1 - c^{k+1} < 1,$$

we have $\sum_{i=0}^{k} c^i < \frac{1}{1-c}$ and the inequalities (45) and (46) yield

$$\|\overline{x}_{k+1} - \overline{x}\| \leq \frac{2\delta}{1-c} \leq \frac{\varepsilon}{2}$$

and

$$\|x_{k+1} - \overline{x}\| \leq \frac{\varepsilon}{2},$$

such that $\overline{x}_{k+1}$ and x_{k+1} are elements of $B_\varepsilon(\overline{x})$. This ends the proof of the first part by induction.

For the second part, we first prove the convergence of the sequence $(\overline{x}_k)_{k\in\mathbb{N}}$ of projections. To this end, we show that this sequence in $M \cap N \cap B_\varepsilon(\overline{x})$ is Cauchy. Consider inequality (H_3) and write for all indices k, l with $l \geq k \geq 0$:

$$\|\overline{x}_l - \overline{x}_k\| \leq \sum_{i=k+1}^{l} \|\overline{x}_i - \overline{x}_{i-1}\| \leq 2\delta \sum_{i=k+1}^{l} c^i \leq \frac{2\delta}{1-c} c^{k+1}, \tag{47}$$

where the last inequality comes from

$$(1-c)\sum_{i=k+1}^{l} c^i = \sum_{i=k+1}^{l} (c^i - c^{i+1}) = c^{k+1} - c^{l+1} < c^{k+1}.$$

Thus, the sequence $(\overline{x}_k)_{k\in\mathbb{N}}$ is Cauchy and hence converges to an element $x^* \in M \cap N$. More precisely, we have

$$\|\overline{x}_k - x^*\| \leq \frac{2\delta}{1-c} c^{k+1}.$$

Together with inequality (H_2), this implies the following:

$$\|x_k - x^*\| \leq \|x_k - \overline{x}_k\| + \|\overline{x}_k - x^*\| \leq \delta c^k + \frac{2\delta}{1-c} c^{k+1} = \left(1 + \frac{2c}{1-c}\right) \delta c^k.$$

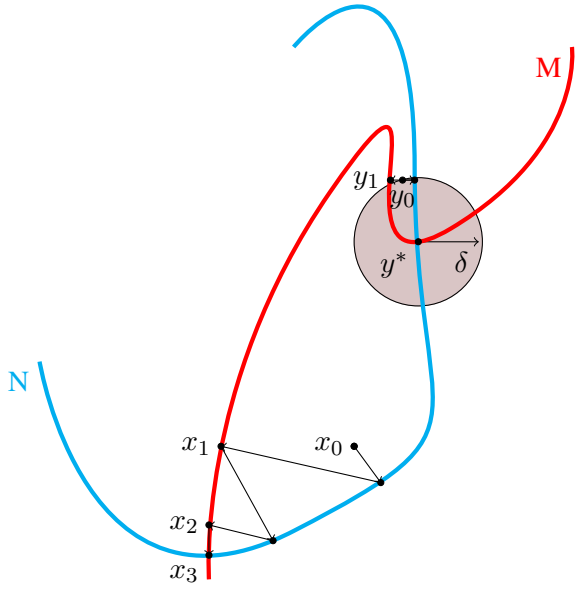
Thus, we have $\|x_k - x^*\| \leq a c^k$ for every $k \geq 0$, completing the proof. □

This theorem provides local convergence of the alternating projections approach for transverse manifolds around a point $\overline{x}$ of the intersection. It also provides the convergence rate depending on a parameter $c \in [c(M, N; \overline{x}), 1]$. For this to make sense, it is necessary to have $c(M, N; \overline{x}) < 1$, which is always the case due to Lemma 5.36.

To visualize the alternating projections method for transverse manifolds and to substantiate the local convergence, let us have a look at the Figure 5.8.

First, we consider the starting point x_0 and its orthogonal projection onto the blue manifold N. The point $P_N(x_0)$ is then projected onto the red manifold M. This is our next iterate x_1. We iterate this process and we see that the point x_3 is already very close to an element of the intersection $M \cap N$. In the limit, we will obtain a point in $M \cap N$. Since the manifolds are transverse in this intersection point, this is an example for the local convergence as described in Theorem 5.43. To show that we only have local convergence, we next consider the starting point y_0. Similar to x_0, we first project y_0 onto N and then we project $P_N(y_0)$ back onto M. This gives the next iterate y_1. If we now project y_1 onto N and $P_N(y_1)$ onto M, we obtain the same point y_1 again. So we reached a fixed point of the alternating projections approach, even though a point $y^* \in M \cap N$ and $\delta > 0$ exist such that $y_0 \in B_\delta(y^*)$. In this case, the value δ is not small enough, i.e., the starting

Figure 5.8: Alternating projections for two one-dimensional manifolds in $\mathbb{R}^2$



point y_0 is not close enough to y^*. Choosing a smaller radius $\widetilde{\delta}$ and a starting point $\widetilde{y}_0 \in B_{\widetilde{\delta}}(y^*)$ would ensure convergence of the iterates to y^* due to Theorem 5.43.

As mentioned in [70, Corollary 4.1], the convergence rate in Theorem 5.43 extends to the case of closed convex sets in the following setting.

Corollary 5.44. *Let A and B be two closed convex sets in the Euclidean space $\mathbb{E}$ such that the boundaries $\mathrm{bd}(A)$ and $\mathrm{bd}(B)$ are smooth manifolds. If the intersection $A \cap B$ is nonempty, then the sequence $(x_k)_{k\in\mathbb{N}}$, with x_0 given and $x_{k+1} = P_A P_B(x_k)$, is well defined and converges to a point $x^* \in A \cap B$. If the manifolds $\mathrm{bd}(A)$ and $\mathrm{bd}(B)$ intersect transversally in x^*, the sequence $(x_k)_{k\in\mathbb{N}}$ converges linearly with a rate depending on $c(\mathrm{bd}(A), \mathrm{bd}(B); x^*)$.*

So far, the alternating projections approach was analysed for two manifolds only. In the following section, we will extend the idea of alternating projections between manifolds to cyclic projections among a sequence of manifolds.

5.4 Cyclic Projections Among a Sequence of Manifolds

Hitherto, we showed the convergence of the alternating projections approach for two transverse smooth manifolds based on the results in [70]. We will use the approaches therein and the method of cyclic projections as introduced in Section 5.1 for subspaces to extend the result to the case of multiple intersecting manifolds $M_1, M_2, \ldots, M_n$. For the convergence result in this setting, we need that the intersection $\bigcap_{i=1}^n M_i$ itself is a C^k manifold around x with $T_{\bigcap_{i=1}^n M_i}(x) = \bigcap_{i=1}^n T_{M_i}(x)$. This can be ensured by the following lemma.

Lemma 5.45. *Let $M_1 \dots, M_n$ be C^k manifolds with $\overline{x} \in M := \bigcap_{i=1}^n M_i$. Further assume that for every $i \in \{2, \dots, n\}$, the manifolds M_i and $\left(\bigcap_{j<i} M_j\right)$ intersect transversally in $\overline{x}$. Then $\bigcap_{i=1}^n M_i$ itself is a C^k manifold around $\overline{x}$ with $T_{\bigcap_{i=1}^n M_i}(\overline{x}) = \bigcap_{i=1}^n T_{M_i}(\overline{x})$.*

Proof. We will prove this result inductively on the number of manifolds l. For $l = 2$, Lemma 5.33 shows that $M_{12} := M_1 \cap M_2$ is a C^k manifold and $T_{M_{12}}(\overline{x}) = T_{M_1}(\overline{x}) \cap T_{M_2}(\overline{x})$. Now we consider three manifolds M_1, M_2, M_3. By assumption, the manifolds M_3 and M_{12} are transverse such that Lemma 5.33 shows that

$$M_{123} := M_{12} \cap M_3 = (M_1 \cap M_2) \cap M_3$$

is a C^k manifold and

$$T_{M_{123}}(\overline{x}) = T_{M_{12}}(\overline{x}) \cap T_{M_3}(\overline{x}) = T_{M_1}(\overline{x}) \cap T_{M_2}(\overline{x}) \cap T_{M_3}(\overline{x}).$$

For increasing numbers of manifolds, this approach can be equivalently extended such that the statement will hold for every $l = 2, \dots, n$. □

Having this, we can now give the following asymptotical improvement result.

Theorem 5.46. *Let $M_1, \dots, M_n$ be C^2 manifolds around a common point $\overline{x} \in M := \bigcap_{i=1}^n M_i$. Further assume that for every $i \in \{2, \dots, n\}$, the manifolds M_i and $\left(\bigcap_{j<i} M_j\right)$ intersect transversally in $\overline{x}$ and let $T(\overline{x}) := \bigcap_{i=1}^n T_{M_i}(\overline{x})$. Then we have:*

$$\limsup_{x \to \overline{x},\, x \notin M} \frac{\left\|(P_{M_n} P_{M_{n-1}} \dots P_{M_1})(x) - P_M(x)\right\|}{\|x - P_M(x)\|} \leq c,$$

where

$$c = \left(1 - \prod_{i=1}^{n-1} \sin^2 \alpha \left(T_{M_i}(\overline{x}),\, \bigcap_{j=i+1}^{n} T_{M_j}(\overline{x})\right)\right)^{\frac{1}{2}}.$$

Proof. Due to Lemma 5.34, there exists $\varepsilon > 0$ such that the operators $P_{M_1}, \dots, P_{M_n}$ and P_M are of class C^1 and well defined in the neighbourhood $B_\varepsilon(\overline{x})$. To make sure that the fraction in the result is well defined, consider $\delta := \frac{\varepsilon}{2^{n-1}}$ and $x \in B_\delta(\overline{x})$ such that

$$\begin{aligned}\|\overline{x} - P_{M_1}(x)\| &= \|\overline{x} - x + x - P_{M_1}(x)\| \leq \|\overline{x} - x\| + \|x - P_{M_1}(x)\| \\ &\leq 2\|x - \overline{x}\| \leq 2\delta < \varepsilon.\end{aligned}$$

Thus, $(P_{M_2} P_{M_1})$ is well defined on $B_\delta(\overline{x})$ and

$$\begin{aligned}\|\overline{x} - (P_{M_2} P_{M_1})(x)\| &= \|\overline{x} - P_{M_1}(x) + P_{M_1}(x) - (P_{M_2} P_{M_1})(x)\| \\ &\leq \|\overline{x} - P_{M_1}(x)\| + \|P_{M_1}(x) - P_{M_2}(P_{M_1}(x))\| \\ &\leq 2\|P_{M_1}(x) - \overline{x}\| \leq 2^2\|x - \overline{x}\| \leq 4\delta < \varepsilon.\end{aligned}$$

Hence, $(P_{M_3}P_{M_2}P_{M_1})$ is well defined on $B_\delta(\overline{x})$ and by induction we get

$$\begin{aligned}
&\|\overline{x} - \left(P_{M_{n-1}}P_{M_{n-2}}\dots P_{M_1}\right)(x)\| \\
&= \|\overline{x} - \left(P_{M_{n-2}}\dots P_{M_1}\right)(x) + \left(P_{M_{n-2}}\dots P_{M_1}\right)(x) - \left(P_{M_{n-1}}P_{M_{n-2}}\dots P_{M_1}\right)(x)\| \\
&\leq 2\|\overline{x} - \left(P_{M_{n-2}}\dots P_{M_1}\right)(x)\| \leq 2^{n-1}\|x-\overline{x}\| \leq 2^{n-1}\delta = \varepsilon,
\end{aligned}$$

which shows $P_{M_n}P_{M_{n-1}}\dots P_{M_1}$ is also well defined on $B_\delta(\overline{x})$ and of class C^1. Thus, the fraction is well defined.

Now consider an arbitrary sequence $(x_r)_{r\in\mathbb{N}}$ in $B_\delta(\overline{x})\backslash M$ tending to $\overline{x}$. We will write $\overline{x}_r := P_M(x_r)$ to shorten notation. Thus, we have

$$(P_{M_n}P_{M_{n-1}}\dots P_{M_1})(x_r) - \overline{x}_r = (P_{M_n}P_{M_{n-1}}\dots P_{M_1})(x_r) - (P_{M_n}P_{M_{n-1}}\dots P_{M_1})(\overline{x}_r). \quad (48)$$

With the help of Lemma 5.45, we know that M is again a C^2 manifold. Lemma 5.34 then shows that the operator P_M is continuous and it follows

$$\lim_{r\to\infty}\overline{x}_r = \overline{x},$$

such that (48) can be written in the following way, using continuous differentiability:

$$\left(P_{M_n}P_{M_{n-1}}\dots P_{M_1}\right)(x_r) - \overline{x}_r = \nabla\left(P_{M_n}P_{M_{n-1}}\dots P_{M_1}\right)(\overline{x}_r)(x_r-\overline{x}_r) + o\left(\|x_r-\overline{x}_r\|\right). \quad (49)$$

And by the chain rule and Lemma 5.34, we have

$$\nabla(P_{M_n}P_{M_{n-1}}\dots P_{M_1})(\overline{x}_r) = P_{T_{M_n}(\overline{x}_r)}P_{T_{M_{n-1}}(\overline{x}_r)}\dots P_{T_{M_1}(\overline{x}_r)}. \quad (50)$$

Due to the Lemma 5.45, we have that $T(\overline{x}_r) = T_M(\overline{x}_r)$ and since $(x_r - \overline{x}_r) \in N_M(\overline{x}_r) = T_M(\overline{x}_r)^\perp$, we get

$$P_{T(\overline{x}_r)}(x_r-\overline{x}_r) = P_{T_M(\overline{x}_r)}(x_r-\overline{x}_r) = 0.$$

Thus, we can write

$$\begin{aligned}
&P_{T_{M_n}(\overline{x}_r)}P_{T_{M_{n-1}}(\overline{x}_r)}\dots P_{T_{M_1}(\overline{x}_r)}(x_r-\overline{x}_r) \\
&= \left(P_{T_{M_n}(\overline{x}_r)}P_{T_{M_{n-1}}(\overline{x}_r)}\dots P_{T_{M_1}(\overline{x}_r)} - P_{T(\overline{x}_r)}\right)(x_r-\overline{x}_r).
\end{aligned} \quad (51)$$

Combining equations (49), (50) and (51) gives

$$\begin{aligned}
&\frac{\left\|\left(P_{M_n}P_{M_{n-1}}\dots P_{M_1}\right)(x_r) - \overline{x}_r\right\|}{\|x_r-\overline{x}_r\|} \\
&= \left\|\left(P_{T_{M_n}(\overline{x}_r)}P_{T_{M_{n-1}}(\overline{x}_r)}\dots P_{T_{M_1}(\overline{x}_r)} - P_{T(\overline{x}_r)}\right)(x_r-\overline{x}_r)\right\|\frac{1}{\|x_r-\overline{x}_r\|} + o(1).
\end{aligned}$$

With the help of Theorem 5.10 and taking the limsup, we finally get

$$\limsup_{x\to\overline{x},\, x\notin M} \frac{\left\|(P_{M_n}P_{M_{n-1}}\dots P_{M_1})(x)-P_M(x)\right\|}{\|x-P_M(x)\|}\leq c,$$

with

$$c=\left(1-\prod_{i=1}^{n-1}\sin^2\alpha\left(T_{M_i}(\overline{x}),\bigcap_{j=i+1}^{n}T_{M_j}(\overline{x})\right)\right)^{\frac{1}{2}},$$

completing the proof. □

Corollary 5.47. *Under the assumptions of Theorem 5.46, we even get*

$$\limsup_{x\to\overline{x},\, x\notin M} \frac{\left\|(P_{M_n}P_{M_{n-1}}\dots P_{M_1})^n(x)-P_M(x)\right\|}{\|x-P_M(x)\|}\leq c^n,$$

with the same value c as in Theorem 5.46.

Moreover, we know that for every constant $\tilde{c}>c$, with c as in Theorem 5.46, there exists $\varepsilon>0$ such that for any starting point $x\in B_\varepsilon(\overline{x})$, we have

$$\left\|(P_{M_n}P_{M_{n-1}}\dots P_{M_1})(x)-P_M(x)\right\|\leq\tilde{c}\,\|x-P_M(x)\|\,. \tag{52}$$

Furthermore, the following lemma holds.

Lemma 5.48. *Let $M_1,\dots,M_n$ be C^2 manifolds around a point $\overline{x}\in M:=\bigcap_{i=1}^n M_i$. Further assume that for every $i\in\{1,\dots,n\}$ and every index set $J\subseteq\{1,\dots,n\}\setminus\{i\}$, the manifolds M_i and $\left(\bigcap_{j\in J}M_j\right)$ intersect transversally in $\overline{x}$. Let c be as defined in Theorem 5.46. Then we have $0\leq c<1$.*

Proof. By construction, we have $c\in[0,1]$. Moreover, Lemma 5.45 shows that for any $i\in\{1,\dots,n\}$, the intersection $\bigcap_{j=i+1}^n M_j$ itself is a C^2 manifold and for every $i\in\{1,\dots,n\}$, we have

$$\bigcap_{j=i+1}^{n}T_{M_j}(\overline{x})=T_{\bigcap_{j=i+1}^n M_j}(\overline{x}).$$

For every $i\in\{1,\dots,n\}$, Definition 5.35, Lemma 5.45 and Lemma 5.36 now show that

$$c\left(M_i,\bigcap_{j=i+1}^{n}M_j\,;\overline{x}\right)=\cos\alpha\left(T_{M_i}(\overline{x}),\bigcap_{j=i+1}^{n}T_{M_j}(\overline{x})\right)=\cos\alpha\left(T_{M_i}(\overline{x}),T_{\bigcap_{j=i+1}^n M_j}(\overline{x})\right)<1,$$

where the angle α is defined as in Definition 5.7. This yields

$$\alpha\left(T_{M_i}(\overline{x}),\bigcap_{j=i+1}^{n}T_{M_j}(\overline{x})\right)>0,$$

for every $i\in\{1,\dots,n\}$. The inequality $c<1$ now follows from Remark 5.11, completing the proof. □

Having this result, it is now possible to derive the following convergence result.

Theorem 5.49. *Let $\mathbb{E}$ be a Euclidean space and let $M_1, \ldots, M_n$ be manifolds around a point $\overline{x} \in M := \bigcap_{i=1}^n M_i$. Further assume that for every $i \in \{1, \ldots, n\}$ and index set $J \subseteq \{1, \ldots, n\} \setminus \{i\}$, the manifolds M_i and $\left(\bigcap_{j \in J} M_j\right)$ intersect transversally in $\overline{x}$. Further let the starting point $x_0 \in \mathbb{E}$ be close enough to $\overline{x}$. Then the method of cyclic projections and its sequence $(x_k)_{k \in \mathbb{N}}$, with $x_{k+1} = (P_{M_n} P_{M_{n-1}} \ldots P_{M_1})(x_k)$ for every $k \geq 0$, is well defined and $d(\{x_k\}, \bigcap_{i=1}^n M_i)$ decreases Q-linearly to zero for $k \to \infty$.*

More precisely, given any $1 > \tilde{c} > c$, with c as in Theorem 5.46, and x_0 close enough to $\overline{x}$, the iterates satisfy

$$d\left(\{x_{k+1}\}, \bigcap_{i=1}^n M_i\right) \leq \tilde{c} \cdot d\left(\{x_k\}, \bigcap_{i=1}^n M_i\right) \quad \text{for every } \; k \geq 0,$$

where $d(A, B)$ is the distance between two sets A and B as in Definition 5.15. Furthermore, the iterates x_k converge linearly to a point $x^* \in \bigcap_{i=1}^n M_i$. That is, for some constant $a > 0$, we have:

$$\|x_k - x^*\| \leq a\tilde{c}^k \quad \text{for every } \; k \geq 0.$$

Proof. Due to Lemma 5.48, we know that $c < 1$. Now choose $\tilde{c}$ such that $c < \tilde{c} < 1$ and $\varepsilon > 0$ such that (52) is satisfied. Set $\delta = (1 - \tilde{c})\frac{\varepsilon}{4} > 0$ and consider any starting point $x \in B_\delta(\overline{x})$. We will prove the theorem in two parts and use the notation in the proof of [70, Theorem 4.3].

For the first part, we will show by induction that the sequence $(x_k)_{k \in \mathbb{N}}$ is well defined and both x_k and its projection $\overline{x}_k := P_{\bigcap_{i=1}^n M_i}(x_k)$ are elements of $B_\varepsilon(\overline{x})$ and satisfy the following system of inequalities for every $k \geq 0$.

$$\|x_k - \overline{x}_{k-1}\| \leq \delta\tilde{c}^k, \tag{H_1}$$

$$\|x_k - \overline{x}_k\| \leq \delta\tilde{c}^k, \tag{H_2}$$

$$\|\overline{x}_k - \overline{x}_{k-1}\| \leq 2\delta\tilde{c}^k, \tag{H_3}$$

$$\|\overline{x}_k - \overline{x}\| \leq 2\left(\sum_{i=0}^k \tilde{c}^i\right)\delta, \tag{H_4}$$

$$\|x_k - \overline{x}\| \leq 2\left(\sum_{i=0}^k \tilde{c}^i\right)\delta. \tag{H_5}$$

Let us define $\overline{x}_{-1} := \overline{x}_0$. Since $\|x_0 - \overline{x}_0\| \leq \|x_0 - \overline{x}\| \leq \delta$, inequalities (H_1), (H_2), (H_3) and (H_5) are satisfied for $k = 0$. For (H_4) with $k = 0$, we have

$$\|\overline{x}_0 - \overline{x}\| \leq \|\overline{x}_0 - x_0\| + \|x_0 - \overline{x}\| \leq 2\delta.$$

So the inequalities (H_1) to (H_5) hold for $k = 0$ and we have $x_0, \overline{x}_0 \in B_\varepsilon(\overline{x})$. Now assume that these inequalities hold for an arbitrary $k \geq 0$ and further assume that $x_k, \overline{x}_k \in B_\varepsilon(\overline{x})$. We will prove that these conditions also hold for k replaced by $k + 1$.

If $x_k \in \bigcap_{i=1}^n M_i$, there is nothing to prove. On the other hand, if $x_k \notin \bigcap_{i=1}^n M_i$, we have $x_k \in B_\varepsilon(\overline{x})$. Hence, $(P_{M_n} P_{M_{n-1}} \dots P_{M_1})(x_k) = x_{k+1}$ is well defined and the inequality

$$\|x_{k+1} - \overline{x}_k\| \leq \tilde{c}\|x_k - \overline{x}_k\|$$

holds according to (52). This gives

$$d\left(\{x_{k+1}\}, \bigcap_{i=1}^n M_i\right) \leq \|x_{k+1} - \overline{x}_k\| \leq \tilde{c}\|x_k - \overline{x}_k\| = \tilde{c} \cdot d\left(\{x_k\}, \bigcap_{i=1}^n M_i\right). \tag{53}$$

To see that the inequalities (H_1) to (H_5) hold for k replaced by $k+1$, we use the same arguments as in Theorem 5.43 and again, this yields $x_{k+1}, \overline{x}_{k+1} \in B_\varepsilon(\overline{x})$. This ends the proof of the first part by induction.

For the second part, we first prove the convergence of the sequence $(\overline{x}_k)_{k\in\mathbb{N}}$ of projections. For this, we show that this sequence in $\bigcap_{i=1}^n M_i \cap B_\varepsilon(\overline{x})$ is Cauchy. Consider inequality (H_3) and write for all indices k, l with $l \geq k \geq 0$:

$$\|\overline{x}_l - \overline{x}_k\| \leq \sum_{i=k+1}^{l} \|\overline{x}_i - \overline{x}_{i-1}\| \leq 2\delta \sum_{i=k+1}^{l} \tilde{c}^i \leq \frac{2\delta}{1-\tilde{c}} \tilde{c}^{k+1}, \tag{54}$$

where the last inequality comes from

$$(1-\tilde{c}) \sum_{i=k+1}^{l} \tilde{c}^i = \sum_{i=k+1}^{l} (\tilde{c}^i - \tilde{c}^{i+1}) = \tilde{c}^{k+1} - \tilde{c}^{l+1} < \tilde{c}^{k+1}.$$

Thus, the cyclic projections sequence is Cauchy and hence converges to an element $x^* \in \bigcap_{i=1}^n M_i$. More precisely, we have

$$\|\overline{x}_k - x^*\| \leq \frac{2\delta}{1-\tilde{c}} \tilde{c}^{k+1}.$$

Together with inequality (H_2), this implies the following:

$$\|x_k - x^*\| \leq \|x_k - \overline{x}_k\| + \|\overline{x}_k - x^*\| \leq \left(1 + \frac{2\tilde{c}}{1-\tilde{c}}\right) \delta \tilde{c}^k.$$

Thus, we have $\|x_k - x^*\| \leq a\tilde{c}^k$ for every $k \geq 0$, completing the proof. □

This gives the desired convergence result for cyclic projections among a sequence of manifolds. In the following section, we will consider more general closed sets. The results in the next section will be used to derive a local convergence result for our main algorithm in Chapter 6.

5.5 Alternating Projections on Closed Sets and on Semialgebraic Sets

As we will see in this section, it is also possible to apply the alternating projections approach to general closed sets, which are not necessarily convex. In this context, Drusvyatskiy and coauthors developed fundamental results in [41] and [42], which are summarized in this section.

First, we will generalize the concept of transversality to closed sets. For this, it is necessary to consider a generalization of the normal space as given in Definition 5.31.

Definition 5.50. *Consider a Euclidean space $\mathbb{E}$, a set $Q \subseteq \mathbb{E}$ and $x \in Q$. Vectors in the set*

$$N_Q^p(x) := \{\lambda u \mid \lambda \in \mathbb{R} \text{ with } \lambda > 0,\ u \in \mathbb{E},\ x \in P_Q(x+u)\}$$

are called proximal normals to Q at x, and the set $N_Q^p(x)$ is called proximal normal cone to Q at x. Here $P_Q(x+u)$ denotes the set of best approximations to the vector $x+u$ from Q.

Let $(x_n)_{n\in\mathbb{N}}$ be a sequence in Q tending to x. Then the limits of the proximal normals are called normals. They form the normal cone $N_Q(x)$.

Remark 5.51. *If Q is a smooth manifold, then the set $N_Q(x)$ coincides with normal space in Definition 5.31 and therefore we can use the same notation.*

Again, the definition of a transverse intersection is understood pointwise for points in the intersection, now for two closed sets, cf. [41, Definition 3.2.1].

Definition 5.52. *Consider two closed sets Q and R in $\mathbb{E}$ and let $\overline{x} \in Q \cap R$. Then*

(a) Q and R are called transverse at $\overline{x}$ if

$$N_Q(\overline{x}) \cap (-N_R(\overline{x})) = \{0\}.$$

(b) Q and R are called intrinsically transverse at $\overline{x}$ with modulus $\kappa \in (0,1]$ if there exists $\varepsilon > 0$ such that for any $x \in Q \cap B_\varepsilon(\overline{x})$ and $y \in R \cap B_\varepsilon(\overline{x})$, we have

$$\max\left\{d\left(\left\{\frac{1}{\|y-x\|}(y-x)\right\}, N_Q(x)\right),\ d\left(\left\{\frac{1}{\|y-x\|}(y-x)\right\}, -N_R(x)\right)\right\} \geq \kappa,$$

where $d(\cdot,\cdot)$ denotes the minimal distance between two sets, as introduced in Definition 5.15.

Remark 5.53. *Here it should be mentioned that transversality implies intrinsic transversality, as shown in [42, Proposition 3.3].*

Having these definitions, we can now show the main result for alternating projections between closed sets, as given in [42, Theorem 6.1]. This result can also be found in [41, Theorem 3.2.3].

Theorem 5.54. *Let Q, R be closed subsets of a Euclidean space $\mathbb{E}$. Let Q and R be intrinsically transverse at a point $\overline{x} \in Q \cap R$ with $\kappa > 0$. Then for any constant $c \in (0, \kappa)$, there exists $\delta > 0$ such that for every starting point $x \in B_\delta(\overline{x})$, the alternating projections method converges linearly with rate $(1-c^2)$ to a point in $Q \cap R$.*

Figure 5.9: Alternating projections between closed sets

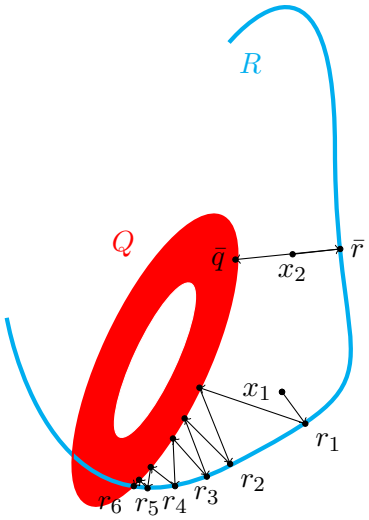


Figure 5.9 illustrates the alternating projections between closed sets. Both sets are nonconvex in this example. Considering starting point x_1, we can see that the von Neumann sequence in R converges to a point in $Q \cap R$ in the limit. To see that we only have local convergence in general, consider starting point x_2. Here the projection onto R gives the point $\bar{r}$. Projecting this point onto Q gives the point $\bar{q}$. Projecting $\bar{q}$ back onto R returns $\bar{r}$ again such that the sequence does not converge to a point in the intersection (but to a fixed point of the von Neumann sequence).

For the rest of this section, we will consider special closed sets in the Euclidean space $\mathbb{E}$, and we consider the following definition, see for example [30].

Definition 5.55. *A closed subset X of $\mathbb{E}$ is called semialgebraic if*

$$X = \{x \in \mathbb{E} \mid f_i(x) \geq 0 \text{ for every } i = 1, \ldots, m\},$$

where $f_1, f_2, \ldots, f_m : \mathbb{E} \to \mathbb{R}$ are polynomial functions, or if X is a finite union of such sets.

Theorem 5.54 clearly generalizes the convergence of alternating projections to the case of intersecting closed sets. According to the following lemma, cf. [42, Theorem 7.1], it is possible to derive a convergence result for the alternating projections approach for closed semialgebraic sets.

Lemma 5.56. *Consider two closed semialgebraic sets $X, Y \subseteq \mathbb{E}$. Then for almost every $x \in \mathbb{E}$, transversality holds at every point in the (possibly empty) intersection $X \cap (Y - x)$.*

With this, it is now possible to determine a convergence result for alternating projections between semialgebraic sets, cf. [42, Theorem 7.3].

Theorem 5.57. *Let X, Y be two nonempty closed semialgebraic subsets of a Euclidean space $\mathbb{E}$. In addition, let X be bounded. If the alternating projections approach starts in a point $x \in Y$ close enough to X, then the distance $d(\{x_n\}, X \cap Y)$ of the iterates to the intersection of X and Y converges to zero. Hence, every limit point is an element of $X \cap Y$.*

To check transversality or intrinsic transversality in advance, it would be necessary to know a priori a point in the intersection $X \cap Y$. Thus, the result in Theorem 5.57 is strong since we do not have to verify a transversality property in advance. Especially no point in the intersection is needed, which would make the alternating projections approach redundant.

