The Hong Kong Polytechnic University

Department of Applied Mathematics

Algorithms and Applications of Semidefinite Space Tensor Conic Convex Program

Yi Xu

A thesis submitted in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

April 2013

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

____________________________(Signed)

_________Xu Yi________(Name of student)

Dedicate to my family.

Abstract

This thesis focuses on studying the algorithms and applications of positive semi-definite space tensors. A positive semi-definite space tensors are a special type semi-definite tensors with dimension 3. Positive semi-definite space tensors have some applications in real life, such as the medical imaging. However, there isn't an algorithm with good performance to solve an optimization problem with the positive semi-definite space tensor constraint, and the structure of positive semi-definite space tensors is not well explored. In this thesis, firstly, we try to analysis the properties of positive semi-definite space tensors; Then, we construct practicable algorithms to solve an optimization problem with the positive semi-definite space tensor constraint; Finally we use positive semi-definite space tensors to solve some medical problems. The main contributions of this thesis are shown as follows.

Firstly, we study the methods to verify the semi-definiteness of space tensors and the properties of H-eigenvalue of tensors. As a basic property of space tensors, the positive semi-definiteness show significant importance in theory. However, there is not a good method to verify the positive semi-definiteness of space tensors. Based upon the nonnegative polynomial theory, we present two methods to verify whether a space tensor positive semi-definite or not. Furthermore, we study the smallest H-eigenvalue of tensors by the relationship between the smallest H-eigenvalue of tensors and their positive semi-definiteness.

Secondly, we consider the positive semi-definite space tensor cone constrained convex program, its structure and algorithms. We study defining functions, defining sequences and polyhedral outer approximations for this positive semi-definite space tensor cone, give an error bound for the polyhedral outer approximation approach, and thus establish convergence of three polyhedral outer approximation algorithms for solving this problem. We then study some other approaches for solving this structured convex program. These include the conic linear programming approach, the nonsmooth convex program approach and the bi-level program approach. Some numerical examples are presented.

Thirdly, we apply positive semi-definite tensors into medical brain imagining. Because of the well-known limitations of diffusion tensor imaging (DTI) in regions of low anisotropy and multiple fiber crossing, high angular resolution diffusion imaging (HARDI) and Q-Ball Imaging (QBI) are used to estimate the probability density function (PDF) of the average spin displacement of water molecules. In particular, QBI is used to obtain the diffusion orientation distribution function (ODF) of these multiple fiber crossing. The ODF, as a probability distribution function, should be nonnegative. We propose a novel technique to guarantee nonnegative ODF by minimizing a positive semi-definite space tensor convex optimization problem. Based upon convex analysis and optimization theory, we derive its optimality conditions. And then we propose a gradient descent algorithm for solving this problem. We also present formulas for determining the principal directions (maxima) of the ODF. Numerical examples on synthetic data as well as MRI data are displayed to demonstrate our approach.

Acknowledgements

I am grateful to the several persons who have supported me in various ways during my PhD program and would like to thank their assistance.

First and foremost, I wish to express my deep thanks to my supervisor, Prof. Liqun Qi, for his enlightening guidance, invaluable discussions and insightful ideas throughout the years. His wisdom benefits me in whole PhD program, and let me know how to study, how to work, how to life.

Furthermore, I would like to express my heartfelt appreciation to Prof. Yaxiang Yuan, Dr. Gaohang Yu and Dr. Xinzheng Zhang, for their great help for my study during my PhD programme. Without their help, I can't finish my research in time.

Finally, I would like to express my special thanks to my family and my friends for their love, encouragement and support.

Contents

List of Figures

List of Tables

List of Notations

$\Re$	the set of real numbers
$\mathbb{C}$	the set of complex numbers
$\Re^n$	the set of n-dimensional real vectors
$\Re^{m\times n}$	the set of $m \times n$ real matrices
$S^{n\times n}$	the set of $m \times n$ real symmetric matrices
$ULR^{m\times m}$	set of $m \times m$ upleft triangular real matrixes
$\mathbb{C}^{m\times n}$	set of $m \times n$ complex matrices
$\mathcal{P}_k^n$	the set of the polynomials in $x \in \Re^k$ with degree n
$\Re_n^m$	the set of n dimension m order real tensors
S_n^m	the set of n dimension m order real symmetric tensors
$S_{n,+}^m$	the set of n dimension m order real symmetric semi-definite tensors
$Re(x)$	the real part of $x \in \mathbb{C}$
$Im(x)$	the imaginary part of $x \in \mathbb{C}$
$E(\mathcal{A})$	a map from $S_3^m \mapsto ULR^{(m+1)\times(m+1)}$
$\lvert\Omega\rvert$	the number of element in the set Ω
x^T	transpose of matrix/vector x
$\lVert x\rVert_2$	$: \sqrt{x^T x}$
$< \mathcal{A}, \mathcal{B} >$	$:= \sum_{i_1,\cdots,i_m} \mathcal{A}_{i_1,\cdots,\imath_m}\mathcal{B}_{i_1,\cdots,\imath_m}$

$\Vert\mathcal{A}\Vert_2$	$:= \sqrt{< \mathcal{A}, \mathcal{A} >}$
$M >> N$	M is far more larger than N

Chapter 1

Introduction

1.1 Background

Since Qi defined the eigenvalues of tensors[54] in 2005, the research on tensors has got more and more attention. Now there are several research directions on tensors:

- 1. Eigenvalues of Tensors,
- 2. Applications in Biomedical Engineering,
- 3. Nonnegative Tensors,
- 4. Spectral Hypergraph Theory,
- 5. Computational Polynomial Optimization,
- 6. Tensor Decomposition,
- 7. Space Tensor Conic Program.

A lot of papers and applications are already finished on above research directions in recent years. For nonnegative tensors, Chang, Friedland, Pearson *et al*. studied the properties of several kinds of nonnegative tensors [9, 10, 11, 48]. A method [42] for computing the largest H-eigenvalue of a nonnegative tensor is studied. The linear convergence of this method is proved under some assumptions [79, 80] . Spectral hypergraph theory is another hot topic on tensors. In 2012 and 2013, the international conference on spectral theory of tensors are held in Tianjin China and Fuzhou China. In these conferences, experts discussed the relationship between the spectrums of tensors and hypergraphs. They gave us plenty of good and new results [33, 34, 38]. In [58] and [61], tensors are used to construct a program which is used for medical imagining. This program overcomes some defects in traditionary methods. In this thesis, we focus on studying the space tensor conic program. This problem involves several research direction, such as eigenvalues of tensors, computational polynomial optimization, applications in biomedical engineering, etc. For example, the program in [58, 61] is a space tensor conic program. But now there is not a good method to solve such program with special structure of space tensor conic, even the structure of space tensor conic is not clear now, so it inspires us to do this research from study the positive semi-definite space tensor.

1.1.1 The Positive Semi-definite Tensor

A n dimension m order real tensor $\mathcal{A}$ consistes of $\mathcal{A}_{i_1,\cdots,i_m} \in \Re, i_j \in \{1,\cdots,n\}$. Real matrixes and vectors could be considered as 2 order tensor and 1 order tensor, respectively. We denote the set of the n dimension m order real tensor as $\Re_n^m$. Especially, we call this type of tensors as the space tensors, when $n = 3$. If the entries of $\mathcal{A}$ are invariant under any index permutation, we call $\mathcal{A}$ as a symmetric tensor, which we call as the supersymmetric tensor in [53], and the set of symmetric real tensors is denoted by S_n^m.

The product of a tensor $\mathcal{A} \in \Re_n^m$ and vectors $x_1, x_2, \cdots, x_m \in \Re^n$ is defined as [53, 54]:

$$\mathcal{A}x_1x_2\cdots x_m = \sum_{i_1,\cdots,i_m=1}^{n} \mathcal{A}_{i_1,i_2,\cdots,i_m}(x_1)_{i_1}(x_2)_{i_2}\cdots(x_m)_{i_m}. \tag{1.1}$$

Especially, when $x_i, i = 1, \cdots, m$ are the same as x, the above equation could be written as $\mathcal{A}x^m$ in short.

We call $\mathcal{A} \in S_n^m$ as a positive semi-definite tensor, when $\mathcal{A}$ satisfies:

$$\mathcal{A}x^m \geqslant 0, \forall\ x \in \Re^n. \tag{1.2}$$

If $\mathcal{A}$ satisfies (1.2), we denote it as $\mathcal{A} \geq 0$.

We call $\mathcal{A} \in S_n^m$ as a positive definite tensor, when $\mathcal{A}$ satisfies:

$$\mathcal{A}x^m > 0, \forall\ 0 \neq x \in \Re^n. \tag{1.3}$$

If $\mathcal{A}$ satisfies (1.3), we denote it as $\mathcal{A} > 0$.

We denote the set of the positive semi-definite tensor as $S_{n,+}^m$. Obviously, when m is odd, $\mathcal{A}$ could not be positive semi-definite, so in this thesis, we always suppose that m is even.

The definition of the positive semi-definite tensor is just like the definition in the matrix case. In matrix case, we have several methods to verify whether a matrix positive is semi-definite or not. For example, finding all the eigenvalues of the matrix is a practical method. In [53, 54], Qi defined H-eigenvalue (eigenvalue) and Z-eigenvalue (E-eigenvalue) of $\mathcal{A} \in S_n^m$. And in [53], we know that when $\mathcal{A} \geq 0$, its H-eigenvalues or Z-eigenvalues must be nonnegative, so we could compute all H-eigenvalues or Z-eigenvalues of $\mathcal{A}$ to verify the positive semi-definiteness of $\mathcal{A}$. We could use the elimination method to get them [61]. However, it is very difficult, especially when the dimension or order of $\mathcal{A}$ is large, since we need to solve a large degree polynomial at last. As is well known, when the degree of the polynomial is large, the algorithm of finding the roots of polynomials is not stable. We also could use another algorithm to find H-eigenvalues or Z-eigenvalues of $\mathcal{A}$, such as the power method (TPM). In [42, 82], Ng and Zhou provided an algorithm for finding the largest eigenvalue of a nonnegative tensor $\mathcal{A}$, and in [79, 80, 81], we proved the convergence theorem of this algorithm for some kinds of nonnegative tensors. From these paper, it is easy to find that this algorithm only suits some kinds of nonnegative tensors, and it can only get the largest H-eigenvalue of $\mathcal{A}$ under some assumptions. For the normal tensor, SS-HOPM [36] gave us another approach. Using this method, we may get the smallest H-eigenvalue of $\mathcal{A}$, but it is not guaranteed. In other words, if we find a convergence point by SS-HOPM, and it is corresponding to a negative H-eigenvalue of $\mathcal{A}$, we could verify $\mathcal{A}$ is not positive semi-definite. But if we get the convergence point corresponding to a nonnegative H-eigenvalue, we could not verify whether there is a negative H-eigenvalue of $\mathcal{A}$.

1.1.2 The Positive Semi-definite Tensor in the Magnetic Resonance Imaging

It is well-known that the popular magnetic resonance imaging (MRI) model, the diffusion tensor imaging model (DTI) [4, 5, 6] breaks down in regions of low anisotropy and multiple fiber crossing. In order to overcome the defects of DTI, Tuch et al [73] proposed a novel approach, high angular resolution diffusion imaging (HARDI) in 2002. In 2004, Tuch [72] further introduced Q-ball imaging (QBI) to reconstruct the diffusion orientation distribution function (ODF) of the underlying fiber population of a biological tissue.

The ODF is a function on the unit sphere describing the probability averaged over the voxel that a particle will diffuse into any solid angle. As the water molecules in normal tissues tend to diffuse along fibers when contained in fiber bundles [6], the principal directions (maxima) of the ODF agree with the true synthetic fiber directions. Tuch [72] showed that the ODF can be estimated directly from the raw HARDI signal on a single sphere of q-space by the Funk-Radon transformation. This is the main idea of QBI.

The Funk-Radon transformation involves integral with the Dirac delta function.

In [17], Descoteax *et al.* proposed to use the Funk-Hecke theorem and higher order spherical harmonics to obtain a mathematical simplification of the Funk-Radon transformation. It was shown that the ODF estimation in [17] is up to 15 times faster than the numerical method in [72].

In [16], Descoteax *etal.* showed that spherical harmonics and higher order tensors restricted to the unit sphere were each a base of the same functional space. Ghosh *etal.* [24] presented a polynomial based approach to extract maxima of the orientation distribution function in diffusion MRI. In [8], using the Z-eigenvalue concept, Bloy and Verma proposed to determine the principal directions (maxima) of the ODF by their curvatures. An excellent recent survey of higher order diffusion tensor methods in diffusion MRI is made by Ghosh and Deriche [23].

In the study of the apparent diffusion coefficient (ADC) profile, in 2003, Ozarslan and Mareci [45] proposed to model the ADC profile with higher order diffusion tensors (HODT), that can reflect more complex micro-geometries of biological tissues. An intrinsic property of the diffusivity profile is positive semi-definite [2, 3, 12, 16, 22, 75]. In [58], Qi proposed to approximate the ADC profile by a positive semi-definite diffusion tensor of either second or higher order. In that paper, Qi showed that we may regard a higher order tensor as a vector, and the diffusivity function as the inner product of two vectors. Such a viewpoint not only is convenient, but also provides an additional mathematical tool. For example, in [58], Qi used this vision to derive the subgradients of the smallest Z-eigenvalue function of the diffusivity function, which is a measure of the positive definiteness of the diffusivity function.

The ODF, as a probability distribution function, should be nonnegative, which means the tensor $\mathcal{A}$ in ODF must be positive semi-definite. In the MRI research, the tensor $\mathcal{A}$ are calculated by the least squares method. Let $\bar{\mathcal{A}}$ be the solution of the least squares problem. Because of noise, it is not guaranteed that the tensor, obtained by the least squares method, are positive semi-definite. MRI researchers have made efforts to correct this noise effect. But until now, no ODF models have addressed this well. Recently, Tournier *et al.* [71] proposed a nonnegativity penalty fiber orientation distribution (FOD) model by penalizing FOD values. That model does not completely forbid negative FOD values. Some other approaches had been proposed to preserve positive semi-definiteness of a diffusion tensor of second order or fourth order [2, 3, 16, 22]. None of them can work for arbitrary high order diffusion tensors. In [58], a comprehensive model, called PSDT (positive semi-definite tensor), was proposed to approximate the diffusivity function by a positive semi-definite diffusion tensor of either second or higher order. A nonnegative diffusion orientation distribution function model was further proposed in [61].

It was proved in [58] that $S^m_{3,+}$ is a closed convex cone. The PSDT minimization problem was proposed there:

$$\min\{P(x) \equiv (H(\mathcal{X}) - H(\bar{\mathcal{X}}))^T Q(H(\mathcal{X}) - H(\bar{\mathcal{X}})) : \mathcal{X} \in S^m_{3,+}\}, \tag{1.4}$$

where $\bar{\mathcal{X}} \in S^m_3$, Q is an $n \times n$ positive semi-definite matrix, $H(\mathcal{X}) : S^m_3 \to \Re^{\frac{(m+1)(m+2)}{2}}$ which stretches a symmetric tensor into a vector.

In chapter 4, we will study algorithms for solving the following structured convex program:

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in S^m_{3,+}\}, \tag{1.5}$$

where $f : S^m_3 \to \Re$ and $g : S^m_3 \to \Re^p$ are twice continuously differentiable convex functions. Clearly, the PSDT problem (1.4) is a special case of (1.5).

1.2 Summary of Contributions of the Thesis

The original contributions of this thesis are as follows:

- Two kinds of the methods based on the theory of the nonnegative polynomial are presented to verify whether the tensor is positive semi-definite or not. The structure of the cone of the positive semi-definite space tensor is analyzed, some properties of $S^m_{n,+}$ are shown. By the relationship of positive semi-definite tensors and their smallest H-eigenvalue, the properties of the smallest H-eigenvalue of tensors are discussed. Algorithms for verifying the positive semi-definiteness of tensors are presented, and the numerical results are presented to show the effectiveness of the algorithms.

- The structure and algorithms of the positive semi-definite space tensor cone constrained convex program (PSDT) are considered. We study defining functions, defining sequences and polyhedral outer approximations for this positive semi-definite space tensor cone, give an error bound for the polyhedral outer approximation approach, and thus establish convergence of three polyhedral outer approximation algorithms for solving this problem. Some other approaches for solving this structured convex program is studied too, including the conic linear program approach, the nonsmooth convex program approach and the bi-level program approach. Some numerical examples are presented.

- We proposes a novel technique to guarantee nonnegative ODF by minimizing a convex optimization problem, which involves a convex quadratic objective function constrained by the nonnegativity requirement on the smallest Z-eigenvalue of the diffusivity tensor. Based upon convex analysis and optimization theory, we derive its optimality conditions. And then we propose a gradient descent algorithm for solving this problem. We also present formulas for determining the principal directions (maxima) of the ODF. Numerical examples on synthetic data as well as MRI data are displayed to demonstrate our approach.

1.3 Organization of the Thesis

The thesis is structured as follows.

- Chapter 2 reviews the preliminary knowledge.

- Chapter 3 focuses on how to verify the positive semi-definiteness of a symmetric space tensor. Firstly, we give two methods based on the theory of the nonnegative polynomial to deal with this problem. Then we use the relationship between the positive semi-definiteness of tensors and their smallest H-eigenvalue (Z-eigenvalue) to study the properties of the smallest H-eigenvalue of tensors. Thirdly, some algorithms are presented to verify the positive semi-definiteness of space tensors. In the last part of this chapter, we give some numerical test results.

- Chapter 4 works on how to compute the space positive semi-definite tensor cone program. Firstly, we study the structure of the positive semi-definite space tensor cone program and its polyhedral outer approximation algorithms. We also study the other methods on solving the positive semi-definite space tensor cone program in section 4.2 in this chapter. Finally, we present the numerical results in the last section of chapter 4.

- In chapter 5, we first show that there is a constant linear transformation relation between the vector versions of the raw HARDI signal and the ODF in the homogeneous polynomial basis. Such a linear transformation connection between the HARDI signal and the ODF not only saves the computational time, but also makes the nonnegative ODF model possible. Based upon this, a nonnegative ODF model is presented. In section 5.2, we present formulas for determining the principal directions (maxima) of the ODF, based on optimization theory, which is more precise. Numerical examples on synthetic data as well as MRI data are displayed in section 5.3 to demonstrate our approach.

- Chapter 6 concludes the whole thesis and plans the future work.

Chapter 2

Preliminaries

In this chapter, we introduce some basic definitions, theorems and notions used in this thesis.

2.1 The Definition and Properties of the Eigenvalue and E-eigenvalue of Tensors

In [53], Qi introduced the definition of the H-eigenvalue (eigenvalue) and the Z-eigenvalue (E-eigenvalue) of a tensor $\mathcal{A} \in \Re_n^m$.

For a vector $x \in \Re^n$, we use x_i to denote its components, and $x^{[m]}$ to denote a vector in $\Re^n$ such that

$$x_i^{[m]} = x_i^m$$

for all i. By definition of the tensor product in [53, 54], $\mathcal{A}x^{m-1}$ with a vector $x \in \Re^n$ denotes a vector in $\Re^n$, whose ith component is

$$\sum_{i_2,\cdots,i_m}^{n} \mathcal{A}_{i,i_2,\cdots,i_m} x_{i_2} \cdots x_{i_m}$$

.

A real number $\lambda \in \Re$ is called an H-eigenvalue of $\mathcal{A}$, iff $\exists x \in \Re^n$ satisfies

$$\mathcal{A}x^{m-1} = \lambda x^{[m-1]},$$

x is called the H-eigenvector corresponding to λ.

A real number $\lambda \in \Re$ is call a Z-eigenvalue of $\mathcal{A}$, iff $\exists x \in \Re^n$ satisfies

$$\begin{cases} \mathcal{A}x^{m-1} = \lambda x \\ x^T x = 1 \end{cases},$$

x is called the Z-eigenvector corresponding to λ.

A number $\lambda \in \mathbb{C}$ is call an eigenvalue of $\mathcal{A}$, iff $\exists x \in \mathbb{C}^n$ satisfies

$$\mathcal{A}x^{m-1} = \lambda x^{[m-1]},$$

x is called the eigenvector corresponding to λ.

A number $\lambda \in \mathbb{C}$ is called an E-eigenvalue of $\mathcal{A}$, iff $\exists x \in \mathbb{C}^n$ satisfies

$$\begin{cases} \mathcal{A}x^{m-1} = \lambda x \\ x^T x = 1 \end{cases},$$

x is called the E-eigenvector corresponding to λ.

The two groups of the definitions of the eigenvalue of a tensor $\mathcal{A}$ are different at the regions which λ and x belong to. It makes some different properties.

We import an important theorem here.

Theorem 2.1. *[53] Assume that m is even. The following conclusions hold for $\mathcal{A} \in S_n^m$:*

(a) $\mathcal{A}$ always has H-eigenvalues. $\mathcal{A}$ is positive definite (positive semi-definite) if and only if all of its H-eigenvalues are positive (nonnegative).

(b) $\mathcal{A}$ always has Z-eigenvalues. A is positive definite (positive semi-definite) if and only if all of its Z-eigenvalues are positive (nonnegative).

2.2 Nonnegative Polynomials

If a polynomial $\mathcal{P}_k^n \ni p(x) \geqslant (>) 0, \forall x \in \Re^k$, we call $p(x)$ is a nonnegative (positive) polynomial. The theory of nonnegative polynomial has been studied for a long time, since Hilbert's 17th problems were proposed. A lot of theories on this topic have been discovered. Here we only present the theorems we need as shown in theorem 2.2. More details could be found in [50, 35, 63, 31, 52, 46, 65, 43, 37, 39, 51].

Theorem 2.2. *[63]*

(a) $p(x) \in \mathcal{P}_1^m, p(x) \geqslant 0,$ *then* $\exists q_1(x), q_2(x) \in \mathcal{P}_1^{\frac{m}{2}}$, $p(x) = q_1^2(x) + q_2^2(x)$;

(b) a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^4, p(x) \geqslant 0,$ *then* $\exists q_i(x) \in \mathcal{P}_3^2, i = 1, 2, 3, 4,$ $p(x) = \sum_{i=1}^4 q_i^2(x)$.

