Polynomial Optimization: Structures, Algorithms, and Engineering Applications

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

BO JIANG

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

SHUZHONG ZHANG

August, 2013

Acknowledgements

I am deeply indebted to my advisor Professor Shuzhong Zhang, who brought me into serval interesting research topics, and has provided me constant supports and encouragements. Thanks to him, I had the great fortune to start my Ph.D. study at the Chinese University of Hong Kong (CUHK), and then continued to complete the whole Ph.D. program at the University of Minnesota. I have learned a lot from him. In particular, I am greatly inspired by his never-ending passion to understand how things work in a deep way, which, on several occasions, pushed me toward the direction leading to the beautiful results that I would not have expected before.

I would like to express my sincere gratitude to the members of my thesis committee, Professors Bill Cooper, John Carlsson and Zhi-Quan (Tom) Luo, for their careful reading and valuable comments on my dissertation. I would not have a smooth transition from CUHK to University of Minnesota, if I did not receive the strong support and help from Bill. I have learned a lot on the computational geometry and the algorithm design techniques in optimization from the two group discussions led by John and Tom respectively.

My gratitude extends to my other collaborators: Shiqian Ma, Zhening Li, Simai He, Augusto Aubry, and Antonio De Maio. The numerous fruitful discussions with them have stimulated a lot of exiting joint works. Especially, I would like to thank Shiqian and Zhening for their academic insights and career advice, and the email-discussion with Augusto is a memorable experience.

My officemates and friends in both CUHK and University of Minnesota have made my study enjoyable. Particularly, I would like to acknowledge the friendship and the support that I received from: Youfeng Su, Kencheng Wang, Binyang Li, Dakui Wang, Lanjun Zhou, Bilian Chen, Xiaoguo Wang, Keith Wong, Ke Hou, Yi Yang, Yanyi Yang,

Shuang Chen, Yibin Chen, Chenhao Du, Yan Liu, and Fan Jia.

I am greatly indebted to my parents, Shujie Zhang and Ligui Jiang, who have always filled my life with unconditional love and support. I would also like to express my deepest gratitude to my wife, Liu Yang, for her company and support throughout my Ph.D. study.

This dissertation is devoted in part to my mentor, Professor Xuexiang Huang, at Fundan University. I may never start research on optimization and pursue the Ph.D. degree if I did not meet him. It has been almost ten months since his passing away. I hope he could be proud of my academic achievement should he still be around to read this thesis.

Dedication

This work is dedicated to my parents and to Professor Xuexiang Huang.

Abstract

As a fundamental model in Operations Research, polynomial optimization has been receiving increasingly more attention in the recent years, due to its versatile modern applications in engineering such as biomedical engineering, signal processing, material science, speech recognition, and so on. In this thesis, we present a systematic study of polynomial optimization. First, we study the structures of various polynomial functions, based on which efficient algorithms will consequently be developed. The newly developed solution methods will be tested on a variety of engineering applications. Specifically, we first study the class of nonnegative polynomials. Six fundamentally important types of nonnegative quartic functions coagulating into a chain in decreasing order will be presented. This structure is useful in understanding the nature of quartic polynomials. We further consider the polynomial sized representation of a very specific nonnegative polynomial, and this representation enables us to address an open question asserting that the computation of the matrix $2 \mapsto 4$ norm is NP-hard in general. Then we proceed to studying polynomial function in random variables and establish a series of fundamental probability inequalities. Similar to the relationship between the symmetric matrices and the quadratic functions, there also exists a one-to-one correspondence between the super-symmetric tensors and the homogeneous polynomials, and this leads to the knowledge of tensor related problems. We then proceed to a new notion of matrix-rank for the even order tensors. Unlike the CP-rank of tensors, the matrix-rank is easy to compute. On the computational side, the afore-mentioned probability inequalities lead to new approximation algorithms with better approximation ratios than the ones known in the literature. We also propose approximation algorithms for polynomial optimization in complex variables. At the same time, we consider first order algorithms such as the alternating direction method of multipliers (ADMM), and the maximum block improvement algorithms (MBI). Finally, we test the new algorithms by solving real engineering problems including the tensor PCA problem, the tensor recovery problem in computer vision and the radar waveform design problem. Excellent performances of the proposed methods have been confirmed by our numerical results.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation and Literature Review

Polynomial optimization is a fundamental model in the field of Operations Research. Basically, it is to maximize (or minimize) a polynomial objective function, subject to certain polynomial constraints. Recently this problem has attracted much attention, due to its widely applications in various engineering problems such as biomedical engineering [1, 2, 3], signal processing [4, 5, 6], material science [7], speech recognition [8]. To motivate our study and illustrate the usefulness of polynomial optimization, we shall mention a few concrete examples as immediate applications below. For instance, in the field of biomedical engineering, Ghosh et al. [2] formulated a fiber detection problem in Diffusion Magnetic Resonance Imaging (MRI) by maximizing a homogenous polynomial function, subject to a spherical constraint, Zhang et al. [3] proposed a new framework for co-clustering of gene expression data based on generic multi-linear optimization (a special case of polynomial optimization) model. There are also many polynomial optimization problems arising from signal processing, see e.g. Maricic et al. [4], a quartic polynomial model was proposed for blind channel equalization in digital communication, and in Aittomaki and Koivunen [9] the problem of beampattern synthesis in array signal processing was formulated as a complex multivariate quartic minimization problem. Soare, Yoon, and Cazacu [7] proposed some 4th, 6th and 8th order homogeneous polynomials to describe the plastic anisotropy of orthotropic sheet metal, a typical problem in material sciences.

As we have seen, it is basically impossible to list, even partially, the successful stories of polynomial optimization. However, compared with the intensive and extensive study of quadratic problems, the study of higher order polynomial optimization, even the quartic model, is quite limited. For example, consider the following query:

> given a fourth degree polynomial function in n variables, can one easily tell if the function is convex or not?

This simple-looking question was first put forward by Shor [10] in 1992, which turned out later to be a very challenging question to answer. For almost two decades, the question remained open. Only until recently Ahmadi *et al.* [11] proved that checking the convexity of a general quartic polynomial function is actually strongly NP-hard. Notice that checking the convexity of a quadratic function is an easy problem. Therefore, their groundbreaking result not only settled this particular open problem, but also helped to indicate that the study of generic polynomial optimization will be all the more compelling and interesting.

This Ph.D. thesis aims at approaching polynomial optimization by first studying the structures of various polynomial functions, and then proposing efficient algorithms to solve various polynomial optimization models, then finally presenting results of novel engineering applications via polynomial optimization models.

The first step towards the systematic study on polynomial optimization, of course, is to understand how the polynomial functions behave. In particular, we shall focus on the nonnegativity of polynomials, this is because there is an intrinsic connection between optimizing a polynomial function and the description of *all* the polynomial functions that are nonnegative over a given domain. For the case of quadratic polynomials, this connection was explored by Sturm and Zhang in [12], and later for the bi-quadratic case in Luo *et al.* [13]. For higher order polynomial function, historically, such investigations can be traced back to the 19th century when the relationship between nonnegative polynomial functions and the sum of squares (SOS) of polynomials was explicitly studied. Hilbert [14] in 1888 showed that the only three classes of polynomial functions where this is generically true can be explicitly identified: (1) univariate polynomials; (2) multivariate quadratic polynomials; (3) bivariate quartic polynomials. There are certainly other interesting classes of nonnegative polynomials. For instance, the convex polynomial functions, and sos-convex polynomials introduced by Helton and Nie [15] can both

be categorized into certain kind of nonnegative polynomial.

Besides, a very interesting and specific nonnegative polynomial in the form of $\left(\sum_{i=1}^{n} x_i^2\right)^d$ deserves a further investigation. Hilbert showed that this kind of polynomial bears certain 'rank one' representation structure. In particular for any fixed positive integers d and n, there always exist rational vectors $a^1, a^2, ..., a^t \in \mathbb{R}^n$ such that

$$\left(\sum_{i=1}^{n} x_i^2\right)^d = \sum_{j=1}^{t} \left((a^j)^\top x\right)^{2d}, \text{ where } x = (x_1, x_2, \cdots, x_n) \in \mathbb{R}^n. \tag{1.1}$$

For instance, when $n = 4$ and $d = 2$, we have

$$(x_1^2 + x_2^2 + x_3^2 + x_4^2)^2 = \frac{1}{6} \sum_{1 \le i < j \le 4} (x_i + x_j)^4 + \frac{1}{6} \sum_{1 \le i < j \le 4} (x_i - x_j)^4,$$

which is called Liouville's identity. In the literature, identity (1.1) is often referred to as Hilbert's identity, and this identity turns out to be extremely useful. For example, with the help of (1.1), Reznick [16] managed to prove the following result:

> Let $p(x)$ be $2d$-th degree homogeneous positive polynomial in n variables. Then there exists a positive integer r and vectors $c_1, c_2, \cdots, c_t \in \mathbb{R}^n$ such that
>
> $$\|x\|_2^{2r-2d} p(x) = \sum_{i=1}^{t} (c_i^\top x)^{2r}$$
>
> for all $x \in \mathbb{R}^n$.

Reznick's result above solves Hilbert's seventeenth problem constructively (albeit only for the case $p(x)$ is positive definite). Hilbert (see [17]) proved (1.1) constructively, however, in his construction, the number of $2d$ powered linear items on the right hand side is $(2d+1)^n$, which is exponential in n. For practical purposes, this representation is too lengthy. As a matter of fact, by Carathéodory's theorem [18], one can argue that in principle there exist no more than $\binom{n+2d-1}{2d}$ items in the expression (1.1). Unfortunately, Carathéodory's theorem is non-constructive, and thus an open problem remains: For fixed d, find a *polynomially sized* representation for (1.1).

Therefore, conducting a systematic study on the nonnegative polynomials becomes one of the central topics in this thesis.

Besides, we also interested in the probability in the form of

$$\operatorname*{Prob}_{\xi\sim S_0}\left\{f(\xi)\geq\tau\max_{x\in S}f(x)\right\}\geq\theta, \tag{1.2}$$

where $f(\cdot)$ is certain polynomial function, $\tau>0$ and $0<\theta<1$ are certain constants. This is because that most classical results in probability theory is to upper bound the tail of a distribution (e.g. the Markov inequality and the Chebyshev inequality), say $\operatorname{Prob}\{\xi\geq a\}\leq b$. However, in some applications a *lower* bound for such probability can be relevant.

One interesting example is a result due to Ben-Tal, Nemirovskii, and Roos [19], where they proved a lower bound of $1/8n^2$ for the probability that a homogeneous quadratic form of n binary i.i.d. Bernoulli random variables lies above its mean. More precisely, they proved the following:

> Let $F\in\mathbb{R}^{n\times n}$ be a symmetric matrix and $\xi=(\xi_1,\xi_2,\cdots,\xi_n)$ be i.i.d. Bernoulli random variables, each taking values 1 and -1 with equal probability. Then $\operatorname{Prob}\{\xi^\top F\xi\geq\operatorname{tr}(F)\}\geq 1/8n^2$.

As a matter of fact, the author went on to conjecture in [19] that the lower bound can be as high as $1/4$, which was very recently disproved by Yuan [20]. However, the exact universal constant lower bound remains open. A significant progress on this conjecture is due to He *et al.* [21], where the authors improved the lower bound of $1/8n^2$ to 0.03. Note that the result of He *et al.* [21] also holds for any ξ_i's being i.i.d. standard normal variables. Luo and Zhang [22] provides a constant lower bound for the probability that a homogeneous quartic function of a zero mean multi-variate normal distribution lies above its mean, which was a first attempt to extend such probability bound for functions of random vector beyond quadratic. Our goal is to establish (1.2) for generic polynomial.

It is also very helpful to study the various properties of tensors, since there is a one-to-one mapping from the homogeneous polynomial functions with degree d to the d-th order super-symmetric tensors (see Section 1.3). Besides, the emergence of multidimensional data in signal processing, computer vision, medical imaging and machine learning can also be viewed as tensors. Therefore tensor-based data analysis attracted more and more research attention. In practice, the underlying tensor often appears to be equipped with

some low-rank structure. However, the commonly used tensor *CP-rank* [23, 24] is hard to compute [25]. Therefore, in this thesis, we propose the so-called matrix-rank for even order tensors, and show that this rank is easy to compute and bears many interesting properties. And this matrix-rank can be further used in the low-rank optimization problems.

Our second main research focus deals with designing efficient algorithms that solve particular kinds of polynomial optimization problems; this is already a challenging task. For example, even the simplest instances of polynomial optimization, such as maximizing a cubic polynomial over a sphere, is NP-hard (Nesterov [26]). To find the global optimizer, Lasserre [27, 28] and Parrilo [29, 30] developed an approach called SOS, however this method has only theoretical appeal, since it needs solving a (possibly large) Semidefinite Program and in many cases the solver based on SOS method can merely get a bound (not feasible solution) for the optimal value before it stops. Therefore, it is natural to ask whether one can design efficient algorithms for a large class of polynomial optimization problems with provable approximation guarantees. As a first step towards answering that question, de Klerk et al. [31] considered the problem of optimizing a fixed degree even form over the sphere and designed a polynomial time approximation scheme. The problem of optimizing a more general multivariate polynomial was not addressed until Luo and Zhang [22] designed the first polynomial time approximation algorithm for optimizing a multivariate quartic polynomial over a region defined by quadratic inequalities. Sooner afterward, Ling et al. [32] studied a special quartic optimization model, which is to maximize a bi-quadratic function over two spherical constraints. Most recently, He, Li and Zhang presented a series of works on general homogenous (inhomogenous, discrete) polynomial optimization [33, 34, 35]. So [36] reinvestigate sphere constrained homogeneous polynomial problems and propose a deterministic algorithm with an improved approximation ratio. For a comprehensive survey on the topic, one may refer to the recent Ph.D. thesis of Li [37].

In case of complex valued polynomial optimization, Ben-Tal, Nemirovski and Roos [38] first studied complex quadratic optimization model with objective function being nonnegative by using complex matrix cube theorem. Zhang and Huang [39], So, Zhang and Ye [40] considered complex quadratic (include conjugate items) problems. After that Huang and Zhang [41] also considered bi-linear complex optimization problems.

However, to the best of our knowledge there is no result in the literature on approximation algorithms for higher order complex polynomial optimization problems. In this thesis we also provide various approximation algorithms to complex polynomial optimizations as well as some real valued polynomial optimizations with spherical (binary) constraints.

1.2 Main Contributions and Organization

This thesis is organized as follows. We shall start our discussion by exploring various structures of polynomial functions and tensors, which will cover Chapters 2 to 5.

In Chapter 2, we first introduce the definition of six different nonnegative quartic polynomials. Then we shall prove they form an interesting hierarchical structure (Theorem 2.3.1). The computational complexity of each kind of nonnegative quartic polynomial is also discussed. In the final section of this chapter, we study the so-called quartic conic programming. In fact many quartic optimizations including bi-quadratic assignment problems (Corollary 2.5.3) and finding eigenvalues of super-symmetric tensors (Corollary 2.5.4) can be modeled as a quartic conic problem.

In Chapter 3, we point out that polynomial sized representation of Hilbert's identity (i.e. (1.1)) is equivalent to constructing the k-wise zero correlated random variables with polynomial sized sample space (Theorem 3.2.2). And when the supporting set of random variable satisfies certain symmetric condition, the k-wise zero correlated random variables can be constructed in an elegant way. Consequently, we provide a polynomial sized representation of (1.1), when d is 2 (Theorem 3.4.1). This result can be further extended to complex polynomials. As an application, we applied our new construction to prove that computing the matrix $2 \mapsto 4$ norm problem is NP-hard, whose complexity status was previously unknown (cf. [42]).

We propose matrix-rank for even order tensors in Chapter 4. In particular, we unfold an even order tensor into a matrix, whose row index is formed by one half of the indices of the tensor and column index is obtained by the other half. The matrix-rank of the original tensor is exactly the matrix of the resulting matrix. For 4-th order tensor, we show that CP-rank of the tensor can be both lower and upper bounded by the matrix-rank multiplied by a constant related to dimension n (Theorems 4.3.1, 4.3.4). Moreover,

for super-symmetric tensor, we show that the CP-rank one tensor and the matrix-rank one tensor coincide (Theorem 4.4.7).

In Chapter 5, we set out to explore the probability bound in the form of (1.2). The function $f(\cdot)$ under consideration is either a multi-linear form or is a polynomial function (In the following description, we use their equivalent tensor representations; see Section 1.3 for relationship between polynomials and tensors). To enable probability bounds in the form of (1.2), we will need some structure in place. In particular, we consider the choice of the structural sets S_0 and S respectively as follows:

1. Consider $S = \mathbb{B}^{n_1 \times n_2 \times \cdots \times n_d}$ and $S_0 = \{X \in S \mid \mathrm{rank}\,(X) = 1\}$, and $\mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$. If we draw ξ uniformly over S_0, then

$$\mathrm{Prob}\left\{\mathcal{F} \bullet \xi \geq c_3^{d-1}\sqrt{\frac{\delta \ln n_d}{\prod_{i=1}^d n_i}}\max_{X\in S}\mathcal{F}\bullet X = c_3^{d-1}\sqrt{\frac{\delta \ln n_d}{\prod_{i=1}^d n_i}}\,\|\mathcal{F}\|_1\right\} \geq \frac{c_1(\delta)c_3^{2d-2}}{n_d^{\delta}\prod_{i=2}^d n_i^{i-1}},$$

where $c_1(\delta)$ is a constant depended only on constant $\delta \in (0, \frac{1}{2})$ and $c_3 = \frac{8}{25\sqrt{5}}$. Moreover, the order of $\sqrt{\frac{\ln n_d}{\prod_{i=1}^d n_i}}$ cannot be improved if the bound is required to be at least a polynomial function of $\frac{1}{n_d}$.

2. Consider $S = \{X \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d} \mid X \bullet X = 1\}$ and $S_0 = \{X \in S \mid \mathrm{rank}\,(X) = 1\}$, and $\mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$. If we draw ξ uniformly over S_0, then

$$\begin{aligned}&\mathrm{Prob}\left\{\mathcal{F} \bullet \xi \geq \frac{1}{2^{\frac{d-1}{2}}}\sqrt{\frac{\gamma \ln n_d}{\prod_{i=1}^d n_i}}\max_{X\in S}\mathcal{F}\bullet X = \frac{1}{2^{\frac{d-1}{2}}}\sqrt{\frac{\gamma \ln n_d}{\prod_{i=1}^d n_i}}\,\|\mathcal{F}\|_2\right\}\\ \geq\ & \frac{c_2(\gamma)}{4^{d-1}n_d^{2\gamma}\sqrt{\ln n_d}\prod_{i=1}^{d-1} n_i},\end{aligned}$$

where $c_2(\gamma)$ is a constant depended only on constant $\gamma \in (0, \frac{n_d}{\ln n_d})$. Moreover, the order of $\sqrt{\frac{\ln n_d}{\prod_{i=1}^d n_i}}$ cannot be improved if the bound is required to be at least a polynomial function of $\frac{1}{n_d}$.

3. Consider $S = \{X \in \mathbb{B}^{n^d} \mid X \text{ is super-symmetric}\}$ and $S_0 = \{X \in S \mid \mathrm{rank}\,(X) = 1\}$, and a square-free super-symmetric tensor $\mathcal{F} \in \mathbb{R}^{n^d}$. If we draw ξ uniformly over S_0, then there exists a universal constant $c > 0$, such that

$$\mathrm{Prob}\left\{\mathcal{F} \bullet \xi \geq \sqrt{\frac{d!}{16n^d}}\max_{X\in S}\mathcal{F}\bullet X = \sqrt{\frac{d!}{16n^d}}\,\|\mathcal{F}\|_1\right\} \geq c.$$

Moreover, when $d = 2$ or $d = 4$, the order of $n^{-\frac{d}{2}}$ cannot be improved.

4. Consider $S = \{X \in \mathbb{R}^{n^d} : X \bullet X = 1, X \text{ is super-symmetric}\}$ and $S_0 = \{X \in S \mid \text{rank}\,(X) = 1\}$, and a square-free super-symmetric tensor $\mathcal{F} \in \mathbb{R}^{n^d}$. If we draw ξ uniformly over S_0, then there exists a universal constant $c > 0$, such that

$$\text{Prob}\left\{\mathcal{F} \bullet \xi \geq \sqrt{\frac{d!}{48(4n)^d}} \max_{X \in S} \mathcal{F} \bullet X = \sqrt{\frac{d!}{48(4n)^d}}\, \|\mathcal{F}\|_2\right\} \geq c.$$

After ending the discussion on structures of polynomial functions, we shall focus on approximation algorithms for various polynomial optimizations through Chapters 6 to 7.

In Chapter 6 by utilizing probability inequalities obtained in Chapter 5, we propose a new simple randomization algorithm, which improve upon the previous approximation ratios [33, 35].

Chapter 7 is devoted to the approximation algorithms for complex polynomial optimization, where the objective function can either be the complex multi-linear form, the complex homogeneous polynomial or the conjugate polynomial, while the constraints can be the m-th roots of unity constraint, the unity constraint or the complex spherical constraint.

The third part of this thesis is to study various applications of polynomial optimization and the results established in the previous chapters will play important roles in solving these problems.

Specifically, the rank-one equivalence between CP-rank and matrix-rank leads to a new formulation of tensor PCA problem in Chapter 8 and the alternating direction method of multipliers is proposed to solve this new formulation.

We study low-rank tensor optimization in Chapter 9. Precisely, they are low-rank tensor completion problem and robust tensor recovery problem respectively. To make these problems tractable, we first replace the CP-rank by the matrix-rank and then consider their convex reformulations. Moreover, some numerical examples are provided to show that matrix-rank works well in these problems.

Finally, in Chapter 10, we propose a cognitive approach to design phase-only modulated waveforms sharing a desired ambiguity function. This problem can be formulated as a conjugate quartic optimization with the unite circle constraint. The tensor representation of the conjugate polynomial studied in Chapter 7 is helpful to design the

so-called maximum block improvement algorithm to solve this optimization problem. The performance of this algorithm is justified by our numerical results.

1.3 Notations and Preliminaries

Throughout this thesis, we use the lower-case letters to denote vectors (e.g. $x \in \mathbb{R}^n$), the capital letters to denote matrices (e.g. $A \in \mathbb{R}^{n^2}$). For a given matrix A, we use $\|A\|_*$ to denote the nuclear norm of A, which is the sum of all the singular values of A. The boldface letter $\boldsymbol{i}$ represents the imaginary unit (i.e. $\boldsymbol{i} = \sqrt{-1}$). The transpose, the conjugate, and the conjugate transpose operators are denoted by the symbols $(\cdot)^\top$, $\bar{(\cdot)}$, and $(\cdot)^\dagger$ respectively. For any complex number $z = a + \boldsymbol{i}b \in \mathbb{C}$ with $a, b \in \mathbb{R}$, its real part is denoted by $\mathrm{Re}\ z = a$, and its modulus by $|z| = \sqrt{z^\dagger z} = \sqrt{a^2 + b^2}$. For $x \in \mathbb{C}^n$, its norm is denoted by $\|x\| := \left(\sum_{i=1}^n |x_i|^2\right)^{\frac{1}{2}}$.

A tensor in real field is a high dimensional array of real data, usually in calligraphic letter, and is denoted as $\mathcal{A} = (\mathcal{A}_{i_1 i_2 \cdots i_m})_{n_1 \times n_2 \times \cdots \times n_m}$. The space where $n_1 \times n_2 \times \cdots \times n_m$ dimensional real-valued tensor resides is denoted by $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$. We call $\mathcal{A}$ super-symmetric if $n_1 = n_2 = \cdots = n_m$ and $\mathcal{A}_{i_1 i_2 \cdots i_m}$ is invariant under any permutation of $(i_1, i_2, ..., i_m)$, i.e., $\mathcal{A}_{i_1 i_2 \cdots i_m} = \mathcal{A}_{\pi(i_1, i_2, \cdots, i_m)}$, $\pi(i_1, i_2, \cdots, i_m)$ is any permutation of indices $(i_1, i_2, \cdots, i_m)$. The set of all distinct permutations of the indices $\{i_1, i_2, \cdots, i_d\}$ is denoted by $\Pi(i_1 i_2 \cdots i_d)$. The space where $\underbrace{n \times n \times \cdots \times n}_{m}$ super-symmetric tensors reside is denoted by $\mathbf{S}^{n^m}$.

A generic form is a homogeneous polynomial function in n variables, or specifically the function

$$f(x) = \sum_{1 \le i_1 \le \cdots \le i_m \le n} a_{i_1 \cdots i_m} x_{i_1} \cdots x_{i_m}, \tag{1.3}$$

where $x = (x_1, \cdots, x_n)^\top \in \mathbb{R}^n$. In fact, super-symmetric tensors are bijectively related to forms. In particular, restricting to m-th order tensors, for a given super-symmetric tensor $\mathcal{A} \in \mathbf{S}^{n^m}$, the form in (1.3) can be uniquely determined by the following operation:

$$f(x) = \mathcal{A}(\underbrace{x, \cdots, x}_{m}) := \sum_{1 \le i_1, \cdots, i_m \le n} \mathcal{A}_{i_1 \cdots i_m} x_{i_1} \cdots x_{i_m}, \tag{1.4}$$

where $x \in \mathbb{R}^n$, $\mathcal{A}_{i_1 \cdots i_m} = a_{i_1 \cdots i_m} / |\Pi(i_1 \cdots i_m)|$, and vice versa. (This is the same as the

one-to-one correspondence between a symmetric matrix and a quadratic form.)

Special cases of tensors are vector ($m = 1$) and matrix ($m = 2$), and tensors can also be seen as a long vector or a specially arranged matrix. For instance, the tensor space $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ can also be seen as a matrix space $\mathbb{R}^{(n_1 \times n_2 \times \cdots \times n_{m_1}) \times (n_{m_1+1} \times n_{m_1+2} \times \cdots \times n_m)}$, where the row is actually an m_1 array tensor space and the column is another $m - m_1$ array tensor space. Such connections between tensor and matrix re-arrangements will play an important role in this thesis. As a convention in this thesis, if there is no other specification we shall adhere to the Euclidean norm (i.e. the L_2-norm) for vectors and tensors; in the latter case, the Euclidean norm is also known as the Frobenius norm, and is sometimes denoted as $\|\mathcal{A}\|_F = \sqrt{\sum_{i_1,i_2,\ldots,i_m} \mathcal{A}^2_{i_1 i_2 \cdots i_m}}$. Regarding the products, we use $\otimes$ to denote the outer product for tensors; that is, for $\mathcal{A}_1 \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ and $\mathcal{A}_2 \in \mathbb{R}^{n_{m+1} \times n_{m+2} \times \cdots \times n_{m+\ell}}$, $\mathcal{A}_1 \otimes \mathcal{A}_2$ is in $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_{m+\ell}}$ with

$$(\mathcal{A}_1 \otimes \mathcal{A}_2)_{i_1 i_2 \cdots i_{m+\ell}} = (\mathcal{A}_1)_{i_1 i_2 \cdots i_m} (\mathcal{A}_2)_{i_{m+1} \cdots i_{m+\ell}},$$

and $\otimes$ also denotes the outer product between vectors, in other words,

$$(x^1 \otimes x^2 \otimes \cdots \otimes x^m)_{i_1 i_2 \cdots i_m} = \prod_{k=1}^{m} (x^k)_{i_k}.$$

The inner product between tensors $\mathcal{A}_1$ and $\mathcal{A}_2$ residing in the same space $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ is denoted

$$\mathcal{A}_1 \bullet \mathcal{A}_2 = \sum_{i_1,i_2,\ldots,i_m} (\mathcal{A}_1)_{i_1 i_2 \cdots i_m} (\mathcal{A}_2)_{i_1 i_2 \cdots i_m}.$$

Under this light, a multi-linear form $\mathcal{A}(x^1, x^2, ..., x^m)$ can also be written in inner/outer products of tensors as

$$\mathcal{A} \bullet (x^1 \otimes \cdots \otimes x^m) := \sum_{i_1,\cdots,i_m} \mathcal{A}_{i_1,\cdots,i_m} (x^1 \otimes \cdots \otimes x^m)_{i_1,\cdots,i_m} = \sum_{i_1,\cdots,i_m} \mathcal{A}_{i_1,\cdots,i_m} \prod_{k=1}^{m} x^k_{i_k}.$$

Given an even order tensor $\mathcal{A} \in \mathbf{S}^{n^{2d}}$ and a tensor $\mathcal{X} \in \mathbb{R}^{n^d}$, we may also define the following operation (in the same spirit as (1.4)):

$$\mathcal{A}(\mathcal{X}, \mathcal{X}) := \sum_{1 \le i_1,\cdots,i_{2d} \le n} \mathcal{A}_{i_1 \cdots i_{2d}} \mathcal{X}_{i_1 \cdots i_d} \mathcal{X}_{i_{d+1} \cdots i_{2d}}.$$

Chapter 2

Cones of Nonnegative Quartic Forms

2.1 Introduction

2.1.1 Motivation

Checking the convexity of a quadratic function boils down to testing the positive semidefiniteness of its Hessian matrix in the domain. Since the Hessian matrix is constant, the test can be done easily. We also mentioned the following simple-looking question due to Shor [10] in 1992:

> Given a fourth degree polynomial function in n variables, can one still easily tell if the function is convex or not?

This question remains open for almost two decades, until recently Ahmadi et al. [11] proved that checking the convexity of a general quartic polynomial function is actually strongly NP-hard. This groundbreaking result helps to highlight a crucial difference between quartic and quadratic polynomials and attracts our attention to the quartic polynomial functions.

Among all the quartic polynomials, we are particularly interested in the nonnegative quartic polynomials, that is the output of the polynomial function is nonnegative. This is because there is an intrinsic connection between optimizing a polynomial function and

the description of *all* the polynomial functions that are nonnegative over a given domain. For the case of quadratic polynomials and bi-quadratic function, this connection was explored in [12] and [13] respectively. Such investigations can be traced back to the 19th century when the relationship between nonnegative polynomial functions and the sum of squares (SOS) of polynomials was explicitly studied. Hilbert [14] in 1888 showed that there are only three classes of nonnegative polynomial functions: (1) univariate polynomials; (2) multivariate quadratic polynomials; (3) bivariate quartic polynomials, which can be represented as sum of squares of polynomial functions. Since polynomial functions with a fixed degree form a vector space, and the *nonnegative* polynomials and the *SOS* polynomials form two convex cones respectively within that vector space, the afore-mentioned results can be understood as a specification of three particular cases where these two convex cones coincide, while in general of course the cone of nonnegative polynomials is larger. There are certainly other interesting convex cones in the same vector space. For instance, the convex polynomial functions form yet another convex cone in that vector space. Helton and Nie [15] introduced the notion of sos-convex polynomials, to indicate the polynomials whose Hessian matrix can be decomposed as a sum of squares of polynomial matrices. All these classes of convex cones are important in their own rights.

There are some substantial recent progresses along the relationships among various nonnegative polynomials. As we mentioned earlier, e.g. the question of Shor [10] regarding the complexity of deciding the convexity of a quartic polynomial was nicely settled by Ahmadi et al. [11]. It is also natural to inquire if the Hessian matrix of a convex polynomial is sos-convex. Ahmadi and Parrilo [43] gave an example to show that this is not the case in general. Blekherman proved that a convex polynomial is not necessary a sum of squares [44] if the degree of the polynomial is larger than two. However, Blekherman's proof is not constructive, and it remains an open problem of constructing a concrete example of convex polynomial which is not a sum of squares. Reznick [45] studied the sum of even powers of linear forms, the sum of squares of forms, and the positive semidefinite forms.

In view of the cones formed by the polynomial functions (e.g. the cones of nonnegative polynomials, the convex polynomials, the SOS polynomials and the sos-convex polynomials), it is natural to inquire about their relational structures, complexity status

and the description of their interiors. We aim to conduct a systematic study on those topics in this chapter, to bring together much of the known results in the context of our new findings, and to present them in a self-contained manner. In a way there is a 'phase transition' in terms of complexity when the scope of polynomials goes beyond quadratics. Compared to the quadratic case (cf. Sturm and Zhang [12]), the structure of the quartic forms is far from being clear. We believe that the class of quartic polynomial functions (or the class of quartic forms) is an appropriate subject of study on its own right, beyond quadratic functions (or matrices). There are at least three immediate reasons to elaborate on the quartic polynomials, rather than polynomial functions of other (or general) degrees. First of all, nonnegativity is naturally associated with even degree polynomials, and the quartic polynomial is next to quadratic polynomials in that hierarchy. Second, quartic polynomials represent a landscape *after* the 'phase transition' takes place. However, dealing with quartic polynomials is still manageable, as far as notations are concerned. Finally, from an application point of view, quartic polynomial optimization is by far the most relevant polynomial optimization model beyond quadratic polynomials. The afore-mentioned examples such as kurtosis risks in portfolio management ([46]), the bi-quadratic optimization models ([32]), and the nonlinear least square formulation of sensor network localization ([47]) are all such examples. In this chapter, due to the one-to-one correspondence between a super-symmetric tensor and a homogenous polynomial, we provide various characterizations of several important convex cones in the fourth order super-symmetric tensor space, present their relational structures and work out their complexity status. Therefore, our results can be helpful in tensor optimization (see [48, 49] for recent development in sparse or low rank tensor optimization). We shall also motivate the study by some examples from applications.

2.1.2 Introducing the Quartic Forms

In this subsection we shall formally introduce the definitions of the quartic forms in the super-symmetric fourth order tensor space. The set of n-dimensional super-symmetric fourth order tensors is denoted by $\mathbf{S}^{n^4}$. In the remainder of this chapter, we shall frequently use a super-symmetric tensor $\mathcal{F} \in \mathbf{S}^{n^4}$ to indicate a quartic form $\mathcal{F}(x, x, x, x)$, i.e., the notion of "super-symmetric fourth order tensor" and "quartic form" are used interchangeably.

Let us start with the well known notion of positive semidefinite (PSD) and the sum of squares (SOS) of polynomials.

Definition 2.1.1. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called quartic PSD if*

$$\mathcal{F}(x,x,x,x) \geq 0 \quad \forall\, x \in \mathbb{R}^n. \tag{2.1}$$

The set of all quartic PSD forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{S}^{n^4}_+$.

If a quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ can be written as a sum of squares of polynomial functions, then these polynomials must be quadratic forms, i.e.,

$$\mathcal{F}(x,x,x,x) = \sum_{i=1}^{m} \left(x^\top A^i x\right)^2 = (x \otimes x \otimes x \otimes x) \bullet \sum_{i=1}^{m} A^i \otimes A^i,$$

where $A^i \in \mathbf{S}^{n^2}$, the set of symmetric matrices. However, $\sum_{i=1}^m \left(A^i \otimes A^i\right) \in \overrightarrow{\mathbf{S}}^{n^4}$ is only partial-symmetric, and may not be exactly $\mathcal{F}$, which must be super-symmetric. To place it in the family $\mathbf{S}^{n^4}$, a symmetrization operation is required. Since $x \otimes x \otimes x \otimes x$ is super-symmetric, we still have $(x \otimes x \otimes x \otimes x) \bullet \text{sym}\left(\sum_{i=1}^m A^i \otimes A^i\right) = (x \otimes x \otimes x \otimes x) \bullet \sum_{i=1}^m A^i \otimes A^i$.

Definition 2.1.2. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called quartic SOS if $\mathcal{F}(x,x,x,x)$ is a sum of squares of quadratic forms, i.e., there exist m symmetric matrices $A^1, \ldots, A^m \in \mathbf{S}^{n^2}$ such that*

$$\mathcal{F} = \text{sym}\left(\sum_{i=1}^{m} A^i \otimes A^i\right) = \sum_{i=1}^{m} \text{sym}\left(A^i \otimes A^i\right).$$

The set of quartic SOS forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{\Sigma}^2_{n,4}$.

As all quartic SOS forms constitute a convex cone, we have

$$\mathbf{\Sigma}^2_{n,4} = \text{sym cone}\left\{A \otimes A \,|\, A \in \mathbf{S}^{n^2}\right\}.$$

Usually, for a given $\mathcal{F} = \text{sym}\left(\sum_{i=1}^m A^i \otimes A^i\right)$ it maybe a challenge to write it explicitly as a sum of squares, although the construction can in principle be done in polynomial-time by semidefinte programming (SDP), which however is costly. In this sense, having a quartic SOS tensor in super-symmetric form may not always be beneficial, since the super-symmetry can destroy the SOS structure.

Since $\mathcal{F}(X,X)$ is a quadratic form, the usual sense of nonnegativity carries over. Formally we introduce this notion below.

Definition 2.1.3. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called quartic matrix PSD if*

$$\mathcal{F}(X, X) \geq 0 \quad \forall\, X \in \mathbb{R}^{n^2}.$$

The set of quartic matrix PSD forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{S}_+^{n^2 \times n^2}$.

We remark that the matrix PSD forms is essentially equivalent to the cone of PSD moment matrices; see e.g. [50]. But our definition here is more clear and straightforward.

Related to the sum of squares for quartic forms, we now introduce the notion to the *sum of quartics* (SOQ): If a quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is SOQ, then there are m vectors $a^1, \ldots, a^m \in \mathbb{R}^n$ such that

$$\mathcal{F}(x, x, x, x) = \sum_{i=1}^{m} \left(x^\top a^i\right)^4 = (x \otimes x \otimes x \otimes x) \bullet \sum_{i=1}^{m} a^i \otimes a^i \otimes a^i \otimes a^i.$$

Definition 2.1.4. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called quartic SOQ if $\mathcal{F}(x, x, x, x)$ is a sum of fourth powers of linear forms, i.e., there exist m vectors $a^1, \ldots, a^m \in \mathbb{R}^n$ such that*

$$\mathcal{F} = \sum_{i=1}^{m} a^i \otimes a^i \otimes a^i \otimes a^i.$$

The set of quartic SOQ forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{\Sigma}_{n,4}^4$.