We will use Theorem 5.57 to show local convergence of a first method to derive completely positive factorizations as a special application of alternating projections on semialgebraic sets. In the following chapters, we will have a closer look at this approach and show its convergence theoretically in Chapter 6, and illustrate the convergence for concrete examples in Chapter 7.

6 Applying Alternating Projections to Construct $\mathcal{CP}$-Factorizations

In this chapter, we will show algorithmic approaches to generate completely positive factorizations for matrices in the interior or at the boundary of the completely positive cone. Here a local convergence result will be given, based on the convergence of alternating projections between two semialgebraic sets. Moreover, we will see modifications of these approaches, which are still able to generate cp-factorizations for a given completely positive matrix, but take only a fraction of the computation time of the first approaches. The results in Sections 6.1 and 6.2 can also be found in the submitted article [50].

6.1 An Alternating Projections Approach for $\mathcal{CP}$-Factorizations

First, we will give a concrete approach, proving whether a given matrix A is completely positive. To this end, consider an initial factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$, where either $r = \operatorname{cpr}(A)$ if this quantity happens to be known, or otherwise we use the bound from Lemma 2.32 and set $r = \operatorname{cp}_n$. In both cases, we can ensure such an initial factorization with the approach introduced in Section 3.6.

Let us go back to the nonconvex feasibility problem (28) in Section 4.1. Therefore, we consider the problem:

$$\begin{aligned} \text{find} \quad & Q \\ \text{s.t.} \quad & BQ \geq 0 \\ & Q \in \mathcal{O}_r. \end{aligned} \tag{28}$$

As shown in Theorem 4.1, A is completely positive if and only if problem (28) is feasible. Introducing the polyhedral cone

$$\mathcal{P} := \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq 0\},$$

we can write (28) as

$$\text{find} \quad Q \in \mathcal{P} \cap \mathcal{O}_r. \tag{55}$$

We will use the method of alternating projections between semialgebraic sets, as introduced in Section 5.5, to obtain a point in the intersection of these two sets. In order to apply this method, we need to be able to project onto the sets $\mathcal{P}$ and $\mathcal{O}_r$. For this, we use the following results. To shorten notation, we will write $\|A\|$ for the Frobenius norm of a matrix A and $\|x\|$ for the Euclidean norm

of a vector x. This yields $\|A\| = \|\operatorname{vec}(A)\|$, where $\operatorname{vec}(A)$ stacks the columns of A in a long column vector. Moreover, the Frobenius norm is unitarily invariant, which means that for any two unitary matrices U, V, we have $\|UAV\| = \|A\|$. To see this, we write

$$\|UAV\|^2 = \operatorname{trace}((V^T A^T U^T)(UAV)) = \operatorname{trace}(A^T A) = \|A\|^2.$$

Now, for the projection onto $\mathcal{P}$, we have the following result.

Lemma 6.1. *The projection of a matrix M onto $\mathcal{P}$ is unique and computing it amounts to solving a second order cone problem (SOCP):*

$$\begin{array}{ll} \min & \|X - M\| \\ \text{s.t.} & BX \geq 0 \end{array} \quad \Leftrightarrow \quad \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq -BM \\ & (t, \operatorname{vec}(Y)) \in SOC. \end{array}$$

Proof. Since $\mathcal{P}$ is a polyhedral cone and hence convex, the projection of a matrix M onto $\mathcal{P}$ is unique. To show that it is sufficient to solve the mentioned second order cone problem, let $Y := X - M$ such that

$$\begin{array}{ll} \min & \|X - M\| \\ \text{s.t.} & BX \geq 0 \end{array} \Leftrightarrow \begin{array}{ll} \min & \|Y\| \\ \text{s.t.} & BY + BM \geq 0 \end{array} \Leftrightarrow \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq -BM \\ & t \geq \|Y\| \end{array}$$

$$\Leftrightarrow \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq -BM \\ & t \geq \|\operatorname{vec}(Y)\| \end{array} \Leftrightarrow \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq -BM \\ & (t, \operatorname{vec}(Y)) \in SOC, \end{array}$$

completing the proof. □

Note that SOCPs can be solved in polynomial time using interior point methods. Hence, the projection onto $\mathcal{P}$ can be calculated via solving a second order cone problem. Here for example the solvers SDPT3 (cf. [92] or [93]) or Sedumi (cf. [91]) can be used.

On the other hand, the projection of a matrix M onto $\mathcal{O}_r$, the set of orthogonal matrices, always exists since $\mathcal{O}_r$ is compact, as shown in Lemma 3.2. However, it may not be unique due to the nonconvexity of $\mathcal{O}_r$. We therefore consider the best approximation as introduced in Definition 5.1 and for manifolds in equation (35). If we denote by $P_{\mathcal{O}_r}(M)$ the set of best approximations to $\mathcal{O}_r$ in M, computing an element of $P_{\mathcal{O}_r}(M)$ can be done through the polar decomposition of M according to the following lemma, a proof of which can be found in [12, Corollary 5.6.4 and Fact 9.9.42].

Lemma 6.2. *Let $M \in \mathbb{R}^{r\times r}$. Then there exists the so called polar decomposition of M, i.e., there exist a positive semidefinite matrix $T \in \mathbb{R}^{r\times r}$ and an orthogonal matrix $Q \in \mathbb{R}^{r\times r}$ such that*

$$M = TQ.$$

For any unitarily invariant norm $\|\cdot\|$, we have

$$\|M - Q\| \leq \|M - U\| \text{ for all } U \in \mathcal{O}_r.$$

We take this $Q \in P_{\mathcal{O}_r}(M)$. The polar decomposition can be computed via the singular value decomposition, as the following lemma shows. For a short review and some basic facts on the singular value decomposition, the reader is again referred to the Appendix of this thesis.

Lemma 6.3. *To obtain the polar decomposition, and therefore a best approximation of a given matrix M in $\mathcal{O}_r$, we take the singular value decomposition $M = U\Sigma V^T$ of M. Here U, V are orthogonal matrices and Σ is a diagonal matrix containing the singular values of M. Then for the polar decomposition of M, we have*

$$M = TQ, \text{ where } T = U\Sigma U^T \text{ and } Q = UV^T.$$

Proof. First, we observe

$$TQ = U\Sigma U^T UV^T = U\Sigma V^T = M$$

and since $U, V \in \mathcal{O}_r$, we get $Q \in \mathcal{O}_r$. On the other hand, we know that T is positive semidefinite since Σ contains the nonnegative singular values of M and is therefore positive semidefinite. □

Remark 6.4. *Let $M \in \mathbb{R}^{r\times r}$. The computation of the polar decomposition of M can be done in $O(r^3)$ steps according to Lemma 6.3 and [48, Section 5.4.5].*

Hence, we can efficiently compute the projections $P_{\mathcal{P}}(M)$ and $P_{\mathcal{O}_r}(M)$ of a matrix M onto $\mathcal{P}$ and $\mathcal{O}_r$, respectively. The alternating projections method to compute a factorization of a completely positive matrix A now reads as follows:

Algorithm 1 Alternating projections between $\mathcal{P}$ and $\mathcal{O}_r$

Input: $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}(A)$; initial matrix $Q_0 \in \mathcal{O}_r$

1: $k \leftarrow 0$
2: **while** $BQ_k \ngeq 0$ **do**
3: $\quad P_k \leftarrow P_{\mathcal{P}}(Q_k)$
4: $\quad Q_{k+1} \leftarrow P_{\mathcal{O}_r}(P_k)$
5: $\quad k \leftarrow k + 1$
6: **end while**

Output: $Q_k \in \mathcal{O}_r$ and a completely positive factorization $A = (BQ_k)(BQ_k)^T$

Numerically, we can stop the algorithm whenever $BQ_k \geq -\varepsilon E$ for some $\varepsilon > 0$ or the iteration counter k reaches its predefined maximum $k_{\max}$. For the numerical experiments in Chapter 7, we will use $k_{\max} = 5000$ and $\varepsilon = 10^{-15}$ for most instances.

To prove a local convergence result for this approach, we will first show that both sets $\mathcal{P}$ and $\mathcal{O}_r$ are semialgebraic sets as introduced in Definition 5.55.

Lemma 6.5. *The sets $\mathcal{P}$ and $\mathcal{O}_r$ are semialgebraic sets.*

Proof. By definition we have $\mathcal{O}_r = \{Q \in \mathbb{R}^{r\times r} \mid QQ^T - I = 0\}$. The set is therefore given as a solution set of a set of polynomial equations and is therefore semialgebraic. On the other hand, we have $\mathcal{P} = \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq 0\}$. The set is therefore given as a solution set of a set of polynomial (even linear) inequalities and is therefore semialgebraic as well. □

Local convergence for Algorithm 1 is now ensured by the following theorem, which can also be found in the submitted article, cf. [50, Theorem 4.2].

Theorem 6.6. *Let $A \in \mathcal{CP}_n$. Let $A = BB^T$ be any initial factorization with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}(A)$. Define $\mathcal{P} := \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq 0\}$. Then we have:*

(a) $\mathcal{P} \cap \mathcal{O}_r \neq \emptyset$,

(b) if started at a point Q_0 close to $\mathcal{P} \cap \mathcal{O}_r$, then Algorithm 1 converges to a point $Q^* \in \mathcal{P} \cap \mathcal{O}_r$. In this case, $A = (BQ^*)(BQ^*)^T$ is a completely positive factorization of A.

Proof. (a): It follows from $A \in \mathcal{CP}_n$ and $r \geq \operatorname{cpr}(A)$ that there exists $C \in \mathbb{R}^{n\times r}$, $C \geq 0$ with $A = CC^T$. Since $A = BB^T = CC^T$, Lemma 3.11 implies that there exists $Q \in \mathcal{O}_r$ such that $BQ = C \geq 0$, i.e., $Q \in \mathcal{P} \cap \mathcal{O}_r$.

(b): Both $\mathcal{P}$ and $\mathcal{O}_r$ are closed semialgebraic sets due to Lemma 6.5. Moreover, $\mathcal{O}_r$ is bounded, as shown in Lemma 3.2. The convergence result for this setting now follows by applying the result in [42, Theorem 7.3], which can also be found in Theorem 5.57 of this thesis. □

So Algorithm 1 provides local convergence to a point in the intersection of both sets and therefore a cp-factorization of the given matrix A. But in Step 3 of Algorithm 1, we have to solve an SOCP in every iteration to compute $P_{\mathcal{P}}(Q_k)$. Even though this can be done in polynomial time, it is still very costly. Here it turns out that a slight modification of this step provides a much better numerical performance. In the following section, we will have a closer look at this modified method.

6.2 Modifying the Alternating Projections Method

Although solving an SOCP in every projection step onto $\mathcal{P}$ can be computed in polynomial time, it is still very costly. Instead of projecting onto the set $\mathcal{P}$ in Algorithm 1 via the SOCP reformulation introduced in Lemma 6.1, we proceed as follows. Nevertheless, we will lose the local convergence theory.

As a substitute for computing the projection $P_{\mathcal{P}}(Q)$ of Q onto $\mathcal{P} := \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq 0\}$, we rather project BQ onto the nonnegative orthant by computing a matrix $D \in \mathbb{R}^{n\times r}$ through

$$D_{ij} := \max\{(BQ)_{ij}, 0\} \quad \text{for all } i = 1, \ldots, n \text{ and } j = 1, \ldots, r. \tag{56}$$

Since the nonnegative orthant of the matrix space $\mathbb{R}^{n\times r}$ is convex, this projection is always unique.

Note that $D \in \mathbb{R}^{n\times r}$, so in order to obtain an approximation to Q in $\mathcal{P}$, we need to lift D into the space $\mathbb{R}^{r\times r}$. For this, we will use the following tool, cf. [60, Theorem 2] and the main result in [80]:

Lemma 6.7. *Let B, D be as introduced above and consider the equation $BX = D$. If the equation is solvable, then the complete set of solutions is given as*

$$S := \left\{ X = B^+D + (I - B^+B)Y \;\middle|\; Y \in \mathbb{R}^{n\times n} \right\} \subseteq \mathbb{R}^{r\times r},$$

where B^+ denotes the Moore-Penrose-inverse.

In case the equation $BX = D$ has no solution, we get that $\|BX - D\|$ is minimal if and only if $X \in S$.

For some basic facts on the Moore-Penrose-inverse, the reader is again referred to the Appendix of this thesis.

Based on Lemma 6.7, we further have the following result, cf. [80].

Lemma 6.8. *Let B, D be as introduced above and consider the equation $BX = D$. Further, let $\bar{X} := B^+D$. Then*

$$\|\bar{X}\|^2 \leq \|X\|^2 \quad \textit{for every } X \in S.$$

Proof. Let $X \in S$. Then there exists a matrix Y such that

$$X = B^+D + (I - B^+B)Y.$$

First, we will show that B^+D and $(I - B^+B)Y$ are orthogonal, based on the properties of B^+:

$$\begin{aligned}
\left\langle B^+D, (I - B^+B)Y \right\rangle &= \operatorname{trace}\left((B^+D)^T(I - B^+B)Y\right) \\
&= \operatorname{trace}\left((B^+BB^+D)^T(I - B^+B)Y\right) \\
&= \operatorname{trace}\left((B^+D)^TB^+B(I - B^+B)Y\right) \\
&= \operatorname{trace}\left((B^+D)^TB^+(B - BB^+B)Y\right) \\
&= \operatorname{trace}\left((B^+D)^TB^+(B - B)Y\right) \\
&= 0.
\end{aligned}$$

Hence,

$$\|X\|^2 = \|B^+D + (I - B^+B)Y\|^2 = \|B^+D\|^2 + \|(I - B^+B)Y\|^2 \geq \|B^+D\|^2 = \|\bar{X}\|^2,$$

completing the proof. □

Remark 6.9. *$\bar{X}$ is therefore called the least squares solution of the equation $BX = D$. And since $(I - B^+B)Y$ is the projection of Y onto the kernel of B, we can write every solution of the equation $BX = D$ as a sum of the least squares solution and an element of the kernel of B. More generally, we know that $X = B^+D + (I - B^+B)Y$ is the solution of the least squares Problem $\|BX - D\|^2$, which is closest to Y.*

Based on this remark, we can now give an approximation to Q in $\mathcal{P}$, which can be computed easily and especially without solving an SOCP.

Lemma 6.10. *Let D be the projection of BQ onto the nonnegative orthant. If $D = BQ$, then $Q = P_{\mathcal{P}}(Q) \in \mathcal{P}$. If on the other hand $D \neq BQ$, we let B^+ denote the Moore-Penrose-inverse of B and define*

$$\widehat{P} := B^+D + (I - B^+B)Q \ \in \mathbb{R}^{r\times r}.$$

If the equation $BX = D$ has a solution X, then $\widehat{P} \in \mathcal{P}$.

Proof. If $D = BQ$, then $BQ \geq 0$, i.e., $Q \in \mathcal{P}$ and therefore Q equals its projection onto $\mathcal{P}$. Otherwise let $D \neq BQ$ and assume that $BX = D$ has a solution X. The assumption that $D \neq BQ$ means that Q does not solve the equation $BX = D$. Since the equation is solvable, $\widehat{P}$ is a solution due to Lemma 6.7. Furthermore, it is the unique solution which minimizes the distance to Q according to Remark 6.9. Thus, $\widehat{P}$ is the projection of Q onto the set $\{X \in \mathbb{R}^{r\times r} \mid BX = D\}$. This set is a subset of $\mathcal{P}$ since $D \geq 0$. So in this case, we have $\widehat{P} \in \mathcal{P}$. □

In addition, if the equation $BX = D$ does not have a solution, then $X = \widehat{P}$ minimizes the residual $\|BX - D\|$ and among all minimizers it is the one closest to Q. In this case, we get from the properties of B^+ that

$$B\widehat{P} = BB^+D + (B - BB^+B)Q = BB^+D,$$

and if in addition the rows of B are linearly independent (which is true if A has full rank), then $BB^+ = I_n$, which implies that $B\widehat{P} = D \geq 0$, and hence again $\widehat{P} \in \mathcal{P}$. If the rows of B are linearly dependent, then it may happen that $\widehat{P} \notin \mathcal{P}$, however this does not seem to impair the good numerical performance.

From now on, we will take $\widehat{P}$ as an approximation of $P_{\mathcal{P}}(Q)$. This reasoning leads to the following modification of Algorithm 1:

Algorithm 2 Modified algorithm for completely positive matrix factorizations

Input: $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}(A)$; initial matrix $Q_0 \in \mathcal{O}_r$

1: $k \leftarrow 0$
2: **while** $BQ_k \ngeq 0$ **do**
3: $\quad D \leftarrow \max\{BQ_k, 0\}$ entrywise
4: $\quad \widehat{P_k} \leftarrow B^+ D + (I - B^+B)Q_k$
5: $\quad Q_{k+1} \leftarrow P_{\mathcal{O}_r}(\widehat{P_k})$
6: $\quad k \leftarrow k + 1$
7: **end while**

Output: $Q_k \in \mathcal{O}_r$ and a completely positive factorization $A = (BQ_k)(BQ_k)^T$

Clearly, if Algorithm 2 terminates, then it yields a completely positive factorization of A. However, since Algorithm 2 is not a pure alternating projections method, we do not get a local con-

vergence result like in Theorem 6.6. Nevertheless, numerical experiments in Chapter 7 will show that this algorithm is highly efficient.

In the following section, we will see that it is possible to modify Algorithms 1 and 2 to show that a matrix is even an element of the interior of the completely positive cone.

6.3 Algorithms for Matrices in the Interior of the Completely Positive Cone

In this section, we will first see an algorithm based on Algorithm 1, which can be used to gain a certificate for a given matrix A to be an element of the interior of the completely positive cone. Assume that we are given an initial factorization $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$, where either $r = \mathrm{cpr}^+(A)$ if this quantity happens to be known, or if the exact cp^+-rank is unknown, then the bound $r = \mathrm{cp}_n^+$ from Lemma 2.32 may be used. As described in Theorem 4.7, solving one of the feasibility problems in (32) or (33) is equivalent to proving $A \in \mathrm{int}(\mathcal{CP}_n)$.

To use Algorithm 1 in this setting, we have to replace the set $\mathcal{P}$ with one of the following sets

$$\mathcal{P}_{\varepsilon,1} := \left\{ Q \in \mathbb{R}^{r\times r} \;\middle|\; BQ_{ij} \geq \left\{ \begin{array}{ll} \varepsilon, & j = 1,\ i = 1,\dots,n \\ 0, & j \neq 1,\ i = 1,\dots,n \end{array} \right\} \right\} \quad \text{or}$$
$$\mathcal{P}_{\varepsilon,2} := \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq \varepsilon E_{n\times r}\},$$

where $\varepsilon > 0$ is a small threshold value.

Remark 6.11. *If we denote by E_ε the matrix whose entries of the first column are equal to ε and all other entries are equal to 0, then this yields*

$$\mathcal{P}_{\varepsilon,1} = \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq E_\varepsilon\}.$$

Similar to Lemma 6.1, we have the following result:

Lemma 6.12. *The projection of a matrix M onto $\mathcal{P}_{\varepsilon,1}$ or $\mathcal{P}_{\varepsilon,2}$ is unique and computing it amounts to solving an explicit second order cone problem (SOCP) for each of the sets:*

For the projection onto $\mathcal{P}_{\varepsilon,1}$, we consider the problems

$$\begin{array}{ll} \min & \|X - M\| \\ \text{s.t.} & BX \geq E_\varepsilon \end{array} \quad \Leftrightarrow \quad \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq E_\varepsilon - BM \\ & (t, \mathrm{vec}(Y)) \in SOC. \end{array}$$

And for the projection onto $\mathcal{P}_{\varepsilon,2}$, we consider the problems

$$\begin{array}{ll} \min & \|X - M\| \\ \text{s.t.} & BX \geq \varepsilon E_{n\times r} \end{array} \quad \Leftrightarrow \quad \begin{array}{ll} \min & t \\ \text{s.t.} & BY \geq \varepsilon E_{n\times r} - BM \\ & (t, \mathrm{vec}(Y)) \in SOC. \end{array}$$

Proof. Since $\mathcal{P}_{\varepsilon,1}$ and $\mathcal{P}_{\varepsilon,2}$ are polyhedral sets and hence convex, the projection of a matrix M onto $\mathcal{P}_{\varepsilon,1}$ or $\mathcal{P}_{\varepsilon,2}$ is unique. With the same argument as in Lemma 6.1 and $Y := X - M$, we get for $\mathcal{P}_{\varepsilon,1}$:

$$\begin{aligned}
&\min \quad \|X - M\| && && \min \quad \|Y\| && \Leftrightarrow && \min \quad t \\
&\text{s.t.} \quad BX \geq 0 && \Leftrightarrow && \text{s.t.} \quad BY \geq E_\varepsilon - BM && && \text{s.t.} \quad BY \geq E_\varepsilon - BM \\
& && && && && t \geq \|Y\| \\
& && \Leftrightarrow && \min \quad t && \Leftrightarrow && \min \quad t \\
& && && \text{s.t.} \quad BY \geq E_\varepsilon - BM && && \text{s.t.} \quad BY \geq E_\varepsilon - BM \\
& && && t \geq \|\operatorname{vec}(Y)\| && && (t, \operatorname{vec}(Y)) \in SOC,
\end{aligned}$$

Replacing E_ε with $\varepsilon E_{n\times r}$ shows the result for $\mathcal{P}_{\varepsilon,2}$, completing the proof. □

So we can compute the projection of a matrix onto $\mathcal{P}_{\varepsilon,1}$ or $\mathcal{P}_{\varepsilon,2}$. This gives rise to the following algorithms, which are based on Algorithm 1. For the first algorithm, we consider the set $\mathcal{P}_{\varepsilon,1}$.

Algorithm 3 Alternating projections between $\mathcal{P}_{\varepsilon,1}$ and $\mathcal{O}_r$

Input: $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$; initial matrix $Q_0 \in \mathcal{O}_r$; $\varepsilon > 0$

1: $k \leftarrow 0$
2: **while** $BQ_k \not\geq E_\varepsilon$ **do**
3: $\quad P_k \leftarrow P_{\mathcal{P}_{\varepsilon,1}}(Q_k)$
4: $\quad Q_{k+1} \leftarrow P_{\mathcal{O}_r}(P_k)$
5: $\quad k \leftarrow k + 1$
6: **end while**

Output: $Q_k \in \mathcal{O}_r$ and a cp-factorization $A = (BQ_k)(BQ_k)^T$ with $(BQ_k) \geq E_\varepsilon$

Alternatively, considering the set $\mathcal{P}_{\varepsilon,2}$ gives rise to the following algorithm.

Algorithm 4 Alternating projections between $\mathcal{P}_{\varepsilon,2}$ and $\mathcal{O}_r$

Input: $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$; initial matrix $Q_0 \in \mathcal{O}_r$; $\varepsilon > 0$

1: $k \leftarrow 0$
2: **while** $BQ_k \not\geq \varepsilon E_{n\times r}$ **do**
3: $\quad P_k \leftarrow P_{\mathcal{P}_{\varepsilon,2}}(Q_k)$
4: $\quad Q_{k+1} \leftarrow P_{\mathcal{O}_r}(P_k)$
5: $\quad k \leftarrow k + 1$
6: **end while**

Output: $Q_k \in \mathcal{O}_r$ and a cp-factorization $A = (BQ_k)(BQ_k)^T$ with $(BQ_k) > 0$

In addition, note that $\mathcal{P}_{\varepsilon,1}$ and $\mathcal{P}_{\varepsilon,2}$ are semialgebraic sets since they are the solution set to a set of polynomial inequalities. Hence, Theorem 6.6 extends to this setting and we get the following result.

Theorem 6.13. *Let $A \in \operatorname{int}(\mathcal{CP}_n)$. Further let $A = BB^T$ be any initial factorization with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$. Define $\mathcal{P}_{\varepsilon,1} = \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq E_\varepsilon\}$ and $\mathcal{P}_{\varepsilon,2} = \{Q \in \mathbb{R}^{r\times r} \mid BQ \geq \varepsilon E_{n\times r}\}$ for $\varepsilon > 0$ and E_ε as introduced in Remark 6.11. Then we have:*

(a) There exists $\varepsilon > 0$ such that $\mathcal{P}_{\varepsilon,1} \cap \mathcal{O}_r \neq \emptyset$ and $\mathcal{P}_{\varepsilon,2} \cap \mathcal{O}_r \neq \emptyset$.

(b) Let $\varepsilon > 0$ be small enough. If started at a point Q_0 close to $\mathcal{P}_{\varepsilon,1} \cap \mathcal{O}_r$ (resp. to $\mathcal{P}_{\varepsilon,2} \cap \mathcal{O}_r$), then Algorithm 3 (resp. Algorithm 4) converges to a point $Q^* \in \mathcal{P}_{\varepsilon,1} \cap \mathcal{O}_r$ (resp. $Q^* \in \mathcal{P}_{\varepsilon,2} \cap \mathcal{O}_r$). In this case, $A = (BQ^*)(BQ^*)^T$ is a completely positive factorization of A with $BQ^* \geq E_\varepsilon$ ($BQ^* > 0$ resp.), showing that $A \in \operatorname{int}(\mathcal{CP}_n)$.

Proof. We will prove both parts separately.

(a): Since $A \in \operatorname{int}(\mathcal{CP}_n)$ and $r \geq \operatorname{cpr}^+(A)$, it follows from equation (7) that there exists $C \in \mathbb{R}^{n\times r}$ such that $C > 0$ and $A = CC^T$. Since $A = BB^T = CC^T$, Lemma 3.11 implies that there exists $Q \in \mathcal{O}_r$ such that $BQ = C > 0$. Let $\varepsilon_1 := \min_{i,j} C_{ij} > 0$, then $Q \in \mathcal{P}_{\varepsilon_1,2} \cap \mathcal{O}_r$. On the other hand, due to equation (8), there exists $\bar{C} \in \mathbb{R}^{n\times r}$ such that $\bar{C} \geq 0$, the first column of $\bar{C}$ is entrywise strictly positive and $A = \bar{C}\bar{C}^T$. Again Lemma 3.11 implies that there exists $Q \in \mathcal{O}_r$ such that $BQ = \bar{C}$. Let $\varepsilon_2 := \min_i \bar{C}_{i1} > 0$, then $Q \in \mathcal{P}_{\varepsilon_2,1} \cap \mathcal{O}_r$. Finally, we define $\varepsilon := \min\{\varepsilon_1, \varepsilon_2\}$, proving part (a).

(b): Both $\mathcal{P}_{\varepsilon,1}$ (respectively $\mathcal{P}_{\varepsilon,2}$) and $\mathcal{O}_r$ are closed semialgebraic sets and $\mathcal{O}_r$ is bounded. The convergence result now follows by applying [42, Theorem 7.3], which was given as Theorem 5.57 in this thesis. □

We can also give a modified algorithm based on Algorithm 2 to avoid second order cone problems in every projection step onto $\mathcal{P}_{\varepsilon,1}$ or $\mathcal{P}_{\varepsilon,2}$. To this end, it is necessary to replace the projection onto the nonnegative orthant in equation (56) with the following projection for some $\varepsilon > 0$:

If we use $\mathcal{P}_{\varepsilon,1}$, let

$$D^{\varepsilon,1}_{ij} := \max\{BQ_{ij}, (E_\varepsilon)_{ij}\} \quad \text{for all } i = 1, \ldots, n \text{ and } j = 1, \ldots, r,$$

where E_ε is as defined in Remark 6.11. Further for $\mathcal{P}_{\varepsilon,2}$, let

$$D^{\varepsilon,2}_{ij} := \max\{BQ_{ij}, \varepsilon\} \quad \text{for all } i = 1, \ldots, n \text{ and } j = 1, \ldots, r.$$

Compared to Lemma 6.10, we have the following stronger result in this setting.

Lemma 6.14. *Consider the matrices $D^{\varepsilon,1}$ and $D^{\varepsilon,2}$ as defined above. If $BQ = D^{\varepsilon,i}$ for any $i \in \{1, 2\}$, then $Q \in \mathcal{P}_{\varepsilon,i}$. Moreover, let B^+ denote the Moore-Penrose-inverse of B and define*

$$\widehat{P}^{\varepsilon,i} := B^+ D^{\varepsilon,i} + (I - B^+ B)Q \in \mathbb{R}^{r\times r},$$

for every $i \in \{1, 2\}$. Then $\widehat{P}^{\varepsilon,i} \in \mathcal{P}_{\varepsilon,i}$ for every $i \in \{1, 2\}$.

Proof. Since the proofs are equal for any choice of $i \in \{1, 2\}$, will will prove the result in general for $i \in \{1, 2\}$: If $D^{\varepsilon,i} = BQ$, then $Q \in \mathcal{P}_{\varepsilon,i}$ by definition. Now let $D^{\varepsilon,i} \neq BQ$ and assume

that $BX = D^{\varepsilon,i}$ has a solution X. Since $D^{\varepsilon,i} \neq BQ$, the matrix Q does not solve the equation $BX = D^{\varepsilon,i}$. Since the equation is solvable, $\widehat{P}^{\varepsilon,i}$ is a solution due to Lemma 6.7. Moreover, it is the unique solution of the equation $BX = D^{\varepsilon,i}$ which minimizes the distance to Q according to Remark 6.9. Thus, the matrix $\widehat{P}^{\varepsilon,i}$ is the projection of Q onto the set $\{X \in \mathbb{R}^{r\times r} \mid BX = D^{\varepsilon,i}\}$. This set is a subset of $\mathcal{P}_{\varepsilon,i}$ by definition. So we get $\widehat{P}^{\varepsilon,i} \in \mathcal{P}_{\varepsilon,i}$.

Furthermore, we can show that there always exists a solution X to $BX = D^{\varepsilon,i}$. To see this, note that $X = \widehat{P}^{\varepsilon,i}$ minimizes the residual $\|BX - D^{\varepsilon,i}\|$ and among all minimizers it is the one closest to Q. In this case, the properties of B^+ yield

$$B\widehat{P}^{\varepsilon,i} = BB^+D^{\varepsilon,i} + (B - BB^+B)Q = BB^+D^{\varepsilon,i}.$$

Since $A \in \operatorname{int}(\mathcal{CP}_n)$, we know A is of full rank by definition. Thus, the rows of B are linearly independent inducing $BB^+ = I_n$, cf. Lemma A.3. This now implies $B\widehat{P}^{\varepsilon,i} = D^{\varepsilon,i}$ and hence again $\widehat{P}^{\varepsilon,i} \in \mathcal{P}_{\varepsilon,i}$. □

Thus, we will take $\widehat{P}^{\varepsilon,1}$ resp. $\widehat{P}^{\varepsilon,2}$ as an approximation of $P_{\mathcal{P}_{\varepsilon,1}}(Q)$ respectively $P_{\mathcal{P}_{\varepsilon,2}}(Q)$. This motivates the following modified algorithm. Here we consider only $P_{\varepsilon,2}$. To obtain the Algorithm for $P_{\varepsilon,1}$, we simply have to replace the set $D^{\varepsilon,2}$ with $D^{\varepsilon,1}$ and the algorithm terminates if $BQ_k \geq E_\varepsilon$, where E_ε is as defined in Remark 6.11.