(c) a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^{2k}, p(x) \geqslant 0,$ *then there exist not all null homogeneous polynomial* $p_1(x) \in \mathcal{P}_3^{K_1}$ *and homogeneous polynomial* $p_2(x) \in \mathcal{P}_3^{K_2}$ *in* $x \in \Re^3$, *where* $K_1 = 2\lfloor\frac{(k-1)^2}{2}\rfloor,\ K_2 = 2k + K_1,$ *so that*

$$p_1(x)p(x) = p_2(x),$$

where $p_1(x) = \sum q_{1,i}^2(x), q_{1,i}(x) \in \mathcal{P}_3^{\frac{K_1}{2}}$, $p_1(x) = \sum q_{2,i}^2(x), q_{2,i}(x) \in \mathcal{P}_3^{\frac{K_2}{2}}$

2.3 Cone Programs

In this thesis, we will see two kinds of cone programs, the variable of which is constrained by a cone constraint. We introduce them in brief. The most common cone program is the semi-definite program (sdp), for which the variable is a symmetric positive semi-definite matrix X. The general form of sdp is:

$$\begin{aligned} \min \quad & f(X) \\ s.t. \quad & g(X) \leqslant 0 \\ & X \geq 0, \end{aligned} \tag{2.1}$$

where $f(X) : S^{n\times n} \to \Re, g(X) : S^{n\times n} \to \Re^p$. A lot of optimization problems can be turned into (2.1), such as Linear program (LP), Second Order Cone program

(SOCP). Now we usually use the interior point method to solve (2.1), which is an iteration method with the iteration points in the interior of the feasible set of (2.1). More details and algorithms of sdp could be found in [76, 68, 41, 77, 78, 30]. Some softwares for solving the linear sdp are directly available [20, 69, 70, 74].

We will see another cone program at chapter 4 and chapter 5. The general form of this program is

$$\begin{aligned} \min \quad & f(\mathcal{X}) \\ s.t. \quad & g(\mathcal{X}) \leqslant 0 \\ & \mathcal{X} \geq 0, \end{aligned} \tag{2.2}$$

where $\mathcal{X}$ is a symmetric tensor, $f(\mathcal{X}) : S_n^m \to \Re, g(\mathcal{X}) : S_n^m \to \Re^p$. (2.2) is like (2.1), except that the variable becomes a symmetric tensor from a symmetric matrix. This change makes some differences, we will discuss its applications, algorithms and properties in chapter 4 and chapter 5.

It is noticed that all the experiments in this thesis are implemented in Matlab by using a personal computer with a dual-core 2.4G CPU and 2GB RAM.

Chapter 3

On Semidefiniteness Space Tensor Cone

If we want to use the positive semi-definite tensors in normal life and work, we must know how to verify whether a tensor positive semi-definite or not. In this chapter, we discuss some methods to verify the positive semi-definiteness of a space tensor $\mathcal{A} \in S_3^m$, where m is even.

Firstly, we present some notions and symbols used in this chapter.

Notions and Symbols: $N_n(x)$ in $x \in \Re$ means a column vector $(1, x, \cdots, x^n)^T \in \Re^{n+1}$; $M_n(x)$ in $x \in \Re^3$ is denoted by $(B(i,j,k)x_1^i x_2^j x_3^k)$ in lexicographic order, where $B(i,j,k) = \sqrt{\dfrac{n!}{i!j!k!}}$. For example $M_2(x) = (x_1^2, \sqrt{2}x_1x_2, \sqrt{2}x_1x_3, x_2^2, \sqrt{2}x_2x_3, x_3^2)^T$, so if $x_1^2 + x_2^2 + x_3^2 = 1$, we get $||M_n(x)||_2 = 1$; $< A, B >= \sum_i \sum_j A_{i,j}B_{i,j}$; $\mathcal{C}$ is denoted by a set of matrices $\{\mathcal{C}_i\}$; $\mathcal{C}X$ denotes a vector $(< \mathcal{C}_i, X >)$, $\mathcal{C}^T y = \sum_i \mathcal{C}_i y_i$; the dual of the cone K is defined as $K^* = \{x| < x, y >\geqslant 0, \forall y \in K\}$; $B(x, \varepsilon) = \{x|||x - x^*|| \leqslant \varepsilon\}$; $Rank(A)$ is the rank of a matrix A; I is the identity matrix.

3.1 Two Methods for Verifying the Positive Semi-definiteness of a Tensor

In this section, we present two methods for verifying whether a tensor $\mathcal{A} \in S_3^m$ is positive semi-definite not. Firstly, we introduce an important lemma:

Lemma 3.1. *(1) the next two statements are equivalent.*

a) $p(x) \in \mathcal{P}_1^m, p(x) \geqslant 0, \forall x \in \Re$

b) $\exists\ H \in S^{(\frac{m}{2}+1)\times(\frac{m}{2}+1)}, H \succeq 0,$ so that

$$N_{\frac{m}{2}}(x)^T H N_{\frac{m}{2}}(x) = p(x);$$

(2) the next two statements are equivalent.

a) for a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^4, p(x) \geqslant 0, \forall x \in \Re^3$

b) $\exists\ H \in S^{6\times 6}, H \neq 0, H \succeq 0,$ so that

$$M_2^T(x) H M_2(x) = p(x);$$

(3) the next two statements are equivalent.

a) for a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^{2k}, p(x) \geqslant 0, \forall x \in \Re^3$

b) $\exists$ not all null homogeneous polynomial $p_1(x) \in \mathcal{P}_3^{K_1}$ and homogeneous polynomial $p_2(x) \in \mathcal{P}_3^{K_2}$ in $x \in \Re^3$, where $K_1 = 2\lfloor\frac{(k-1)^2}{2}\rfloor,\ K_2 = 2k + K_1,$ so that

$$p_1(x)p(x) = p_2(x),$$

where

$$p_1(x) = M_{\frac{K_1}{2}}^T(x) H_1 M_{\frac{K_1}{2}}(x), H_1 \in S^{\bar{K_1}\times\bar{K_1}}, H_1 \succeq 0,$$

$$p_2(x) = M_{\frac{K_2}{2}}^T(x) H_2 M_{\frac{K_2}{2}}(x), H_2 \in S^{\bar{K_2}\times\bar{K_2}}, H_2 \succeq 0,$$

where $\bar{K_i} = \frac{(K_i+4)(K_i+2)}{8}, i = 1, 2.$

Proof. From theorem 2.2, we know that

(a) $p(x) \in \mathcal{P}_1^m, p(x) \geqslant 0$, then $\exists q_1(x), q_2(x) \in \mathcal{P}_1^{\frac{m}{2}}$, $p(x) = q_1^2(x) + q_2^2(x)$;

(b) a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^4, p(x) \geqslant 0$, then $\exists q_i(x) \in \mathcal{P}_3^2, i = 1, 2, 3, 4$, $p(x) = \sum_{i=1}^4 q_i^2(x)$.

(c) a not null homogeneous polynomial $p(x) \in \mathcal{P}_3^{2k}, p(x) \geqslant 0$, then $\exists$ not all null homogeneous polynomial $p_1(x) \in \mathcal{P}_3^{K_1}$ and homogeneous polynomial $p_2(x) \in \mathcal{P}_3^{K_2}$ in $x \in \Re^3$, where $K_1 = 2\lfloor \frac{(k-1)^2}{2} \rfloor$, $K_2 = 2k + K_1$, so that

$$p_1(x)p(x) = p_2(x),$$

where $p_1(x) = \sum q_{1,i}^2(x), q_{1,i}(x) \in \mathcal{P}_3^{\frac{K_1}{2}}$, $p_1(x) = \sum q_{2,i}^2(x), q_{2,i}(x) \in \mathcal{P}_3^{\frac{K_2}{2}}$.

(1) Notice when $q_i(x) \in \mathcal{P}_1^{\frac{m}{2}}, i = 1, 2, \exists q_i \in \Re^{\frac{m}{2}+1}, i = 1, 2$. Thus $q_i(x) = q_i^T N_{\frac{m}{2}}(x)$, then $q_1^2(x) + q_2^2(x) = \sum_{i=1}^2 q_i^T N_{\frac{m}{2}}(x) q_i^T N_{\frac{m}{2}}(x) = N_{\frac{m}{2}}^T(x) \sum_{i=1}^2 q_i q_i^T N_{\frac{m}{2}}(x)$. Let $H = \sum_{i=1}^2 q_i q_i^T$, so ⇒ in (1) is true. If $\exists H \succeq 0$, which satisfies $N_{\frac{m}{2}}(x)^T H N_{\frac{m}{2}}(x) = p(x)$, it is easy to see $p(x) \geqslant 0$. ⇐ is true.

(2) Because $\sum_{i=1}^4 q_i^2(x), q_i(x) \in \mathcal{P}_3^2 \Leftrightarrow M_2^T(x) \sum_{i=1}^4 q_i q_i^T M_2(x)$, where $q_i(x) = q_i^T M_2(x)$. Let $H = \sum_{i=1}^4 q_i q_i^T$, so we have ⇒ too. ⇐ is also easy to get.

(3) ⇒ could be proved by the same way in (2). ⇐: Because $p_l(x) = M_{\frac{K_l}{2}}^T(x) H_l M_{\frac{K_l}{2}}(x) = \sum_i^{r_l} q_{l,i}^2(x), l = 1, 2, r_l = Rank(H_l)$, where $q_{l,r}(x) = q_{l,r}^T M_{\frac{K_l}{2}}(x), H_l = \sum_{i=1}^{r_l} q_{l,i} q_{l,i}^T$. We just need to prove that if $\exists p_1(x) = \sum q_{1,i}^2(x), q_{1,i}(x) \in \mathcal{P}_3^{\frac{K_1}{2}}$, $p_1(x) = \sum q_{2,i}^2(x), q_{2,i}(x) \in \mathcal{P}_3^{\frac{K_2}{2}}$, $p_1(x)p(x) = p_2(x)$, $p_1(x), p_2(x)$ not all zero, then $p(x) \geqslant 0, \forall x \in \Re^3$. It is clear that $p_1(x) \geqslant 0, p_2(x) \geqslant 0$, so $p(x)$ must be nonnegative on $\{x | p_1(x) > 0\}$. Let $\Delta = \{x | p_1(x) = 0\}$, if $\exists x^* \in \Delta$, $p(x^*) < 0$, then $\exists \varepsilon > 0$, $p(x) < 0, \forall x \in B(x^*, \varepsilon)$. Now we have $p_1(x) = 0, \forall x \in B(x^*, \varepsilon)$ which means $p_1(x) \equiv 0$, $p_2(x) \equiv 0$, it contradicts that $p_1(x), p_2(x)$ are not all null, so $p(x) \geqslant 0, \forall x \in \Re^3$. □

3.1.1 The First Method

From lemma 3.1(3), for a tensor $\mathcal{A} \in S_3^m$, where $m = 2k(k \geqslant 1)$, $\mathcal{A} \succeq 0 \Leftrightarrow \mathcal{A}x^m \geqslant 0, \forall x \in \Re^3$, which is equivalent to $\exists H_1 \succeq 0, H_2 \succeq 0$, so that $M^T_{\frac{K_1}{2}}(x)H_1M_{\frac{K_1}{2}}(x)\mathcal{A}x^m = M^T_{\frac{K_2}{2}}(x)H_2M_{\frac{K_2}{2}}(x)$, where K_1, K_2 is defined in the lemma 3.1(3). Because there is a vector p satisfying $\mathcal{A}x^m = p^TM_m(x)$, by comparing the coefficients of every term, we could get the corresponding equations:

$$\left\{\begin{array}{ccc} \mathcal{C}_1H_1 & = & p_1, \\ \mathcal{C}_2H_2 & = & p_2, \\ Ap_1 & = & p_2, \\ H_1 \succeq 0, & & H_2 \succeq 0, \end{array}\right. \tag{3.1}$$

where $A \in \Re^{\hat{K_2}\times\hat{K_1}}$ satisfying that $(Ap_1)^TM_{K_2}(x) = \mathcal{A}x^mp_1^TM_{K_1}(x)$, $\hat{K_i} = \frac{(K_i+2)(K_i+1)}{2}$; $\mathcal{C}_1 = \{\mathcal{C}_t^1\}, \mathcal{C}_t^1 \in \Re^{\bar{K_1}\times\bar{K_1}}, t = 1, \cdots, \hat{K_1}$, $\mathcal{C}_2 = \{\mathcal{C}_t^2\}, \mathcal{C}_t^2 \in \Re^{\bar{K_2}\times\bar{K_2}}, t = 1, \cdots, \hat{K_2}$,

$$(\mathcal{C}_t^l)_{i,j} = \left\{\begin{array}{cc} 1 & i+j = t+1 \\ 0 & otherwise \end{array}\right., l = 1, 2,$$

$\bar{K_i}$ are defined in lemma 3.1(3).

Using above symbols, we setup the next program:

$$\begin{array}{rl} val = \max & < I, H_1 > \\ s.t. & \mathcal{C}_1H_1 = p_1, \\ & \mathcal{C}_2H_2 = p_2, \\ & Ap_1 = p_2, \\ & H_1 \succeq 0, \qquad H_2 \succeq 0, \\ & < I, H_1 >\leqslant 1. \end{array} \tag{3.2}$$

If $\mathcal{A} \neq 0, \mathcal{A} \succeq 0$, $\exists$ $H_1 \succeq 0, H_2 \succeq 0$ and not all them are zero matrixes, in other words, $< I, H_1 >> 0$ or $< I, H_2 >> 0$. Because the constraints is homogenous except $< I, H_1 > \leqslant 1$, we have $val = 1$. If $\mathcal{A} \ngeq 0$, because $H_1 = 0, H_2 = 0$ are always feasible for (3.2), then we have $val = 0$. It is easy ot see that $< I, H_1 >$ and $< I, H_2 >$ are good marks for verifying whether $\mathcal{A}$ is positive semi-definite or not. We summarize the above discussion into theorem 3.1:

Theorem 3.1. *(1) Suppose $\mathcal{A} \in S_3^{2k}, \mathcal{A} \neq 0$, then $val = 1 \Leftrightarrow \mathcal{A} \geq 0$; $val = 0 \Leftrightarrow \mathcal{A} \ngeq 0$.*

(2) If $\mathcal{A} \geq 0$, and $\mathcal{A} \nsucc 0, \mathcal{A} \neq 0$, then $\mathcal{A}$ has a 0 H-eigenvalue and Z-eigenvalue, and $H_2 \geq 0$, $H_2 \nsucc 0$. Furthermore, supposing x is the H-eigenvector (Z-eigenvector) of $\mathcal{A}$ corresponding to H-eigenvalue (Z-eigenvalue) 0, then $M_{K_2}(x)$ is the eigenvector of H_2 corresponding to eigenvalue 0.

Proof. From the above discission, (1) is true.

(2) It is easy to see that the first part of (2) is true. If $\mathcal{A} \geq 0$, from lemma 3.1(3) and (1), we get $M_{K_1}^T(x) H_1 M_{K_1}(x) \mathcal{A} x^m = M_{K_2}^T(x) H_2 M_{K_2}(x)$ and $< I, H_1 >> 0$. If $\mathcal{A}$ has 0 H-eigenvalue and Z-eigenvalue, then $\exists x \in \Re^3$, such that $\mathcal{A} x^m = 0 \Rightarrow M_{K_2}^T(x) H_2 M_{K_2}(x) = 0$. If x is H-eigenvector (Z-eigenvector) corresponding to H-eigenvalue (Z-eigenvalue) 0, then $M_{K_2}^T(x) H_2 M_{K_2}(x) = 0$. Since $H_2 \geq 0$, which tell us that $M_{K_2}(x)$ is the eigenvector corresponding to 0 eigenvalue of H_2. □

From theorem 3.1, if we get H_2, we could find all the eigenvectors of H_2 corresponding to 0 which could be expressed by $M_{K_2}(x)$, then we bring them back to $\mathcal{A} x^{m-1}$ to test whether x is the H-eigenvector (Z-eigenvector) of $\mathcal{A}$ corresponding to 0. The disadvantage of the first method is that the scale of the program (3.2) is $O(m^4)$. Thus when the m is lager, the program is not solvable in the numerical view.

3.1.2 The Second Method

Notice that when $n = 3$, for every $\mathcal{A}$, there exists an upleft triangular matrix $E \in ULR^{(m+1)\times(m+1)}$, then $\mathcal{A} x^m = \sum_{i=0}^{m} \sum_{j=0}^{m-i} E_{i+1,j+1} x_1^i x_2^j x_3^{m-i-j}$. We set this map $\mathcal{A}$ to

E as $E(\mathcal{A}) : S_3^m \mapsto \Re^{(m+1)\times(m+1)}$, it is easy to see that $E(\mathcal{A})$ is a bijection.

Lemma 3.2. *When m is even,*

$$\mathcal{A} \in S_{3,+}^m \Leftrightarrow E = E(\mathcal{A}) satisfies \begin{cases} \sum_{i=0}^m \sum_{j=0}^{m-i} E_{i+1,j+1} y_1^i y_2^j \geqslant 0, & \forall y_1, y_2 \in \Re \\ \sum_{i=0}^m E_{i+1,m-i+1} y_1^i \geqslant 0 & \forall y_1 \in \Re \\ E_{m+1,1} \geqslant 0 \end{cases}$$

Proof. Because $\mathcal{A} \in S_{3,+}^m \Leftrightarrow \mathcal{A}x^m \geqslant 0, \forall x \in \Re^3$, we discuss three cases:

1) $x_3 \neq 0, x_2, x_1$ are arbitrary, then $\mathcal{A}x^m = x_3^m \sum_{i=0}^m \sum_{j=0}^{m-i} E_{i+1,j+1} (\frac{x_1}{x_3})^i (\frac{x_2}{x_3})^j$. Because m is even, $y_1 = \frac{x_1}{x_3}$, $y_2 = \frac{x_2}{x_3}$, then $\sum_{i=0}^m \sum_{j=0}^{m-i} E_{i+1,j+1} y_1^i y_2^j \geqslant 0$.

2) $x_3 = 0, x_2 \neq 0, x_1$ is arbitrary, then $\mathcal{A}x^m = \sum_{i=0}^m E_{i+1,m-i+1} x_1^i x_2^{m-i} \geqslant 0$. Because $x_2 \neq 0$, let $y_1 = \frac{x_1}{x_2}$, then $\mathcal{A}x^m = x_2^m \sum_{i=0}^m E_{i+1,m-i+1} y_1^i \geqslant 0$, because m is even, $\sum_{i=0}^m E_{i+1,m-i+1} y_1^i \geqslant 0$.

3) $x_3 = 0, x_2 = 0, x_1$ is arbitrary, then $\mathcal{A}x^m = E_{m+1,1} x_1^m$, so $E_{m+1,1} \geqslant 0$. □

Theorem 3.2. *Suppose $p(x) = p^T N_m(x) \in \mathcal{P}_1^m$, where $p \in \Re^{m+1}$, then the next two statements are equivalent:*

(1) $p(x) \geqslant 0, \forall x \in \Re$

(2) the value of the program

$$\begin{aligned} \min \quad & p^T y \\ s.t. \quad & \mathcal{C}^T y \succeq 0. \end{aligned} \tag{3.3}$$

is nonnegative, where $\mathcal{C} = \{\mathcal{C}_i\}, i = 1, \cdots, m+1$, $(\mathcal{C}_i)_{l,k} = \begin{cases} 1, & l+k = i+1 \\ 0, & else \end{cases}, \mathcal{C}_i \in \Re^{(\frac{m}{2}+1)\times(\frac{m}{2}+1)}$.

Proof. If $p(x) \in \mathcal{P}_1^m, p(x) \geqslant 0, \forall x \in \Re$, from lemma 3.1(1) we know that there exists a matrix $A \succeq 0, A \in \Re^{(1+\frac{m}{2})\times(1+\frac{m}{2})}$ satisfying $N_{\frac{m}{2}}(x)^T A N_{\frac{m}{2}}(x) = p(x)$. Since

$p(x) = p^T N_m(x)$, by comparing the coefficient of x^i, we have $\mathcal{C}A = p, A \succeq 0$ which means

$$\begin{cases} \mathcal{C}X = p \\ X \succeq 0 \end{cases} \tag{3.4}$$

has solution. We will prove that if (3.4) has solution, then

$$\begin{cases} p^T y < 0 \\ \mathcal{C}^T y \succeq 0 \end{cases} \tag{3.5}$$

doesn't have solution. If (3.4) and (3.5) have solutions X^*, y^*, then $< \mathcal{C}X^* - p, y^* >= 0$, so we have $p^T y^* =< \mathcal{C}X^*, y^* >=< \mathcal{C}^T y^*, X^* >$. Since $X^* \succeq 0, \mathcal{C}^T y^* \succeq 0$, we have $< \mathcal{C}^T y^*, X^* >\geqslant 0$, which is contrary to $p^T y^* < 0$. If (3.4) has solution, (3.5) doesn't have solution, which means the value of (3.3) is nonnegative.

If the value of (3.3) is nonnegative, which means $\forall y \in \Re^{m+1}$ satisfying $\mathcal{C}^T y \succeq 0$, $p^T y$ is nonnegative. Let $y = N_m(x), \mathcal{C}^T y = N_{1+\frac{m}{2}}(x) N_{1+\frac{m}{2}}(x)^T \succeq 0$, since $p^T N_m(x) = p(x)$, we have $p(x) \geqslant 0, \forall x \in \Re$. $\Box$

Theorem 3.3 shows that $\mathcal{A} \in S^m_{3,+}$ is equal to that both the values of two programs below are nonnegative.

Theorem 3.3. $\mathcal{A} \in S^m_{3,+} \Leftrightarrow$ *the values of the programs*

$$\begin{aligned} \min \quad & w_1^T E w_2 \\ s.t. \quad & \mathcal{C}^T w_1 \succeq 0, \\ & \mathcal{C}^T w_2 \succeq 0, \end{aligned} \tag{3.6}$$

$$\begin{aligned} \min \quad & e^T w \\ s.t. \quad & \mathcal{C}^T w \succeq 0 \end{aligned} \tag{3.7}$$

are nonnegative, where $\mathcal{C}$ *is defined in theorem 3.2,* $E = E(\mathcal{A}), e = e(\mathcal{A}) = (E_{i+1,m-i+1})_{i=0}^m$.