As all quartic SOQ forms also constitute a convex cone, we denote

$$\mathbf{\Sigma}_{n,4}^4 = \operatorname{cone}\left\{a \otimes a \otimes a \otimes a \,|\, a \in \mathbb{R}^n\right\} \subseteq \mathbf{\Sigma}_{n,4}^2.$$

In the case of quadratic functions, it is well known that for a given homogeneous form (i.e., a symmetric matrix, for that matter) $A \in \mathbf{S}^{n^2}$ the following two statements are equivalent:

- A is positive semidefinite (PSD): $A(x, x) := x^\top A x \geq 0$ for all $x \in \mathbb{R}^n$.
- A is a sum of squares (SOS): $A(x, x) = \sum_{i=1}^{m} (x^\top a^i)^2$ (or equivalently $A = \sum_{i=1}^{m} a^i \otimes a^i$) for some $a^1, \ldots, a^m \in \mathbb{R}^n$.

It is therefore clear that the four types of quartic forms defined above are actually different extensions of the nonnegativity. In particular, quartic PSD and quartic matrix PSD forms are extended from quadratic PSD, while quartic SOS and SOQ forms are in

the form of summation of nonnegative polynomials, and are extended from quadratic SOS. We will show later that there is an interesting hierarchical relationship for general n:

$$\mathbf{\Sigma}^4_{n,4} \subsetneq \mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{n,4} \subsetneq \mathbf{S}^{n^4}_+. \tag{2.2}$$

Recently, a class of polynomials termed the *sos-convex polynomials* (cf. Helton and Nie [15]) has been brought to attention, which is defined as follows (see [51] for three other equivalent definitions of the sos-convexity):

> A multivariate polynomial function $f(x)$ is sos-convex if its Hessian matrix $H(x)$ can be factorized as $H(x) = (M(x))^\top M(x)$ with a polynomial matrix $M(x)$.

The reader is referred to [43] for applications of the sos-convex polynomials. In this chapter, we shall focus on $\mathbf{S}^{n^4}$ and investigate sos-convex quartic forms with the hierarchy (2.2). For a quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$, it is straightforward to compute its Hessian matrix $H(x) = 12\mathcal{F}(x, x, \cdot, \cdot)$, i.e.,

$$(H(x))_{ij} = 12\mathcal{F}(x, x, e^i, e^j) \quad \forall\, 1 \le i, j \le n,$$

where $e^i \in \mathbb{R}^n$ is the vector whose i-th entry is 1 and other entries are zeros. Therefore $H(x)$ is a quadratic matrix of x. If $H(x)$ can be decomposed as $H(x) = (M(x))^\top M(x)$ with $M(x)$ being a polynomial matrix, then $M(x)$ must be linear with respect to x.

Definition 2.1.5. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called quartic sos-convex, if there exists a linear matrix $M(x)$ of x, such that its Hessian matrix*

$$12\mathcal{F}(x, x, \cdot, \cdot) = (M(x))^\top M(x).$$

The set of quartic sos-convex forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$.

Helton and Nie [15] proved that a nonnegative polynomial is sos-convex, then it must be SOS. In particular, if the polynomial is a quartic form, by denoting the i-th row of the linear matrix $M(x)$ to be $x^\top A^i$ for $i = 1, \ldots, m$ and some matrices $A^1, \ldots, A^m \in \mathbb{R}^{n^2}$, then $(M(x))^\top M(x) = \sum_{i=1}^m (A^i)^\top x x^\top A^i$. Therefore

$$\mathcal{F}(x, x, x, x) = x^\top \mathcal{F}(x, x, \cdot, \cdot)x = \frac{1}{12} x^\top (M(x))^\top M(x)x = \frac{1}{12}\sum_{i=1}^m \left(x^\top A^i x\right)^2 \in \mathbf{\Sigma}^2_{n,4}.$$

In addition, the Hessian matrix for a quartic sos-convex form is obviously positive semidefinite for any $x \in \mathbb{R}^n$. Hence sos-convexity implies convexity. Combining these two facts, we conclude that a quartic sos-convex form is both SOS and convex, which motivates us to study the last quartic forms in this chapter.

Definition 2.1.6. *A quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$ is called convex and SOS, if it is both quartic SOS and convex. The set of quartic convex and SOS forms in $\mathbf{S}^{n^4}$ is denoted by $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}$.*

Here $\mathbf{S}^{n^4}_{\text{cvx}}$ is denoted to be the set of all convex quartic forms in $\mathbf{S}^{n^4}$.

2.1.3 The Contributions and the Organization

All the sets of the quartic forms defined in Section 2.1.2 are clearly convex cones. The remainder of this chapter is organized as follows. In Section 2.2, we start by studying the cones: $\mathbf{S}^{n^4}_+$, $\mathbf{\Sigma}^2_{n,4}$, $\mathbf{S}^{n^2\times n^2}_+$, and $\mathbf{\Sigma}^4_{n,4}$. We first show that they are all closed, and that they can be presented in different formulations. As an example, the cone of quartic SOQ forms is

$$\mathbf{\Sigma}^4_{n,4} = \text{cone}\left\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbb{R}^n\right\} = \text{sym cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}_+,\, \text{rank}\,(A)=1\right\},$$

which can also be written as

$$\text{sym cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}_+\right\},$$

meaning that the rank-one constraint can be removed without affecting the cone itself. We know that among these four cones there are two primal-dual pairs: $\mathbf{S}^{n^4}_+$ is dual to $\mathbf{\Sigma}^4_{n,4}$, and $\mathbf{\Sigma}^2_{n,4}$ is dual to $\mathbf{S}^{n^2\times n^2}_+$, and a hierarchical relationship $\mathbf{\Sigma}^4_{n,4} \subsetneq \mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{n,4} \subsetneq \mathbf{S}^{n^4}_+$ exists. Although all these results can be found in [45, 50] thanks to various representations of quartic forms, it is beneficial to present them in a unified manner in the super-symmetric tensor space. Moreover, the tensor representation of quartic forms has interest in its own right. For instance, it sheds some light on how symmetric property changes the nature of quartic cones. To see this, let us consider an SOS quartic form $\sum_{i=1}^m (x^\top A^i x)^2$, which will become quartic matrix PSD if $\sum_{i=1}^m A^i\otimes A^i$ is already a super-symmetric tensor (Theorem 2.2.3). If we further assume $m=1$, then we have $\text{rank}\,(A^1)=1$ (Theorem 2.4 in [52]) meaning that $A^1\otimes A^1 = a\otimes a\otimes a\otimes a$ for

some a, is quartic SOQ. Besides, explicit examples are also very important for people to understand the quartic functions. It is worth mentioning that the main work of Ahmadi and Parrilo [43] is to provide a polynomial which is convex but not sos-convex. Here we present a new explicit quartic form, which is matrix PSD but not SOQ; see Example 2.2.11.

In Section 2.3, we further study two more cones: $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ and $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}$. Interestingly, these two new cones can be nicely placed in the hierarchical scheme (2.2) for general n:

$$\mathbf{\Sigma}^4_{n,4} \subsetneq \mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subseteq \left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}\right) \subsetneq \mathbf{\Sigma}^2_{n,4} \subsetneq \mathbf{S}^{n^4}_+. \tag{2.3}$$

The complexity status of all these cones are summarized in Section 2.4, including some well known results in the literature, our new finding is that testing the convexity is still NP-hard even for sums of squares quartic forms (Theorem 2.4.4). As a result, we show that $\mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subsetneq \left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}\right)$ unless $P = NP$, completing the picture presented in (2.3), on the premise that $P \neq NP$. The low dimensional cases of these cones are also discussed in Section 2.4. Specially, for the case $n = 2$, all the six cones reduce to only two distinctive ones, and for the case $n = 3$, they reduce to exactly three distinctive cones. In addition, we study two particular simple quartic forms: $\left(x^\top x\right)^2$ and $\sum_{i=1}^n {x_i}^4$. Since they both belong to $\mathbf{\Sigma}^4_{n,4}$, which is the smallest cone in our hierarchy, one may ask whether or not they belong to the interior of $\mathbf{\Sigma}^4_{n,4}$. Intuitively, it may appear plausible that $\sum_{i=1}^n {x_i}^4$ is in the interior of $\mathbf{\Sigma}^4_{n,4}$, for it is the quartic extension of quadratic unite form $\sum_{i=1}^n {x_i}^2$. However, our results show that $\sum_{i=1}^n {x_i}^4$ is not in $\text{Int}\,(\mathbf{S}^{n^4}_{\text{cvx}}) \supsetneq \text{Int}\,(\mathbf{\Sigma}^4_{n,4})$ but in $\text{Int}\,(\mathbf{\Sigma}^2_{n,4})$ (Theorem 2.4.10), and $\left(x^\top x\right)^2$ is actually in $\text{Int}\,(\mathbf{\Sigma}^4_{n,4})$ (Theorem 2.4.11), implying that $\left(x^\top x\right)^2$ is more 'positive' than $\sum_{i=1}^n {x_i}^4$.

Finally, in Section 2.5 we discuss applications of quartic conic programming, including bi-quadratic assignment problems and eigenvalues of super-symmetric tensors.

2.2 Quartic PSD Forms, Quartic SOS Forms, and the Dual Cones

Let us now consider the first four cones of quartic forms introduced in Section 2.1.2: $\mathbf{\Sigma}^4_{n,4}$, $\mathbf{S}^{n^2\times n^2}_+$, $\mathbf{\Sigma}^2_{n,4}$, and $\mathbf{S}^{n^4}_+$.

2.2.1 Closedness

Proposition 2.2.1. *$\mathbf{\Sigma}^4_{n,4}$, $\mathbf{S}^{n^2\times n^2}_+$, $\mathbf{\Sigma}^2_{n,4}$, and $\mathbf{S}^{n^4}_+$ are all closed convex cones.*

While $\mathbf{S}^{n^4}_+$ and $\mathbf{S}^{n^2\times n^2}_+$ are evidently closed, by a similar argument as in [12] it is also easy to see that the cone of quartic SOS forms $\mathbf{\Sigma}^2_{n,4} := \operatorname{sym}\operatorname{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}$ is closed. The closedness of $\mathbf{S}^{n^2\times n^2}_+$, $\mathbf{\Sigma}^2_{n,4}$ and $\mathbf{S}^{n^4}_+$ were also known in polynomial optimization, e.g. [50]. The closedness of the cone of quartic SOQ forms $\mathbf{\Sigma}^4_{n,4}$ was proved in Proposition 3.6 of [45] for general even degree forms. In fact, we have a slightly stronger result below:

Lemma 2.2.2. *If $\mathbf{D}\subseteq\mathbb{R}^n$ is closed, then* cone $\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$ *is closed.*

Proof. Suppose that $\mathcal{F}\in \operatorname{cl}\operatorname{cone}\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$, then there is a sequence of quartic forms $\mathcal{F}^k\in\operatorname{cone}\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$ $(k=1,2,\dots)$, such that $\mathcal{F}=\lim_{k\to\infty}\mathcal{F}^k$. Since the dimension of $\mathbf{S}^{n^4}$ is $N:=\binom{n+3}{4}$, it follows from Carathéodory's theorem that for any given $\mathcal{F}^k$, there exists an $n\times(N+1)$ matrix Z^k, such that

$$\mathcal{F}^k=\sum_{i=1}^{N+1} z^k(i)\otimes z^k(i)\otimes z^k(i)\otimes z^k(i),$$

where $z^k(i)$ is the i-th column vector of Z^k, and is a positive multiple of a vector in $\mathbf{D}$. Now define $\operatorname{tr}\mathcal{F}^k=\sum_{j=1}^n \mathcal{F}^k_{jjjj}$, then

$$\sum_{i=1}^{N+1}\sum_{j=1}^{n}(Z^k_{ji})^4=\operatorname{tr}\mathcal{F}^k\to\operatorname{tr}\mathcal{F}.$$

Thus, the sequence $\{Z^k\}$ is bounded, and have a cluster point Z^*, satisfying $\mathcal{F}=\sum_{i=1}^{N+1} z^*(i)\otimes z^*(i)\otimes z^*(i)\otimes z^*(i)$. Note that each column of Z^* is also a positive multiple of a vector in $\mathbf{D}$, it follows that $\mathcal{F}\in\operatorname{cone}\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$. □

The cone of quartic SOQ forms is closely related to the fourth moment of a multi-dimensional random variable. Given an n-dimensional random variable $\xi=(\xi_1,\dots,\xi_n)^\top$ on the support set $\mathbf{D}\subseteq\mathbb{R}^n$ with density function p, its fourth moment is a super-symmetric fourth order tensor $\mathcal{M}\in\mathbf{S}^{n^4}$, whose (i,j,k,ℓ)-th entry is

$$\mathcal{M}_{ijk\ell}=\mathsf{E}\left[\xi_i\xi_j\xi_k\xi_\ell\right]=\int_{\mathbf{D}} x_ix_jx_kx_\ell\, p(x)dx.$$

Suppose the fourth moment of ξ is finite, then by the closedness of $\mathbf{\Sigma}^4_{n,4}$, we have

$$\mathcal{M} = \mathsf{E}\left[\xi \otimes \xi \otimes \xi \otimes \xi\right] = \int_{\mathbf{D}} (x \otimes x \otimes x \otimes x)\, p(x) dx \in \text{cone}\left\{a \otimes a \otimes a \otimes a \,|\, a \in \mathbb{R}^n\right\} = \mathbf{\Sigma}^4_{n,4}.$$

Conversely, for any $\mathcal{M} \in \mathbf{\Sigma}^4_{n,4}$, it is easy to verify that there exists an n-dimensional random variable whose fourth moment is just $\mathcal{M}$. Thus, the set of all the finite fourth moments of n-dimensional random variables is exactly $\mathbf{\Sigma}^4_{n,4}$, similar to the fact that all possible covariance matrices form the cone of positive semidefinite matrices.

2.2.2 Alternative Representations

In this subsection we present some alternative forms of the same cones that we have discussed. Some of these alternative representations are more convenient to use in various applications. We first introduce a new class of tensor: a fourth order tensor $\mathcal{G} \in \mathbb{R}^{n^4}$ *strongly partial-symmetric*, if

$$\mathcal{G}_{ijk\ell} = \mathcal{G}_{jik\ell} = \mathcal{G}_{ij\ell k} = \mathcal{G}_{k\ell ij} \quad \forall\, 1 \le i, j, k, \ell \le n.$$

Essentially this means that the tensor form is symmetric for the first and the last two indices respectively, and is also symmetric by swapping the first two and the last two indices. The set of all partial-symmetric fourth order tensors in $\mathbb{R}^{n^4}$ is denoted by $\overrightarrow{\mathbf{S}}^{n^4}$. Obviously $\mathbf{S}^{n^4} \subsetneq \overrightarrow{\mathbf{S}}^{n^4} \subsetneq \mathbb{R}^{n^4}$ if $n \ge 2$.

Theorem 2.2.3. *For the quartic polynomials cones introduced, we have the following equivalent representations:*

1. *For the cone of quartic SOS forms*

$$\begin{aligned}
\mathbf{\Sigma}^2_{n,4} &:= \text{sym cone}\left\{A \otimes A \,|\, A \in \mathbf{S}^{n^2}\right\} \\
&= \text{sym}\left\{\mathcal{F} \in \overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X, X) \ge 0,\, \forall\, X \in \mathbf{S}^{n^2}\right\} \\
&= \text{sym}\left\{\mathcal{F} \in \mathbb{R}^{n^4} \,|\, \mathcal{F}(X, X) \ge 0,\, \forall\, X \in \mathbf{S}^{n^2}\right\};
\end{aligned}$$

2. *For the cone of quartic matrix PSD forms*

$$\begin{aligned}
\mathbf{S}^{n^2 \times n^2}_+ &:= \left\{\mathcal{F} \in \mathbf{S}^{n^4} \,|\, \mathcal{F}(X, X) \ge 0,\, \forall\, X \in \mathbb{R}^{n^2}\right\} \\
&= \left\{\mathcal{F} \in \mathbf{S}^{n^4} \,|\, \mathcal{F}(X, X) \ge 0,\, \forall\, X \in \mathbf{S}^{n^2}\right\} \\
&= \mathbf{S}^{n^4} \bigcap \text{cone}\left\{A \otimes A \,|\, A \in \mathbf{S}^{n^2}\right\};
\end{aligned}$$

3. For the cone of quartic SOQ forms

$$\mathbf{\Sigma}^4_{n,4} := \text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbb{R}^n\} = \text{sym cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}_+\right\}.$$

The remaining of this subsection is devoted to the proof of Theorem 2.2.3.

Let us first study the equivalent representations for $\mathbf{\Sigma}^2_{n,4}$ and $\mathbf{S}^{n^2\times n^2}_+$. To verify a quartic matrix PSD form, we should check the operations of quartic forms on matrices. In fact, the quartic matrix PSD forms can be extended to the space of strongly partial-symmetric tensors $\overrightarrow{\mathbf{S}}^{n^4}$. It is not hard to verify that for any $\mathcal{F}\in\overrightarrow{\mathbf{S}}^{n^4}$, it holds that

$$\mathcal{F}(X,Y)=\mathcal{F}(X^\top,Y)=\mathcal{F}(X,Y^\top)=\mathcal{F}(Y,X)\quad\forall\, X,Y\in\mathbb{R}^{n^2}, \tag{2.4}$$

which implies that $\mathcal{F}(X,Y)$ is invariant under the transpose operation as well as the operation to swap the X and Y matrices. Indeed, it is easy to see that the partial-symmetry of $\mathcal{F}$ is a necessary and sufficient condition for (2.4) to hold. We have the following property for quartic matrix PSD forms in $\overrightarrow{\mathbf{S}}^{n^4}$, similar to that for $\mathbf{S}^{n^2\times n^2}_+$ in Theorem 2.2.3.

Lemma 2.2.4. *For strongly partial-symmetric four order tensors, it holds that*

$$\begin{aligned}
\overrightarrow{\mathbf{S}}^{n^2\times n^2}_+ &:= \left\{\mathcal{F}\in\overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbb{R}^{n^2}\right\}\\
&= \left\{\mathcal{F}\in\overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbf{S}^{n^2}\right\} \qquad (2.5)\\
&= \text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}. \qquad (2.6)
\end{aligned}$$

Proof. Observe that for any skew-symmetric $Y\in\mathbb{R}^{n^2}$, i.e., $Y^\top=-Y$, we have

$$\mathcal{F}(X,Y)=-\mathcal{F}(X,-Y)=-\mathcal{F}(X,Y^\top)=-\mathcal{F}(X,Y)\quad\forall\, X\in\mathbb{R}^{n^2},$$

which implies that $\mathcal{F}(X,Y)=0$. As any square matrix can be written as the sum of a symmetric matrix and a skew-symmetric matrix, say for $Z\in\mathbb{R}^{n^2}$, by letting $X=(Z+Z^\top)/2$ which is symmetric, and $Y=(Z-Z^\top)/2$ which is skew-symmetric, we have $Z=X+Y$. Therefore,

$$\mathcal{F}(Z,Z)=\mathcal{F}(X+Y,X+Y)=\mathcal{F}(X,X)+2\mathcal{F}(X,Y)+\mathcal{F}(Y,Y)=\mathcal{F}(X,X).$$

This implies the equivalence between $\mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbb{R}^{n^2}$ and $\mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbf{S}^{n^2}$, which proves (2.5).

To prove (2.6), first note that cone $\left\{A \otimes A \,|\, A \in \mathbf{S}^{n^2}\right\} \subseteq \{\mathcal{F} \in \overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X) \geq 0, \forall\, X \in \mathbb{R}^{n^2}\}$. Conversely, given any $\mathcal{G} \in \overrightarrow{\mathbf{S}}^{n^4}$ with $\mathcal{G}(X,X) \geq 0, \forall\, X \in \mathbb{R}^{n^2}$, we may rewrite $\mathcal{G}$ as an $n^2 \times n^2$ symmetric matrix $M_{\mathcal{G}}$. Therefore

$$(\mathrm{vec}\,(X))^\top M_{\mathcal{G}}\,\mathrm{vec}\,(X) = \mathcal{G}(X,X) \geq 0 \quad \forall\, X \in \mathbb{R}^{n^2},$$

which implies that $M_{\mathcal{G}}$ is positive semidefinite. Let $M_{\mathcal{G}} = \sum_{i=1}^m z^i (z^i)^\top$, where

$$z^i = \left(z^i_{11}, \ldots, z^i_{1n}, \ldots, z^i_{n1}, \ldots, z^i_{nn}\right)^\top \quad \forall\, 1 \leq i \leq m.$$

Note that for any $1 \leq k, \ell \leq n$, $\mathcal{G}_{k\ell\ell k} = \sum_{i=1}^m z^i_{k\ell} z^i_{\ell k}$, $\mathcal{G}_{k\ell k\ell} = \sum_{i=1}^m (z^i_{k\ell})^2$ and $\mathcal{G}_{\ell k\ell k} = \sum_{i=1}^m (z^i_{\ell k})^2$, as well as $\mathcal{G}_{k\ell\ell k} = \mathcal{G}_{k\ell k\ell} = \mathcal{G}_{\ell k\ell k}$ by partial-symmetry of $\mathcal{G}$. We have

$$\sum_{i=1}^m (z^i_{k\ell} - z^i_{\ell k})^2 = \sum_{i=1}^m (z^i_{k\ell})^2 + \sum_{i=1}^m (z^i_{\ell k})^2 - 2\sum_{i=1}^m z^i_{k\ell} z^i_{\ell k} = \mathcal{G}_{k\ell k\ell} + \mathcal{G}_{\ell k\ell k} - 2\mathcal{G}_{k\ell\ell k} = 0,$$

which implies that $z^i_{k\ell} = z^i_{\ell k}$ for any $1 \leq k, \ell \leq n$. Therefore, we may construct a symmetric matrix $Z^i \in \mathbf{S}^{n^2}$, such that $\mathrm{vec}\,(Z^i) = z^i$ for all $1 \leq i \leq m$. We have $\mathcal{G} = \sum_{i=1}^m Z^i \otimes Z^i$, and so (2.6) is proven. □

For the first part of Theorem 2.2.3, the first identity follows from (2.6) by applying the symmetrization operation on both sides. The second identity is quite obvious. Essentially, for any $\mathcal{F} \in \mathbb{R}^{n^4}$, we may make it being strongly partial-symmetric by averaging the corresponding entries, to be denoted by $\mathcal{F}_0 \in \overrightarrow{\mathbf{S}}^{n^4}$. It is easy to see that $\mathcal{F}_0(X,X) = \mathcal{F}(X,X)$ for all $X \in \mathbf{S}^{n^2}$ since $X \otimes X \in \overrightarrow{\mathbf{S}}^{n^4}$, which implies that $\left\{\mathcal{F} \in \mathbb{R}^{n^4} \,|\, \mathcal{F}(X,X) \geq 0, \forall\, X \in \mathbf{S}^{n^2}\right\} \subseteq \left\{\mathcal{F} \in \overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X) \geq 0, \forall\, X \in \mathbf{S}^{n^2}\right\}$. The reverse inclusion is trivial.

For the second part of Theorem 2.2.3, it follows from (2.5) and (2.6) by restricting to $\mathbf{S}^{n^4}$. Let us now turn to proving the last part of Theorem 2.2.3, which is an alternative representation of the quartic SOQ forms. Obviously we need only to show that

$$\mathrm{sym\,cone}\left\{A \otimes A \,|\, A \in \mathbf{S}^{n^2}_+\right\} \subseteq \mathrm{cone}\left\{a \otimes a \otimes a \otimes a \,|\, a \in \mathbb{R}^n\right\}.$$

Since there is a one-to-one mapping from quartic forms to fourth order super-symmetric tensors, it suffices to show that for any $A \in \mathbf{S}^{n^2}_+$, the function $(x^\top A x)^2$ can be written as a form of $\sum_{i=1}^m (x^\top a^i)^4$ for some $a^1, \ldots, a^m \in \mathbb{R}^n$. Note that the so-called Hilbert's identity (see e.g. Barvinok [18]) asserts the following:

For any fixed positive integers d and n, there always exist m real vectors $a^1, \ldots, a^m \in \mathbb{R}^n$ such that $(x^\top x)^d = \sum_{i=1}^m (x^\top a^i)^{2d}$.

In fact, when $d = 2$, we shall propose a polynomial-time algorithm to find the aforementioned representations, in Chapter 3, where the number m is bounded by a polynomial of n, although in the original version of Hilbert's identity m is exponential in n. Since we have $A \in \mathbf{S}_+^{n^2}$, replacing x by $A^{1/2}y$ in Hilbert's identity when $d = 2$, one gets $(y^\top Ay)^2 = \sum_{i=1}^m (y^\top A^{1/2} a^i)^4$. The desired decomposition follows, and this proves the last part of Theorem 2.2.3.

2.2.3 Duality

In this subsection, we shall discuss the duality relationships among these four cones of quartic forms. Note that $\mathbf{S}^{n^4}$ is the ground vector space within which the duality is defined, unless otherwise specified.

Theorem 2.2.5. *The cone of quartic PSD forms and the cone of quartic SOQ forms are primal-dual pair, i.e.,* $\mathbf{\Sigma}_{n,4}^4 = \left(\mathbf{S}_+^{n^4}\right)^*$ *and* $\mathbf{S}_+^{n^4} = \left(\mathbf{\Sigma}_{n,4}^4\right)^*$. *The cone of quartic SOS forms and the cone of quartic matrix PSD forms are primal-dual pair, i.e.,* $\mathbf{S}_+^{n^2 \times n^2} = \left(\mathbf{\Sigma}_{n,4}^2\right)^*$ *and* $\mathbf{\Sigma}_{n,4}^2 = \left(\mathbf{S}_+^{n^2 \times n^2}\right)^*$.

Remark that the primal-dual relationship between $\mathbf{\Sigma}_{n,4}^4$ and $\mathbf{S}_+^{n^4}$ was already proved in Theorem 3.7 of [45] for general even degree forms. The primal-dual relationship between $\mathbf{S}_+^{n^2 \times n^2}$ and $\mathbf{\Sigma}_{n,4}^2$ was also mentioned in Theorem 3.16 of [45] for general even degree forms. Here we give the proof in the language of quartic tensors. Let us start by discussing the primal-dual pair $\mathbf{\Sigma}_{n,4}^4$ and $\mathbf{S}_+^{n^4}$. In Proposition 1 of [12], Sturm and Zhang proved that for the quadratic forms, $\{A \in \mathbf{S}^{n^2} \,|\, x^\top Ax \geq 0, \forall\, x \in \mathbf{D}\}$ and cone $\{aa^\top \,|\, a \in \mathbf{D}\}$ are a primal-dual pair for any closed $\mathbf{D} \subseteq \mathbb{R}^n$. We observe that a similar structure holds for the quartic forms as well. The first part of Theorem 2.2.5 then follows from next lemma.

Lemma 2.2.6. *If* $\mathbf{D} \subseteq \mathbb{R}^n$ *is closed, then* $\mathbf{S}_+^{n^4}(\mathbf{D}) := \{\mathcal{F} \in \mathbf{S}^{n^4} \,|\, \mathcal{F}(x,x,x,x) \geq 0, \forall\, x \in \mathbf{D}\}$ *and* cone $\{a \otimes a \otimes a \otimes a \,|\, a \in \mathbf{D}\}$ *are a primal-dual pair, i.e.,*

$$\mathbf{S}_+^{n^4}(\mathbf{D}) = (\text{cone}\,\{a \otimes a \otimes a \otimes a \,|\, a \in \mathbf{D}\})^* \tag{2.7}$$

and

$$\left(\mathbf{S}_+^{n^4}(\mathbf{D})\right)^* = \text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}.$$

Proof. Since $\text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$ is closed by Lemma 2.2.2, we only need to show (2.7). In fact, if $\mathcal{F}\in\mathbf{S}_+^{n^4}(\mathbf{D})$, then $\mathcal{F}\bullet(a\otimes a\otimes a\otimes a)=\mathcal{F}(a,a,a,a)\geq 0$ for all $a\in\mathbf{D}$. Thus $\mathcal{F}\bullet\mathcal{G}\geq 0$ for all $\mathcal{G}\in\text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$, which implies that $\mathcal{F}\in(\text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\})^*$. Conversely, if $\mathcal{F}\in(\text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\})^*$, then $\mathcal{F}\bullet\mathcal{G}\geq 0$ for all $\mathcal{G}\in\text{cone}\,\{a\otimes a\otimes a\otimes a \,|\, a\in\mathbf{D}\}$. In particular, by letting $\mathcal{G}=x\otimes x\otimes x\otimes x$, we have $\mathcal{F}(x,x,x,x)=\mathcal{F}\bullet(x\otimes x\otimes x\otimes x)\geq 0$ for all $x\in\mathbf{D}$, which implies that $\mathcal{F}\in\mathbf{S}_+^{n^4}(\mathbf{D})$. □

Let us turn to the primal-dual pair of $\mathbf{S}_+^{n^2\times n^2}$ and $\mathbf{\Sigma}_{n,4}^2$. For technical reasons, we shall momentarily lift the ground space from $\mathbf{S}^{n^4}$ to the space of strongly partial-symmetric tensors $\overrightarrow{\mathbf{S}}^{n^4}$. This enlarges all the dual objects. To distinguish these two dual objects, let us use the notation '$\mathbf{K}^{\overrightarrow{*}}$' to indicate the dual of convex cone $\mathbf{K}\in\mathbf{S}^{n^4}\subseteq\overrightarrow{\mathbf{S}}^{n^4}$ generated in the space $\overrightarrow{\mathbf{S}}^{n^4}$, while '$\mathbf{K}^*$' is the dual of $\mathbf{K}$ generated in the space $\mathbf{S}^{n^4}$.

Lemma 2.2.7. *For strongly partial-symmetric tensors, the cone* $\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}$ *is self-dual with respect to the space* $\overrightarrow{\mathbf{S}}^{n^4}$*, i.e.,* $\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}=\left(\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}\right)^{\overrightarrow{*}}$.

Proof. According to Proposition 1 of [12] and the partial-symmetry of $\overrightarrow{\mathbf{S}}^{n^4}$, we have

$$\left(\text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}\right)^{\overrightarrow{*}} = \left\{\mathcal{F}\in\overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbf{S}^{n^2}\right\}.$$

By Lemma 2.2.4, we have

$$\overrightarrow{\mathbf{S}}_+^{n^2\times n^2} = \left\{\mathcal{F}\in\overrightarrow{\mathbf{S}}^{n^4} \,|\, \mathcal{F}(X,X)\geq 0,\,\forall\, X\in\mathbf{S}^{n^2}\right\} = \text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}.$$

Thus $\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}$ is self-dual with respect to the space $\overrightarrow{\mathbf{S}}^{n^4}$. □

Notice that by definition and Lemma 2.2.7, we have

$$\mathbf{\Sigma}_{n,4}^2 = \text{sym}\,\text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\} = \text{sym}\,\overrightarrow{\mathbf{S}}_+^{n^2\times n^2} = \text{sym}\left(\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}\right)^{\overrightarrow{*}},$$

and by the alternative representation in Theorem 2.2.3 we have

$$\mathbf{S}_+^{n^2\times n^2} = \mathbf{S}^{n^4}\bigcap\text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\} = \mathbf{S}^{n^4}\bigcap\overrightarrow{\mathbf{S}}_+^{n^2\times n^2}.$$

Therefore the duality between $\mathbf{S}_+^{n^2\times n^2}$ and $\mathbf{\Sigma}_{n,4}^2$ follows immediately from the following lemma.

Lemma 2.2.8. *If $\mathbf{K} \subseteq \overrightarrow{\mathbf{S}}^{n^4}$ is a closed convex cone and $\mathbf{K}^{\overrightarrow{*}}$ is its dual with respect to the space $\overrightarrow{\mathbf{S}}^{n^4}$, then $\mathbf{K} \bigcap \mathbf{S}^{n^4}$ and $\operatorname{sym}\mathbf{K}^{\overrightarrow{*}}$ are a primal-dual pair with respect to the space $\mathbf{S}^{n^4}$, i.e., $\left(\mathbf{K} \bigcap \mathbf{S}^{n^4}\right)^* = \operatorname{sym}\mathbf{K}^{\overrightarrow{*}}$ and $\mathbf{K} \bigcap \mathbf{S}^{n^4} = \left(\operatorname{sym}\mathbf{K}^{\overrightarrow{*}}\right)^*$.*

Proof. For any $\mathcal{G} \in \operatorname{sym}\mathbf{K}^{\overrightarrow{*}} \subseteq \mathbf{S}^{n^4}$, there is a $\mathcal{G}' \in \mathbf{K}^{\overrightarrow{*}} \subseteq \overrightarrow{\mathbf{S}}^{n^4}$, such that $\mathcal{G} = \operatorname{sym}\mathcal{G}' \in \mathbf{S}^{n^4}$. We then have $\mathcal{G}_{ijk\ell} = \frac{1}{3}(\mathcal{G}'_{ijk\ell} + \mathcal{G}'_{ikj\ell} + \mathcal{G}'_{i\ell jk})$. Thus for any $\mathcal{F} \in \mathbf{K} \bigcap \mathbf{S}^{n^4} \subseteq \mathbf{S}^{n^4}$, it follows that

$$\begin{aligned}
\mathcal{F} \bullet \mathcal{G} &= \sum_{1\le i,j,k,\ell\le n} \frac{\mathcal{F}_{ijk\ell}(\mathcal{G}'_{ijk\ell} + \mathcal{G}'_{ikj\ell} + \mathcal{G}'_{i\ell jk})}{3} \\
&= \sum_{1\le i,j,k,\ell\le n} \frac{\mathcal{F}_{ijk\ell}\mathcal{G}'_{ijk\ell} + \mathcal{F}_{ikj\ell}\mathcal{G}'_{ikj\ell} + \mathcal{F}_{i\ell jk}\mathcal{G}'_{i\ell jk}}{3} \\
&= \mathcal{F} \bullet \mathcal{G}' \ge 0.
\end{aligned}$$

Therefore $\mathcal{G} \in \left(\mathbf{K} \bigcap \mathbf{S}^{n^4}\right)^*$, implying that $\operatorname{sym}\mathbf{K}^{\overrightarrow{*}} \subseteq \left(\mathbf{K} \bigcap \mathbf{S}^{n^4}\right)^*$.

Moreover, if $\mathcal{F} \in \left(\operatorname{sym}\mathbf{K}^{\overrightarrow{*}}\right)^* \subseteq \mathbf{S}^{n^4}$, then for any $\mathcal{G}' \in \mathbf{K}^{\overrightarrow{*}} \subseteq \overrightarrow{\mathbf{S}}^{n^4}$, we have $\mathcal{G} = \operatorname{sym}\mathcal{G}' \in \operatorname{sym}\mathbf{K}^{\overrightarrow{*}}$, and $\mathcal{G}' \bullet \mathcal{F} = \mathcal{G} \bullet \mathcal{F} \ge 0$. Therefore $\mathcal{F} \in \left(\mathbf{K}^{\overrightarrow{*}}\right)^{\overrightarrow{*}} = \operatorname{cl}\mathbf{K} = \mathbf{K}$, which implies that $\left(\operatorname{sym}\mathbf{K}^{\overrightarrow{*}}\right)^* \subseteq \left(\mathbf{K} \bigcap \mathbf{S}^{n^4}\right)$. Finally, the duality relationship holds by the bipolar theorem and the closedness of these cones. □

2.2.4 The Hierarchical Structure

The last part of this section is to present a hierarchy among these four cones of quartic forms. The main result is summarized in the theorem below.

Theorem 2.2.9. *If $n \ge 4$, then*

$$\mathbf{\Sigma}_{n,4}^4 \subsetneq \mathbf{S}_+^{n^2\times n^2} \subsetneq \mathbf{\Sigma}_{n,4}^2 \subsetneq \mathbf{S}_+^{n^4}.$$

For the low dimension cases ($n \le 3$), we shall present it in Section 2.4.2. Evidently a quartic SOS form is quartic PSD, implying $\mathbf{\Sigma}_{n,4}^2 \subseteq \mathbf{S}_+^{n^4}$. By invoking the duality operation and using Theorem 2.2.5 we have $\mathbf{\Sigma}_{n,4}^4 \subseteq \mathbf{S}_+^{n^2\times n^2}$, while by the alternative

representation in Theorem 2.2.3 we have $\mathbf{S}_+^{n^2\times n^2} = \mathbf{S}^{n^4} \bigcap \text{cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}$, and by the very definition we have $\mathbf{\Sigma}_{n,4}^2 = \text{sym cone}\left\{A\otimes A \,|\, A\in\mathbf{S}^{n^2}\right\}$. Therefore $\mathbf{S}_+^{n^2\times n^2} \subseteq \mathbf{\Sigma}_{n,4}^2$. Finally, the strict containing relationship is a result of the following examples.