Algorithm 5 Modified algorithm for the interior of the completely positive cone

Input: $A = BB^T$ with $B \in \mathbb{R}^{n\times r}$ and $r \geq \operatorname{cpr}^+(A)$; initial matrix $Q_0 \in \mathcal{O}_r$; $\varepsilon > 0$

1: $k \leftarrow 0$
2: **while** $BQ_k \not\geq \varepsilon E_{n\times r}$ **do**
3: $\quad D^{\varepsilon,2} \leftarrow \max\{BQ_k, \varepsilon E_{n\times r}\}$ entrywise
4: $\quad \widehat{P_k} \leftarrow B^+ D^{\varepsilon,2} + (I - B^+B)Q_k$
5: $\quad Q_{k+1} \leftarrow P_{\mathcal{O}_r}(\widehat{P_k})$
6: $\quad k \leftarrow k + 1$
7: **end while**

Output: $Q_k \in \mathcal{O}_r$ and a completely positive factorization $A = (BQ_k)(BQ_k)^T$ with $(BQ_k) > 0$

Again, this approach is not a pure alternating projections method such that we loose the local convergence result in this case.

In the following chapter, we will see numerical experiments which prove that the modified Algorithms 2 and 5 are highly efficient. Furthermore, we will illustrate the convergence of Algorithm 1 and 4 for concrete examples.

7 Numerical Results

In this chapter, we will analyse the numerical performance of the algorithms introduced in Chapter 6. Some of the experiments mentioned here can also be found in the submitted article [50].

The following numerical results were carried out on a computer with 88 Intel Xenon ES-2699 cores (2.2 Ghz each) and a total of 0.792 TB RAM. The algorithms were implemented in MatlabR2017a, the SOCPs in Algorithms 1 and 4 were solved using Yalmip R20170626 and SDPT3 4.0.

The experiments were carried out as follows: If A is of full rank, then we use the Cholesky factorization as the initial factorization $\widetilde{B}$; otherwise, we compute $\widetilde{B}$ via the eigendecompostion as in Section 3.2. For the given value $r \geq \operatorname{cpr}(A)$, we generate from $\widetilde{B}$ a matrix $B \in \mathbb{R}^{n\times r}$ with $A = BB^T$ by column replication as described in Lemma 3.23. We use column replication throughout, only in Section 7.6 we also use appending zero columns as described in Section 3.6 for comparison.

We produce the random starting point Q_0 by generating a random $r \times r$-matrix M using the Matlab command `randn` and then setting $Q_0 \leftarrow P_{\mathcal{O}_r}(M)$. Algorithm 1 (resp. Algorithm 2) terminates successfully at iteration k if $BQ_k \geq -10^{-15}$, it terminates unsuccessfully if a maximum number of iterations (usually 5000) is reached. Since not all starting points lead to a successful termination, we repeat the experiment for a number of different starting points (usually 100 starting points).

7.1 A Specifically Structured Example in Different Dimensions

First of all, we will show that Algorithm 2 terminates successfully in different dimensions. To this end, consider the following example, cf. [85, Example 7.4].

Example 7.1. *Let e_n denote the all-ones-vector in $\mathbb{R}^n$ and consider the matrix*

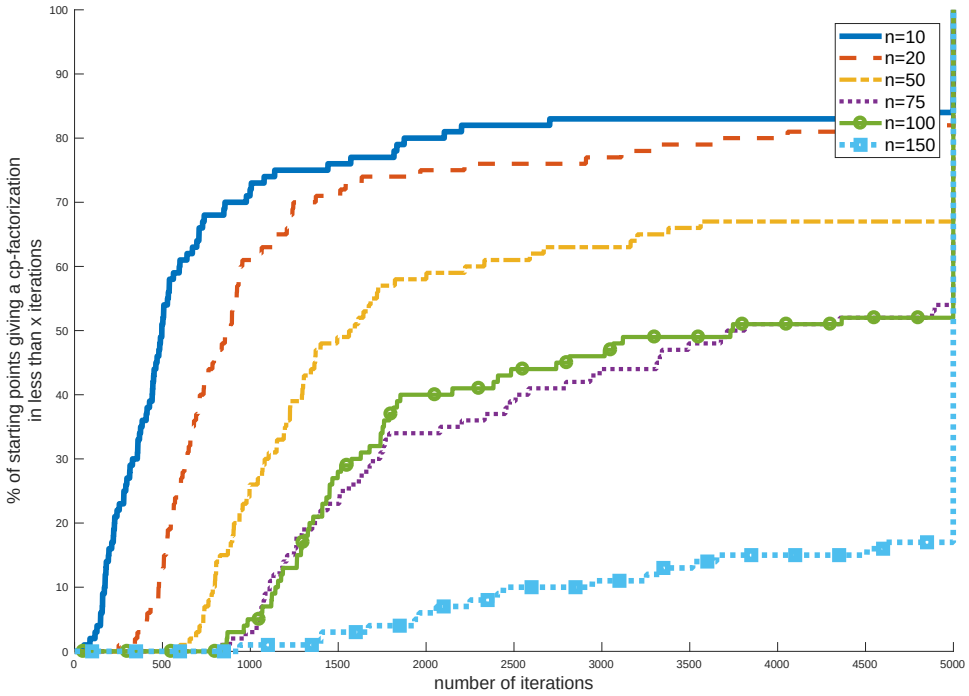
$$A_n := \begin{pmatrix} 0 & e_{n-1}^T \\ e_{n-1} & I_{n-1} \end{pmatrix}^T \begin{pmatrix} 0 & e_{n-1}^T \\ e_{n-1} & I_{n-1} \end{pmatrix} \in \mathbb{R}^{n\times n}.$$

It has been shown in [85, Example 7.4] that $A_n \in \operatorname{int}(\mathcal{CP}_n)$ for every $n \geq 2$. By construction, it is clear that $\operatorname{cpr}(A_n) = n$. Nevertheless, the given factorization does not verify the membership to the interior of the completely positive cone since every column has at least one zero entry.

For the first experiment, we will try to factorize A_n for the values $n \in \{10, 20, 50, 75, 100, 150\}$. In addition, since the exact cp-rank is known, we use $r = \operatorname{cpr}(A_n) = n$ and a maximum of 5000 iterations per starting point. For each A_n, we tested 100 starting points and plotted the percentage

of successfully terminating starting points depending on the number of iterations. The results are illustrated in Figure 7.1.

Figure 7.1: Success rate of Algorithm 2 for Example 7.1 with different values of n.



On the abscissa, we find the number of iterations upper bounded by 5000, the maximum number of iterations. On the axis of ordinates, we see the percentage of starting points out of the 100 starting points providing a cp-factorization in less then x iterations. First, we notice in Figure 7.1 that Algorithm 2 succeeds in factorizing A_n in all cases. However, the percentage of successful starting points varies. Let us have a closer look at the blue line, representing $n = 10$, and at the dotted purple line, representing $n = 75$, in comparison. Here we see that after 500 iterations approximately 65% of the starting points already provided a cp-factorization for $n = 10$, whereas for $n = 75$ the algorithm did not terminate successfully so far. After 3000 iterations conversely, the algorithm terminates successfully for approximately 45% of the starting points for $n = 75$. For $n = 10$ more than 80% of the starting points led to a cp-factorization. These values will not increase, given an increasing number of iterations. Thus, in total we notice that for $n = 10$ more than 80% of the starting points, and for $n = 75$ more than 50% of the starting points returned a cp-factorization of A in less than 5000 iterations. Moreover, the percentage of successful starting points decreases for increasing n. Here we should keep in mind that we used the exact cp-rank information, and not the upper bound cp_n for the cp-rank, for the number of columns in our initial factorization.

This example therefore verifies the successful termination of Algorithm 2. A concrete cp-factorization for $n = 10$, returned by the algorithm, is $A_{10} = CC^T$, where

$$C = \begin{pmatrix} 0.5716 & 0.4287 & 0.0532 & 0.0000 & 0.5069 & 2.8415 & 0.1569 & 0.3434 & 0.1138 & 0.0000 \\ 0.4191 & 0.3664 & 0.0544 & 0.1965 & 0.5360 & 0.0361 & 1.1595 & 0.1242 & 0.0149 & 0.0000 \\ 0.0000 & 0.1171 & 0.1892 & 0.8072 & 0.3366 & 0.1984 & 0.4776 & 0.3755 & 0.0137 & 0.8815 \\ 0.3279 & 0.3561 & 1.1101 & 0.1012 & 0.4358 & 0.0637 & 0.3159 & 0.3811 & 0.1635 & 0.2394 \\ 1.0491 & 0.1137 & 0.0821 & 0.0000 & 0.5241 & 0.0000 & 0.1859 & 0.1382 & 0.0436 & 0.7413 \\ 0.0000 & 0.3273 & 0.0195 & 0.0000 & 1.2030 & 0.0101 & 0.1399 & 0.5775 & 0.0000 & 0.3034 \\ 0.2479 & 1.1492 & 0.0489 & 0.0000 & 0.1329 & 0.0353 & 0.2987 & 0.4310 & 0.0034 & 0.5671 \\ 0.5300 & 0.6589 & 0.1273 & 0.8149 & 0.6319 & 0.0077 & 0.0169 & 0.0333 & 0.4516 & 0.0000 \\ 0.6267 & 0.2148 & 0.0616 & 0.4197 & 0.1999 & 0.0046 & 0.2810 & 1.1234 & 0.0177 & 0.0000 \\ 0.2355 & 0.2335 & 0.0546 & 0.0000 & 0.3507 & 0.0762 & 0.4681 & 0.5151 & 1.0344 & 0.4516 \end{pmatrix}.$$

Since all the entries are nonnegative, this is clearly a cp-factorization of A_{10}. Moreover, since for example the second columns is even entrywise strictly positive and the matrix A_{10} is of full rank, this factorization proves $A_{10} \in \operatorname{int}(\mathcal{CP}_{10})$, based on the results in Theorem 2.19. Here the given factorization in Example 7.1 does not provide a certificate for $A_{10} \in \operatorname{int}(\mathcal{CP}_{10})$. So even without applying Algorithm 5, it may happen that the resulting decomposition proves the membership to the interior of the completely positive cone.

For this experiment, the value r was equal to the cp-rank of the input matrix A_n for every n. So what happens if we allow more than $\operatorname{cpr}(A_n)$ columns in our initial factorization? This question will be answered in the following section.

7.2 The Influence of the Parameter r

For the Algorithms 1 and 2, we need an input parameter $r \geq \operatorname{cpr}(A)$ and for the Algorithms 4 and 5 an input parameter $r \geq \operatorname{cpr}^+(A)$. However, for a general input matrix $A \in \mathbb{R}^{n\times n}$, it is usually impossible to compute $\operatorname{cpr}(A)$ or $\operatorname{cpr}^+(A)$. Therefore, it becomes necessary to use the bounds cp_n or cp_n^+ from Lemma 2.32, i.e., to use $r = \operatorname{cp}_n \geq \operatorname{cpr}(A)$ or $r = \operatorname{cp}_n^+ \geq \operatorname{cpr}^+(A)$. In the following experiment, we will see the influence of the parameter r. We therefore fixed the parameter n in Example 7.1 and considered increasing values of $r \geq n$. Figure 7.2 shows the performance of Algorithm 2 in this setting. Here $n = 6$ was used with a maximum number of 5000 iterations per starting point and a total of 1000 starting points for each value of r.

Figure 7.2 shows that the algorithm produces a cp-factorization of A_6 for each value of r. Note, however, that the percentage of successful starting points increases for increasing r such that for $r \geq 15$ nearly every starting point gives a cp-factorization of A_6 in less than 750 iterations. Considering $r = 6$, we notice that even after 5000 iterations the algorithm only terminates successfully for around 75% of the starting points. In addition, Figure 7.2 indicates that an increasing value of r leads to a decreasing average number of iterations until termination. If we compare the yellow dashed graph, representing $r = 12$, and the green graph, representing $r = 18$, we can see a clear shift to the left for increasing r, showing a decreasing average number of iterations until termination, even if the success rate is similar.

A similar picture can be obtained for the matrix A_n from Example 7.1 for other values of n. For this, see for example Figure 7.3, where we fixed the value $n = 10$ and considered again several values of $r \geq \operatorname{cpr}(A_{10})$.

Figure 7.2: Success rate of Algorithm 2 for the matrix A_6 from Example 7.1 using different values of $r \geq \operatorname{cpr}(A_6) = 6$.

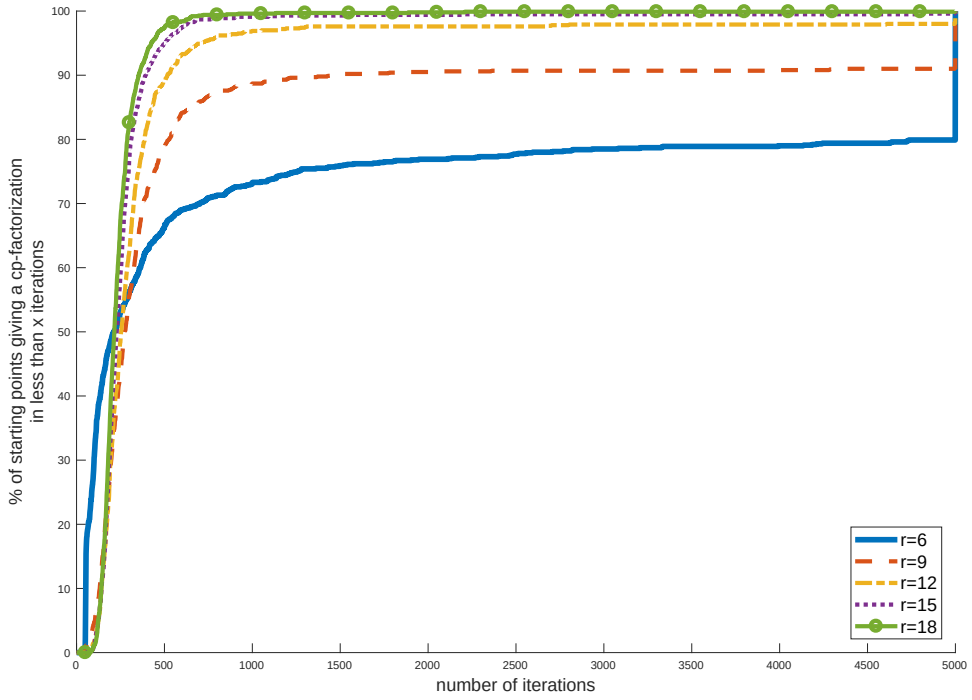
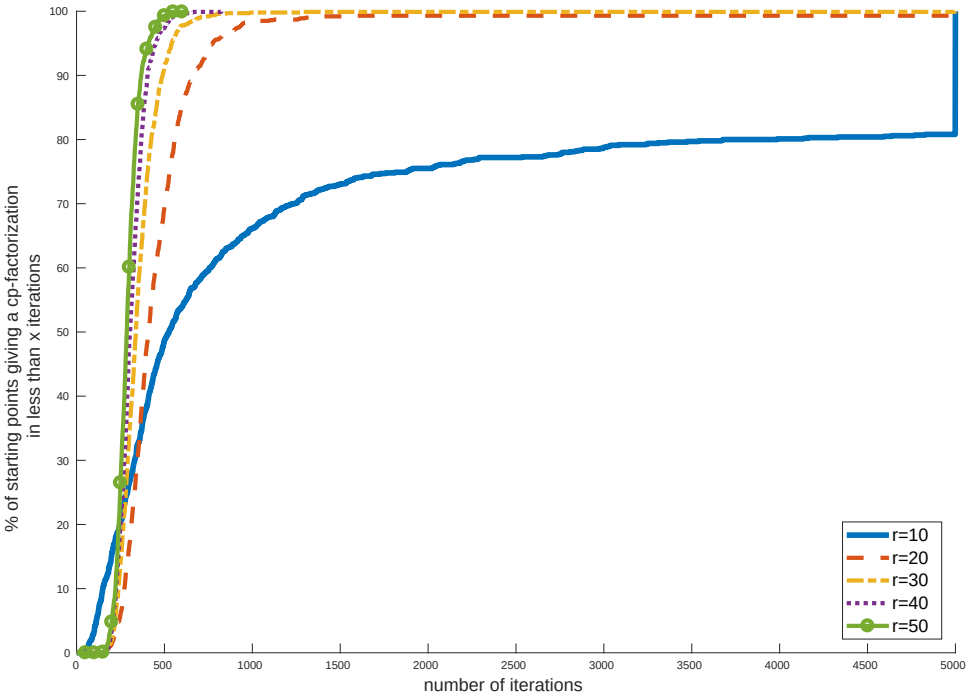


Figure 7.3: Success rate of Algorithm 2 for the matrix A_{10} from Example 7.1 using different values of $r \geq \operatorname{cpr}(A_{10}) = 10$.



Here we can also observe the following: Increasing the parameter r up to 40 gives a success rate of 100%, such that any of the 1000 starting points returned a cp-factorization of A_{10}. In conclusion, we see that it can make sense to increase the value of r, even though the exact cp-rank is known, such that more starting points will yield a successful termination. Of course, this results in a trade off between the increasing order r, leading to higher computation times, and an increasing success rate for the starting points. Hence, for numerical applications, this gives rise to the following suggestion:

Remark 7.2. *As a recommended recipe for the choice of the parameter r, we start the algorithm with $r = n$ and increase r gradually up to cp_n, stopping whenever the algorithm successfully outputs a factorization. The reason behind this recommendation is that obviously iterations are faster for smaller values of r (recall that the algorithm works with matrices in $\mathbb{R}^{r\times r}$), cf. also Table 7.2 in Section 7.9 for this observation. Moreover, for small n, the bound $\mathrm{cp}_n \approx n$, whereas for large n, we have $\mathrm{cp}_n \gg n$. Numerical evidence shows that for most matrices $\mathrm{cpr}(A) \ll \mathrm{cp}_n$, so setting $r = \mathrm{cp}_n$ results in unnecessarily long computation times.*

The matrices in Example 7.1 are of full rank and therefore the cp-rank can not be smaller than n due to Lemma 2.24 (b). In the following section, we will therefore consider a matrix which is not of full rank and is therefore not an element of the interior of the completely positive cone.

7.3 A Low cp-rank Matrix Without Known Factorization

Hitherto, we considered matrices for which a cp-factorization is already known. In this section, we consider a matrix which is known to be completely positive, but for which no factorization is known.

Example 7.3. *Consider the following matrix from [88, Example 2.7]:*

$$A = \begin{pmatrix} 41 & 43 & 80 & 56 & 50 \\ 43 & 62 & 89 & 78 & 51 \\ 80 & 89 & 162 & 120 & 93 \\ 56 & 78 & 120 & 104 & 62 \\ 50 & 51 & 93 & 62 & 65 \end{pmatrix}$$

According to the sufficient condition from [88, Theorem 2.5] (see also Theorem 2.13), this matrix is completely positive with $\mathrm{cpr}(A) = \mathrm{rank}(A) = 3$.

We try to factorize this matrix using Algorithm 2 with $r = 5$. We use the 5×5 eigenvalue decomposition of A which gives the initial factorization matrix

$$B = \begin{pmatrix} 0.0000 & 0.0000 & 0.1341 & -1.5233 & 6.2178 \\ 0.0000 & 0.0000 & -1.6716 & 1.9775 & 7.4361 \\ 0.0000 & 0.0000 & 1.5900 & -0.9900 & 12.5893 \\ -0.0000 & -0.0000 & 0.2307 & 3.0140 & 9.7398 \\ 0.0000 & -0.0000 & -1.4547 & -3.0168 & 7.3337 \end{pmatrix},$$

where the absolute values of all the ± 0.0000 entries are less than or equal to 10^{-7}. Due to the fact that $\mathrm{rank}(A) = 3$, we only have three relevant columns. Using the randomly generated starting

matrix

$$Q_0 = \begin{pmatrix} -0.1279 & -0.2308 & 0.7810 & -0.0522 & -0.5636 \\ 0.7308 & -0.6428 & -0.0056 & -0.2022 & 0.1084 \\ -0.0936 & -0.3503 & 0.1076 & 0.8965 & 0.2308 \\ 0.3208 & 0.1351 & -0.4341 & 0.3342 & -0.7607 \\ 0.5812 & 0.6265 & 0.4358 & 0.2026 & 0.1967 \end{pmatrix}, \tag{57}$$

Algorithm 2 provides the following cp-factorization in 0.019 seconds and after 262 iterations, again with a precision of 10^{-7}:

$$A = \tilde{B}\tilde{B}^T, \text{ with } \tilde{B} = \begin{pmatrix} 3.1801 & 3.0200 & 4.6654 & 0.0000 & 0.0000 \\ 5.5713 & 5.1616 & 2.0779 & 0.0000 & 0.0000 \\ 8.2927 & 4.9557 & 8.2869 & 0.0000 & 0.0000 \\ 8.4857 & 4.6641 & 3.1999 & 0.0000 & 0.0000 \\ 2.3517 & 4.9677 & 5.8984 & 0.0000 & 0.0000 \end{pmatrix}. \tag{58}$$

Since two zero columns appear in the matrix B, this factorization confirms that $\operatorname{cpr}(A) = 3$ and gives an explicit cp-factorization of A for the first time.

Since A is not of full rank, $A \notin \operatorname{int}(\mathcal{CP}_5)$. So this example shows that Algorithm 2 can also factorize matrices on the boundary of the completely positive cone. We will compare the performance of Algorithm 2 in the interior and on the boundary of $\mathcal{CP}_n$ in Section 7.7. Before that, we will show that Algorithm 1, again applied to Example 7.3, also returns a cp-factorization.

7.4 A Concrete Example for Algorithm 1

We can also apply Algorithm 1 to Example 7.3 in order to obtain a cp-factorization. In contrast to Algorithm 2 and the results in the previous section, Algorithm 1 provides the following cp-factorization in 3.8 seconds and after 6 iterations:

$$A = \tilde{B}\tilde{B}^T, \text{ with } \tilde{B} = \begin{pmatrix} 1.6834 & 2.2445 & 0.8958 & 2.8246 & 4.9343 \\ 1.5748 & 5.1870 & 2.7833 & 3.9399 & 3.0571 \\ 5.0415 & 5.5538 & 2.2720 & 4.8436 & 8.7816 \\ 4.1271 & 6.6558 & 3.2893 & 3.8871 & 4.0912 \\ 0.3538 & 2.1936 & 0.9657 & 4.2716 & 6.3940 \end{pmatrix}.$$

Here we use the following starting point

$$Q_0 = \begin{pmatrix} 0.2730 & -0.4657 & 0.1679 & -0.1818 & -0.8046 \\ -0.4681 & 0.5075 & -0.1734 & -0.6082 & -0.3513 \\ 0.6708 & 0.0676 & 0.0054 & -0.6566 & 0.3380 \\ 0.3533 & 0.6915 & 0.4903 & 0.3086 & -0.2478 \\ 0.3628 & 0.2069 & -0.8374 & 0.2659 & -0.2315 \end{pmatrix}.$$

To show that Algorithm 1 provides a different result compared to Algorithm 2 for the same starting point, we test Algorithm 1 also for the starting point Q_0 in equation (57). Here after the very first iteration and 1.81 seconds Algorithm 1 returns

$$A = \tilde{B}\tilde{B}^T, \text{ with } \tilde{B} = \begin{pmatrix} 4.1600 & 4.1090 & 2.0356 & 1.3517 & 0.9161 \\ 3.8291 & 2.7471 & 4.4371 & 2.4489 & 3.7559 \\ 7.9270 & 7.9547 & 4.7735 & 1.0762 & 3.4555 \\ 5.0406 & 4.2915 & 5.5257 & 0.9113 & 5.3677 \\ 5.1462 & 4.7580 & 2.1849 & 3.3308 & 0.0992 \end{pmatrix}. \tag{59}$$

Comparing the factorizations in equations (58) and (59), we can see that both algorithms provide different factorizations for the same starting point. Moreover, Algorithm 2 takes only 1% of the computation time of Algorithm 1 for this starting value, even though more iterations are necessary to obtain a cp-factorization. In addition, the factorization in equation (59) does not prove the equation $\operatorname{cpr}(A) = 3$, in contrast to equation (58). Hence, Example 7.3 gives a first hint on the differences in the results of both algorithms. In the following section, we will therefore have a closer look at the performance of Algorithms 1 and 2 in comparison.

7.5 Algorithms 1 and 2 in Comparison

The next experiment is again based on Example 7.3 and compares the performance of Algorithms 1 and 2. Here we test the same 100 starting points for both algorithms, choosing 500 as a maximal number of iterations per starting point. Even though we already know the cp-rank of the given matrix, we will chose $r = \mathrm{cp}_5 = 12$ to compare the performance of both algorithms in the setting of no further information. The success rate for each algorithm is plotted in Figure 7.4.

Figure 7.4: Success rates of Algorithms 1 and 2 in comparison.

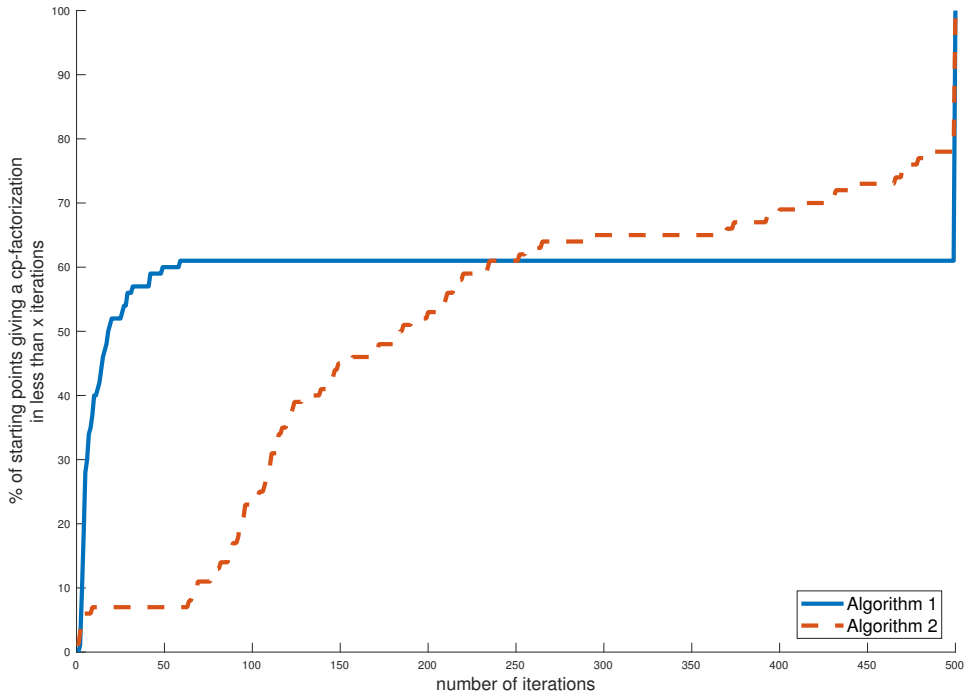


Figure 7.4 shows that Algorithms 1 and 2 only have local convergence, since not every starting point returns a cp-factorization in less than 500 iterations. For Algorithm 1, this substantiates the results in Theorem 6.6. If we look at the success rate after 500 iterations, Algorithm 1 provides a completely positive factorization for 61% of the starting points, whereas Algorithm 2 terminates successfully for 78% of the starting points. In addition, we see that the number of iterations necessary to compute a factorization is lower for Algorithm 1. Especially, if we consider the success rate after 100 iterations, Algorithm 1 already reaches its maximal success rate whereas for Algorithm 2 only around 20% of the starting points return a cp-factorization. However, the iterations of Algorithm 1 are much more expensive: running all 100 starting points in Algorithm 2 takes 4.5 seconds, but 5656 seconds for Algorithm 1.

This shows that Algorithm 2 is much faster in total: although it may need more iterations than Algorithm 1, the numerical cost of a single iteration is much smaller. Moreover, the percentage of successfully terminating starting points is higher for Algorithm 2.

Next, we will take a closer look at the initial factorizations and their influence on the performance of Algorithm 2, i.e. we will compare the performance of Algorithm 2 for the two different possibilities to obtain a suitable initial factorization introduced in Section 3.6.

7.6 Column Replication Versus Appending Zero Columns

In Section 3.6, we mentioned two possible ways of expanding an initial factorization matrix $B \in \mathbb{R}^{n\times n}$ into a matrix with $r \geq \operatorname{cpr}(A)$ columns: either by column replication as described in (24), or by appending zero columns as introduced in (23). To see which approach performs numerically better, we use Example 7.1 for $n = 5$ and $r = \operatorname{cp}_5 = 12$ such that column replication or appending zero columns is necessary for our initial factorization. We use the same 100 starting points for both approaches and a maximum of 500 iterations per starting point. The performance of Algorithm 2 in this setting is illustrated in Figure 7.5.

Figure 7.5: Success rate of Algorithm 2 for column replication versus appending zero columns.

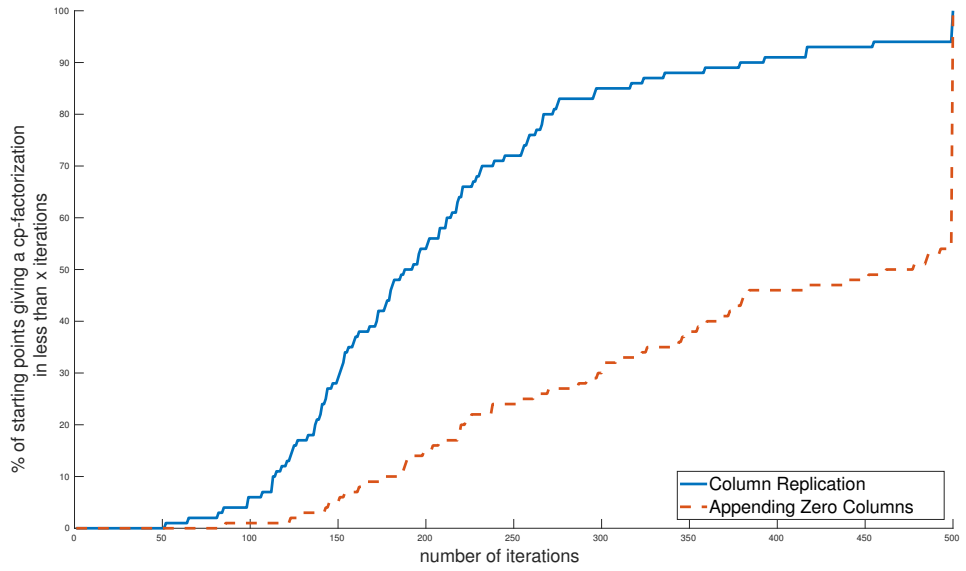


Figure 7.5 shows that when appending zero columns, roughly 55% of the starting points yield a cp-factorization in less than 500 iterations, as opposed to 95% if we use column replication. Focussing on the starting points for which the algorithm terminates successfully, the mean number of iterations needed to provide a cp-factorization is 212 for the column replication approach and 346 if we append zero columns. For those cases where both approaches were successful, column replication was always faster, i.e., terminated after fewer iterations. Moreover, given any number of iterations in between 0 and 500, appending zero columns never returns a higher success rate for this example. These results suggest that the column replication approach is numerically more efficient than appending zero columns.