Proof. By lemma 3.2, we just need to consider three cases:

1) Let $q(y_1, y_2) = \sum_{i=0}^{m}\sum_{j=0}^{m-i} E_{i+1,j+1}y_1^i y_2^j$, $q(y_1, y_2)$ could be considered as a polynomial in y_1 with the parameter y_2, we use $q_{y_2}(y_1)$ to express this. If $q(y_1, y_2) \geqslant 0, \forall y_1, y_2$, it means $\forall y_2, q_{y_2}(y_1) \geqslant 0, \forall y_1$, where $q_{y_2}(y_1) = q_{y_2}^T N_m(y_1) = N_m(y_2)^T E(\mathcal{A})N_m(y_1)$. By theorem 3.2, $\forall y_2$, $q_{y_2}(y_1) \geqslant 0, \forall y_1 \Leftrightarrow \forall y_2$, the value of the program

$$\begin{aligned} \min \quad & q_{y_2}^T w_1 \\ s.t. \quad & \mathcal{C}^T w_1 \succeq 0 \end{aligned} \tag{3.8}$$

is nonnegative. $q_{y_2}^T w_1$ could be considered as a polynomial in y_2 with the parameter w_1, we have $q_{w_1}(y_2) = w_1^T E(\mathcal{A})^T N_m(y_2)$, so the nonnegative value of (3.8) means that $\forall w_1$ satisfying $\mathcal{C}^T w_1 \succeq 0$, $q_{w_1}(y_2) \geqslant 0, \forall y_2$. By theorem 3.2, we get $\forall w_1$ satisfying $\mathcal{C}^T w_1 \succeq 0$, $q_{w_1}(y_2) \geqslant 0, \forall y_2 \Leftrightarrow$ the value of the program

$$\begin{aligned} \min \quad & w_2^T E(\mathcal{A}) w_1 \\ s.t. \quad & \mathcal{C}^T w_2 \succeq 0 \end{aligned} \tag{3.9}$$

is nonnegative. So $q(y_1, y_2) \geqslant 0, \forall y_1, y_2 \Leftrightarrow$ the value of the program

$$\begin{aligned} \min \quad & w_2^T E(\mathcal{A}) w_1 \\ s.t. \quad & \mathcal{C}^T w_1 \succeq 0 \\ & \mathcal{C}^T w_2 \succeq 0 \end{aligned} \tag{3.10}$$

is nonnegative.

2) $\sum_{i=0}^{m} E_{i+1,m-i+1}y_1^i = e^T N_m(y_1) \geqslant 0$, by theorem 3.2, it is equivalent to that the value of the program

$$\begin{aligned} \min \quad & e^T w \\ s.t. \quad & \mathcal{C}^T w \succeq 0 \end{aligned} \tag{3.11}$$

is nonnegative.

3) From lemma 3.2, $E_{m+1,1} \geqslant 0$, but it is implied in the nonnegative value of (3.7). □

Notice that it is easy to check whether the value of the program (3.7) is nonnegative or not, but it is difficult to solve (3.6), because it is a bilinear sdp program.

Let $\bar{E}(\mathcal{A}) = \frac{1}{2}\begin{pmatrix} 0 & E(\mathcal{A}) \\ E(\mathcal{A})^T & 0 \end{pmatrix}$, which is a symmetric matrix, and $\bar{E}(\mathcal{A}) : S_3^m \mapsto S^{2(m+1)\times 2(m+1)}$ is a bijection too. We denote its inverse operation as $\bar{E}^{-1}$. Let

$$\bar{\mathcal{C}_i} = \begin{cases} \begin{pmatrix} \mathcal{C}_i & 0 \\ 0 & 0 \end{pmatrix} & i = 1, \cdots, m+1 \\ \begin{pmatrix} 0 & 0 \\ 0 & \mathcal{C}_{i-m-1} \end{pmatrix} & i = m+2, \cdots, 2m+2 \end{cases},$$

where $\mathcal{C}_i$ is defined in theorem 3.2, so (3.6) could be turned to

$$\begin{aligned} \min \quad & w^T \bar{E}(\mathcal{A}) w \\ s.t. \quad & \bar{\mathcal{C}}^T w \geq 0 \end{aligned} \tag{3.12}$$

Use above symbols, the below corollary is true.

Corollary 3.1. $S_{3,+}^m = \bar{E}^{-1}(\{ww^T | \bar{\mathcal{C}}^T w \geq 0\}^* \bigcap \Omega\}) \bigcap \{\mathcal{A} \in S_3^m | e(\mathcal{A}) = \mathcal{C}X, X \geq 0\}$, *where* $\Omega = \{Y | Y = \begin{pmatrix} 0 & Z \\ Z^T & 0 \end{pmatrix}, Z \in ULR^{(m+1)\times(m+1)}\}$.

Proof. From theorem 3.3, it is easy to see that $S_{3,+}^m$ is the intersection of two parts.

$$S_{3,+}^m = \Lambda_1 \bigcap \Lambda_2,$$

where

$$\Lambda_1 = \{\mathcal{A} \in S_3^m | \bar{E}(\mathcal{A}) \text{ makes the value of (3.12) nonnegative}\},$$

$$\Lambda_2 = \{\mathcal{A} \in S_3^m | e(\mathcal{A}) \text{ makes the value of (3.7) nonnegative}\}.$$

1) $\Lambda_1 = \{\mathcal{A} \in S_3^m | w^T \bar{E}(\mathcal{A}) w \geqslant 0, \forall\ w$ satisfying $\bar{\mathcal{C}}^T w \geq 0\} = \{\mathcal{A} \in S_3^m | < \bar{E}(\mathcal{A}), ww^T >\geqslant 0, \forall\ w$ satisfying $\bar{\mathcal{C}}^T w \geq 0\}$.

Because $\{Y \in S^{m\times m} | < Y, ww^T >\geqslant 0, \forall w$ satisfying $\bar{\mathcal{C}}^T w \geq 0\} = \{ww^T | \bar{\mathcal{C}}^T w \geq 0\}^*$. Since $\bar{E}(\mathcal{A}) \in \Omega = \{Y | Y = \begin{pmatrix} 0 & Z \\ Z^T & 0 \end{pmatrix}, Z \in ULR^{(m+1)\times(m+1)}\}$, then we have

$$\Lambda_1 = \bar{E}^{-1}(\{ww^T | \bar{\mathcal{C}}^T w \geq 0\}^* \bigcap \Omega).$$

2) $\Lambda_2 = \{\mathcal{A} \in S_3^m | e(\mathcal{A})^T y \geqslant 0, \forall\ y$ satisfying $\mathcal{C}^T y \geq 0\} = \{\mathcal{A} \in S_3^m | e(\mathcal{A}) = \mathcal{C}X, X \geq 0\}$. □

3.2 Finding the Smallest H-eigenvalue

Because the positive semi-definiteness of a tensor and its smallest H-eigenvalue are related, if we get the positive semi-definiteness of a tensor, it is natural to ask how we could get its smallest H-eigenvalue. This is the aim of this section.

The program (3.13) in lemma 3.3 is similar as (3.6). It gives us a way to find the global minimal of a polynomial $p(x) \in \mathcal{P}_1^m$.

Lemma 3.3. *Suppose $p(x) = p^T N_m(x) \in \mathcal{P}_1^m$, where $p \in \Re^{m+1}$. Let v_1 and Π_1 be the value and minimal solution set of*

$$\begin{aligned} \min \quad & p^T y \\ s.t. \quad & \mathcal{C}^T y \geq 0, \\ & e_1^T y = 1, \end{aligned} \tag{3.13}$$

where $\mathcal{C}$ is defined in theorem 3.2, $e_1 = (1, 0, 0, \cdots 0)^T$, v_2 and Π_2 are the value and minimal solution set of

$$\min p(x),$$

then

(1) if $v_2 \neq -\infty$, then $\forall x^* \in \Pi_2$, $N_m(x^*) \in \Pi_1$;

(2) if $v_1 \neq -\infty$, then $\exists$ x^* satisfying that $N_m(x^*) \in \Pi_1$ and $x^* \in \Pi_2$;

(3) $v_1 = -\infty \Leftrightarrow v_2 = -\infty$;

so $v_1 = v_2$.

Proof. The KKT equations of (3.13) is

$$\begin{cases} \mathcal{C}X + ue_1 = p, \\ e_1^T y = 1, \\ X \succeq 0, \mathcal{C}^T y \succeq 0, \\ < \mathcal{C}^T y, X >= 0 \end{cases} \tag{3.14}$$

where $e_1 = (1, 0, \cdots, 0)^T \in \Re^{m+1}$.

(1)$\forall x^* \in \Pi_2$, let $y = N_m(x^*)$, then $\mathcal{C}^T y = N_{1+\frac{m}{2}}(x^*) N_{1+\frac{m}{2}}(x^*)^T \succeq 0$. Taking $u = p(x^*)$, we have $q(x) = (p - p(x^*)e_1)^T N_m(x) = p(x) - p(x^*) \geqslant 0, \forall x \in \Re$, then by lemma 3.1(1) $\exists X^* \succeq 0$ satisfying $\mathcal{C}X^* = p - p(x^*)e_1$, and $< \mathcal{C}^T N_m(x^*), X^* >=< \mathcal{C}X^*, N_m(x^*) >=< p - p(x^*)e_1, N_m(x^*) >= 0$ so $N_m(x^*), p(x^*), X^*$ solve (3.14). Because (3.13) is a linear sdp program, then $N_m(x^*)$ must be in the optimal solution set of (3.13).

(3) $\Leftarrow$: If $v_1 = -\infty$, and $v_2 = p(x^*) \neq -\infty$, where x^* is a minimal solution, then $N_m(x^*), p(x^*)$, X^* are the solutions of (3.14), where X^* is defined in (1). Then from the theory of the linear sdp program, it shows that $v_1 \neq -\infty$, which contradicts that $v_1 = -\infty$, so we have $v_2 = -\infty$.

$\Rightarrow$: If $v_2 = -\infty$, for every a small enough number $M < 0$, $\exists \hat{x}$, $p(\hat{x}) = M$. Noticing that $N_m(\hat{x})$ is the feasible point of (3.13), we have $v_1 = -\infty$.

(2) $v_1 \neq -\infty$, then from (3), $v_2 \neq -\infty$, so $\exists$ $x^*, x^* \in \Pi_2$. From (1), $N_m(x^*)$ is a minimal solution of (3.13). □

By lemma 3.3, we could solve a linear sdp program to find the global minimal of a polynomial. In this way, we overcome the disadvantage of algorithm for finding roots of the polynomial.

For the 3 dimension case, we could get lemma 3.4.

Lemma 3.4. *Suppose* $p(x) \in \mathcal{P}_3^m, p(x) = \sum_{i+j+k \leqslant m} \mathscr{D}_{i,j,k} x_1^i x_2^j x_3^k$, *we construct next program*

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w_2 w_3 \\ s.t. \quad & \mathcal{C}^T w_i \geq 0, i = 1, 2, 3 \\ & e_1^T w_i = 1, i = 1, 2, 3, \end{aligned} \tag{3.15}$$

where $\mathcal{C}$ *and* e_1 *are defined in lemma 3.3. Let* v_1 *and* Π_1 *be the minimal value and solution of (3.15),* v_2 *and* Π_2 *be the minimal value and solution of*

$$\min_{x \in \Re^3} p(x),$$

then

(1) If $v_2 \neq -\infty$*, then* $\forall (x_1^*, x_2^*, x_3^*)^T \in \Pi_2$*,* $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*)) \in \Pi_1$*;*

(2) If $v_1 \neq -\infty$*, then* $\exists\ (x_1^*, x_2^*, x_3^*)$ *so that* $(x_1^*, x_2^*, x_3^*) \in \Pi_2$*,*

$$(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*)) \in \Pi_1;$$

(3) $v_1 = -\infty \Leftrightarrow v_2 = -\infty$*;*

so $v_1 = v_2$*.*

Proof. (3) $\Leftarrow$: Because $v_2 = -\infty$, then $\exists\ x_k = (x_{k,1}, x_{k,2}, x_{k,3})^T \in \Re^3, p(x_k) \to -\infty$. It is easy to show that $(N_m(x_{k,1}), N_m(x_{k,2}), N_m(x_{k,3}))$ is the feasible point of (3.15), we have

$$\mathscr{D} N_m(x_{k,1}) N_m(x_{k,2}) N_m(x_{k,3}) = p(x_k) \to -\infty,$$

so the value of (3.15) is $-\infty$.

$\Rightarrow$: Because $v_1 = -\infty$, $\exists\ w_1, w_2$, satisfying $\mathcal{C}^T w_i \geq 0$ and $e_1^T w_i = 1, i = 1, 2$, then the value of

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w_2 w \\ s.t. \quad & \mathcal{C}^T w \geq 0, \\ & e_1^T w = 1 \end{aligned} \tag{3.16}$$

is $-\infty$. From lemma 3.3(3), $\min_{x \in \Re} \mathscr{D} w_1 w_2 N_m(x) = -\infty$. For a small enough number M, $\exists\ x_1$ so that $\mathscr{D} w_1 w_2 N_m(x) = M$, then we consider

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w N_m(x_1) \\ s.t. \quad & \mathcal{C}^T w \geq 0, \\ & e_1^T w = 1. \end{aligned} \tag{3.17}$$

It is easy to see that the value u_2 of (3.17) is not larger than M. If $u_2 = -\infty$, then by lemma (3.4), $\min_{x \in \Re} \mathscr{D} w_1 N_m(x) N_m(x_1) = -\infty$, we could find another small enough number $M' \leqslant M$, $\exists\ x_2$, $\mathscr{D} w_1 N_m(x_2) N_m(x_1) = M'$. If $u_2 \neq -\infty$, then $\exists\ x_2$ so that $\mathscr{D} w_1 N_m(x_2) N_m(x_1) \leqslant M$. By the same way, we could get x_3, so that $\mathscr{D} N_m(x_3) N_m(x_2) N_m(x_1) \leqslant M$, which means $v_2 = -\infty$.

(1) Because $v_2 \neq -\infty$, then $v_1 \neq -\infty$. For every $(x_1^*, x_2^*, x_3^*)^T \in \Pi_2$, it is easy to see that $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*))$ is the feasible point of (3.15), we have $v_1 \leqslant v_2 = \mathscr{D} N_m(x_1^*) N_m(x_2^*) N_m(x_3^*)$. Let (w_1^*, w_2^*, w_3^*) be the minimal solution of (3.15), we consider next program

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w_2^* w_3^*, \\ s.t. \quad & \mathcal{C}^T w_1 \geq 0 \\ & e_1^T w_1 = 1 \end{aligned} \tag{3.18}$$

By lemma 3.3, solving (3.18) is equivalent to finding the global minimal value and solution of $q_1(x) = \mathscr{D} N_m(x_1) w_2^* w_3^*$. Suppose $x_1^{**} \in \arg\min_{x \in \Re} q_1(x)$. Then from lemma 3.3, $N_m(x_1^{**})$ belongs to the set of the global minimal solution of (3.18), the value of (3.18) is v_1.

It is easy to get that the value of

$$\begin{aligned} \min \quad & \mathscr{D} N_m(x_1^{**}) w_2 w_3^*, \\ s.t. \quad & \mathcal{C}^T w_2 \geq 0, \\ & e_1^T w_2 = 1, \end{aligned} \tag{3.19}$$

is v_1 too, then $N_m(x_2^{**})$ is the global minimal solution of (3.19), where

$$x_2^{**} \in \arg\min_{x \in \Re} q_2(x) = \mathscr{D} N_m(x_1^{**}) N_m(x) w_3^*.$$

It is also easy to get that the value of

$$\begin{aligned} \min \quad & \mathscr{D} N_m(x_1^{**}) N_m(x_2^{**}) w_3, \\ s.t. \quad & \mathcal{C}^T w_3 \geq 0, \\ & e_1^T w_3 = 1, \end{aligned} \tag{3.20}$$

is v_1, then $N_m(x_3^{**})$ is the global minimal solution of (3.20), where

$$x_3^{**} \in \arg\min_{x \in \Re} q_3(x) = \mathscr{D} N_m(x_1^{**)} N_m(x_2^{**}) N_m(x).$$

Now we get $v_1 = \mathscr{D} N_m(x_1^{**}) N_m(x_2^{**}) N_m(x_3^{**}) = p(x_1^{**}, x_2^{**}, x_3^{**}) \geqslant v_2$, so $v_1 = v_2$, which means v_2 and $(N_m(x_1^{**}), N_m(x_2^{**}), N_m(x_3^{**}))$ are global minimal value and solution of (3.15). Notice that $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*))$ are also feasible point of (3.15), and $v_2 = \mathscr{D} N_m(x_1^*) N_m(x_2^*) N_m(x_3^*)$, so $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*)) \in \Pi_1$.

(2) If $v_1 \neq -\infty$, then $v_2 \neq -\infty$. From (1), we have $\exists (x_1^*, x_2^*, x_3^*) \in \Pi_2$ and $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*)) \in \Pi_1$. $\square$

Using above lemma, we could construct following theorem which could give out the smallest H-eigenvalue of $\mathcal{A} \in S_3^m$.

Theorem 3.4. *For $\mathcal{A} \in S_3^m$, let* $\mathscr{D}_{i,j,k} = \begin{cases} (E(\mathcal{A}))_{i,j} & k = m+3-i-j \\ 0 & else \end{cases}$ $\mathscr{D} \in \Re^3_{m+1}$.

Suppose the smallest H-eigenvalue of $\mathcal{A}$ and its H-eigenvector are λ_1 and $(x_1^*, x_2^*, x_3^*)^T$, and (3.21) has minimal solution, then λ_1, $(N_m(x_1^*),\ N_m(x_2^*),\ N_m(x_3^*))$ are the minimal value and optimal solution of the program

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w_2 w_3 \\ s.t. \quad & \mathcal{C}^T w_i \succeq 0, i = 1,2,3 \\ & e_1^T w_i = 1, i = 1,2,3 \\ & e_{m+1}^T (w_1 + w_2 + w_3) = 1, \end{aligned} \tag{3.21}$$

where e_1 is defined in lemma 3.3, $e_{m+1} = (0,0,0\cdots,1)^T \in \Re^{m+1}, w_i \in \Re^{m+1}$.

Proof. Notice that the smallest H-eigenvalue λ_1 of $\mathcal{A} \in S_3^m$, which must exist by theorem 2.1, could be defined as the global minimal value of the next program [53]:

$$\begin{aligned} \min \quad & \mathcal{A}x^m \\ s.t. \quad & x_1^m + x_2^m + x_3^m = 1, \end{aligned} \tag{3.22}$$

Firstly, we will prove that $\lambda_1 = \min_x \max_\lambda L(x,\lambda) = \max_\lambda \min_x L(x,\lambda)$, $L(x,\lambda) = \mathcal{A}x^m - \lambda(x_1^m + x_2^m + x_3^m - 1)$. $\lambda_1 = \min_x \max_\lambda L(x,\lambda)$ is easy to get from the optimization theory. We just need to prove $\min_x \max_\lambda L(x,\lambda) = \max_\lambda \min_x L(x,\lambda)$ true.

It is oblivious that $\min_x \max_\lambda L(x,\lambda) \geqslant \max_\lambda \min_x L(x,\lambda)$. Because $\mathcal{A}x^m \geqslant \lambda_1(x_1^m + x_2^m + x_3^m), \forall x \in \Re^3$, we have $L(x,\lambda) \geqslant (\lambda_1 - \lambda)(x_1^m + x_2^m + x_3^m) + \lambda$, $\max_\lambda \min_x L(x,\lambda) \geqslant \max_\lambda \min_x (\lambda_1 - \lambda)(x_1^m + x_2^m + x_3^m) + \lambda = \lambda_1$. Because $\lambda_1 = \min_x \max_\lambda L(x,\lambda) \geqslant \max_\lambda \min_x L(x,\lambda) \geqslant \lambda_1$, we have $\lambda_1 = \min_x \max_\lambda L(x,\lambda) = \max_\lambda \min_x L(x,\lambda)$.

From lemma 3.3, we could solve

$$\begin{aligned} \min \quad & \mathscr{D} w_1 w_2 w_3 + \lambda(e_m^T(w_1 + w_2 + w_3) - 1) \\ s.t. \quad & \mathcal{C}^T w_i \geq 0, i = 1, 2, 3 \\ & e_1^T w_i = 1, i = 1, 2, 3 \end{aligned} \tag{3.23}$$

to get the minimal value of $\min_x L(x, \lambda)$.

Let $\Lambda = \{(w_1, w_2, w_3) | \ \mathcal{C}^T w_i \geq 0, e_1^T w_i = 1, i = 1, 2, 3\}$, and suppose the minimal value of (3.21) is $\hat{\lambda}$, since $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*))$ is the feasible point of (3.21), then we have $\mathscr{D} N_m(x_1^*) N_m(x_2^*) N_m(x_3^*) = \lambda_1 \geqslant \hat{\lambda} = \min_{(w_1, w_2, w_3) \in \Lambda} \max_\lambda \bar{L}(w_i, \lambda) \geqslant \max_\lambda \min_{(w_1, w_2, w_3) \in \Lambda} \bar{L}(w_i, \lambda) = \max_\lambda \min_x L(x, \lambda) = \min_x \max_\lambda L(x, \lambda) = \lambda_1$, where $\bar{L}(w_i, \lambda) = \mathscr{D} w_1 w_2 w_3 + \lambda(e_m^T(w_1 + w_2 + w_3) - 1)$, so $\lambda_1 = \hat{\lambda}$, and $(N_m(x_1^*), N_m(x_2^*), N_m(x_3^*))$ must be the minimal solution of (3.21). □

We could not get a similar theorem for Z-eigenvalue, since we could not get $\min_x \max_\lambda L(x, \lambda) = \max_\lambda \min_x L(x, \lambda)$, where $L(x, \lambda) = \mathcal{A} x^m - \lambda(x^T x - 1)$ is the lagrange function of

$$\begin{aligned} \min \quad & \mathcal{A} x^m \\ s.t. \quad & x_1^2 + x_2^2 + x_3^2 = 1, \end{aligned} \tag{3.24}$$

which defines the smallest Z-eigenvalue of $\mathcal{A}$.