Example 2.2.10. *(Quartic forms in $\mathbf{S}_+^{n^4} \setminus \mathbf{\Sigma}_{n,4}^2$ when $n = 4$). Let $g_1(x) = {x_1}^2(x_1 - x_4)^2 + {x_2}^2(x_2-x_4)^2 + {x_3}^2(x_3-x_4)^2 + 2x_1x_2x_3(x_1+x_2+x_3-2x_4)$ and $g_2(x) = {x_1}^2{x_2}^2 + {x_2}^2{x_3}^2 + {x_3}^2{x_1}^2 + {x_4}^4 - 4x_1x_2x_3x_4$, then both $g_1(x)$ and $g_2(x)$ are in $\mathbf{S}_+^{4^4} \setminus \mathbf{\Sigma}_{4,4}^2$. We refer the interested readers to [53] for more information on these examples and the study on these two cones.*

Example 2.2.11. *(A quartic form in $\mathbf{S}_+^{n^2\times n^2} \setminus \mathbf{\Sigma}_{n,4}^4$ when $n = 4$). Construct $\mathcal{F} \in \mathbf{S}^{4^4}$, whose only nonzero entries (taking into account the super-symmetry) are $\mathcal{F}_{1122} = 4$, $\mathcal{F}_{1133} = 4$, $\mathcal{F}_{2233} = 4$, $\mathcal{F}_{1144} = 9$, $\mathcal{F}_{2244} = 9$, $\mathcal{F}_{3344} = 9$, $\mathcal{F}_{1234} = 6$, $\mathcal{F}_{1111} = 29$, $\mathcal{F}_{2222} = 29$, $\mathcal{F}_{3333} = 29$, and $\mathcal{F}_{4444} = 3 + \frac{25}{7}$. One may verify straightforwardly that $\mathcal{F}$ can be decomposed as $\sum_{i=1}^7 A^i \otimes A^i$, with $A^1 = \begin{bmatrix} \sqrt{7} & 0 & 0 & 0 \\ 0 & \sqrt{7} & 0 & 0 \\ 0 & 0 & \sqrt{7} & 0 \\ 0 & 0 & 0 & \frac{5}{\sqrt{7}} \end{bmatrix}$,*

$$A^2 = \begin{bmatrix} 0 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 3 & 0 \end{bmatrix}, \quad A^3 = \begin{bmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \\ 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{bmatrix}, \quad A^4 = \begin{bmatrix} 0 & 0 & 0 & 3 \\ 0 & 0 & 2 & 0 \\ 0 & 2 & 0 & 0 \\ 3 & 0 & 0 & 0 \end{bmatrix},$$

$$A^5 = \begin{bmatrix} -2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A^6 = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \text{and} \quad A^7 = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

According to Theorem 2.2.3, we have $\mathcal{F} \in \mathbf{S}_+^{4^2\times 4^2}$. Recall $g_2(x)$ in Example 2.2.10, which is quartic PSD. Denote $\mathcal{G}$ to be the super-symmetric tensor associated with $g_2(x)$, thus $\mathcal{G} \in \mathbf{S}_+^{4^4}$. One computes that $\mathcal{G} \bullet \mathcal{F} = 4 + 4 + 4 + 3 + \frac{25}{7} - 24 < 0$. By the duality result as stipulated in Theorem 2.2.5, we conclude that $\mathcal{F} \notin \mathbf{\Sigma}_{4,4}^4$.

Example 2.2.12. *(A quartic form in $\mathbf{\Sigma}_{n,4}^2 \setminus \mathbf{S}_+^{n^2\times n^2}$ when $n = 3$). Let $g_3(x) = 2{x_1}^4 + 2{x_2}^4 + \frac{1}{2}{x_3}^4 + 6{x_1}^2{x_3}^2 + 6{x_2}^2{x_3}^2 + 6{x_1}^2{x_2}^2$, which is obviously quartic SOS. Now recycle*

the notation and denote $\mathcal{G} \in \mathbf{\Sigma}^2_{3,4}$ *to be the super-symmetric tensor associated with* $g_3(x)$*, and we have* $\mathcal{G}_{1111} = 2$*,* $\mathcal{G}_{2222} = 2$*,* $\mathcal{G}_{3333} = \frac{1}{2}$*,* $\mathcal{G}_{1122} = 1$*,* $\mathcal{G}_{1133} = 1$*, and* $\mathcal{G}_{2233} = 1$*. If we let* $X = \mathrm{Diag}\,(1, 1, -4) \in \mathbf{S}^{3^2}$*, then*

$$\begin{aligned}\mathcal{G}(X,X) = \sum_{1\le i,j,k,\ell\le 3} \mathcal{G}_{ijk\ell}X_{ij}X_{k\ell} = \sum_{1\le i,k\le 3} \mathcal{G}_{iikk}X_{ii}X_{kk} &= \begin{pmatrix} 1 \\ 1 \\ -4 \end{pmatrix}^\top \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & \frac{1}{2} \end{bmatrix} \begin{pmatrix} 1 \\ 1 \\ -4 \end{pmatrix} \\ &= -2,\end{aligned}$$

implying that $\mathcal{G} \notin \mathbf{S}^{3^2\times 3^2}_+$*.*

2.3 Cones Related to Convex Quartic Forms

In this section we shall study the cone of quartic sos-convex forms $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$, and the cone of quartic forms which are both SOS and convex $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\mathrm{cvx}}$. The aim is to incorporate these two new cones into the hierarchical structure as depicted in Theorem 2.2.9.

Theorem 2.3.1. *If* $n \ge 4$*, then*

$$\mathbf{\Sigma}^4_{n,4} \subsetneq \mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subseteq \left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\mathrm{cvx}}\right) \subsetneq \mathbf{\Sigma}^2_{n,4} \subsetneq \mathbf{S}^{n^4}_+.$$

As mentioned in Section 2.1.2, an sos-convex homogeneous quartic polynomial function is both SOS and convex (see also [43]), which implies that $\mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subseteq \left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\mathrm{cvx}}\right)$. Moreover, the following example shows that a quartic SOS form is not necessarily convex.

Example 2.3.2. *(A quartic form in* $\mathbf{\Sigma}^2_{n,4} \backslash \mathbf{S}^{n^4}_{\mathrm{cvx}}$ *when* $n = 2$*). Let* $g_4(x) = (x^\top Ax)^2$ *with* $A \in \mathbf{S}^{n^2}$*, and its Hessian matrix is* $\nabla^2 g_4(x) = 8Axx^\top A + 4x^\top AxA$*. In particular, by letting* $A = \begin{bmatrix} -3 & 0 \\ 0 & 1 \end{bmatrix}$ *and* $x = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$*, we have* $\nabla^2 f(x) = \begin{bmatrix} 0 & 0 \\ 0 & 8 \end{bmatrix} + \begin{bmatrix} -12 & 0 \\ 0 & 4 \end{bmatrix} \not\succeq 0$*, implying that* $g_4(x)$ *is not convex.*

The above example suggests that $\left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\mathrm{cvx}}\right) \subsetneq \mathbf{\Sigma}^2_{n,4}$ when $n \ge 2$. Next we shall prove the assertion that $\mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ when $n \ge 3$. To this end, let us first quote a result on the sos-convex functions due to Ahmadi and Parrilo [43]:

If $f(x)$ is a polynomial with its Hessian matrix being $\nabla^2 f(x)$, then $f(x)$ is sos-convex if and only if $y^\top \nabla^2 f(x) y$ is a sum of squares in (x, y).

For a quartic form $\mathcal{F}(x, x, x, x)$, its Hessian matrix is $12\mathcal{F}(x, x, \cdot, \cdot)$. Therefore, $\mathcal{F}$ is quartic sos-convex if and only if $\mathcal{F}(x, x, y, y)$ is is a sum of squares in (x, y). Now if $\mathcal{F} \in \mathbf{S}_+^{n^2 \times n^2}$, then by Theorem 2.2.3 we may find matrices $A^1, \ldots, A^m \in \mathbf{S}^{n^2}$ such that $\mathcal{F} = \sum_{t=1}^m A^t \otimes A^t$. We have

$$\begin{aligned} \mathcal{F}(x, x, y, y) = \mathcal{F}(x, y, x, y) &= \sum_{t=1}^m \sum_{1 \le i,j,k,\ell \le n} x_i y_j x_k y_\ell A^t_{ij} A^t_{k\ell} \\ = \sum_{t=1}^m \left(\sum_{1 \le i,j \le n} x_i y_j A^t_{ij} \right) \left(\sum_{1 \le k,\ell \le n} x_k y_\ell A^t_{k\ell} \right) &= \sum_{t=1}^m \left(x^\top A^t y \right)^2, \end{aligned}$$

which is a sum of squares in (x, y), hence sos-convex. This proves $\mathbf{S}_+^{n^2 \times n^2} \subseteq \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$, and the example below rules out the equality when $n \ge 3$.

Example 2.3.3. *(A quartic form in $\mathbf{\Sigma}^2_{\nabla^2_{n,4}} \setminus \mathbf{S}_+^{n^2 \times n^2}$ when $n = 3$). Recall $g_3(x) = 2{x_1}^4 + 2{x_2}^4 + \frac{1}{2}{x_3}^4 + 6{x_1}^2{x_3}^2 + 6{x_2}^2{x_3}^2 + 6{x_1}^2{x_2}^2$ in Example 2.2.12, which is shown not to be quartic matrix PSD. Moreover, it is straightforward to compute that*

$$\begin{aligned} \nabla^2 g_3(x) = 24 \begin{pmatrix} x_1 \\ x_2 \\ \frac{x_3}{2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \frac{x_3}{2} \end{pmatrix}^\top + 12 \begin{pmatrix} 0 \\ x_3 \\ x_2 \end{pmatrix} \begin{pmatrix} 0 \\ x_3 \\ x_2 \end{pmatrix}^\top &+ 12 \begin{pmatrix} x_3 \\ 0 \\ x_1 \end{pmatrix} \begin{pmatrix} x_3 \\ 0 \\ x_1 \end{pmatrix}^\top \\ &+ 12 \begin{bmatrix} {x_2}^2 & 0 & 0 \\ 0 & {x_1}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \succeq 0, \end{aligned}$$

which implies that $g_3(x)$ is quartic sos-convex.

A natural question regarding to the hierarchical structure in Theorem 2.3.1 is whether $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}} = \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ or not. In fact, the relationship between convex, sos-convex, and SOS is a highly interesting subject which attracted many speculations in the literature. Ahmadi and Parrilo [43] gave an explicit example with a three dimensional homogeneous form of degree eight, which they showed to be both convex and SOS but not sos-convex. However, for quartic polynomials (degree four), such an explicit instant in

$\left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}\right) \setminus \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ is not in sight. Notwithstanding the difficulty in constructing an explicit quartic example, on the premise that $P \neq NP$ in Section 2.4 we will show that $\left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}\right) \neq \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$. With that piece of information, the chain of containing relations manifested in Theorem 2.3.1 is complete, under the assumption that $P \neq NP$. However, the following open question remains:

Question 2.3.1. *Find an explicit instant in* $\left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\text{cvx}}\right) \setminus \mathbf{\Sigma}^2_{\nabla^2_{n,4}}$*: a quartic form that is both SOS and convex, but not sos-convex.*

The two newly introduced cones in this section are related to the convexity properties. Some more words on convex quartic forms are in order here. As mentioned in Section 2.1.2, for a quartic form $\mathcal{F} \in \mathbf{S}^{n^4}_{\text{cvx}}$, its Hessian matrix is $12\mathcal{F}(x, x, \cdot, \cdot)$. Therefore, $\mathcal{F}$ is convex if and only if $\mathcal{F}(x, x, \cdot, \cdot) \succeq 0$ for all $x \in \mathbb{R}^n$, which is equivalent to $\mathcal{F}(x, x, y, y) \geq 0$ for all $x, y \in \mathbb{R}^n$. In fact, it is also equivalent to $\mathcal{F}(X, Y) \geq 0$ for all $X, Y \in \mathbf{S}^{n^2}_+$. To see why, we first decompose the positive semidefinite matrices X and Y, and let $X = \sum_{i=1}^n x^i (x^i)^\top$ and $Y = \sum_{j=1}^n y^j (y^j)^\top$ (see e.g. Sturm and Zhang [12]). Then

$$\begin{aligned} \mathcal{F}(X, Y) &= \mathcal{F}\left(\sum_{i=1}^n x^i (x^i)^\top, \sum_{j=1}^n y^j (y^j)^\top\right) \\ &= \sum_{1 \leq i,j \leq n} \mathcal{F}\left(x^i (x^i)^\top, y^j (y^j)^\top\right) \\ &= \sum_{1 \leq i,j \leq n} \mathcal{F}\left(x^i, x^i, y^j, y^j\right) \geq 0, \end{aligned}$$

if $\mathcal{F}(x, x, y, y) \geq 0$ for all $x, y \in \mathbb{R}^n$. Note that the converse is trivial, as it reduces to let X and Y be rank-one positive semidefinite matrices. Thus we have the following equivalence for the quartic convex forms.

Proposition 2.3.4. *For a given quartic form* $\mathcal{F} \in \mathbf{S}^{n^4}$*, the following statements are equivalent:*

- $\mathcal{F}(x, x, x, x)$ *is convex;*
- $\mathcal{F}(x, x, \cdot, \cdot)$ *is positive semidefinite for all* $x \in \mathbb{R}^n$*;*
- $\mathcal{F}(x, x, y, y) \geq 0$ *for all* $x, y \in \mathbb{R}^n$*;*

- $\mathcal{F}(X,Y) \geq 0$ *for all* $X, Y \in \mathbf{S}_+^{n^2}$.

For the relationship between the cone of quartic convex forms and the cone of quartic SOS forms, Example 2.3.2 has ruled out the possibility that $\mathbf{\Sigma}_{n,4}^2 \subseteq \mathbf{S}_{\rm cvx}^{n^4}$, while Blekherman [44] proved that $\mathbf{\Sigma}_{n,4}^2$ is not contained in $\mathbf{S}_{\rm cvx}^{n^4}$. Therefore these two cones are indeed distinctive. According to Blekherman [44], the cone of quartic convex forms is actually much bigger than the cone of quartic SOS forms when n is sufficiently large. However, at this point we are not aware of any explicit instant belonging to $\mathbf{S}_{\rm cvx}^{n^4} \setminus \mathbf{\Sigma}_{n,4}^2$. In fact, according to a recent working paper of Ahmadi et al. [54], this kind of instants exist only when $n \geq 4$. Anyway, the following challenge remains:

Question 2.3.2. *Find an explicit instant in* $\mathbf{S}_{\rm cvx}^{n^4} \setminus \mathbf{\Sigma}_{n,4}^2$*: a quartic convex form that is not quartic SOS.*

2.4 Complexities, Low-Dimensional Cases, and the Interiors of the Quartic Cones

In this section, we study the computational complexity issues for the membership queries regarding these cones of quartic forms. Unlike their quadratic counterparts where the positive semidefniteness can be checked in polynomial-time, the case for the quartic cones are substantially subtler. We also study the low dimension cases of these cones, as a complement to the result of hirarchary relationship in Theorem 2.3.1. Finally, the interiors for some quartic cones are studied.

2.4.1 Complexity

Let us start with easy cases. It is well known that deciding whether a general polynomial function is SOS can be done in polynomial-time, by resorting to checking the feasibility of an SDP. Therefore, the membership query for $\mathbf{\Sigma}_{n,4}^2$ can be done in polynomial-time. By the duality relationship claimed in Theorem 2.2.5, the membership query for $\mathbf{S}_+^{n^2 \times n^2}$ can also be done in polynomial-time. In fact, for any quartic form $\mathcal{F} \in \mathbf{S}^{n^4}$, we may rewrite $\mathcal{F}$ as an $n^2 \times n^2$ matrix, to be denoted by $M_{\mathcal{F}}$, and then Theorem 2.2.3 assures that $\mathcal{F} \in \mathbf{S}_+^{n^2 \times n^2}$ if and only if $M_{\mathcal{F}}$ is positive semedefinite, which can be checked in polynomial-time. Moreover, as discussed in Section 2.3, a quartic form $\mathcal{F}$ is sos-convex

if and only if $y^\top \left(\nabla^2 \mathcal{F}(x,x,x,x)\right) y = 12\mathcal{F}(x,x,y,y)$ is a sum of squares in (x,y), which can also be checked in polynomial-time. Therefore, the membership checking problem for $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ can be done in polynomial-time as well. Summarizing, we have:

Proposition 2.4.1. *Whether a quartic form belongs to $\mathbf{\Sigma}^2_{n,4}$, $\mathbf{S}^{n^2\times n^2}_+$, or $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$, can be verified in polynomial-time.*

Unfortunately, the membership checking problems for all the other cones that we have discussed so far are difficult. To see why, let us introduce a famous cone of quadratic functions: the copositive cone

$$\mathbf{C} := \left\{ A \in \mathbf{S}^{n^2} \mid x^\top A x \geq 0, \, \forall\, x \in \mathbb{R}^n_+ \right\},$$

whose membership query is known to be co-NP-complete. The dual of the copositive cone is the cone of completely positive matrices, defined as

$$\mathbf{C}^* := \text{cone} \left\{ x x^\top \mid x \in \mathbb{R}^n_+ \right\}.$$

Recently, Dickinson and Gijben [55] gave a formal proof for the NP-completeness of the membership problem for $\mathbf{C}^*$.

Proposition 2.4.2. *It is co-NP-complete to check if a quartic form belongs to $\mathbf{S}^{n^4}_+$ (the cone of quartic PSD forms).*

Proof. We shall reduce the problem to checking the membership of the copositive cone $\mathbf{C}$. In particular, given any matrix $A \in \mathbf{S}^{n^2}$, construct an $\mathcal{F} \in \mathbf{S}^{n^4}$, whose only nonzero entries are

$$\mathcal{F}_{iikk} = \mathcal{F}_{ikik} = \mathcal{F}_{ikki} = \mathcal{F}_{kiik} = \mathcal{F}_{kiki} = \mathcal{F}_{kkii} = \begin{cases} \frac{A_{ik}}{3} & i \neq k, \\ A_{ik} & i = k. \end{cases} \tag{2.8}$$

For any $x \in \mathbb{R}^n$,

$$\begin{aligned} \mathcal{F}(x,x,x,x) &= \sum_{1\leq i<k\leq n} \left(\mathcal{F}_{iikk} + \mathcal{F}_{ikik} + \mathcal{F}_{ikki} + \mathcal{F}_{kiik} + \mathcal{F}_{kiki} + \mathcal{F}_{kkii}\right) {x_i}^2 {x_k}^2 + \sum_{i=1}^n \mathcal{F}_{iiii} {x_i}^4 \\ &= \sum_{1\leq i,k\leq n} A_{ik} {x_i}^2 {x_k}^2 = (x\circ x)^\top A (x \circ x), \end{aligned} \tag{2.9}$$

where the symbol '$\circ$' represents the Hadamard product. Denote $y = x \circ x \geq 0$, and then $\mathcal{F}(x,x,x,x) \geq 0$ if and only if $y^\top A y \geq 0$. Therefore $A \in \mathbf{C}$ if and only if $\mathcal{F} \in \mathbf{S}_+^{n^4}$ and the reduction is complete. $\square$

We remark that Proposition 2.4.2 was already known in the literature, see e.g. [56]. However a formal proof is rarely seen.

Proposition 2.4.3. *It is NP-hard to check if a quartic form belongs to $\mathbf{\Sigma}_{n,4}^4$ (the cone of quartic SOQ forms).*

Proof. Similarly, the problem can be reduced to checking the membership of the completely positive cone $\mathbf{C}^*$. In particular, given any matrix $A \in \mathbf{S}^{n^2}$, construct an $\mathcal{F} \in \mathbf{S}^{n^4}$, whose only nonzero entries are defined exactly as in (2.8). If $A \in \mathbf{C}^*$, then $A = \sum_{t=1}^m a^t (a^t)^\top$ for some $a^1, \ldots, a^m \in \mathbb{R}_+^n$. By the construction of $\mathcal{F}$, we have

$$\mathcal{F}_{iikk} = \mathcal{F}_{ikik} = \mathcal{F}_{ikki} = \mathcal{F}_{kiik} = \mathcal{F}_{kiki} = \mathcal{F}_{kkii} = \begin{cases} \sum_{t=1}^m \frac{a_i^t a_k^t}{3} & i \neq k, \\ \sum_{t=1}^m \left(a_i^t\right)^2 & i = k. \end{cases}$$

Denote $A^t = \mathrm{Diag}\,(a^t) \in \mathbf{S}_+^{n^2}$ for all $1 \leq t \leq m$. It is straightforward to verify that

$$\mathcal{F} = \sum_{t=1}^m \mathrm{sym}\,\left(A^t \otimes A^t\right) = \mathrm{sym}\,\left(\sum_{t=1}^m A^t \otimes A^t\right).$$

Therefore by Theorem 2.2.3 we have $\mathcal{F} \in \mathbf{\Sigma}_{n,4}^4$.

Conversely, if $A \notin \mathbf{C}^*$, then there exits a vector $y \in \mathbb{R}_+^n$, such that $y^\top A y < 0$. Define a vector $x \in \mathbb{R}_+^n$ with $x_i = \sqrt{y_i}$ for all $1 \leq i \leq n$. By (2.9), we have

$$\mathcal{F} \bullet (x \otimes x \otimes x \otimes x) = \mathcal{F}(x,x,x,x) = (x \circ x)^\top A (x \circ x) = y^\top A y < 0.$$

Therefore, by the duality relationship in Theorem 2.2.5, we have $\mathcal{F} \notin \mathbf{\Sigma}_{n,4}^4$. Since $A \in \mathbf{C}^*$ if and only if $\mathcal{F} \in \mathbf{\Sigma}_{n,4}^4$ and so it follows that $\mathbf{\Sigma}_{n,4}^4$ is a hard cone. $\square$

Recently, Burer [57] showed that a large class of mixed-binary quadratic programs can be formulated as copositive programs where a linear function is minimized over a linearly constrained subset of the cone of completely positive matrices. Later, Burer and Dong [58] extended this equivalence to general nonconvex quadratically constrained

quadratic program whose feasible region is nonempty and bounded. From the proof of Proposition 2.4.3, the cone of completely positive matrices can be imbedded into the cone of quartic SOQ forms. Evidently, these mixed-binary quadratic programs can also be formulated as linear conic program with the cone $\mathbf{\Sigma}^4_{n,4}$. In fact, the modeling power of $\mathbf{\Sigma}^4_{n,4}$ is much greater, which we shall discuss in Section 2.5 for further illustration.

Before concluding this subsection, a final remark on the cone $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\rm cvx}$ is in order. Recall the recent breakthrough [11] mentioned in Section 2.1, that checking the convexity of a quartic form is strongly NP-hard. However, if we are given more information, that the quartic form to be considered is a sum of squares, will this make the membership easier? The answer is still no, as the following theorem asserts.

Theorem 2.4.4. *Deciding the convexity of a quartic SOS form is strongly NP-hard. In particular, it is strongly NP-hard to check if a quartic form belongs to* $\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\rm cvx}$.

Proof. Let $G = (V, E)$ be a graph with V being the set of n vertices and E being the set of edges. Define the following bi-quadratic form associated with graph G as follows:

$$b_G(x, y) := 2 \sum_{(i,j)\in E} x_i x_j y_i y_j.$$

Ling et al. [32] showed that the problem $\max_{\|x\|_2=\|y\|_2=1} b_G(x, y)$ is strongly NP-hard. Define

$$b_{G,\lambda}(x, y) := \lambda(x^\top x)(y^\top y) - b_G(x, y) = \lambda(x^\top x)(y^\top y) - 2 \sum_{(i,j)\in E} x_i x_j y_i y_j.$$

Then determining the nonnegativity of $b_{G,\lambda}(x, y)$ in (x, y) is also strongly NP-hard, due to the fact that the problem $\max_{\|x\|_2=\|y\|_2=1} b_G(x, y)$ can be polynomially reduced to it. Let us now construct a quartic form in (x, y) as

$$f_{G,\lambda}(x, y) := b_{G,\lambda}(x, y) + n^2 \left(\sum_{i=1}^n {x_i}^4 + \sum_{i=1}^n {y_i}^4 + \sum_{1\le i<j\le n} {x_i}^2 {x_j}^2 + \sum_{1\le i<j\le n} {y_i}^2 {y_j}^2 \right).$$

Observe that

$$f_{G,\lambda}(x, y) = g_{G,\lambda}(x, y) + \sum_{(i,j)\in E} (x_i x_j - y_i y_j)^2 + (n^2 - 1) \sum_{(i,j)\in E} ({x_i}^2 {x_j}^2 + {y_i}^2 {y_j}^2),$$

where $g_{G,\lambda}(x,y) := \lambda(x^\top x)(y^\top y) + n^2\left(\sum_{i=1}^n({x_i}^4+{y_i}^4)+\sum_{(i,j)\notin E}({x_i}^2{x_j}^2+{y_i}^2{y_j}^2)\right)$. Therefore $f_{G,\lambda}(x,y)$ is quartic SOS in (x,y). Moreover, according to Theorem 2.3 of [11] with $\gamma = 2$, we know that $f_{G,\lambda}(x,y)$ is convex if and only if $b_{G,\lambda}(x,y)$ is nonnegative. The latter being strongly NP-hard, therefore checking the convexity of the quartic SOS form $f_{G,\lambda}(x,y)$ is also strongly NP-hard. □

With the help of Theorem 2.4.4 and Proposition 2.4.1, which claim that $\mathbf{\Sigma}^2_{n,4}\bigcap\mathbf{S}^{n^4}_{\text{cvx}}$ is a hard cone while $\mathbf{\Sigma}^2_{\nabla^2_{n,4}}$ is easy, we conclude the following complete hierarchical structure to clarify one last containing relationship in Theorem 2.3.1. (Note that Theorem 2.3.1 only concludes $\mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subseteq \left(\mathbf{\Sigma}^2_{n,4}\bigcap\mathbf{S}^{n^4}_{\text{cvx}}\right)$.)

Corollary 2.4.5. *Assuming $P \neq NP$, for general n we have*

$$\mathbf{\Sigma}^4_{n,4} \subsetneq \mathbf{S}^{n^2\times n^2}_+ \subsetneq \mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subsetneq \left(\mathbf{\Sigma}^2_{n,4}\bigcap\mathbf{S}^{n^4}_{\text{cvx}}\right) \subsetneq \mathbf{\Sigma}^2_{n,4} \subsetneq \mathbf{S}^{n^4}_+. \tag{2.10}$$

The relationship among these six cones of quartic forms is depicted in Figure 2.1, where a primal-dual pair is painted by the same color. The chain of containing relationship is useful especially when some of the cones are hard while others are easy. One obvious possible application is to use an 'easy' cone either as restriction or as relaxation of a hard one. Such scheme is likely to be useful in the design of approximation algorithms.

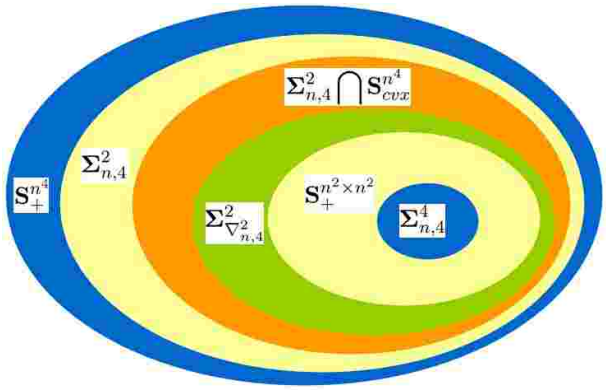

Figure 2.1: Hierarchy for the cones of quartic forms

2.4.2 The Low Dimensional Cases

The chain of containing relations (2.10) holds for general dimension n. Essentially the strict containing relations are true for $n \geq 4$, except that we do not know if $\mathbf{\Sigma}^2_{\nabla^2_{n,4}} \subsetneq \left(\mathbf{\Sigma}^2_{n,4} \bigcap \mathbf{S}^{n^4}_{\rm cvx}\right)$ holds true or not. To complete the picture, in this subsection we discuss quartic forms in low dimensional cases: $n = 2$ and $n = 3$. Specifically, when $n = 2$, the six cones of quartic forms reduce to two distinctive ones; while $n = 3$, they reduce to three distinctive cones.

Proposition 2.4.6. *For the cone of bi-variate quartic forms, it holds that*

$$\mathbf{\Sigma}^4_{2,4} = \mathbf{S}^{2^2\times 2^2}_+ = \mathbf{\Sigma}^2_{\nabla^2_{2,4}} = \left(\mathbf{\Sigma}^2_{2,4} \bigcap \mathbf{S}^{2^4}_{\rm cvx}\right) \subsetneq \mathbf{\Sigma}^2_{2,4} = \mathbf{S}^{2^4}_+.$$

Proof. By a well known equivalence between nonnegative polynomial and sum of squares due to Hilbert [14] (for bi-variate quartic polynomials), we have $\mathbf{\Sigma}^2_{n,4} = \mathbf{S}^{n^4}_+$ for $n \leq 3$, by noticing that Hilbert's result is true for *inhomogeneous* polynomials and our cones are for homogeneous forms. Now, the duality relationship in Theorem 2.2.5 leads to $\mathbf{\Sigma}^4_{2,4} = \mathbf{S}^{2^2\times 2^2}_+$. Next let us focus on the relationship between $\mathbf{S}^{2^2\times 2^2}_+$ and $\mathbf{\Sigma}^2_{2,4} \bigcap \mathbf{S}^{2^4}_{\rm cvx}$. In fact we shall prove below that $\mathbf{S}^{2^4}_{\rm cvx} \subseteq \mathbf{S}^{2^2\times 2^2}_+$, i.e., any bi-variate convex quartic form is quartic matrix PSD.

For bi-variate convex quartic form $\mathcal{F}$ with

$$\mathcal{F}_{1111} = a_1, \mathcal{F}_{1112} = a_2, \mathcal{F}_{1122} = a_3, \mathcal{F}_{1222} = a_4, \mathcal{F}_{2222} = a_5,$$

we have $f(x) = \mathcal{F}(x, x, x, x) = a_1 {x_1}^4 + 4a_2 {x_1}^3 x_2 + 6a_3 {x_1}^2 {x_2}^2 + 4a_4 x_1 {x_2}^3 + a_5 {x_2}^4$, and

$$\nabla^2 f(x) = 12 \left[\begin{array}{cc} a_1 {x_1}^2 + 2a_2 x_1 x_2 + a_3 {x_2}^2 & a_2 {x_1}^2 + 2a_3 x_1 x_2 + a_4 {x_2}^2 \\ a_2 {x_1}^2 + 2a_3 x_1 x_2 + a_4 {x_2}^2 & a_3 {x_1}^2 + 2a_4 x_1 x_2 + a_5 {x_2}^2 \end{array} \right] \succeq 0 \quad \forall\, x_1, x_2 \in \mathbb{R}. \tag{2.11}$$

Denote $A^1 = \left[\begin{array}{cc} a_1 & a_2 \\ a_2 & a_3 \end{array} \right]$, $A^2 = \left[\begin{array}{cc} a_2 & a_3 \\ a_3 & a_4 \end{array} \right]$ and $A^3 = \left[\begin{array}{cc} a_3 & a_4 \\ a_4 & a_5 \end{array} \right]$, and (2.11) is equivalent to

$$\left[\begin{array}{cc} x^\top A^1 x & x^\top A^2 x \\ x^\top A^2 x & x^\top A^3 x \end{array} \right] \succeq 0 \quad \forall\, x \in \mathbb{R}^2. \tag{2.12}$$

According to Theorem 4.8 and the subsequent discussions in [13], it follows that (2.12) is equivalent to $\begin{bmatrix} A^1 & A^2 \\ A^2 & A^3 \end{bmatrix} \succeq 0$. Therefore,

$$\mathcal{F}(X, X) = (\operatorname{vec}(X))^\top \begin{bmatrix} A^1 & A^2 \\ A^2 & A^3 \end{bmatrix} \operatorname{vec}(X) \geq 0 \quad \forall\, X \in \mathbb{R}^{2^2},$$

implying that $\mathcal{F}$ is quartic matrix PSD. This proves $\mathbf{S}_+^{2^2\times 2^2} = \mathbf{\Sigma}_{2,4}^2 \bigcap \mathbf{S}_{\text{cvx}}^{2^4}$. Finally, Example 2.3.2 for $\mathbf{\Sigma}_{2,4}^2 \setminus \mathbf{S}_{\text{cvx}}^{2^4}$ leads to $\mathbf{\Sigma}_{2,4}^2 \bigcap \mathbf{S}_{\text{cvx}}^{2^4} \neq \mathbf{\Sigma}_{2,4}^2$. □

It remains to consider the case $n = 3$. Our previous discussion concluded that $\mathbf{\Sigma}_{3,4}^2 = \mathbf{S}_+^{3^4}$, and so by duality $\mathbf{\Sigma}_{3,4}^4 = \mathbf{S}_+^{3^2\times 3^2}$. Moreover, in a recent working paper Ahmadi et al. [54] showed that every tri-variate convex quartic polynomial is sos-convex, implying $\mathbf{\Sigma}_{\nabla_{3,4}^2}^2 = \left(\mathbf{\Sigma}_{3,4}^2 \bigcap \mathbf{S}_{\text{cvx}}^{3^4}\right)$. So we have at most three distinctive cones of quartic forms. Example 2.3.2 in $\mathbf{\Sigma}_{2,4}^2 \setminus \mathbf{S}_{\text{cvx}}^{2^4}$ and Example 2.3.3 in $\mathbf{\Sigma}_{\nabla_{3,4}^2}^2 \setminus \mathbf{S}_+^{3^2\times 3^2}$ show that there are in fact three distinctive cones.

Proposition 2.4.7. *For the cone of tri-variate quartic forms, it holds that*

$$\mathbf{\Sigma}_{3,4}^4 = \mathbf{S}_+^{3^2\times 3^2} \subsetneq \mathbf{\Sigma}_{\nabla_{3,4}^2}^2 = \left(\mathbf{\Sigma}_{3,4}^2 \bigcap \mathbf{S}_{\text{cvx}}^{3^4}\right) \subsetneq \mathbf{\Sigma}_{3,4}^2 = \mathbf{S}_+^{3^4}.$$

2.4.3 Interiors of the Cones

Unlike the cone of nonnegative quadratic forms, where its interior is completely decided by the positive definiteness, the interior of quartic forms is much more complicated. Here we study two particular simple quartic forms: $\left(x^\top x\right)^2$ whose corresponding tensor is $\operatorname{sym}(I \otimes I)$, and $\sum_{i=1}^n {x_i}^4$ whose corresponding tensor is denoted by $\mathcal{I}$. As we shall show later, even for these two simple forms, to decide if they belong to the interior of certain quartic forms is already nontrivial.

First, it is easy to see that both $\operatorname{sym}(I \otimes I)$ and $\mathcal{I}$ are in the interior of $\mathbf{S}_+^{n^4}$. This is because the inner product between $\mathcal{I}$ and any nonzero form in $\mathbf{\Sigma}_{n,4}^4$ (the dual cone of $\mathbf{S}_+^{n^4}$) is positive. The same situation holds for $\operatorname{sym}(I \otimes I)$. Besides, they are both in $\mathbf{\Sigma}_{n,4}^4$ according to Theorem 2.2.3. Then one may want to know whether they are both in the interior of $\mathbf{\Sigma}_{n,4}^4$. Intuitively, $\mathcal{I}$ seems to be in the interior of $\mathbf{\Sigma}_{n,4}^4$ since it is analogous to the unit matrix in the space of symmetric matrices. However, we have the following counterintuitive result.

Proposition 2.4.8. *It holds that* $\operatorname{sym}(I \otimes I) \in \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$ *and* $\mathcal{I} \notin \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$.

Before providing the proof, let us first discuss the definition of $\operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$. Following Definition 2.1.3, one may define a quartic form $\mathcal{F} \in \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$ if

$$\mathcal{F}(X, X) > 0 \quad \forall\, X \in \mathbb{R}^{n^2} \setminus O. \tag{2.13}$$

However, this condition is sufficient but not necessary. Since for any $\mathcal{F} \in \mathbf{S}^{n^4}$ and any skewness matrix Y, we have $\mathcal{F}(Y, Y) = 0$ according to the proof of Lemma 2.2.4, which leads to empty interior for $\mathbf{S}_+^{n^2 \times n^2}$ if we strictly follow (2.13). Noticing that $\mathbf{S}_+^{n^2 \times n^2} = \left\{ \mathcal{F} \in \mathbf{S}^{n^4} \,|\, \mathcal{F}(X, X) \geq 0,\ \forall\ X \in \mathbf{S}^{n^2} \right\}$ by Theorem 2.2.3, the interior of $\mathbf{S}_+^{n^2 \times n^2}$ shall be correctly defined as follows, which is easy to verify by checking the standard definition of the cone interior.

Definition 2.4.9. *A quartic form* $\mathcal{F} \in \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$ *if and only if* $\mathcal{F}(X, X) > 0$ *for any* $X \in \mathbf{S}^{n^2} \setminus O$.

Proof of Proposition 2.4.8. For any $X \in \mathbf{S}^{n^2} \setminus O$, we observe that $\operatorname{sym}(I \otimes I)(X, X) = 2(\operatorname{tr}(X))^2 + 4\operatorname{tr}(XX^\top) > 0$, implying that $\operatorname{sym}(I \otimes I) \in \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$.

To prove the second part, we let $Y \in \mathbf{S}^{n^2} \setminus O$ with $\operatorname{diag}(Y) = 0$. Then we have $\mathcal{I}(Y, Y) = \sum_{i=1}^n Y_{ii}^2 = 0$, implying that $\mathcal{I} \notin \operatorname{Int}(\mathbf{S}_+^{n^2 \times n^2})$. □

Our main result in this subsection are the follow theorems, which exactly indicate $\mathcal{I}$ and $\operatorname{sym}(I \otimes I)$ in the interior of a particular cone in the hierarchy (2.10), respectively.

Theorem 2.4.10. *It holds that* $\mathcal{I} \notin \operatorname{Int}(\mathbf{S}_{\mathrm{cvx}}^{n^4})$ *and* $\mathcal{I} \in \operatorname{Int}(\mathbf{\Sigma}_{n,4}^2)$.