In the following section, we will have a closer look at the performance of Algorithm 2 at the boundary and in the interior of the completely positive cone.

7.7 Performance of Algorithm 2 on the Boundary and in the Interior of $\mathcal{CP}_n$

In this section, we illustrate how Algorithm 2 behaves for matrices on the boundary of $\mathcal{CP}_n$. We start with an instance where the algorithm fails. Consider again the matrix in Example 2.43.

Example 7.4. *Consider the following matrix taken from [44]:*

$$A_{DS} = \begin{pmatrix} 8 & 5 & 1 & 1 & 5 \\ 5 & 8 & 5 & 1 & 1 \\ 1 & 5 & 8 & 5 & 1 \\ 1 & 1 & 5 & 8 & 5 \\ 5 & 1 & 1 & 5 & 8 \end{pmatrix} \in \mathcal{CP}_5 \setminus \operatorname{int}(\mathcal{CP}_5).$$

Neither Algorithm 1 nor Algorithm 2 succeed in factorizing this matrix. In view of Theorem 6.6, this may seem surprising, however we can suspect that the region of local convergence of Algorithm 1 is so small that numerical precision prevents us from finding a starting point there.

In the following, we will see that for slight perturbations of this matrix, Algorithm 2 becomes successful in finding a factorization. To this end, we investigate convex combinations of A and the following matrix $C \in \operatorname{int}(\mathcal{CP}_5)$:

$$C = MM^T, \quad \text{where } M = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Table 7.1 shows the performance of Algorithm 2 for $A_\lambda := \lambda A_{DS} + (1-\lambda)C$ for different values of $\lambda \in [0,1]$:

λ	time (sec.)	success rate (%)	λ	time (sec.)	success rate (%)
0	3	100	0.8	51	50
0.1	5	99	0.9	72	23
0.2	9	95	0.95	69	26
0.3	15	93	0.97	69	36
0.4	28	82	0.976	71	24
0.5	32	73	0.977	79	5
0.6	34	72	0.9774	80	1
0.7	46	51	0.9775	83	0

Table 7.1: Performance of Algorithm 2 for the matrix A_λ for different values of $\lambda \in [0,1]$ and $r = \mathrm{cp}_5^+ = 12$. For each A_λ, we run Algorithm 2 with 100 starting points and a maximum of 5000 iterations per starting point. Column 2 shows the total computation time of Algorithm 2 for all 100 starting points.

For the matrix C itself, every starting point returns a cp-factorization but as we approach the boundary of $\mathcal{CP}_n$, the success rate decreases and the computing time increases. The latter is because the closer we get to the boundary, the higher the number of starting points for which the algorithms runs for a high number of iterations. Note that only very close to the matrix A_{DS} does the success rate decrease rapidly, and we even find a factorization of the matrix A_λ with $\lambda = 0.97749$.

Nevertheless, Example 7.3 and the numerical results in Section 7.3 prove that Algorithm 2 works well for some matrices at the boundary. In the following, we will consider more difficult instances for which both algorithms struggle to return a cp-factorization.

7.8 Other Difficult Instances

Whereas Algorithm 2 succeeded in factorizing the matrix of low cp-rank from Example 7.3, it failed for all the matrices of high cp-rank introduced in Section 2.5. As shown, these are specifically constructed matrices for which $\mathrm{cpr}(A) \gg \mathrm{rank}(A)$. Algorithm 2 failed for these instances, even by gradually increasing r up to two or three times $\mathrm{cpr}(A)$. So for these artificially generated matrices, Algorithm 2 seems to struggle finding a suitable initial orthogonal matrix, even for higher values of r. Similar observations hold for Algorithm 1 such that both algorithms struggle to find cp-factorizations for these matrices.

Nevertheless, the following example proves that Algorithm 2 can factorize matrices for which the cp-rank is greater than the rank. To this end, let us go back to Example 2.27 and consider the matrix

$$A_{cp4} = \begin{pmatrix} 6 & 3 & 3 & 0 \\ 3 & 5 & 1 & 3 \\ 3 & 1 & 5 & 3 \\ 0 & 3 & 3 & 6 \end{pmatrix}$$

of rank 3 with $\mathrm{cpr}(A_{cp4}) = 4$. Using the exact cp-rank information, we set $r = 4$. This yields the initial factorization $A_{cp4} = BB^T$ with

$$B = \begin{pmatrix} 0 & 0.0000 & -1.7321 & 1.7321 \\ 0 & -1.4142 & -0.0000 & 1.7321 \\ 0 & 1.4142 & 0.0000 & 1.7321 \\ 0 & -0.0000 & 1.7321 & 1.7321 \end{pmatrix}.$$

Then, starting Algorithm 2 with the initial orthogonal matrix

$$Q_0 = \begin{pmatrix} -0.7722 & -0.2153 & -0.5685 & -0.1851 \\ -0.2036 & 0.6411 & -0.1983 & 0.7129 \\ 0.1634 & -0.7227 & -0.1605 & 0.6520 \\ 0.5793 & 0.1423 & -0.7822 & -0.1800 \end{pmatrix}$$

returns the following factorization in 0.0187 seconds and after 314 iterations:

$$A_{cp4} = CC^T, \text{ with } C = \begin{pmatrix} 1.2759 & 2.0910 & 0.0000 & 0.0000 \\ 1.2759 & 0.6562 & 1.7151 & 0.0000 \\ 0.0000 & 1.4347 & 0.0341 & 1.7148 \\ 0.0000 & 0.0000 & 1.7492 & 1.7148 \end{pmatrix}.$$

Since all the entries are nonnegative, this is a cp-factorization of A_{cp4}, proving that Algorithm 2 can factorize matrices of cp-rank greater than the rank.

In addition, we can try to apply Algorithm 2 to recover the stable set of a graph from the solution of problem (12). Unfortunately, this is unsuccessful. The reason is that typically the solution of (12) is a matrix that contains a number of zero entries, which imposes a certain sparsity pattern for the factorization matrix B as well: whenever $A_{ij} = 0$ and $A = BB^T$, then the columns B_i and B_j necessarily have to have disjoint support. Generating an orthogonal matrix that provides this is extremely unlikely, so without further adjustments, Algorithm 2 will typically fail for such instances.

In the following, we will show that the algorithms may fail in these special cases but will terminate successfully for randomly generated examples even of high order.

7.9 Randomly Generated Examples of Higher Order

Next, we investigate randomly generated matrices of higher order to see how the algorithm depends on the order of the input matrix. The instances were generated as follows: First, we generate a random $n \times k$ matrix B using the Matlab command `randn`. Next, we compute C by setting $C_{ij} := |B_{ij}|$ for all i, j, and finally we take $A = CC^T$ as the matrix to be factorized by our algorithm. By construction, we have $A \in \mathcal{CP}_n$ with $\operatorname{cpr}(A) \leq k$. Table 7.2 illustrates the results for $k = 2n$.

n	cp_n^+	r	# of initial Q_0	# of iterations	time (sec.)
35	627	36	58.6	1937	192
50	1272	51	92	1765	504
50	1272	151	1	177	0.9
100	5047	151	1	1365	9.8
100	5047	301	1	183	4
150	11322	201	1.2	2384	42
200	20097	301	1.1	1547	47
1000	500497	1500	1.1	2454	1092
2000	2000997	3000	1.2	2993	9675

Table 7.2: Performance of Algorithm 2 for randomly generated matrices A for several values of n and $k = 2n$. For $n \leq 50$, we used 100 starting points, for $n > 50$, we used 10 starting points. In each case, we used a maximum of 5000 iterations per starting point. The numbers in columns 4-6 represent the average of 100 randomly generated instances.

Table 7.2 shows that for every value of n and $r > k$, Algorithm 2 is successful for all randomly generated instances.

Taking a more detailed look at the case $n = 50$ and especially the case $r = n + 1$, we see that 92 starting points were necessary to obtain a cp-factorization, on average of the 100 runs. Here each run takes more than 500 seconds on average. For the case $r = 3n + 1$ on the other hand, all starting points lead to a successful termination. Moreover, each of the 100 runs takes only one second on average. Table 7.2 shows similar results for $n = 100$. This might be due to the fact that $n + 1$ is not necessarily a sufficient number of columns in our initial factorization, since we just know that $n \leq \operatorname{cpr}(A) \leq k = 2n$. Hence, we conclude that using a higher value of r can lead to immense time savings, although a single iteration will take longer.

In addition, we also tested matrices of high order. As can be seen from Table 7.2, time consumption increases exponentially for increasing n, but even for $n = 2000$, Algorithm 2 provides a factorization. Here every run takes 160 minutes on average.

In addition, we notice that the upper bound cp_n^+ increases quadratically for increasing n, such that, as already mentioned in Section 7.2, it is recommended to start with a value $r \ll \operatorname{cp}_n^+$. But on the other hand, if the value r is chosen too small, the number of columns in the initial factorization might be insufficient, as we can see in Table 7.2 for several values of n. For most instances $r = 1.5n$ was sufficient, still fulfilling $r \ll \operatorname{cp}_n^+$.

In the following section, we will compare the performance of Algorithm 2 with an existing algorithm by Ding et al. introduced in [39].

7.10 Comparison with an Algorithm by Ding et al.

In [39], Ding et al. proposed a simple algorithm for cp-factorizations. Their algorithm works by updating a randomly chosen initial factorization. For the reader's convenience, this method is stated as Algorithm 6.

Algorithm 6 The algorithm by Ding et al. [39]

Input: matrix $A \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, qmax $\in \mathbb{N}$, MaxIter $\in \mathbb{N}$, $\beta \in (0, 1]$

for $q = 1 :$ qmax **do**
  $B \leftarrow \operatorname{randn}(n, r)$
  optional: $B \leftarrow \max(AB(B^TB)^{-1}, 0)$
  while ($\|A - BB^T\|_2 \geq 10^{-12}$ or $\min_{i,j} B_{ij} \leq -10^{-15}$) or $k <$ MaxIter **do**
    for $i = 1 : n$ **do**
      for $j = 1 : r$ **do**
        $B_{ij} \leftarrow B_{ij} \left(1 - \beta + \beta \frac{(AB)_{ij}}{(BB^TB)_{ij}}\right)$
      end for
    end for
  end while
end for

Here we use $\beta = 0.5$ as suggested in [39], qmax = 100 and MaxIter = 5000. For each r, we perform 100 runs of Algorithm 6.

r	av. computation time (sec.)	av. success rate (%)	av. number of iterations
3	4.8	18.5	50
4	5.5	13.5	51
5	5.5	15.6	51
10	5.1	4.5	52
15	5.9	2.4	51

Table 7.3: Performance of Algorithm 6 applied to Example 2.22.

Table 7.3 shows the performance of this method for the matrix

$$A := \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix}$$

as in Example 2.22. Note that this matrix has cp-rank 3 due to Remark 2.36, so setting $r = 3$ in Algorithm 6 would be optimal. The numbers in the table represent averages over the 100 runs: average computation time for 100 initial values, average percentage of successful starting points, and the average number of iterations if a successful initial matrix B is chosen.

Table 7.3 indicates that both the computation time and the average number of iterations are independent of r, however the success rate is very sensitive to this parameter.

To compare these results with the performance of Algorithm 2, we run the same experiment for Algorithm 2 with 100 starting points and a maximum of 5000 iterations per starting point and values $r \in \{3, 4, 5, 10, 15\}$, again applied to the matrix in Example 2.22. The results are summed up in the following table. The numbers in the table represent averages over the 100 runs: average computation time for 100 initial values, average percentage of successful starting points, and the average number of iterations if a successful initial matrix B is chosen.

r	av. computation time (sec.)	av. success rate (%)	av. number of iterations
3	4.4	59.5	51
4	2.8	79.7	51
5	2.1	88	50
10	0.8	99.2	50
15	1.8	99.9	50

Table 7.4: Performance of Algorithm 2 applied to Example 2.22.

Here we notice that the average number of iterations is again independent of r and is similar to results for Algorithm 6. But on the other hand, the average computation time is decreasing for increasing values of r up to 10, in contrast to the stable computation time measured for Algorithm 6. This is due to the fact that the success rate is increasing until it reaches nearly 100%. Increasing the number of columns r in our initial factorization to more than 10 still increases the already very high success rate on average, but the drawback of the increased order of the matrices is bigger such that in total the average computation time increases again. Overall, we again get the result that

a higher value of r leads to a higher success rate, indicating that the success rate of Algorithm 2 does not drop for higher values of r, in contrast to the success rate of Algorithm 6.

The observation that Algorithm 6 is sensitive to r is even more striking if we apply Algorithm 6 to Example 7.3, a 5×5 matrix A with $\operatorname{cpr}(A) = \operatorname{rank}(A) = 3$, as can be seen in Table 7.5: In that example, the algorithm terminates only if $r = \operatorname{cpr}(A)$ is used, otherwise it fails completely.

r	av. computation time (sec.)	av. success rate (%)
3	8.5	4.5
5	9.7	0
10	9.3	0
15	9.2	0

Table 7.5: Performance of Algorithm 6 by Ding et al. with $\beta = 0.5$, qmax = 100 and MaxIter = 5000, applied to Example 7.3 and using different values of r. The numbers are averages of 100 runs per value of r.

Again we compare the results in Table 7.5 to the performance of Algorithm 2 in this setting. We test 100 starting points and a maximum of 5000 iterations per starting point for different values of $r \geq n$. Similar to the previous tables, the numbers in Table 7.6 represent the average of 100 runs.

r	av. computation time (sec.)	av. success rate (%)	av. number of iterations
5	7.1	60.9	50
10	4.5	95.5	50
15	8.7	97.9	51

Table 7.6: Performance of Algorithm 2 applied to Example 7.3.

Here we see that for this example we can again obtain better results with Algorithm 2. The average number of iterations is still constant compared to the results for the matrix in Example 2.22. Even for $r = 5$, we already have a success rate of more than 60% for Algorithm 2, whereas Algorithm 6 fails completely. Again increasing the value r results in an increasing success rate on average. For $r \geq 10$ more than 95% of the starting points returned a cp-factorization.

Remark 7.5. *To use $r < n$ for our algorithms, we need further adjustments. These adjustments are introduced in Chapter 8 as a method to derive nonnegative matrix factorizations for a given matrix.*

In addition, we test the performance of Algorithm 1 under the same conditions but with a maximum number of 500 iterations per starting point. The results can be found in Table 7.7 and represent the average of 100 runs.

r	av. computation time (sec.)	av. success rate (%)	av. number of iterations
5	5275	90.1	50

Table 7.7: Performance of Algorithm 1 applied to Example 7.3.

This proves that applying Algorithm 1 to Example 7.3 with an initial factorization of order 5×5 increases the success rate up to more than 90% on average, albeit with an increased computation time. So, in total, we see that Algorithm 6 fails completely for $r \geq 5$ whereas Algorithms 1 and 2 have a success rate of up to 98% in this setting.

More experiments with randomly generated matrices also showed that the algorithm of Ding et al. is highly sensitive with respect to the parameter r and works badly if an inappropriate r is chosen. Since in general, the cp-rank is not known a priori, this strong sensitivity with respect to r must be considered a huge drawback. This is a disadvantage of Algorithm 6 which Algorithm 2 does not exhibit, and hence we consider Algorithm 2 more stable and robust with respect to the algorithm parameters.

In the following, we will will compare the performance of Algorithm 2 to a method introduced by Jarre and Schmallowsky, cf. [62].

7.11 Comparison with a Method by Jarre and Schmallowsky

As already mentioned in Section 3.1, Jarre and Schmallowsky [62] introduced a method to obtain a certificate for a given matrix to be completely positive. Their method is based on an augmented primal dual method (cf. [61]) and aims to solve a certain second order cone problem, where it is necessary to solve Lyapunov equations to obtain a cp-factorization. For more details on this approach, the reader is referred to [62, Section 2].

In the following, we will compare the performance of this approach to the performance of Algorithm 2. As our first experiment, we consider again the matrix A_{cp4} as introduced in Example 2.27:

$$A_{cp4} = \begin{pmatrix} 6 & 3 & 3 & 0 \\ 3 & 5 & 1 & 3 \\ 3 & 1 & 5 & 3 \\ 0 & 3 & 3 & 6 \end{pmatrix}.$$

As shown in Section 7.8, Algorithm 2 returns a cp-factorization for this matrix in 0.0187 seconds for the considered starting point. For the approach by Jarre and Schmallowsky on the other hand, the user has to choose the number of columns for the factorization, such that we used the smallest possible value (5 in this case) greater than or equal to 4, the cp-rank of A_{cp4}. Then their approach was not able to return a cp-factorization for A_{cp4} of order 4×5. Instead, the approach returns a the matrix

$$B = \begin{pmatrix} 1.0383 & 0.3235 & 0.3234 & 0.0000 & 2.1753 \\ 0.4283 & 1.8625 & 0.0005 & 0.6358 & 0.9803 \\ 0.4283 & 0.0005 & 1.8625 & 0.6358 & 0.9803 \\ 0.0006 & 0.8341 & 0.8341 & 2.1598 & 0.0043 \end{pmatrix} \in \mathbb{R}^{4\times 5}$$

with $\|A - BB^T\|_F = 1.1674$. Hence, the approach provides a completely positive approximation to A instead of a cp-factorization. Moreover, this observation extends to the case where we allow more than 5 columns for the factorization. To this end, we analyse the quality of the approximation

of the approach by Jarre and Schmallowsky by estimating $\|A_{cp4} - BB^T\|_F$ with $B \in \mathbb{R}^{4\times r}$ for several values of r. The results are summed up in the following table.

r	$\|A_{cp4} - BB^T\|_F$
5	1.1674
6	1.3903
7	0.3723
8	0.3723
9	0.1262
10	0.4633

Table 7.8: Quality of the approximation of the approach by Jarre and Schmallowsky applied to Example 2.27.

As can be seen from Table 7.8, the quality of the approximation is sensitive to the choice of r. Nevertheless, the returned factorizations never provide a cp-factorization of A_{cp4}, whereas Algorithm 2 returned a cp-factorization $A_{cp4} = CC^T$ with $\|A - CC^T\|_F \leq 10^{-14}$. This proves that the method by Jarre and Schmallowsky clearly returns an approximation $A \approx BB^T$ in general instead of a cp-factorization $A = BB^T$ with (numerically) strict equality, like Algorithm 2 does.

Nevertheless, for some instances, the quality of the approximation of the approach by Jarre and Schmallowsky and of Algorithm 2 are comparable and both approaches return a cp-factorization. To see this, we consider randomly generated instances. With the same technique as introduced in Section 7.9 and given $n \in \mathbb{N}$, we generate a random $n \times 2n$ matrix B using the Matlab command `randn`. Next, we compute C by setting $C_{ij} := |B_{ij}|$ for all i, j, and finally we take $A = CC^T$ as the matrix to be factorized by both approaches.

If we apply this approach to $n = 7$, we obtain the following matrix as our next example:

$$A = \begin{pmatrix} 13.7162 & 8.1090 & 10.0752 & 7.3940 & 10.7332 & 4.2551 & 9.8380 \\ 8.1090 & 9.9804 & 6.7089 & 6.1420 & 8.3044 & 3.9286 & 6.7772 \\ 10.0752 & 6.7089 & 9.6070 & 5.6133 & 8.3782 & 3.6309 & 6.6191 \\ 7.3940 & 6.1420 & 5.6133 & 10.2718 & 8.1573 & 4.3024 & 5.9480 \\ 10.7332 & 8.3044 & 8.3782 & 8.1573 & 12.8697 & 4.1992 & 9.1688 \\ 4.2551 & 3.9286 & 3.6309 & 4.3024 & 4.1992 & 2.5807 & 3.3640 \\ 9.8380 & 6.7772 & 6.6191 & 5.9480 & 9.1688 & 3.3640 & 13.1836 \end{pmatrix}.$$

For $r = 10$, the approach by Jarre and Schmallowsky returns the following cp-factorization in 0.4077 seconds: $A = BB^T$, with

$$B = \begin{pmatrix} 1.7734 & 0.1505 & 0.6719 & 0.1970 & 0.2793 & 0.2526 & 0.1826 & 2.8187 & 1.0669 & 0.8942 \\ 0.0080 & 1.3661 & 0.0000 & 0.3043 & 0.1793 & 0.2435 & 0.3595 & 1.6954 & 1.0568 & 1.9518 \\ 0.6721 & 0.0432 & 1.7315 & 0.0829 & 0.2259 & 0.2708 & 0.0121 & 1.9629 & 0.984 & 1.0967 \\ 0.0763 & 0.3062 & 0.0000 & 1.3124 & 0.1996 & 0.2576 & 0.2959 & 1.4663 & 2.471 & 0.0110 \\ 0.2570 & 0.2373 & 0.2292 & 0.2489 & 1.8249 & 0.1256 & 0.2112 & 2.5483 & 1.4300 & 0.8387 \\ 0.0290 & 0.1997 & 0.1637 & 0.2194 & 0.0000 & 0.7260 & 0.2026 & 0.8241 & 1.0080 & 0.4491 \\ 0.0995 & 0.3678 & 0.0000 & 0.3108 & 0.1788 & 0.2680 & 1.5625 & 3.224 & 0.0509 & 0.0002 \end{pmatrix}$$

and $\|A - BB^T\|_F \leq 10^{-14}$.

For the same matrix A and $r = 10$, Algorithm 2 returns the following cp-factorization in 0.0381 seconds: $A = CC^T$, with

$$C = \begin{pmatrix} 0.3444 & 2.1460 & 0.2384 & 0.5727 & 1.9311 & 0.1196 & 0.0503 & 2.0715 & 0.7345 & 0.1752 \\ 1.0831 & 2.1524 & 1.6663 & 0.7409 & 0.8276 & 0.2355 & 0.0570 & 0.3151 & 0.0000 & 0.0761 \\ 0.1144 & 1.8852 & 0.0000 & 1.1551 & 1.1350 & 1.1921 & 0.0342 & 1.4079 & 0.0852 & 0.0764 \\ 0.0025 & 2.8284 & 0.0000 & 0.0000 & 0.0000 & 0.0047 & 0.0000 & 0.0907 & 1.4773 & 0.2854 \\ 1.5264 & 2.2389 & 0.0000 & 0.0000 & 1.6931 & 0.9844 & 0.0212 & 0.5427 & 1.1283 & 0.3514 \\ 0.4031 & 1.4953 & 0.0000 & 0.2069 & 0.0000 & 0.0000 & 0.0048 & 0.3708 & 0.0184 & 0.0397 \\ 1.9081 & 1.0882 & 0.5006 & 0.7516 & 0.3537 & 0.0000 & 0.0654 & 2.0581 & 1.7313 & 0.4246 \end{pmatrix}$$

and $\|A - CC^T\|_F \leq 10^{-14}$. Thus, to achieve a comparable quality of approximation like Algorithm 2, the method by Jarre and Schmallowsky takes more time for this concrete example. Here the exact factor is 10.7.

Moreover, we can apply the approach by Jarre and Schmallowsky to randomly generated matrices for higher values of n and different choices of r, in order to compare the performance of this approach to the performance of Algorithm 2 in this setting, which can be found in Table 7.2. The results for the approach by Jarre and Schmallowsky in this setting are collected in Table 7.9.

n	r	time (sec.)
35	36	5.16
50	51	17.12
50	151	$28,84$
100	151	53.79
100	301	88.66
200	301	198.59

Table 7.9: Performance of the approach by Jarre and Schmallowsky for randomly generated matrices A for several values of n and $k = 2n$. The numbers in column 3 represent the average of 100 randomly generated instances.

If we compare the results in Table 7.2 and Table 7.9, we notice that especially in smaller dimensions and for r chosen close to n, the approach by Jarre and Schmallowsky is less time consuming since the algorithm does not enforce (numerically) strict equality for the factorization $A = BB^T$. Considering higher dimensions and especially the cases where r is chosen more distant to n, the method of Jarre and Schmallowsky starts struggling with the increasing order, whereas Algorithm 2 does not exhibit this drawback in general. In contrast to the method by Jarre and Schmallowsky, Algorithm 2 is less time consuming if, for a given order n, we choose a larger value for r. More precisely, if we consider the case $n = 100$ and $r = 301$, Algorithm 2 provides a cp-factorization in 4 seconds on average of the 100 randomly generated instances, whereas the method by Jarre and Schmallowsky takes more than 88 seconds on average to return a cp-factorization. If we take a detailed look at the case $n = 200$ and $r = 301$, the method by Jarre and Schmallowsky takes more than 198 seconds on average of 100 randomly generated instances, whereas Algorithm 2 takes only 42 seconds on average. For these concrete examples, both approaches returned a cp-factorization in any case.

In addition, we applied the method by Jarre and Schmallowsky to randomly generated instances of order 1000, where we chose $r = 1500$. In this setting, the computation time exceeded 5677 seconds on average of 15 randomly generated instances to reach a matrix B such that $\|A - BB^T\|_F \leq 10^{-10}$. Therefore, Algorithm 2 is again faster on average, as Table 7.2 shows.

The numerical results in this section substantiate that especially in higher dimensions Algorithm 2 is less time consuming and, if terminating, always returns a cp-factorization, even for the matrices at the boundary of the completely positive cone. Moreover, Algorithm 2 is less time consuming in case the exact cp-rank of the given matrix is unknown (which is in general always the case) such that we need to use a reasonable upper bound on the cp-rank for the choice of r.

7.12 A Real Life Application in Statistics

As mentioned in Section 2.7, completely positive programming can be used in statistics and more precisely in the area of multivariate extremes, cf. [27]. Here we consider a pairwise dependence matrix Σ, containing the tail dependence of a multivariate regularly-varying random vector. This matrix Σ can be shown to be completely positive, such that we can try to factorize such a matrix Σ. For the following experiments, Cooley and Thibaud provided different matrices Σ, which were obtained directly from real data.

For the first experiment in this setting, consider the matrix

$$\Sigma = \begin{pmatrix} 6.1875 & 5.8750 & 5.8750 & 5.2500 & 4.3125 \\ 5.8750 & 10.3125 & 7.8750 & 8.2500 & 8.0000 \\ 5.8750 & 7.8750 & 12.6250 & 9.6250 & 8.7500 \\ 5.2500 & 8.2500 & 9.6250 & 11.5625 & 6.3125 \\ 4.3125 & 8.0000 & 8.7500 & 6.3125 & 8.8125 \end{pmatrix}.$$

For this matrix, Algorithm 2 returns the following factorizations $A = B_i B_i^T$ for $i \in \{1, 2, 3\}$, with

$$B_1 = \begin{pmatrix} 0.6588 & 0.5438 & 1.7947 & 1.4781 & 0.2277 \\ 2.1293 & 0.5364 & 0.6916 & 1.7792 & 1.3591 \\ 0.8345 & 2.5656 & 1.2861 & 0.8303 & 1.7330 \\ 0.4916 & 1.8741 & 0.0000 & 2.4309 & 1.3781 \\ 2.1843 & 1.5622 & 0.6822 & 0.3881 & 0.9924 \end{pmatrix} \in \mathbb{R}^{5\times 5}, \quad B_2 = \begin{pmatrix} 0.0077 & 2.1630 & 0.8771 & 0.8095 & 0.2904 \\ 1.3422 & 1.2768 & 1.7145 & 1.5153 & 1.2826 \\ 0.0000 & 1.4213 & 0.0000 & 3.0627 & 1.1066 \\ 0.0818 & 0.6542 & 1.7515 & 2.8390 & 0.0000 \\ 1.5820 & 1.1039 & 0.0193 & 1.9116 & 1.1986 \end{pmatrix} \in \mathbb{R}^{5\times 5},$$

$$B_3 = \begin{pmatrix} 0.4350 & 1.5167 & 1.1971 & 0.0000 & 1.0013 & 1.0541 & 0.2593 & 0.2899 \\ 1.2273 & 0.7209 & 1.9365 & 0.0000 & 0.0000 & 1.3857 & 0.0000 & 1.6175 \\ 2.0036 & 0.5622 & 1.4211 & 1.7310 & 0.6016 & 1.5892 & 0.6244 & 0.0350 \\ 2.3119 & 0.4022 & 2.0777 & 0.0000 & 0.1405 & 0.6385 & 1.1375 & 0.1325 \\ 0.7529 & 0.1023 & 1.6964 & 1.4998 & 0.0715 & 1.3171 & 0.0000 & 1.1697 \end{pmatrix} \in \mathbb{R}^{5\times 8}$$

and several other matrices B of order 5×5 or 5×8, factorizing Σ.

Having these factorizations, Cooley and Thibaud (cf. [27]) were able to analyse the probability of two sets, associated with the factorizations of Σ, of being in an extreme set. In general, the cp-factorization can be used to estimate probabilities of extreme events or to simulate realizations with pairwise dependence, summarized by Σ.

Moreover, we applied Algorithm 2 to derive cp-factorizations for a certain matrix $\Sigma \in \mathbb{R}^{44\times 44}$. For this matrix, only 52 columns for the initial factorization were necessary to ensure the convergence of Algorithm 2 to a cp-factorization. The fact that we require only 52 columns, which shows that $\operatorname{cpr}(\Sigma) \leq 52$, was surprising for the application in statistics and hence allowed a new perspective on the statistical application.

Overall, this application proves that the presented method in this thesis can be applied to real world applications. Moreover, they can produce new insights on the considered topic.

In the following, we will see that Algorithms 4 and 5 are also stable approaches to show that a given matrix is an element of the interior of the completely positive cone.

7.13 Examples for Algorithms 4 and 5

To show the performance of Algorithms 4 and 5, we will consider again the matrix

$$A := \begin{pmatrix} 18 & 9 & 9 \\ 9 & 18 & 9 \\ 9 & 9 & 18 \end{pmatrix} \in \operatorname{int}(\mathcal{CP}_3)$$

as introduced in Example 2.22. Even though we already know a factorization proving $A \in \operatorname{int}(\mathcal{CP}_3)$, we will use $r = cp_3^+ = 4$ for the number of columns in our initial factorization.