3.3 Algorithms for Verifying the Positive Semi-definiteness of a Tensor

In this section, we present algorithms to verify the positive semi-definiteness of a tensor $\mathcal{A}$. It is easy to get the algorithm 3.1 from theorem 3.1(1).

Algorithm 3.1. *S1: Given a tensor $\mathcal{A} \in S_3^m$, then generate A;*

S2: Solving (3.2) to get val;

S3: If $val = 1$, report $\mathcal{A}$ is positive semi-definite; if $val = 0$, report $\mathcal{A}$ is not semi-definite.

From theorem 3.3, it is easy to see that if we wish to check $\mathcal{A}$ is positive semi-definite or not, we just need to solve (3.6) and (3.7). Because (3.7) is a linear sdp, it can be solved by any linear sdp program software. The difficult part is to solve (3.6) or (3.12). They are not convex sdp programs. We could use an alternate direction method (als) to solve (3.6), but it could not guarantee to give us the global solution of (3.6). Because we only need to check the value of (3.6), we could use different start points for als method. If one of the convergent points finds the negative optimal value, we can see that $\mathcal{A}$ is not positive semi-definite from theorem 3.3.

Algorithm 3.2. *S1: Given a tensor $\mathcal{A} \in S_3^m$, get $E = E(\mathcal{A}), e = e(\mathcal{A})$, $\varepsilon > 0$;*

S2: Solve program (3.7) to get the optimal value ν_1, if $\nu_1 < -\varepsilon$, stop and report that $\mathcal{A}$ is not positive semi-definite;

S3: Select a start point w_1^0 satisfying $\mathcal{C}^T w_1^0 \geq 0$, $\nu_2^0 = w_1^{0T} E w_1^0$ and $k = 0$;

S4: Solve

$$\begin{aligned} \min \quad & w_1^{kT} E^T w \\ s.t. \quad & \mathcal{C}^T w \geq 0 \\ & ||w||_\infty \leqslant 1 \end{aligned} \tag{3.25}$$

to get the optimal solution w_2^k;

S5: Solve

$$\begin{aligned} \min \quad & w_2^{kT} E w \\ s.t. \quad & \mathcal{C}^T w \geq 0 \\ & ||w||_\infty \leqslant 1 \end{aligned} \tag{3.26}$$

to get the optimal solution w_1^{k+1}*;*

S6: Set $\nu_2^{k+1} = w_2^{kT} E w_1^k$*, if* $\nu_2^{k+1} < -\varepsilon$*, then stop and report that* $\mathcal{A}$ *is not positive semi-definite;*

S7: If $\nu_2^{k+1} - \nu_2^k > -\varepsilon$*, stop; else goto S4,* $k = k+1$.

If algorithm 3.2 terminates at S7, we could choose another start point w_1^0. In numerical tests, we suggest to use five different points $(1, 1, \cdots, 1)^T$, $(1, 0, \cdots, 0)^T$, $(0, \cdots, 0, 1)^T$, $(1, 0, \cdots, 0, 1)^T$, and $(1, -1, 1, \cdots, -1, 1)^T$ as the start points. If algorithm 3.2 with all above start points still terminate at S7, we could conjecture that $\mathcal{A}$ is positive semi-definite.

Theorem 3.5. *If algorithm 3.2 does't stop at Step2 and Step6, then* v_2 *must converge.*

Proof. Because the feasible set of

$$\begin{array}{cc} \min & w_1^T E w_2 \\ s.t. & \mathcal{C}^T w_1 \geq 0, \\ & \mathcal{C}^T w_2 \geq 0, \\ & ||w_1||_2 \leqslant 1, ||w_2||_2 \leqslant 1, \end{array} \tag{3.27}$$

is bounded, we have the value of (3.27) is low bounded. From the algorithm 3.2, it is easy to get $v_2^k \geqslant v_2^{k+1}$, so v_2 converges. □

There is a linear sdp program in algorithm 3.1. We could use any sdp software to solve it [20, 69, 70, 74], and its result *val* is necessary and sufficient mark to check whether $\mathcal{A}$ is positive semi-definite or not. If m is small, we prefer to use algorithm 3.1. When m is large, even when $m = 16, 20$, the scale of program (3.2) are $561, 1362$, while the program of (3.6) and (3.7) are $m+1$, we prefer to use algorithm 3.2 in numerical view.

Table 3.1: The results of algorithm 3.2 for first four test examples

	the smallest Z-eigenvalue of $\mathcal{A}$	psd	the result of algorithm 3.2
(a)	0.75	Yes	Yes
(b)	-0.8571	No	No
(c)	-0.8285	No	No
(d)	-0.8285	No	No

3.4 Numerical Results

In this part, we give some numerical examples to test our theories and algorithms.

Firstly, we construct some test tensors. We present 5 kinds of $\mathcal{A}$, where the last one is random.

(a) $\mathcal{A} \in S_3^4$, $\mathcal{A}_{i,i,i,i} = i, i = 1, 2, 3$, $\mathcal{A}_{i,i,i+1,i+1} = \mathcal{A}_{i,i+1,i,i+1} = \mathcal{A}_{i+1,i+1,i,i} = \mathcal{A}_{i+1,i,i+1,i} = \mathcal{A}_{i+1,i,i,i+1} = \mathcal{A}_{i,i+1,i+1,i} = 1, i = 1, 2$, and zero otherwise.

(b) $\mathcal{A} \in S_3^4$, $\mathcal{A}_{i,i,i,i} = i, i = 1, 2, 3$, $\mathcal{A}_{i,i,i+1,i+1} = \mathcal{A}_{i,i+1,i,i+1} = \mathcal{A}_{i+1,i+1,i,i} = \mathcal{A}_{i+1,i,i+1,i} = \mathcal{A}_{i+1,i,i,i+1} = \mathcal{A}_{i,i+1,i+1,i} = -1, i = 1, 2$, and zero otherwise.

(c) $\mathcal{A} \in S_3^8$, $\mathcal{A}_{i,i,\cdots,i} = 2$, $\mathcal{A}_{i,i,\cdots,1} = \mathcal{A}_{i,i,\cdots,1,i} = \mathcal{A}_{i,1,\cdots,i} = \mathcal{A}_{1,i,\cdots,i} = 1, i = 2, 3$, and zero otherwise.

(d) $\mathcal{A} \in S_3^8$, $\mathcal{A}_{i,i,\cdots,i} = 2$, $\mathcal{A}_{i,i,\cdots,1} = \mathcal{A}_{i,i,\cdots,1,i} = \mathcal{A}_{i,1,\cdots,i} = \mathcal{A}_{1,i,\cdots,i} = -1, i = 2, 3$, and zero otherwise.

(e) $\mathcal{A} \in S_3^m$, $\mathcal{A}_{i_1,i_2,\cdots,i_m}$ is random in [-1, 1].

We will test algorithm 3.2. For testing algorithm 3.2, we first use the elimination method to calculate all Z-eigenvalues to show the positive semi-definiteness of the test examples. Then we use different start points to test algorithm 3.2. If all the results aren't right, we mark it as failure, otherwise we mark it as success. For the last random test example, we test 500 times, and get the ratio of the success.

Table 3.2: The results of algorithm 3.2 for (e)

m	the success ratio of the algorithm 3.2
4	99.8%
6	99.6%
8	99.6%
10	99.0%

From table 3.1 and table 3.2, it is easy to see that algorithm 3.2 has the good performance for verifying the positive semi-definiteness of tensors.

Chapter 4

Some Algorithms for Semidefinite Space Tensor Conic Convex Program

In this chapter, we focus on how to solve program (1.5), which is a positive semidefinite space tensor cone program. It is easy to get the KKT functions of (1.5):

$$\begin{aligned} \partial f(\mathcal{X}) + \sum_{i=1}^{p} \mu_i \partial g_i(\mathcal{X}) &= \mathcal{Y} \\ \mu \geqslant 0, g(\mathcal{X}) &\leqslant 0 \\ \mu_i g_i(\mathcal{X}) = 0, i &= 1, \cdots, p \\ \mathcal{X} \in S_{3,+}^{m}, \mathcal{Y} &\in S_{3,+}^{m,*} \\ < \mathcal{X}, \mathcal{Y} > &= 0 \end{aligned} \tag{4.1}$$

4.1 Structure and Polyhedral Outer Approximation Algorithms

4.1.1 Two Assumptions

We now make two assumptions on (1.5).

Assumption 4.1. *(1.5) has an interior point $\hat{\mathcal{X}}$.*

This assumption is needed for many algorithms [25, 26]. In the case of (1.4), it holds naturally, as by [60], any positive definite vector $\mathcal{X}$ in S_3^m is an interior point of $S_{3,+}^m$.

Assumption 4.2. *(1.5) has an optimal solution $\mathcal{X}^*$ in a simple compact region R, such as a ball ($B = \{\mathcal{A} | \|\mathcal{A}\|_2 \leqslant \Delta\}$).*

This assumption is also needed for many algorithms [25, 26]. If this assumption holds, then (1.5) is equivalent to the bounded convex program

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in R \cap S_{3,+}^m\}. \tag{4.2}$$

Proposition 4.1. *Suppose that assumption 4.1 holds and that function f is strongly convex, i.e., there is a positive constant c such that for any $\mathcal{X}, \tilde{\mathcal{X}} \in S_3^m$, we have*

$$f(\mathcal{X}) \geqslant f(\tilde{\mathcal{X}}) + < \nabla f(\tilde{\mathcal{X}}), (\mathcal{X} - \tilde{\mathcal{X}}) > + \frac{c}{2}\|\mathcal{X} - \tilde{\mathcal{X}}\|^2.$$

Then assumption 4.2 holds with

$$R = \left\{\mathcal{X} \in S_n^m | \|\mathcal{X} - \hat{\mathcal{X}}\| \leqslant \frac{2}{c}\left\|\nabla f(\hat{\mathcal{X}})\right\|\right\}.$$

Proof. Suppose that $\mathcal{X} \in S_3^m$ and $f(\mathcal{X}) \leqslant f(\hat{\mathcal{X}})$. Then

$$f(\hat{\mathcal{X}}) \geqslant f(\mathcal{X}) \geqslant f(\hat{\mathcal{X}}) + < \nabla f(\hat{\mathcal{X}}), (\mathcal{X} - \hat{\mathcal{X}}) > + \frac{c}{2}\|\mathcal{X} - \hat{\mathcal{X}}\|^2.$$

This implies that $\mathcal{X} \in R$. As R is a compact region, the conclusion follows. □

Note that for problem (1.4), the objective function is strongly convex if Q is positive definite.

4.1.2 Defining Functions

Suppose that $\phi : S_3^m \to \Re$ is a concave function and

$$S_{3,+}^m = \{\mathcal{X} \in S_n^m | \phi(\mathcal{X}) \geqslant 0\}.$$

Such as the minimal eigenvalue of $\mathcal{X}$ could be considered as a type of ϕ. Then we call ϕ a defining function of the cone $S^m_{3,+}$ and we may rewrite (1.5) as

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \phi(\mathcal{X}) \geqslant 0\}. \tag{4.3}$$

Let $\bar{\Omega}$ be a compact convex set in $\Re^3$ and the origin is an interior point of $\bar{\Omega}$. Let Ω be the boundary surface of $\bar{\Omega}$. Then

$$\phi_{\bar{\Omega}}(\mathcal{X}) = \min\{\mathcal{X}y^m : y \in \bar{\Omega}\}$$

and

$$\phi_{\Omega}(\mathcal{X}) = \min\{\mathcal{X}y^m : y \in \Omega\}$$

are concave functions as they are defined by minimization problems. Whenever $\mathcal{X}$ is positive semi-definite, $\phi_{\bar{\Omega}}(\mathcal{X}) = 0$. On the other hand, if $\mathcal{X}$ is positive definite, then $\phi_{\Omega}(\mathcal{X}) > 0$; if $\mathcal{X}$ is positive semi-definite but not positive definite, then $\phi_{\Omega}(\mathcal{X}) = 0$. Hence, both $\phi_{\bar{\Omega}}$ and ϕ_{Ω} are defining functions of $S^m_{3,+}$, but ϕ_{Ω} presents more information than $\phi_{\bar{\Omega}}$. Furthermore, when deal with space tensors, $\bar{\Omega}$ is three-dimensional, while Ω is two-dimensional. Thus, we only use ϕ_{Ω} in the following discussion. We call Ω a defining surface of $S^m_{3,+}$.

Let $\bar{\Omega} = \bar{B}$ be the unit ball in $\Re^3$. Then

$$\phi_B(\mathcal{X}) = \min\{\mathcal{X}y^m : y_1^2 + y_2^2 + y_3^2 = 1\}$$

is the smallest Z-eigenvalue function discussed in [53, 56, 60, 58].

Let $\bar{\Omega} = \bar{C}$ be the unit cube in $\Re^3$. Let $C_1 = \{(1, t_1, t_2) : -1 \leqslant t_1, t_2, \leqslant 1\}$, $C_2 = \{(t_1, 1, t_2) : -1 \leqslant t_1, t_2, \leqslant 1\}$, $C_3 = \{(t_1, t_2, 1) : -1 \leqslant t_1, t_2, \leqslant 1\}$ and $D = C_1 \cup C_2 \cup C_3$. Let

$$\phi_D(\mathcal{X}) = \min\{\mathcal{X}y^m : y \in D\}.$$

Then we see that ϕ_D is also a defining function of $S^m_{3,+}$, though D is only about a half of C, as we have $\mathcal{X}(-y)^m = \mathcal{X}y^m$ since m is even. Thus, we also call D a defining surface of $S^m_{3,+}$. Then we have

$$\phi_D(\mathcal{X}) = \min\{\phi_k(\mathcal{X}) : k = 1, 2, 3\},$$

where for $k = 1, 2, 3$,

$$\phi_k(\mathcal{X}) = \min\{d_k(t) : -1 \leqslant t_1, t_2 \leqslant 1\}, \tag{4.4}$$

$t \in \Re^2, d_1(t) = \mathcal{X}\begin{pmatrix} 1 \\ t_1 \\ t_2 \end{pmatrix}^m, d_2(t) = \mathcal{X}\begin{pmatrix} t_1 \\ 1 \\ t_2 \end{pmatrix}^m$ and $d_3(t) = \mathcal{X}\begin{pmatrix} t_1 \\ t_2 \\ 1 \end{pmatrix}^m$. As the minimization problem for defining ϕ_k only involves a two-variable vector t in the unit square, computationally, the defining function ϕ_D may be better than the defining function ϕ_B.

The defining functions ϕ_B and ϕ_D are both nonsmooth. A question is if there exists a smooth defining function or a set of smooth defining functions of $S^m_{3,+}$.

4.1.3 Polyhedral Outer Approximations

Let $F \subset \Re^3$ be a finite set. Let $y \in F$. By the definition of $S^m_{3,+}$, for any $\mathcal{X} \in S^m_{3,+}$, we have

$$\mathcal{X}y^m = \sum_{i=0}^{m}\sum_{j=0}^{m-i} E(\mathcal{X})_{i+1,j+1} y_1^i y_2^j y_3^{m-i-j} \geqslant 0, \tag{4.5}$$

where $E(\mathcal{X})$ is defined in section 3.1.2. Note that this is a linear constraint with respect to $\mathcal{X}$. Let

$$S^m_{3,+}(F) = \{\mathcal{X} \in S^m_3 | \mathcal{X}y^m \geqslant 0, \ \forall y \in F\}.$$

Then $S^m_{3,+}(F)$ is a polyhedral cone and $S^m_{3,+} \subset S^m_{3,+}(F)$. We call $S^m_{3,+}(F)$ an polyhedral out approximation of $S^m_{3,+}$, generated by the finite set F. We may use $S^m_{3,+}(F)$ to relax the nonsmooth constraints of (1.5). We have the following twice continuously differentiable convex program:

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in S^m_{3,+}(F)\}, \tag{4.6}$$

The twice continuously differentiable convex program (4.6) is a relaxation of the nonsmooth convex program (1.5). We may solve (4.6) by a conventional method. Suppose that $\mathcal{X}_F$ is a solution of (4.6). If $\mathcal{X}_F \in S^m_{3,+}$, then $\mathcal{X}_F$ is also an optimal solution of (4.6). Otherwise, $f(\mathcal{X}_F)$ is a lower bound of (4.6). Furthermore, we have the following error bound result.

Proposition 4.2. *Suppose that assumption 4.1 holds and ϕ is a defining function of $S^m_{3,+}$. Assume that (4.6) has an optimal solution $\mathcal{X}_F$. If $\phi(\mathcal{X}_F) < 0$, then we have*

$$0 \leqslant f^* - f(\mathcal{X}_F) \leqslant \sigma \left[f(\hat{\mathcal{X}}) - f(\mathcal{X}_F) \right],$$

where

$$f^* = \inf\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in S^m_3\},$$

$$\sigma = \frac{-\phi(\mathcal{X}_F)}{\phi(\hat{\mathcal{X}}) - \phi(\mathcal{X}_F)}, \hat{\mathcal{X}} \in S^m_{3,+}$$

and $0 < \sigma < 1$.

Proof. Since (4.6) is a relaxation of (1.5), we have $0 \leqslant f^* - f(\mathcal{X}_F)$. Since $\hat{\mathcal{X}} \in S^m_{3,+}$ and $\phi(\mathcal{X}_F) < 0$, we have $0 < \sigma < 1$. Then

$$\phi((1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}}) \geqslant (1-\sigma)\phi(\mathcal{X}_F) + \sigma\phi(\hat{\mathcal{X}}) = 0.$$

Thus, $(1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}} \in S^m_{3,+}$. Since $g(\mathcal{X}_F) \leqslant 0$ and $g(\hat{\mathcal{X}}) \leqslant 0$, we have $g((1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}}) \leqslant 0$. This implies that $(1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}}$ is a feasible point of (1.5). We have $f^* \leqslant f((1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}})$. Thus, we have

$$\begin{array}{rcl} 0 & \leqslant & f^* - f(\mathcal{X}_F) \\ & \leqslant & f((1-\sigma)\mathcal{X}_F + \sigma\hat{\mathcal{X}}) - f(\mathcal{X}_F) \\ & \leqslant & \sigma[f(\hat{\mathcal{X}}) - f(\mathcal{X}_F)]. \end{array}$$

This proves the proposition. □

This proposition says that if $-\phi(\mathcal{X}_F)$ is small, then $f(\mathcal{X}_F)$ is close to f^*. In the next theorem, we will show that if F is denser on a compact defining surface Ω, then $f(\mathcal{X}_F)$ is closer to f^*. We now need assumption 4.2. Instead of solving (4.6), we now solve

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in R \cap S^m_{3,+}(F)\}. \tag{4.7}$$

Theorem 4.1. *Suppose that assumptions 4.1 and 4.2 hold. Let $F \subset \Omega$, where Ω is a compact defining surface of $S^m_{3,+}$. Let $\phi = \phi_\Omega$ be the defining function associated with Ω. Then (4.7) has an optimal solution $\mathcal{X}_F$. If $\phi(\mathcal{X}_F) \geqslant 0$, then $\mathcal{X}_F$ is an optimal solution of (1.5) and (4.2). If $\phi(\mathcal{X}_F) < 0$, then we have*

$$-\phi(\mathcal{X}_F) \leqslant M\rho(F)$$

and

$$0 \leqslant f(\mathcal{X}^*) - f(\mathcal{X}_F) \leqslant \bar{\sigma}\left[f(\hat{\mathcal{X}}) - f(\mathcal{X}_F)\right],$$

where $\mathcal{X}^*$ is an optimal solution of (4.2),

$$\bar{\sigma} = \frac{M\rho(F)}{\phi(\hat{\mathcal{X}}) + M\rho(F)},$$

$0 < \bar{\sigma} < 1$, M is a constant depending on R and Ω only, and

$$\rho(F) = \max\{\mathrm{dist}(y, F) : y \in \Omega\}.$$

Proof. By assumptions 4.1 and 4.2, (4.7) has an optimal solution. Denote this optimal solution as $\mathcal{X}_F$. If $\phi(\mathcal{X}_F) \geqslant 0$, then the conclusions hold obviously. Suppose that $\phi(\mathcal{X}_F) < 0$. Suppose that y_F is an optimal solution of

$$\phi(\mathcal{X}_F) = \min\{\mathcal{X}_F y^m : y \in \Omega\}.$$

Then $\phi(\mathcal{X}_F) = \mathcal{X}_F y_F^m$. Let $\bar{y}$ be the the closest point in F to y_F. Then $\mathcal{X}_F \bar{y}^m \geqslant 0$ and

$$\|y_F - \bar{y}\| \leqslant \rho(F).$$

Let M be the Lipschitz constant of the function $L(\mathcal{X}, y) \equiv \mathcal{X} y^m$ on the compact set $R \times \Omega$. Then

$$-\phi(\mathcal{X}_F) = -\mathcal{X}_F y_F^m \leqslant \mathcal{X}_F \bar{y}^m - \mathcal{X}_F y_F^m \leqslant M \rho(F).$$

The remaining conclusions now follow from proposition 4.2. □

4.1.4 The Basic Polyhedral Outer Approximation Algorithm

By theorem 4.1, if we let $F_k \subset \Omega$ such that $\rho(F_k) \to 0$ as $k \to \infty$, and denote $\mathcal{X}^{(k)} = \mathcal{X}_{F_k}$, where $\mathcal{X}_{F_k}$ is an optimal solution of (4.7) with $F = F_k$, then we have a polyhedral outer approximation algorithm. To distinguish this algorithm from its improved version in next subsection, we call this algorithm the basic polyhedral outer approximation algorithm.

Theorem 4.2. *Under assumptions 4.1 and 4.2, we have*

$$\lim_{k \to \infty} f(\mathcal{X}^{(k)}) = f^*,$$

where f^* is the optimal objective function value of (4.2), and any accumulation point of $\{\mathcal{X}^{(k)}\}$ is an optimal solution of (4.2).