Proof. To prove the first part, we denote quartic form $\mathcal{F}_\epsilon$ to be $\mathcal{F}_\epsilon(x, x, x, x) = \sum_{i=1}^n x_i^4 - \epsilon x_1^2 x_2^2$, which is perturbed from $\mathcal{I}$. By Proposition 2.3.4, $\mathcal{F}_\epsilon \in \mathbf{S}_{\mathrm{cvx}}^{n^4}$ if an only if

$$\mathcal{F}_\epsilon(x, x, y, y) = \sum_{i=1}^n x_i^2 y_i^2 - \frac{\epsilon}{6}\left(x_1^2 y_2^2 + x_2^2 y_1^2 + 4x_1 x_2 y_1 y_2\right) \geq 0 \quad \forall\, x, y \in \mathbb{R}^n.$$

However, choosing $\hat{x} = (1, 0, 0, \ldots, 0)$ and $\hat{y} = (0, 1, 0, \ldots, 0)$ leads to $\mathcal{F}_\epsilon(\hat{x}, \hat{x}, \hat{y}, \hat{y}) = -\frac{\epsilon}{6} < 0$ for any $\epsilon > 0$. Therefore $\mathcal{F}_\epsilon \notin \mathbf{S}_{\mathrm{cvx}}^{n^4}$, implying that $\mathcal{I} \notin \operatorname{Int}(\mathbf{S}_{\mathrm{cvx}}^{n^4})$.

For the second part, recall that the dual cone of $\mathbf{\Sigma}_{n,4}^2$ is $\mathbf{S}_+^{n^2 \times n^2}$. It suffices to show that $\mathcal{I} \cdot \mathcal{F} > 0$ for any $\mathcal{F} \in \mathbf{S}_+^{n^2 \times n^2} \setminus \mathcal{O}$, or equivalently $\mathcal{I} \cdot \mathcal{F} = 0$ for $\mathcal{F} \in \mathbf{S}_+^{n^2 \times n^2}$ implies

$\mathcal{F} = \mathcal{O}$. Now rewrite $\mathcal{F}$ as an $n^2 \times n^2$ symmetric matrix $M_{\mathcal{F}}$. Clearly, $\mathcal{F} \in \mathbf{S}_+^{n^2 \times n^2}$ implies $M_{\mathcal{F}} \succeq 0$, with its diagonal components $\mathcal{F}_{ijij} \geq 0$ for any i, j, in particular $\mathcal{F}_{iiii} \geq 0$ for any i. Combing this fact and the assumption that $\mathcal{I} \cdot \mathcal{F} = \sum_{i=1}^n \mathcal{F}_{iiii} = 0$ yeilds $\mathcal{F}_{iiii} = 0$ for any i. Next, we noticed that for any $i \neq j$, the matrix $\begin{bmatrix} \mathcal{F}_{iiii} & \mathcal{F}_{iijj} \\ \mathcal{F}_{jjii} & \mathcal{F}_{jjjj} \end{bmatrix}$ is a principle minor of the positive semidefinite matrix $M_{\mathcal{F}}$; as a result $\mathcal{F}_{iijj} = 0$ for any $i \neq j$. Since $\mathcal{F}$ is super-symmetric, we further have $\mathcal{F}_{ijij} = \mathcal{F}_{iijj} = 0$. Therefore $\operatorname{diag}(M_{\mathcal{F}}) = 0$, which combining $M_{\mathcal{F}} \succeq 0$ leads to $M_{\mathcal{F}} = O$. Hence $\mathcal{F} = \mathcal{O}$ and the conclusion follows. □

Theorem 2.4.11. *It holds that* $\operatorname{sym}(I \otimes I) \in \operatorname{Int}(\mathbf{\Sigma}_{n,4}^4)$.

Proof. By the duality relationship between $\mathbf{\Sigma}_{n,4}^4$ and $\mathbf{S}_+^{n^4}$, it suffices to show that any $\mathcal{F} \in \mathbf{S}_+^{n^4}$ with $\operatorname{sym}(I \otimes I) \cdot \mathcal{F} = 0$ implies $\mathcal{F} = \mathcal{O}$. For this qualified $\mathcal{F}$, we have $\mathcal{F}(x, x, x, x) \geq 0$ for any $x \in \mathbb{R}^n$. For any given i, let $x_i = 1$ and other entries be zeros, and it leads to

$$\mathcal{F}_{iiii} \geq 0 \quad \forall\, i. \tag{2.14}$$

Next, let $\xi \in \mathbb{R}^n$ whose entries are i.i.d. symmetric Bernoulli random variables, i.e., $\operatorname{Prob}\{\xi_i = 1\} = \operatorname{Prob}\{\xi_i = -1\} = \frac{1}{2}$ for all i. Then it is easy to compute

$$\mathsf{E}[\mathcal{F}(\xi, \xi, \xi, \xi)] = \sum_{i=1}^n \mathcal{F}_{iiii} + 6 \sum_{1 \leq i < j \leq n} \mathcal{F}_{iijj} \geq 0. \tag{2.15}$$

Besides, for any given $i \neq j$, let $\eta \in \mathbb{R}^n$ where η_i and η_j are independent symmetric Bernoulli random variables and other entries are zeros. Then

$$\mathsf{E}[\mathcal{F}(\eta, \eta, \eta, \eta)] = \mathcal{F}_{iiii} + \mathcal{F}_{jjjj} + 6\mathcal{F}_{iijj} \geq 0 \quad \forall\, i \neq j. \tag{2.16}$$

Since we assume $\operatorname{sym}(I \otimes I) \cdot \mathcal{F} = 0$, it follows that

$$\sum_{i=1}^n \mathcal{F}_{iiii} + 2 \sum_{1 \leq i < j \leq n} \mathcal{F}_{iijj} = \frac{1}{3} \left(\sum_{i=1}^n \mathcal{F}_{iiii} + 6 \sum_{1 \leq i < j \leq n} \mathcal{F}_{iijj} \right) + \frac{2}{3} \sum_{i=1}^n \mathcal{F}_{iiii} = 0. \tag{2.17}$$

Combining (2.14), (2.15) and (2.17), we get

$$\mathcal{F}_{iiii} = 0 \quad \forall\, i. \tag{2.18}$$

It further leads to $\mathcal{F}_{iijj} \geq 0$ for any $i \neq j$ by (2.16). Combining this result again with (2.17) and (2.18), we get

$$\mathcal{F}_{iijj} = 0 \quad \forall\, i \neq j. \tag{2.19}$$

Now it suffices to prove $\mathcal{F}_{iiij} = 0$ for all $i \neq j$, $\mathcal{F}_{iijk} = 0$ for all distinctive i, j, k, and $\mathcal{F}_{ijk\ell} = 0$ for all distinctive i, j, k, ℓ. To this end, for any given $i \neq j$, let $x \in \mathbb{R}^n$ where $x_i = t^2$ and $x_j = \frac{1}{t}$ and other entries are zeros. By (2.18) and (2.19), it follows that

$$\mathcal{F}(x, x, x, x) = 4\mathcal{F}_{iiij}\, x_i^3 x_j + 4\mathcal{F}_{ijjj}\, x_i x_j^3 = 4\mathcal{F}_{iiij}\, t^5 + 4\mathcal{F}_{ijjj}/t \geq 0 \quad \forall\, i \neq j.$$

Letting $t \to \pm\infty$, we get

$$\mathcal{F}_{iiij} = 0 \quad \forall\, i \neq j. \tag{2.20}$$

For any given distinctive i, j, k, let $x \in \mathbb{R}^n$ whose only nonzero entries are x_i, x_j and x_k, and we have

$$\mathcal{F}(x, x, x, x) = 12\mathcal{F}_{iijk}\, x_i^2 x_j x_k + 12\mathcal{F}_{jjik}\, x_j^2 x_i x_k + 12\mathcal{F}_{kkij}\, x_k^2 x_i x_j \geq 0 \quad \forall \text{ distincive } i, j, k.$$

Taking $x_j = 1, x_k = \pm 1$ in the above leads to $\pm(\mathcal{F}_{iijk}\, x_i^2 + \mathcal{F}_{jjik}\, x_i) + \mathcal{F}_{kkij}\, x_i \geq 0$ for any $x_i \in \mathbb{R}$, and we get

$$\mathcal{F}_{iijk} = 0 \quad \forall \text{ distincive } i, j, k. \tag{2.21}$$

Finally, for any given distinctive i, j, k, ℓ, let $x \in \mathbb{R}^n$ whose only nonzero entries are x_i, x_j, x_k and x_ℓ, and we have

$$\mathcal{F}(x, x, x, x) = 24\mathcal{F}_{ijk\ell}\, x_i x_j x_k x_\ell \geq 0 \quad \forall \text{ distincive } i, j, k, \ell.$$

Taking $x_i = x_j = x_k = 1$ and $x_\ell = \pm 1$ leads to

$$\mathcal{F}_{ijk\ell} = 0 \quad \forall \text{ distincive } i, j, k, \ell. \tag{2.22}$$

Combining equations (2.18), (2.19), (2.20), (2.21) and (2.22) yields $\mathcal{F} = \mathcal{O}$. □

2.5 Quartic Conic Programming

The study of quartic forms in the previous sections gives rise some new modeling opportunities. In this section we shall discuss quartic conic programming, i.e., optimizing

a linear function over the intersection of an affine subspace and a cone of quartic forms. In particular, we shall investigate the following quartic conic programming model:

$$\begin{array}{lll} (QCP) & \max & \mathcal{C} \bullet \mathcal{X} \\ & \text{s.t.} & \mathcal{A}^i \bullet \mathcal{X} = b_i,\ i = 1, \ldots, m \\ & & \mathcal{X} \in \mathbf{\Sigma}^4_{n,4}, \end{array}$$

where $\mathcal{C}, \mathcal{A}^i \in \mathbf{S}^{n^4}$ and $b_i \in \mathbf{R}$ for $i = 1, \ldots, m$. As we will see later, a large class of non-convex quartic polynomial optimization models can be formulated as a special class of (QCP). In fact we will study a few concrete examples to show the modeling power of the quartic forms that we introduced.

2.5.1 Quartic Polynomial Optimization

Quartic polynomial optimization received much attention in the recent years; see e.g. [22, 32, 33, 34, 36, 59]. Essentially, all the models studied involve optimization of a quartic polynomial function subject to some linear and/or homogenous quadratic constraints, including spherical constraints, binary constraints, the intersection of co-centered ellipsoids, and so on. Below we consider a very general quartic polynomial optimization model:

$$\begin{array}{lll} (P) & \max & p(x) \\ & \text{s.t.} & (a^i)^\top x = b_i,\ i = 1, \ldots, m \\ & & x^\top A^j x = c_j,\ j = 1, \ldots, l \\ & & x \in \mathbb{R}^n, \end{array}$$

where $p(x)$ is a general inhomogeneous quartic polynomial function.

We first homogenize $p(x)$ by introducing a new homogenizing variable, say x_{n+1}, which is set to one, and get a homogeneous quartic form

$$p(x) = \mathcal{F}(\bar{x}, \bar{x}, \bar{x}, \bar{x}) = \mathcal{F} \bullet (\bar{x} \otimes \bar{x} \otimes \bar{x} \otimes \bar{x}),$$

where $\mathcal{F} \in \mathbf{S}^{(n+1)^4}$, $\bar{x} = \begin{pmatrix} x \\ x_{n+1} \end{pmatrix}$ and $x_{n+1} = 1$. By adding some redundant constraints,

we have an equivalent formulation of (P):

$$\begin{array}{ll} \max & \mathcal{F}(\bar{x},\bar{x},\bar{x},\bar{x}) \\ \text{s.t.} & (a^i)^\top x = b_i,\ \left((a^i)^\top x\right)^2 = {b_i}^2,\ \left((a^i)^\top x\right)^4 = {b_i}^4,\ i=1,\ldots,m \\ & x^\top A^j x = c_j,\ \left(x^\top A^j x\right)^2 = {c_j}^2,\ j=1,\ldots,l \\ & \bar{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{n+1}. \end{array}$$

The objective function of the above problem can be taken as a linear function of $\bar{x}\otimes\bar{x}\otimes\bar{x}\otimes\bar{x}$, and we introduce new variables of a super-symmetric fourth order tensor $\bar{\mathcal{X}} \in \mathbf{S}^{(n+1)^4}$. The notations x, X, and $\mathcal{X}$ extract part of the entries of $\bar{\mathcal{X}}$, which are defined as:

$$\begin{array}{lll} x \in \mathbb{R}^n & x_i = \bar{\mathcal{X}}_{i,n+1,n+1,n+1} & \forall\, 1 \le i \le n, \\ X \in \mathbf{S}^{n^2} & X_{i,j} = \bar{\mathcal{X}}_{i,j,n+1,n+1} & \forall\, 1 \le i,j \le n, \\ \mathcal{X} \in \mathbf{S}^{n^4} & \mathcal{X}_{i,j,k,\ell} = \bar{\mathcal{X}}_{i,j,k,\ell} & \forall\, 1 \le i,j,k,\ell \le n. \end{array}$$

Essentially they can be treated as linear constraints on $\bar{\mathcal{X}}$. Now by taking $\bar{\mathcal{X}} = \bar{x}\otimes\bar{x}\otimes\bar{x}\otimes\bar{x}$, $\mathcal{X} = x\otimes x\otimes x\otimes x$, and $X = x\otimes x$, we may equivalently represent the above problem as a quartic conic programming model with a rank-one constraint:

$$\begin{array}{lll} (Q) & \max & \mathcal{F}\bullet\bar{\mathcal{X}} \\ & \text{s.t.} & (a^i)^\top x = b_i,\ (a^i\otimes a^i)\bullet X = {b_i}^2,\ (a^i\otimes a^i\otimes a^i\otimes a^i)\bullet\mathcal{X} = {b_i}^4,\ i=1,\ldots,m \\ & & A^j\bullet X = c_j,\ (A^j\otimes A^j)\bullet\mathcal{X} = {c_j}^2,\ j=1,\ldots,l \\ & & \bar{\mathcal{X}}_{n+1,n+1,n+1,n+1} = 1,\ \bar{\mathcal{X}} \in \mathbf{\Sigma}^4_{n+1,4},\ \mathrm{rank}\,(\bar{\mathcal{X}}) = 1. \end{array}$$

Dropping the rank-one constraint, we obtain a relaxation problem, which is exactly in the form of quartic conic program (QCP):

$$\begin{array}{lll} (RQ) & \max & \mathcal{F}\bullet\bar{\mathcal{X}} \\ & \text{s.t.} & (a^i)^\top x = b_i,\ (a^i\otimes a^i)\bullet X = {b_i}^2,\ (a^i\otimes a^i\otimes a^i\otimes a^i)\bullet\mathcal{X} = {b_i}^4,\ i=1,\ldots,m \\ & & A^j\bullet X = c_j,\ (A^j\otimes A^j)\bullet\mathcal{X} = {c_j}^2,\ j=1,\ldots,l \\ & & \bar{\mathcal{X}}_{n+1,n+1,n+1,n+1} = 1,\ \bar{\mathcal{X}} \in \mathbf{\Sigma}^4_{n+1,4}. \end{array}$$

Interestingly, the relaxation from (Q) to (RQ) is not lossy; or, to put it differently, (RQ) is a tight relaxation of (Q), under some mild conditions.

Theorem 2.5.1. *If $A^j \in \mathbf{S}^{n^2}_+$ for all $1 \le j \le l$ in the model (P), then (RQ) is equivalent to (P) in the sense that: (i) they have the same optimal value; (ii) if $\bar{\mathcal{X}}$ is*

optimal to (RQ)*, then* x *is in the convex hull of the optimal solution of* (P)*. Moreover, the minimization counterpart of* (P) *is also equivalent to the minimization counterpart of* (RQ)*.*

Theorem 2.5.1 shows that (P) is in fact a conic quartic program (QCP) when the matrices A^j's in (P) are positive semidefinite. Notice that the model (P) actually includes quadratic inequality constraints $x^\top A^j x \leq c_j$ as its subclasses, for one can always add a slack variable $y_j \in \mathbb{R}$ with $x^\top A^j x + {y_j}^2 = c_j$, while reserving the new data matrix $\begin{bmatrix} A^j & 0 \\ 0 & 1 \end{bmatrix}$ in the quadratic term still being positive semidefinite.

As mentioned before, Burer [57] established the equivalence between a large class of mixed-binary quadratic programs and copositive programs. Theorem 2.5.1 may be regarded as a quartic extension of Burer's result. The virtue of this equivalence is to alleviate the highly non-convex objective and/or constraints of (QCP) and retain the problem in convex form, although the difficulty is all absorbed into the dealing of the quartic cone, which is nonetheless a convex one. Note that this is characteristically a property for polynomial of degree higher than 2: the SDP relaxation for similar quadratic models can never be tight.

In the following, we shall present the proof of Theorem 2.5.1. Since the proof for their minimization counterparts is exactly the same, we only prove the equivalence relation for the maximization problems. That is, we shall prove the equivalence between (Q) and (RQ).

To start with, let us first investigate the feasible regions of these two problems, to be denoted by $\text{feas}\,(Q)$ and $\text{feas}\,(RQ)$ respectively. The relationship between $\text{feas}\,(Q)$ and $\text{feas}\,(RQ)$ is revealed by the following lemma.

Lemma 2.5.2. *It holds that* $\text{conv}\,(\text{feas}\,(Q)) \subseteq \text{feas}\,(RQ) = \text{conv}\,(\text{feas}\,(Q)) + \mathbf{P}$*, where*

$$\mathbf{P} := \text{cone}\left\{ \begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \middle| \begin{array}{ll} (a^i)^\top x = 0 & \forall\, 1 \leq i \leq m, \\ x^\top A^j x = 0 & \forall\, 1 \leq j \leq l \end{array} \right\} \subset \mathbf{\Sigma}^4_{n+1,4}.$$

Proof. First, it is obvious that $\text{conv}\,(\text{feas}\,(Q)) \subseteq \text{feas}\,(RQ)$, since (RQ) is a relaxation of (Q) and $\text{feas}\,(RQ)$ is convex. Next we notice that the recession cone of $\text{feas}\,(RQ)$ is

equal to

$$\left\{ \bar{\mathcal{X}} \in \mathbf{\Sigma}^4_{n+1,4} \;\middle|\; \begin{array}{ll} \mathcal{X}_{n+1,n+1,n+1,n+1} = 0, & \\ (a^i)^\top x = 0,\ (a^i \otimes a^i) \bullet X = 0,\ (a^i \otimes a^i \otimes a^i \otimes a^i) \bullet \mathcal{X} = 0 & \forall\, 1 \le i \le m, \\ A^j \bullet X = 0,\ (A^j \otimes A^j) \bullet \mathcal{X} = 0 & \forall\, 1 \le j \le l \end{array} \right\}.$$

Observing that $\bar{\mathcal{X}} \in \mathbf{\Sigma}^4_{n+1,4}$ and $\mathcal{X}_{n+1,n+1,n+1,n+1} = 0$, it is easy to see that $x = 0$ and $X = 0$. Thus the recession cone of feas (RQ) is reduced to

$$\left\{ \bar{\mathcal{X}} \in \mathbf{\Sigma}^4_{n+1,4} \;\middle|\; \begin{array}{ll} \mathcal{X}_{n+1,n+1,n+1,n+1} = 0,\ x = 0,\ X = 0, & \\ (a^i \otimes a^i \otimes a^i \otimes a^i) \bullet \mathcal{X} = 0 & \forall\, 1 \le i \le m, \\ (A^j \otimes A^j) \bullet \mathcal{X} = 0 & \forall\, 1 \le j \le l \end{array} \right\} \supseteq \mathbf{P},$$

which proves feas $(RQ) \supseteq$ conv (feas $(Q)) + \mathbf{P}$.

Finally, we shall show the inverse inclusion, i.e., feas $(RQ) \subseteq$ conv (feas $(Q)) + \mathbf{P}$. Suppose $\bar{\mathcal{X}} \in$ feas (RQ), then it can be decomposed as

$$\bar{\mathcal{X}} = \sum_{k \in K} \begin{pmatrix} y^k \\ \alpha_k \end{pmatrix} \otimes \begin{pmatrix} y^k \\ \alpha_k \end{pmatrix} \otimes \begin{pmatrix} y^k \\ \alpha_k \end{pmatrix} \otimes \begin{pmatrix} y^k \\ \alpha_k \end{pmatrix}, \tag{2.23}$$

where $\alpha_k \in \mathbb{R}$, $y^k \in \mathbb{R}^n$ for all $k \in K$. Immediately we have

$$\sum_{k \in K} {\alpha_k}^4 = \mathcal{X}_{n+1,n+1,n+1,n+1} = 1. \tag{2.24}$$

Now divide the index set K into two parts, with $K_0 := \{k \in K \mid \alpha_k = 0\}$ and $K_1 := \{k \in K \mid \alpha_k \neq 0\}$, and let $z^k = y^k/\alpha_k$ for all $k \in K_1$. The decomposition (2.23) is then equivalent to

$$\bar{\mathcal{X}} = \sum_{k \in K_1} {\alpha_k}^4 \begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} + \sum_{k \in K_0} \begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix}.$$

If we can prove that

$$\begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} \otimes \begin{pmatrix} z^k \\ 1 \end{pmatrix} \in \text{feas}\,(Q) \quad \forall\, k \in K_1 \tag{2.25}$$

$$\begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix} \otimes \begin{pmatrix} y^k \\ 0 \end{pmatrix} \in \mathbf{P} \qquad \forall\, k \in K_0 \tag{2.26}$$

then by (2.24), we shall have $\bar{\mathcal{X}} \in$ conv (feas $(Q)) + \mathbf{P}$, proving the inverse inclusion.

In the following we shall prove (2.25) and (2.26). Since $\bar{\mathcal{X}} \in \text{feas}\,(RQ)$, together with $x = \sum_{k\in K} {\alpha_k}^3 y^k$, $X = \sum_{k\in K} {\alpha_k}^2 y^k \otimes y^k$, and $\mathcal{X} = \sum_{k\in K} y^k \otimes y^k \otimes y^k \otimes y^k$, we obtain the following equalities:

$$\begin{aligned}
&\sum_{k\in K} {\alpha_k}^3 (a^i)^\top y^k = b_i,\ \sum_{k\in K} {\alpha_k}^2 \left((a^i)^\top y^k\right)^2 = {b_i}^2,\ \sum_{k\in K} \left((a^i)^\top y^k\right)^4 = {b_i}^4, \quad \forall\, 1 \le i \le m \\
&\sum_{k\in K} {\alpha_k}^2 (y^k)^\top A^j y^k = c_j,\ \sum_{k\in K} \left((y^k)^\top A^j y^k\right)^2 = {c_j}^2, \qquad\qquad \forall\, 1 \le j \le l.
\end{aligned}$$

As a direct consequence of the above equalities and (2.24), we have

$$\begin{aligned}
&\left(\sum_{k\in K} {\alpha_k}^2 \cdot \alpha_k (a^i)^\top y^k\right)^2 = {b_i}^2 = \left(\sum_{k\in K} {\alpha_k}^4\right)\left(\sum_{k\in K} {\alpha_k}^2 \left((a^i)^\top y^k\right)^2\right), \quad \forall\, 1 \le i \le m \\
&\left(\sum_{k\in K} {\alpha_k}^2 \left((a^i)^\top y^k\right)^2\right)^2 = {b_i}^4 = \left(\sum_{k\in K} {\alpha_k}^4\right)\left(\sum_{k\in K} \left((a^i)^\top y^k\right)^4\right), \qquad \forall\, 1 \le i \le m \\
&\left(\sum_{k\in K} {\alpha_k}^2 (y^k)^\top A^j y^k\right)^2 = {c_j}^2 = \left(\sum_{k\in K} {\alpha_k}^4\right)\left(\sum_{k\in K} \left((y^k)^\top A^j y^k\right)^2\right), \qquad \forall\, 1 \le j \le l.
\end{aligned}$$

Noticing that the equalities hold for the above Cauchy-Schwarz inequalities, it follows that for every $1 \le i \le m$ and every $1 \le j \le l$, there exist $\delta_i, \epsilon_i, \theta_j \in \mathbb{R}$, such that

$$\delta_i {\alpha_k}^2 = \alpha_k (a^i)^\top y^k,\ \epsilon_i {\alpha_k}^2 = \left((a^i)^\top y^k\right)^2 \text{ and } \theta_j {\alpha_k}^2 = (y^k)^\top A^j y^k \quad \forall\, k \in K. \tag{2.27}$$

If $\alpha_k = 0$, then $(a^i)^\top y^k = 0$ and $(y^k)^\top A^j y^k = 0$, which implies (2.26). Moreover, due to (2.27) and (2.24),

$$\delta_i = \delta_i \left(\sum_{k\in K} {\alpha_k}^4\right) = \sum_{k\in K} \delta_i {\alpha_k}^2 \cdot {\alpha_k}^2 = \sum_{k\in K} \alpha_k (a^i)^\top y^k \cdot {\alpha_k}^2 = b_i \quad \forall\, 1 \le i \le m.$$

Similarly, we have $\theta_j = c_j$ for all $1 \le j \le l$. If $\alpha_k \neq 0$, noticing $z^k = y^k/\alpha_k$, it follows from (2.27) that

$$\begin{aligned}
&(a^i)^\top z^k = (a^i)^\top y^k/\alpha_k = \delta_i = b_i \qquad\qquad \forall\, 1 \le i \le m \\
&(z^k)^\top A^j z^k = (y^k)^\top A^j y^k/{\alpha_k}^2 = \theta_j = c_j \quad \forall\, 1 \le j \le l,
\end{aligned}$$

which implies (2.25). □

Proof of Theorem 2.5.1: we notice that if A^j is positive semidefinite, then $x^\top A^j x = 0 \iff A^j x = 0$. Therefore, $\begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \otimes \begin{pmatrix} x \\ 0 \end{pmatrix} \in \mathbf{P}$ implies that x is a recession direction of the feasible region for (P). With this property and using a similar argument of Theorem 2.6 in [57], Theorem 2.5.1 follows immediately. □

2.5.2 Biquadratic Assignment Problems

The biquadratic assignment problem $(BQAP)$ is a generalization of the quadratic assignment problem (QAP), which is to minimize a quartic polynomial of an assignment matrix:

$$\begin{array}{lll} (BQAP) & \min & \sum_{1\le i,j,k,\ell,s,t,u,v\le n} \mathcal{A}_{ijk\ell}\mathcal{B}_{stuv}X_{is}X_{jt}X_{ku}X_{\ell v} \\ & \text{s.t.} & \sum_{i=1}^n X_{ij} = 1,\ j = 1,\ldots,n \\ & & \sum_{j=1}^n X_{ij} = 1,\ i = 1,\ldots,n \\ & & X_{ij} \in \{0,1\},\ i,j = 1,\ldots,n \\ & & X \in \mathbb{R}^{n^2}, \end{array}$$

where $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n^4}$. This problem was first considered by Burkard et al. [60] and was shown to have applications in the VLSI synthesis problem. After that, several heuristics for $(BQAP)$ were developed by Burkard and Cela [61], and Mavridou et al. [62].

In this subsection we shall show that $(BQAP)$ can be formulated as a quartic conic program (QCP). First notice that the objective function of $(BQAP)$ is a fourth order polynomial function with respect to the variables X_{ij}'s, where X is taken as an n^2-dimensional vector. The assignment constraints $\sum_{i=1}^n X_{ij} = 1$ and $\sum_{j=1}^n X_{ij} = 1$ are clearly linear equality constraints. Finally by imposing a new variable $x_0 \in \mathbb{R}$, and the binary constraints $X_{ij} \in \{0,1\}$ is equivalent to

$$\begin{pmatrix} X_{ij} \\ x_0 \end{pmatrix}^\top \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{pmatrix} X_{ij} \\ x_0 \end{pmatrix} = \frac{1}{4} \quad \text{and} \quad x_0 = \frac{1}{2},$$

where the coefficient matrix in the quadratic term is indeed positive semidefinite. Applying Theorem 2.5.1 we have the following result:

Corollary 2.5.3. *The biquadratic assignment problem $(BQAP)$ can be formulated as a quartic conic program (QCP).*

2.5.3 Eigenvalues of Fourth Order Super-Symmetric Tensor

The notion of eigenvalue for matrices has been extended to tensors, proposed by Lim [63] and Qi [64] independently. Versatile extensions turned out to be possible, among which the most popular one is called *Z-eigenvalue* (in the notion by Qi [64]). Restricting to the space of fourth order super-symmetric tensors $\mathbf{S}^{n^4}$, $\lambda \in \mathbb{R}$ is called a Z-eigenvalue of the super-symmetric tensor $\mathcal{F} \in \mathbf{S}^{n^4}$, if the following system holds

$$\begin{cases} \mathcal{F}(x, x, x, \cdot) = \lambda\, x, \\ x^\top x = 1, \end{cases}$$

where $x \in \mathbb{R}^n$ is the corresponding eigenvector with respect to λ. Notice that the Z-eigenvalues are the usual eigenvalues for a symmetric matrix, when restricting to the space of symmetric matrices $\mathbf{S}^{n^2}$. We refer interested readers to [63, 64] for various other definitions of tensor eigenvalues and [65] for their applications in polynomial optimizations.

Observe that x is a Z-eigenvector of the fourth order tenor $\mathcal{F}$ if and only if x is a KKT point to following polynomial optimization problem:

$$\begin{array}{lll} (E) & \max & \mathcal{F}(x, x, x, x) \\ & \text{s.t.} & x^\top x = 1. \end{array}$$

Furthermore, x is the Z-eigenvector with respect to the largest (resp. smallest) Z-eigenvalue of $\mathcal{F}$ if and only if x is optimal to (E) (resp. the minimization counterpart of (E)). As the quadratic constraint $x^\top x = 1$ satisfies the condition in Theorem 2.5.1, we reach the following conclusion:

Corollary 2.5.4. *The problem of finding a Z-eigenvector with respect to the largest or smallest Z-eigenvalue of a fourth order super-symmetric tensor $\mathcal{F}$ can be formulated as a quartic conic program (QCP).*

To conclude this section, as well as the whole chapter, we remark here that quartic conic problems have many potential application, alongside their many intriguing theoretical properties. The hierarchical structure of the quartic cones that we proved in the previous sections paves a way for possible relaxation methods to be viable. For instance, according to the hierarchy relationship (2.10), by relaxing the cone $\mathbf{\Sigma}^4_{n,4}$ to an easy cone

$\mathbf{S}_+^{n^2\times n^2}$ lends a hand to solve the quartic conic problem approximately. The quality of such solution methods and possible enhancements remain our future research topic.

Chapter 3

Polynomial Sized Representation of Hilbert's Identity

3.1 Introduction

The so-called Liouville formula states that

$$(x_1^2 + x_2^2 + x_3^2 + x_4^2)^2 = \frac{1}{6}\sum_{1\le i<j\le 4}(x_i + x_j)^4 + \frac{1}{6}\sum_{1\le i<j\le 4}(x_i - x_j)^4, \tag{3.1}$$

which is straightforward to verify. Interestingly, such 'rank-one' decomposition of the positive quartic form can be extended, giving rise to an identity known as Hilbert's identity, which asserts that for any fixed positive integers d and n, there always exist real vectors $a_1, a_2, ..., a_t \in \mathbb{R}^n$ such that

$$(x^\top x)^d = \sum_{i=1}^{t}(a_i^\top x)^{2d}, \tag{3.2}$$

for any $x \in \mathbb{R}^n$. It worths mentioning that Hilbert's identity is a fundamental result in mathematics. As we mentioned at the very beginning of this thesis, Reznick [16] managed to solve Hilbert's seventeenth problem constructively when the polynomial function is positive definite. Moreover, this identity can be readily extended to a more general setting. For any given $A \succeq 0$, by letting $y = A^{\frac{1}{2}}x$ and applying (3.2), one has

$$(x^\top Ax)^2 = (y^\top y)^2 = \sum_{j=1}^{t}\left((b^j)^\top y\right)^{2d} = \sum_{j=1}^{t}\left((b^j)^\top A^{\frac{1}{2}}x\right)^{2d},$$

which guarantees the existence of vectors $a^1, a^2, \cdots, a^t \in \mathbb{R}^n$ with $a^j = A^{\frac{1}{2}} b^j$ for $j = 1, 2, \ldots, t$ such that

$$(x^\top A x)^d = \sum_{j=1}^{t} \left((a^j)^\top x \right)^{2d}. \tag{3.3}$$

(3.2) was first proved by Hilbert (see [17]), and he showed that

> Given fixed positive integers d and n, there exist $2d+1$ real numbers $\beta_1, \beta_2, \cdots, \beta_{2d+1}$, $2d+1$ positive real numbers $\rho_1, \rho_2, \cdots, \rho_{2d+1}$, and a positive real number α_d, such that
>
> $$(x^\top x)^d = \frac{1}{\alpha_d} \sum_{i_1=1}^{n} \sum_{i_2=1}^{n} \cdots \sum_{i_{2d+1}=1}^{n} \rho_{i_1} \rho_{i_2} \cdots \rho_{i_{2d+1}} (\beta_{i_1} x_1 + \beta_{i_2} x_2 + \cdots + \beta_{i_{2d+1}} x_{i_{2d+1}})^{2d}. \tag{3.4}$$

It is obvious that the number of $2d$-powered linear terms on the right hand side of (3.4) is $(2d+1)^n$, which is exponential with respect to n and thus inefficient for practical purposes. In general, the presentation of $(x^\top x)^d$ is not unique. For example, one may verify that

$$(x_1^2+x_2^2+x_3^2)^2 = \frac{1}{3} \sum_{i=1}^{3} x_i^4 + \frac{1}{3} \sum_{1 \le i < j \le 3} \sum_{\beta_j = \pm 1} (x_i + \beta_j x_j)^4 = \frac{2}{3} \sum_{i=1}^{3} x_i^4 + \frac{1}{3} \sum_{\substack{\beta_2 = \pm 1 \\ \beta_3 = \pm 1}} (x_1 + \beta_2 x_2 + \beta_3 x_3)^4,$$

which leads to two different representations of the form $(x_1^2 + x_2^2 + x_3^2)^2$. An interesting question is to find a succinct (preferably the shortest) representation among all the different representations, including the one from Hilbert's construction. By Carathéodory's theorem, there exists a decomposition such that the value of t in (3.2) is no more than $\binom{n+2d-1}{2d} + 1$. Unfortunately, Carathéodory's theorem is non-constructive. This motivates us to construct a *polynomial-size* representation, i.e., $t = O(n^k)$ for some constant k in (3.2).

Toward this end, let's first reinvestigate the construction (3.4) given by Hilbert. Define i.i.d. random variables $\xi_1, \xi_2, \cdots, \xi_n$ with supporting set $\Delta = \{\beta_1, \beta_2, \cdots, \beta_{2d+1}\}$, and let $\text{Prob}(\xi_k = \beta_i) = \frac{\rho_i}{\gamma_d}$ for all $1 \le i \le 2d+1$ and $1 \le k \le n$, where $\gamma_d = \sum_{i=1}^{2d+1} \beta_i$.

Then identity (3.4) is equivalent to

$$(x^\top x)^d = \frac{\gamma_d^d}{\alpha_d}\mathsf{E}\left[\left(\sum_{j=1}^n \xi_j x_j\right)^{2d}\right] = \frac{\gamma_d^d}{\alpha_d}\sum_{p\in\mathbb{P}_{2d}^n}\mathsf{E}\left[\prod_{j=1}^n \xi_j^{p_j}\right]\prod_{j=1}^n x_j^{p_j} = \frac{\gamma_d^d}{\alpha_d}\sum_{p\in\mathbb{P}_{2d}^n}\prod_{j=1}^n \mathsf{E}\left[\xi_j^{p_j}\right]\prod_{j=1}^n x_j^{p_j}, \tag{3.5}$$

where $\mathbb{P}_k^n := \left\{(p_1, p_2, \cdots, p_n)^\top \in \mathbb{Z}_+^n \mid p_1 + p_2 + \cdots + p_n = k\right\}$. In light of formula (3.5), we learn that the length of representation of $(x^\top x)^d$ equals to the size of sample space spanned by random variables $\xi_1, \xi_2, \cdots, \xi_n$ and there are $(2d+1)^n$ possible outcomes in total. As a consequence, the issue of reducing the representation of $(x^\top x)^d$ boils down whether we can find another n random variables $\eta_1, \eta_2, \cdots, \eta_n$ with smaller sample space such that

$$(x^\top x)^d = \frac{\gamma_d^d}{\alpha_d}\mathsf{E}\left[\left(\sum_{j=1}^n \xi_j x_j\right)^{2d}\right] = \frac{\gamma_d^d}{\alpha_d}\mathsf{E}\left[\left(\sum_{j=1}^n \eta_j x_j\right)^{2d}\right]. \tag{3.6}$$

This issue will be particularly addressed in Section 3.2 and a key concept called k-wise zero-correlation will also be introduced there. Then we discuss how to construct k-wise zero-correlated random variables in Section 3.3, and find the polynomial sized representation of Hilbert's identity in Section 3.4. Finally, in Section 3.5 we conclude this chapter with an application of showing the matrix $2 \mapsto 4$ norm problem, whose computational complexity was previously, is actually NP-hard unknown.

3.2 k-Wise Zero-Correlation Random Variables

In this section, let us first introduce the new notion of *k-wise uncorrelated random variables*, which may appear to be completely unrelated to the discussion of Hilbert's identity at first glance.

Definition 3.2.1. *(k-**wise zero-correlation**) A set of random variables $\{\xi_1, \xi_2, \cdots, \xi_n\}$ is called k-wise zero-correlated if*

$$\mathsf{E}\left[\prod_{j=1}^n \xi_j^{p_j}\right] = \prod_{j=1}^n \mathsf{E}\left[\xi_j^{p_j}\right] \quad \forall\, p_1, p_2, \cdots, p_n \in \mathbb{Z}_+ \text{ with } \sum_{i=1}^n p_i = k. \tag{3.7}$$

To relate this definition to our Hilbert's identity, we consider $2d$-wise uncorrelated random variables $\eta_1, \eta_2, \cdots, \eta_n$, where each one is identical to ξ_1 in (3.5). Therefore, we have $\mathsf{E}\left[\eta_j^p\right] = \mathsf{E}\left[\xi_1^p\right]$ and $\mathsf{E}\left[\prod_{j=1}^n \eta_j^{p_j}\right] = \prod_{j=1}^n \mathsf{E}\left[\eta_j^{p_j}\right]$ $\forall\, p \in \mathbb{P}_{2d}^n$, which lead to (3.6). In other words, $\eta_1, \eta_2, \cdots, \eta_n$ constitute another representation of Hilbert's identity. We summarize this result in the following result as preparation for the later discussion.