Thus, we start with the initial factorization $A = BB^T$, where

$$B = \begin{pmatrix} 2.1213 & 1.2247 & 2.4495 & 2.4495 \\ -2.1213 & 1.2247 & 2.4495 & 2.4495 \\ 0.0000 & -2.4495 & 2.4495 & 2.4495 \end{pmatrix}. \tag{60}$$

This factorization is computed via the column replication approach, based on the factorization in equation (19) of A. For $\varepsilon = 0.1$ and the starting point

$$Q_0 = \begin{pmatrix} 0.5794 & 0.6095 & 0.3569 & 0.4067 \\ 0.4636 & -0.1065 & 0.3493 & -0.8073 \\ -0.3467 & -0.3148 & 0.8574 & 0.2134 \\ -0.5737 & 0.7197 & 0.1245 & -0.3706 \end{pmatrix},$$

Algorithm 5 returns after 0.012 seconds and 88 iterations the following factorization:

$$A = \tilde{B}\tilde{B}^T, \text{ with } \tilde{B} = \begin{pmatrix} 0.2629 & 2.3362 & 3.5296 & 0.1229 \\ 3.6184 & 0.1000 & 2.2107 & 0.1000 \\ 1.9501 & 2.5196 & 0.6420 & 2.7271 \end{pmatrix}.$$

This factorization is entrywise greater than or equal to $\varepsilon = 0.1$ and therefore shows $A \in \operatorname{int}(\mathcal{CP}_3)$. On the other hand, again starting with the initial matrix B in equation (60) and for $\varepsilon = 0.1$, Algorithm 4 returns for the starting matrix

$$Q_0 = \begin{pmatrix} 0.3741 & -0.8766 & 0.2043 & -0.2233 \\ -0.3010 & -0.4225 & -0.6309 & 0.5770 \\ 0.6418 & 0.1496 & 0.2442 & 0.7114 \\ 0.5980 & 0.1752 & -0.7075 & -0.3335 \end{pmatrix}$$

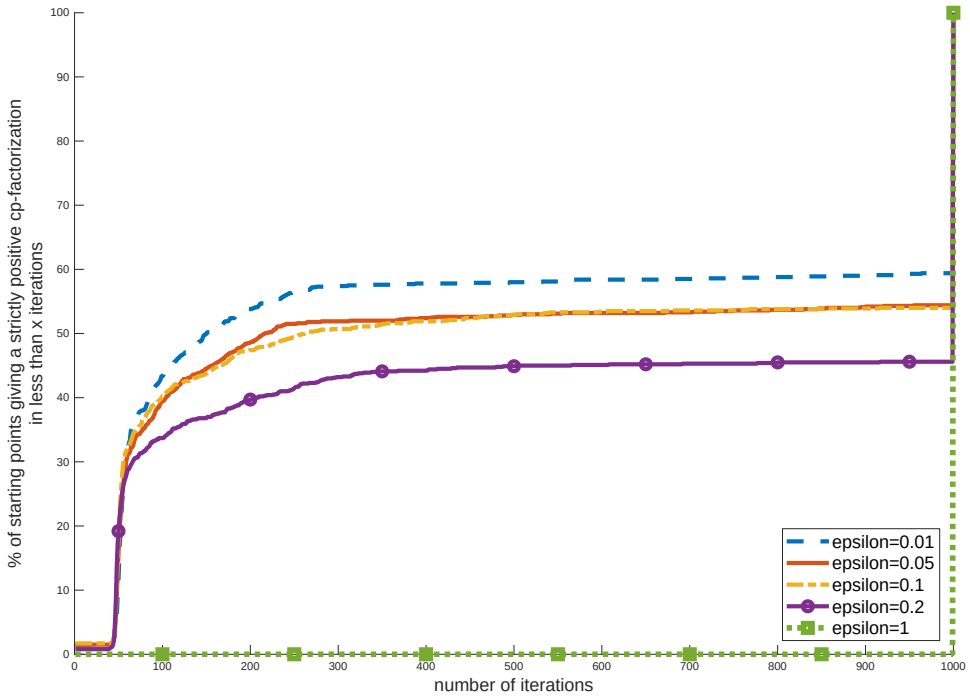
and after 4.9 seconds and 11 iterations the following factorization:

$$A = \hat{B}\hat{B}^T, \text{ with } \hat{B} = \begin{pmatrix} 0.5754 & 3.5983 & 1.8978 & 1.0581 \\ 0.8018 & 0.1625 & 2.2318 & 3.5142 \\ 3.8463 & 0.9917 & 1.2121 & 0.8678 \end{pmatrix}.$$

This factorization is again entrywise greater than or equal to $\varepsilon = 0.1$. Thus, both algorithms can be used to obtain a certificate for the matrix to be an element of the interior of the completely positive cone.

The influence of the parameter ε on the performance of Algorithm 5 is now illustrated in Figure 7.6. For those results, we test the success rate of 1000 starting points with a maximum of 1000 iterations per starting point for several values of ε, again for the matrix in Example 2.22. We use Algorithm 5 and as before, we fix the value $r = 4$.

Figure 7.6: Success rate of Algorithm 5 for Example 2.22.



In Figure 7.6, we can see that the success rate decreases for increasing values of ε. Especially if ε is chosen too large, Algorithm 5 will not converge any more. This drawback is illustrated for $\varepsilon = 1$, where the Algorithm fails for every starting point. But if $\varepsilon > 0$ is chosen small enough, we can obtain a strictly positive cp-factorization in more than 50% of the starting points Q_0 in less than 1000 iterations.

In the following chapter, we will introduce the connection of completely positive matrix factorization to general nonnegative matrix factorization and we will show how the Algorithms 1 and 2 can be adapted to become applicable in this context.

8 Nonnegative Matrix Factorization

In this chapter, we will take a closer look at the topic of nonnegative matrix factorization. First, we will consider the symmetric case and throughout this chapter, we will see that factorizing completely positive matrices is closely related to symmetric nonnegative matrix factorization. But on the other hand, it will also turn out that these two topics are still different and further adjustments are necessary to be able to apply Algorithms 1 and 2 in this setting. Here we will have a closer look at these adjustments in the setting of symmetric nonnegative matrix factorization and how the adapted versions of the Algorithms 1 and 2 perform in this setting. Afterwards, we will consider the general (non-symmetric) case of nonnegative matrix factorization. Here we will see that only Algorithm 2 can be modified to be numerically applicable in this context.

8.1 Symmetric Nonnegative Matrix Factorization

Let $A \in \mathbb{R}_+^{n\times n}$ be a symmetric matrix. For the symmetric nonnegative matrix factorization of A, consider the following problem, which can be found for example in [19, Equation (1)], [39, Equation (11)], [59, Problem 7.3] or [71, Equation (2)].

Definition 8.1. *Given a symmetric matrix $A \in \mathbb{R}_+^{n\times n}$ and $k \ll n$, the solution matrix $B \in \mathbb{R}_+^{n\times k}$ of*

$$\min_{B\in\mathbb{R}_+^{n\times k}} \|A - BB^T\|_F^2$$

yields the symmetric nonnegative matrix factorization BB^T of A.

Here we should note that in contrast to the cp-factorization, it is sufficient to have $A \approx BB^T$ instead of strict equality, since for a symmetric nonnegative matrix factorization, we are looking for a nonnegative matrix B giving a lowrank approximation BB^T to the matrix A. Completely positive matrix factorization can therefore be seen as a special case of the symmetric case of nonnegative matrix factorization, albeit without the low rank constraint. For the cp-factorizations we consider $k \geq \operatorname{cpr}(A)$, such that k is always greater than or equal to the rank of the matrix. In the setting of symmetric nonnegative matrix factorization, we are looking for a factorization that is entrywise nonnegative and gives a rank-k low-rank approximation to the given matrix such that $k < \operatorname{rank}(A) \leq n$.

Moreover, we can rewrite the problem in Definition 8.1 based on the notation we introduced in the previous chapters. More precisely, in the context of symmetric nonnegative matrix factoriza-

tion, we are looking for a matrix $X \in \mathcal{CP}_n$ with $\operatorname{cpr}(X) = k$, which solves the problem:

$$\min_{\substack{X \in \mathcal{CP}_n \\ \operatorname{cpr}(X)=k}} \|A - X\|_F^2.$$

Thus, the symmetric nonnegative matrix factorization searches for the best completely positive approximation of cp-rank k to A. Since in general, it is not possible to compute the cp-rank of a given completely positive matrix (cf. Section 2.4), we will use the rank of the matrix as a lower bound for the cp-rank, which is often tight, as shown in Chapter 3. For this, we will try to factorize a rank-k approximation of A instead of A itself.

The symmetric nonnegative matrix factorization is related to data clustering, particularly Kernel K-means clustering and Laplacian-based spectral clustering, as discussed in [39]. As a concrete example, it can be used to analyse the structure of a given dataset, like facial poses, as shown in [56], or heterogeneous microbiome data, as introduced in [71]. As we will see later, symmetric nonnegative matrix factorization can be seen as a special case of the general nonnegative matrix factorization. Therefore, more applications of this approach will be mentioned in the subsequent Section 8.2. In the following section, we will discuss the question of how to derive a symmetric nonnegative matrix factorization. Here two methods, which are based on Algorithms 1 and 2, will be introduced.

8.1.1 Algorithms for Symmetric Nonnegative Matrix Factorization

To compute a solution to the problem in Definition 8.1, and therefore a symmetric nonnegative matrix factorization of a given matrix A and given order k, there already exist several methods. A first algorithm was already given in Algorithm 6 by Ding et al. in [39], since we do not have any restrictions for the value k. Newton-like methods for symmetric nonnegative matrix factorization can be found in [66, Section 3]. For further methods on symmetric nonnegative matrix factorization, the reader is referred to Borhani et al. in [19]. Here the authors introduce an accelerated proximal gradient method and a certain alternating direction approach and show its convergence.

In the following, we will show that Algorithms 1 and 2 can also be applied to this setting. To this end, we will factorize a rank-k approximation to the given matrix A, instead of A itself. To determine the best rank-k approximation of a matrix in norm sense, we will use the well known theorem by Eckart and Young, cf. [45], which was proven to hold for any unitarily invariant norm by Mirsky, cf. [74]. The results as presented can be found for example in [58] and [90, Theorems 6.1 and 6.3].

Theorem 8.2. *Let $A \in \mathbb{R}^{n\times m}$ and consider its singular value decomposition $A = U\Sigma V^T$, where $U \in \mathcal{O}_n$, $V \in \mathcal{O}_m$ and*

$$\Sigma = \left(\begin{array}{ccc|ccc} \sigma_1 & & & & \vdots & \\ & \ddots & & \cdots & 0 & \cdots \\ & & \sigma_l & & \vdots & \\ \hline & \vdots & & & \vdots & \\ \cdots & 0 & \cdots & \cdots & 0 & \cdots \\ & \vdots & & & \vdots & \end{array}\right) \in \mathbb{R}^{n\times m},$$

where $\operatorname{rank}(A) = l$ and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_l > 0$ are the (positive) singular values of A. So, A can be written as

$$A = \sum_{i=1}^{l} \sigma_i u_i v_i^T,$$

where u_i respectively v_i is the i-th column of the matrix U respectively V for every $i \in \{1, \ldots, l\}$. Then for $k \leq l = \operatorname{rank}(A)$, the best rank-$k$ approximation (in the Frobenius norm) of A is given by

$$A_k := \sum_{i=1}^{k} \sigma_i u_i v_i^T.$$

In other words,

$$A_k = \operatorname{argmin}\left\{\|A - X\|_F^2 \;\middle|\; X \in \mathbb{R}^{n\times m} \text{ with } \operatorname{rank}(X) \leq k\right\}$$

with corresponding minimal value

$$\|A - A_k\|_F^2 = \sum_{i=k+1}^{m} \sigma_i^2.$$

Moreover, if $\sigma_k > \sigma_{k+1}$, then A_k is the unique global minimizer.

The following lemma now motivates the use of Algorithm 1 or 2 to obtain a symmetric nonnegative matrix factorization and gives rise to the subsequent remarks.

Lemma 8.3. *Let $A \in \mathcal{S}_n$ and consider its best rank-k approximation A_k from Theorem 8.2 for some $k \leq n$. Further assume that $A_k \in \mathcal{CP}_n$. Then any cp-factorization $A_k = BB^T$ with $B \in \mathbb{R}^{n\times k}$ of A_k is a solution to the problem in Definition 8.1 and therefore a symmetric nonnegative matrix factorization of A of rank k.*

Proof. Let $A_k = BB^T$ be a cp-factorization of the matrix A_k. Then $\|A - A_k\|_F$ is minimal among all matrices of rank k due to Theorem 8.2. Thus, $\|A - BB^T\|_F$ is minimal and since $B \in \mathbb{R}^{n\times k}$ with $B \geq 0$, we have that BB^T is a symmetric nonnegative matrix factorization of A. □

Thus, for a symmetric nonnegative matrix factorization, it is sufficient to find a cp-factorization for the best rank-k approximation of the given matrix of order $n \times k$. This gives rise to Algorithm 7.

Algorithm 7 Symmetric nonnegative matrix factorization based on Algorithm 1

Input: $A \in \mathbb{R}^{n\times n}$ with its singular value decomposition $A = U\Sigma V^T$; $k \leq n$; initial matrix $Q_0 \in \mathcal{O}_k$

1: $A_k \leftarrow \sum_{i=1}^{k} \sigma_i u_i v_i^T$
2: $[V_k, \Sigma_k] \leftarrow \text{eig}(A_k)$
3: $B_k \leftarrow V_k\sqrt{\Sigma_k} \in \mathbb{R}^{n\times k}$
4: $i \leftarrow 0$
5: **while** $B_kQ_i \ngeq 0$ **do**
6: $\quad P_i \leftarrow P_{\mathcal{P}}(Q_i)$
7: $\quad Q_{i+1} \leftarrow P_{\mathcal{O}_k}(P_i)$
8: $\quad i \leftarrow i + 1$
9: **end while**

Output: $Q_i \in \mathcal{O}_r$ and a symmetric nonnegative matrix factorization $(B_kQ_i)(B_kQ_i)^T$ of A.

Here in step 1, we determine the best rank-k approximation A_k to A, followed by the computation of the eigendecomposition of the matrix A_k in step 2. With this, we compute the matrix B_k in step 3, which now replaces the given initial factorization matrix B in Algorithm 1. The algorithm then continues like Algorithm 1, based on the matrix B_k. More precisely, we consider the set $\mathcal{P} = \{Q \in \mathbb{R}^{k\times k} \mid B_kQ \geq 0\}$ and the set $\mathcal{O}_k$ for the alternating projections approach. Again, we will use the second order cone approach, introduced in Lemma 6.1, to compute the projection onto the set $\mathcal{P}$. For the projection onto $\mathcal{O}_k$, we use the polar decomposition from Lemma 6.2. If the algorithm terminates successfully, it returns an orthogonal matrix Q_i giving a symmetric nonnegative matrix factorization $(B_kQ_i)(B_kQ_i)^T$ of A due to Lemma 8.3.

Remark 8.4. *Since for the symmetric nonnegative matrix factorization it is necessary to determine an entrywise nonnegative matrix $B \in \mathbb{R}^{n\times k}$, it is not necessary to use column replication to obtain a feasible initial factorization. The algorithm of course terminates successfully only for completely positive matrices A_k with $\text{cpr}(A_k) = k$. For the case $\text{cpr}(A_k) > k$, it is possible to take the best rank-$(k-1)$ approximation A_{k-1} of the given matrix A and to check whether $\text{cpr}(A_{k-1}) = k$. Unfortunately, even if $A_{k-1} = BB^T$ with $B \in \mathbb{R}^{n\times k}$ is a cp-factorization of A_{k-1}, this factorization is not necessarily a symmetric nonnegative matrix factorization of A for a given value k since $\text{rank}(B) = k - 1$ and due to Theorem 8.2, it may happen that $\|A - A_k\|_F < \|A - A_{k-1}\|_F = \|A - BB^T\|_F$.*

Furthermore, we have the following convergence result:

Theorem 8.5. *Let $A \in \mathcal{S}_n$ such that the best rank-k approximation $A_k \in \mathcal{CP}_n$. Let $A_k = B_k{B_k}^T$ be any initial factorization with $B_k \in \mathbb{R}^{n\times k}$ and assume that $\text{cpr}(A_k) = k \leq \text{rank}(A)$. Define $\mathcal{P} := \{Q \in \mathbb{R}^{k\times k} \mid B_kQ \geq 0\}$. Then we have:*

(a) $\mathcal{P} \cap \mathcal{O}_k \neq \emptyset$,

(b) if started at a point Q_0 close to $\mathcal{P} \cap \mathcal{O}_k$, then Algorithm 7 converges to a point $Q^* \in \mathcal{P} \cap \mathcal{O}_k$. In this case, $A_k = (B_kQ^*)(B_kQ^*)^T$ is a completely positive factorization of A_k, which yields a symmetric nonnegative matrix factorization $(B_kQ^*)(B_kQ^*)^T$ of A.

Proof. (a): It follows from $A_k \in \mathcal{CP}_n$ and $k = \operatorname{cpr}(A)$ that there exists $C \in \mathbb{R}^{n\times k}$ such that $C \geq 0$ and $A = CC^T$. Since $A = BB^T = CC^T$ and the matrices B, C are of the same order, Lemma 3.11 implies that there exists $Q \in \mathcal{O}_k$ such that $BQ = C \geq 0$, i.e., $Q \in \mathcal{P} \cap \mathcal{O}_k$.

(b): Both $\mathcal{P}$ and $\mathcal{O}_k$ are closed semialgebraic sets due to Lemma 6.5. Moreover, $\mathcal{O}_k$ is bounded, as shown in Lemma 3.2. The convergence result now follows by applying [42, Theorem 7.3], which can be found in Theorem 5.57 of this thesis. □

Instead of using Algorithm 1 as a basis, where it is necessary to solve a second order cone problem in every projection step onto the set $\mathcal{P}$, we can also use Algorithm 2. This motivates Algorithm 8.

Algorithm 8 Symmetric nonnegative matrix factorization based on Algorithm 2

Input: $A \in \mathbb{R}^{n\times n}$ with its singular value decomposition $A = U\Sigma V^T$; $k \leq n$; initial matrix $Q_0 \in \mathcal{O}_k$

1: $A_k \leftarrow \sum_{i=1}^{k} \sigma_i u_i v_i^T$
2: $[V_k, \Sigma_k] \leftarrow \text{eig}(A_k)$
3: $B_k \leftarrow V_k\sqrt{\Sigma_k} \in \mathbb{R}^{n\times k}$
4: $i \leftarrow 0$
5: **while** $B_kQ_i \not\geq 0$ **do**
6: $\quad D \leftarrow \max\{B_kQ_i, 0\}$ entrywise
7: $\quad \widehat{P}_i \leftarrow B_k^+ D + (I - {B_k}^+B_k)Q_i$
8: $\quad Q_{i+1} \leftarrow P_{\mathcal{O}_k}(\widehat{P}_i)$
9: $\quad i \leftarrow i + 1$
10: **end while**

Output: $Q_i \in \mathcal{O}_r$ and a symmetric nonnegative matrix factorization $(B_kQ_i)(B_kQ_i)^T$ of A.

The steps 1 to 3 in Algorithm 8 are equal to the first steps in Algorithm 7, such that we start the algorithms with the same initial factorization $B_k{B_k}^T$ of A_k. Compared to Algorithm 7 and based on Lemma 6.10, we compute $\widehat{P}$ to approximate $P_{\mathcal{P}}(Q)$, as motivated in Section 6.2 and as implemented in Algorithm 2. Nevertheless, this modified approach is again not a pure alternating projections approach such that we can not prove a local convergence result like in Theorem 8.5. In the following, we will analyse the numerical performance of Algorithms 7 and 8 for concrete examples.

8.1.2 Numerical Results for Symmetric Nonnegative Matrix Factorization

The following numerical results were again carried out on a computer with 88 Intel Xenon ES-2699 cores (2.2 Ghz each) and a total of 0.792 TB RAM. The algorithms were implemented in MatlabR2017a, the SOCPs in Algorithm 7 were solved using Yalmip R20170626 and SDPT3 4.0. The algorithms terminate successfully at iteration i if $B_kQ_i \geq -10^{-15}$, it terminates unsuccessfully if a maximum number of iterations (usually 5000) is reached.

As a first example, we consider again the matrix A_{DS} in Example 2.43. Here we saw in Section 7.7 that Algorithms 1 and 2 fail to show that the matrix is completely positive. In the following, we will show that with the help of Algorithms 7 or 8, we can give a symmetric nonnegative matrix factorization for this matrix. To this end, we fix the parameter $k = 2$. Then we have the following best rank-2 approximation A_2 to A_{DS} due to Theorem 8.2:

$$A_{DS} = \begin{pmatrix} 8 & 5 & 1 & 1 & 5 \\ 5 & 8 & 5 & 1 & 1 \\ 1 & 5 & 8 & 5 & 1 \\ 1 & 1 & 5 & 8 & 5 \\ 5 & 1 & 1 & 5 & 8 \end{pmatrix} = \sum_{i=1}^{5} \sigma_i u_i v_i^T$$

$$A_2 = \begin{pmatrix} 7.7889 & 5.1708 & 0.9348 & 0.9348 & 5.1708 \\ 5.1708 & 4.3618 & 3.0528 & 3.0528 & 4.3618 \\ 0.9348 & 3.0528 & 6.4798 & 6.4798 & 3.0528 \\ 0.9348 & 3.0528 & 6.4798 & 6.4798 & 3.0528 \\ 5.1708 & 4.3618 & 3.0528 & 3.0528 & 4.3618 \end{pmatrix} = \sum_{i=1}^{2} \sigma_i u_i v_i^T,$$

where $\sum_{i=1}^{5} \sigma_i u_i v_i^T$ denotes the singular value decomposition of A_{DS}. The initial factorization $A_2 = B_2 B_2^T$ is then given via the eigendecomposition of A_2 and the matrix B_2 reads as

$$B_2 = \begin{pmatrix} -2.0000 & 1.9465 \\ -2.0000 & 0.6015 \\ -2.0000 & -1.5747 \\ -2.0000 & -1.5747 \\ -2.0000 & 0.6015 \end{pmatrix}.$$

As shown in the proof to Corollary 3.13, an entrywise nonpositive column can be easily transformed to a nonnegative column, using an orthogonal matrix. We therefore use

$$A_2 = \begin{pmatrix} 2.0000 & 1.9465 \\ 2.0000 & 0.6015 \\ 2.0000 & -1.5747 \\ 2.0000 & -1.5747 \\ 2.0000 & 0.6015 \end{pmatrix} \begin{pmatrix} 2.0000 & 1.9465 \\ 2.0000 & 0.6015 \\ 2.0000 & -1.5747 \\ 2.0000 & -1.5747 \\ 2.0000 & 0.6015 \end{pmatrix}^T$$

as an initial factorization. Starting Algorithm 7 with this input matrix, the very first starting point returns the following decomposition in 2 seconds.

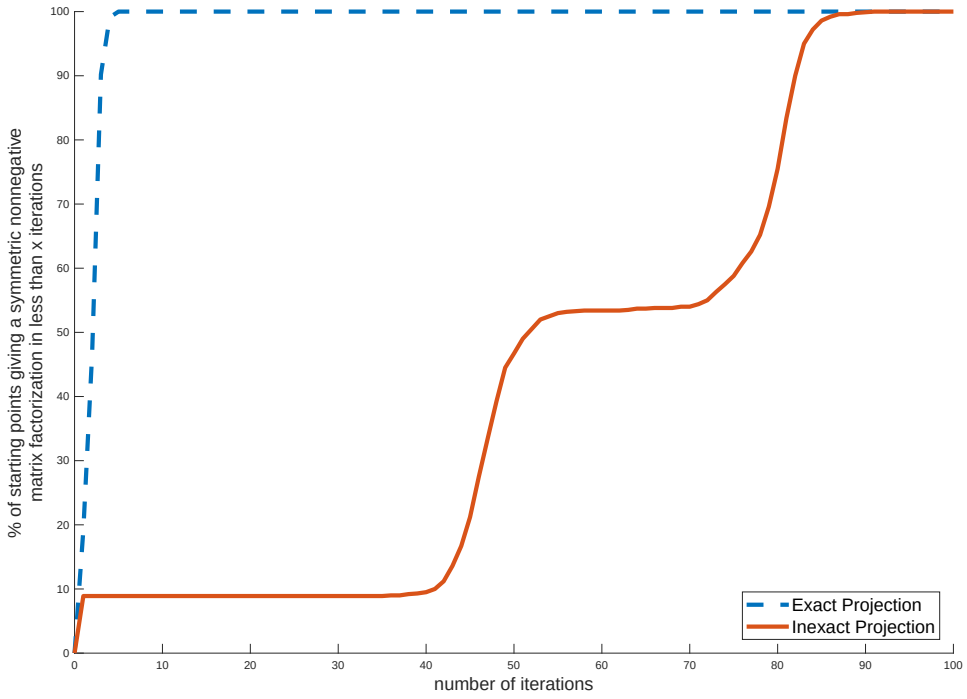
$$A_2 = \begin{pmatrix} 2.7897 & 0.0817 \\ 1.8238 & 1.0177 \\ 0.2609 & 2.5321 \\ 0.2609 & 2.5321 \\ 1.8238 & 1.0177 \end{pmatrix} \begin{pmatrix} 2.7897 & 0.0817 \\ 1.8238 & 1.0177 \\ 0.2609 & 2.5321 \\ 0.2609 & 2.5321 \\ 1.8238 & 1.0177 \end{pmatrix}^T$$

Here only one iteration was necessary. This decomposition is an exact cp-factorization of A_2 and therefore a symmetric nonnegative matrix factorization of A according to Lemma 8.3. Whereas we could not verify the membership of A_{DS} to the completely positive cone algorithmically, we can give a symmetric nonnegative matrix factorization of rank 2 for this matrix.

Apart from Algorithm 7, we can also apply Algorithm 8 to A_{DS} with $k = 2$. A comparison of the performance of Algorithms 7 and 8 for the same 1000 starting points, again for the matrix A_{DS},

can be found in Figure 8.1. In both cases, we consider 100 as a maximum number of iterations per starting point for each approach. But it turns out that in no case 100 iterations were necessary.

Figure 8.1: Success rates of Algorithms 7 and 8 for Example 2.43 with $k = 2$ for the same starting points.

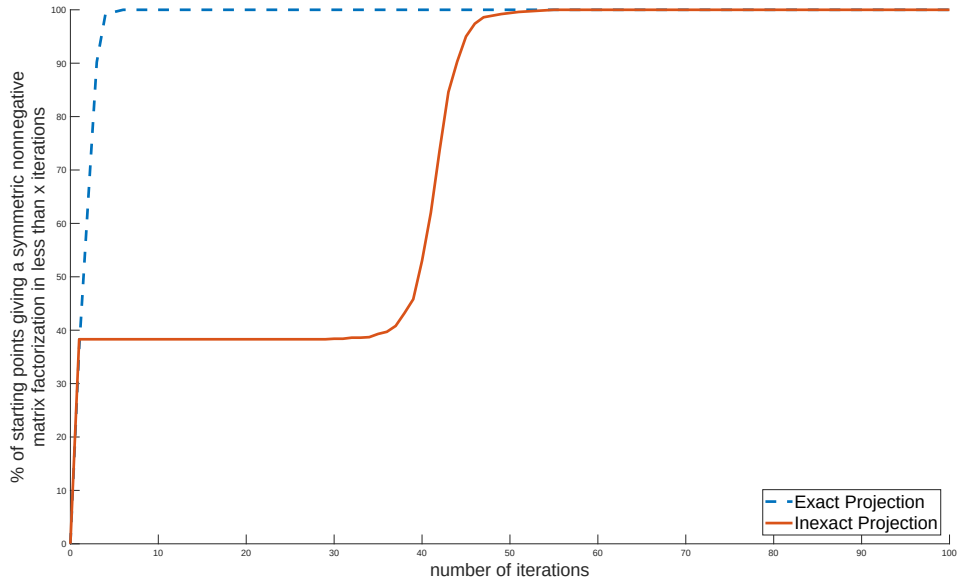


To analyse the graphs in Figure 8.1 in more detail, we first focus on the blue dashed graph, representing the performance of Algorithm 7. Here we notice that any of the 1000 starting points returns a symmetric nonnegative matrix factorization in less than 5 iterations. Considering the red graph, representing the performance of Algorithm 8, we see that to exceed a success rate of 50%, it becomes necessary to allow more than 50 iterations. But after around 80 iterations any starting point returns a symmetric nonnegative matrix factorization of A_{DS}. For the 1000 starting points, Algorithm 7 takes around 350 seconds, whereas Algorithm 8 is less time consuming and takes only 1.3 seconds. This is again due to the fact that it is not necessary to solve an SOCP in every iteration step in Algorithm 8. But since the order of the SOCP is in general smaller than n, the drawback of Algorithm 7 compared to Algorithm 8 is smaller than the drawback of Algorithm 1 compared to Algorithm 2.

Increasing values of k lead to a failure in both approaches. This is presumably due to the fact that both approaches start struggling to show A_k is completely positive for $k > 2$, analogue to the results shown in Section 7.7.

Similar to the first example, we obtain the following results on symmetric nonnegative matrix factorization of the matrix A in Example 2.22, again for $k = 2$. Here an illustrative comparison of the performance of Algorithms 7 and 8 for this matrix can be found in Figure 8.2.

Figure 8.2: Success rates of Algorithms 7 and 8 for Example 2.22 with $k = 2$ for the same starting points.

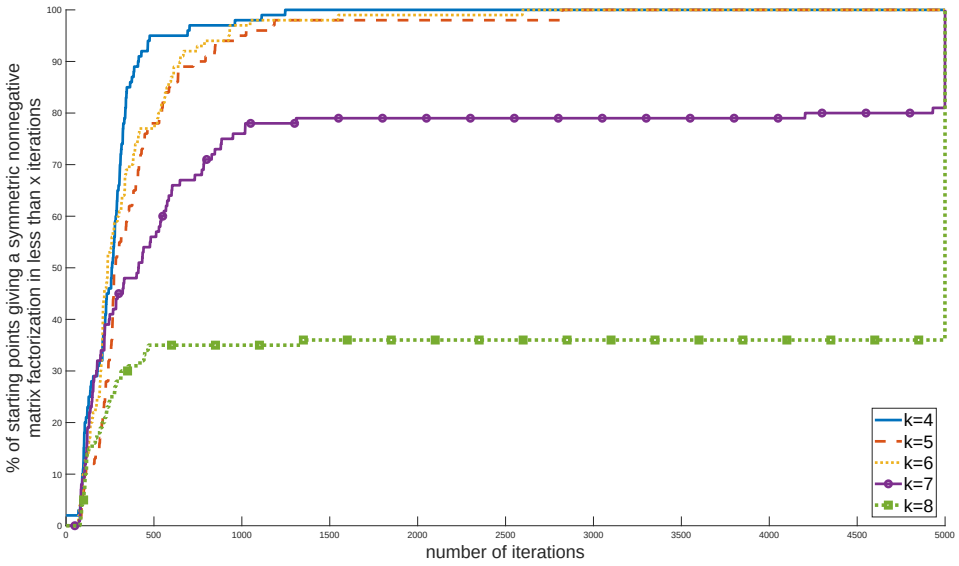


Here again, we test the same 1000 starting points for both approaches and as before, both algorithms return a symmetric nonnegative matrix factorization in less then 100 iterations for any starting point. Moreover, Algorithm 7 takes again at most 5 iterations to return a factorization.