The proof of theorem 4.2 follows the proof of theorem 4.1. We omit it. We now discuss possible schemes to construct F_k. Let $\Omega = D$, the closed half surface of the unit cube. Let F_0 be the set of the seven vertices of D. Let F_k be the set of grid points on D, with the grid length as $\frac{1}{2^{k-1}}$, for $k = 0, 1, \cdots$. Then $\rho(F_k) = \frac{\sqrt{2}}{2^k}$. Furthermore, we have $F_k \subset F_{k+1}$ and hence $f(\mathcal{X}^{(k)}) \leqslant f(\mathcal{X}^{(k+1)})$ for all k. Denote $\mathcal{S}_k = S_{3,+}^m(F_k)$ for $k = 0, 1, \cdots$. Then from (4.5), we have

$$\mathcal{S}_0 = \left\{ \begin{array}{ll} \mathcal{X} \in S_3^m : & \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{X})_{i+1,j+1} \geqslant 0, \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{X})_{i+1,j+1} (-1)^{i+j} \geqslant 0, \\ & \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{X})_{i+1,j+1} (-1)^{j} \geqslant 0, \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{X})_{i+1,j+1} (-1)^{i} \geqslant 0 \end{array} \right\} \quad (4.8)$$

and

$$\mathcal{S}_1 = \left\{ \begin{array}{ll} & E(\mathcal{X})_{m+1,1} \geqslant 0, E(\mathcal{X})_{1,m+1} \geqslant 0, E_{1,1} \geqslant 0, \\ \mathcal{X} \in \mathcal{S}_0 : & \sum_{i=0}^{m} E(\mathcal{X})_{i+1,m} \geqslant 0, \sum_{i=0}^{m} E(\mathcal{X})_{i+1,m-i+1}(-1)^i \geqslant 0, \\ & \sum_{i=0}^{m} E(\mathcal{X})_{i+1,1} \geqslant 0, \sum_{i=0}^{m} E(\mathcal{X})_{i+1,1}(-1)^i \geqslant 0, \\ & \sum_{i=0}^{m} E(\mathcal{X})_{1,i+1} \geqslant 0, \sum_{i=0}^{m} E(\mathcal{X})_{1,i+1}(-1)^i \geqslant 0 \end{array} \right\} \qquad (4.9)$$

4.1.5 Defining Sequences and An Iterative Polyhedral Outer Approximation Algorithm

In the basic polyhedral outer approximation algorithm, the number of constraints in (4.7) is huge when k is big. This is not practical. Furthermore, the sequence $\{\mathcal{X}^{(k)} : k = 0, 1, \cdots\}$ is not iterative. The point $\mathcal{X}^{(k+1)}$ is obtained without using the knowledge of $\mathcal{X}^{(k)}$. We now explore an improved version of that algorithm. We call a sequence in $\Re^3$ a defining sequence of $S_{3,+}^m$ if for any $\mathcal{X} \in S_3^m$, $\mathcal{X}$ is positive definite if and only if $\mathcal{X}y^m \geqslant 0$ for any y in that sequence. We see that the sequence consisting of all points in F_k for all k is such a defining sequence. We now give a general formula for this sequence. As $\mathcal{X}y^m$ is an even function, for grid points y and $-y$, we only need to include one of them. Thus, in F_0, we only need to include four vertices:

$$G_0 = \{(1, 1, 1), (1, 1, -1), (1, -1, 1), (-1, 1, 1)\}.$$

We see that $\mathcal{S}_0 = S_{3,+}^m(G_0)$. Then for any $\mathcal{X} \in S_3^m$, $\mathcal{X}$ is positive semi-definite if and only if $\mathcal{X}y^m \geqslant 0$ for any $y \in \Gamma \equiv G_0 \cup D_1 \cup D_2 \cup D_3$, where

$$D_1 = \{(1, t_1, t_2) : -1 \leqslant t_1 \leqslant 1, -1 < t_2 < 1\},$$

$$D_2 = \{(t_2, 1, t_1) : -1 \leqslant t_1 \leqslant 1, -1 < t_2 < 1\},$$

$$D_3 = \{(t_1, t_2, 1) : -1 \leqslant t_1 \leqslant 1, -1 < t_2 < 1\}.$$

We see that G_0, D_1, D_2 and D_3 are disjoint each other, and if $y \in \Gamma$ then $-y \notin \Gamma$. Hence, we may take all the grid points from Γ. Let

$$G_k = (F_k \backslash F_{k-1}) \cap \Gamma$$

for $k = 1, 2, \cdots$. Then the set

$$G = G_0 \cup G_1 \cup G_2 \cup \cdots$$

is what we want. We may write out the general formula for points in G_k as: $(t_1, t_2, 1)$, $(t_2, 1, t_1)$, $(t_1, t_2, 1)$ for $t_1 = -1 + \frac{i}{2^{k-1}}$ and $t_2 = -1 + \frac{2j-1}{2^{k-1}}$, for $i = 0, \cdots, 2^k$ and $j = 1, \cdots, 2^{k-1}$, and $(t_1, t_2, 1), (t_2, 1, t_1), (t_1, t_2, 1)$ for $t_1 = -1 + \frac{2i-1}{2^{k-1}}$ and $t_2 = -1 + \frac{j}{2^{k-2}}$, for $i = 1, \cdots, 2^{k-1}$ and $j = 1, \cdots, 2^{k-1} - 1$. Then $|G_k| = 9 \times 2^{2k-2}$.

We now give an iterative polyhedral outer approximation algorithm. Suppose that assumptions 4.1 and 4.2 hold.

Algorithm 4.1. *(An Iterative Polyhedral Outer Approximation Algorithm)*

S1: Let $k = 0$, $\mu_{-1} = 1$ and $F = G_0$. Let N be a positive integer.

S2: Let $\mu_k = \mu_{k-1}$. Compute an optimal solution of (4.7) and denote it as $\mathcal{X}^{(k)}$.

S3: If $\mu_k \geqslant N$, stop. Otherwise, let $H_k = \{y \in G_{\mu_k} : \mathcal{X}^{(k)} y^m < 0\}$.

S4: If $H_k = \emptyset$, let $\mu_k = \mu_k + 1$ and go to S3. Otherwise, let $F = H_k \cup F$ and $k = k + 1$. Go to Step 1.

Theorem 4.3. *Suppose that assumptions 4.1 and 4.2 hold. Then we have $f(\mathcal{X}^{(k)}) \leqslant f(\mathcal{X}^{(k+1)}) \leqslant f(\mathcal{X}^*)$ for all k. If $N \to \infty$, then we have a sequence $\{\mathcal{X}^{(k)} : k = 0, 1, \cdots\} \subset R$. If $\mathcal{X}^{**}$ is an accumulation point of $\{\mathcal{X}^{(k)}\}$, then $\mathcal{X}^{**}$ is an optimal solution of (1.5).*

Proof. At the kth iteration, $\mathcal{X}^{(k)}$ is an optimal solution of (4.7). On the other hand, $\mathcal{X}^{(k+1)}$ is an optimal solution of

$$\min\{f(\mathcal{X}) : g(\mathcal{X}) \leqslant 0, \mathcal{X} \in R \cap S^m_{3,+}(H_k \cup F)\}, \tag{4.10}$$

which is a restriction of (4.7). Hence, $f(\mathcal{X}^{(k)}) \leqslant f(\mathcal{X}^{(k+1)})$. As (4.7) and (4.10) are relaxations of (4.2), we have $f(\mathcal{X}^{(k)}) \leqslant f(\mathcal{X}^{(k+1)}) \leqslant f(\mathcal{X}^*)$ for all k. This proves the first conclusion.

Suppose that $N \to \infty$ and $\mathcal{X}^{(k_i)} \to \mathcal{X}^{**}$, where $\{k_i : i = 0, 1, \cdots\}$ is a subsequence of $\{0, 1, \cdots\}$. Assume that $\phi(\mathcal{X}^{**}) = -\epsilon < 0$. Then there is $y^{**} \in D$ such that $\mathcal{X}^{**}(y^{**})^m = \phi(\mathcal{X}^{**}) = -\epsilon < 0$. Because of the structure of G_μ, there is $K_1 > 0$ and $\delta > 0$ such that for all $\mu \geqslant K_1$, there is a $y_\mu \in G_\mu$ such that $y_\mu \in N(y^{**}; \delta) \equiv \{y \in D : \|y - y^{**}\| \leqslant \delta\}$ and $\mathcal{X}^{**}(y_\mu) \leqslant -\frac{2}{3}\epsilon$. Since $\mathcal{X}^{(k_i)} \to \mathcal{X}^{**}$, there is a $K_2 \geqslant K_1$ such that for all $k_i \geqslant K_2$ and $y \in N(y^{**}; \delta)$,

$$\left|\mathcal{X}^{(k_i)} y^m - \mathcal{X}^{**} y^m\right| \leqslant \frac{1}{3}\epsilon. \tag{4.11}$$

As $\mu_{k_i} \geqslant k_i$, we have $y_{\mu_{k_i}} \in N(y^{**}; \delta)$ such that $\mathcal{X}^{**}(y_{\mu_{k_i}})^m \leqslant -\frac{2}{3}\epsilon$. Combining it with (4.11), we have

$$\mathcal{X}^{(k_i)}(y_{\mu_{k_i}})^m \leqslant -\frac{1}{3}\epsilon < 0 \tag{4.12}$$

and

$$\mathcal{X}^{(k_{i+1})}(y_{\mu_{k_i}})^m \leqslant -\frac{1}{3}\epsilon < 0. \tag{4.13}$$

According to (4.12), we have $y_{\mu_{k_i}} \in H_k$. This implies that $y_{\mu_{k_i}} \in F$ forever after the kth iteration. Thus

$$\mathcal{X}^{(k_{i+1})}(y_{\mu_{k_i}})^m \geqslant 0,$$

as $\mathcal{X}^{(k_{i+1})}$ is an optimal solution of (4.13), with $y_{\mu_{k_i}} \in F$ now. This contradicts (4.13). Hence, we have $\phi(\mathcal{X}^{**}) \geqslant 0$. This shows that $\mathcal{X}^{**}$ is an optimal solution of (1.5). The proof of the theorem is completed. □

4.1.6 Another Iterative Polyhedral Outer Approximation Algorithm

In algorithm 4.1, the set F expands steadily as k increases. Hence, the number of constraints in (4.7), though is much less than the number of constraints in (4.6) in the basic polyhedral outer approximation algorithm, is still somewhat large when k is big. We now give another iterative polyhedral outer approximation algorithm to improve this. Suppose that assumptions 4.1 and 4.2 hold.

Algorithm 4.2. *(Another Iterative Polyhedral Outer Approximation Algorithm)*

S1: Let $k = 0$, $\mu_{-1} = 1$ and $F = G_0$. Let N be a positive integer.

S2: Let $\mu_k = \mu_{k-1}$. Compute an optimal solution of (4.7) and denote it as $\mathcal{X}^{(k)}$.

S3: If $\mu_k \geqslant N$, stop. Otherwise, let $H_k = \{y \in G_{\mu_k} : \mathcal{X}^{(k)}y^m < 0\}$.

S4: If $H_k = \emptyset$, let $\mu_k = \mu_k + 1$ and go to S3. Otherwise, let $\bar{F} = \{y \in F : \mathcal{X}^{(k)}y^m = 0\}$. Let $F = H_k \cup \bar{F}$ and $k = k + 1$. Go to S2.

Theorem 4.4. *Suppose that assumptions 4.1 and 4.2 hold. Then we have $f(\mathcal{X}^{(k)}) \leqslant f(\mathcal{X}^{(k+1)}) \leqslant f(\mathcal{X}^*)$ for all k. If $N \to \infty$, then we have a sequence $\{\mathcal{X}^{(k)} : k = 0, 1, \cdots\} \subset R$. If furthermore $\{\mathcal{X}^{(k)}\}$ converges to a point $\mathcal{X}^{**}$, then $\mathcal{X}^{**}$ is an optimal solution of (1.5).*

The proof of this theorem is similar to the proof of theorem 4.3. We omit it.

Remark We see that the number of constraints in (4.7) is stable in algorithm 4.2. A cost of this improvement is that the second conclusion of theorem 4.4 is a little weaker than the second conclusion of theorem 4.3. □

4.2 The Other Approaches

4.2.1 The Conic Linear Program Approach

If f is a convex quadratic function and g is an affine function, then (1.5) can be converted to a conic linear program (CLP) via a second-order cone transformation [60]. Theoretically, CLP problems can be solved by polynomial-time interior point algorithms [40, 41]. In this approach, a self-concordant function needs to be established. The smallest Z-eigenvalue function $\phi_B(\mathcal{X})$ has some properties of a self-concordant function. But it is nonsmooth and thus does not satisfy the differentiability requirements of a self-concordant function. Further investigation on possible smooth defining functions of the positive semi-definite space tensor cone $S^m_{3,+}$ is needed to establish a self-concordant function.

A closed convex cone is a symmetric cone if it is a self-dual and homogeneous cone [19]. However, unlike the semi-definite problem (SDP) and the second-order cone problem (SOCP), where the semi-definite matrix cone and the second-order cone are self-dual, the positive semi-definite space tensor cone $S^m_{3,+}$ is not self-dual [60]. We may check if the positive semi-definite space tensor cone $S^m_{3,+}$ is a homogeneous cone or not. A closed convex cone $\mathcal{K}$ with nonempty interior is homogeneous if for any two interior points u and v of $\mathcal{K}$, there exists an invertible linear mapping T such that $T(\mathcal{K}) = \mathcal{K}$ and $T(u) = v$, i.e., the group of automorphisms of $\mathcal{K}$ acts transitively on the interior of $\mathcal{K}$. If the positive semi-definite space tensor cone $S^m_{3,+}$ is a homogeneous cone, then we may use the techniques on homogeneous cones [28] to construct interior point algorithms to solve the CLP with the conic constraint $\mathcal{X} \in S^m_{3,+}$. If the positive semi-definite space tensor cone $S^m_{3,+}$ is not a homogeneous cone, then we may further check if it is a hyperbolic cone [28, 62] or not. Hyperbolic cones contain homogeneous cones as a subclass [28, 62]. If the positive semi-definite space tensor cone $S^m_{3,+}$ is a hyperbolic cone, then we may use the techniques on hyperbolic cones [28, 62] to construct interior point algorithms to solve the concerned CLP. Finally, it is possible that the positive semi-definite space tensor cone $S^m_{3,+}$ is not a hyperbolic cone. Then we will analyze the properties of the positive semi-definite space tensor cone further, to see what kind of interior point algorithms are suitable for solving the concerned CLP.

Therefore, the conic linear program approach for solving this problem is not ready for practical use at this moment.

4.2.2 The Nonsmooth Convex Program Approach

We now discuss algorithms for solving the nonsmooth convex program (1.5). Under assumptions 4.1 and 4.2, we may covert it to a standard convex feasibility problem [25, 26]. A convex feasibility problem is to compute a point in a convex set $\bar{\mathcal{S}}$, where $\bar{\mathcal{S}}$ is contained in a compact set R, and is assumed to contain an interior [25, 26]. Then we need an oracle which for every point $\bar{x} \in R$ returns either a statement that $\bar{x}$ is feasible, or a cutting plane to separate $\bar{x}$ from the feasible set. With such an oracle, we may apply the analytic center cutting plane method in [25, 26], and obtain a convergence estimate in $O(n(\log 1/\epsilon)^2)$ calls to the oracle. As f and g are twice smooth, it is easy to handle them. The key part of this oracle is to solve

$$\phi_D(x) = \min\{\mathcal{X}y^m : y \in D\}.$$

We have the following proposition.

Proposition 4.3. *Given $\mathcal{X} \in S^m_3$. Then $\mathcal{X} \in S^m_{3,+}$ if and only if the following three conditions are satisfied:*

(i) $\mathcal{X} \in \mathcal{S}_0$;

(ii) for any stationary point s of ψ_k for $-1 \leqslant s \leqslant 1$ and $k = 1, \cdots, 6$, we have $\psi_k(s) \geqslant 0$, where $\psi_1(s) = \mathcal{X}\begin{pmatrix} 1 \\ 1 \\ s \end{pmatrix}^m$, $\psi_2(s) = \mathcal{X}\begin{pmatrix} 1 \\ -1 \\ s \end{pmatrix}^m$, $\psi_3(s) = \mathcal{X}\begin{pmatrix} 1 \\ s \\ 1 \end{pmatrix}^m$, $\psi_4(s) =$

$\mathcal{X}\begin{pmatrix}1\\s\\-1\end{pmatrix}^m$, $\psi_5(s) = \mathcal{X}\begin{pmatrix}s\\1\\1\end{pmatrix}^m$ *and* $\psi_6(s) = \mathcal{X}\begin{pmatrix}s\\1\\-1\end{pmatrix}^m$*;*

(iii) for any stationary point t of d_k, where $-1 \leqslant t_1, t_2 \leqslant 1$ and $k = 1, 2, 3$, we have $d_k(t) \geqslant 0$.

Proof. The "only if" part follows from the definition of positive semi-definiteness of $\mathcal{X}$. The "if" part follows from (4.4) and the fact that ϕ_D is a defining function of $S^m_{3,+}$. Note that C has 12 edges and D contains 9 edges. Because the fact that $\mathcal{X}y^m$ is an even function for y, we only need to consider six edges in (ii). □

The condition (i) is easy to check. It is not difficult to find stationary points of ψ_k for $k = 1, 2, 3$. To find stationary points of d_k, we need to solve

$$\begin{cases} \frac{\partial}{\partial t_1} d_k(t) &= 0, \\ \frac{\partial}{\partial t_2} d_k(t) &= 0. \end{cases}$$

This is a two-variable polynomial system. We may use the Sylvester formula [15] to solve it.

4.2.3 The Bi-Level Program Approach

We can write our problem (1.5) into the following bi-level program problem:

$$\begin{array}{ll} \min & f(\mathcal{X}) \\ s.t. & g(\mathcal{X}) \leqslant 0, \\ & \min\limits_{y \in \Re^3} \mathcal{X}y^m \geqslant 0. \end{array}$$

Due to the harmonic property of the function $\mathcal{X}y^m$, the above program is equivalent to

$$\begin{array}{ll} \min & f(\mathcal{X}) \\ s.t. & g(\mathcal{X}) \leqslant 0, \\ & \phi(\mathcal{X}) = \min\limits_{y \in \Re^3, ||y||=1} \mathcal{X}y^m \geqslant 0. \end{array} \tag{4.14}$$

As discussed in subsection 4.1.3, $\phi(\mathcal{X})$ is a concave function. Thus (4.14) is a convex program. The dual program can be written as

$$\begin{aligned} \max_{\tau \in \Re^p, \rho \in \Re} \quad & L(\tau, \rho) \\ s.t. \quad & \tau \geqslant 0, \rho \geqslant 0, \end{aligned} \tag{4.15}$$

where

$$L(\tau, \rho) = \min_{\mathcal{X}} \{ f(\mathcal{X}) + \tau^T g(\mathcal{X}) - \rho \phi(\mathcal{X}) \}. \tag{4.16}$$

Thus we can apply a dual algorithm for solving (4.15).

Algorithm 4.3. *(Dual Algorithm for Structured Convex Problems)*

S1: $\tau_0 = 0, \rho_0 = 0,$ $k = 0.$

S2: Compute

$$z^{(k)} := \nabla L(\tau_k, \rho_k) = \begin{pmatrix} \nabla_\tau L(\tau_k, \rho_k) \\ \nabla_\rho L(\tau_k, \rho_k) \end{pmatrix}. \tag{4.17}$$

S3: if $z^{(k)} \leqslant 0$ *and*

$$\left(z^{(k)}\right)^T \begin{pmatrix} \tau_k \\ \rho_k \end{pmatrix} = 0,$$

then stop.

Carry out a line search, namely computing $\alpha_k > 0$ *and let*

$$\begin{pmatrix} \tau_{k+1} \\ \rho_{k+1} \end{pmatrix} := \left[\begin{pmatrix} \tau_k \\ \rho_k \end{pmatrix} + \alpha_k z^{(k)} \right]_+ ,$$

$k := k + 1,$ *and go to S2.*

It is easy to show that the above algorithm is convergent if certain line search conditions are satisfied. Indeed, the above algorithm is a truncated gradient method for the dual problem. The analysis of Calamai and Moré [14] can be used. Namely, we can try

$$\alpha_k = \min[2\alpha_{k-1},\, \gamma]$$

first, and if necessary, we reduce α_k by a fractor of 2 until the line search condition

$$L(\tau_{k+1}, \rho_{k+1}) \geqslant L(\tau_k, \rho_k) + c_1(\nabla L(\tau_k, \rho_k))^T \begin{pmatrix} \tau_{k+1} - \tau_k \\ \rho_{k+1} - \rho_k \end{pmatrix}$$

is satisfied, where $\gamma > 0$ and $c_1 \in (0, 0.5)$ are constants. More details can be found in chapter 11 of Sun and Yuan [67].

4.3 Numerical Results

We report some numerical experiments for algorithm 4.1 and algorithm 4.2. For algorithm 4.1 and 4.2, we take $N = 9$. When k is equal to N, there are 9×2^{16} points in G_k and $\rho(F_k)$ is about 2^{-8}. The problems we test for algorithm 4.1 and 4.2 are:

Example 4.1.

$$\min \left\{ \frac{1}{2}(H(\mathcal{X}) - H(\bar{\mathcal{X}}))^T Q (H(\mathcal{X}) - H(\bar{\mathcal{X}})) : AE(\mathcal{X}) \leqslant b, \mathcal{X} \in S^m_{3,+} \right\}$$

Example 4.2.

$$\min \left\{ \frac{1}{2}(H(\mathcal{X}) - H(\bar{\mathcal{X}}))^T Q (H(\mathcal{X}) - H(\bar{\mathcal{X}})) : \mathcal{X} \in S^m_{3,+} \right\}$$

In examples 4.1 and 4.2, $H(\mathcal{X})$ is a map from S^m_3 to $\Re^{\frac{(m+1)(m+2)}{2}}$, $Q = PDP^T$ is a positive definite matrix, where the orthogonal matrix P and the diagonal matrix D as well as $\bar{\mathcal{X}} \in S^m_3$ are all generated randomly. To make it easy to compare, we set

$A = (1, 1 \cdots, 1) \in \Re^{1\times \frac{(m+1)(m+1)}{2}}$, $b = \frac{(m+1)(m+2)}{2}$. Example 4.2 is the PSDT model in [58].

We test 7 different orders for every algorithms, 4 small orders and 3 lager orders: m=4; m=8; m=12; m=16; m=20; m=30; m=40, the dimensions of $H(\mathcal{X})$ are 15, 45, 91, 153, 231, 496, 861, respectively.