Proposition 3.2.2. *If $\xi_1, \xi_2, \cdots, \xi_n$ are i.i.d. random variables, and $\eta_1, \eta_2, \cdots, \eta_n$ are $2d$-wise zero-correlated, satisfying the moments constraints $\mathsf{E}\left[\eta_j^p\right] = \mathsf{E}\left[\xi_1^p\right]$ for all $0 < p \leq 2d$ and $1 \leq j \leq n$, then $\mathsf{E}\left[\left(\sum_{j=1}^n \xi_j x_j\right)^{2d}\right] = \mathsf{E}\left[\left(\sum_{j=1}^n \eta_j x_j\right)^{2d}\right]$.*

Now we can see that the key of reducing the length of representation in (3.2) is to construct $2d$-wise zero-correlated random variables satisfying certain moments conditions, such that the sample space is as small as possible.

Before addressing the issue of finding such random variables, below we shall first discuss a related notion known as k-wise independence.

Definition 3.2.3. *(**k-wise independence**) A set of random variables $\Xi = \{\xi_1, \xi_2, \cdots, \xi_n\}$ with each taking values on the set $\Delta = \{\delta_1, \delta_2, \cdots, \delta_q\}$ is called k-wise independent, if any k different random variables $\xi_{i_1}, \xi_{i_2}, \cdots, \xi_{i_k}$ of Ξ are independent, i.e.,*

$$\text{Prob}\left\{\xi_{i_1} = \delta_{i_1}, \xi_{i_2} = \delta_{i_2}, \cdots, \xi_{i_k} = \delta_{i_k}\right\} = \prod_{j=1}^{k} \text{Prob}\left\{\xi_{i_j} = \delta_{i_j}\right\} \quad \forall\, \delta_{i_j} \in \Delta,\, j = 1, 2, \ldots, k.$$

Note that when $k = 2$, k-wise independence is usually called pair-wise independence. Since 1980's, k-wise independence has been a popular topic in theoretical computer science. Essentially, working with k-wise independence (instead of the full independence) means that one can reduce the size of the sample space in question. In many cases, this feature is crucial. For instance, when $\Delta = \{0, 1\}$ and $\text{Prob}\{\xi_1 = 0\} = \text{Prob}\{\xi_1 = 1\} = \frac{1}{2}$, Alon, Babai, and Itai [66] constructed a sample space of size being approximately $n^{\frac{k}{2}}$. For the same Δ, when $\xi_1, \xi_2, \cdots, \xi_n$ are independent but not identical, Karloff and Mansour [67] proved that the size of sample space can be upper bounded by $O(n^k)$. In the case of $\Delta = \{0, 1, \ldots, q-1\}$ with q being a prime number, the total number of random variables being k-wise independent are quite restricted. For given $k < q$, Joffe [68] showed that there are up to $q+1$ random variables form a k-wise independent set and the size of the sample space is q^k.

Clearly, k-wise independence implies k-wise zero-correlation. Therefore, we may apply the existing results of k-wise independence to get k-wise zero-correlation random variables. However, the afore-mentioned constructions of k-wise independent random variables heavily depend on the structure of Δ (e.g., it requires that $|\Delta| = 2$ or $k < |\Delta|$). Moreover, the construction of k-wise independent random variables is typically complicated and technically involved (see [67]). In fact, for certain problems (e.g., polynomial-size representation of Hilbert's identity in this case), we only need the random variables to be k-wise zero-correlated. Therefore in next section, we propose a tailor-made simple construction which suits the structure of k-wise zero-correlation random variables. As we shall see later, our approach can handle more general setting of the following supporting set

$$\Delta_q := \{1, \omega_q, \cdots, \omega_q^{q-1}\}, \text{ with } \omega_q = e^{\boldsymbol{i}\frac{2\pi}{q}} = \cos\frac{2\pi}{q} + \boldsymbol{i}\sin\frac{2\pi}{q} \text{ and } q \text{ is prime}, \quad (3.8)$$

and k can be any parameter. Conceptually, our approach is rather generic: the k-wise zero-correlated random variables are constructed based only on the product of a small set of i.i.d. random variables with their powers; the sample space would be polynomial-size if the number of such i.i.d. random variables is $O(\log n)$.

3.3 Construction of k-wise Zero-Correlated Random Variables

In this section, we shall construct k-wise zero-correlated random variables, which are identical and uniformly distributed on Δ_q defined by (3.8). The rough idea is as follows. We first generate m i.i.d. random variables $\xi_1, \xi_2, \cdots, \xi_m$, based on which we can define new random variables $\eta_1, \eta_2, \cdots, \eta_n$ such that $\eta_i := \prod_{1 \le j \le m} \xi_j^{c_{ij}}$ for $i = 1, 2, \ldots, n$. Therefore, the size of sample space of $\{\eta_1, \eta_2, \cdots, \eta_n\}$ is bounded above by q^m, which yields a polynomial-size space if we let $m = O(\log_q n)$. The remaining part of this section is devoted to the discussion of the property for the power indices c_{ij}'s, in order to guarantee $\eta_1, \eta_2, \cdots, \eta_n$ to be k-wise zero-correlated, and how to find those power indices.

3.3.1 k-wise Regular Sequence

Let us start with a couple of notations and definitions for the preparation. Suppose c is a number with m digits and $c[\ell]$ is the value of its ℓ-th bit. We call c to be endowed with the base q, if $c[\ell] \in \{0, 1, \ldots, q-1\}$ for all $1 \leq \ell \leq m$. In other words, $c = \sum_{\ell=1}^{m} c[\ell] q^{\ell-1}$. Recall we have defined a subset $\mathbb{P}_k^n \subseteq \mathbb{Z}_+^n$ to be $\left\{(p_1, p_2, \cdots, p_n)^\top \in \mathbb{Z}_+^n \mid p_1 + p_2 + \cdots + p_n = k\right\}$. Then for any given prime number q, we define another set associated with q below

$$\mathbb{P}_k^n(q) := \{p \in \mathbb{P}_k^n \mid \exists\, i\, (1 \leq i \leq n) \text{ such that } q \nmid p_i\}.$$

It is easy to see that $|\mathbb{P}_k^n(q)| \leq |\mathbb{P}_k^n| = \binom{n+k-1}{k}$.

Now we can define the concept of k-wise regular sequence as follows.

Definition 3.3.1. *A sequence of m digits numbers $\{c_1, c_2, \cdots, c_n\}$ of base q is called k-wise regular if for any $p \in \mathbb{P}_k^n(q)$, there exists $\ell\, (1 \leq \ell \leq m)$ such that*

$$\sum_{j=1}^{n} p_j \cdot c_j[\ell] \neq 0 \ mod\ q.$$

Why are we interested in such regular sequences? The answer lies in the following proposition.

Proposition 3.3.2. *Suppose m digits numbers $\{c_1, c_2, \cdots, c_n\}$ of base q are k-wise regular, where q is a prime number, and $\xi_1, \xi_2, \cdots, \xi_m$ are i.i.d. random variables uniformly distributed on Δ_q. Then $\eta_1, \eta_2, \cdots, \eta_n$ with*

$$\eta_i := \prod_{1 \leq \ell \leq m} \xi_\ell^{c_i[\ell]},\ i = 1, 2, \cdots, n \tag{3.9}$$

are k-wise zero-correlated.

Proof. Let $\eta_1, \eta_2, \cdots, \eta_n$ be defined as in (3.9). As ξ_i is uniformly distributed on Δ_q for $1 \leq i \leq m$ and q is prime, we have

$$\mathsf{E}[\xi_i^p] = \mathsf{E}\left[\eta_j^p\right] = \begin{cases} 1, & \text{if } q \mid p, \\ 0, & \text{otherwise.} \end{cases}$$

For any given $p \in \mathbb{P}_k^n$, if $q \mid p_i$ for all $1 \le i \le n$, then

$$\begin{aligned}\mathsf{E}\left[\prod_{j=1}^{n}\eta_j^{p_j}\right] &= \mathsf{E}\left[\left(\prod_{1\le\ell\le m}\xi_\ell^{p_1\cdot c_1[\ell]}\right)\left(\prod_{1\le\ell\le m}\xi_\ell^{p_2\cdot c_2[\ell]}\right)\cdots\left(\prod_{1\le\ell\le m}\xi_\ell^{p_n\cdot c_n[\ell]}\right)\right] \\ &= \prod_{1\le\ell\le m}\mathsf{E}\left[\xi_\ell^{\sum_{j=1}^n p_j\cdot c_j[\ell]}\right] = 1 = \prod_{j=1}^{n}\mathsf{E}\left[\eta_j^{p_j}\right].\end{aligned}$$

Otherwise, there exists some i such that $q \nmid p_i$ (in this case $p \in \mathbb{P}_k^n(q)$), which implies that $\mathsf{E}[\eta_i^{p_i}] = 0$. Moreover by k-wise regularity, we can find some ℓ_0 satisfying $\sum_{j=1}^n p_j \cdot c_j[\ell_0] \neq 0 \bmod q$. Therefore

$$\mathsf{E}\left[\prod_{j=1}^{n}\eta_j^{p_j}\right] = \prod_{1\le\ell\le m}\mathsf{E}\left[\xi_\ell^{\sum_{j=1}^n p_j\cdot c_j[\ell]}\right] = 0 = \prod_{j=1}^{n}\mathsf{E}\left[\eta_j^{p_j}\right],$$

and the conclusion follows. □

3.3.2 A Randomized Algorithm

We shall now focus on how to find such k-wise regular sequence $\{c_1, c_2, \cdots, c_n\}$ of base q. First, we present a randomized process, in which $c_i[\ell]$ is randomly and uniformly chosen from $\{0, 1, \ldots, q-1\}$ for all $1 \le i \le n$ and $1 \le \ell \le m$. The algorithm is as follows.

Algorithm RAN

Input:	Dimension n and $m := \lceil k \log_q n \rceil$.
Output:	A sequence $\{c_1, c_2, \cdots, c_n\}$ in m digits of base q.
Step 0:	Construct $S = \{(\underbrace{0, \cdots, 0, 0}_{m}), (\underbrace{0, \cdots, 0, 1}_{m}), \cdots, (\underbrace{q-1, \cdots, q-1}_{m})\}$ of base q.
Step 1:	Independently and uniformly take $c_i \in S$ for $i = 1, 2, \ldots, n$.
Step 2:	Assemble the sequence $\{c_1, c_2, \cdots, c_n\}$ and exit.

Theorem 3.3.3. *If $1 < k < n$ and q is a prime number, then* **Algorithm RAN** *returns a k-wise m-digit regular sequence $\{c_1, c_2, \cdots, c_n\}$ of base q with probability at least $1 - \frac{(1.5)^{k-1}}{k!}$, which is independent of n and q.*

Proof. Since $\{c_1, c_2, \cdots, c_n\}$ is a sequence of m-digit numbers of base q, if it is not regular, then there exist $p \in \mathbb{P}_k^n$, such that

$$\sum_{j=1}^{n} p_j \cdot c_j[\ell] = 0 \bmod q \quad \forall\, 1 \le \ell \le m.$$

Therefore, we have

$$\begin{aligned} & \text{Prob}\left\{\{c_1, c_2, \cdots, c_n\} \text{ is not } k\text{-wise regular}\right\} \\ \le & \sum_{p \in \mathbb{P}_k^n(q)} \text{Prob}\left\{\sum_{j=1}^{n} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m\right\}. \end{aligned}$$

For any given $p \in \mathbb{P}_k^n(q)$, we may without loss of generality assume that $q \nmid p_n$. If we fix $c_1, c_2, \cdots, c_{n-1}$, as q is prime, then there is only one solution for c_n such that $\sum_{j=1}^{n} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m$. Combining the fact that $c_1, c_2, \cdots, c_n$ are independently and uniformly generated, we have

$$\begin{aligned} & \text{Prob}\left\{\sum_{j=1}^{n} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m\right\} \\ = & \text{Prob}\left\{\sum_{j=1}^{n} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m \,\middle|\, c_1 = d_1, c_2 = d_2, \cdots, c_{n-1} = d_{n-1}\right\} \cdot \\ & \sum_{d_1, d_2, \cdots, d_{n-1} \in S} \text{Prob}\left\{c_1 = d_1, c_2 = d_2, \cdots, c_{n-1} = d_{n-1}\right\} \\ = & \frac{1}{q^m} \sum_{d_1, d_2, \cdots, d_{n-1} \in S} \text{Prob}\left\{c_1 = d_1, c_2 = d_2, \cdots, c_{n-1} = d_{n-1}\right\} \\ \le & \frac{1}{n^k}. \end{aligned} \tag{3.10}$$

Finally,

$$\begin{aligned} & \text{Prob}\left\{\{c_1, c_2, \cdots, c_n\} \text{ is } k\text{-wise regular}\right\} \\ = & 1 - \text{Prob}\left\{\{c_1, c_2, \cdots, c_n\} \text{ is not } k\text{-wise regular}\right\} \\ \ge & 1 - |\mathbb{P}_k^n(q)| \cdot \frac{1}{n^k} \ge 1 - |\mathbb{P}_k^n| \cdot \frac{1}{n^k} = 1 - \binom{n+k-1}{k} \cdot \frac{1}{n^k} \ge 1 - \frac{(1.5)^{k-1}}{k!}. \end{aligned}$$

□

For some special q and k, in particular relating to the the simplest case of Hilbert's identity (4-wise regular sequence of base 2), the lower bound of the probability in Theorem 3.3.3 can be improved.

Proposition 3.3.4. *If $k = 4$ and $q = 2$, then* **Algorithm RAN** *returns a 4-wise regular sequence $\{c_1, c_2, \cdots, c_n\}$ of base 2 with probability at least $1 - \frac{1}{2n^2} - \frac{1}{4!}$.*

The proof is similar to that of Theorem 3.3.3, and thus is omitted.

3.3.3 De-Randomization

Although k-wise regular sequence always exists and can be found with high probability, one may however wish to construct such regular sequence *deterministically*. In fact, this is possible if we apply Theorem 3.3.3 in a slightly different manner, which is shown in the following algorithm. Basically, we first start with a small size of regular set C, and enumerate all the remaining numbers in order to find c such that $C \cup \{c\}$ is also regular. Updating C with $C \cup \{c\}$, and repeat this procedure until the cardinality of C reaches n. Moreover, thanks to the polynomial-size sample space, this 'brutal force' approach still runs in polynomial-time.

Algorithm DET

Input: Dimension n and $m := \lceil k \log_q n \rceil$.

Output: A sequence $\{c_1, c_2, \cdots, c_n\}$ in m digits of base q.

Step 0: Construct $S = \{(\underbrace{0, \cdots, 0, 0}_{m}), (\underbrace{0, \cdots, 0, 1}_{m}), \cdots, (\underbrace{q-1, \cdots, q-1}_{m})\}$ of base q, and a sequence $C := \{c_1, c_2, \cdots, c_k\}$ in m digits, where $c_i := (0, \cdots, 0, 1, \underbrace{0, \cdots, 0}_{k-1})$ for $i = 1, 2, \ldots, k$. Let the index count be $\tau := k$.

Step 1: If $\tau = n$, then go to Step 2; Otherwise enumerate $S \setminus C$ to find a $c \in S \setminus C$ such that $C \cup \{c\}$ is k-wise regular. Let $c_{\tau+1} := c$, $C := C \cup \{c_{\tau+1}\}$ and $\tau := \tau + 1$, and return to Step 1.

Step 2: Assemble the sequence $\{c_1, c_2, \cdots, c_n\}$ and exit.

It is obvious that the initial sequence $\{c_1, c_2, \cdots, c_k\}$ is k-wise regular. In order for **Algorithm DET** to exit successfully, it remains to argue that it is always possible to

expand the k-wise regular sequence by one in Step 1, as long as $\tau < n$.

Theorem 3.3.5. *Suppose that $3 \le k \le \tau < n$, q is a prime number, and C with $|C| = \tau$ is k-wise regular. If we* uniformly *pick $c_{\tau+1}$ from S, then*

$$\text{Prob}\left\{C \cup \{c_{\tau+1}\} \text{ is } k\text{-wise regular}\right\} \ge 1 - \frac{(1.5)^k}{k!}\left(\frac{\tau+1}{n}\right)^k,$$

ensuring that $\{c_{\tau+1} \in S \mid C \cup \{c_{\tau+1}\}$ is k-wise regular$\} \neq \emptyset$.

Proof. Like in the proof of Theorem 3.3.3, we have

$$\begin{aligned} &\text{Prob}\left\{C \cup \{c_{\tau+1}\} \text{ is not } k\text{-wise regular}\right\} \\ \le &\sum_{p \in \mathbb{P}_k^{\tau+1}(q)} \text{Prob}\left\{\sum_{j=1}^{\tau+1} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m\right\}. \end{aligned}$$

For any $p \in \mathbb{P}_k^{\tau+1}(q)$, since q is prime, by using a similar argument as of (5.4), we can get

$$\text{Prob}\left\{\sum_{j=1}^{\tau+1} p_j \cdot c_j[\ell] = 0 \bmod q, \forall\, 1 \le \ell \le m\right\} \le \frac{1}{n^k}.$$

Therefore,

$$\begin{aligned} \text{Prob}\left\{C \cup \{c_{\tau+1}\} \text{ is } k\text{-wise regular}\right\} \ge 1 - \left|\mathbb{P}_k^{\tau+1}(q)\right| \frac{1}{n^k} &\ge 1 - \binom{\tau+k}{k}\frac{1}{n^k} \\ &\ge 1 - \frac{(1.5)^k}{k!}\left(\frac{\tau+1}{n}\right)^k > 0. \end{aligned}$$

□

By the above theorem, Step 1 of **Algorithm DET** guarantees to expand k-wise regular sequence of base q before reaching the desired cardinality $\tau = n$. A straightforward computation shows that **Algorithm DET** requires an overall complexity of $O(n^{2k-1} \log_q n)$.

3.4 Polynomial-Size Representation of Hilbert's Identity

3.4.1 Polynomial-Size Representation of Quartic Hilbert's Identity

Armed with k-wise zero-correlated random variables, we are able to construct polynomial-size representation of the fourth moments tensor. In Hilbert's construction (3.5), the

supporting set Δ is too complicated to apply the result in Section 3.3. However as we mentioned earlier, such decomposition of (3.5) is not unique. In fact, when $d = 2$, we observe that

$$(x^\top x)^2 = \left(\sum_{i=1}^n x_i^2\right)^2 = \frac{2}{3}\sum_{i=1}^n x_i^4 + \frac{1}{3}\,\mathsf{E}\left[\left(\sum_{j=1}^n \xi_j x_j\right)^4\right], \tag{3.11}$$

where $\xi_1, \xi_2, \cdots, \xi_n$ are i.i.d. symmetric Bernoulli random variables. Applying either **Algorithm RAN** or **Algorithm DET** leads to a 4-wise regular sequence of base 2, based on which we can define random variables $\eta_1, \eta_2, \cdots, \eta_n$ as we did in (3.9). Proposition 3.3.2 guarantees that $\eta_1, \eta_2, \cdots, \eta_n$ are 4-wise zero-correlated, and it is easy to check that

$$\mathsf{E}[\eta_j] = \mathsf{E}[\eta_j^3] = \mathsf{E}[\xi_1] = \mathsf{E}[\xi_1^3] = 0,\ \mathsf{E}[\eta_j^2] = \mathsf{E}[\eta_j^4] = \mathsf{E}[\xi_1^2] = \mathsf{E}[\xi_1^4] = 1 \quad \forall\, 1 \le j \le n.$$

Thus, by Proposition 3.2.2, we have that $\mathsf{E}\left[\left(\sum_{j=1}^n \eta_j x_j\right)^4\right] = \mathsf{E}\left[\left(\sum_{j=1}^n \xi_j x_j\right)^4\right]$. Moreover, the size of sample space of $\{\eta_1, \eta_2, \cdots, \eta_n\}$ is at most $2^{\lceil k \log_q n \rceil} \le 2n^4$, which means the new representation has at most $n+2n^4$ fourth powered terms leading to the following main result.

Theorem 3.4.1. *Given a positive integer n, we can find τ ($\le 2n^4$) vectors $b^1, b^2, \cdots, b^\tau \in \mathbb{R}^n$ in polynomial time, such that*

$$(x^\top x)^2 = \frac{2}{3}\sum_{i=1}^n x_i^4 + \sum_{j=1}^\tau \left((b^j)^\top x\right)^4 \quad \forall\, x \in \mathbb{R}^n.$$

In fact, the result above can be extended to a more general setting as follows.

Corollary 3.4.2. *Given a positive semidefinite matrix $A \in \mathbb{R}^{n\times n}$, we can find τ ($\le 2n^4 + n$) vectors $a^1, a^2, \cdots, a^\tau \in \mathbb{R}^n$ in polynomial time, such that*

$$(x^\top A x)^2 = \sum_{i=1}^\tau \left((a^i)^\top x\right)^4 \quad \forall\, x \in \mathbb{R}^n.$$

Proof. By letting $y = A^{\frac{1}{2}}x$ and applying Theorem 3.4.1, we can find $b^1, b^2, \cdots, b^\tau$ in polynomial time with $\tau \le 2n^4$, such that

$$(x^\top A x)^2 = (y^\top y)^2 = \frac{2}{3}\sum_{i=1}^n y_i^4 + \sum_{j=1}^\tau \left((b^j)^\top y\right)^4 = \sum_{i=1}^n \left(\left(\frac{2}{3}\right)^{\frac{1}{4}} (e^i)^\top A^{\frac{1}{2}} x\right)^4 + \sum_{j=1}^\tau \left((b^j)^\top A^{\frac{1}{2}} x\right)^4.$$

The conclusion follows by letting $a^i = \left(\frac{2}{3}\right)^{\frac{1}{4}} A^{\frac{1}{2}} e^i$ for $i = 1, 2, \ldots, n$, and $a^{i+n} = A^{\frac{1}{2}} b^i$ for $i = 1, 2, \ldots, \tau$. □

3.4.2 Polynomial-Size Representation of of qd-th degree Hilbert's Identity

In this subsection we are going to generalize the result in Section 3.4.1 to qd-th degree polynomial. That is for any given positive integers q, d and n, we want to vectors $a^1, a^2, \cdots, a^t \in \mathbb{R}^n$, such that

$$\left(\sum_{i=1}^{n} x_i^q\right)^d = \sum_{j=1}^{t} \left((a^j)^\top x\right)^{qd} \quad \forall\, x \in \mathbb{R}^n. \tag{3.12}$$

Unfortunately, the above does not hold in general, as the following counter example shows.

Example 3.4.3. *The function $f(x) = (x_1^3 + x_2^3)^2 = x_1^6 + 2x_1^3x_2^3 + x_2^6$ cannot be decomposed in the form of (3.12) with $q = 3$ and $d = 2$, i.e., a sum of sixth powered linear terms.*

This can be easily proven by contradiction. Suppose we can find $a^1, a^2, \cdots, a^t \in \mathbb{R}^n$, such that

$$x_1^6 + 2x_1^3x_2^3 + x_2^6 = \sum_{i=1}^{t} (a_i x_1 + b_i x_2)^6 . \tag{3.13}$$

There must exist some (a_j, b_j) with $a_j b_j \neq 0$, since otherwise there is no monomial $x_1^3 x_2^3$ in the right hand side of (3.13). As a consequence, the coefficient of monomial $x_1^2 x_2^4$ in the right hand side of (3.13) is at least $\binom{6}{2} a_j^2 b_j^4 > 0$, which is null on the left side of the equation, leading to a contradiction.

In the same vein one can actually show that (3.12) cannot hold for any $q \geq 3$. Therefore, we turn to qd-th degree complex polynomial, i.e., both the coefficients and variables in (3.12) are now allowed to take complex values. Similar to (3.11), we have the following identity:

$$\left(\sum_{j=1}^{n} x_j^q\right)^2 = \left(1 - \frac{2}{\binom{2q}{q}}\right) \sum_{j=1}^{n} x_j^{2q} + \frac{2}{\binom{2q}{q}} \mathsf{E}\left[\left(\sum_{i=1}^{n} \xi_i x_i\right)^{2q}\right], \tag{3.14}$$

where $\xi_1, \xi_2, \cdots, \xi_n$ are i.i.d. random variables uniformly distributed on Δ_q. Moreover, we can further prove (3.12) for more general complex case.

Proposition 3.4.4. *Given any positive integers q and n, there exist $a^1, a^2, \cdots, a^\tau \in \mathbb{C}^n$ such that*

$$\left(\sum_{i=1}^{n} x_i^q\right)^{2^d} = \sum_{j=1}^{\tau} \left((a^j)^\top x\right)^{2^d q} \quad \forall\, x \in \mathbb{C}^n. \tag{3.15}$$

Proof. This proof is based on mathematical induction. The case $d = 1$ is already guaranteed by (3.14). Suppose that (3.15) is true for $d-1$, then there exist $b^1, b^2, \cdots, b^t \in \mathbb{C}^n$ such that

$$\left(\sum_{i=1}^{n} x_i^q\right)^{2^d} = \left(\left(\sum_{i=1}^{n} x_i^q\right)^{2^{d-1}}\right)^2 = \left(\sum_{j=1}^{t} \left((b^j)^\top x\right)^{2^{d-1} q}\right)^2.$$

By applying (3.14) to the above identity, there exist $c^1, c^2, \cdots, c^\tau \in \mathbb{C}^t$, such that

$$\left(\sum_{i=1}^{n} x_i^q\right)^{2^d} = \left(\sum_{j=1}^{t} \left((b^j)^\top x\right)^{2^{d-1} q}\right)^2 = \sum_{i=1}^{\tau} \left(\sum_{j=1}^{t} (c^i)_j \cdot (b^j)^\top x\right)^{2^d q} = \sum_{i=1}^{\tau} \left((c^i)^\top B^\top x\right)^{2^d q},$$

where $B = (b^1, b^2, \cdots, b^t) \in \mathbb{C}^{n \times t}$. Letting $a^i = Bc^i$ $(1 \le i \le \tau)$ completes the inductive step. □

The next step is to reduce the number τ in (3.15). Under the condition that q is prime, we can get a k-wise regular sequence of base q using either **Algorithm RAN** or **Algorithm DET**. With the help of Theorem 3.2.2, we can further get a polynomial-size representation of complex Hilbert's identity and complex $2^d q$-th moments tensor, by applying a similar argument in Theorem 3.4.1.

Theorem 3.4.5. *Given positive integers q and n with q being prime, we can find $\tau \le O\left(n^{(2q)^{2^{d-1}}}\right)$ vectors $a^1, a^2, \cdots, a^\tau \in \mathbb{C}^n$ in polynomial time, such that*

$$\left(\sum_{i=1}^{n} x_i^q\right)^{2^d} = \sum_{i=1}^{\tau} \left((a^i)^\top x\right)^{(2^d q)} \quad \forall\, x \in \mathbb{C}^n.$$

3.5 Matrix $q \mapsto p$ Norm Problem

In this section, we shall illustrate the power of polynomial-size representation of moments tensor by a specific example. In particular, we consider the problem of computing

the so-called $q \mapsto p$ $(1 \leq p, q \leq \infty)$ norm of a matrix A, defined as follows:

$$\|A\|_{q\mapsto p} := \max_{\|x\|_q=1} \|Ax\|_p.$$

This problem can be viewed as a natural extension of several useful problems. For instance, the case $p = q = 2$ corresponds to the largest singular value of A. The case $(p, q) = (1, \infty)$ corresponds to the bilinear optimization problem in binary variables, which is related to the so-called matrix cut norm and Grothendieck's constant; see Alon and Naor [69]. In case $p = q$, the problem becomes the matrix p-norm problem, which has applications in scientific computing; cf. [70].

In terms of the computational complexity, three easy cases are well known: (1) $q = 1$ and $p \geq 1$ is a rational number; (2) $p = \infty$ and $q \geq 1$ is a rational number; (3) $p = q = 2$. Steinberg [71] showed that computing $\|A\|_{q\mapsto p}$ is NP-hard for general $1 \leq p < q \leq \infty$, and she further conjectured that the above mentioned three cases are the only exceptional easy cases where the matrix $q \mapsto p$ norm can be computed in polynomial time. Hendrickx and Olshevsky [72] made some progress along this line by figuring out the complexity status of the "diagonal" case of $p = q$. Moreover, very recently Bhaskara and Vijayaraghavan [42] proved that this problem is NP-hard to approximate to any constant factor when $2 < p \leq q$. However, the problem of determining the complexity status for the case $p > q$ still remains open. Here we shall show that the problem $\|A\|_{q\mapsto p}$ is NP-hard when $q = 2$ and $p = 4$. To this end, let us first present the following lemma.

Lemma 3.5.1. *Given positive integers n, i, j with $1 \leq i < j \leq n$, we can find $t\,(\leq 2n^4 + n + 2)$ vectors $a^1, a^2, \cdots, a^t$ in polynomial time, such that*

$$2x_i^2 x_j^2 + (x^\top x)^2 = \sum_{k=1}^{t} \left((a^k)^\top x\right)^4.$$

Proof. Recall in Theorem 3.4.1, we can find $\tau\,(\leq 2n^4)$ vectors $a^1, a^2, \cdots, a^\tau \in \mathbb{R}^n$ in polynomial time, such that

$$\frac{2}{3}\sum_{\ell=1}^{n} x_\ell^4 + \sum_{\ell=1}^{\tau} \left((a^\ell)^\top x\right)^4 = (x^\top x)^2. \tag{3.16}$$

On the other hand, one verifies straightforwardly that for $1 \le i \ne j \le n$ we have

$$\frac{1}{2}\left((x_i + x_j)^4 + (x_i - x_j)^4\right) + x_i^4 + x_j^4 + 2\sum_{1\le \ell \le n,\, \ell \ne i,j} x_\ell^4 = 6x_i^2 x_j^2 + 2\sum_{\ell=1}^{n} x_\ell^4. \qquad (3.17)$$

Dividing by 3 on both sides of (3.17) and then summing up with $\sum_{\ell=1}^{\tau}\left((a^\ell)^\top x\right)^4$ yields

$$\begin{aligned} & \sum_{\ell=1}^{\tau}\left((a^\ell)^\top x\right)^4 + \frac{1}{3}\left(\frac{1}{2}\left((x_i + x_j)^4 + (x_i - x_j)^4\right) + x_i^4 + x_j^4 + 2\sum_{1\le \ell \le n,\, \ell \ne i,j} x_\ell^4\right) \\ = & \sum_{\ell=1}^{\tau}\left((a^\ell)^\top x\right)^4 + 2x_i^2 x_j^2 + \frac{2}{3}\sum_{\ell=1}^{n} x_\ell^4 \\ = & \ 2x_i^2 x_j^2 + (x^\top x)^2, \end{aligned}$$

where the last equality is due to (3.16). □

Now we are in a position to prove the main theorem of this section.

Theorem 3.5.2. *Computing* $\|A\|_{2\mapsto 4} = \max_{\|x\|_2=1} \|Ax\|_4$ *is NP-hard.*

Proof. The reduction is made from computing the maximum (vertex) independence set of a graph. In particular, for a given graph $G = (V, E)$, Nesterov [26] showed that the following problem can be reduced from the maximum independence number problem:

$$\begin{array}{ll} \max & 2\sum_{(i,j)\in E,\, i<j} x_i^2 x_j^2 \\ \text{s.t.} & \|x\|_2 = 1,\ x \in \mathbb{R}^n, \end{array}$$

hence is NP-hard. Moreover, the above is obviously equivalent to

$$\begin{array}{lll} (P) & \max & 2\sum_{(i,j)\in E,\, i<j} x_i^2 x_j^2 + |E| \cdot \|x\|_2^4 = \sum_{(i,j)\in E,\, i<j}\left(2x_i^2 x_j^2 + (x^\top x)^2\right) \\ & \text{s.t.} & \|x\|_2 = 1,\ x \in \mathbb{R}^n. \end{array}$$

By Lemma 3.5.1, the objective in (P) can be expressed by no more than $|E| \cdot (2n^4 + n + 2)$ number of fourth powered linear terms, making (P) be an instance of $\|A\|_{2\mapsto 4}$ (polynomial-size). The polynomial reduction is thus complete. □

Suppose that p' and q' are the *conjugates* of p and q respectively, i.e., $\frac{1}{p} + \frac{1}{p'} = 1$ and $\frac{1}{q} + \frac{1}{q'} = 1$. By using the fact that $\|x\|_p = \max_{\|y\|_{p'}=1} y^\top x$, one can prove that $\|A\|_{q\mapsto p} = \|A^\top\|_{p'\mapsto q'}$. Therefore, Theorem 3.5.2 implies that computing $\|A\|_{\frac{4}{3}\mapsto 2}$ is also NP-hard.

Chapter 4

Matrix-Rank of Even Order Tensors

4.1 Introduction

Due to the emergence of multidimensional data in computer vision, medical imaging, machine learning, quantum entanglement problems, signal processing and web data, tensor-based multidimensional data analysis has attracted more and more attentions. On the other hand, in practice, the tensor formed by the underlying multidimensional data often bears some low-rank structure, although the actual data may not appear so due to arbitrary errors. Therefore, it becomes extremely important to understand the rank of tensors. Different from the matrix, the definition of tensor rank is not unique. One most commonly used of such notions is the so-called *CP rank*, which is a natural extension of the rank of the matrix.

Definition 4.1.1. *Given a tensor $\mathcal{F} \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_d}$, the CP rank of $\mathcal{F}$ in denoted by* $\operatorname{rank}_{CP}(\mathcal{F})$ *is the smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r} a^{1,i} \otimes a^{2,i} \otimes \cdots \otimes a^{d,i}, \tag{4.1}$$

where $a^{k,i} \in \mathbb{C}^{n_i}$ for $k = 1, \ldots, d$ and $i = 1, \ldots, r$.

The idea of decomposing a tensor into an (asymmetric) outer product of vectors can be backed to 1927 [73, 74]. This concept becomes popular after the rediscovery in 1970s

in the form of CANDECOMP (canonical decomposition) by Carroll and Chang [23] and PARAFAC (parallel factors) by Harshman [24]. Like many following up literatures of [23, 24] , *CP* refers to the abbreviations of CANDECOMP and PARAFAC.

But determining the rank of a specific given tensor is already a difficult task, which is NP-hard in general [25]. To give an impression of the difficulty involved in computing tensor ranks, note that there is a particular $9 \times 9 \times 9$ tensor (cf. [75]) whose rank is only known to be in between 18 and 23.

One way to deal with this difficulty is to unfold the tensor into a matrix in some way and the rank of the matrix is easy to compute. A typical matricization technique is the so-called mode-n matricization [76]. Roughly speaking, given a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$, its mode-$n$ matricization denoted by $A(n)$ is to arrange n-th index of $\mathcal{A}$ to be the column index of resulting matrix and merge other indices of $\mathcal{A}$ into the row index of $A(n)$. The so-called n-rank of $\mathcal{A}$ is the vector with m dimension such that its n-th component corresponds to the column rank of mode-n matrix $A(n)$. The notion of n-rank has been widely used in the problems of tensor decomposition. Most recently, Liu *et al.* [49] and Gandy *et al.* [77] considered the low-n-rank tensor recovery problem, which were the first attempts to solve low-rank tensor optimization problems.

However, up till now, the relationship between the n-rank and CP rank is still unclear. Therefore, in the following we shall introduce a new scheme to unfold a tensor into a matrix, where we use half of the indices of tensor to form the row index of a matrix and use the other half as the column index. Most importantly, in the next a few sections, we manage to establish some connection between the CP rank of the tensor and the rank of the resulting unfolding matrix. We first introduce the following notion of square unfolding for an even order tensor.

Definition 4.1.2. *The square unfolding for an even order tensor* $\mathcal{F} \in \mathbb{C}^{n_1 \times n_2 \cdots \times n_{2d}}$ *denoted by* $\boldsymbol{M}(\mathcal{F}) \in \mathbb{C}^{(n_1 \cdots n_d) \times (n_{d+1} \cdots n_{2d})}$*, is defined as*

$$\boldsymbol{M}(\mathcal{F})_{k\ell} := \mathcal{F}_{i_1 \cdots i_{2d}},$$

where

$$k = \sum_{j=2}^{d} (i_j - 1) \prod_{q=1}^{j-1} n_q + i_1, 1 \leq i_j \leq n_j, 1 \leq j \leq d,$$

$$\ell = \sum_{j=d+2}^{2d} (i_j - 1) \prod_{q=d+1}^{j-1} n_q + i_{d+1}, , d+1 \leq j \leq 2d.$$

Notice that $\boldsymbol{M}(\mathcal{F})$ is a matrix, whose rank is well defined and easy to compute. However, such way of unfolding is not unique. To see this, let's look at the permutated tensor $\mathcal{F}_\pi$ of $\mathcal{F}$ under π, where π is a permutation operator on $(1 \cdots 2d)$ and the set of all such operators is denoted by Π_{2d}, i.e.:

$$(\mathcal{F}_\pi)_{i_1 \cdots i_{2d}} = \mathcal{F}_{\pi(i_1, \cdots, i_{2d})}.$$

Now we can see that the resulting unfolding matrix $\boldsymbol{M}(\mathcal{F}_\pi)$ are different, although the tensors $\mathcal{F}_\pi$ for any π, correspond to the same tensor $\mathcal{F}$. Taking this situation into our consideration, we propose a new rank definition for tensor called matrix-rank.