To show the influence of the parameter k, we will again use the method described in Section 7.9 to obtain randomly generated matrices. In this concrete setting, we use the Matlab command `randn` to generate a random matrix $B \in \mathbb{R}^{n\times n}$ for a given scalar value n. Next, we compute C by setting $C_{ij} := |B_{ij}|$ for all i, j, and finally take $A = CC^T$ as the matrix for which we want to find a symmetric nonnegative matrix factorization. Then by construction $A \in \mathcal{CP}_n$ and again, as an initial factorization for both Algorithms 7 and 8, we consider the the matrix B_k as the exact initail factorization of the best rank-k approximation A_k of A.

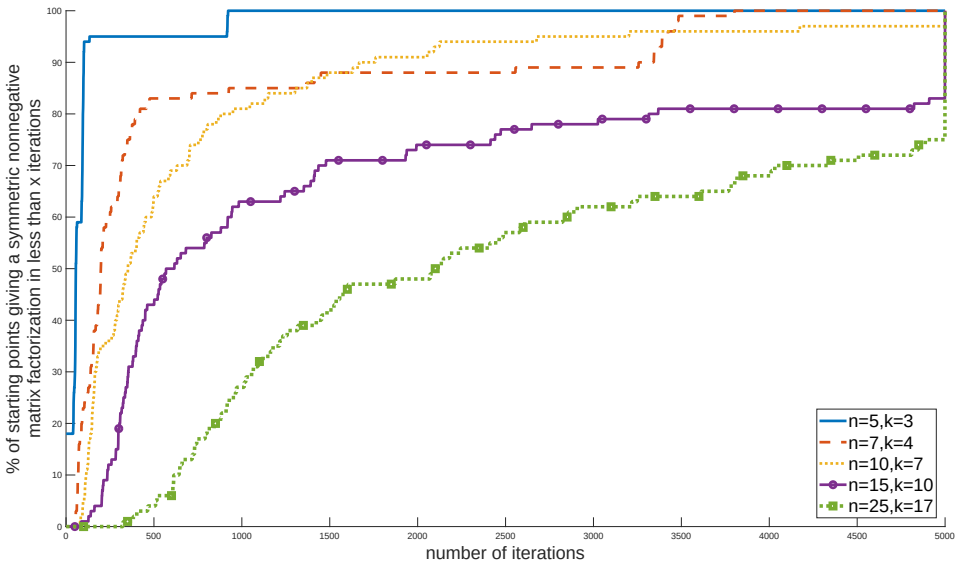
The performance of Algorithm 8 for $n = 10$ and several values of $k \leq n$ can be found in Figure 8.3. Here we test the same randomly generated matrix $A \in \mathbb{R}^{10\times 10}$ and its best rank-k approximation A_k for every k. To be more precise, for every k, we use 100 randomly chosen initial orthogonal matrices and test if the algorithm terminates successfully in less than 5000 iterations. As it turns out, it is possible to determine a symmetric nonnegative matrix factorization of A for every value of k. But the success rate depends on the parameter k. In particular, the closer k gets to the order n of A, the lower the success rate. For $k \leq 6$, the algorithm terminates successfully in less than 3000 iterations for every starting point, whereas for $k = 8$ for instance, even after 5000 iterations only around 45% of the starting points return a symmetric nonnegative matrix factorization. Combined with the results of the numerical experiments in Section 7.1, we can therefore see that the success rate of the presented Algorithms 2 and 8 (which is based on Algorithm 2) is higher, the more distant r (or k respectively) is chosen to n.

Figure 8.3: Success rate of Algorithm 8 for random matrices A of order 10×10 for several values of k and 100 randomly chosen starting points for each k.



In the following experiment, we will analyse the influence of the order n of the matrix A on the performance of Algorithm 8. We will test the performance of the algorithm for randomly generated matrices $A \in \mathbb{R}^{n \times n}$ for several values of n. For every n we fix $k = \lfloor 0.7n \rfloor$, where $\lfloor \cdot \rfloor$ is again the floor function, and we test 100 randomly chosen initial starting points Q_0 and a maximum of 5000 iterations per starting point. The success rate of Algorithm 8 in this setting is illustrated in Figure 8.4.

Figure 8.4: Success rate of Algorithm 8 for random matrices of order $n \times n$ with $k = \lfloor 0.7n \rfloor$.



Here we see that again for every n, the algorithm terminates successfully for some starting points. Moreover, for every $n \leq 7$, the algorithm returns a symmetric nonnegative matrix factorization for every starting point. If we consider $n = 15$ for instance, we notice that this property does not hold for larger values of n since in this case only around 80% of the starting points lead to a symmetric nonnegative matrix factorization. In total, we have some similarities to the results in Section 7.2, where we illustrated the performance of Algorithm 2 in a similar setting.

In the following, we will consider the general case of nonnegative matrix factorization, where the given matrix is not necessarily square and we are looking for a nonnegative factorization with matrices of different order in general. Here it will turn out that some of the results obtained so far can also be applied to this generalized framework.

8.2 General Nonnegative Matrix Factorization

An introduction to the nonnegative matrix factorization problem can be found for example in [47]. As shown there, for a nonnegative matrix factorization of a matrix $A \in \mathbb{R}^{n\times m}$, consider the following problem, which can be found for example in [47, Equation (3)] or more general in [97, Equation (2)].

Definition 8.6. *Let $A \in \mathbb{R}_+^{n\times m}$ and $k \ll \min\{n, m\}$, then the solution matrices $B \in \mathbb{R}_+^{n\times k}$ and $C \in \mathbb{R}_+^{k\times m}$ of*

$$\min_{B\in\mathbb{R}_+^{n\times k},\, C\in\mathbb{R}_+^{k\times m}} \|A - BC\|_F^2$$

yield the nonnegative matrix factorization BC of A.

Similar to the symmetric case, we are looking for a certain nonnegative low-rank approximation of A. Adding further constraints to this problem or slight changes in the objective function lead to various specially structured nonnegative matrix factorization problems. For a comprehensive collection of these problems, the reader is referred to [97]. For example, if we drop the assumption that A is entrywise nonnegative in Definition 8.6 and allow one of the matrices B, C to have negative entries, this defines the so called *Semi Nonnegative Matrix Factorization*, cf. [97, Section 2.2]. This problem is motivated by data clustering.

Another example is the so called *Sparse Nonnegative Matrix Factorization*, cf. [97, Sections 2.7-2.9], where we add the (possibly weighted) penalty term $\sum_{i,j} C_{ij}$ to the objective function in Definition 8.6 to ensure sparsity for the matrix C.

Also the symmetric case in Section 8.1 can be seen as a special case of the problem in Definition 8.6.

Nonnegative matrix factorization itself can be seen as a special subclass of so called constrained low-rank matrix approximation problems as introduced in [47]. Therefore, various applications of the nonnegative matrix factorization approach are related to this topic. One very illustrative application, again mentioned in [47], is hyperspectral imaging. In contrast to a standard RGB image where every pixel has 3 channels, every pixel of a hyperspectral image is represented via more than 100 channels, which correspond to deeper information of several wavelengths of the image,

some of them blind to the human eye. By that, a deeper analysis of the structure of the image becomes possible. This method boils down to the nonnegative matrix factorization framework.

But even more general, in the context of data science, nonnegative matrix factorization can be used for so called intelligent data analysis, as shown in [23] as the second chapter in the book of Naik [77]. Especially when the quantities are known to be nonnegative, for example due to physical rules, nonnegative matrix factorization can be used to determine part-based representations of given data. Here a concrete example, again given in [23], is educational data mining. For a survey on this topic, the reader is referred to [75]. Here the goal is to collect, store and analyse data obtained from learning and evaluation processes of students.

Other possible applications are multi-document summarization, see for example [22], or analysis of magnetic resonance spectroscopy data, as shown in [67]. As already mentioned for the symmetric case, nonnegative matrix factorization is closely related to data clustering. For more details on this application, the reader is referred to [68].

In the following, we will have a closer look at algorithmic approaches to obtain a nonnegative matrix factorization. Moreover, we will show that the results obtained so far in this thesis can be generalized to this setting

8.2.1 Generalizing the Results to the Framework of Nonnegative Matrix Factorization

To show that the results in Chapter 6 can also be applied in the context of nonnegative matrix factorization, a first thing to mention is that Theorem 8.2 holds for every $A \in \mathbb{R}^{n\times m}$. Thus, we can again consider the best rank-k approximation A_k to A. Similar to the symmetric case, now the idea is to obtain a nonnegative factorization for the matrix A_k. Moreover, the following result holds.

Lemma 8.7. *Let $A \in \mathbb{R}^{n\times m}_+$ and consider its best rank-k approximation A_k from Theorem 8.2 for some $k \leq \min\{n, m\}$. Then any factorization $A_k = BC$ with $B \in \mathbb{R}^{n\times k}_+$ and $C \in \mathbb{R}^{k\times m}_+$ of A_k is a solution to the problem in Definition 8.6 and therefore a nonnegative matrix factorization of A for a given value $k \leq \min\{n, m\}$.*

Proof. Let $A_k = BC$ be a nonnegative factorization of the matrix A_k. Then $\|A - A_k\|_F$ is minimal among all matrices of rank k due to Theorem 8.2. Thus, $\|A - BC\|_F$ is minimal and since $B \in \mathbb{R}^{n\times k}_+$ and $C \in \mathbb{R}^{k\times m}_+$, we know that BC is a nonnegative factorization of A. □

Thus, to obtain a nonnegative matrix factorization of A, it is sufficient to factorize $A_k = BC$, where $B \in \mathbb{R}^{n\times k}$ and $C \in \mathbb{R}^{k\times m}$ are entrywise nonnegative matrices. To this end, a first step is to compute an initial factorization, which is not necessarily entrywise nonnegative. For this, the following approach can be used.

Starting with $A = U\Sigma V^T$, the singular value decomposition of A, we can obtain the best rank-k approximation of A, as shown in Theorem 8.2, as

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T,$$

where $\operatorname{rank}(A) \geq k$ and the u_i respectively v_i represent the i-th column of U respectively V. This can also be written in matrix form as

$$A_k = U_k \Sigma_k V_k^T,$$

where the matrices $U_k,\ \Sigma_k,\ V_k$ are the truncated versions of U, Σ and V. More precisely, we have, again with u_i respectively v_i denoting the i-th column of U respectively V,

$$\begin{aligned} A_k &= \underbrace{\begin{pmatrix} u_1 & \dots & u_n \end{pmatrix}}_{\in \mathbb{R}^{n\times n}} \underbrace{\left(\begin{array}{ccc|ccc} \sigma_1 & & & & \vdots & \\ & \ddots & & \cdots & 0 & \cdots \\ & & \sigma_k & & \vdots & \\ \hline & \vdots & & & \vdots & \\ \cdots & 0 & \cdots & \cdots & 0 & \cdots \\ & \vdots & & & \vdots & \end{array}\right)}_{\in \mathbb{R}^{n\times m}} \underbrace{\begin{pmatrix} v_1 & \dots & v_m \end{pmatrix}^T}_{\in \mathbb{R}^{m\times m}} \\ &= \underbrace{\begin{pmatrix} u_1 & \dots & u_k \end{pmatrix}}_{=:U_k \in \mathbb{R}^{n\times k}} \underbrace{\begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{pmatrix}}_{=:\Sigma_k \in \mathbb{R}^{k\times k}} \underbrace{\begin{pmatrix} v_1 & \dots & v_k \end{pmatrix}^T}_{=:V_k^T \in \mathbb{R}^{k\times m}}. \end{aligned}$$

An initial factorization $A = B_k C_k$, where $B_k \in \mathbb{R}^{n\times k}$ and $C_k \in \mathbb{R}^{k\times m}$ are not necessarily entrywise nonnegative, can now be obtained by setting

$$B_k := U_k \sqrt{\Sigma_k} \in \mathbb{R}^{n\times k} \text{ and } C_k := \sqrt{\Sigma_k} V_k^T \in \mathbb{R}^{k\times m}. \tag{61}$$

Now we discuss the question of how to transform this factorization into a nonnegative factorization. Due to the lack of symmetry in this general framework, we can not apply Lemma 3.11 in this setting, as the following example substantiates. Therefore, it becomes necessary to introduce a generalized version of this lemma. As it turns out, orthogonal matrices are not the key tool any more.

Example 8.8. *Let*

$$B = \begin{pmatrix} 6 & 0 \\ 0 & 6 \end{pmatrix},\ C = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},\ D = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \text{ and } F = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}.$$

Then we have $BC = DF$, but there does not exist a $Q \in \mathcal{O}_2$ such that $BQ = D$ and $Q^T C = F$ since we have

$$\underbrace{\begin{pmatrix} 6 & 0 \\ 0 & 6 \end{pmatrix}}_{B} \cdot \underbrace{\begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}}_{Q} = \underbrace{\begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}}_{D} \quad \text{and} \quad \underbrace{\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}}_{Q^{-1}} \cdot \underbrace{\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}}_{C} = \underbrace{\begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}}_{F},$$

where $Q \notin \mathcal{O}_2$. But $Q \in \mathbb{R}^{2\times 2}$ is a nonsingular matrix with $BQ = D$ and $Q^{-1}D = F$.

This now motivates the following results. Similar to Section 3.3 and especially to the results in Lemmas 3.10 and 3.11, we need the following properties to show that nonsingular matrices will replace orthogonal matrices in this setting. From now on, we assume that $k \leq \min\{n, m\}$.

Lemma 8.9. *Let $B, D \in \mathbb{R}^{n\times k}$ and $C, F \in \mathbb{R}^{k\times m}$ with $BC = DF$. Further assume that all four matrices are of rank k. For $i = 1, \ldots, n$, let B_{i*}, D_{i*}, $(BC)_{i*}$ resp. $(DF)_{i*}$ denote the i-th rows of B, D, BC resp. DF. In the same way, for $j = 1, \ldots, m$, let C_{*j}, F_{*j}, $(BC)_{*j}$ resp. $(DF)_{*j}$ denote the j-th column of C, F, BC resp. DF. Further let $R(B) \subseteq \mathbb{R}^k$ resp. $R(D) \subseteq \mathbb{R}^k$ denote the subspaces spanned by the rows of B and D, respectively. And in the same way, let $S(C) \subseteq \mathbb{R}^k$ resp. $S(F) \subseteq \mathbb{R}^k$ denote the subspace spanned by the columns of C and F, respectively. Then:*

(a)

$$B_{n*} = \sum_{i=1}^{n-1} \lambda_i B_{i*} \quad \text{if and only if} \quad (BC)_{n*} = \sum_{i=1}^{n-1} \lambda_i (BC)_{i*}, \tag{62}$$

with the same scalar values λ_i ($i = 1, \ldots, n$) in both equations.

(b)

$$C_{*m} = \sum_{j=1}^{m-1} \lambda_j C_{*j} \quad \text{if and only if} \quad (BC)_{*m} = \sum_{j=1}^{m-1} \lambda_j (BC)_{*j}, \tag{63}$$

with the same scalar values λ_j ($j = 1, \ldots, m$) in both equations.

(c) There exists a linear map $f : R(B) \to R(D)$ such that $f(B_{i*}) = D_{i*}$ for all $i = 1, \ldots, n$.

(d) There exists a linear map $g : S(C) \to S(F)$ such that $g(C_{*j}) = F_{*j}$ for all $j = 1, \ldots, m$.

Proof. Although we will find some similarities in the proofs of parts (a) and (b), and as well as in (c) and (d), the complete proof of each part will be given for the reader's convenience.

(a) We will prove both directions separately. To show that the right hand side in (62) is necessary, we assume that the left hand side holds and we consider the last row $(BC)_{n*}$ of BC. Thus, we have

$$(BC)_{n*} = B_{n*}C = \left(\sum_{i=1}^{n-1} \lambda_i B_{i*}\right) C = \sum_{i=1}^{n-1} \lambda_i B_{i*}C = \sum_{i=1}^{n-1} \lambda_i (BC)_{i*},$$

such that the equation on the right hand side in (62) holds.

Conversely, assume that the equation on the right hand side of (62) holds. Since $(BC)_{ij} = B_{i*}C_{*j}$ for every $i = 1, \ldots, n$ and $j = 1, \ldots, m$, we have for any entry $(BC)_{nj}$ of the row $(BC)_{n*}$ that

$$B_{n*}C_{*j} = (BC)_{nj} = \sum_{i=1}^{n-1} \lambda_i (BC)_{ij} = \sum_{i=1}^{n-1} \lambda_i B_{i*}C_{*j}$$

for every $j = 1, \dots, m$. This gives

$$\left(B_{n*} - \sum_{i=1}^{n-1} \lambda_i B_{i*} \right) C_{*j} = 0 \text{ for every } j = 1, \dots, m.$$

Thus, we get

$$\left(B_{n*} - \sum_{i=1}^{n-1} \lambda_i B_{i*} \right) C = 0.$$

Moreover, since C is of full row-rank, the Moore-Penrose-inverse C^+ is a right inverse of C, cf. Lemma A.3, and we get

$$\left(B_{n*} - \sum_{i=1}^{n-1} \lambda_i B_{i*} \right) CC^+ = 0 \cdot C^+ \quad \Leftrightarrow \quad \left(B_{n*} - \sum_{i=1}^{n-1} \lambda_i B_{i*} \right) = 0.$$

This now proves the equality on the left hand side in (62).

(b) Again, we will prove both directions separately. To show that the right hand side in (63) is necessary, we assume that the left hand side holds and we consider the last column $(BC)_{*m}$ of BC. Thus, we have

$$(BC)_{*m} = BC_{*m} = B \left(\sum_{j=1}^{m-1} \lambda_j C_{*j} \right) = \sum_{j=1}^{m-1} \lambda_j BC_{*j} = \sum_{j=1}^{m-1} \lambda_j \left(BC \right)_{*j},$$

such that the equation on the right hand side in (63) holds.

Conversely, assume that the equation on the right hand side of (63) holds. Since $(BC)_{ij} = B_{i*}C_{*j}$ for every $i = 1, \dots, n$ and $j = 1, \dots, m$, we have for any entry $(BC)_{im}$ of the column $(BC)_{*m}$ that

$$B_{i*}C_{*m} = (BC)_{im} = \sum_{j=1}^{m-1} \lambda_j \left(BC \right)_{ij} = \sum_{j=1}^{m-1} \lambda_j B_{i*} C_{*j} = B_{i*} \sum_{j=1}^{m-1} \lambda_j C_{*j}$$

for every $i = 1, \dots, n$. This gives

$$B_{i*} \left(C_{*m} - \sum_{j=1}^{m-1} \lambda_j C_{*j} \right) = 0 \text{ for every } i = 1, \dots, n. \tag{64}$$

Thus, we get

$$B \left(C_{*m} - \sum_{j=1}^{m-1} \lambda_j C_{*j} \right) = 0.$$

Moreover, since B is of full column-rank, the Moore-Penrose-inverse B^+ is a left inverse

of B, cf. Lemma A.3, and we get

$$B^+B\left(C_{*m} - \sum_{j=1}^{m-1} \lambda_j C_{*j}\right) = B^+ \cdot 0 \quad \Leftrightarrow \quad \left(C_{*m} - \sum_{j=1}^{m-1} \lambda_j C_{*j}\right) = 0.$$

This now proves the equality on the left hand side in (63).

(c) If $k = n$, there is nothing to prove. Thus, we assume that $k = n-1$ and without loss of generality, we assume that the last row B_{n*} of B is linearly dependent on the rows $B_{1*}, \ldots, B_{(n-1)*}$. Thus, we have

$$B_{n*} = \sum_{i=1}^{n-1} \lambda_i B_{i*}, \tag{65}$$

for some scalar values $\lambda_1, \ldots, \lambda_{n-1}$. Let f denote the unique linear function with

$$f(B_{i*}) = D_{i*} \quad \text{for all } i = 1, \ldots, n-1.$$

It remains to show that $f(B_{n*}) = D_{n*}$. Applying part (a) to equation (65) shows

$$(BC)_{n*} = \sum_{i=1}^{n-1} \lambda_i (BC)_{i*} = \sum_{i=1}^{n-1} \lambda_i (DF)_{i*},$$

where the last equality holds since $BC = DF$. Moreover,

$$(BC)_{n*} = (DF)_{n*}$$

and again with part (a), we get

$$D_{n*} = \sum_{i=1}^{n-1} \lambda_i D_{i*},$$

with the same scalar values $\lambda_i, \ldots, \lambda_{n-1}$. Finally, we get

$$f(B_{n*}) = f\left(\sum_{i=1}^{n-1} \lambda_i B_{i*}\right) = \sum_{i=1}^{n-1} \lambda_i f\left(B_{i*}\right) = \sum_{i=1}^{n-1} \lambda_i D_{i*} = D_{n*},$$

concluding the case $k = n - 1$.

If $k < n - 1$, we can apply the same technique such that the linear map f exists for any $k \in \{1, \ldots, n\}$. This eventually proves the existence of f in part (c).

(d) Similar to part (c), if $k = m$, there is nothing to prove. We therefore assume that $k = m - 1$ and without loss of generality, we assume that the last column C_{*m} of C is linearly dependent on the columns $C_{*1}, \ldots, C_{*(m-1)}$. Thus, we have

$$C_{*m} = \sum_{j=1}^{m-1} \lambda_j C_{*j}, \tag{66}$$

for some scalar values $\lambda_1, \ldots, \lambda_{m-1}$. Let g denote the unique linear function with

$$g(C_{*j}) = F_{*j} \quad \text{for all } j = 1, \ldots, m-1.$$

It remains to show that $g(C_{*m}) = F_{*m}$. Applying part (b) to equation (66) shows

$$(BC)_{*m} = \sum_{j=1}^{m-1} \lambda_j (BC)_{*j} = \sum_{j=1}^{m-1} \lambda_j (DF)_{*j},$$

where the last equality holds since $BC = DF$. Moreover,

$$(BC)_{*m} = (DF)_{*m}$$

and again with part (b), we get

$$F_{*m} = \sum_{j=1}^{m-1} \lambda_j F_{*j},$$

with the same scalar values $\lambda_j, \ldots, \lambda_{m-1}$. Finally, we get

$$g(C_{*m}) = g\left(\sum_{j=1}^{m-1} \lambda_j C_{*j}\right) = \sum_{j=1}^{m-1} \lambda_j g\left(C_{*j}\right) = \sum_{j=1}^{m-1} \lambda_j F_{*j} = F_{*m},$$

concluding the case $k = m - 1$.

If $k < m - 1$, we can apply the same technique such that the linear map g exists for any $k \in \{1, \ldots, m\}$. This eventually proves the existence of g in part (d).

□

Now we can prove the following general result.

Lemma 8.10. *Let $B, D \in \mathbb{R}^{n \times k}$, both of rank k, and $C, F \in \mathbb{R}^{k \times m}$, again both of rank k. Then we have $BC = DF$ if and only if there exists a nonsingular matrix $Q \in \mathbb{R}^{k \times k}$ such that $BQ = D$ and $Q^{-1}C = F$.*

Proof. We will prove both directions separately. For the if part, let $Q \in \mathbb{R}^{k \times k}$ be nonsingular with $BQ = D$ and $Q^{-1}C = F$. Then we have $DF = BQQ^{-1}C = BC$. For the reverse part, let B_{i*} resp. D_{i*} denote the i-th rows of B resp. D. Further let $R(B) \subseteq \mathbb{R}^k$ resp. $R(D) \subseteq \mathbb{R}^k$ denote the subspaces spanned by the rows of B and D, respectively. Since B and D are of the same rank, Lemma 8.9 (c) shows that there exists a linear map $f : R(B) \to R(D),\ x^T \mapsto x^T A_f$ such that $f(B_{i*}) = B_{i*}A_f = D_{i*}$ for all $i \in \{1, \ldots, n\}$. Moreover, for $i = 1, \ldots, m$, let C_{*j} resp. F_{*j} denote the columns of C resp. F. Further let $S(C) \subseteq \mathbb{R}^k$ resp. $S(F) \subseteq \mathbb{R}^k$ denote the subspaces spanned by the columns of C and F, respectively. Since C and F are of the same rank, Lemma 8.9 (d) shows that there exists a linear map $g : S(C) \to S(F),\ y \mapsto A_g y$ such that $g(C_{*j}) = A_g C_{*j} = F_{*j}$ for all $j \in \{1, \ldots, m\}$. Due to the equality $BC = DF$, we have

$$B_{i*}C_{*j} = D_{i*}F_{*j} \text{ for every } i \in \{1, \ldots, n\} \text{ and } j \in \{1, \ldots, m\}. \tag{67}$$

Furthermore, since $\operatorname{rank}(B) = \operatorname{rank}(D) = k$, the matrix A_f is nonsingular and f is bijective, such that we have for every i, j

$$B_{i*}C_{*j} = f^{-1}(D_{i*})C_{*j} = D_{i*}A_f^{-1}C_{*j} \text{ and } D_{i*}F_{*j} = D_{i*}A_gC_{*j}.$$

By equation (67), we therefore get

$$D_{i*}A_f^{-1}C_{*j} = D_{i*}A_gC_{*j}, \tag{68}$$

for every $i = 1, \dots, n$ and $j = 1, \dots, m$. Moreover, C is of full row-rank such that the Moore-Penrose-inverse C^+ is a right inverse of C, cf. Lemma A.3. The matrix D is of full column-rank such that D^+ is a left inverse of D. Thus, equation (68) can be rewritten as

$$DA_f^{-1}C = DA_gC \quad \Leftrightarrow \quad D^+DA_f^{-1}CC^+ = D^+DA_gCC^+ \quad \Leftrightarrow \quad A_f^{-1} = A_g,$$

such that A_f is the inverse of A_g and vice versa, which completes the proof. □

This lemma therefore shows that instead of using orthogonal matrices to transform one matrix factorization of type $A = BC$ into another one, we will apply the generalized result of Lemma 8.10 and we will therefore work with nonsingular matrices. It may become important to make sure that a given matrix is nonsingular. To ensure this, we can use the following approach.

Lemma 8.11. *Consider a singular matrix $A \in \mathbb{R}^{n\times n}$ and its singular value decomposition $A = U\Sigma V^T$. Then there exists $\tilde{\Sigma} \in \mathbb{R}^{n\times n}$ such that $U\tilde{\Sigma}V^T$ is nonsingular.*

Proof. Since A is singular, some of the diagonal entries of Σ are equal to zero. Further let $\sigma_{\min} := \min\{\Sigma_{ii} \mid \Sigma_{ii} \neq 0\}$. Then we construct a diagonal matrix $\tilde{\Sigma}$ with

$$\tilde{\Sigma}_{ii} = \begin{cases} \Sigma_{ii}, & \text{if } \Sigma_{ii} > 0 \\ \sigma_{\min}, & \text{if } \Sigma_{ii} = 0. \end{cases}$$

Having this, we define $\tilde{A} := U\tilde{\Sigma}V^T$ and it follows that

$$\det(\tilde{A}) = \det(U\tilde{\Sigma}V^T) = \underbrace{\det(U)}_{\in\{-1,1\}}\underbrace{\det(\tilde{\Sigma})}_{>0}\underbrace{\det(V^T)}_{\in\{-1,1\}} \neq 0$$

and $\tilde{A}$ is therefore nonsingular. □

This regularization is optimal in the sense that A is the best low-rank approximation of $\tilde{A}$ of fixed rank according to Theorem 8.2.

Now we can formulate the main result of this section, which will motivate the algorithm for nonnegative matrix factorization later on. For this, we define the polyhedral cones

$$\begin{aligned} &\mathcal{P}_B := \{Q \in \mathbb{R}^{k\times k} \mid B_kQ \geq 0\} \text{ and} \\ &\mathcal{P}_C := \{Q \in \mathbb{R}^{k\times k} \mid Q \text{ is nonsingular and } Q^{-1}C_k \geq 0\}. \end{aligned} \tag{69}$$

Based on these definitions, we now get the following theorem.

Theorem 8.12. *Let $A \in \mathbb{R}^{n\times m}$ and $k \leq \min\{n, m\}$. Consider the best rank-k approximation A_k of A and its initial factorization $A_k = B_k C_k$ as introduced in* (61)*. Then, to obtain a nonnegative matrix factorization of A of rank k, it is sufficient to find a matrix $Q \in \mathbb{R}^{k\times k}$ in the intersection $\mathcal{P}_B \cap \mathcal{P}_C$.*

Proof. If $Q \in \mathcal{P}_B \cap \mathcal{P}_C$, we know that Q is nonsingular by definition of $\mathcal{P}_C$. Moreover, we have

$$A_k = B_k C_k = (B_k Q)(Q^{-1} C_k).$$

By definition of $\mathcal{P}_B$ and $\mathcal{P}_C$, we get with Lemma 8.7:

$$A_k = \underbrace{(B_k Q)}_{\geq 0} \underbrace{(Q^{-1} C_k)}_{\geq 0}$$

is a nonnegative matrix factorization of A, completing the proof. □

We can therefore reduce the problem of finding a nonnegative matrix factorization to solving the problem

$$\text{find} \quad Q \in \mathcal{P}_B \cap \mathcal{P}_C, \tag{70}$$

as a generalization of the problem in (55).

We can now extend the results for the alternating projections method to generate cp-factorizations, presented in Chapter 6, to the framework of nonnegative matrix factorization. Here we will see in the following sections that it is possible to derive an algorithmic approach to obtain a nonnegative matrix factorization, based on the tools we introduced to obtain cp-factorizations.

8.2.2 Exact Projection Algorithm for Nonnegative Matrix Factorization

There already exist several update rules to compute a nonnegative matrix factorization. A survey on several update rules and algorithms can be found for example in [47, Chapter 6], [59, Chapter 3] or in [69, Chapters 4-5].

For our approach, we will go back to the problem in (70) to find a matrix Q in the intersection $\mathcal{P}_B \cap \mathcal{P}_C$. Here a first thing to analyse is how to project onto the sets $\mathcal{P}_B$ and $\mathcal{P}_C$ introduced in (69). Here we can use the result in Lemma 6.1, proving that the projection onto $\mathcal{P}_B$ is unique and can be computed by solving a certain second order cone problem.

Unfortunately, this does not hold for the set $\mathcal{P}_C$ since this set is not even closed (due to the fact that the set of nonsingular matrices is not closed), such that we can not project onto $\mathcal{P}_C$.

In the following, we will show that it is possible to apply a different approach to obtain a matrix in the set $\mathcal{P}_C$. For this, we will extend the results in Section 6.2 to the framework of nonnegative matrix factorization. This generates a modified and applicable algorithm, which is introduced in the following section.