The $num(Con)$, $num(G)$, $\lambda_{min}\left(\mathcal{X}^{(k)}\right)$ and k in the following tables mean the number of the constraints in the relaxation program, the number of points in G_k, the minimum Z-eigenvalue of $\mathcal{X}^{(k)}$ and the number of iteration when Algorithm stops. We use the method in [58] to calculate all the eigenvalues and eigenvectors of $\mathcal{X}^{(k)}$, and take the least one as $\lambda_{min}\left(\mathcal{X}^{(k)}\right)$. Failure means that the time Algorithm cost is too long or the data overflows from the Ram. At each order we test 5 examples, and get the averages of $time(s)$ $num(Con)$ and $\lambda_{min}\left(\mathcal{X}^{(k)}\right)$.

Table 4.1: Example 4.1 for algorithms 4.1 and 4.2 in the small order case

		Algorithm 4.1			Algorithm 4.2		
$num(G)$	m	$time(s)$	$num(Con)$	$\lambda_{min}\left(\mathcal{X}^{(k)}\right)$	$time(s)$	$num(Con)$	$\lambda_{min}\left(\mathcal{X}^{(k)}\right)$
9×2^{16}	4	0.1187	7.4	-1.79×10^{-4}	0.1156	7.6	-1.79×10^{-4}
9×2^{16}	8	0.947	11.8	-2.75×10^{-4}	0.778	7.6	-1.85×10^{-4}
9×2^{16}	12	1.861	15.2	-1.36×10^{-3}	1.713	12	-6.1×10^{-5}
9×2^{16}	16	6.685	31	-1.31×10^{-4}	4.77	17.6	-7.57×10^{-5}

Table 4.2: Example 4.2 for algorithms 4.1 and 4.2 in the small order case

		Algorithm 4.1			Algorithm 4.2		
$num(G)$	m	$time(s)$	$num(Con)$	$\lambda_{min}\left(\mathcal{X}^{(k)}\right)$	$time(s)$	$num(Con)$	$\lambda_{min}\left(\mathcal{X}^{(k)}\right)$
9×2^{16}	4	0.0844	5.4	-1.46×10^{-4}	0.1344	5.8	-2.6×10^{-4}
9×2^{16}	8	0.6438	10	-1.69×10^{-4}	0.8218	10.2	-3.2×10^{-4}
9×2^{16}	12	1.437	14.4	-6.6×10^{-5}	2.034	18.6	-1.42×10^{-4}
9×2^{16}	16	6.929	17	-1.13×10^{-4}	5.809	27	-1.16×10^{-4}

We see that algorithm 4.1 and 4.2 can be used to solve some large scale problems. The failure in table was caused by the storage problem. It is worth improving the

Table 4.3: Example 4.1 for algorithms 4.1 and 4.2 in the lager order case

		Algorithm 4.1		Algorithm 4.2	
$num(G)$	m	$time(s)$	$num(Con)$	$time(s)$	$num(Con)$
9×2^{16}	20	13.2	38.4	13.5	38
9×2^{16}	30	152	64	144.1	34
9×2^{16}	40	failure		failure	

Table 4.4: Example 4.2 for algorithms 4.1 and 4.2 in the lager order case

		Algorithm 4.1		Algorithm 4.2	
$num(G)$	m	$time(s)$	$num(Con)$	$time(s)$	$num(Con)$
9×2^{16}	20	13.9	40	14.1	38.4
9×2^{16}	30	136.3	66.2	141.1	33.4
9×2^{16}	40	665.9	81	695.1	129

storage use in these two algorithms. On the other hand, algorithm 4.1 is efficient for solving some small order problems for $Q = I$.

Chapter 5

An Application of Semidefinite Space Tensor Conic Convex Program

In this chapter, we present a nonnegative ODF model. The ODF values are strictly nonnegative in our model.

5.1 Nonnegative Diffusion Orientation Distribution Function

Suppose that we use an m order symmetric space tensor $\mathcal{A}$ to denote the raw HARDI signal S. Here m should be an even number as the signal is antipodally symmetric. Let $x = (x_1, x_2, x_3)$ be a unit direction. Then the HARDI signal at the direction x has the value

$$S(x) = \mathcal{A}x^m = \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{A})_{i+1,j+1} x_1^i x_2^j x_3^{m-i-j}. \tag{5.1}$$

Obviously, there are

$$\bar{n} = \sum_{i=1}^{m+1} i = \frac{1}{2}(m+1)(m+2)$$

independent components $E(\mathcal{A})$. Let $k = j + 1 + i(2m + 3 - i)/2$, $s_k = E(\mathcal{A})_{i-1,j-1}$ and $\hat{x}_k = x_1^i x_2^j x_3^{m-i-j}$. We set the map $\mathcal{A}$ to s as $H(\mathcal{A}) : S_3^m \mapsto \Re^{\frac{(m+1)(m+2)}{2}}$. Then we may rewrite (5.1) as

$$S(x) = s^T \hat{x},$$

i.e., we may regard $S(x)$ as the scalar product of vectors s and $\hat{x}$ in $\Re^{\bar{n}}$. This point of view will be useful later.

Similarly, we may use an m order symmetric tensor $\mathcal{B}$ to denote the ODF Ψ. Let $x = (x_1, x_2, x_3)$ be a unit direction. Then the ODF value at the direction x has the value

$$\Psi(x) = \mathcal{B}x^m = \sum_{i=0}^{m} \sum_{j=0}^{m-i} E(\mathcal{B})_{i+1,j+1} x_1^i x_2^j x_3^{m-i-j} = u^T \hat{x}, \tag{5.2}$$

and $u = (u_1, \cdots, u_k, \cdots, u_{\bar{n}})^T$ is a vector in $\Re^{\bar{n}}$ with $u_k = E(\mathcal{B})_{i,j}$.

We have seen that from (5.1) and (5.2), the properties of $\mathcal{A}$ and $\mathcal{B}$ could be presented by s and u. So, for convenience, we say that:

Definition 5.1. *We say that a vector $s = H(\mathcal{A}) \in \Re^{\frac{(m+1)(m+2)}{2}}$ has a property iff the corresponding tensor $\mathcal{A} \in S_3^m$ has this property.*

Let

$$\Omega = \{x \in \Re^3 : x_1^2 + x_2^2 + x_3^2 = 1\}.$$

We also use spherical co-ordinates (θ, ϕ), $0 \leqslant \theta \leqslant \pi, 0 \leqslant \phi \leqslant 2\pi$, with

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{pmatrix}.$$

Let Y_l^q denote the spherical harmonics (SH) of order l and degree q. Explicitly, it is given as follows

$$Y_l^q = \sqrt{\frac{2l+1}{4\pi} \cdot \frac{(l-q)!}{(l+q)!}} P_l^q(\cos\theta) e^{iq\phi},$$

where P_l^q is an associated Legendre polynomials, which can be obtained analytically from the following set of equations,

$$P_l(x) = \frac{1}{2^l l!}(\frac{d}{dx})^l (x^2-1)^l,$$

$$P_l^q(x) = (-1)^q (1-x^2)^{\frac{q}{2}} (\frac{d}{dx})^q P_l(x), \quad q \geqslant 0$$

$$P_l^{-q}(x) = (-1)^q \frac{(l-q)!}{(l+q)!} P_l^q(x).$$

Let $l = 0, 2, 4, \cdots, m$ and $q = -l, \cdots, 0, \cdots, l$. A single index p in terms of l and q is used such that $p \equiv p(l,q) = (l^2+l+2)/2+q$. Then $p = 1, \cdots, n$. If $p = p(l,q)$, then define $l(p) = l$. Explicitly, if $(l^2-l+2)/2 \leqslant p \leqslant (l^2+3l+2)/2$, then $l(p) = l$. Consequently, we have $l(1) = 0$, $l(p) = 2$ for $2 \leqslant p \leqslant 6$, $l(p) = 4$ for $7 \leqslant p \leqslant 15$, so on.

As in [8, 16, 17], the real spherical harmonics of order less than or equal to m, are

$$R_p(\theta,\phi) = \begin{cases} \sqrt{2} Re(Y_l^{|q|}), & \text{if } -l \leqslant q \leqslant 0, \\ \sqrt{2}(-1)^{q+1} Im(Y_l^q), & \text{if } 0 < q \leqslant l, \end{cases}$$

for $p = 1, \cdots, n$, where $Re(Y_l^q)$ and $Im(Y_l^q)$ represent the real and imaginary parts of Y_l^q respectively. Thus, the HARDI signal S can be described as

$$S(\theta,\phi) = \sum_{p=1}^{n} c_p R_p(\theta,\phi).$$

Moreover, Descoteaux *et al.* [17] showed that the ODF can be expressed as

$$\Psi(\theta,\phi) = \sum_{p=1}^{n} \underbrace{2\pi P_{l(p)}(0) c_p}_{c'_p} R_p(\theta,\phi),$$

where $P_{l(p)}(0)$ is a Legendre polynomial with simple expression

$$P_{l(p)}(0) = (-1)^{\frac{l(p)}{2}} \frac{3 \cdot 5 \cdots (l(p)-1)}{2 \cdot 4 \cdots l(p)}.$$

This demonstrated that the ODF can be estimated by scaling of the HARDI signal's spherical harmonic coefficients.

It is noticed that, as a probability distribution function, the ODF should be nonnegative over its entire domain. However, as pointed out in [47], this constraint has been more challenging to satisfy within the spherical harmonics framework. In the following theorem, we show that there is a constant linear transformation relation between the vector versions of the HARDI signal and the ODF in the homogeneous polynomial basis. Such a linear transformation connection between the HARDI signal and the ODF makes the nonnegative ODF model possible.

Theorem 5.1. *For $p, k = 1, \cdots, n$, let*

$$t_{pk} = \int_{\Omega} x_1^i x_2^j x_3^{m-i-j} R_p(x) d\Omega.$$

Then $T = (t_{pk})$ is an $n \times n$ invertible matrix. Let D be an $n \times n$ diagonal matrix with its diagonal elements as $P_{l(1)}(0), \cdots, P_{l(n)}(0)$. Let $A = 2\pi T^{-1} D T$. Then we have $u = As$.

Proof. We note that both the mth-order tensor polynomials restricted to the sphere and the even order spherical harmonics up to order m, are bases for the same function space. So, for the vector version s of a HARDI signal S, there exists a vector c of spherical harmonic coefficients such that $c = Ts$ (see [16] for details). Thus, T is invertible. By [17], we have

$$\Psi(x) = \sum_{p=1}^{n} 2\pi P_{l(p)}(0) c_p R_p(x),$$

where $c_p, p = 1, \cdots, n$ are the spherical harmonics series coefficients of $S(x)$. Hence, the spherical harmonics series coefficients of $\Psi(x)$ are $\psi_p = 2\pi P_{l(p)}(0) c_p$. Let c and ψ

are vectors in $\Re^n$ with components c_p and ψ_p for $p = 1, \cdots, n$, respectively. Then we have $c = Ts$ and $\psi = Tu$. Thus, $u = 2\pi T^{-1}DTs = As$. The proof is complete. $\square$

Suppose that HARDI samples in N gradient directions $\{\mathbf{g}_h : \mathbf{g}_h \in \Omega, h = 1, \cdots, N\}$, $N >> n$, and the corresponding HARDI signals on these N gradients are $\{d_h : h = 1, \cdots, N\}$. Then $\{\hat{\mathbf{g}}_h = H(\mathbf{g}_h) : h = 1, \cdots, N\}$ are N vectors in $\Re^n$. We assume that $\{\hat{\mathbf{g}}_h : h = 1, \cdots, N\}$ spans $\Re^n$, i.e., there are n vectors among these N vectors, which are linearly independent, or we say that $\{\hat{\mathbf{g}}_h : h = 1, \cdots, N\}$ has rank n. We call this assumption the **full rank assumption**. This assumption is necessary such that the N gradient directions $\{\mathbf{g}_h : h = 1, \cdots, N\}$ can reflect the signal $S(\mathbf{g})$ sufficiently. Let C be an $n \times N$ matrix, whose column vectors are $\hat{\mathbf{g}}_h, h = 1, \cdots, N$. Let $B = CC^T$. Then B is an $n \times n$ positive semi-definite symmetric matrix. Under the full rank assumption, B is a positive definite symmetric matrix. We also let d be a vector in $\Re^N$, with components $\{d_h : h = 1, \cdots, N\}$.

The least squares problem for finding an mth order tensor $\mathcal{A}$ to reflect the signal $S(\mathbf{g})$ is to find $\bar{s} \in \Re^n$ such that

$$F(\bar{s}) = \min_{s \in \Re^n} F(s), \tag{5.3}$$

where

$$F(s) = \sum_{h=1}^{N} (S(\mathbf{g}_h) - d_h)^2 = \sum_{h=1}^{N} \left(s^T\hat{\mathbf{g}}_h - d_h\right)^2.$$

It is well-known that under the full rank assumption the solution of the least squares problem (5.3) is

$$\bar{s} = B^{-1}Cd. \tag{5.4}$$

The function F is a convex quadratic function. Actually, by (5.4), for any $s \in \Re^n$, we have

$$F(s) = (s - \bar{s})^T B(s - \bar{s}). \tag{5.5}$$

With $u = As$, given by theorem 5.1, we have ODF $\Psi(x) = u^T\hat{x}$. However, in this way, we cannot guarantee $\Psi(x) \geqslant 0$ for all $x \in \Omega$. In fact, $\Psi(x) \geqslant 0$ for all x if and only if u is positive semi-definite in the sense of definition 5.1. Thus, based on the results of [58], we formulate a new model as

$$F(s^*) = \min\{F(s) : \lambda_{\min}(As) \geqslant 0\}. \tag{5.6}$$

Here, $\lambda_{\min}(u)$ is smallest Z-eigenvalue of u

$$\lambda_{\min}(u) = \min\{\Psi(x) : x_1^2 + x_2^2 + x_3^2 = 1\}. \tag{5.7}$$

Its solution method was given in [58]. For completeness, we give it in section 5.4.

We now have the following theorem:

Theorem 5.2. *$\lambda_{\min}(As)$ is a continuous concave function of s. Hence, (5.6) is a convex optimization problem.*

If $\lambda_{\min}(A\bar{s}) \geqslant 0$, then $s^* = \bar{s}$ is a global minimizer of (5.6). If the full rank assumption holds, then (5.6) has a unique global minimizer.

Suppose that the full rank assumption holds and $\lambda_{\min}(A\bar{s}) < 0$. Then , s^* is the unique global minimizer of (5.6) if and only if there is a positive number μ such that

$$\left\{ \begin{array}{lcl} B(s^* - \bar{s}) & = & \mu A^T \hat{x^*}, \\ \lambda_{\min}(As^*) & = & 0, \end{array} \right. \tag{5.8}$$

$$\left\{ \begin{array}{lcl} (s^*)^T B(s^* - \bar{s}) & = & 0, \\ (\hat{x^*})^T As^* & = & 0, \end{array} \right. \tag{5.9}$$

where $A^T\hat{x^*}$ is a subgradient [64] of the concave function $\lambda_{\min}$ at s^*.

Proof. Let $s^{(1)}, s^{(2)} \in \Re^n, 0 \leqslant t \leqslant 1$ and $s = ts^{(1)} + (1-t)s^{(2)}$. Suppose x^* is a global minimizer of (5.7). Then $(x_1^*)^2 + (x_2^*)^2 + (x_3^*)^2 = 1$ and

$$\begin{array}{rcl} \lambda_{\min}(As) & = & \Psi(x^*) \\ & = & t(\hat{x^*})^T As^{(1)} + (1-t)(\hat{x^*})^T As^{(2)} \\ & \geqslant & t\lambda_{\min}(As^{(1)}) + (1-t)\lambda_{\min}(As^{(2)}). \end{array}$$

This shows that $\lambda_{\min}(As)$ is a concave function of s. Since $\lambda_{\min}(As)$ is a concave function defined in the whole space $\Re^n$, according to convex analysis [64], it is a continuous function. By (5.5), F is a convex quadratic function of s. Hence, (5.7) is a convex optimization problem.

If $\lambda_{\min}(A\bar{s}) \geqslant 0$, then $\bar{s}$ satisfies the constraint of (5.6). Since $\nabla F(\bar{s}) = 0$, $s^* = \bar{s}$ is a global minimizer of (5.6). If the full rank assumption holds, then F is strictly convex. Hence, (5.6) has a unique global minimizer in this case.

Suppose that the full rank assumption holds and $\lambda_{\min}(A\bar{s}) < 0$. Then, (5.6), has a unique global minimizer s^*, $\lambda_{\min}(As^*) \geqslant 0$ and $F(s^*) > F(\bar{s})$. Suppose that $\lambda_{\min}(As^*) > 0$. Since $\lambda_{\min}(As)$ is continuous, in the segment connecting s^* and $\bar{s}$, there is $\tilde{s}$ such that $\lambda_{\min}(A\tilde{s}) = 0$ and $F(\tilde{s}) < F(s^*)$, which contradicts that s^* is a global minimizer of (5.6). Hence, we have $\lambda_{\min}(As^*) = 0$. Now, (5.8) follows from (5.5) and the optimality condition of the convex optimization problem (5.6). By (5.7), we have

$$\lambda_{min}(As^*) = \Psi(x^*) = (s^*)^T A^T \hat{x^*}.$$

From this and the second equation of (5.8), we have the second equation of (5.9). Let the two sides of the first equation of (5.8) take inner product with s^*. Combining with the second equation of (5.9), we have the first equation of (5.9). □

Suppose that x is a global minimizer of (5.7). Since $\Psi(x) = u^T \hat{x}$, we have

$$\lambda_{min}(As) = \hat{x}^T As. \tag{5.10}$$

When m is even, if x is a global minimizer of (5.7), then $y = -x$ is also a global minimizer of (5.7). However, we have $\hat{y} = \hat{x}$ in this case. Therefore, such $\hat{x}$ in (5.10), generated by a global minimizer x, may still be unique even if the global minimizers are not unique. By convex analysis, we know that if such $\hat{x}$ in (5.10) is unique, then $\lambda_{\min}(As)$ is differentiable at s and its gradient is $A^T\hat{x}$. If such $\hat{x}$ is not unique, then any of such $A^T\hat{x}$ is a subgradient of $\lambda_{\min}(As)$ at s and the subdifferential of

$\lambda_{\min}(As)$ at s is the convex hull of all such $A^T\hat{x}$. With such knowledge of gradients and subgradients of $\lambda_{\min}(As)$, we may solve convex optimization problem (5.6) by a standard convex programming method [32].

Under the full rank assumption, we may use (5.4) to calculate $\bar{s}$. If $\lambda_{\min}(A\bar{s}) \geqslant 0$, then $s^* = \bar{s}$ and we have the solution needed. If $\lambda_{\min}(A\bar{s}) < 0$, by Theorem 2, $\lambda_{\min}(As^*) = 0$. Hence, in this case, we only need to solve an equality constrained optimization problem.

$$F(s^*) = \min\{F(s) : \lambda_{\min}(As) = 0\}. \tag{5.11}$$

Then we have $u = As^*$ and the nonnegative ODF $\Psi(x) = u^T\hat{x}$.

Clearly, problem (5.11) still has (5.8) as its optimality condition under the full rank assumption. We may consider to solve this equality constrained problem by some standard minimization methods [44].

Let $\mathcal{P}^+$ denote a projection operator which may project a tensor onto the positive semi-definite cone. A projected gradient descent algorithm for solving nonnegative ODF model can be presented as follows.

Algorithm 5.1 (PSD-ODF). *S1: Given constant $\varepsilon > 0$. Calculate $\bar{s}$ by (5.4) and then get an initial guess $\bar{u} = A\bar{s}$. If $\lambda_{\min}(u) \geqslant 0$, Stop; Otherwise set $u_1 = \mathcal{P}^+(u)$.*

S2: Set $d_1 = -A^{-T}BA^{-1}(u_1 - \bar{u})$ and $k = 1$.

S3: If $|u_k^T A^{-T}BA^{-1}(u_k - \bar{u})| > \varepsilon$ or $\lambda_{\min}(u_k) > \varepsilon$ does not hold, stop.

S4: Compute t_k such that $u_{k+1} := u_k + t_k d_k$ satisfies $u_{k+1}^T A^{-T}BA^{-1}(u_{k+1} - \bar{u}) = 0$;

S5: Compute $\lambda_{\min}(u_{k+1})$. if $\lambda_{\min}(u_{k+1}) < 0$, then set $u_{k+1} = \mathcal{P}^+(u_{k+1})$.

S6: Compute $d_{k+1} = -A^{-T}BA^{-1}(u_{k+1} - \bar{u})$ *and set* $k = k+1$*, goto S3.*

One of the advantages of homogeneous polynomial basis over spherical harmonic basis is that the local maxima of the ODF can be easily computed. In the next section, we present formulas for determining the principal directions of the ODF.

5.2 Principal Directions of ODF

Recently, using the Z-eigenvalue concept introduced in [53], Bloy and Verma [8] proposed to determine the principal directions (maxima) of the ODF by their curvatures. In this section, we use optimization theory to determine the principal directions (maxima) of the ODF.

After a nonnegative ODF u is found, we may solve

$$\max\{\Psi(x) : x_1^2 + x_2^2 + x_3^2 = 1\}. \tag{5.12}$$

The optimization conditions of (5.12) are (5.15), the same as the optimality conditions of (5.7). Actually, in section 5.4, we give the method to calculate all the Z-eigenvalues [53] of u. The largest Z-eigenvalue $\lambda_{\max}$ gives the global maximum value of u. Then, the corresponding solution x, called as the Z-eigenvector of u, associated with $\lambda_{\max}$, is the leading principal direction of u. The Z-eigenvectors of u, associated with the other Z-eigenvalues, include local maximizers, local minimizers and saddle points. Suppose that (x, λ) is a solution of (5.15). Then λ is a Z-eigenvalue of u, x is a stationary point of (5.12).