Definition 4.1.3. *Given an even order tensor mathcal$F \in \mathbb{C}^{n_1 \times n_2 \cdots \times n_{2d}}$, the matrix-rank of $\mathcal{F}$ denoted by* $\operatorname{rank}_M(\mathcal{F})$ *is the smallest rank of all the possible unfolding matrices, i.e.*

$$\operatorname{rank}_M(\mathcal{F}) = \min_{\pi \in \Pi_{2d}} \operatorname{rank}\left(\boldsymbol{M}(\mathcal{F}_\pi)\right).$$

In other words, $\operatorname{rank}_M(\mathcal{F})$ *is the smallest integer r such that*

$$\mathcal{F}_\pi = \sum_{i=1}^{r} \mathcal{A}^i \otimes \mathcal{B}^i,$$

holds for some permutation $\pi \in \Pi_{2d}$, where $\mathcal{A}^i \in \mathbb{C}^{n_{j_1} \times \cdots \times n_{j_d}}$ and $\mathcal{B}^i \in \mathbb{C}^{n_{j_{d+1}} \times \cdots \times n_{j_{2d}}}$ with $(j_1, \cdots, j_{2d}) = \pi(1, \cdots, 2d)$.

In Chapter 1, we have introduced super-symmetric tensor in the real field. For the tensors in the complex field, we can define the super-symmetry in the same manner and denote the set of all m-th order complex super-symmetric tensor by $\mathbb{S}^{n^m}$. For a super-symmetric tensor $\mathcal{F}$, we may want to find a better rank-one decomposition than that in (4.1), which leads to the following notion of symmetric CP rank.

Definition 4.1.4. *Given $\mathcal{F} \in \mathbb{S}^{n^m}$, the symmetric CP rank of $\mathcal{F}$ denoted by* $\operatorname{rank}_{SCP}(\mathcal{F})$ *is the smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r} \underbrace{a^i \otimes \cdots \otimes a^i}_{m},$$

with $a_i \in \mathbb{C}^n$.

On the other hand, we can decompose $\mathcal{F}$ by following the rule of (4.1) and get an asymmetric CP rank. Obviously, the asymmetric CP rank is definitely less than the symmetric one. Therefore, a natural question arises: whether these two ranks are equivalent? In the matrix case, the answer is affirmative i.e., rank and symmetric rank of a matrix are equal. However, when regarding the case of higher order tensor, it becomes an open problem proposed by Comon *et al.* [78]:

Question 4.1.1. *For a super-symmetric tensor, is it true that its asymmetric CP rank is equal to the symmetric one?*

Due to the super-symmetric property, $\mathcal{F}_\pi = \mathcal{F}$ for any $\pi \in \Pi(1 \cdots 2d)$. Therefore, we can define symmetric matrix-rank of a super-symmetric matrix as shown below.

Definition 4.1.5. *Given $\mathcal{F} \in \mathbb{S}^{n^{2d}}$, the symmetric matrix-rank of $\mathcal{F}$ denoted by* $\operatorname{rank}_{SM}(\mathcal{F})$ *is the rank of the symmetric matrix $\boldsymbol{M}(\mathcal{F})$, i.e.* $\operatorname{rank}_{SM}(\mathcal{F}) = \operatorname{rank}(\boldsymbol{M}(\mathcal{F}))$*, or equivalently is the smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r} \mathcal{B}^i \otimes \mathcal{B}^i, \tag{4.2}$$

where $\mathcal{B}^i \in \mathbb{C}^{n^d}$ for all $i = 1, \ldots, r$.

In the same vein, we can convert a matrix with appropriate dimensions to a tensor. In other words, the inverse of the operator $\boldsymbol{M}(\cdot)$ can be defined in the same manner.

Quite different from the CP rank, asymmetric matrix-rank and symmetric matrix-rank of a super-symmetric tensor $\mathcal{F}$ are equivalent, since they both correspond to the rank of the same matrix $\boldsymbol{M}(\mathcal{F})$. Notice in the decomposition (4.2), we don't need the tensor $\mathcal{A}^i$ to be super-symmetric. However, we can impose such requirement and get the notion of *strongly symmetric matrix-rank*.

Definition 4.1.6. *Given $\mathcal{F} \in \mathbb{S}^{n^{2d}}$, the strongly symmetric matrix-rank of $\mathcal{F}$ denoted by* $\operatorname{rank}_{SSM}(\mathcal{F})$ *is smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r} \mathcal{A}^i \otimes \mathcal{A}^i, \text{ with } \mathcal{A} \in \mathbb{S}^{n^d} \ \forall\, i = 1, \ldots, r. \tag{4.3}$$

Actually, we shall see in Section 4.2 that strongly symmetric matrix-rank and standard symmetric matrix-rank are equivalent. Then this property can be applied to establish some relationship between matrix-rank and CP rank in Section 4.3. Furthermore, we manage to show the rank-one equivalence between these two ranks for real valued super-symmetric tensor in the last section of this chapter.

4.2 Some Properties about Strongly Symmetric Matrix Rank

In this section, we shall show a very nice property of matrix for even order super-symmetric tensor.

Theorem 4.2.1. *Given an even order super-symmetric tensor $\mathcal{F} \in \mathbb{S}^{n^{2d}}$, its symmetric and strongly symmetric matrix-rank are the same, i.e.* $\operatorname{rank}_{SSM}(\mathcal{F}) = \operatorname{rank}_{SM}(\mathcal{F})$.

We remark that, the theorem above combined with that fact that $\operatorname{rank}_{M}(\mathcal{F}) = \operatorname{rank}_{SM}(\mathcal{F})$ implies that, for the super-symmetric tensor $\mathcal{F}$, the three types of matrix-rank are the *same*, i.e., $\operatorname{rank}_{M}(\mathcal{F}) = \operatorname{rank}_{SM}(\mathcal{F}) = \operatorname{rank}_{SSM}(\mathcal{F})$. Therefore, in the following sections of this chapter, we refer to the matrix-rank as the strongly symmetric matrix-rank for super-symmetric tensor.

The rest of this section is devoted to the proof of Theorem 4.2.1. In order to proceed, we need to introduce some new notations and technical preparations. We call a tensor $\mathcal{F} \in \mathbb{C}^{n^m}$ is symmetric with respect to indices $\{1, \ldots, d\}$, if

$$\mathcal{F}_{i_1,\cdots,i_d,i_{d+1},\cdots,i_m} = \mathcal{F}_{\pi(i_1,\cdots,i_d),i_{d+1},\cdots,i_m},$$

where $\pi \in \Pi(1, \cdots, d)$. In the following, we denote $\pi_{i,j} \in \Pi(1, \cdots, m)$ to be the permutation that exchange the order of i-th and j-th components and hold the positions of others. Then we have some easy facts below.

Lemma 4.2.2. *Suppose a tensor $\mathcal{F} \in \mathbb{C}^{n^m}$ is symmetric with respect to indices $\{1, \ldots, d\}$. Then the tensor*

$$\mathcal{F} + \sum_{j=1}^{d} \mathcal{F}_{\pi_{j,d+1}} \tag{4.4}$$

is symmetric with respect to indices $\{1, \ldots, d+1\}$.

Lemma 4.2.3. *For a given tensor $\mathcal{F} \in \mathbb{C}^{n^m}$ being symmetric with respect to indices $\{1, \ldots, d\}$, we have*

$$\begin{aligned}\left(\sum_{j=1}^{k}\left(\mathcal{F}-\mathcal{F}_{\pi_{j,d+1}}\right)\right)_{\pi_{\ell,d+1}} &= k\cdot\mathcal{F}_{\pi_{\ell,d+1}}-\sum_{j\neq\ell}\mathcal{F}_{\pi_{j,d+1}}-\mathcal{F}\\ &= -k\left(\mathcal{F}-\mathcal{F}_{\pi_{\ell,d+1}}\right)+\sum_{j\neq\ell}\left(\mathcal{F}-\mathcal{F}_{\pi_{j,d+1}}\right). \qquad (4.5)\end{aligned}$$

Now we are in position to present a key lemma below

Lemma 4.2.4. *Suppose $\mathcal{F} \in \mathbb{S}^{n^{2m}}$ and*

$$\mathcal{F}=\sum_{i=1}^{r}\mathcal{B}^i\otimes\mathcal{B}^i, \text{ where } \mathcal{B}^i \in \mathbb{C}^{n^m} \text{ is symmetric with respect to } \{1,\ldots,d\}.$$

Then there exist tensors $\mathcal{A}^i \in \mathbb{C}^{n^m}$, which is symmetric with respect to $\{1, \ldots, d+1\}$, for $i = 1, \ldots, r$ such that

$$\mathcal{F}=\sum_{i=1}^{r}\mathcal{A}^i\otimes\mathcal{A}^i.$$

Proof. Construct $\mathcal{A}^i = \frac{1}{d+1}\left(\mathcal{B}^i + \sum_{j=1}^{d}\mathcal{B}^i_{\pi_{j,d+1}}\right)$ such that $\mathcal{A}^i$ is symmetric with respect to $\{1, \ldots, d+1\}$, for $i = 1, \ldots, r$, due to Lemma 4.2.2. As a result,

$$\mathcal{B}^i=\mathcal{A}^i+\sum_{j=1}^{d}\mathcal{C}^i_j, \text{ with } \quad \mathcal{C}^i_j=\frac{1}{d+1}\left(\mathcal{B}^i-\mathcal{B}^i_{\pi_{j,d+1}}\right).$$

Since $\mathcal{F}$ is super-symmetric, $\mathcal{F} = \mathcal{F}_{\pi_{1,d+1}} = \mathcal{F}_{\pi_{m+1,m+d+1}} = \mathcal{F}_{\pi_{1,d+1}\pi_{m+1,m+d+1}}$, which is to say

$$\mathcal{F}=\sum_{i=1}^{r}\mathcal{B}^i\otimes\mathcal{B}^i=\sum_{i=1}^{r}\mathcal{B}^i\otimes\mathcal{B}^i_{\pi_{1,d+1}}=\sum_{i=1}^{r}\mathcal{B}^i_{\pi_{1,d+1}}\otimes\mathcal{B}^i=\sum_{i=1}^{r}\mathcal{B}^i_{\pi_{1,d+1}}\otimes\mathcal{B}^i_{\pi_{1,d+1}} \qquad (4.6)$$

By Lemma 4.5, we have

$$\mathcal{B}^i_{\pi_{1,d+1}}=\left(\mathcal{A}^i+\sum_{j=1}^{d}\mathcal{C}^i_j\right)_{\pi_{1,d+1}}=\mathcal{A}^i+\sum_{j=2}^{d}\mathcal{C}^i_j-d\cdot\mathcal{C}^i_1.$$

Plugging this equality into formula (4.6) yields

$$\begin{aligned}
\mathcal{F} &= \sum_{i=1}^{r} \mathcal{B}^i \otimes \mathcal{B}^i = \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=1}^{d} \mathcal{C}_j^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=1}^{d} \mathcal{C}_j^i \right) && (4.7)\\
&= \sum_{i=1}^{r} \mathcal{B}^i \otimes \mathcal{B}^i_{\pi_{1,d+1}} = \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=1}^{d} \mathcal{C}_j^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i - d \cdot \mathcal{C}_1^i \right) && (4.8)\\
&= \sum_{i=1}^{r} \mathcal{B}^i_{\pi_{1,d+1}} \otimes \mathcal{B}^i = \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i - d \cdot \mathcal{C}_1^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=1}^{d} \mathcal{C}_j^i \right) && (4.9)\\
&= \sum_{i=1}^{r} \mathcal{B}^i_{\pi_{1,d+1}} \otimes \mathcal{B}^i_{\pi_{1,d+1}}\\
&= \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i - d \cdot \mathcal{C}_1^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i - d \cdot \mathcal{C}_1^i \right). && (4.10)
\end{aligned}$$

Therefore, it can be checked that

$$\frac{(4.10) + d \times (4.9) + d \times (4.8) + d^2 \times (4.7)}{1 + 2d + d^2} \Rightarrow \mathcal{F} = \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i \right).$$

Now, we again use the fact that $\mathcal{F} \in \mathbb{S}^{m^{2d}}$, hence $\mathcal{F} = \mathcal{F}_{\pi_{2,d+1}} = \mathcal{F}_{\pi_{m+2,m+d+1}} = \mathcal{F}_{\pi_{2,d+1}\pi_{m+2,m+d+1}}$. We can apply the argument above by replacing $\mathcal{B}^i$ with $\mathcal{A}^i + \sum_{j=2}^{d} \mathcal{C}_j^i$ and get $\mathcal{F} = \sum_{i=1}^{r} \left(\mathcal{A}^i + \sum_{j=3}^{d} \mathcal{C}_j^i \right) \otimes \left(\mathcal{A}^i + \sum_{j=3}^{d} \mathcal{C}_j^i \right)$. Finally, we can repeat the procedure above until $\mathcal{F} = \mathcal{A}^i \otimes \mathcal{A}^i$ and conclusion follows. □

Finally, Theorem 4.2.1 can be viewed as a direct consequence of Lemma 4.2.4.

Proof of Theorem 4.2.1: Suppose $\text{rank}_{SM}(\mathcal{F}) = r$, that is there exist r tensors $\mathcal{B}^i \in \mathbb{C}^{n^d}$ such that $\mathcal{F} = \sum_{i=1}^{r} \mathcal{B}^i \otimes \mathcal{B}^i$. By applying Lemma 4.2.4 at most d times, we can find r super-symmetric tensors $\mathcal{A}^i \in \mathbb{S}^{n^d}$ such that $\mathcal{F} = \sum_{i=1}^{r} \mathcal{A}^i \otimes \mathcal{A}^i$. As a result, we have $\text{rank}_{SSM}(\mathcal{F}) \leq r = \text{rank}_{SM}(\mathcal{F})$. On the other hand, it's obvious that $\text{rank}_{SM}(\mathcal{F}) \leq \text{rank}_{SSM}(\mathcal{F})$. Combining these two inequalities leads to the conclusion. □

4.3 Bounding CP Rank through Matrix Rank

In this section, we focus on fourth order tensor and will show that CP rank can be both lower and upper bounded by the matrix-rank multiplied by a constant related to dimension n. Let's first look at the asymmetric case.

Theorem 4.3.1. *Suppose $\mathcal{F} \in \mathbb{C}^{n_1 \times n_2 \times n_3 \times n_4}$ with $n_1 \leq n_2 \leq n_3 \leq n_4$. Then it holds that*

$$\operatorname{rank}_M(\mathcal{F}) \leq \operatorname{rank}_{CP}(\mathcal{F}) \leq n_1 n_3 \cdot \operatorname{rank}_M(\mathcal{F}).$$

Proof. Suppose $\operatorname{rank}_{CP}(\mathcal{F}) = r$, i.e.

$$\mathcal{F} = \sum_{i=1}^{r} a^{1,i} \otimes a^{2,i} \otimes a^{3,i} \otimes a^{4,i} \text{ with } a^{k,i} \in \mathbb{C}^{n_i} \text{ for } k = 1, \ldots, 4 \text{ and } i = 1, \ldots, r.$$

By letting $A^i = a^{1,i} \otimes a^{2,i}$ and $B^i = a^{3,i} \otimes a^{4,i}$, we get $\mathcal{F} = \sum_{i=1}^{r} A^i \otimes B^i$. Thus $\operatorname{rank}_M(\mathcal{F}) \leq r = \operatorname{rank}_{CP}(\mathcal{F})$.

On the other hand, suppose that the matrix-rank of $\mathcal{F}$ is r_M. So there exist $(j_1, j_2, j_3, j_4) = \pi(1, 2, 3, 4)$ for some $\pi \in \Pi(1, 2, 3, 4)$ such that

$$\mathcal{F} = \sum_{i=1}^{r_M} A^i \otimes B^i \text{ with } A^i \in \mathbb{C}^{n_{j_1} \times n_{j_2}}, B^i \in \mathbb{C}^{n_{j_3} \times n_{j_4}} \text{ for } i = 1, \ldots, r_M.$$

Then $\operatorname{rank}(A^i) \leq \ell_1$ and $\operatorname{rank}(B^i) \leq \ell_2$ for all $i = 1, \ldots, r_M$, where $\ell_1 = \min\{n_{j_1}, n_{j_2}\}$ and $\ell_2 = \min\{n_{j_3}, n_{j_4}\}$. This is to say matrices A^i and B^i can be further decomposed as the summation of at most ℓ_1 and ℓ_2 rank-one terms respectively. Therefore, $\mathcal{F}$ can be decomposed as the summation of at most $r_M \ell_1 \ell_2$ rank-one tensors, which is to say $\operatorname{rank}_{CP}(\mathcal{F}) \leq \min\{n_{j_1}, n_{j_2}\} \cdot \min\{n_{j_3}, n_{j_4}\} \cdot \operatorname{rank}_M(\mathcal{F}) \leq n_1 n_3 \cdot \operatorname{rank}_M(\mathcal{F})$. □

To present the result for the super-symmetric tensor, we shall first provide some technical preparations.

Lemma 4.3.2. *Suppose $\sum_{i=1}^{r} A^i \otimes A^i = \mathcal{F} \in \mathbb{S}^{n^4}$ with $A^i = \sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i}$ and $a^{j_i} \in \mathbb{C}^n$ for $i = 1, \ldots, r$, $j_i = 1, \ldots, m_i$. Then it holds that*

$$\begin{aligned} \mathcal{F} \;=\; & \sum_{i=1}^{r} \sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i} \otimes a^{j_i} \otimes a^{j_i} + \\ & \sum_{i=1}^{r} \sum_{j_i \neq k_i} \tfrac{1}{3} \left(a^{j_i} \otimes a^{j_i} \otimes a^{k_i} \otimes a^{k_i} + a^{j_i} \otimes a^{k_i} \otimes a^{j_i} \otimes a^{k_i} + a^{j_i} \otimes a^{k_i} \otimes a^{k_i} \otimes a^{j_i} \right) \end{aligned}$$

Proof. Since $\mathcal{F}$ is super-symmetric, $\mathcal{F}_{ijk\ell} = \mathcal{F}_{ikj\ell} = \mathcal{F}_{i\ell kj}$. Consequently,

$$\begin{aligned}
\mathcal{F} &= \sum_{i=1}^{r} A^i \otimes A^i \\
&= \sum_{i=1}^{r} \left(\sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i} \right) \\
&= \sum_{i=1}^{r} \left(\sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i} \otimes a^{j_i} \otimes a^{j_i} + \sum_{j_i \neq k_i} a^{j_i} \otimes a^{j_i} \otimes a^{k_i} \otimes a^{k_i} \right) \qquad (4.11) \\
&= \sum_{i=1}^{r} \left(\sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i} \otimes a^{j_i} \otimes a^{j_i} + \sum_{j_i \neq k_i} a^{j_i} \otimes a^{k_i} \otimes a^{j_i} \otimes a^{k_i} \right) \qquad (4.12) \\
&= \sum_{i=1}^{r} \left(\sum_{j_i=1}^{m_i} a^{j_i} \otimes a^{j_i} \otimes a^{j_i} \otimes a^{j_i} + \sum_{j_i \neq k_i} a^{j_i} \otimes a^{k_i} \otimes a^{k_i} \otimes a^{j_i} \right) \qquad (4.13)
\end{aligned}$$

Then the conclusion follows by dividing the summation of (4.11), (4.12) and (4.13) by 3. □

Lemma 4.3.3. *Suppose $a_1, \cdots, a_m$ are m vectors and $\xi_1, \cdots, \xi_m$ are the i.i.d. symmetric Bernoulli random variables. Then it holds that*

$$\begin{aligned}
&\mathsf{E}\left[\left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \right] \\
= &\sum_{j=1}^{m} a^j \otimes a^j \otimes a^j \otimes a^j + \sum_{i \neq j} \left(a^i \otimes a^i \otimes a^j \otimes a^j + a^i \otimes a^j \otimes a^i \otimes a^j + a^i \otimes a^j \otimes a^j \otimes a^i \right)
\end{aligned}$$

Proof. First of all, we can write out the formula of the expectation term

$$\begin{aligned}
&\mathsf{E}\left[\left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j \right) \right] \\
= &\sum_{i,j,k,\ell} \mathsf{E}\left[\xi_i a^i \otimes \xi_j a^j \otimes \xi_k a^k \otimes \xi_\ell a^\ell \right] \\
= &\sum_{i,j,k,\ell} \mathsf{E}\left[\xi_i \xi_j \xi_k \xi_\ell \right] a^i \otimes a^j \otimes a^k \otimes a^\ell. \qquad (4.14)
\end{aligned}$$

Since $\xi_1, \cdots, \xi_m$ are the i.i.d. with $\mathsf{E}[\xi_1] = 0$,

$$\mathsf{E}\left[\xi_i\xi_j\xi_k\xi_\ell\right] = \begin{cases} 1, & \text{when } i=j=k=\ell \text{ or there are two nonintersect pairs in } \{i,j,k,\ell\} \\ 0, & \text{otherwise.} \end{cases} \tag{4.15}$$

Therefore,

$$\begin{aligned} & \sum_{i,j,k,\ell} \mathsf{E}\left[\xi_i\xi_j\xi_k\xi_\ell\right] a^i \otimes a^j \otimes a^k \otimes a^\ell \\ = & \sum_{j=1}^{n} a^j \otimes a^j \otimes a^j \otimes a^j + \sum_{i<j} \big(a^i \otimes a^i \otimes a^j \otimes a^j + a^j \otimes a^j \otimes a^i \otimes a^i \\ & + a^i \otimes a^j \otimes a^i \otimes a^j + a^j \otimes a^i \otimes a^j \otimes a^i + a^i \otimes a^j \otimes a^j \otimes a^i + a^j \otimes a^i \otimes a^i \otimes a^j\big) \\ = & \sum_{j=1}^{m} a^j \otimes a^j \otimes a^j \otimes a^j + \sum_{i\neq j} \big(a^i \otimes a^i \otimes a^j \otimes a^j + a^i \otimes a^j \otimes a^i \otimes a^j + a^i \otimes a^j \otimes a^j \otimes a^i\big) \end{aligned}$$

Plug the above equality into formula (4.14) and the conclusion follows. □

Suppose tensor $\mathcal{F} \in \mathbb{S}^{n^4}$ and $\operatorname{rank}_M(\mathcal{F}) = r$. By Theorem 4.2.1, $\mathcal{F} = \sum_{i=1}^{r} A^i \otimes A^i$ with $A^i = \sum_{j=1}^{m_i} a^{i_j} \otimes a^{i_j}$ for $i = 1, \ldots, r$. Let $\xi_{1_1}, \cdots, \xi_{1_{m_1}}, \xi_{2_1}, \cdots, \xi_{r_{m_r}}$ be the i.i.d. symmetric Bernoulli random variables. Combining Lemma 4.3.2 and 4.3.3 gives us:

$$\begin{aligned} \mathcal{F} = \sum_{i=1}^{r} A^i \otimes A^i = & \frac{2}{3} \sum_{i=1}^{r} \sum_{j=1}^{m_i} a^{i_j} \otimes a^{i_j} \otimes a^{i_j} \otimes a^{i_j} + \\ & \frac{1}{3} \sum_{i=1}^{r} \mathsf{E}\left[\left(\sum_{j=1}^{m_i} \xi_{i_j} a^{i_j}\right) \otimes \left(\sum_{j=1}^{m_i} \xi_{i_j} a^{i_j}\right) \otimes \left(\sum_{j=1}^{m_i} \xi_{i_j} a^{i_j}\right) \otimes \left(\sum_{j=1}^{m_i} \xi_{i_j} a^{i_j}\right)\right] \end{aligned} \tag{4.16}$$

which is a rank-one representation of $\mathcal{F}$. However, from the discussion in Chapter 3, we know that the condition of all ξ_is are independent is too strong and we only need ξ_is to be 4-wise zero-correlated. In fact, we consider a slightly stronger condition here, which is 4-wise independence (see Section 3.2 for details). In particular, we can find 4-wise independent random variables $\eta_{i_1}, \cdots, \eta_{i_{m_i}}$ such that

$$\begin{aligned} & \mathsf{E}\left[\left(\sum_{j=1}^{m} \xi_j a^j\right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j\right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j\right) \otimes \left(\sum_{j=1}^{m} \xi_j a^j\right)\right] \\ = & \mathsf{E}\left[\left(\sum_{j=1}^{m} \eta_j a^j\right) \otimes \left(\sum_{j=1}^{m} \eta_j a^j\right) \otimes \left(\sum_{j=1}^{m} \eta_j a^j\right) \otimes \left(\sum_{j=1}^{m} \eta_j a^j\right)\right]. \end{aligned} \tag{4.17}$$

Moreover, for m random variables which are 4-wise independent, Alon, Babai, and Itai [66] constructed a sample space of size being less than $4m^2$. Now we can plug (4.17) into (4.16) and get

$$\begin{aligned}\mathcal{F} &= \frac{2}{3}\sum_{i=1}^{r}\sum_{j=1}^{m_i} a^{i_j}\otimes a^{i_j}\otimes a^{i_j}\otimes a^{i_j} + \\ &\quad \frac{1}{3}\sum_{i=1}^{r}\mathsf{E}\left[\left(\sum_{j=1}^{m_i}\eta_{i_j}a^{i_j}\right)\otimes\left(\sum_{j=1}^{m_i}\eta_{i_j}a^{i_j}\right)\otimes\left(\sum_{j=1}^{m_i}\eta_{i_j}a^{i_j}\right)\otimes\left(\sum_{j=1}^{m_i}\eta_{i_j}a^{i_j}\right)\right],\end{aligned}$$

where $\eta_{i_1},\cdots,\eta_{i_{m_i}}$ are 4-wise independent, the amount of rank-one terms on the right side is less than $\sum_{i=1}^{r} m_i + 4\sum_{i=1}^{r} {m_i}^2 \leq r(n+4n^2)$, and the inequality is due to the fact $m_i = \text{rank}\,(A^i) \leq n$. Therefore, we manage to reduce the number of rank-one terms in the representation (4.16) and get following main theorem of this section.

Theorem 4.3.4. *Suppose $\mathcal{F} \in \mathbb{S}^{n^4}$ and $\text{rank}_M(\mathcal{F})$ is the matrix-rank of $\mathcal{F}$. Then we can both upper bound and lower bound the symmetric CP rank of $\mathcal{F}$ in terms of $\text{rank}_M(\mathcal{F})$. In particular, it holds that*

$$\text{rank}_M(\mathcal{F}) \leq \text{rank}_{SCP}(\mathcal{F}) \leq (n+4n^2)\text{rank}_M(\mathcal{F}).$$

4.4 Rank-One Equivalence between Matrix Rank and Symmetric CP Rank

In this section, we restrict our attention to real valued tensors. The real symmetric CP rank for real super-symmetric tensor is defined as follows.

Definition 4.4.1. *Suppose $\mathcal{F} \in \mathbf{S}^{n^{2d}}$, the real symmetric CP rank of $\mathcal{F}$ denoted by $\text{rank}_{RCP}(\mathcal{F})$ is the smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r}\lambda_i \underbrace{a^i\otimes\cdots\otimes a^i}_{2d}, \tag{4.18}$$

where $a^i \in \mathbb{R}^n, \lambda_i \in \mathbb{R}^1$.

Notice that the complex rank and the real rank of a tensor can be different [78], although it is not the case for matrix. However, the complex matrix-rank and the real

matrix-rank of a tensor is the same, which is again, a nice property of matrix-rank. In the following, we further investigate the relationship between matrix-rank and real symmetric CP rank for real valued super-symmetric tensors, and show that they are equivalent when one of them equals to one.

We state a result regarding the rank-one super-symmetric tensor.

Lemma 4.4.2. *If a d-th order tensor* $\mathcal{A} = \lambda\, b \otimes \underbrace{a \otimes a \otimes \cdots \otimes a}_{d-1}$ *for some* $\lambda \neq 0 \in \mathbb{R}^1$ *and* $a, b \neq 0 \in \mathbb{R}^n$*, is super-symmetric, then we have* $b = \pm a$.

Proof. Since $\mathcal{A}$ is super-symmetric, from Theorem 4.1 in [79], we know that

$$\max_{\|x\|=1} |\mathcal{A}(\underbrace{x, \cdots, x}_{d})| = \max_{\|x^i\|=1, i=1,\ldots,d} \mathcal{A}(x^1, \cdots, x^d) = |\lambda| \cdot \|b\| \cdot \|a\|^{d-1}.$$

So there must exist a x^* with $\|x^*\| = 1$ such that $|\lambda| \cdot |b^\top x^*| \cdot |a^\top x^*|^{d-1} = |\lambda| \cdot \|b\| \cdot \|a\|^{d-1}$, which implies $b = \pm a$. □

We then give the definition of *proper tensor*.

Definition 4.4.3. *We call* $\mathcal{A} \in \mathbb{R}^{n^d}$ *a proper n dimensional d-th order tensor if for any index k there exists a d-tuple* $\{i_1, \cdots, i_d\} \supseteq \{k\}$ *such that* $\mathcal{A}_{i_1,\cdots,i_d} \neq 0$.

We have the following lemma for a proper super-symmetric tensor.

Lemma 4.4.4. *Suppose* $\mathcal{A} \in \mathbb{R}^{n^d}$ *is a proper n dimensional d-th order tensor such that* $\mathcal{F} = \mathcal{A} \otimes \mathcal{A} \in \mathbf{S}^{n^{2d}}$*, i.e.,* $\mathcal{F}$ *is super-symmetric. If* $\mathcal{A}$ *is also super-symmetric, then the diagonal element* $\mathcal{A}_{k^d} \neq 0$ *for all* $1 \leq k \leq n$.

Proof. For any given index k, suppose there is an m-tuple $(i_1 \cdots i_m)$ such that $\mathcal{A}_{i_1\cdots i_m k^{d-m}} = 0$. For any $j_{m+1} \neq k$, we have,

$$\begin{aligned}
\mathcal{A}^2_{i_1\cdots i_m j_{m+1} k^{d-m-1}} &= \mathcal{F}_{i_1\cdots i_m j_{m+1} k^{d-m-1} i_1\cdots i_m j_{m+1} k^{d-m-1}} \\
&= \mathcal{F}_{i_1\cdots i_m k^{d-m} i_1\cdots i_m j_{m+1} j_{m+1} k^{d-m-2}} \\
&= \mathcal{A}_{i_1\cdots i_m k^{d-m}}\, \mathcal{A}_{i_1\cdots i_m j_{m+1} j_{m+1} k^{d-m-2}} = 0.
\end{aligned}$$

This implies that

$$\mathcal{A}_{i_1\cdots i_m j_{m+1}\cdots j_\ell k^{d-\ell}} = 0, \quad \forall\ 0 \leq m < \ell < d, \text{ and } j_{m+1}, \cdots, j_\ell \neq k.$$

Therefore, if there is an index k with $\mathcal{A}_{k,\cdots,k} = 0$, then $\mathcal{A}_{j_1\cdots j_\ell k^{d-\ell}} = 0$ for all $0 < \ell < d$ and $j_1, \cdots, j_\ell \neq k$. This, combined with the assumption that $\mathcal{A}$ is a super-symmetric tensor, contradicts the fact that $\mathcal{A}$ is proper. □

We further prove the following proposition for super-symmetric tensor.

Proposition 4.4.5. *Suppose $\mathcal{A} \in \mathbb{R}^{n^d}$ is an n dimensional d-th order tensor. The following two statements are equivalent:*

$$\begin{array}{ll} (i) & \mathcal{A} \in \mathbf{S}^{n^d}, \text{ and } \operatorname{rank}_{RCP}(\mathcal{A}) = 1; \\ (ii) & \mathcal{A} \otimes \mathcal{A} \in \mathbf{S}^{n^{2d}}. \end{array}$$

Proof. We shall first show (i) $\Longrightarrow$ (ii). Suppose $\mathcal{A} \in \mathbf{S}^{n^d}$ with $\operatorname{rank}_{RCP}(\mathcal{A}) = 1$. Then there exits a vector $a \in \mathbb{R}^n$ and a scaler $\lambda \neq 0 \in \mathbb{R}^1$ such that $\mathcal{A} = \lambda \underbrace{a \otimes a \otimes \cdots \otimes a}_{d}$. Consequently, $\mathcal{A} \otimes \mathcal{A} = \lambda^2 \underbrace{a \otimes a \otimes \cdots \otimes a}_{2d} \in \mathbf{S}^{n^{2d}}$.

Now we prove (ii) $\Longrightarrow$ (i). We denote $\mathcal{F} = \mathcal{A} \otimes \mathcal{A} \in \mathbf{S}^{n^{2d}}$. For any d-tuples $\{i_1, \cdots, i_d\}$, and one of its permutations $\{j_1, \cdots, j_d\} \in \pi(i_1, \cdots, i_d)$, it holds that

$$\begin{aligned} (\mathcal{A}_{i_1,\cdots,i_d} - \mathcal{A}_{j_1,\cdots,j_d})^2 &= \mathcal{A}^2_{i_1,\cdots,i_d} + \mathcal{A}^2_{j_1,\cdots,j_d} - 2\mathcal{A}_{i_1,\cdots,i_d}\mathcal{A}_{j_1,\cdots,j_d} \\ &= \mathcal{F}_{i_1,\cdots,i_d,i_1,\cdots,i_d} + \mathcal{F}_{j_1,\cdots,j_d,j_1,\cdots,j_d} - 2\mathcal{F}_{i_1,\cdots,i_d,j_1,\cdots,j_d} = 0, \end{aligned}$$

where the last equality is due to the fact that $\mathcal{F}$ is super-symmetric. Therefore, $\mathcal{A}$ is super-symmetric. In the following, we will prove that $\mathcal{A} \in \mathbb{R}^{n^d}$ is a rank-one tensor by induction on d. It is evident that $\mathcal{A}$ is rank-one when $d = 1$. Now we assume that $\mathcal{A}$ is rank-one when $\mathcal{A} \in \mathbb{R}^{n^{d-1}}$ and we will show that the conclusion holds when the order of $\mathcal{A}$ is d.

For $\mathcal{A} \in \mathbb{R}^{n^d}$, we already proved that $\mathcal{A}$ is super-symmetric. Now assume $\mathcal{A}$ is proper, by Lemma 4.4.4 we know that $\mathcal{A}_{k^d} \neq 0$ for all $1 \leq k \leq n$. We further observe that $\mathcal{F} \in \mathbf{S}^{n^{2d}}$ implies

$$\mathcal{A}_{i_1\cdots i_{d-1}j}\mathcal{A}_{k^d} = \mathcal{F}_{i_1\cdots i_{d-1}jk^d} = \mathcal{F}_{i_1\cdots i_{d-1}k^d j} = \mathcal{A}_{i_1\cdots i_{d-1}k}\mathcal{A}_{k^{d-1}j},$$

for any $(i_1, \cdots, i_{d-1})$. As a result,

$$\mathcal{A}_{i_1\cdots i_{d-1}j} = \frac{\mathcal{A}_{k^{d-1}j}}{\mathcal{A}_{k^d}} \mathcal{A}_{i_1\cdots i_{d-1}k}, \quad \forall\ j, k, (i_1, \cdots, i_{d-1}).$$

Now we can construct a vector $b \in \mathbb{R}^n$ with $b_j = \frac{\mathcal{A}_{k^{d-1}j}}{\mathcal{A}_{k^d}}$ and a tensor $\mathcal{A}(k) \in \mathbb{R}^{n^{d-1}}$ with $\mathcal{A}(k)_{i_1 \cdots i_{d-1}} = \mathcal{A}_{i_1 \cdots i_{d-1} k}$, such that

$$\mathcal{A} = b \otimes \mathcal{A}(k), \tag{4.19}$$

and

$$\mathcal{F} = b \otimes \mathcal{A}(k) \otimes b \otimes \mathcal{A}(k) = b \otimes b \otimes \mathcal{A}(k) \otimes \mathcal{A}(k),$$

where the last equality is due to $\mathcal{F} \in \mathbf{S}^{n^{2d}}$. On the other hand, we notice that $\mathcal{A}_{j^{d-1}k} \neq 0$ for all $1 \leq j \leq n$. This is because if this is not true then we would have

$$0 = \mathcal{A}_{j^{d-1}k} \mathcal{A}_{k^{d-1}j} = \mathcal{A}_{j^d} \mathcal{A}_{k^d},$$

which contradicts the fact that $\mathcal{A}$ is proper. This means that all the diagonal elements of $\mathcal{A}(k)$ are nonzero, implying that $\mathcal{A}(k)$ is a proper tensor. Moreover, $\mathcal{A}(k) \otimes \mathcal{A}(k) \in \mathbf{S}^{n^{2d-2}}$, because $\mathcal{F}$ is super-symmetric. Thus by induction, we can find a vector $a \in \mathbb{R}^n$ and a scaler $\lambda \neq 0 \in \mathbb{R}^1$ such that

$$\mathcal{A}(k) = \lambda \underbrace{a \otimes a \otimes \cdots \otimes a}_{d-1}.$$

Plugging the above into (4.19), we get $\mathcal{A} = \lambda\, b \otimes \underbrace{a \otimes a \otimes \cdots \otimes a}_{d-1}$. Since $\mathcal{A}$ is super-symmetric, by Lemma 4.4.2 $b = \pm a$ and thus $\mathcal{A}$ is of rank one.

Recall that we have assumed $\mathcal{A}$ is proper in the above argument. Now suppose this is not true. Without loss of generality, we assume $k+1, ..., n$ are all such indices that $\mathcal{A}_{j_1 \cdots j_d} = 0$ if $\{j_1, \cdots, j_d\} \supseteq \{\ell\}$ with $k+1 \leq \ell \leq n$. Now introduce tensor $\mathcal{B} \in \mathbb{R}^{k^d}$ such that $\mathcal{B}_{i_1, \cdots, i_d} = \mathcal{A}_{i_1, \cdots, i_d}$ for any $1 \leq i_1, \cdots, i_d \leq k$. Obviously $\mathcal{B}$ is proper. Moreover, since $\mathcal{A} \otimes \mathcal{A} \in \mathbf{S}^{n^{2d}}$, it follows that $\mathcal{B} \otimes \mathcal{B} \in \mathbf{S}^{n^{2d}}$. Thanks to the argument above, there exists a vector $b \in \mathbb{R}^k$ such that $\mathcal{B} = \underbrace{b \otimes \cdots \otimes b}_{d}$. Finally, by letting $a^\top = (b^\top, \underbrace{0, \cdots, 0}_{n-k})$, we have $\mathcal{A} = \underbrace{a \otimes \cdots \otimes a}_{d}$. □

For the purpose of following discussion, we introduce below the vectorization of a tensor.