8.2.3 Modified Algorithm for Nonnegative Matrix Factorization

To obtain an applicable algorithm to compute a general nonnegative matrix factorization, we will start with some initial nonsingular matrix Q_0. Similar to the method in Section 6.2, we will use the projection of the product $B_k Q_0$ onto the nonnegative orthant, where B_k is as defined in (61). So, we define the matrix $D \in \mathbb{R}^{n\times k}$ entrywise as

$$D_{ij} := \max\{(B_k Q_0)_{ij}, 0\} \quad \text{for all } i = 1, \dots, n \text{ and } j = 1, \dots, k. \tag{71}$$

Due to the lack of symmetry in this setting, we also need to project $Q_0^{-1} C_k$ onto the nonnegative orthant, where C_k is as defined in (61). Dropping the inverse of the matrix first, we define the matrix $F \in \mathbb{R}^{k\times m}$ entrywise as

$$F_{ij} := \max\{(Q_0 C_k)_{ij}, 0\} \quad \text{for all } i = 1, \dots, k \text{ and } j = 1, \dots, m. \tag{72}$$

Since the nonnegative orthant of the matrix space $\mathbb{R}^{n\times k}$ (resp. $\mathbb{R}^{k\times m}$) is convex, these projections are always unique. Moreover, it will be necessary to solve the equations $B_k X = D$ for X and $Y C_k = F$ for Y. For this, we will apply a method which is similar to the approach in Lemma 6.7. To be more precise, we have the following result.

Lemma 8.13. *Let B_k and C_k be as introduced in* (61). *Further let D, F be as defined in* (71) *and* (72). *Consider the equation $B_k X = D$. Then $\|B_k X - D\|_F$ is minimal if and only if $X = B_k^+ D$. Furthermore, consider the equation $Y C_k = F$. Then $\|Y C_k - F\|_F$ is minimal if and only if $Y = F {C_k}^+$. Here B_k^+ respectively C_k^+ denotes the Moore-Penrose-inverse of B_k and C_k, respectively.*

Proof. We will prove this result separately for B_k and C_k. First, we focus on the equation $B_k X = D$ and assume that there exists a solution X. According to Lemma 6.7, the complete set of solutions of this equation is the set

$$\left\{ X = {B_k}^+ D + (I - B_k^+ B_k) A \;\middle|\; A \in \mathbb{R}^{k\times k} \right\} \subseteq \mathbb{R}^{k\times k}. \tag{73}$$

Moreover, we know that $\operatorname{rank}(A) \geq k$ and by construction, this implies $\operatorname{rank}(B_k) = k$. Thus, B_k is of full column rank and due to the properties of the Moore-Penrose-inverse, we get $B_k^+ B_k = I$, such that the solution set in (73) boils down to the singleton $\{B_k^+ D\}$. In this case, we therefore have $\|B_k X - D\|_F = 0$ if and only if $X = B_k^+ D$. On the other hand, for the case where there doest not exist a solution X of $B_k X = D$, Lemma 6.7 shows that the residual $\|B_k X - D\|_F$ is minimal if and only if $X = B_k^+ D$.

If we focus on the equation $Y C_k = F$ on the other hand, we know that the complete set of solutions is given as the set

$$\left\{ Y = F C_k^+ + A(I - C_k {C_k}^+) \;\middle|\; A \in \mathbb{R}^{k\times k} \right\} \subseteq \mathbb{R}^{k\times k}, \tag{74}$$

again due to Lemma 6.7. Similar to the first case, we have that $C_k \in \mathbb{R}^{k\times m}$ is of rank k

and therefore of full row rank. This yields $C_k C_k^+ = I$ due to the properties of the Moore-Penrose-inverse. Hence, the solution set in (74) boils down to the singleton $\{FC_k^+\}$. Thus, $\|YC_k - F\|_F = 0$ if and only if $Y = FC_k^+$. For the case where there doest not exist a solution Y of $YC_k = F$, the statement follows again by applying Lemma 6.7. □

With this result, we can now give an approximation to Q in $\mathcal{P}_B$ respectively $\mathcal{P}_C$, which can be computed easily.

Lemma 8.14.

(a) Let D be the projection of B_kQ onto $\mathbb{R}_+^{n\times k}$. If $D = B_kQ$, then $Q = P_{\mathcal{P}_B}(Q) \in \mathcal{P}_B$. If on the other hand $D \neq B_kQ$ and the equation $B_kX = D$ is solvable for X, then $B_k^+D \in \mathcal{P}_B$.

(b) Let F be the projection of QC_k onto $\mathbb{R}_+^{k\times m}$. If $QC_k = F$ and Q is nonsingular, then $Q^{-1} = P_{\mathcal{P}_C}(Q^{-1}) \in \mathcal{P}_C$. On the other hand, let $F \neq QC_k$ and assume that the matrix FC_k^+ is nonsingular. Further assume that the equation $YC_k = F$ is solvable for Y. Then we have $(FC_k^+)^{-1} \in \mathcal{P}_C$.

Proof. We will prove both parts separately.

(a) If $D = B_kQ$, then $B_kQ \geq 0$, i.e., $Q \in \mathcal{P}_B$ and Q is therefore its projection onto $\mathcal{P}_B$. Otherwise, let $D \neq B_kQ$ and assume that the equation $B_kX = D$ is solvable for X. Then Lemma 8.13 yields $X = B_k^+D$. Therefore, B_k^+D is the projection of Q onto the set $\{X \in \mathbb{R}^{k\times k} \mid B_kX = D\}$. Since $D \geq 0$, we get $B_k^+D \in \mathcal{P}_B$.

(b) If $QC_k = F$ and Q is nonsingular, then $QC_k \geq 0$ and $Q^{-1} \in \mathcal{P}_C$ by definition. Moreover, in this case, Q^{-1} is its projection onto $\mathcal{P}_C$. Otherwise, let $F \neq QC_k$ and assume that $YC_k = F$ has a solution Y. Hence, $Y = FC_k^+$ due to Lemma 8.13. Thus, FC_k^+ is the projection of Q onto the set $\{Y \in \mathbb{R}^{k\times k} \mid YC_k = F\}$. Since FC_k^+ is nonsingular by assumption and $F \geq 0$ by definition, we get $(FC_k^+)^{-1} \in \mathcal{P}_C$.

□

In addition, if the equation $B_kX = D$ does not have a solution, then $X := B_k^+D$ minimizes the residual $\|B_kX - D\|_F$. In this case, we get $B_kX = B_kB_k^+D$. Here $B_kB_k^+ \neq I$ in general since B_k is not of full row-rank. Thus, it may happen that $B_k^+D \notin \mathcal{P}_B$, however this does not seem to impair the good numerical performance.

If on the other hand the equation $YC_k = F$ does not have a solution, we get with Lemma 8.13 that $Y := FC_k^+$ minimizes the residual $\|YC_k - F\|_F$. Thus, $YC_k = FC_k^+C_k$. In this equation we have $C_k^+C_k \neq I$ in general since C_k is not of full column rank. Hence, even if FC_k^+ is nonsingular, it may happen that $(FC_k^+)^{-1} \notin \mathcal{P}_C$. However, this does not seem to impair the good numerical performance either.

If we combine the results in Lemmas 8.13 and 8.14, it is possible to derive matrices in $\mathcal{P}_B$ or $\mathcal{P}_C$ without solving an SOCP. Moreover, in Lemma 8.14, we assumed that FC_k^+ is a nonsingular matrix. This is equivalent to a certain rank assumption for F, as the following lemma shows. Here in addition, a similar result for B_k^+D holds and will be used for the algorithmic approach.

Lemma 8.15.

(a) In our setting, the matrix $FC_k^+ \in \mathbb{R}^{k\times k}$ is nonsingular if and only if $F \in \mathbb{R}^{k\times m}$ is of rank k.

(b) In addition, the matrix $B_k^+ D \in \mathbb{R}^{k\times k}$ is nonsingular if and only if $D \in \mathbb{R}^{n\times k}$ is of rank k.

Proof. (a) First observe that by Sylvester's inequality, cf. [12, Corollary 2.5.10], we have

$$\operatorname{rank}(F) + \operatorname{rank}(C_k^+) - k \leq \operatorname{rank}(FC_k^+) \leq \min\{\operatorname{rank}(F), \operatorname{rank}(C_k^+)\}.$$

Since $\operatorname{rank}(C_k^+) = \operatorname{rank}(C_k) = k$, this yields

$$\operatorname{rank}(F) \leq \operatorname{rank}(FC_k^+) \leq \min\{\operatorname{rank}(F), k\}. \tag{75}$$

Now observe that FC_k^+ is nonsingular if and only if $\operatorname{rank}(FC_k^+) = k$. Due to (75), this is true if and only if $\operatorname{rank}(F) = k$.

(b) Analogue to the first part, since $\operatorname{rank}(B_k^+) = k$, Sylvester's inequality yields

$$\operatorname{rank}(D) \leq \operatorname{rank}(B_k^+ D) \leq \min\{k, \operatorname{rank}(D)\},$$

such that $\operatorname{rank}(B_k^+ D) = k$ if and only if $\operatorname{rank}(D) = k$.

□

From now on, we will take $B_k^+ D$ respectively $(FC_k^+)^{-1}$ as an approximation of $P_{\mathcal{P}_B}(Q)$, respectively $P_{\mathcal{P}_C}(Q)$. This reasoning leads to Algorithm 9.

Algorithm 9 Modified algorithm for nonnegative matrix factorization

Input: $A \in \mathbb{R}^{n\times m}$ with its singular value decomposition $A = U\Sigma V^T$; $k \leq \min\{n, m\}$; initial nonsingular matrix $Q_0 \in \mathbb{R}^{k\times k}$

$A_k \leftarrow \sum_{i=1}^{k} \sigma_i u_i v_i^T$
$[U_k, \Sigma_k, V_k] \leftarrow \text{svd}(A_k)$
$B_k \leftarrow U_k\sqrt{\Sigma_k} \in \mathbb{R}^{n\times k}$ and $C_k \leftarrow \sqrt{\Sigma_k}V_k^T \in \mathbb{R}^{k\times m}$
$i \leftarrow 0$
while $B_kQ_i \not\geq 0$ or $Q_i^{-1}C_k \not\geq 0$ **do**
  $D \leftarrow \max\{B_kQ_i, 0\}$ entrywise
  $Q_D \leftarrow {B_k}^+ D$
  if $\operatorname{rank}(D) < k$ **then**
    Regularize Q_D
  end if
  $F \leftarrow \max\{Q_D^{-1}C_k, 0\}$ entrywise
  $Q_F \leftarrow FC_k{}^+$
  if $\operatorname{rank}(F) < k$ **then**
    Regularize Q_F
  end if
  $Q_{i+1} \leftarrow Q_F^{-1}$
  $i \leftarrow i + 1$
end while

Output: Nonsingular matrix Q_i and a nonnegative matrix factorization $(B_kQ_i)(Q_i^{-1}C_k)$ of A.

In Algorithm 9, the first 3 steps calculate exactly the initial factorization $A = B_k C_k$ as introduced in (61). In step 5, the main while loop of the algorithm starts. As long as our factorization matrices $B_k Q_i$ and $Q_i^{-1} C_k$ have some negative entries, we at first define D as the entrywise maximum introduced in (71). In step 7, we then look for a solution Q_D of $\min \|B_k Q_D - D\|_F$. As shown in Lemma 8.14, we know that $B_k^+ D$ is the unique solution. Moreover, if the equation $B_k Q = D$ is solvable for Q, we get that $B_k^+ D \in \mathcal{P}_B$. To obtain a matrix in $\mathcal{P}_B \cap \mathcal{P}_C$, and since the set $\mathcal{P}_C$ is based on the inverse of the considered matrices, we check in step 8 whether $B_k^+ D$ is nonsingular, based on the result in Lemma 8.15. If this is not the case, we use the regularization approach in Lemma 8.11 to make sure that the generated matrix is nonsingular and we can compute its inverse. Using this inverse matrix as the matrix Q_0 in (72), we obtain the matrix F in step 11 as the introduced entrywise maximum. In step 12, we then look for a solution Q_F of the problem $\min \|Q_F C_k - F\|_F$. Due to Lemma 8.14, the solution is given as $F C_k^+$. Then in step 13, we check whether the matrix $F C_k^+$ is nonsingular, again based on the result in Lemma 8.15. According to Lemma 8.14, we know that if $F C_k^+$ is nonsingular and the equation $Q C_k = F$ is solvable, we have $(F C_k^+)^{-1} \in \mathcal{P}_C$. Then we take $(F C_k^+)^{-1}$ as our next iterate Q_{i+1}. If on the other hand the matrix $F C_k^+$ is singular, we again apply the regularization technique in Lemma 8.11 to obtain the inverse of the regularized matrix as our next iterate Q_{i+1} in step 16. Then a new loop starts as long as we have not reached a nonnegative matrix factorization.

For the numerical implementation, we further add a maximum number of iterations to make sure that the algorithm terminates. In the following section, we will have a closer look at the numerical performance of Algorithm 9.

8.2.4 Numerical Results for Nonnegative Matrix Factorization

As before, the numerical results were carried out on a computer with 88 Intel Xenon ES-2699 cores (2.2 Ghz each) and a total of 0.792 TB RAM. Algorithm 9 was implemented in MatlabR2017a. For randomly generated matrices, again the Matlab command `randn` was used. Numerically, the algorithm terminates successfully if it returns matrices $(B_k Q_i)$ and $(Q_i^{-1} C_k)$ which are entrywise greater than or equal to -10^{-12} and it terminates without success if these conditions are not fulfilled and the maximum number of iterations is reached.

For a first example, consider the following randomly generated matrix

$$A = \begin{pmatrix} 7 & 3 & 5 & 4 & 13 & 1 & 1 & 4 \\ 6 & 2 & 2 & 4 & 10 & 16 & 8 & 2 \\ 9 & 11 & 1 & 9 & 9 & 5 & 3 & 14 \\ 3 & 10 & 1 & 13 & 19 & 7 & 0 & 13 \\ 21 & 4 & 2 & 2 & 20 & 6 & 4 & 3 \end{pmatrix} \in \mathbb{R}^{5\times 8} \tag{76}$$

and the low-rank parameter k, fixed to 3 in this example. Then at first, we compute the best rank-3

approximation A_3 to A, as shown in Theorem 8.2. We get

$$A_3 = \begin{pmatrix} 8.9667 & 4.2210 & 1.5947 & 3.5945 & 11.1316 & 1.6312 & 0.8958 & 4.8216 \\ 6.0643 & 2.0481 & 1.4711 & 3.9834 & 10.0323 & 16.0878 & 7.8358 & 1.9519 \\ 5.5122 & 8.9454 & 1.4337 & 9.6742 & 13.5665 & 4.7817 & 1.0274 & 11.5364 \\ 5.1814 & 11.2683 & 1.5752 & 12.5851 & 15.9546 & 7.0005 & 1.5595 & 14.6929 \\ 20.5074 & 3.6837 & 3.3835 & 2.1058 & 20.3494 & 5.7566 & 4.2303 & 2.8895 \end{pmatrix}.$$

To generate a nonnegative matrix factorization with $k = 3$ for the matrix A, we start with the following initial factorization, based on the technique introduced in (61). More precisely, we have $A_3 = B_3C_3$ with

$$A_3 = \underbrace{\begin{pmatrix} -2.2164 & 0.3624 & 1.0742 \\ -2.4818 & 1.1425 & -3.2988 \\ -3.1257 & -1.7305 & 0.3226 \\ -3.7949 & -2.4724 & -0.0604 \\ -3.8469 & 2.8992 & 1.3067 \end{pmatrix}}_{B_3 \in \mathbb{R}^{5\times 3}} \underbrace{\begin{pmatrix} -3.0116 & -1.9850 & -0.6127 & -2.0773 & -4.6122 & -2.1422 & -0.9359 & -2.3673 \\ 2.4952 & -1.5193 & 0.3004 & -1.8944 & 0.6106 & 0.5318 & 0.8395 & -2.3186 \\ 1.2916 & 0.3463 & 0.1190 & -0.3008 & 0.6402 & -3.0811 & -1.3805 & 0.3863 \end{pmatrix}}_{C_3 \in \mathbb{R}^{3\times 8}}.$$

Since $\operatorname{rank}(A) = 5$, we have $\operatorname{rank}(B_3) = \operatorname{rank}(C_3) = 3$ by definition. As an initial nonsingular matrix, we take

$$Q_0 = \begin{pmatrix} 0.4397 & 0.0464 & 0.2059 \\ 0.3518 & 0.8796 & 0.0828 \\ 0.2594 & 0.3400 & 0.4412 \end{pmatrix}.$$

Based on these input variables, Algorithm 9 takes 153 iterations and 0.0144 seconds to provide entrywise nonnegative matrices $\widetilde{B_3}$ and $\widetilde{C_3}$ such that

$$A_3 = \underbrace{\begin{pmatrix} 0.7170 & 0.0226 & 0.0047 \\ 0.2196 & 1.3784 & 0.0020 \\ 0.4051 & 0.1073 & 0.0128 \\ 0.3506 & 0.2145 & 0.0165 \\ 1.5761 & 0.4598 & 0.0014 \end{pmatrix}}_{\widetilde{B_3} \in \mathbb{R}_+^{5\times 3}} \underbrace{\begin{pmatrix} 12.2890 & 1.6756 & 1.9019 & 0.1499 & 10.9775 & 0.1221 & 1.0755 & 1.0565 \\ 2.4108 & 0.2940 & 0.6981 & 1.8080 & 4.5600 & 11.2559 & 5.5133 & 0.0000 \\ 21.5553 & 644.1498 & 46.0188 & 736.7893 & 675.0740 & 275.6546 & 0.0000 & 868.9005 \end{pmatrix}}_{\widetilde{C_3} \in \mathbb{R}_+^{3\times 8}}.$$

Due to Theorem 8.2, we have

$$\|A - A_3\|_F = \|A - \widetilde{B_3}\widetilde{C_3}\|_F \leq \|A - X\|_F,$$

for every matrix $X \in \mathbb{R}^{n\times m}$ with $\operatorname{rank}(X) \leq 3$. Combined with the fact that the matrices $\widetilde{B_3}$ and $\widetilde{C_3}$ are entrywise nonnegative, we eventually get that $\widetilde{B_3}\widetilde{C_3}$ is the desired nonnegative matrix factorization of A of rank 3.

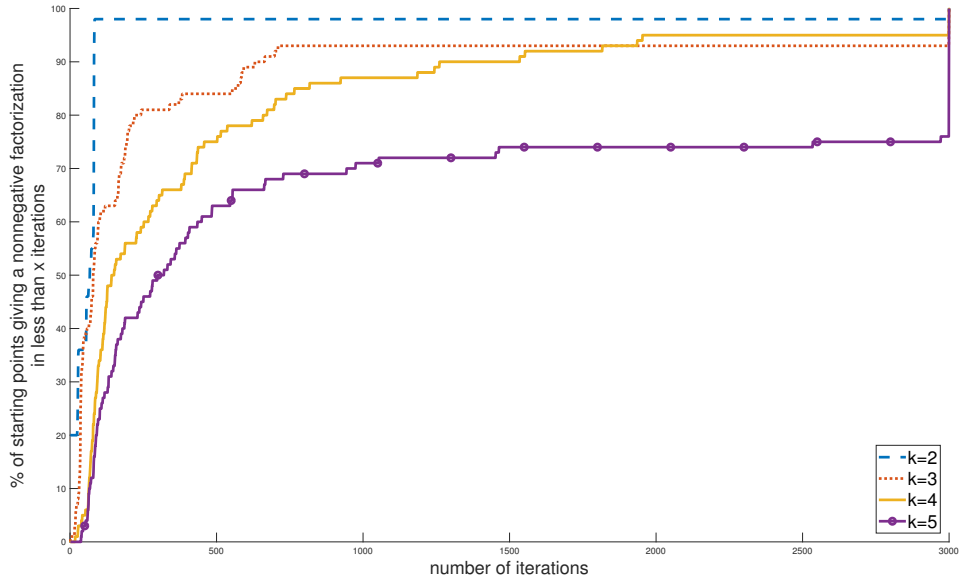
Moreover, in the following experiment, we will have a closer look at the influence of the parameter k for a given matrix $A \in \mathbb{R}^{m\times n}$.

To this end, we consider the matrix

$$A = \begin{pmatrix} 16 & 40 & 29 & 9 & 42 & 36 & 24 & 26 \\ 19 & 41 & 30 & 11 & 26 & 31 & 22 & 30 \\ 24 & 34 & 50 & 36 & 25 & 42 & 41 & 48 \\ 13 & 24 & 26 & 25 & 16 & 34 & 28 & 35 \\ 9 & 39 & 29 & 18 & 19 & 39 & 19 & 38 \end{pmatrix} \in \mathbb{R}^{5\times 8}.$$

This matrix is generated as the product of two entrywise nonnegative matrices $A_1 \in \mathbb{N}^{5\times 5}$ and $A_2 \in \mathbb{N}^{5\times 8}$, which are randomly generated. For the experiment, we test the performance of 100 starting points Q_0 with a maximum of 3000 iterations per starting point in Algorithm 9 for different values of k. The results are collected in Figure 8.5.

Figure 8.5: Success rate of Algorithm 9 for a given matrix of order 5×8 and different values of k.



As we notice in Figure 8.5, the performance of Algorithm 9 depends on the choice of k. Whereas for $k = 2$ nearly every starting point returns a nonnegative matrix factorization, the success rate decreases for higher values of k. Taking $k = 4$ still returns a success rate of more than 90%. For $k = 5$, it is still possible to derive a nonnegative matrix factorization of A. This therefore shows that it is not necessary to add a low-rank constraint to obtain a nonnegative matrix factorization. Nevertheless, in most applications, the low-rank approach is part of the nonnegative matrix factorization. Figure 8.5 therefore substantiates the good performance of Algorithm 9 for different choices of k for the same matrix.

Clearly, the performance depends on the best rank-k approximation A_k of A, as the following remark states.

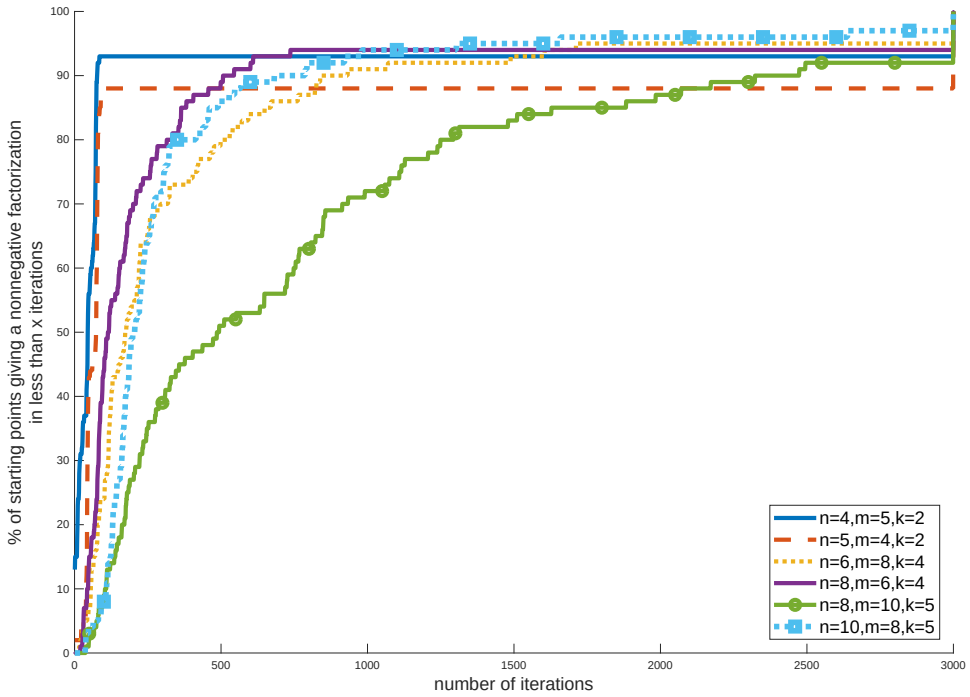
Remark 8.16. *Consider the best rank-k approximation A_k of the given matrix A. Then Algorithm 9 will terminate successfully for the parameter k only if there exists an exact factorization $A_k = B_k C_k$ with $B_k \in \mathbb{R}_+^{n\times k}$ and $C_k \in \mathbb{R}^{k\times m}$. Clearly, this can only be true if $A_k \in \mathbb{R}_+^{n\times m}$.*

Similar to the symmetric case of nonnegative matrix factorization in Section 8.1.2, we will analyse the influence of the order of the given matrix A on the performance of Algorithm 9. To this end, we consider randomly generated matrices $A \in \mathbb{R}^{n\times m}$ for different values of n and m. The instances are generated as follows: Given the values n, m, we define $l := \min\{n, m\}$ and with this, we construct matrices $B \in \mathbb{R}^{n\times l}$ and $C \in \mathbb{R}^{l\times m}$ using the Matlab command `randn`.

Then we take the absolute value of the entries of B and C to make sure that the generated matrices are entrywise nonnegative. Eventually, we define $A = BC$ as our test matrix A.

For each matrix generated this way, we set $k = \lfloor 0.7l \rfloor$ and analyse the success rate of 100 randomly generated initial nonsingular matrices Q_0. For every Q_0, we allow at most 3000 iterations. The performance of Algorithm 9 in this setting is illustrated in Figure 8.6.

Figure 8.6: Success rate of Algorithm 9 for matrices of variable order $n \times m$ and k fixed to $\lfloor 0.7 \cdot \min\{n, m\} \rfloor$.



Here a first thing to mention is that the algorithm terminates successfully in every case. So, the success in total is independent of the order of the initial matrix A. Moreover, it does not make a difference if $m > n$ or $m < n$. Especially for small dimensions like $n = 4,\ m = 5$ and $n = 5,\ m = 4$, plotted as the solid blue and the dashed red line, it turns out that the algorithm terminates successfully for around 90% of the initial nonsingular matrices. Furthermore, if the algorithm terminates successfully for one of the initial nonsingular matrices, a nonnegative matrix factorization is provided in less than 100 iterations. Increasing values of m and n do not seem to influence the success rate in less then 3000 iterations, but it takes more iterations on average to return a nonnegative matrix factorization. Here especially the green line, representing $n = 8,\ m = 10$, illustrates this behaviour. Altogether, Figure 8.6 shows that the here presented method to derive nonnegative matrix factorizations works well for matrices A, which are generated via the above mentioned approach. Clearly, this proves that the generalized techniques in Section 8.2.1, which are motivated by the techniques in Section 6.1 to obtain cp-factorizations, can be used in Algorithm 9 to derive nonnegative matrix factorizations of given rank k.

In the last section, we will now collect the results and approaches introduced in this thesis.

9 Conclusion and Further Remarks

In this thesis, we introduced methods to derive completely positive and nonnegative matrix factorizations. To be more precise, in the first part, we saw some fundamental facts on the completely positive matrix cone and especially some applications of completely positive programming. For these approaches, it is necessary to prove whether a given matrix is completely positive, a task which is known to be NP-hard. Moreover, we saw that beside proving the membership to the completely positive cone, having an explicit cp-factorization is important to obtain the optimal solution in some applications. Not least because of this motivation, we introduced some relevant definitions and properties for factorizations of completely positive matrices, here notably the cp-rank.

Based on the fact that factorizations for completely positive matrices are not unique, we analysed in Chapter 3 the relation of different cp-factorizations of the same matrix. As it turned out, orthogonal matrices can be used to transform one cp-factorization into another one. Nevertheless, for this result, it is necessary to have two factorizations of equal order. In this context we saw that it is possible to extend the number of columns in a given cp-factorization arbitrarily, without loosing the key properties.

This approach can also be applied to any factorization $A = BB^T$, where B is not necessarily entrywise nonnegative. Based on this fact, we could show that proving the membership to the completely positive cone boils down to a feasibility problem of two intersecting sets in Chapter 4. Furthermore, we saw that these results can also be extended to prove the membership of a matrix to the interior of the completely positive cone.

The resulting feasibility problems now motivated to study the well known approach of alternating projections to obtain a point in the intersection of two sets. Here an introduction to this topic for several types of sets was given. We especially focused on convergence results and results on the convergence rate of this approach in several settings. Considering more than two sets led us to the cyclic projections approach. Here again, some known facts on the convergence rate were given. Moreover, in Section 5.4, we saw that the convergence results for alternating projections between manifolds can be extended to convergence results for cyclic projections among a sequence of manifolds. We further saw, considering the results in [41], that alternating projections can also be applied to semialgebraic sets.

This fact was then used to construct a first algorithm to prove the membership of a given matrix to the completely positive cone in Chapter 6. Furthermore, it was possible to derive a local convergence result for this approach. For this first algorithm, it was necessary to solve a second order cone problem in every projection step. Although the solution of these problems can be obtained in polynomial time, it was possible to modify the first algorithm in order to achieve even lower computation times. Another possibility to decrease the computation time, which was not shown in this

thesis, would be to use parallel computing. Here it should be mentioned that both Algorithms 1 and 2 are easy to run in parallel since for a given starting point the computation runs completely independent of any other run. The same holds for the other algorithms introduced in this thesis, which are based on the methods in Algorithm 1 or 2.

Nevertheless, we saw in Chapter 7 that especially the modified approach works very fast in most instances. Again we distinguished between matrices at the boundary and in the interior of the completely positive cone. Whereas for the interior the algorithms terminated for any instance, the algorithms could not provide a cp-factorization for certain matrices at the boundary. Nevertheless, slight perturbations of these matrices were sufficient to ensure a successful termination. All in all, it was possible to factorize matrices of order up to 2000.

As we showed in Chapter 8, it is possible to use the approaches for cp-factorizations also in the setting of nonnegative matrix factorizations, albeit with some further adjustments. Especially in the context of symmetric nonnegative matrix factorization, we saw that the generated results extend to this setting in a very similar way. For the case of general nonnegative matrix factorization on the other hand, it was necessary to replace orthogonal matrices with nonsingular matrices to obtain a factorization algorithm. Furthermore, we saw that the resulting feasibility problems are now based on nonclosed sets, such that it became necessary to compute approximations to these sets in order to obtain an applicable algorithm. Numerical experiments showed that it is possible to use the key tools of the completely positive factorization approaches as well in this setting.

Thus, the presented methods throughout this thesis can be applied to various settings, whenever a completely positive or nonnegative factorization is needed.

Appendix: Singular Value Decomposition and Pseudoinverse Matrices

For the reader's convenience, we will have closer look at some properties of the singular value decomposition and the Moore-Penrose-inverse in this appendix. For a short survey on this area and for small dimensional concrete examples, the reader is for example referred to [49].

First, we define the singular values of a matrix and its singular value decomposition, cf. [12, Definition 5.6.1 and Theorem 5.6.3].

Theorem A.1. *Let $A \in \mathbb{R}^{n\times m}$ be nonzero and let $r = \operatorname{rank}(A)$. Then the eigenvalues of AA^T are nonnegative and they are equal to the eigevalues of A^TA. Hence, the square roots of the eigenvalues are real numbers.*

The singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_{\min\{n,m\}} = 0$ of A are then defined as the square root of the eigenvalues of A^TA and AA^T.