The Hessian of the Lagrangian function of (5.15) at (x, λ)

$$\nabla_x^2 L(x, \lambda) = R - m(m-1)\lambda I,$$

Where I is the 3×3 unit matrix and $R = (r_{ij})$ is a 3×3 symmetric matrix. Denotes

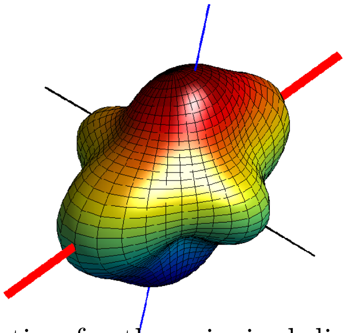

Figure 5.1: Illustration for the principal directions of ODF

$m(i,j) = m - i - j$, then

$$
\begin{aligned}
r_{11} &= \sum_{i=2}^{m}\sum_{j=0}^{m-i} i(i-1)E(\mathcal{B})_{i+1,j+1}x_1^{i-2}x_2^{j}x_3^{m(i,j)},\\
r_{12} &= \sum_{i=1}^{m}\sum_{j=1}^{m-i} ijE(\mathcal{B})_{i+1,j+1}x_1^{i-1}x_2^{j-1}x_3^{m(i,j)},\\
r_{13} &= \sum_{i=1}^{m}\sum_{j=0}^{m-i-1} im(i,j)E(\mathcal{B})_{i+1,j+1}x_1^{i-1}x_2^{j}x_3^{m(i,j)-1},\\
r_{22} &= \sum_{i=0}^{m}\sum_{j=2}^{m-i} j(j-1)E(\mathcal{B})_{i+1,j+1}x_1^{i}x_2^{j-2}x_3^{m(i,j)},\\
r_{23} &= \sum_{i=0}^{m}\sum_{j=1}^{m-i-1} jm(i,j)E(\mathcal{B})_{i+1,j+1}x_1^{i}x_2^{j-1}x_3^{m(i,j)-1},\\
r_{33} &= \sum_{i=0}^{m}\sum_{j=0}^{m-i-2} m(i,j)(m(i,j)-1)E(\mathcal{B})_{i+1,j+1}x_1^{i}x_2^{j}x_3^{m(i,j)-2}.
\end{aligned}
\tag{5.13}
$$

Clearly, x is an eigenvector of R with eigenvalue $m(m-1)\lambda$. Then, by optimization theory and [56], if the other two eigenvalues of R are less than $m(m-1)\lambda$, then x is a local maximizer of (5.15). On the other hand, if x is a local maximizer of (5.15), then the other two eigenvalues of R are less than or equal to $m(m-1)\lambda$.

In figure 5.1, the red line denotes the maximum principal direction, the blue one is the second principal direction, and the black one is the third principal direction.

Suppose that x is a local maximizer of (5.12) with Z-eigenvalue λ. If λ is less than or equal to one of the other Z-eigenvalues which are corresponding to saddle points or local minimizers, then such a direction is not significant. We thus call a Z-eigenvalue of u a principal Z-eigenvalue, if it is greater than all the other eigenvalues which are corresponding to saddle points or local minimizers. Then, a Z-eigenvector of u, associated with a principal Z-eigenvalue, is regarded as a principal direction. In figure 5.1, we give an illustration for the principal direction of ODF.

5.3 Experimental Results

In this section, we report some experimental results on our method applied to simulated dataset as well as real human brain dataset. Firstly, we generate synthetic HARDI data by the following multilinear model [1]:

$$S(g_i) = \sum_{k=1}^{n} p_k e^{-bg_i D_k g_i} + \text{noise}, \tag{5.14}$$

where $n \in \{0, 1, 2, 3\}$ is the number of fibers, p_k is the proportion of tissue in the voxel that corresponds to the k^{th} fiber ($\sum_{k=1}^{f} p_k = 1$), b is the b-value, g_i is the i^{th} gradient direction for $i \in \{1, \cdots, 81\}$, and D_k is the diffusion tensor of the k^{th} fiber. This synthetic data generation is relatively standard and has advantage of analytic computation of the ODF [17]. The noise was typically generated by Rician noise (complex Gaussion noise) with standard deviation of $1/\sigma$, producing a signal to noise ratio (SNR) of σ. In our experiments, unless special instructions, the b value equals to 3000 sec/mm^2 and the diffusion tensors were selected such as $D_k = diag(1700, 200, 200) \times 10^{-6}$ mm^2/sec for $k = 1, 2, 3$. And we generated Rician-corrupted data S as done in [18]. For each noise-free data x, we computed S

Table 5.1: Z-eigenvalues and eigenvectors of a 4th order tensor, where the ODF was estimated by LS method

	x_1	x_2	x_3	λ
1	-0.0055	0.0002	1.0000	-0.7344
2	0.0048	1.0000	0.0024	-0.6048
3	-0.1634	0.7117	0.6832	0.0878
4	0.1559	0.7119	0.6847	0.0906
5	0.0137	0.7211	0.6927	0.0941
6	-0.1689	-0.7120	0.6816	0.0945
7	0.1577	-0.7135	0.6826	0.0985
8	0.0174	-0.7225	0.6911	0.1020

as:

$$S = \sqrt{\left(\frac{x}{\sqrt{2}} + n_r\right)^2 + \left(\frac{x}{\sqrt{2}} + n_i\right)^2}$$

where n_r and $n_i \sim \mathcal{N}(0, \sigma^2)$. The value S is the realisation of a random variable with a Rician p.d.f. of parameters x and σ.

First, we demonstrate qualitatively that our method can guarantee nonnegative diffusivity by comparing it with the Least Squares (LS) method. The LS method is a simple approach to estimate the coefficients of an ODF function, which is fast but does not guarantee positive diffusivity. We estimate the ODF fitting with 4th order diffusion tensor in the homogeneous polynomial basis using these two algorithms. For the single tensor model, the ODF function (the SNR was fixed to 35) estimated by the LS method, fitting with a 4th order tensor, is $ODF(x) = u^T\hat{x}$, where u is a 15-dim vector with $u(1) = -0.7344$, $u(2) = -0.0010$, $u(3) = 1.7255$, $u(4) = -0.0140$, $u(5) = -0.6048$, $u(6) = 0.0142$, $u(7) = 0.0055$, $u(8) = 0.0203$, $u(9) = -0.0096$, $u(10) = -0.1701$, $u(11) = 0.0035$, $u(12) = -0.2132$, $u(13) = -0.0650$, $u(14) = -0.0035$, $u(15) = 7.1234$. Using the method provided in Appendix, we can compute all the Z-eigenvalues and the associated eigenvectors, which are listed in table 5.1.

From table 5.1, we can see that there are two negative eigenvalues and the smallest

Table 5.2: Z-eigenvalues and eigenvectors of a 4th order tensor, where the ODF was estimated by PSD method

	g_1	g_2	g_3	λ
1	-0.0023	0.0001	1.0000	0.0001
2	0.0017	1.0000	0.0015	0.1297
3	-0.0100	0.7160	0.6980	1.2786
4	-0.0135	-0.7168	0.6971	1.2862

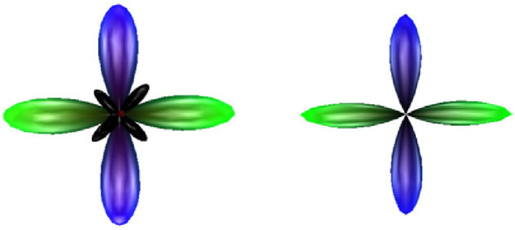

Figure 5.2: (borrowed from [47] to illustrate effects of the approach with guaranteed nonnegative diffusivity.) Without projection (left), the final ODF estimation contains negative-valued regions (the side lobes in black). Meanwhile, enabling projection results in elimination of these negative side lobes (right).

Z-eigenvalue is -0.7344, attained at $(-0.0055, 0.0002, 1.0000)$. This demonstrates the need for enforcing the positive semi-definite property of the estimated tensor since negative diffusivity profiles are not meaningful from the point of view of physics.

But our method can guarantee positive diffusivity. In the same case, the ODF function estimated by the our method is $ODF(x) = u^T\hat{x}$, with $u(1) = 0.0001$, $u(2) = -0.001$, $u(3) = 4.9965$, $u(4) = -0.014$, $u(5) = 0.1297$, $u(6) = 0.0142$, $u(7) = 0.0055$, $u(8) = 0.0203$, $u(9) = -0.0096$, $u(10) = 3.1009$, $u(11) = 0.0035$, $u(12) = 3.0578$, $u(13) = -0.065$, $u(14) = -0.0035$, $u(15) = 7.8579$. We compute all Z-eigenvalues and the associated eigenvectors and list them in Table 2. We can see that the smallest Z-eigenvalue is 0.0001, attained at $(-0.0023, 0.0001, 1.0000)$.

We also borrowed a picture from [47] to illustrate effects of the approach with guaranteed nonnegative diffusivity. If a final ODF estimation contains negative diffusivity, it will result in negative side lobes as indicated in black in figure 5.2. Mean-

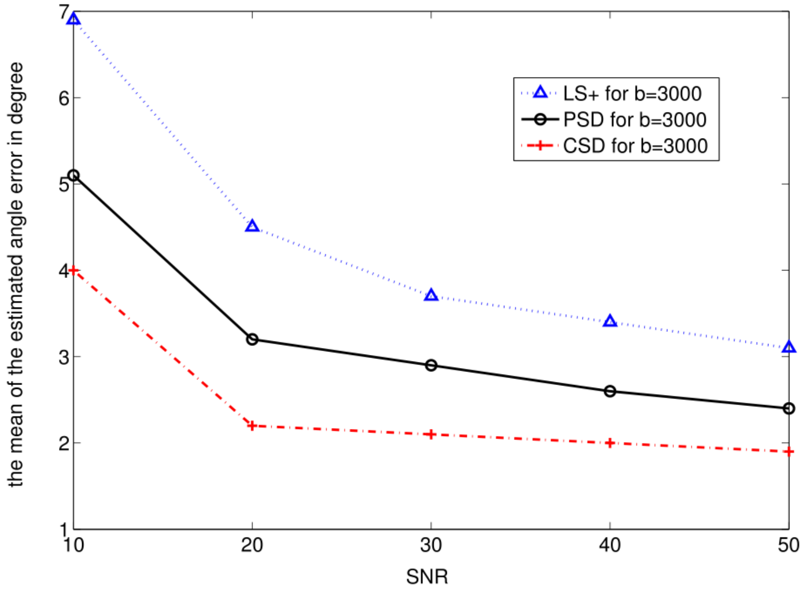

Figure 5.3: Effects of varying SNR on the detected maximum of the ODF estimations by LS+ method, CSD method and our positive semi-definite (PSD) approach for $b = 3000$.

while, the approach with guaranteed nonnegative diffusivity can eliminate these side lobes completely.

In figure 5.4, the red line is for $b = 1000$, while black is for $b = 3000$. The y-axis is the mean of the estimated angular error.

Next, in order to compare the robustness of our method in the presence of noise, we generated the signals by (5.14) at 5 different SNR ranging from 10 to 50 and repeated the experiments 10 times. We estimate the ODF fitting with 6th order diffusion tensor. We choose to compare our PSD method against the following two methods: (1) a nonnegative constrained Least Squares (LS+) method: solve (5.4) and by using the linear transformation to get $\bar{u} = A\bar{s}$, then set negative ODF as zero, i.e. project it onto the nonnegative space, finally get the solution $u = \mathcal{P}^+(\bar{u})$; (2) and a constrained spherical deconvolution (CSD) method [71]. For the CSD method, a constraint is introduced to minimize the appearance of negative values in the reconstructed FOD. But it does not completely forbid negative FOD values.

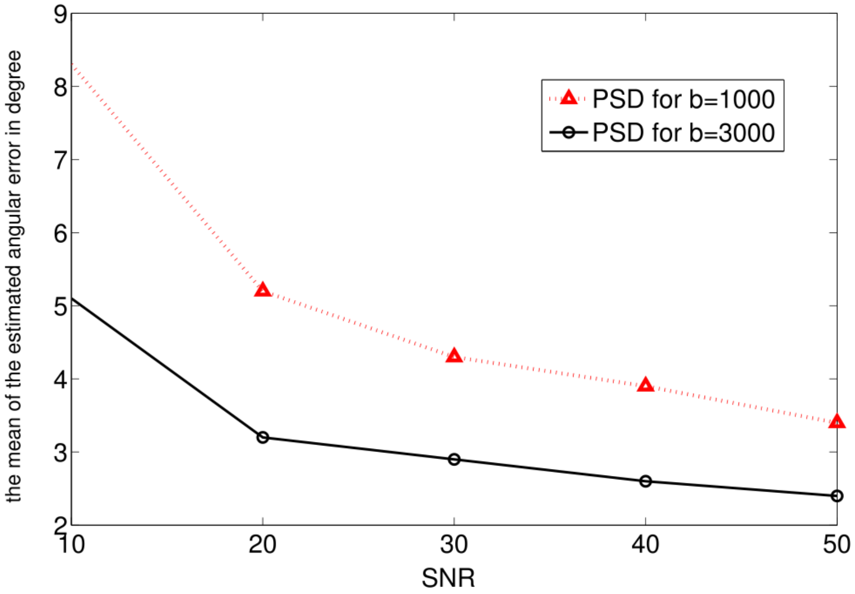

Figure 5.4: Effects of varying SNR on the detected maximum of the ODF estimations by PSD method for different b-values.

We implemented the CSD method with $\lambda = 1$ and $\tau = 10\%$ of the mean initial FOD amplitude (see [71] for a detailed description of these parameters). Then, we computed the means of angles errors in degree between the actual fiber orientations and the maxima of estimated ODF / FOD. The results are plotted in figure 5.3. As would be expected the means of the degree error decreases as the SNR increases. The PSD method compares favorably to the least squares (LS+) method. Comparing with the PSD method, the CSD method gains an improvement of approximately 1°. Figure 5.4 also shows the precision of PSD method in the presence of varying levels of noise, for different b-values. The red line is for $b = 1000$, while black is for $b = 3000$. As we can see from figure 5.4, when $SNR = 50$, the angular error is about 3.3° at $b = 1000$ while it is about 2.4° at $b = 3000$.

Next we worked on a phantom dataset [49], which was acquired on a GE Healcare Signa 1.5T scanner. It had 4000 gradient directions and for our experiments we used a b-value of $4000 s/mm^2$. The phantom had a geometry of two fiber bundles

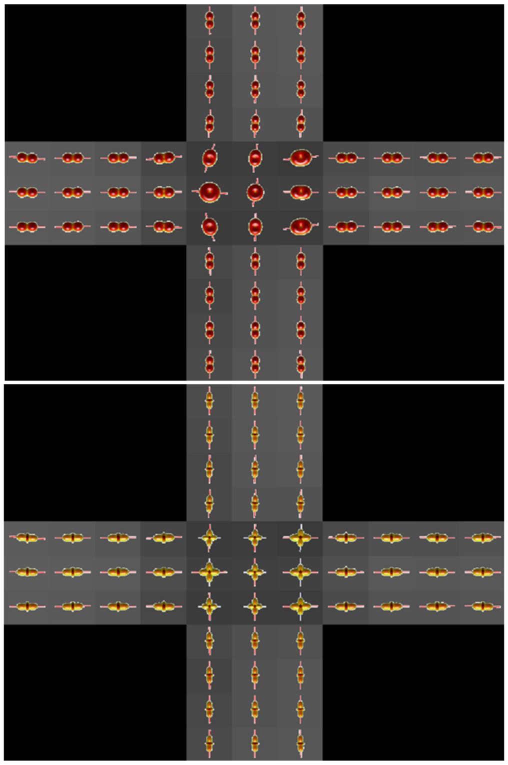

Figure 5.5: ODF profile from a Phantom dataset.

crossing perpendicularly close to X-axis and the Y-axis. Figure 5.5 shows the ODF profiles. We estimated the 2nd order diffusion tensor (shown in the up) and the 4th order diffusion tensor (shown in the down) using our method. The extracted principle direction of ODF indicate the known fiber geometry of the phantom. These results also demonstrate that high order tensor estimation is necessary since 2nd order diffusion tensor fails to approximate complex local tissue structure.

Figure 5.6 is shown on the Generalized Fractional Anisotropy (GFA) map, where GFA was defined by Tuch [72] as $GFA = \frac{std(ODF)}{rms(ODF)}$. We also reconstructed ODFs with principal directions for a region of interest, where contains crossing fiber bundles.

In the next experiment, we are interested to estimate the ODF profiles from human brain dataset with size of $90 \times 90 \times 60$, which was acquired on a 1.5T scanner at $b = 1000s/mm^2$ using 60 encoding directions, with voxel dimensions of $1.875mm \times$

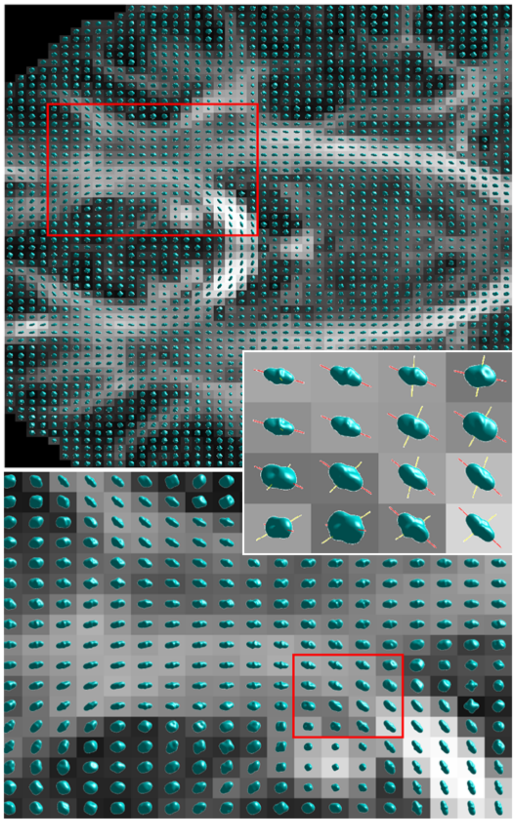

Figure 5.6: Reconstructed ODFs from human brain with sharpening

$1.875mm \times 2mm$. We show some reconstructed ODFs with sharpening in figure 5.6, in which the parameter $\alpha = 0.006$. We can detect there are multiple fibers in which some crossing are due to diverging or splitting fibers. We also show some reconstructed ODFs with principal directions in the region of interest (ROI). These results show that our nonnegative ODF profiles model can depict the characterization of diffusion anisotropy which was consistent with known neuroanatomy.

5.4 Appendix: The Solution Method for (5.7)

According to optimization theory, the optimality conditions of (5.7) have the form:

$$\left\{\begin{array}{lcl} \sum_{i=1}^{m}\sum_{j=0}^{m-i} ib_{i,j}x_1^{i-1}x_2^jx_3^{m(i,j)} & = & m\lambda x_1, \\ \sum_{i=0}^{m}\sum_{j=1}^{m-i} jb_{i,j}x_1^{i}x_2^{j-1}x_3^{m(i,j)} & = & m\lambda x_2, \\ \sum_{i=0}^{m}\sum_{j=0}^{m-i-1} m(i,j)b_{i,j}x_1^{i}x_2^{j}x_3^{m(i,j)-1} & = & m\lambda x_3, \\ x_1^2+x_2^2+x_3^2=1. & & \end{array}\right. , \tag{5.15}$$

where $b_{i,j} = E(\mathcal{B})_{i+1,j+1}$. The additional "$m$" on the right hand sides of the first three equations make it the same as the definition of Z-eigenvalues [53, 55, 56, 57, 8] for the symmetric tensor x. If (x, λ) is a solution of (5.15), then x is a stationary point of (5.7) and

$$\lambda = \Psi(x) \tag{5.16}$$

is a Z-eigenvalue of u. Then, the smallest Z-eigenvalue of u is the optimal value of (5.7).

We may solve (5.15) in the following way:

Case 1: $x_3 = x_2 = 0$. By (5.15), this only happens if $b_{m-1,1} = b_{m-1,0} = 0$. In this case, $x_1 = \pm 1$, $\lambda = b_{m,0}$.

Case 2: $x_3 = x_1 = 0$. By (5.15), this only happens if $b_{1,m-1} = b_{0,m-1} = 0$. In this case, $x_2 = \pm 1$, $\lambda = b_{0,m}$.