Definition 4.4.6. *The vectorization, $\boldsymbol{V}(\mathcal{F})$, of tensor $\mathcal{F} \in \mathbb{R}^{n^m}$ is defined as*

$$\boldsymbol{V}(\mathcal{F})_k := \mathcal{F}_{i_1 \cdots i_m},$$

where

$$k = \sum_{j=1}^{m} (i_j - 1) n^{m-j} + 1, 1 \leq i_1, \cdots, i_m \leq n.$$

Since the operator is $\boldsymbol{V}(\cdot)$ is a one-to-one correspondence, he inverse of $\boldsymbol{V}(\cdot)$ can be defined in the same manner.

Now we are ready to present the main result of this section.

Theorem 4.4.7. *Suppose $\mathcal{X} \in \mathbf{S}^{n^{2d}}$ and $X = \boldsymbol{M}(\mathcal{X}) \in \mathbb{R}^{n^d \times n^d}$. Then we have*

$$\operatorname{rank}(\mathcal{X})_{RCP} = 1 \iff \operatorname{rank}_M(\mathcal{X}) = \operatorname{rank}(X) = 1.$$

Proof. As remarked earlier, that $\operatorname{rank}(\mathcal{X})_{RCP} = 1 \implies \operatorname{rank}(X) = 1$ is evident. To see this, suppose $\operatorname{rank}(\mathcal{X})_{RCP} = 1$ and $\mathcal{X} = \underbrace{x \otimes \cdots \otimes x}_{2d}$ for some $x \in \mathbb{R}^n$. By constructing $\mathcal{Y} = \underbrace{x \otimes \cdots \otimes x}_{d}$, we have $X = \boldsymbol{M}(\mathcal{X}) = \boldsymbol{V}(\mathcal{Y})\boldsymbol{V}(\mathcal{Y})^\top$, which leads to $\operatorname{rank}(X) = 1$.

To prove the other implication, suppose that we have $\mathcal{X} \in \mathbf{S}^{n^{2d}}$ and $\boldsymbol{M}(\mathcal{X})$ is of rank one, i.e. $\boldsymbol{M}(\mathcal{X}) = yy^\top$ for some vector $y \in \mathbb{R}^{n^d}$. Then $\mathcal{X} = \boldsymbol{V}^{-1}(y) \otimes \boldsymbol{V}^{-1}(y)$, which combined with Proposition 4.4.5 implies $\boldsymbol{V}^{-1}(y)$ is supper-symmetric and of rank one. Thus there exits $x \in \mathbb{R}^n$ such that $\boldsymbol{V}^{-1}(y) = \underbrace{x \otimes \cdots \otimes x}_{d}$ and $\mathcal{X} = \underbrace{x \otimes \cdots \otimes x}_{2d}$. □

As it turns out, the rank-one equivalence theorem can be extended to the non-super-symmetric tensors. Let's focus on the following partial-symmetric tensor.

Definition 4.4.8. *A 4-th order tensor $\mathcal{G} \in \mathbb{R}^{(nm)^2}$ is called partial-symmetric if $\mathcal{G}_{ijk\ell} = \mathcal{G}_{kji\ell} = \mathcal{G}_{i\ell kj}, \forall i, j, k, \ell$. The space of all 4-th order partial-symmetric tensor is denoted by $\overrightarrow{\overrightarrow{\mathbf{S}}}^{(mn)^2}$.*

Definition 4.4.9. *Given $\mathcal{G} \in \overrightarrow{\overrightarrow{\mathbf{S}}}^{(mn)^2}$, the partial symmetric CP rank of $\mathcal{G}$ denoted by $\operatorname{rank}_{PCP}(\mathcal{G})$ is the smallest integer r such that*

$$\mathcal{F} = \sum_{i=1}^{r} \lambda_i \cdot a^i \otimes b^i \otimes a^i \otimes b^i,$$

with $a^i, b^i \in \mathbb{R}^n$, and $\lambda_i \in \mathbb{R}^1$.

Theorem 4.4.10. *Suppose A is an $n \times m$ dimensional matrix. Then the following two statements are equivalent:*
(i) $\operatorname{rank}_{PCP}(A) = 1$*;*
(ii) $A \otimes A \in \overrightarrow{\mathbf{S}}^{(nm)^2}$.
In other words, suppose $\mathcal{F} \in \overrightarrow{\mathbf{S}}^{(nm)^2}$*, then* $\operatorname{rank}(\mathcal{F})_{PCP} = 1 \iff \operatorname{rank}_M(\mathcal{F}) = \operatorname{rank}(F) = 1$*, where* $F = \boldsymbol{M}(\mathcal{F})$.

Proof. (i) $\Longrightarrow$ (ii) is obvious. Suppose $\operatorname{rank}(A) = 1$, say $A = ab^\top$ for some $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. Then $\mathcal{G} = A \otimes A = a \otimes b \otimes a \otimes b$ is partial-symmetric.

Conversely, suppose $\mathcal{G} = A \otimes A \in \overrightarrow{\mathbf{S}}^{(nm)^2}$. Then

$$A_{i_1 j_1} A_{i_2 j_2} = \mathcal{G}_{i_1 j_1 i_2 j_2} = \mathcal{G}_{i_2 j_1 i_1 j_2} = A_{i_2 j_1} A_{i_1 j_2},\ \forall 1 \le i_1, i_2 \le n,\ 1 \le j_1, j_2 \le m,$$

implies $A_{i_1 j_1} A_{i_2 j_2} - A_{i_2 j_1} A_{i_1 j_2} = 0$. That is, every 2×2 minor of matrix A is zero. Thus A is of rank one. $\square$

Chapter 5

Probability Bounds for Polynomial Functions in Random Variables

5.1 Introduction

Let $f(x) : \mathbb{R}^n \to \mathbb{R}$ be a function, and $S \subseteq \mathbb{R}^n$ be a given set, wherewith we consider: $\max_{x \in S} f(x)$. A possible generic approximation method for solving this problem would be randomization and sampling. In particular, we may proceed as follows: (i) choose a suitable and well-structured subset $S_0 \subseteq S$; (ii) design a suitable probability distribution ξ on S_0; (iii) take some random samples and pick the best solution. The quality of this approach, of course, depends on the chance of hitting some 'good solutions' by the random sampling. In other words, a bound in the following format is of crucial importance to us:

$$\operatorname*{Prob}_{\xi \sim S_0} \left\{ f(\xi) \geq \tau \max_{x \in S} f(x) \right\} \geq \theta, \tag{5.1}$$

where $\tau > 0$ and $0 < \theta < 1$ are certain constants.

In another situation, the original problem of interest is $\max_{x \in S_0} f(x)$. Replacing the constraint set to be $x \in S$ is a relaxation and it can help to create an easier problem to analyze. In this setting, a bound like (5.1) is useful in terms of deriving an approximate solution for solving the problem. A good example of this approach is the max-cut

formulation of Goemans and Williamson [80], where S_0 is the set of rank-one positive semidefinite matrices with diagonal elements being all-ones, and S is S_0 dropping the rank-one restriction. In [81, 82, 21], this technique helped in the design of efficient randomized approximation algorithms for solving quadratically constrained quadratic programs by semidefinite programming (SDP) relaxation.

Motivated mainly due to its generic interest and importance, primarily in optimization, the current chapter is devoted to the establishment of inequalities of type (5.1), under various assumptions. Of course such probability estimation cannot hold in general, unless some structures are in place. However, once (5.1) indeed holds, then with probability θ we will get a solution whose value is no worse than τ times the best possible value of $f(x)$ over S. In other words, with probability θ we will be able to generate a τ-approximate solution. In particular, if we independently draw m trials of ξ on S_0 and pick the one with the largest function value, then this process is a randomized approximation algorithm with approximation ratio τ, where the probability to this quality solution is at least $1-(1-\theta)^m$. If $m = \frac{\ln \frac{1}{\epsilon}}{\theta}$ then $1-(1-\theta)^m \geq 1-\epsilon$, and this randomized algorithm indeed runs in polynomial-time in the problem dimensions.

In fact, the framework of our investigation, viz. the probability bound (5.1), is sufficiently rich to include some highly nontrivial results beyond optimization as well. As an example, let $f(x) = a^\top x$ be a linear function, and $S = S_0 = \mathbb{B}^n := \{1, -1\}^n$ be a binary hypercube. Khot and Noar in [83] derived the following probability bound, which can be seen as a nontrivial instance of (5.1).

> For every $\delta \in (0, \frac{1}{2})$, there is a constant $c_1(\delta) > 0$ with the following property: Fix $a = (a_1, a_2, \ldots, a_n)^\top \in \mathbb{R}^n$ and let $\xi_1, \xi_2, \ldots, \xi_n$ be i.i.d. symmetric Bernoulli random variables (taking ± 1 with equal probability), then
>
> $$\text{Prob}\left\{\sum_{j=1}^{n} a_j \xi_j \geq \sqrt{\frac{\delta \ln n}{n}}\, \|a\|_1\right\} \geq \frac{c_1(\delta)}{n^\delta}. \tag{5.2}$$

Since $\max_{x \in \mathbb{B}^n} a^\top x = \|a\|_1$, (5.2) is of type (5.1), with $\tau = \sqrt{\frac{\delta \ln n}{n}}$ and $\theta = \frac{c_1(\delta)}{n^\delta}$. This bound indeed gives rise to an $\Theta\left(\sqrt{\frac{\ln n}{n}}\right)$-approximation algorithm for the binary

constrained trilinear form maximization problem:

$$\begin{array}{ll} \max & \mathcal{A}(x,y,z) := \sum_{i,j,k=1}^{n} a_{ijk} x_i y_j z_k \\ \text{s.t.} & x,y,z \in \mathbb{B}^n. \end{array}$$

To see why, let us denote its optimal solution to be $(x^*, y^*, z^*) = \arg\max_{x,y,z\in\mathbb{B}^n} \mathcal{A}(x,y,z)$. By letting $a = \mathcal{A}(\cdot, y^*, z^*) \in \mathbb{R}^n$ and $\xi_1, \xi_2, \ldots, \xi_n$ be i.i.d. symmetric Bernoulli random variables, it follows from (5.2) that

$$\text{Prob}\left\{ \mathcal{A}(\xi, y^*, z^*) \geq \sqrt{\frac{\delta \ln n}{n}} \, \|\mathcal{A}(\cdot, y^*, z^*)\|_1 \right\} \geq \frac{c_1(\delta)}{n^\delta}. \tag{5.3}$$

Notice that by the optimality of (x^*, y^*, z^*), we have $\|\mathcal{A}(\cdot, y^*, z^*)\|_1 = \mathcal{A}(x^*, y^*, z^*)$. Besides for any fixed ξ, the problem $\max_{y,z\in\mathbb{B}^n} \mathcal{A}(\xi, y, z)$ is a binary constrained bilinear form maximization problem, which admits a deterministic approximation algorithm with approximation ratio 0.03 (see Alon and Naor [69]). Thus we are able to find two vectors $y_\xi, z_\xi \in \mathbb{B}^n$ in polynomial-time such that

$$\mathcal{A}(\xi, y_\xi, z_\xi) \geq 0.03 \max_{y,z\in\mathbb{B}^n} \mathcal{A}(\xi, y, z) \geq 0.03\, \mathcal{A}(\xi, y^*, z^*),$$

which by (8.6.2) implies

$$\text{Prob}\left\{ \mathcal{A}(\xi, y_\xi, z_\xi) \geq 0.03\sqrt{\frac{\delta \ln n}{n}} \, \mathcal{A}(x^*, y^*, z^*) \right\} \geq \frac{c_1(\delta)}{n^\delta}.$$

Now we may independently draw $\xi_1, \xi_2, \ldots, \xi_n$, followed by the algorithm proposed in [69] to solve $\max_{y,z\in\mathbb{B}^n} \mathcal{A}(\xi, y, z)$. If we apply this procedure $\frac{n^\delta \ln \frac{1}{\epsilon}}{c_1(\delta)}$ times and pick the one with the largest objective value, then it is actually a polynomial-time randomized approximation algorithm with approximation ratio $0.03\sqrt{\frac{\delta \ln n}{n}}$, whose chance of getting this quality bound is at least $1 - \epsilon$.

The scope of applications for results of type (5.1) is certainly beyond optimization per se; it is significant in the nature of probability theory itself. Recall that most classical results in probability theory is to upper bound the tail of a distribution (e.g. the Markov inequality and the Chebyshev inequality), say $\text{Prob}\{\xi \geq a\} \leq b$. In other words, these are the *upper* bounds for the probability of a random variable beyond a threshold value. However, in some applications a *lower* bound for such probability can be relevant, in the form of

$$\text{Prob}\{\xi \geq a\} \geq b. \tag{5.4}$$

One interesting example is a result due to Ben-Tal, Nemirovskii, and Roos [19], where they proved a lower bound of $1/8n^2$ for the probability that a homogeneous quadratic form of n i.i.d. symmetric Bernoulli random variables lies above its mean. More precisely, they proved the following:

> If $\mathcal{A} \in \mathbb{R}^{n\times n}$ is a symmetric matrix and $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top$ are i.i.d. symmetric Bernoulli random variables, then $\text{Prob}\left\{\xi^\top \mathcal{A}\xi \geq \text{tr}\,(\mathcal{A})\right\} \geq \frac{1}{8n^2}$.

As a matter of fact, the authors went on to conjecture in [19] that the lower bound can be as high as $\frac{1}{4}$, which was very recently disproved by Yuan [20]. However, the value of the tight bound remains unknown. A significant progress on this conjecture is due to He et al. [21], where the authors improved the lower bound of $\frac{1}{8n^2}$ to 0.03. Note that the result of He et al. [21] also holds for any ξ_i's being i.i.d. standard normal random variables. Luo and Zhang [22] provides a constant lower bound for the probability that a homogeneous quartic function of a zero mean multivariate normal distribution lies above its mean, which was a first attempt to extend such probability bound for functions of random variables beyond quadratic. For a univariate random variable, bounds of type (5.4) and its various extensions can be found in a recent paper by He, Zhang, and Zhang [84].

A well known result of Grünbaum [85] can also be put in the category of probability inequality (5.4). Grünbaum's theorem asserts:

> If $S \subseteq \mathbb{R}^n$ is convex and ξ is uniformly distributed on S, then for any $c \in \mathbb{R}^n$,
>
> $$\text{Prob}\left\{c^\top \xi \geq c^\top \mathsf{E}\xi\right\} \geq e^{-1}.$$

This chapter aims at providing various new lower bounds for inequalities of type (5.1), when f is a multivariate polynomial function. To enable the presentation of our results, let us first briefly recall some notations adopted in this chapter. For any given set $S \subseteq \mathbb{R}^n$, $\xi \sim S$ stands for that ξ is a multivariate *uniform* distribution on the support S. Two types of support sets are frequently used in this chapter, namely

$$\mathbb{B}^n := \{1, -1\}^n \quad \text{and} \quad \mathbb{SH}^n := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}.$$

It is easy to verify the following equivalent relationship:

1. $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top \sim \mathbb{B}^n$ is equivalent to $\xi_i \sim \mathbb{B}\,(i = 1, 2, \ldots, n)$, and ξ_i's are i.i.d. random variables;

2. $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top \sim \mathbb{SH}^n$ is equivalent to $\eta/\|\eta\|_2$, with $\eta = (\eta_1, \eta_2, \ldots, \eta_n)^\top$ and η_i's are i.i.d. standard normal random variables.

To simplify the presentation, the notion $\Theta(f(n))$ signifies the fact that there are positive universal constants α, β and n_0 such that $\alpha f(n) \leq \Theta(f(n)) \leq \beta f(n)$ for all $n \geq n_0$; i.e., it is of the same order as $f(n)$. To avoid confusion, the term *constant* sometimes also refers to a parameter depending only on the dimension of a polynomial function, which is a given number independent of the input data of the problem.

The chapter is organized as follows. In Section 5.2, we present probability inequalities of type (5.1) where f is a multilinear form, and ξ is either a random vector with i.i.d. symmetric Bernoulli random variables, or a uniform distribution over hypersphere. Besides, in Section 5.3 we present another set of probability bounds of homogeneous polynomial function over a general class of independent random variables, including symmetric Bernoulli random variables and uniform distribution over hypersphere.

5.2 Multilinear Tensor Function in Random Variables

In this section we present the following result, which provides tight probability bounds for multilinear form in two different sets of random variables.

Theorem 5.2.1. *Let $\xi^i \sim \mathbb{B}^{n_i}$ $(i = 1, 2, \ldots, d)$ be independent of each other, and $\eta^i \sim \mathbb{S}^{n_i}$ $(i = 1, 2, \ldots, d)$ be independent of each other. For any d-th order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ with $n_1 \leq n_2 \leq \cdots \leq n_d$, and constant $\delta \in (0, \frac{1}{2}), \gamma \in (0, \frac{n_d}{\ln n_d})$, it follows that*

$$\text{Prob}\left\{\mathcal{A}(\xi^1, \xi^2, \ldots, \xi^d) \geq c_3^{d-1}\sqrt{\frac{\delta \ln n_d}{\prod_{i=1}^d n_i}}\,\|\mathcal{A}\|_1\right\} \geq \frac{c_1(\delta)c_3^{2d-2}}{n_d^\delta \prod_{i=2}^d n_i^{i-1}}, \tag{5.5}$$

$$\text{Prob}\left\{\mathcal{A}(\eta^1, \eta^2, \ldots, \eta^d) \geq \frac{1}{2^{\frac{d-1}{2}}}\sqrt{\frac{\gamma \ln n_d}{\prod_{i=1}^d n_i}}\,\|\mathcal{A}\|_2\right\} \geq \frac{c_2(\gamma)}{4^{d-1} n_d^{2\gamma}\sqrt{\ln n_d}\prod_{i=1}^{d-1} n_i}, \tag{5.6}$$

*where $c_1(\delta)$ is a constant depended only on δ, $c_2(\gamma)$ is a constant depended only on γ, and $c_3 := \frac{8}{25\sqrt{5}} \approx 0.1431$. Moreover, the order of magnitude '$\sqrt{\frac{\ln n_d}{\prod_{i=1}^d n_i}}$' inside '*Prob*'*

in (5.5) *and* (5.6) *cannot be improved, if the probability bound on the right-hand-side is at least the reciprocal of a polynomial function in* n_d.

We remark here that the degree d is deemed a fixed constant in our discussion. If we let $S = \mathbb{B}^{n_1 \times n_2 \times \cdots \times n_d}$ and $S_0 = \{X \in \mathbb{B}^{n_1 \times n_2 \times \cdots \times n_d} \mid \text{rank}\,(X) = 1\}$, then (5.5) is in the form of (5.1). Similarly, if we let $S = \mathbb{S}^{n_1 \times n_2 \times \cdots \times n_d}$ and $S_0 = \{X \in \mathbb{S}^{n_1 \times n_2 \times \cdots \times n_d} \mid \text{rank}\,(X) = 1\}$, then (5.6) is in the form of (5.1). For clarity, we shall prove (5.5) and (5.6) separately in the following two subsections. Before doing this, let us first comment on the tightness of the bound $\tau_d := \Theta\left(\sqrt{\frac{\ln n_d}{\prod_{i=1}^d n_i}}\right) = \Theta\left(\sqrt{\frac{\ln \prod_{i=1}^d n_i}{\prod_{i=1}^d n_i}}\right)$, where the last equality holds because d is a fixed constant and $n_i \leq n_d$ for $i = 1, 2, \ldots, d-1$. The tightness of the bounds is due to the inapproximability of computing the diameters of convex bodies, as shown below.

Lemma 5.2.2. (Knot and Naor [83]) *Let* $K \in \mathbb{R}^n$ *be a convex body with a weak optimization oracle. Then there is no randomized oracle-polynomial time algorithm that can compute the* L_1 *diameter of* K *with accuracy* $\Theta\left(\sqrt{\frac{\ln n}{n}}\right)$.

Lemma 5.2.3. (Brieden et al. [86, 87]) *Let* $K \in \mathbb{R}^n$ *be a convex body with a weak optimization oracle. Then there is no randomized oracle-polynomial time algorithm that can compute the* L_2 *diameter of* K *with accuracy* $\Theta\left(\sqrt{\frac{\ln n}{n}}\right)$.

These results in fact lead to the tightness of $\tau_1 = \Theta\left(\sqrt{\frac{\ln n_1}{n_1}}\right)$ in the case $d = 1$ (when the tensor $\mathcal{A}$ in (5.5) and (5.6) is a vector), for, if τ_1 could be improved, then applying the same argument as in the proof of Theorem 3.1 in [83]: drawing enough (polynomial number of) samples of $\xi \in \mathbb{B}^n$ for the L_1 case (respective $\eta \in \mathbb{SH}^n$ for the L_2 case) followed by the oracle-polynomial time algorithm, would then improve the approximation bound τ_1 for the L_1 (respective L_2) diameter.

In fact, τ_1 is a tight bound not only for $\xi \sim \mathbb{B}^n$ but also for other structural distributions on the support set $\mathbb{B}^n$, also due to the inapproximability of computing the L_1 diameters of convex bodies (Lemma 5.2.2). Now, for any given degree d, if we denote $n = \prod_{i=1}^d n_i$, then (5.5) is essentially

$$\text{Prob}\left\{\mathcal{A} \bullet (\xi^1 \otimes \xi^2 \otimes \cdots \otimes \xi^d) \geq \Theta\left(\sqrt{\frac{\ln n}{n}}\right) \|\mathcal{A}\|_1\right\} \geq \Theta\left(\frac{1}{n_d^\alpha}\right) \tag{5.7}$$

for some constant α. Denote $\xi = \xi^1 \otimes \xi^2 \otimes \cdots \otimes \xi^d$, and clearly it is an implementable distribution on the support $\mathbb{B}^n$. Thus (5.7) can be regarded as in the form of (5.5) for $d = 1$. Due to the tightness of τ_1, the bound $\tau_d = \Theta\left(\sqrt{\frac{\ln n_d}{\prod_{i=1}^d n_i}}\right) = \Theta\left(\sqrt{\frac{\ln n}{n}}\right)$ for general d in (5.5), once established, is tight too. The same argument of the structural distribution on the support set $\mathbb{SH}^n$ with $n = \prod_{i=1}^d n_i$ can be applied to prove the tightness of τ_1 in (5.6), using Lemma 5.2.3. It is interesting to note that a completely free ξ and the more restrictive $\xi = \xi^1 \otimes \xi^2 \otimes \cdots \otimes \xi^d$ lies in the fact that the latter is rank-one. Hence, the establishment of (5.5) and (5.6) actually implies that as far as the randomized solution is concerned, the rank-one restriction is immaterial.

5.2.1 Multilinear Tensor Function in Bernoulli Random Variables

This subsection is dedicated to the proof of the first part of Theorem 5.2.1, namely (5.5). Let us start with some technical preparations. First, we have the following immediate probability estimation.

Lemma 5.2.4. *If $\xi \sim \mathbb{B}^n$, then for any vector $a \in \mathbb{R}^n$,*

$$\mathsf{E}|a^\top \xi| \geq 2c_3\|a\|_2.$$

Proof. Denote $z = |a^\top \xi|$, and observe

$$\mathsf{E}z^2 = \mathsf{E}\left[\sum_{i=1}^n \xi_i a_i\right]^2 = \mathsf{E}\left[\sum_{i=1}^n a_i^2 + 2\sum_{1\leq i<j\leq n} \xi_i\xi_j a_i a_j\right] = \sum_{i=1}^n a_i^2 = \|a\|_2^2.$$

Direct computation shows that $\mathsf{E}z^4 \leq 9\left(\mathsf{E}z^2\right)^2$. By the Paley-Zygmund inequality [88], for every $\alpha \in (0,1)$,

$$\text{Prob}\left\{z \geq \sqrt{\alpha \mathsf{E}z^2}\right\} = \text{Prob}\left\{z^2 \geq \alpha \mathsf{E}z^2\right\} \geq (1-\alpha)^2\left(\mathsf{E}z^2\right)^2/\mathsf{E}z^4 \geq (1-\alpha)^2/9.$$

Since $z \geq 0$, we have

$$\mathsf{E}z \geq \text{Prob}\left\{z \geq \sqrt{\alpha \mathsf{E}z^2}\right\}\sqrt{\alpha \mathsf{E}z^2} \geq \frac{(1-\alpha)^2}{9}\sqrt{\alpha \mathsf{E}z^2} = \frac{(1-\alpha)^2\sqrt{\alpha}}{9}\|a\|_2.$$

By maximizing $\frac{(1-\alpha)^2\sqrt{\alpha}}{9}$ over $\alpha \in (0,1)$, we have $\mathsf{E}z \geq \frac{16}{25\sqrt{5}}\|a\|_2 = 2c_3\|a\|_2$. □

We shall establish (5.5) by induction on the degree d. The first inductive step from $d = 1$ to $d = 2$ relies on the next lemma.

Lemma 5.2.5. *If $\xi \sim \mathbb{B}^n$, then for any matrix $A \in \mathbb{R}^{m\times n}$,*

$$\text{Prob}\left\{\|A\xi\|_1 \geq \frac{c_3}{\sqrt{n}}\|A\|_1\right\} \geq \frac{c_3^2}{m}.$$

Proof. Denote $a^i \in \mathbb{R}^n\,(i = 1, 2, \ldots, m)$ to be the i-th row vector of the matrix A. By Lemma 5.2.4 we have for each $i = 1, 2, \ldots, m$,

$$\mathsf{E}\left|\xi^\top a^i\right| \geq 2c_3\|a^i\|_2 \geq \frac{2c_3}{\sqrt{n}}\|a^i\|_1.$$

Summing over all $i = 1, 2, \ldots, m$, we have

$$\mathsf{E}\|A\xi\|_1 = \sum_{i=1}^m \mathsf{E}\left|\xi^\top a^i\right| \geq \frac{2c_3}{\sqrt{n}}\|A\|_1.$$

On the other hand,

$$(\mathsf{E}\|A\xi\|_1)^2 = \left(\sum_{i=1}^m \mathsf{E}\left|\xi^\top a^i\right|\right)^2 \geq \sum_{i=1}^m \left(\mathsf{E}\left|\xi^\top a^i\right|\right)^2 \geq \sum_{i=1}^m 4c_3^2\|a^i\|_2^2 = 4c_3^2\|A\|_2^2,$$

and

$$\mathsf{E}\|A\xi\|_1^2 = \mathsf{E}\left[\sum_{i=1}^m \left|\xi^\top a^i\right|\right]^2 \leq \mathsf{E}\left[m\sum_{i=1}^m \left|\xi^\top a^i\right|^2\right] = m\sum_{i=1}^m \mathsf{E}\left[\xi^\top a^i\right]^2 = m\sum_{i=1}^m \|a^i\|_2^2 = m\|A\|_2^2.$$

Thus by the Paley-Zygmund inequality we conclude that for any $\alpha \in (0, 1)$,

$$\begin{aligned}
\text{Prob}\left\{\|A\xi\|_1 \geq \frac{2\alpha c_3}{\sqrt{n}}\|A\|_1\right\} &\geq \text{Prob}\left\{\|A\xi\|_1 \geq \alpha\mathsf{E}\|A\xi\|_1\right\} \\
&\geq (1-\alpha)^2\frac{(\mathsf{E}\|A\xi\|_1)^2}{\mathsf{E}\|A\xi\|_1^2} \\
&\geq (1-\alpha)^2\frac{4c_3^2\|A\|_2^2}{m\|A\|_2^2}.
\end{aligned}$$

Finally, letting $\alpha = \frac{1}{2}$ proves the lemma. □

We remark that in the above inequality, the coefficient $\frac{c_3}{\sqrt{n}}$ in front of $\|A\|_1$ is independent of the number of rows (m) for matrix A. Towards proving (5.5) by induction for general d, for ease of referencing we state the following simple fact regarding joint conditional probability.

Proposition 5.2.6. *Suppose ξ and η are two random variables with support sets $U \subseteq \mathbb{R}^n$ and $V \subseteq \mathbb{R}^m$ respectively. For $V' \subseteq V$, $W' \subseteq U \times V$ and $\delta > 0$, if*

$$\operatorname*{Prob}_{\xi} \left\{ (\xi, y) \in W' \right\} \geq \delta \quad \forall\, y \in V$$

and

$$\operatorname*{Prob}_{\eta} \left\{ \eta \in V' \right\} > 0,$$

then the joint conditional probability

$$\operatorname*{Prob}_{(\xi,\eta)} \left\{ (\xi, \eta) \in W' \,\middle|\, \eta \in V' \right\} \geq \delta.$$

Proof. Notice that the first assumption is equivalent to

$$\operatorname*{Prob}_{(\xi,\eta)} \left\{ (\xi, \eta) \in W' \,\middle|\, \eta = y \right\} \geq \delta \quad \forall\, y \in V. \tag{5.8}$$

Suppose that η has a density $g(\cdot)$ in V, then

$$\begin{aligned}
\operatorname*{Prob}_{(\xi,\eta)} \left\{ (\xi, \eta) \in W' \,\middle|\, \eta \in V' \right\} &= \operatorname*{Prob}_{(\xi,\eta)} \left\{ (\xi, \eta) \in W', \eta \in V' \right\} \Big/ \operatorname*{Prob}_{\eta} \left\{ \eta \in V' \right\} \\
&= \int_{V'} \operatorname*{Prob}_{(\xi,\eta)} \left\{ (\xi, \eta) \in W', \eta = y \right\} g(y) dy \Big/ \operatorname*{Prob}_{\eta} \left\{ \eta \in V' \right\} \\
&\geq \int_{V'} \delta \cdot g(y) dy \Big/ \operatorname*{Prob}_{\eta} \left\{ \eta \in V' \right\} = \delta.
\end{aligned}$$

The case where η is a discrete random variable can be handled similarly. □

We are now ready to prove (5.5).

Proof of (5.5) **in Theorem 5.2.1**

Proof. The proof is based on induction on d. The case for $d = 1$ has been established by Knot and Naor [83]. Suppose the inequality holds for $d-1$, by treating ξ^1 as a given parameter and taking $\mathcal{A}(\xi^1, \cdot, \cdot, \cdots, \cdot)$ as a tensor of order $d-1$, one has

$$\operatorname*{Prob}_{(\xi^2,\xi^3,\cdots,\xi^d)} \left\{ \mathcal{A}(\xi^1, \xi^2, \cdots, \xi^d) \geq c_3^{d-2} \sqrt{\frac{\delta \ln n_d}{\prod_{i=2}^d n_i}} \, \|\mathcal{A}(\xi^1, \cdot, \cdot, \cdots, \cdot)\|_1 \right\} \geq \frac{c_1(\delta) c_3^{2d-4}}{n_d^\delta \prod_{i=3}^d n_i^{i-2}}.$$

Define the event $E_1 = \left\{ \|\mathcal{A}(\xi^1, \cdot, \cdot, \cdots, \cdot)\|_1 \geq \frac{c_3}{\sqrt{n_1}} \|\mathcal{A}\|_1 \right\}$. By applying Proposition 5.2.6 with $\xi = (\xi^2, \xi^3, \cdots, \xi^d)$ and $\eta = \xi_1$, we have

$$\operatorname*{Prob}_{(\xi^1,\cdots,\xi^d)} \left\{ \mathcal{A}(\xi^1, \cdots, \xi^d) \geq c_3^{d-2} \sqrt{\frac{\delta \ln n_d}{\prod_{i=2}^d n_i}} \, \|\mathcal{A}(\xi^1, \cdot, \cdot, \cdots, \cdot)\|_1 \,\middle|\, E_1 \right\} \geq \frac{c_1(\delta) c_3^{2(d-2)}}{n_d^\delta \prod_{i=3}^d n_i^{i-2}} \tag{5.9}$$

The desired probability can be lower bounded as follows:

$$\begin{aligned}
&\text{Prob}\left\{\mathcal{A}(\xi^1,\xi^2,\cdots,\xi^d)\geq c_3^{d-1}\sqrt{\frac{\delta\ln n_d}{\prod_{i=1}^d n_i}}\,\|\mathcal{A}\|_1\right\}\\
\geq\;& \underset{(\xi^1,\xi^2,\cdots,\xi^d)}{\text{Prob}}\left\{\mathcal{A}(\xi^1,\xi^2,\cdots,\xi^d)\geq c_3^{d-2}\sqrt{\frac{\delta\ln n_d}{\prod_{i=2}^d n_i}}\,\|\mathcal{A}(\xi^1,\cdot,\cdots,\cdot)\|_1\,\middle|\,E_1\right\}\cdot\text{Prob}\,\{E_1\}\\
\geq\;& \frac{c_1(\delta)c_3^{2d-4}}{n_d^\delta\prod_{i=3}^d n_i^{i-2}}\cdot\frac{c_3^2}{\prod_{i=2}^d n_i}=\frac{c_1(\delta)c_3^{2d-2}}{4^{d-1}n_d^\delta\prod_{i=2}^d n_i^{i-1}},
\end{aligned}$$

where the last inequality is due to (5.9) and Lemma 5.2.5. □

5.2.2 Multilinear Tensor Function over Hyperspheres

In this subsection we shall prove the second part of Theorem 5.2.1, namely (5.6). The main construction is analogous to that of the proof for (5.5). First we shall establish a counterpart of inequality (5.2), i.e., we prove (5.6) for $d = 1$, which is essentially the following Lemma 5.2.7. Namely, if we uniformly and independently draw two vectors in $\mathbb{SH}^n$, then there is non-trial probability that their inner product is at least $\left(\sqrt{\frac{\gamma\ln n}{n}}\right)$ for certain positive γ.

Lemma 5.2.7. *For every $\gamma > 0$, if $a, x \sim \mathbb{SH}^n$ with $\gamma\ln n < n$ are drawn independently, then there is a constant $c_2(\gamma) > 0$, such that*

$$\text{Prob}\left\{a^\top x\geq\sqrt{\frac{\gamma\ln n}{n}}\right\}\geq\frac{c_2(\gamma)}{n^{2\gamma}\sqrt{\ln n}}.$$

Proof. By the symmetricity of $\mathbb{SH}^n$, we may without loss of generality assume that a is a given vector in $\mathbb{SH}^n$, e.g. $a = (1, 0, \ldots, 0)^\top$. Let η_i $(i = 1, 2, \ldots, n)$ be i.i.d. standard normal random variables, then $x = \eta/\|\eta\|_2$ and $a^\top x = \eta_1/\|\eta\|_2$.

First, we have for $n \geq 2$

$$\begin{aligned}
\text{Prob}\left\{\eta_1 \geq 2\sqrt{\gamma \ln n}\right\} &= \int_{2\sqrt{\gamma \ln n}}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx \\
&\geq \int_{2\sqrt{\gamma \ln n}}^{4\sqrt{\gamma \ln n}} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx \\
&\geq \int_{2\sqrt{\gamma \ln n}}^{4\sqrt{\gamma \ln n}} \frac{1}{\sqrt{2\pi}} \frac{x}{4\sqrt{\gamma \ln n}} e^{-\frac{x^2}{2}} dx \\
&= \frac{1}{\sqrt{32\pi\gamma \ln n}} \left(\frac{1}{n^{2\gamma}} - \frac{1}{n^{8\gamma}}\right).
\end{aligned}$$

Secondly, we have

$$\text{Prob}\left\{\|\eta\|_2 \geq 2\sqrt{n}\right\} \leq e^{-\frac{2n}{3}}. \tag{5.10}$$

To see why (5.10) holds, we may use a result on the χ^2-distribution estimation by Laurent and Massart (Lemma 1 of [89]): For any vector $b = (b_1, b_2, \ldots, b_n)^\top$ with $b_i \geq 0$ $(i = 1, 2, \ldots, n)$, denote $z = \sum_{i=1}^n b_i(\eta_i^2 - 1)$, then for any $t > 0$,

$$\text{Prob}\left\{z \geq 2\|b\|_2\sqrt{t} + 2\|b\|_\infty t\right\} \leq e^{-t}.$$

Letting b to be the all-one vector and $t = \frac{2n}{3}$ leads to

$$\text{Prob}\left\{\|\eta\|_2^2 \geq \frac{7n}{3} + \sqrt{\frac{8}{3}}n\right\} \leq e^{-\frac{2n}{3}},$$

which implies (5.10).