For the so called singular value decomposition, there exist orthogonal matrices $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{m\times m}$ such that

$$A = U \begin{pmatrix} \Sigma_{r\times r} & 0_{r\times(m-r)} \\ 0_{(n-r)\times r} & 0_{(n-r)\times(m-r)} \end{pmatrix} V^T,$$

where $\Sigma_{r\times r} = \operatorname{Diag}(\sigma_1, \ldots, \sigma_r) \in \mathbb{R}^{r\times r}$ is a diagonal matrix containing the positive singular values of A and $0_{k\times l}$ is a zero matrix of order $k \times l$.

For the singular value decomposition of A, we shortly write $A = U\Sigma V^T$. This can also be written as

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T,$$

where $\operatorname{rank}(A) = r$ and u_i respectively v_i is the i-th column of the matrix U respectively V for every $i \in \{1, \ldots, r\}$.

The singular value decomposition can be used to derive the best rank-k approximation, in Frobenius-norm sense, to a given matrix $A \in \mathbb{R}^{n\times m}$ with $k \leq \min\{m, n\}$. This result, known as the Eckart-Young-Mirsky Theorem, can also be found more detailed in Theorem 8.2.

Theorem A.2. *Let $A \in \mathbb{R}^{n\times m}$ be of rank r and consider its singular value decomposition $A = U\Sigma V^T$. Thus, A can be written as*

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T,$$

where u_i respectively v_i is the i-th column of the matrix U respectively V for every $i \in \{1, \dots, r\}$. Then for $k \leq r = \operatorname{rank}(A)$, the best rank-$k$ approximation (in the Frobenius norm) of A is given by

$$A_k := \sum_{i=1}^{k} \sigma_i u_i v_i^T.$$

To be more precise, we have

$$A_k = \operatorname{argmin} \left\{ \|A - X\|_F^2 \;\middle|\; X \in \mathbb{R}^{n \times m} \text{ with } \operatorname{rank}(X) \leq k \right\},$$

with corresponding minimal value

$$\|A - A_k\|_F^2 = \sum_{i > k+1} \sigma_i^2.$$

Moreover, if $\sigma_k > \sigma_{k+1}$, then A_k is the unique global minimizer.

In the following, we will see that the singular value decomposition can also be used to derive the so called Moore-Penrose-inverse of a matrix $A \in \mathbb{R}^{n \times m}$. We introduce the definition of a generalized Inverse of a matrix first.

The Moore-Penrose-inverse A^+ of a real matrix A is the unique matrix that satisfies the following conditions, see for example [46]:

(a) $AA^+A = A$.

(b) $A^+AA^+ = A^+$.

(c) $(AA^+)^T = AA^+$.

(d) $(A^+A)^T = A^+A$.

Moreover, the Moore-Penrose-inverse exists for any matrix $A \in \mathbb{R}^{m \times n}$. If A is square and non-singular, the inverse A^{-1} fulfills the properties (a)-(d), such that $A^+ = A^{-1}$. So in this case, A^+ is unique. This also holds in general, see for example [72, Theorem 3.2].

In addition, we have that A^+ can be a left or right inverse of A, as the following lemma shows. This result can be found for example in [8, Chapter 1.3, Lemma 2].

Lemma A.3. *Let $A \in \mathbb{R}^{n \times m}$. Then:*

(a) If $n \leq m$ and A is of full row-rank, we have $AA^+ = I_n$.

(b) If $n \geq m$ and A is of full column rank, we have $A^+A = I_m$.

The Moore-Penrose-inverse can now be computed via the singular value decompostion of A, as shown for example in [55]. More precisely, we have the following result.

Lemma A.4. *Consider $A \in \mathbb{R}^{n \times m}$ and its singular value decomposition $A = U\Sigma V^T$. Then*

$$A^+ = V\Sigma^+U^T = V \begin{pmatrix} \Sigma_{r\times r}^{-1} & 0_{r\times(m-r)} \\ 0_{(n-r)\times r} & 0_{(n-r)\times(m-r)} \end{pmatrix}^T U^T = V \begin{pmatrix} \Sigma_{r\times r}^{-1} & 0_{(n-r)\times r} \\ 0_{r\times(m-r)} & 0_{(n-r)\times(m-r)} \end{pmatrix} U^T.$$

In Matlab this approach is already implemented in the command `pinv` and is used for the computations in Algorithms 2 and 5. We can use the Moore-Penrose-inverse of a matrix A to obtain solutions x to the problem $\min_x \|Ax - b\|$, as shown in Lemma 6.7. This is the main advantage of using this generalized inverse in this setting. With the properties presented in this Appendix, it is possible to derive the results in this thesis, which are based on the singular value decomposition or the Moore-Penrose-inverse.

List of Algorithms

1 Alternating projections between $\mathcal{P}$ and $\mathcal{O}_r$ 81
2 Modified algorithm for completely positive matrix factorizations 84
3 Alternating projections between $\mathcal{P}_{\varepsilon,1}$ and $\mathcal{O}_r$ 86
4 Alternating projections between $\mathcal{P}_{\varepsilon,2}$ and $\mathcal{O}_r$ 86
5 Modified algorithm for the interior of the completely positive cone 88

6 The algorithm by Ding et al. [39] 100

7 Symmetric nonnegative matrix factorization based on Algorithm 1 112
8 Symmetric nonnegative matrix factorization based on Algorithm 2 113
9 Modified algorithm for nonnegative matrix factorization 129

List of Figures

5.1 Alternating projections for two linear subspaces in $\mathbb{R}^2$ 46
5.2 Alternating projections for two convex sets in $\mathbb{R}^2$ 52
5.3 Alternating projections for two nonintersecting convex sets in $\mathbb{R}^2$ 52
5.4 Cyclic projections approach for four noninterseting convex sets in $\mathbb{R}^2$ 54
5.5 The polar cone and ε-polar cone are different in general 57
5.6 ε-polar cones for different values of ε 57
5.7 Transversality for manifolds. An example in $\mathbb{R}^2$ 62
5.8 Alternating projections for two one-dimensional manifolds in $\mathbb{R}^2$ 69
5.9 Alternating projections between closed sets 76

7.1 Success rate of Algorithm 2 for Example 7.1 with different values of n. 90
7.2 Success rate of Algorithm 2 for the matrix A_6 from Example 7.1 using different values of $r \geq \operatorname{cpr}(A_6) = 6$. 92
7.3 Success rate of Algorithm 2 for the matrix A_{10} from Example 7.1 using different values of $r \geq \operatorname{cpr}(A_{10}) = 10$. 92
7.4 Success rates of Algorithms 1 and 2 in comparison. 95
7.5 Success rate of Algorithm 2 for column replication vs appending zero columns. . 96
7.6 Success rate of Algorithm 5 for Example 2.22. 108

8.1 Success rates of Algorithms 7 and 8 for Example 2.43 with $k = 2$ for the same starting points. 115
8.2 Success rates of Algorithms 7 and 8 for Example 2.22 with $k = 2$ for the same starting points. 116
8.3 Success rate of Algorithm 8 for random matrices A of order 10×10 for several values of k and 100 randomly chosen starting points for each k. 117
8.4 Success rate of Algorithm 8 for random matrices of order $n \times n$ with $k = \lfloor 0.7n \rfloor$. 117
8.5 Success rate of Algorithm 9 for a given matrix of order 5×8 and different values of k. 132
8.6 Success rate of Algorithm 9 for matrices of variable order $n \times m$ and k fixed to $\lfloor 0.7 \cdot \min\{n, m\} \rfloor$. 133

List of Tables

7.1 Performance of Algorithm 2 on the boundary and in the interior of CP_n 97
7.2 Performance of Algorithm 2 for randomly generated matrices of higher order . . 99
7.3 Performance of Algorithm 6 applied to Example 2.22. 101
7.4 Performance of Algorithm 2 applied to Example 2.22. 101
7.5 Performance of Algorithm 6 applied to Example 7.3 102
7.6 Performance of Algorithm 2 applied to Example 7.3. 102
7.7 Performance of Algorithm 1 applied to Example 7.3. 102
7.8 Quality of the approximation of the approach by Jarre and Schmallowsky applied to Example 2.27. 104
7.9 Performance of the approach by Jarre and Schmallowsky for randomly generated matrices of higher order 105

Nomenclature

Scalars

$\alpha(V_1, V_2)$	Friedrichs angle between two subspaces V_1, V_2.
$c(V_1, \ldots, V_k)$	Cosine of the Friedrichs number between subspaces $V_1, \ldots, V_k$.
$c_i(A_1, A_2; \varepsilon)$	Cosine of the i-th ε-angle between convex sets A_1, A_2.
$c_i(A_1, \ldots, A_k; \varepsilon)$	Cosine of the i-th ε-angle between the convex sets $A_1, A_2, \ldots, A_k$.
$c(M, N; x)$	Cosine of the angle between two manifolds M, N at $x \in M \cap N$.
$\operatorname{codim}(A)$	Codimension of a vector space A.
$\operatorname{cpr}(A)$	cp-rank of a matrix A.
$\operatorname{cpr}^+(A)$	cp$^+$-rank of a matrix A.
$d(A, B)$	Minimal distance between two convex sets A and B.
$\dim(A)$	Dimension of a vector space A.
$\operatorname{rank}(A)$	Rank of a matrix A.
$\rho(A)$	Spectral radius of a matrix A.
$\operatorname{trace}(A)$	Trace of a matrix A.

Vectors

e	All-ones vector.
e_i	i-th unit vector.
$\nabla f(x)$	Gradient or Jacoby matrix of a function f at point x.
$\mathbb{R}^n_+$	Set of entrywise nonnegative vectors of order n.
$\mathbb{R}^n_{++}$	Set of entrywise strictly positive vectors of order n.
$\operatorname{vec}(A)$	Vectorization of a matrix A (stacks the columns of A on top of one another).

Matrices

$\mathcal{COP}_n$	Cone of copositive matrices of order n.
$\mathcal{CP}_n$	Cone of completely positive matrices of order n.
$\mathcal{DNN}_n$	Set of doubly nonnegative matrices of order n.
$\mathcal{D}_n$	Set of diagonal matrices of order n.
$\mathcal{N}_n$	Cone of entrywise nonnegative matrices of order n.
$\mathcal{O}_n$	Set of orthogonal matrices of order n.
$\mathbb{R}_+^{n\times m}$	Set of entrywise nonnegative matrices of order $n \times m$.
$\mathcal{S}_n$	Set of symmetric matrices of order n.
$\mathcal{S}_n^+$	Cone of positive semidefinite matrices of order n.
$\langle A, B\rangle$	Inner product of two matrices A, B.
$A \geq 0$	The matrix A is entrywise nonnegative.
$A > 0$	The matrix A is entrywise strictly positive.
$A \otimes B$	The Kronecker Product of two matrices A and B.
A^+	The Moore-Penrose-inverse of a matrix A.
$A \succcurlyeq 0$	The matrix A is positive semidefinite.
$A \succ 0$	The matrix A is positive definite.
A_{*j}	The j-th column of a matrix A.
A_{i*}	The i-th row of a matrix A.
A_{ij}	The entry of matrix A at position i, j.
$\mathrm{Diag}(x)$	Diagonal matrix with diagonal entries equal to the entries of the vector x.
$E_{n\times m}$	All-ones matrix of order $n \times m$.
E_ε	Matrix with first column entrywise equal to ε and all the other entries are 0.
I_n	Identity matrix of order $n \times n$.
$\max(A, B)$	Entrywise maximum of two matrices A, B.
$0_{n\times m}$	Zero matrix of order $n \times m$.

Sets

$B_r(x)$	The closed ball in $\mathbb{R}^n$ of radius r with center equal to the vector $x \in \mathbb{R}^n$.
$\mathrm{bd}(C)$	Boundary of a set C.
$C^{\circ,\varepsilon}$	ε-polar cone of a closed convex set C.
C°	Polar cone of a closed convex set C.
C^k	Differentiability class of degree k with $k \in [0, \infty]$.
$C^{\perp}$	Orthogonal complement of a set C.
$\mathrm{cl}(C)$	Closure of a set C.
$\mathrm{cone}(C)$	Conic hull of a set C.
$\mathrm{conv}(C)$	Convex hull of a set C.
$\mathbb{E}$	Euclidean space.
$\mathrm{ext}(C)$	Extreme rays of a set C.
$\mathrm{Fix}(f)$	Set of fixed points of a function f.
$\mathrm{int}(C)$	Interior of a set C.
K^*	Dual cone of a cone K.
$\ker(f)$	Kernel of a function f.
$N_M(x)$	Normal space to a manifold M in $x \in M$.
$N_Q(x)$	Normal cone to a closed set Q in $x \in Q$.
$N_Q^p(x)$	Proximal normal cone to a closed set Q in $x \in Q$.
SOC	Second order cone.
$\mathrm{span}(x_1, \ldots, x_n)$	Linear hull of the vectors $x_1, \ldots, x_n$.
$\mathrm{supp}(x)$	Support of a vector x.
$T_M(x)$	Tangent space to a manifold M in $x \in M$.

Problems

$SOCP$	Second order cone problem.
$STQP$	Standard quadratic problem.

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, Princeton, NJ, 2008.

[2] F. Alizadeh and D. Goldfarb. Second-order cone programming. *Mathematical Programming*, 95(1, Ser. B):3–51, 2003.

[3] K. Anstreicher, S. Burer, and P. Dickinson. An algorithm for computing the cp-factorization of a completely positive matrix. *Working paper*, 2015.

[4] C. Badea, S. Grivaux, and V. Müller. A generalization of the Friedrichs angle and the method of alternating projections. *Comptes Rendus Mathématique*, 348(1-2):53–56, 2010.

[5] C. Badea, S. Grivaux, and V. Müller. The rate of convergence in the method of alternating projections. *St. Petersburg Mathematical Journal*, 23(3):413–434, 2012.

[6] H. H. Bauschke and J. M. Borwein. On the convergence of von Neumann's alternating projection algorithm for two sets. *Set-Valued Analysis*, 1(2):185–212, 1993.

[7] H. H. Bauschke and J. M. Borwein. Dykstra's alternating projection algorithm for two sets. *Journal of Approximation Theory*, 79(3):418–443, 1994.

[8] A. Ben-Israel and T. N. E. Greville. *Generalized inverses: Theory and applications*, volume 15 of *CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC*. Springer-Verlag, New York, second edition, 2003.

[9] A. Berman. *Cones, matrices and mathematical programming*. Springer-Verlag, Berlin-New York, 1973. Lecture Notes in Economics and Mathematical Systems, Vol. 79.

[10] A. Berman, M. Dür, and N. Shaked-Monderer. Open problems in the theory of completely positive and copositive matrices. *Electronic Journal of Linear Algebra*, 29:46–58, 2015.

[11] A. Berman and N. Shaked-Monderer. *Completely positive matrices*. World Scientific Publishing Co., Inc., River Edge, NJ, 2003.

[12] D. S. Bernstein. *Matrix mathematics: Theory, facts, and formulas*. Princeton University Press, Princeton, NJ, second edition, 2009.

[13] I. M. Bomze. Copositive optimization—recent developments and applications. *European Journal of Operational Research*, 216(3):509–520, 2012.

[14] I. M. Bomze. Building a completely positive factorization. *Central European Journal of Operations Research*, 26(2):287–305, 2018.

[15] I. M. Bomze, P. J. C. Dickinson, and G. Still. The structure of completely positive matrices according to their CP-rank and CP-plus-rank. *Linear Algebra and its Applications*, 482:191–206, 2015.

[16] I. M. Bomze and W. Schachinger. Multi-standard quadratic optimization: interior point methods and cone programming reformulation. *Computational Optimization and Applications.*, 45(2):237–256, 2010.

[17] I. M. Bomze, W. Schachinger, and R. Ullrich. From seven to eleven: completely positive matrices with high cp-rank. *Linear Algebra and its Applications*, 459:208–221, 2014.

[18] I. M. Bomze, W. Schachinger, and R. Ullrich. New lower bounds and asymptotics for the cp-rank. *SIAM Journal on Matrix Analysis and Applications*, 36(1):20–37, 2015.

[19] R. Borhani, J. Watt, and A. Katsaggelos. Fast and effective algorithms for symmetric nonnegative matrix factorization. *arXiv preprint arXiv:1609.05342*, 2016.

[20] S. Bundfuss. *Copositive matrices, copositive programming, and applications*. PhD thesis, TU Darmstadt, 2009.

[21] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120(2, Ser. A):479–495, 2009.

[22] E. Canhasi and I. Kononenko. Automatic extractive multi-document summarization based on archetypal analysis. In G. R. Naik, editor, *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications*, pages 75–88. Springer Berlin Heidelberg, 2016.

[23] G. Casalino, N. Del Buono, and C. Mencar. Nonnegative matrix factorizations for intelligent data analysis. In G. R. Naik, editor, *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications*, pages 49–74. Springer Berlin Heidelberg, 2016.

[24] A. Cegielski. *Iterative methods for fixed point problems in Hilbert spaces*, volume 2057 of *Lecture Notes in Mathematics*. Springer, Heidelberg, 2012.

[25] W. Cheney and A. A. Goldstein. Proximity maps for convex sets. *Proceedings of the American Mathematical Society*, 10(3):448–450, 1959.

[26] P. L. Combettes and H. J. Trussell. Method of successive projections for finding a common point of sets in metric spaces. *Journal of Optimization Theory and Applications*, 67(3):487–507, 1990.

[27] D. Cooley and E. Thibaud. Decomposition and dependence for high-dimensional extremes. *arXiv preprint arXiv:1612.07190*, 2016.

[28] E. de Klerk. *Aspects of semidefinite programming*, volume 65 of *Applied Optimization*. Kluwer Academic Publishers, Dordrecht, 2002. Interior point algorithms and selected applications.

[29] E. de Klerk and D. V. Pasechnik. Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(4):875–892, 2002.

[30] J. Demmel, J. Nie, and V. Powers. Representations of positive polynomials on noncompact semialgebraic sets via KKT ideals. *Journal of Pure and Applied Algebra*, 209(1):189–200, 2007.

[31] F. Deutsch. The method of alternating orthogonal projections. In S. P. Singh, editor, *Approximation theory, spline functions and applications (Maratea, 1991)*, volume 356 of *NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci.*, pages 105–121. Kluwer Acad. Publ., Dordrecht, 1992.

[32] F. Deutsch. *Best approximation in inner product spaces*, volume 7 of *CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC*. Springer-Verlag, New York, 2001.

[33] F. Deutsch and H. Hundal. The rate of convergence for the cyclic projections algorithm. I. Angles between convex sets. *Journal of Approximation Theory*, 142(1):36–55, 2006.

[34] P. H. Diananda. On non-negative forms in real variables some or all of which are non-negative. *Mathematical Proceedings of the Cambridge Philosophical Society*, 58:17–25, 1962.

[35] P. J. C. Dickinson. An improved characterisation of the interior of the completely positive cone. *Electronic Journal of Linear Algebra*, 20:723–729, 2010.

[36] P. J. C. Dickinson. *The copositive cone, the completely positive cone and their generalisations*. PhD thesis, University of Groningen, 2013.

[37] P. J. C. Dickinson and M. Dür. Linear-time complete positivity detection and decomposition of sparse matrices. *SIAM Journal on Matrix Analysis and Applications*, 33(3):701–720, 2012.

[38] P. J. C. Dickinson and L. Gijben. On the computational complexity of membership problems for the completely positive cone and its dual. *Computational Optimization and Applications*, 57(2):403–415, 2014.

[39] C. Ding, X. He, and H. D. Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 606–610. SIAM, 2005.

[40] J. H. Drew, C. R. Johnson, and R. Loewy. Completely positive matrices associated with M-matrices. *Linear and Multilinear Algebra*, 37(4):303–310, 1994.

[41] D. Drusvyatskiy. *Slope and geometry in variational mathematics*. PhD thesis, Cornell University, 2013.

[42] D. Drusvyatskiy, A. D. Ioffe, and A. S. Lewis. Transversality and alternating projections for nonconvex sets. *Foundations of Computational Mathematics*, 15(6):1637–1651, 2015.

[43] M. Dür. Copositive programming – a survey. In M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels, editors, *Recent Advances in Optimization and its Applications in Engineering*, pages 3–20. Springer Berlin Heidelberg, 2010.

[44] M. Dür and G. Still. Interior points of the completely positive cone. *Electronic Journal of Linear Algebra*, 17:48–53, 2008.

[45] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[46] J. A. Fill and D. E. Fishkind. The Moore-Penrose generalized inverse for sums of matrices. *SIAM Journal on Matrix Analysis and Applications*, 21(2):629–635, 1999.

[47] N. Gillis. Introduction to nonnegative matrix factorization. *SIAG/OPT Views and News*, 25(1), 2017.

[48] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[49] G. Gregorcic. The singular value decomposition and the pseudoinverse. Technical report, University College Cork, Ireland, 2001. https://www.cs.bgu.ac.il/~na131/wiki.files/SVD_application_paper.pdf.

[50] P. Groetzner and M. Dür. A factorization method for completely positive matrices. Preprint, http://www.optimization-online.org/DB_HTML/2018/03/6511.html, 2018.

[51] L. Gubin, B. Polyak, and E. Raik. The method of projections for finding the common point of convex sets. *USSR Computational Mathematics and Mathematical Physics*, 7(6):1–24, 1967.

[52] V. Guillemin and A. Pollack. *Differential topology*. AMS Chelsea Publishing, Providence, RI, 2010. Reprint of the 1974 original.

[53] M. Hall, Jr. and M. Newman. Copositive and completely positive quadratic forms. *Mathematical Proceedings of the Cambridge Philosophical Society*, 59:329–339, 1963.

[54] I. Halperin. The product of projection operators. *Acta Scientiarum Mathematicarum (Szeged)*, 23(1-2):96–99, 1962.

[55] R. E. Hartwig. Singular value decomposition and the Moore-Penrose inverse of bordered matrices. *SIAM Journal on Applied Mathematics*, 31(1):31–41, 1976.

[56] Z. He, S. Xie, R. Zdunek, G. Zhou, and A. Cichocki. Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering. *IEEE Transactions on Neural Networks*, 22(12):2117–2131, 2011.

[57] C. Helmberg. *Semidefinite programming for combinatorial optimization*. Habilitation, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2000.

[58] N. J. Higham. Matrix nearness problems and applications. In M. J. C. Gover and S. Barnett, editors, *Applications of matrix theory (Bradford, 1988)*, volume 22 of *Institute of Mathematics and its Applications Conference Series*, pages 1–27. Oxford Univ. Press, New York, 1989.

[59] N.-D. Ho. *Nonnegative matrix factorization algorithms and applications*. PhD thesis, Université catholique de Louvain, 2008.

[60] M. James. The generalised inverse. *The Mathematical Gazette*, 62(420):109–114, 1978.

[61] F. Jarre and F. Rendl. An augmented primal-dual method for linear conic programs. *SIAM Journal on Optimization*, 19(2):808–823, 2008.

[62] F. Jarre and K. Schmallowsky. On the computation of C^* certificates. *Journal of Global Optimization*, 45(2):281–296, 2009.

[63] R. M. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum, New York, 1972.

[64] S. Kayalar and H. L. Weinert. Error bounds for the method of alternating projections. *Mathematics of Control, Signals, and Systems*, 1(1):43–59, 1988.

[65] V. L. Klee, Jr. Convex bodies and periodic homeomorphisms in Hilbert space. *Transactions of the American Mathematical Society*, 74:10–43, 1953.

[66] D. Kuang, C. Ding, and H. Park. Symmetric nonnegative matrix factorization for graph clustering. In *Proceedings of the 2012 SIAM international conference on data mining*, pages 106–117. 2012.

[67] T. Laudadio, A. C. Sava, Y. Li, N. Sauwen, D. Sima, and S. Van Huffel. NMF in MR spectroscopy. In G. R. Naik, editor, *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications*, pages 161–177. Springer Berlin Heidelberg, 2016.

[68] C. Lazar and A. Doncescu. Non negative matrix factorization clustering capabilities; application on multivariate image segmentation. In L. Barolli, F. Xhafa, and H. Hsu, editors, *International Conference on Complex, Intelligent and Software Intensive Systems, 2009. CISIS'09.*, pages 924–929. IEEE, 2009.

[69] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001.

[70] A. S. Lewis and J. Malick. Alternating projections on manifolds. *Mathematics of Operations Research*, 33(1):216–234, 2008.

[71] Y. Ma, X. Hu, T. He, and X. Jiang. A robust symmetric nonnegative matrix factorization framework for clustering multiple heterogeneous microbiome data. Preprint, `https://www.preprints.org/manuscript/201704.0105/v1`, 2017.

[72] R. MacAusland. The Moore-Penrose inverse and least squares. *Math 420: Advanced Topics in Linear Algebra*, pages 1–10, 2014.

[73] J. E. Maxfield and H. Minc. On the matrix equation $X'X = A$. *Proceedings of the Edinburgh Mathematical Society (2)*, 13:125–129, 1962/1963.

[74] L. Mirsky. Symmetric gauge functions and unitarily invariant norms. *The Quarterly Journal of Mathematics. Oxford. Second Series*, 11:50–59, 1960.

[75] S. K. Mohamad and Z. Tasir. Educational data mining: A review. *Procedia-Social and Behavioral Sciences*, 97:320–324, 2013.

[76] T. S. Motzkin and E. G. Straus. Maxima for graphs and a new proof of a theorem of Turán. *Canadian Journal of Mathematics*, 17:533–540, 1965.

[77] G. R. Naik. *Non-negative Matrix Factorization Techniques: Advances in theory and applications*. Springer, 2016. Signals and communcation Technology.

[78] National Bureau of Standards. Report 1818. *Quarterly Report, April through June 1952*.

[79] J. Nie. The A-truncated K-moment problem. *Foundations of Computational Mathematics*, 14(6):1243–1276, 2014.

[80] M. Planitz. Inconsistent systems of linear equations. *The Mathematical Gazette*, 63(425):181–185, 1979.

[81] R. J. Plemmons. M-matrix characterizations. I. Nonsingular M-matrices. *Linear Algebra and Applications*, 18(2):175–188, 1977.

[82] J. C. Preisig. Copositivity and the minimization of quadratic functions with nonnegativity and quadratic equality constraints. *SIAM Journal on Control and Optimization*, 34(4):1135–1150, 1996.

[83] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

[84] J. Saunderson, P. A. Parrilo, and A. S. Willsky. Semidefinite descriptions of the convex hull of rotation matrices. *SIAM Journal on Optimization*, 25(3):1314–1343, 2015.

[85] B. Shader, N. Shaked-Monderer, and D. B. Szyld. Nearly positive matrices. *Linear Algebra and its Applications*, 449:520–544, 2014.

[86] N. Shaked-Monderer, A. Berman, I. M. Bomze, F. Jarre, and W. Schachinger. New results on the cp-rank and related properties of co(mpletely)positive matrices. *Linear and Multilinear Algebra*, 63(2):384–396, 2015.

[87] K. T. Smith, D. C. Solmon, and S. L. Wagner. Practical and mathematical aspects of the problem of reconstructing objects from radiographs. *Bulletin of the American Mathematical Society*, 83(6):1227–1270, 1977.

[88] W. So and C. Xu. A simple sufficient condition for complete positivity. *Operators and Matrices*, 9(1):233–239, 2015.

[89] J. Sponsel and M. Dür. Factorization and cutting planes for completely positive matrices by copositive projection. *Mathematical Programming*, 143(1-2, Ser. A):211–229, 2014.

[90] G. W. Stewart. *Introduction to matrix computations*. Academic Press [A subsidiary of Harcourt Brace Jovanovich, Publishers], New York-London, 1973. Computer Science and Applied Mathematics.

[91] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11/12(1-4):625–653, 1999. Interior point methods.

[92] K. C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3—a MATLAB software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11/12(1-4):545–581, 1999.

[93] K.-C. Toh, M. J. Todd, and R. H. Tütüncü. On the implementation and usage of SDPT3—a Matlab software package for semidefinite-quadratic-linear programming, version 4.0. In M. F. Anjos and J. B. Lasserre, editors, *Handbook on Semidefinite, Conic and Polynomial Optimization*, volume 166 of *International Series in Operations Research and Management Science*, pages 715–754. Springer US, Boston, MA, 2012.

[94] J. Tura, A. Aloy, R. Quesada, M. Lewenstein, and A. Sanpera. Separability of diagonal symmetric states: a quadratic conic optimization problem. *Quantum*, 2:45, 2018.

[95] J. von Neumann. *Functional Operators. II. The Geometry of Orthogonal Spaces*. Annals of Mathematics Studies, no. 22. Princeton University Press, Princeton, N. J., 1950.

[96] C. Xu. Completely positive matrices. *Linear Algebra and its Applications*, 379:319–327, 2004.

[97] Z.-Y. Zhang. Nonnegative matrix factorization: Models, algorithms and applications. In D. E. Holmes and L. C. Jain, editors, *Data Mining: Foundations and Intelligent Paradigms: Volume 2: Statistical, Bayesian, Time Series and other Theoretical Aspects*, pages 99–134. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

Index

$\mathcal{CP}_n$
- boundary of $\mathcal{CP}_n$, 17
- definition completely positive, 3
- extreme rays of $\mathcal{CP}_n$, 3
- interior of $\mathcal{CP}_n$, 10

ε-angle
- between two convex sets, 58
- for more than two convex sets, 59

angle
- angle between manifolds, 63
- Friedrichs angle, 48
- Friedrichs number, 49

best approximation, 60

clique, 20
- clique number, 19

codimension, 60
complement graph, 20
cone
- ε-polar cone, 56
- dual cone, 4
- normal cone, 75
- pointed cone, 3
- polar cone, 56
- proximal normal cone, 75
- second order cone, 19
  - second order cone problem, 19

conic program, 19
convex hull
- convex hull of orthogonal matrices, 26
- convex hull of rotation matrices, 29

cp$^+$-rank, 12
cp-rank, 12

DJL conjecture, 17

generalized inverse matrix, 138

kernel of a function, 61

manifolds, 60
matrix
- adjacency matrix, 19
- comparison matrix, 8
- copositive matrix, 4
- diagonal matrix, 6
- doubly nonnegative matrix, 5
- entrywise nonnegative matrix, 4
- M-matrix, 8
- nearly positive matrix, 33
- orthogonal matrix, 25
  - reflection matrix, 25
  - rotation matrix, 25
- permutation matrix, 6
- positive semidefinite matrix, 4

minimal distance between two sets, 51

nonnegative matrix factorization, 118
- symmetric nonnegative matrix factorization, 109

normal space, 61
normals, 75
- proximal normals, 75

polar decomposition, 80
projection, 45

semialgebraic set, 76
singular values, 137
- singular value decomposition, 137

stability number, 19
stable set, 19
standard quadratic problem, 20
- multiple standard quadratic problem, 21

tangent space, 61
transversal intersection
- intrinsic transversal intersection for closed sets, 75
- transversal intersection for closed sets, 75
- transversal intersection for manifolds, 61

von neumann sequence, 46
- alternating von neumann sequence, 46