Case 3: $x_3 = 0$, $x_1 \neq 0$ and $x_2 \neq 0$. Then (5.15) becomes

$$\begin{cases} \sum_{i=1}^{m} i b_{i,m-i} x_1^{i-1} x_2^{m-i} & = & m\lambda x_1, \\ \sum_{i=0}^{m-1} (m-i) b_{i,m-i} x_1^{i} x_2^{m-i-1} & = & m\lambda x_2, \\ \sum_{i=0}^{m-1} b_{i,m-i-1} x_1^{i} x_2^{m-i-1} & = & 0, \\ x_1^2 + x_2^2 = 1. \end{cases} \tag{5.17}$$

We may eliminate λ in (5.17) and have the following equations of x_1 and x_2:

$$\begin{cases} \sum_{i=1}^{m} i b_{i,m-i} x_1^{i-1} x_2^{m-i+1} & = & \sum_{i=0}^{m-1} (m-i) b_{i,m-i} x_1^{i+1} x_2^{m-i-1}, \\ \sum_{i=0}^{m-1} b_{i,m-i-1} x_1^{i} x_2^{m-i-1} & = & 0, \\ x_1^2 + x_2^2 = 1. \end{cases}$$

Let $t = \frac{x_1}{x_2}$. We have

$$\begin{cases} \sum_{i=1}^{m} i b_{i,m-i} t^{i-1} & = & \sum_{i=0}^{m-1} (m-i) b_{i,m-i} t^{i+1}, \\ \sum_{i=0}^{m-1} b_{i,m-i-1} t^{i} & = & 0. \end{cases} \tag{5.18}$$

We may solve the two one-variable equations of (5.18) separately. If they have common solutions t, then (5.15) has solutions

$$x_1 = \frac{t}{\sqrt{1+t^2}}, \quad x_2 = \frac{\pm 1}{\sqrt{1+t^2}}, \quad x_3 = 0, \quad \lambda = \Psi(x).$$

Case 4: $x_3 \neq 0$. We may eliminate λ in (5.15) and have the following equations of x:

$$\left\{\begin{array}{lcl}\sum\limits_{i=1}^{m}\sum\limits_{j=0}^{m-i} ib_{ij}x_1^{i-1}x_2^{j}x_3^{m(i,j)+1} & = & \sum\limits_{i=0}^{m}\sum\limits_{j=0}^{m-i-1} m(i,j)b_{ij}x_1^{i+1}x_2^{j}x_3^{m(i,j)-1}, \\ \sum\limits_{i=0}^{m}\sum\limits_{j=1}^{m-i} jb_{ij}x_1^{i}x_2^{j-1}x_3^{m(i,j)+1} & = & \sum\limits_{i=0}^{m}\sum\limits_{j=0}^{m-i-1} m(i,j)b_{ij}x_1^{i}x_2^{j+1}x_3^{m(i,j)-1}, \\ x_1^2+x_2^2+x_3^2=1. & & \end{array}\right. \tag{5.19}$$

Let $w=\frac{x_1}{x_3}$, $v=\frac{x_2}{x_3}$. Then we have

$$\left\{\begin{array}{lcl}\sum\limits_{i=1}^{m}\sum\limits_{j=0}^{m-i} ib_{ij}w^{i-1}v^{j} & = & \sum\limits_{i=0}^{m}\sum\limits_{j=0}^{m-i-1} m(i,j)b_{ij}w^{i+1}v^{j}, \\ \sum\limits_{i=0}^{m}\sum\limits_{j=1}^{m-i} jb_{ij}w^{i}v^{j-1} & = & \sum\limits_{i=0}^{m}\sum\limits_{j=0}^{m-i-1} m(i,j)b_{ij}w^{i}v^{j+1}. \end{array}\right. \tag{5.20}$$

For solving system (5.20), we first regard it as a system of polynomial equations of variable w and rewrite it as

$$\left\{\begin{array}{l}\gamma_0 w^m+\gamma_1 w^{m-1}+\cdots+\gamma_m=0, \\ \tau_0 w^{m-1}+\tau_1 w^{m-2}+\cdots+\tau_{m-1}=0,\end{array}\right.$$

where $\gamma_0,\cdots,\gamma_m,\ \tau_0,\cdots,\tau_{m-1}$ are polynomials of v, which can be calculated by (5.16). By the Sylvester theorem, the above system of polynomial equations in w possesses solutions if and only if its resultant vanishes [15]. The resultant of this system of polynomial equations is the determinant of the following $(2m-1)\times(2m-1)$ matrix

$$V:=\begin{pmatrix} \gamma_0 & \gamma_1 & \cdots & \gamma_{m-2} & \gamma_{m-1} & \gamma_m & \cdots & 0 & 0 \\ 0 & \gamma_0 & \cdots & \gamma_{m-3} & \gamma_{m-2} & \gamma_{m-1} & \cdots & 0 & 0 \\ \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & \cdots & \gamma_1 & \gamma_2 & \gamma_3 & \cdots & \gamma_m & 0 \\ 0 & 0 & \cdots & \gamma_0 & \gamma_1 & \gamma_2 & \cdots & \gamma_{m-1} & \gamma_m \\ \tau_0 & \tau_1 & \cdots & \tau_{m-2} & \tau_{m-1} & 0 & \cdots & 0 & 0 \\ 0 & \tau_0 & \cdots & \tau_{m-3} & \tau_{m-2} & \tau_{m-1} & \cdots & 0 & 0 \\ \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & \cdots & \tau_0 & \tau_1 & \tau_2 & \cdots & \tau_{m-1} & 0 \\ 0 & 0 & \cdots & 0 & \tau_0 & \tau_1 & \cdots & \tau_{m-2} & \tau_{m-1} \end{pmatrix},$$

which is a polynomial equation in variable v. After finding all real roots of this polynomial, we can substitute them to (5.20) to find all the real solutions of w. Then, using $x_1 = \frac{w}{\sqrt{1+w^2+v^2}}$, $x_2 = \frac{v}{\sqrt{1+w^2+v^2}}$, $x_3 = \frac{\pm 1}{\sqrt{1+w^2+v^2}}$, $\lambda = \Psi(x)$, we may find all the solutions of (5.15) in this case.

Combine all the possible solutions of (5.15) in these four cases, and find $\lambda_{\min}(As)$, the smallest value of λ of these solutions. This solves (5.7).

Chapter 6

Conclusions and future work

This chapter draws conclusions on the thesis, and points out some possible research directions related to the work done in this thesis.

6.1 Conclusions

There are a lot of applications with the positive semi-definite space tensor. Whether in theory or in real life, the study of the positive semi-definite space tensor is very useful. In this thesis, we focus on this subject.

How to verify the positive semi-definiteness of a symmetric space tensor is the first thing we are interested. Based on the theory of the nonnegative polynomial, we construct two methods to verify the positive semi-definiteness of a symmetric space tensor in chapter 3. Both methods have their advantages and disadvantages. When the order of the tensor $\mathcal{A}$ is small, we prefer to use the first method, in this case we just need to solve a linear sdp program. But when the order of the tensor $\mathcal{A}$ is large, we prefer to use the second method. Although solving (3.6) is difficult, the sign of the value of (3.6) is our concern. Furthermore, from the numerical results of this chapter, it is easy to see that we almost get the right results by the als method. Based on the relationship of the positive definiteness of tensors and their smallest H-eigenvalue, we also discuss the properties of the smallest H-eigenvalue of tensors from the optimization view, which gives us a new perspective to analyze the smallest H-eigenvalue of tensors.

In chapter 4, we study the algorithms for solving (1.5), which is a program with a positive semi-definite space tensor cone constraint. This program is like the sdp program, but it has its special problems. We try to use the polyhedral outer approximation method to solve (1.5) in section 4.1, which is a grid method. We approximate the positive semi-definite space tensor cone constraint by a series of grids. In theory, we get the error bound between the solution of polyhedral outer approximation method and the real solution. From the view of the numerical tests in section 4.3, the polyhedral outer approximation method works well as our expected. Furthermore, we discuss some other algorithms for solving (1.5) or (1.4) in section 4.2, including the conic linear program approach, the nonsmooth convex program approach and the bi-level program approach.

In chapter 5, we show that there is a constant linear transformation relation between the vector versions of the raw HARDI signal and the ODF in the homogeneous polynomial basis firstly. Such a linear transformation connection makes the nonnegative ODF model possible. Then we propose a new reconstruction framework to estimate nonnegative ODFs from HARDI data. Features of this model includes minimizing a convex optimization problem with a convex quadratic objective function constrained by the nonnegativity requirement on the smallest Z-eigenvalue of the diffusivity tensor. So, it can guarantee the positive semi-definite property of the estimated high order tensor (not limited to 4th order tensor), which is the main contribution of our work. This property is essential since negative diffusivity profiles are not meaningful from the point of view of physics. Finally, based on optimization theory, we present a computational method for determining the principal directions of the ODF. Numerical examples on synthetic data as well as MRI data are displayed to demonstrate our approach.

6.2 Future Work

Firstly, we still don't find a perfect method to verify the positive semi-definiteness of a tensor. The two methods we provided are still needed to be improved. In the future, we will look for another better method to deal with this problem.

Secondly, it is easy to see that when we use the polyhedral outer approximation algorithm, we need to fine the grid more and more dense if we want to get a better approximate solution. It will bring the larger computing cost. This problem is caused by the lack of properties of the positive semi-definite tensor we can use. To solve this problem, we will explore more properties of the positive semi-definite tensors in the future.

Thirdly, we will extend the nonnegative ODF model to tractography. Further, we also would like to use this model to analyze the real datasets at $b = 3000$ (or higher b-values) where fiber crossings could be better detected.

Bibliography

[1] D.C. Alexander, G.J. Barker and S.R. Arridge, "Detection and modeling of non-Gaussion apparent diffusion coefficient profiles in human brain data", *Magnetic Resonance in Medicine*, 48 (2002), 331-340.

[2] A. Barmpoutis, M.S. Hwang, D. Howland, J.R. Forder and B.C. Vemuri, "Regularized positive-definite fourth order tensor filed estimation from DW-MRI", *Neuroimage*, 45 (2009), 5153-5162.

[3] A. Barmpoutis, B. Jian, B.C. Vemuri and T.M. Shepherd, "Symmetric Positive 4th Order Tensors & Their Estimation from Diffusion Weighted MRI", in: *Information Processing and Medical Imaging*, M. Karssemeijer and B. Lelieveldt, eds., (Springer-Verlag, Berlin, 2007), 308-319.

[4] P.J. Basser and D.K. Jones, "Diffusion-tensor MRI: theory, experimental design and data analysis - a technical review", *NMR in Biomedicine,* 15 (2002), 456-467.

[5] P.J. Basser, J. Mattiello and D. LeBihan, "Estimatation of the effective self-diffusion tensor from the NMR spin echo", *Journal of Magnetic Resonance,* B103 (1994), 247-254.

[6] P.J. Basser, J. Mattiello and D. LeBihan, "MR diffusion tensor spectroscopy and imaging", *Biophysica,* 66 (1994), 259-267.

[7] M.S. Bazaraa, C.M. Shetty. Nonlinear Programming Theory and Algorithms. New York: John Wiley and Sons, 1979.

[8] L. Bloy and R. Verma, "On computing the underlying fiber directions from the diffusion orientation distribution function", in: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008*, D. Metaxas, L. Axel, G. Fichtinger and G. Székeley, eds., (Springer-Verlag, Berlin, 2008), 1-8.

[9] K.C. Chang, K. Pearson and T. Zhang, Perron Frobenius Theorem for nonnegative tensors", Commu. Math. Sci., 6 (2008) 507-520.

[10] K.C. Chang, K. Pearson and T. Zhang, Primitivity, the convergence of the NZQ method, and the largest eigenvalue for nonnegative tensors", SIAM Journal on Matrix Analysis and Applications, 32 (2011) 806-819.

[11] S. Friedland, S. Gaubert and L. Han, Perron-Frobenius theorem for nonnegative multilinear forms and extensions", to appear in: Linear Algebra and Applications. http://www.sciencedirect.com/science/article/pii/S0024379511001765

[12] C. Chefd'Hotel, D. Tschumperle, R. Deriche and O. Faugeras, "Regularizing flows for constrained matrix-valued images", *Journal of Mathematical Imaging and Vision*, 20 (2004), 147-162.

[13] D. Cox, J. Little and D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, New York, 1998.

[14] P.H. Calamai and J.J. Moré, "Projected gradient methods for linearly constrained problems", *Math. Programming* 39 (1987), 93-116.

[15] D. Cox, J. Little and D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, New York, 1998.

[16] M. Descoteaux, E. Angelino, S. Fitzgibbons and R. Deriche, "Apparent diffusion coefficients from high angular diffusion imaging: Estimation and applications", *Magnetic Resonance in Medicine*, 56 (2006), 395-410.

[17] M. Descoteaux, E. Angelino, S. Fitzgibbons and R. Deriche, "Regularized, fast, and anaytical q-ball imaging," *Magnetic Resonance in Medicine*, 58 (2007), 497-510.

[18] M. Descoteaux, N. Wiest-Daesslé, S. Prima, C. Barillot and R. Deriche, "Impact of Rician adapted non-local means filtering on HARDI," MICCAI 2008, Part II, LNCS 5242, (2008), 122-130.

[19] J. Faraut and A. Korányi, *Analysis on Symmetric Cones*, Oxford University Press, New York, 1994.

[20] K. Fukisawa, M. Kojima, K. Nakata. SDPA (SemiDefinite Programming Algorithm) user's manualversion 6.00. Research Reports on Mathematical and Computing Sciences Series B : Operations Research Department of Mathematical and Computing Sciences Tokyo Institute of Technology 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552 Japan.

[21] C.A. Floudas, eds.. Handbook of Test Problems in Local and Global Optimization. *Nonconvex Optimization and its Applications*, 33, Kluwer Academic Publishers, Dordrecht, 1999.

[22] A. Ghosh, M. Descoteaux and R. Deriche, "Riemannian framework for estimating symmetric positive definite 4th order diffusion tensors", in: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008*, D. Metaxas, L. Axel, G. Fichtinger and G. Székeley, eds., (Springer-Verlag, Berlin, 2008), 858-865.

[23] A. Ghosh and R. Deriche, " From second to higher order tensors in Diffusion-MRI", in: *Tensors in Image Processing and Computer Vision*, Springer London 2009, 315-334.

[24] A. Ghosh, E. Tsigaridas, M. Descoteaux, P. Comon, B. Mourrain and R. Deriche, " A polynomial based approach to extract the maxima of an antipodally symmetric spherical function and its application to extract directions from the Orientation Distribution Function in Diffusion MRI", Workshop on Computational Diffusion MRI, held in conjunction with the MICCAI 2008 conference, New York, USA, September 2008.

[25] J.L. Goffin, Z.Q. Luo and Y. Ye, "Complex analysis of an interior cutting plane method for convex feasibility problems", *SIAM Journal on Optimization*, 6 (1996), 638-652.

[26] J.L. Goffin and J.P. Vial, "Convex nondifferentiable optimization: A survey focused on the analytical center cutting plane method", *Optimization Methods and Software*, 17 (2002), 805-867.

[27] G. Golub, C. van Loan. Matrix Computations. The John Hopkins University Press, Balti- more, Maryland, 3 edition, 1996.

[28] O. Güler and L. Tuncel, "Characterization of the barrier parameter of homogeneous convex cones", *Math. Prog.*, 81 (1998), 55-76.

[29] C. Helmberg, Semidefinite Programming for Combinatorial Optimization.
in http://www.zib.de/helmberg/semidef.html, 2000.

[30] C. Helmberg, F. Rendl, R.J. Vanderbei, eds. An interior-point method for semidefinite programming. *SIAM J. Optim*, 6 (1996), 342 - 361.

[31] D. Hilbert. Über die Darstellung deniter Formen als Summe von Formenquadraten. *Math. Ann*, 32(1888) 342 - 350.

[32] J-B Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Springer-Verlag, Berlin, 1993.

[33] S. Hu and L. Qi, "Algebraic connectivity of an even uniform hypergraph", Journal of Combinatorial Optimization 24 (2012) 564-579.

[34] S. Hu and L. Qi, "The Laplacian of a uniform hypergraph", to appear in: Journal of Combinatorial Optimization.

[35] T. Jacobi, A. Prestel. Distinguished representations of strictly positive polynomials. *J. Reine. Angew. Math.*, 532 (2001), 223-235.

[36] Tamara G. Kolda and Jackson R. Mayo, Shifted Power Method for Computing Tensor Eigenpairs, *SIAM. J. Matrix Anal. & Appl.* 32(4), 1095C1124

[37] J.B. Lasserre. Global optimization with polynomials and the problem of moments, *SIAM J.Optim.*, 11 (2001), 796-817.

[38] G. Li, L. Qi and G. Yu, "The Z-eigenvalues of a Symmetric Tensor and its Application to Spectral Hypergraph Theory", to appear in: Numerical Linear Algebra with Applications.

[39] M. Marshall. Positive Polynomials and Sums of Squares. Dottorato de Ricerca in Matematica, Dept. di Mat., Univ. Pisa, 2000.

[40] A. Nemirovski, "Advances in Convex Optimization: Conic Programming", in: *Proceedings of the International Congress of Mathematicians 2006*, vol. 1, M. Sanz-Sol, J. Soria, J.L. Varona and J. Verdera, eds., (EMS-European Mathematical Society Publishing House, 2007), 413-444.

[41] Yu. Nesterov and A. Nemirovski, *Interior Point Polynomial Time Methods in Convex Programming*, SIAM Philadelphia, 1994.

[42] M. Ng, L. Qi and G. Zhou, "Finding the largest eigenvalue of a non-negative tensor", *SIAM Journal on Matrix Analysis and Applications* 31 (2009), 1090-1099.

[43] J. Nie, J. Demmel, B. Sturmfels. Minimizing Polynomials via Sum of Squares over the Gradient Ideal. *Math. Program. Ser.A*, 106 (2006), 587-606.

[44] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[45] E. Ozarslan and T.H. Mareci, “Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging”, *Magnetic Resonance in Medicine*, 50 (2003), 955-965.

[46] P.A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Math. Program. Ser.B*, 96 (2003), 293-320.

[47] V. Patel, Y. Shi, P.M. Thompson and A.W. Toga, “Mesh-based spherical deconvolution: A flexible approach to reconstruction of non-negative fiber orientation distributions”, *Neuroimag*, 51 (2010), 1071-1081.

[48] K.J. Pearson, ”Essentially positive tensors”, International Journal of Algebra, 9 (2010) 421-427.

[49] C. Poupon, B. Rieul, I. Kezele, M. Perrin, F. Poupon and J.F. Mangin, “New diffusion phantoms dedicated to the study and validation of HARDI models”, *Magnetic Resonance in Medicine*, 60 (2008), 76-83.

[50] V. Powers, B. Reznick. Polynomials that are positive on an interval.*Trans. Am. math. Soc.*, 352 (2000), 4677 - 4692.

[51] A. Prestel, C.N. Delzell. Positive Polynomials: From Hilbert’s 17th Problem to Real Algebra. Springer-Verlag, New York, 2001.

[52] M. Putinar. Positive polynomials on compact semi-algebraic sets. *Indiana Univ. Math. J.*, 42 (1993), 969 - 984.

[53] L. Qi, “Eigenvalues and invariants of tensors”, *J. Symbolic Computation,* 40 (2005), 1302-1324.

[54] L. Qi, “Eigenvalues of a real supersymmetric tensor”, *J. Math. Anal. Appl.* 325 (2007), 1363C1377.

[55] L. Qi, D. Han and E.X. Wu, “Principal invariants and inherent parameters of diffusion kurtosis tensors”, *Journal of Mathematical Analysis & Applications*, 349 (2009), 165-180.

[56] L. Qi, F. Wang and Y. Wang, “Z-eigenvalue methods for a global polynomial optimization problem”, *Mathematical Programming*, 118 (2009) 301-316.

[57] L. Qi, Y. Wang and E.X. Wu, “D-Eigenvalues of diffusion kurtosis tensors”, *Journal of Computational and Applied Mathematics*, 221 (2008), 150-157.

[58] L. Qi, G. Yu and E.X. Wu, "Higher order positive semi-definite diffusion tensor imaging", *SIAM Journal on Imaging Sciences*, 3 (2010), 416-433.

[59] L. Qi, Y. Xu, Y. Yuan and X. Zhang, A Cone Constrained Convex Program: Structure and Algorithms, *Journal of the Operations Research Society of China*, 1 (1) (2013), 37-53.

[60] L. Qi and Y. Ye, "Space tensor conic programming", Department of Applied Mathematics, The Hong Kong Polytechnic University, Revised in June 2012.

[61] L. Qi, G. Yu and Y. Xu, "Nonnegative diffusion orientation distribution function", *Journal of Mathematical Imaging and Vision*, 45 (2013), 103-113.

[62] J. Renegar, "Hyperbolic programs, and their derivative relaxations", *Found. Comp. Math.*, 6 (2006), 59-79.

[63] B. Reznick. Some Concrete Aspects of Hilbert's 17th Problem. *Contemporary Mathematics AM. math. Soc*, 253 (2000), 251 - 272.

[64] R.T. Rockafellar, *Convex Analysis*, Princeton Publisher, Princeton, 1970.

[65] K. Schmüdgen. The K-moment problem for compact semi-algebraic sets. *Math. Ann.*, 289(2) (1991), 203 - 206.

[66] J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12 (1999), 625-653. Special issue on Interior Point Methods (CD supplement with software).

[67] W. Sun and Y. Yuan, *Optimization Theory and Methods*, Springer Verlag, Berlin, Germany, 2006.

[68] M.J. Todd. Semidefinite optimization. *Acta. Numer.*, 10 (2001), 515-560.

[69] K.C. Toh, R.H. Tutuncu, M.J. Todd. SDPT3 version 4.0 (beta)- a MATLAB software for semidefinite-quadratic-linear programming. [updated in 17 July, 2006]. http://www.math.nus.edu.sg/mattohkc/sdpt3.html.

[70] K.C. Toh, R.H. Tutuncu, M.J. Todd. On the implementation and usage of SDPT3 - a MATLAB software package for semidefinite-quadratic-linear programming version 4.0[OL]. [17 July, 2006]. http://www.math.nus.edu.sg/mattohkc/sdpt3.html.

[71] J-D. Tournier, F. Calamante and A. Connelly, “Robust determination of the fibre orientation distribution in diffusion MRI: Non-negativity constrained super-resolved spherical deconvolution”, *Neuroimag*, 35 (2007), 1459-1472.

[72] D.S. Tuch, “Q-ball imaging”, *Magnetic Resonance in Medicine*, 52 (2004) 1358-1372.

[73] D.S. Tuch, T.G. Reese, M.R. Wiegell, N.G. Makris, J.W. Belliveau and V.J. Wedeen, “High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity”, *Magnetic Resonance in Medicine*, 48 (2002), 454-459.

[74] R.H. Tutuncu, K.C. Toh, M.J. Todd. Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Prog.*, 95 (2003), 189-217.

[75] Z. Wang, B.C. Vemuri, Y. Chen and T.H. Mareci, “A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex DWI”, *IEEE Trans. Med. Imaging*, 23 (2004), 930-939.

[76] H. Wolkowicz, R. Saigal, L. Vandenberghe. Handbook of Semidefinite Programming, Kluwer, 2000.

[77] Y.Y. Ye, Interior Ponit Algorithm: Theory and Analysis. New York: John Wiley and Sons, 1997.

[78] Y.Y. Ye, Linear Conic Programming. Stanford University, December, 2004.

[79] L. Zhang and L. Qi, ”Linear convergence of an algorithm for computing the largest eigenvalue of a nonnegative tensor”, *Numerical Linear Algebra with Applications*, 19 (2012), 830-841.

[80] L. Zhang, L. Qi and Y. Xu, ”Linear convergence of the LZI algorithm for weakly positive tensors”, *Journal of Computational Mathematics*, 30 (2012), 24-33.

[81] L. Zhang, L.Qi, Z. Luo and Y. Xu, ”The dominant eigenvalue of an essentially nonnegative tensor”, *Numer. Linear Algebra Appl.*, 00 (2013), 1-13.

[82] G. Zhou, L. Caccetta and L. Qi, ”Convergence of an algorithm for the largest singular value of a nonnegative rectangular tensor”,*Linear Algebra and Its Applications* 438 (2013), 959-968.