By these two inequalities, we conclude that

$$\begin{aligned}
\text{Prob}\left\{a^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} &= \text{Prob}\left\{\frac{\eta_1}{\|\eta\|_2} \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} \\
&\geq \text{Prob}\left\{\eta_1 \geq 2\sqrt{\gamma \ln n}, \|\eta\|_2 \leq 2\sqrt{n}\right\} \\
&\geq \text{Prob}\left\{\eta_1 \geq 2\sqrt{\gamma \ln n}\right\} - \text{Prob}\left\{\|\eta\|_2 \geq 2\sqrt{n}\right\} \\
&\geq \frac{1}{\sqrt{32\gamma\pi \ln n}} \left(\frac{1}{n^{2\gamma}} - \frac{1}{n^{8\gamma}}\right) - e^{-\frac{2n}{3}}.
\end{aligned}$$

Therefore, there exists $n_0(\gamma) > 0$, depending only on γ, such that

$$\text{Prob}\left\{a^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} \geq \frac{1}{\sqrt{32\gamma\pi \ln n}} \left(\frac{1}{n^{2\gamma}} - \frac{1}{n^{8\gamma}}\right) - e^{-\frac{2n}{3}} \geq \frac{1}{2n^{2\gamma}\sqrt{32\gamma\pi \ln n}} \; \forall\, n \geq n_0(\gamma).$$

On the other hand, $0 < \gamma < \frac{n}{\ln n}$ implies that $\text{Prob}\left\{a^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} > 0$. Therefore

$$\min_{n<n_0(\gamma),\, \gamma \ln n<n,\, n\in\mathbb{Z}} \text{Prob}\left\{a^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} \cdot n^{2\gamma}\sqrt{\ln n} = t(\gamma) > 0,$$

where $t(\gamma)$ depends only on γ. Finally, letting $c_2(\gamma) = \min\left\{t(\gamma), \frac{1}{2\sqrt{32\gamma\pi}}\right\}$ proves the lemma. □

We remark that similar bound was proposed by Brieden et al. (Lemma 5.1 in [86], also in [87]), where the authors showed that

$$\text{Prob}\left\{a^\top x \geq \sqrt{\frac{\ln n}{n}}\right\} \geq \frac{1}{10\sqrt{\ln n}}\left(1 - \frac{\ln n}{n}\right)^{\frac{n-1}{2}},$$

for any $n \geq 2$. Lemma 5.2.7 gives a more flexible bound by incorporating the parameter γ, though the probability bound at $\gamma = 1$ is worse. Now, for any vector $a \in \mathbb{R}^n$, as $a/\|a\|_2 \in \mathbb{SH}^n$, we have for $x \sim \mathbb{SH}^n$

$$\text{Prob}\left\{a^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\|a\|_2\right\} = \text{Prob}\left\{\left(\frac{a}{\|a\|_2}\right)^\top x \geq \sqrt{\frac{\gamma \ln n}{n}}\right\} \geq \frac{c_2(\gamma)}{n^{2\gamma}\sqrt{\ln n}}, \quad (5.11)$$

which implies (5.6) holds when $d = 1$. To proceed to the high order case, let us introduce the following intermediate result, which is analogous to Lemma 5.2.5 in previous subsection.

Lemma 5.2.8. *If $x \sim \mathbb{SH}^n$, then for any matrix $A \in \mathbb{R}^{m\times n}$,*

$$\text{Prob}\left\{\|Ax\|_2 \geq \frac{1}{\sqrt{2n}}\|A\|_2\right\} \geq \frac{1}{4n}.$$

Proof. Let $A^\top A = P^\top \Lambda P$, where P is orthonormal and $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$ (since $A^\top A$ is positive semidefinite). Denote $y = Px$. Since P is orthonormal and $x \sim \mathbb{SH}^n$, we have $y \sim \mathbb{SH}^n$. Notice that $\|Ax\|_2^2 = x^\top A^\top Ax = x^\top P^\top \Lambda P x = y^\top \Lambda y = \sum_{i=1}^n \lambda_i y_i^2$ and $\|A\|_2^2 = \text{tr}\,(A^\top A) = \sum_{i=1}^n \lambda_i$, and the target probability is then

$$\text{Prob}\left\{\|Ax\|_2 \geq \frac{1}{\sqrt{2n}}\|A\|_2\right\} = \text{Prob}\left\{\|Ax\|_2^2 \geq \frac{1}{2n}\|A\|_2^2\right\} = \text{Prob}\left\{\sum_{i=1}^n \lambda_i y_i^2 \geq \frac{1}{2n}\sum_{i=1}^n \lambda_i\right\},$$

where $y \sim \mathbb{SH}^n$.

By the symmetricity of uniform distribution on the sphere, we have $\mathsf{E}[y_1^2] = \mathsf{E}[y_2^2] = \cdots = \mathsf{E}[y_n^2]$. Combining with $\mathsf{E}[\sum_{i=1}^n y_i^2] = 1$ leads to $\mathsf{E}[y_i^2] = \frac{1}{n}$ for all $1 \le i \le n$. Therefore

$$\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right] = \sum_{i=1}^n \lambda_i \mathsf{E}[y_i^2] = \frac{1}{n}\sum_{i=1}^n \lambda_i.$$

We are going to complete the proof by the Paley-Zygmund inequality. To this end, let us estimate $\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]^2$. Again by the symmetricity of uniform distribution on the sphere, we have $\mathsf{E}[y_i^4] = \alpha$ for all $1 \le i \le n$, and $\mathsf{E}[y_i^2 y_j^2] = \beta$ for any $1 \le i < j \le n$, where $\alpha, \beta > 0$ are constants to be to be determined. First

$$1 = \mathsf{E}\left[\sum_{i=1}^n y_i^2\right]^2 \ge \mathsf{E}\left[\sum_{i=1}^n y_i^4\right] = \alpha n \implies \alpha \le \frac{1}{n}.$$

Next

$$0 \le \mathsf{E}[y_1^2 - y_2^2]^2 = \mathsf{E}[y_1^4] + \mathsf{E}[y_2^4] - 2\mathsf{E}[y_1^2 y_2^2] = 2\alpha - 2\beta \implies \beta \le \alpha \le 1/n.$$

Noticing that $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n \ge 0$ leads to

$$\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]^2 = \alpha \sum_{i=1}^n \lambda_i^2 + 2\beta \sum_{1 \le i < j \le n} \lambda_i \lambda_j \le \frac{1}{n}\left(\sum_{i=1}^n \lambda_i\right)^2 = n\left(\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]\right)^2.$$

Finally, by the Paley-Zygmund inequality, we have

$$\begin{aligned}\text{Prob}\left\{\sum_{i=1}^n \lambda_i y_i^2 \ge \frac{1}{2n}\sum_{i=1}^n \lambda_i\right\} &= \text{Prob}\left\{\sum_{i=1}^n \lambda_i y_i^2 \ge \frac{1}{2}\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]\right\} \\ &\ge \left(1 - \frac{1}{2}\right)^2 \frac{\left(\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]\right)^2}{\mathsf{E}\left[\sum_{i=1}^n \lambda_i y_i^2\right]^2} \ge \frac{1}{4n}.\end{aligned}$$

□

With the above preparations, the proof of (5.6) in Theorem 5.2.1 now follows from a similar induction argument as the proof of (5.5); the details are omitted here. Essentially, Lemma 5.2.7 helps with the basis case, and Lemma 5.2.8 helps to complete the inductive step.

5.3 Homogeneous Polynomial Function in Random Variables

The previous section is concerned with tensor forms of independent entry vectors. One important aspect of the tensors is the connection to the polynomial functions. As is well known, a homogeneous d-th degree polynomial uniquely determines a super-symmetric tensor of d entry vectors. In this section we shall discuss the probability for polynomial function of random variables. In our discussion, the notion of *square-free* tensor plays an important role. Essentially, in the case of matrices, 'square-free' is equivalent to that the diagonal elements are all zero. For a general tensor $\mathcal{A} = (a_{i_1 i_2 \ldots i_d})$, 'square-free' means that $a_{i_1 i_2 \ldots i_d} = 0$ whenever at least two indices are equal.

Theorem 5.3.1. *Let $\mathcal{A} \in \mathbb{R}^{n^d}$ be a square-free super-symmetric tensor of order d, and let $f(x) = \mathcal{A}(x, x, \ldots, x)$ be a homogeneous polynomial function induced by $\mathcal{A}$. If $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top$ are independent random variables with $\mathsf{E}\xi_i = 0, \mathsf{E}\xi_i^2 = 1, \mathsf{E}\xi_i^4 \leq \kappa$ for $i = 1, 2, \ldots, n$, then*

$$\text{Prob}\left\{ f(\xi) \geq \sqrt{\frac{d!}{16\kappa}}\, \|\mathcal{A}\|_2 \right\} \geq \frac{2\sqrt{3} - 3}{9d^2(d!)^2 36^d \kappa^d}, \tag{5.12}$$

$$\text{Prob}\left\{ f(\xi) \geq \sqrt{\frac{d!}{16\kappa n^d}}\, \|\mathcal{A}\|_1 \right\} \geq \frac{2\sqrt{3} - 3}{9d^2(d!)^2 36^d \kappa^d}. \tag{5.13}$$

Compared to Theorem 5.2.1 in the previous section, Theorem 5.3.1 only requires the random variables to be independent from each other, and each with a bounded kurtosis, including the Bernoulli random variables and normal random variables as special cases. It is easy to verify that, under the square-free property of $\mathcal{A}$, together with the assumptions $\mathsf{E}\xi_i = 0$ and ξ_i's are independent from each other $(i = 1, 2, \ldots, n)$, we then have $\mathsf{E}[f(\xi)] = 0$. Since $\mathsf{E}\xi_i^2 = 1\,(i = 1, 2, \ldots, n)$, we compute that $\text{Var}\,(f(\xi)) = \Theta\left(\|\mathcal{A}\|_2^2\right)$. This means that the standard deviation of $f(\xi)$ is in the same order of $\|\mathcal{A}\|_2$. Assertion (5.12) essentially states that given any set of independent random variables with bounded kurtosis, any square-free polynomial of these random variables will have a certain thickness of the tail at some point.

The proof for Theorem 5.3.1 is technically involved, and we shall delegate the details to the next section. Although our main results in Theorem 5.3.1 are valid for arbitrary

random variables, it is interesting to discuss its implications when the random variables are uniform distributions on $\mathbb{B}^n$ and $\mathbb{SH}^n$. In case of quadratic polynomial of Bernoulli random variables, we have the following:

Proposition 5.3.2. *If A is a diagonal-free symmetric matrix and $\xi \sim \mathbb{B}^n$, then*

$$\text{Prob}\left\{\xi^\top A\xi \geq \frac{\|A\|_2}{2\sqrt{30}}\right\} \geq \frac{2\sqrt{3}-3}{135}.$$

The proof of this proposition will be discussed in appendix too. We remark that Proposition 5.3.2 is an extension to the result of Ben-Tal, Nemirovskii, and Roos [19] where it was shown that Prob $\left\{x^\top Ax \geq 0\right\} \geq \frac{1}{8n^2}$, and the result of He et al. [21] where it was shown that Prob $\left\{x^\top Ax \geq 0\right\} \geq 0.03$. Essentially, Proposition 5.3.2 is on the probability of *a strict tail* rather than the probability above the mean.

Proposition 5.3.3. *Let $\mathcal{A} \in \mathbb{R}^{n^d}$ be a square-free super-symmetric tensor of order d, and let $f(x) = \mathcal{A}(x, x, \ldots, x)$ be a homogeneous polynomial function induced by $\mathcal{A}$. If $\xi \sim \mathbb{B}^n$, then*

$$\text{Prob}\left\{f(\xi) \geq \sqrt{\frac{d!}{16n^d}}\,\|\mathcal{A}\|_1\right\} \geq \frac{2\sqrt{3}-3}{9d^2(d!)^2 36^d}.$$

*Moreover, the order of magnitude $n^{-\frac{d}{2}}$ inside '*Prob*' cannot be improved for $d = 2, 4$.*

As a remark, Proposition 5.3.3 can be seen as an instance of (5.1) in the case $f(X) = \mathcal{A} \bullet X$, $S = \{X \in \mathbb{B}^{n^d} : X \text{ is super-symmetric}\}$ and $S_0 = \{X \in S : \text{rank}\,(X) = 1\}$. The probability bound in Proposition 5.3.3 directly follows from (5.13), since $\mathsf{E}\xi_i = 0, \mathsf{E}\xi_i^2 = \mathsf{E}\xi_i^4 = 1$ for all $i = 1, 2, \ldots, n$. It remains to show that even in this special case, the bounds are tight when $d = 2$ and $d = 4$, which are illustrated by the following examples.

Example 5.3.4. *For the case $d = 2$, define $A = I - E$, where I is the identity and E is the all-one matrix. In this case, for any $x \in \mathbb{B}^n$, $x^\top Ax = n - (e^\top x)^2 \leq n$ and $\|A\|_1 = n^2 - n$. Therefore $x^\top Ax/\|A\|_1 \leq 1/(n-1)$ for any $x \in \mathbb{B}^n$, implying that the ratio cannot be better than $\Theta\left(n^{-1}\right)$ for any positive probability.*

Example 5.3.5. *For the case $d = 4$, define $\mathcal{A}$ to be the square-free tensor of order 4, with all non-square-free components being -1. It is obvious that $\|\mathcal{A}\|_1 = \Theta\left(n^4\right)$, and*

for any $x \in \mathbb{B}^n$

$$\begin{aligned}
\mathcal{A}(x,x,x,x) &= \sum_{i=1}^{n} x_i^4 + 12 \sum_{i\neq j, j\neq k, i\neq k} x_i^2 x_j x_k + 6\sum_{i\neq j} x_i^2 x_j^2 + 4\sum_{i\neq j} x_i^3 x_j - \left(\sum_{i=1}^{n} x_i\right)^4 \\
&= n + 12(n-2)\sum_{j\neq k} x_j x_k + 3n(n-1) + 4\sum_{i\neq j} x_i x_j - \left(\sum_{i=1}^{n} x_i\right)^4 \\
&= 3n^2 - 2n + (6n-10)\sum_{j\neq k} 2x_j x_k - \left(\sum_{i=1}^{n} x_i\right)^4 \\
&= 3n^2 - 2n + (6n-10)\left(\left(\sum_{i=1}^{n} x_i\right)^2 - n\right) - \left(\sum_{i=1}^{n} x_i\right)^4 \\
&= 3n^2 - 2n - n(6n-10) + (3n-5)^2 - \left(\left(\sum_{i=1}^{n} x_i\right)^2 - (3n-5)\right)^2 \\
&\leq 6n^2 - 22n + 25.
\end{aligned}$$

Thus we have $\mathcal{A}(x,x,x,x)/\|\mathcal{A}\|_1 \leq \Theta\left(n^{-2}\right)$*, implying that the ratio cannot be better than* $\Theta\left(n^{-2}\right)$ *for any positive probability.*

We believe that examples of the above type exist for any given $d \geq 4$; however, so far we are unable to explicitly construct a general example.

Let us now specialize the random variables to be uniformly distributed on the hypersphere. Since the components of the unit vector are not independent, we cannot directly apply Theorem 5.3.1. However, similar results can still be obtained.

Proposition 5.3.6. *Let* $\mathcal{A} \in \mathbb{R}^{n^d}$ *be a square-free super-symmetric tensor of order* d*, and let* $f(x) = \mathcal{A}(x, x, \ldots, x)$ *be a homogeneous polynomial function induced by* $\mathcal{A}$*. If* $\eta \sim \mathbb{SH}^n$*, then*

$$\text{Prob}\left\{f(\eta) \geq \sqrt{\frac{d!}{48(4n)^d}}\,\|\mathcal{A}\|_2\right\} \geq \frac{2\sqrt{3}-3}{9d^2(d!)^2 108^d} - e^{-\frac{2n}{3}}.$$

Proof. Let $\eta = \xi/\|\xi\|_2$ with $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top$ being i.i.d. standard normal random variables. Since $\mathsf{E}\xi_i = 0, \mathsf{E}\xi_i^2 = 1, \mathsf{E}\xi_i^4 = 3$ for all $1 \leq i \leq n$, by applying (5.12) in Theorem 5.3.1 with $\kappa = 3$, we have

$$\text{Prob}\left\{f(\xi) \geq \sqrt{\frac{d!}{48}}\,\|\mathcal{A}\|_2\right\} \geq \frac{2\sqrt{3}-3}{9d^2(d!)^2 108^d}.$$

Together with (5.10), we have

$$\begin{aligned}\text{Prob}\left\{f(\eta) \geq \sqrt{\frac{d!}{48(4n)^d}}\,\|\mathcal{A}\|_2\right\} &= \text{Prob}\left\{f\left(\frac{\xi}{\|\xi\|_2}\right) \geq \sqrt{\frac{d!}{48(4n)^d}}\,\|\mathcal{A}\|_2\right\} \\ &\geq \text{Prob}\left\{f(\xi) \geq \sqrt{d!/48}\,\|\mathcal{A}\|_2, \|\xi\|_2 \leq 2\sqrt{n}\right\} \\ &\geq \text{Prob}\left\{f(\xi) \geq \sqrt{d!/48}\,\|\mathcal{A}\|_2\right\} - \text{Prob}\left\{\|\xi\|_2 \geq 2\sqrt{n}\right\} \\ &\geq \frac{2\sqrt{3}-3}{9d^2(d!)^2 108^d} - e^{-\frac{2n}{3}}.\end{aligned}$$

□

Before concluding this section, we remark that Proposition 5.3.6 can still be categorized to the type of (5.1) with $f(X) = F \bullet X$, $S = \{X \in \mathbb{S}^{n^d} : X \text{ is super-symmetric}\}$ and $S_0 = \{X \in S : \text{rank}\,(X) = 1\}$. Luo and Zhang [22] offered a constant lower bound for the probability that a homogeneous quartic form of a zero mean multivariate normal distribution lies above its mean. In particular, by restricting the distributions to be i.i.d. standard normals and quartic form to be square-free, applying Theorem 5.3.1 in the case of $d = 4$, we obtain a constant bound for the probability that the quartic form above the mean *plus* some constant times the standard deviation. We may view this as a strengthening of the result in [22].

5.4 Proofs of Theorem 5.3.1 and Proposition 5.3.2

The section is devoted to the proof of Theorem 5.3.1, among which Proposition 5.3.2 is proved as a byproduct. First, we observe that $\|\mathcal{A}\|_2 \geq n^{-\frac{d}{2}}\|\mathcal{A}\|_1$ since $\mathcal{A} \in \mathbb{R}^{n^d}$, and thus (5.13) can be immediately derived from (5.12). Hence we shall focus on (5.12).

Furthermore, we observe that Theorem 5.3.1 is almost equivalent to the fact that any homogeneous polynomial function of independent random variables with bounded kurtosis should also have a bounded kurtosis itself, as formulated as follows:

Theorem 5.4.1. *Let $\mathcal{A} \in \mathbb{R}^{n^d}$ be a square-free super-symmetric tensor of order d, and let $f(x) = \mathcal{A}(x, x, \ldots, x)$ be a homogeneous polynomial function induced by $\mathcal{A}$. If $\xi = (\xi_1, \xi_2, \ldots, \xi_n)^\top$ are independent random variables with $\mathsf{E}\xi_i = 0, \mathsf{E}\xi_i^2 = 1, \mathsf{E}\xi_i^4 \leq \kappa$ for all $i = 1, 2, \ldots, n$, then $\mathsf{E}f^4(\xi) \leq d^2(d!)^2 36^d \kappa^d (\mathsf{E}f^2(\xi))^2$.*

Before proving the theorem, let us note another important fact required in the proof, namely if a random variable has a bounded kurtosis, then it has a constant probability above the mean plus some constant proportion of the standard deviation.

Lemma 5.4.2. *For any random variable z with its kurtosis upper bounded by $\kappa > 0$, namely*

$$\mathsf{E}[z - \mathsf{E}z]^4 \leq \kappa \left(\mathsf{E}[z - \mathsf{E}z]^2\right)^2,$$

we have

$$\text{Prob}\left\{z \geq \mathsf{E}z + \frac{\sqrt{\text{Var}(z)}}{4\sqrt{\kappa}}\right\} \geq \frac{2\sqrt{3}-3}{9\kappa}.$$

Proof. By normalizing z, i.e., letting $y = (z - \mathsf{E}z)/\sqrt{\text{Var}(z)}$, we shall have $\mathsf{E}y = 0$, $\mathsf{E}y^2 = 1$ and $\mathsf{E}y^4 \leq \kappa$. Thus we only need to show Prob $\left\{y \geq \frac{1}{4\sqrt{\kappa}}\right\} \geq \frac{2\sqrt{3}-3}{9\kappa}$.

Denote $x = t - y$, where the constant $t > 0$ will be decided later. We have

$$\begin{aligned}
\mathsf{E}x &= t - \mathsf{E}y = t, \\
\mathsf{E}x^2 &= t^2 - 2t\mathsf{E}y + \mathsf{E}y^2 = t^2 + 1, \\
\mathsf{E}x^4 &= t^4 - 4t^3\mathsf{E}y + 6t^2\mathsf{E}y^2 - 4t\mathsf{E}y^3 + \mathsf{E}y^4 \leq t^4 + 6t^2 + 4t\sqrt{\kappa} + \kappa,
\end{aligned}$$

where $(\mathsf{E}y^3)^2 \leq \mathsf{E}y^2\mathsf{E}y^4 \leq \kappa$ is applied in the last inequality.

By applying Theorem 2.3 of [84], for any constant $v > 0$

$$\begin{aligned}
\text{Prob}\{y \geq t\} &= \text{Prob}\{x \leq 0\} \\
&\geq \frac{4(2\sqrt{3}-3)}{9}\left(-\frac{2\mathsf{E}x}{v} + \frac{3\mathsf{E}x^2}{v^2} - \frac{\mathsf{E}x^4}{v^4}\right) \\
&\geq \frac{4(2\sqrt{3}-3)}{9}\left(-\frac{2t}{v} + \frac{3t^2+3}{v^2} - \frac{t^4+6t^2+4t\sqrt{\kappa}+\kappa}{v^4}\right) \\
\left(\text{let } t = \frac{1}{4\sqrt{\kappa}} \text{ and } v = \sqrt{\kappa}\right) &= \frac{4(2\sqrt{3}-3)}{9}\left(-\frac{1}{2\kappa} + \frac{3}{16\kappa^2} + \frac{3}{\kappa} - \frac{1}{256\kappa^4} - \frac{6}{16\kappa^3} - \frac{1}{\kappa^2} - \frac{1}{\kappa}\right) \\
&= \frac{4(2\sqrt{3}-3)}{9}\left(\frac{24}{16\kappa} - \frac{13}{16\kappa^2} - \frac{6}{16\kappa^3} - \frac{1}{256\kappa^4}\right) \\
(\text{notice } \kappa \geq \mathsf{E}y^4 \geq (\mathsf{E}y^2)^2 = 1) &\geq \frac{4(2\sqrt{3}-3)}{9} \cdot \frac{4}{16\kappa} = \frac{2\sqrt{3}-3}{9\kappa}.
\end{aligned}$$

□

Let us now prove Theorem 5.4.1. We start with a special case when $d = 2$ and ξ are symmetric Bernoulli random variables, which helps to illustrate the ideas underlying the proof for the general case.

Proposition 5.4.3. *Let $A \in \mathbb{R}^{n\times n}$ be a diagonal-free symmetric matrix, and let $f(x) = x^\top Ax$. If $\xi \sim \mathbb{B}^n$, then $\mathsf{E}f^4(\xi) \le 15(\mathsf{E}f^2(\xi))^2$.*

Proof. Rewrite $y = f(\xi) = \sum_\sigma a_\sigma \xi^\sigma$, where $\sigma \in \Pi := \{(1,2),(1,3),\dots,(n-1,n)\}$ and $\xi^{(i,j)} := \xi_i\xi_j$. Since $\mathsf{E}\xi_i^d = 0$ for odd d and $\mathsf{E}\xi_i^d = 1$ for even d, the non-zero terms in $\mathsf{E}y^4$ are all in the forms of $a_{ij}a_{ij}a_{ij}a_{ij}$, $a_{ij}a_{ij}a_{ik}a_{ik}$, $a_{ij}a_{ij}a_{k\ell}a_{k\ell}$ and $a_{ij}a_{ik}a_{j\ell}a_{k\ell}$, where we assume i,j,k,ℓ are distinctive. Let us count the different types of terms.

Type A: $a_{ij}a_{ij}a_{ij}a_{ij}$. The total number of such type of terms is $\binom{n}{2}$;

Type B: $a_{ij}a_{ij}a_{ik}a_{ik}$. The total number of such type of terms is $n\cdot\binom{n-1}{2}\cdot\binom{4}{2}$;

Type C: $a_{ij}a_{ij}a_{k\ell}a_{k\ell}$. The total number of such type of terms is $\binom{n}{4}\cdot 3\cdot\binom{4}{2}$;

Type D: $a_{ij}a_{ik}a_{j\ell}a_{k\ell}$. The total number of such type of terms is $\binom{n}{4}\cdot 3\cdot 4!$.

Notice that

$$(\mathsf{E}y^2)^2 = \left(\sum_{\sigma\in\Pi} a_\sigma^2\right)^2 = \sum_{\sigma\in\Pi} a_\sigma^4 + 2\sum_{\sigma_1\neq\sigma_2} a_{\sigma_1}^2 a_{\sigma_2}^2 =: \text{`Part I'} + \text{`Part II'}.$$

Type A terms constitute exactly 'Part I' in $(\mathsf{E}y^2)^2$; each item of Types B and C will appear exactly once in 'Part II' of $(\mathsf{E}y^2)^2$; each term of Type D can be bounded by an average of two terms in 'Part II' of $(\mathsf{E}y^2)^2$ since $a_{ij}a_{ik}a_{j\ell}a_{k\ell} \le (a_{ij}^2a_{k\ell}^2 + a_{ik}^2a_{j\ell}^2)/2$. The number of the terms of Types B, C and D is:

$$n\cdot\binom{n-1}{2}\binom{4}{2} + \binom{n}{4}\cdot 3\cdot\binom{4}{2} + \binom{n}{4}\cdot 3\cdot 4! = \frac{n(n-1)(n-2)(15n-33)}{4} =: N$$

and there are

$$\binom{n}{2}\cdot\left(\binom{n}{2}-1\right) = \frac{n(n-1)(n-2)(n+1)}{4} =: N'$$

terms in 'Part II' of $(\mathsf{E}y^2)^2$. Clearly $N \le 15N'$, which leads to $\mathsf{E}y^4 \le 15(\mathsf{E}y^2)^2$. □

We are now in a position to prove Proposition 5.3.2, which follows from Proposition 5.4.3 and Lemma 5.4.2.

Proof of Proposition 5.3.2

Proof. Since A is diagonal-free and symmetric, it is easy to verify $\mathsf{E}[\xi^\top A\xi] = 0$ and

$$\mathrm{Var}(\xi^\top A\xi) = \sum_{\sigma\in\Pi} a_\sigma^2 = 4\sum_{\sigma\in\Pi}(a_\sigma/2)^2 = 2\|A\|_2^2.$$

By Lemma 5.4.2 we have $\mathrm{Prob}\left\{\xi^\top A\xi \geq \frac{\sqrt{\mathrm{Var}(\xi^\top A\xi)}}{4\sqrt{15}}\right\} \geq \frac{2\sqrt{3}-3}{135}$, the desired inequality holds. □

Let us now come to the proof of main theorem in the appendix.

Proof of Theorem 5.4.1

Proof. Let $I := \{1, 2, \ldots, n\}$ be the index set, and Π be the set containing all the combinations of d distinctive indices in I. Obviously $|\Pi| = \binom{n}{d}$. For any $\pi \in \Pi$, we denote $x^\pi := \prod_{i\in\pi} x_i$ and $x^{\pi_1+\pi_2} := x^{\pi_1}x^{\pi_2}$ (e.g. $x^{\{1,2\}} = x_1x_2$ and $x^{\{1,2\}+\{1,3\}} = x^{\{1,2\}}x^{\{1,3\}} = x_1x_2 \cdot x_1x_3 = x_1^2x_2x_3$).

Since $\mathcal{A}$ is square-free and super-symmetric, y can be written as $\sum_{\pi\in\Pi} a_\pi x^\pi$, or simply $\sum_\pi a_\pi x^\pi$ (whenever we write summation over π, it means the summation over all $\pi \in \Pi$). We thus have

$$\mathsf{E}y^2 = \mathsf{E}\left[\sum_{\pi_1,\pi_2} a_{\pi_1}x^{\pi_1}a_{\pi_2}x^{\pi_2}\right] = \sum_{\pi_1,\pi_2} a_{\pi_1}a_{\pi_2}\mathsf{E}x^{\pi_1+\pi_2} = \sum_{\pi_1=\pi_2} a_{\pi_1}a_{\pi_2}\mathsf{E}x^{\pi_1+\pi_2} = \sum_\pi a_\pi^2.$$

Our task is to bound

$$\mathsf{E}y^4 = \mathsf{E}\left[\sum_{\pi_1,\pi_2,\pi_3,\pi_4} a_{\pi_1}x^{\pi_1}a_{\pi_2}x^{\pi_2}a_{\pi_3}x^{\pi_3}a_{\pi_4}x^{\pi_4}\right] = \sum_{\pi_1,\pi_2,\pi_3,\pi_4} a_{\pi_1}a_{\pi_2}a_{\pi_3}a_{\pi_4}\,\mathsf{E}x^{\pi_1+\pi_2+\pi_3+\pi_4}. \tag{5.14}$$

For any combination quadruple $\{\pi_1, \pi_2, \pi_3, \pi_4\}$, there are in total $4d$ indices, with each index appearing at most 4 times. Suppose there are a number of indices appearing 4 times, b number of indices appearing 3 times, c number of indices appearing twice, and g number of indices appearing once. Clearly $4a+3b+2c+g = 4d$. In order to compute the summation of all the terms $a_{\pi_1}a_{\pi_2}a_{\pi_3}a_{\pi_4}\,\mathsf{E}x^{\pi_1+\pi_2+\pi_3+\pi_4}$ over $\pi_1, \pi_2, \pi_3, \pi_4 \in \Pi$ in (5.14), we shall group them according to different $\{a, b, c, g\}$.

1. $g \geq 1$: as we know $\mathsf{E}x_i = 0$ for all $i \in I$, all the terms in this group will vanish.

2. $b = c = g = 0$: the summation of all the terms in this group is

$$\sum_{\pi_1=\pi_2=\pi_3=\pi_4} a_{\pi_1} a_{\pi_2} a_{\pi_3} a_{\pi_4} \mathsf{E}x^{\pi_1+\pi_2+\pi_3+\pi_4} = \sum_{\pi_1} a_{\pi_1}^4 \mathsf{E}x^{4\pi_1} \leq \kappa^d \sum_{\pi} a_{\pi}^4.$$

3. $g = 0$ and $b + c \geq 1$: we shall classify all the terms in this group step by step. In the following, we assume $|\Pi| \geq 2$ and $n \geq d + 1$ to avoid triviality.

   - It is clear that $4a + 3b + 2c = 4d$, $0 \leq a \leq d - 1$, $0 \leq b \leq (4d - 4a)/3$ and b must be even. In this group, the number of different $\{a, b, c\}$ is at most $\sum_{a=0}^{d-1} \left(1 + \lfloor \frac{4d-4a}{6} \rfloor\right) \leq d^2$.
   - For any given triple $\{a, b, c\}$, there are total $\binom{n}{a}\binom{n-a}{b}\binom{n-a-b}{c}$ number of distinctive ways to assign indices. Clearly, we have $\binom{n}{a}\binom{n-a}{b}\binom{n-a-b}{c} \leq n!/(n-a-b-c)! \leq n!/(n-2d)_+!$.
   - For any given a indices appearing 4 times, b indices appearing 3 times, and c indices appearing twice, we shall count how many distinctive ways they can form a particular combination quadruple $\{\pi_1, \pi_2, \pi_3, \pi_4\}$ (note that orders do count). For the indices appearing 4 times, they do not have choice but to be located in $\{\pi_1, \pi_2, \pi_3, \pi_4\}$ each once; for indices appearing 3 times, each has at most 4 choices; for indices appearing twice, each has at most 6 choices. Therefore, the total number of distinctive ways to formulate the combination of quadruples is upper bounded by $4^b 6^c \leq 6^{2d}$.
   - For any given combination quadruple $\{\pi_1, \pi_2, \pi_3, \pi_4\}$, noticing that $(\mathsf{E}x_i^3)^2 \leq \mathsf{E}x_i^2 \, \mathsf{E}x_i^4 \leq \kappa$ for all $i \in I$, we have $|\mathsf{E}x^{\pi_1+\pi_2+\pi_3+\pi_4}| \leq \kappa^a \cdot (\sqrt{\kappa})^b \cdot 1^c = \kappa^{a+b/2} \leq \kappa^d$.
   - For any given combination quadruple $\{\pi_1, \pi_2, \pi_3, \pi_4\}$, in this group each combination can appear at most twice. Specifically, if we assume $i \neq j$ (implying $\pi_i \neq \pi_j$), then the forms of $\{\pi_1, \pi_1, \pi_1, \pi_2\}$ and $\{\pi_1, \pi_1, \pi_1, \pi_1\}$ do not appear. The only possible forms are $\{\pi_1, \pi_2, \pi_3, \pi_4\}$, $\{\pi_1, \pi_1, \pi_2, \pi_3\}$ and

$\{\pi_1, \pi_1, \pi_2, \pi_2\}$. We notice that

$$\begin{aligned} a_{\pi_1}a_{\pi_2}a_{\pi_3}a_{\pi_4} &\leq (a_{\pi_1}^2a_{\pi_2}^2 + a_{\pi_1}^2a_{\pi_3}^2 + a_{\pi_1}^2a_{\pi_4}^2 + a_{\pi_2}^2a_{\pi_3}^2 + a_{\pi_2}^2a_{\pi_4}^2 + a_{\pi_3}^2a_{\pi_4}^2)/6, \\ a_{\pi_1}a_{\pi_1}a_{\pi_2}a_{\pi_3} &\leq (a_{\pi_1}^2a_{\pi_2}^2 + a_{\pi_1}^2a_{\pi_3}^2)/2, \\ a_{\pi_1}a_{\pi_1}a_{\pi_2}a_{\pi_2} &= a_{\pi_1}^2a_{\pi_2}^2. \end{aligned}$$

Therefore, in any possible form, each $a_{\pi_1}a_{\pi_2}a_{\pi_3}a_{\pi_4}$ can be *on average* upper bounded by *one* item $a_{\pi_1}^2a_{\pi_2}^2(\pi_1 \neq \pi_2)$ in $\sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2$.

Overall, in this group, by noticing the symmetry of Π, the summation of all the terms is upper bounded by $d^2 \cdot \frac{n!}{(n-2d)+!} \cdot 6^{2d} \cdot \kappa^d$ number of items in form of $a_{\pi_1}^2a_{\pi_2}^2(\pi_1 \neq \pi_2)$ in $\sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2$. Notice that there are in total $|\Pi|(|\Pi|-1)/2 = \frac{1}{2}\binom{n}{d}\left(\binom{n}{d}-1\right)$ items in $\sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2$, and each item is evenly distributed. By symmetry, the summation of all the terms in this group is upper bounded by

$$\frac{d^2 \cdot \frac{n!}{(n-2d)+!} \cdot 6^{2d} \cdot \kappa^d}{\frac{1}{2}\binom{n}{d}\left(\binom{n}{d}-1\right)} \sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2 \leq d^2(d!)^2 36^d\kappa^d \cdot 2 \sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2.$$

Finally, we are able to bound $\mathsf{E}y^4$ by

$$\begin{aligned} \mathsf{E}y^4 &\leq \kappa^d \sum_{\pi} a_\pi^4 + d^2(d!)^2 36^d\kappa^d \cdot 2 \sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2 \\ &\leq d^2(d!)^2 36^d\kappa^d \left(\sum_{\pi} a_\pi^4 + 2\sum_{\pi_1\neq\pi_2} a_{\pi_1}^2a_{\pi_2}^2\right) \\ &= d^2(d!)^2 36^d\kappa^d \left(\sum_{\pi} a_\pi^2\right)^2 = d^2(d!)^2 36^d\kappa^d(\mathsf{E}y^2)^2. \end{aligned}$$

Putting the pieces together, the theorem follows. □

Finally, combining Theorem 5.4.1 and Lemma 5.4.2, and noticing $\mathrm{Var}(f(\xi)) = d!\|\mathcal{A}\|_2^2$ in Theorem 5.3.1, lead us to the probability bound (5.12) in Theorem 5.3.1, which concludes the whole proof.

Chapter 6

New Approximation Algorithms for Real Polynomial Optimization

6.1 Introduction

In this chapter, we study the approximability of polynomial optimization in real variables, this is because they are generally intractable from algorithmic point of view. Recently, a bunch of efforts have been made to find some approximation algorithms with worst case performance guarantee for real polynomial optimization over certain type of constraint. The first results in this direction were made by Luo and Zhang [22], who showed a first approximation ratio for homogeneous quartic optimization problems with quadratic constraints. Around the same time, Ling et al. [32] considered a special quartic optimization model, which is to maximize a biquadratic form over two spherical constraints. Since then, some significant progresses have been made by He et al. [33, 34, 35], where the authors derived a series of approximation methods for optimization of any fixed degree polynomial function under various constrains. More recently, So [90] reinvestigated sphere constrained homogeneous polynomial optimization and proposed a deterministic algorithm with an superior approximation ratio. For most recent development on approximation algorithms for homogeneous polynomial optimization, we refer the interested readers to the monograph of Li et al. [59].

As discussed in Chapter 5, the probability bounds in the form of (5.1) have immediate applications in optimization. In the following, we shall apply the bounds derived

in sections 5.2 and 5.3 to polynomial function optimization problems and derive some noval polynomial-time randomized approximation algorithms, with the approximation ratios improving the existing ones in the literature.

In terms of notations adopted in this chapter, two types of constraint sets $\mathbb{B}^n$ and $\mathbb{SH}^n$ are defined as follows

$$\mathbb{B}^n := \{1, -1\}^n \quad \text{and} \quad \mathbb{SH}^n := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}.$$

The notion $\Theta(f(n))$ signifies that there are positive universal constants α, β and n_0 such that $\alpha f(n) \leq \Theta(f(n)) \leq \beta f(n)$ for all $n \geq n_0$, i.e., the same order of $f(n)$. To avoid confusion, the term *constant* sometimes also refers to a parameter depending only on the dimension of a polynomial function, which is a given number independent of the input data of the problem.

Now our results can be summarized in the following:

1. $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{\ln n_i}{n_i}}\right)$-approximation ratio for

$$\begin{array}{ll} \max & \mathcal{F}(x^1, x^2, \ldots, x^d) \\ \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = 1, 2, \ldots, d. \end{array}$$

   This ratio improves that of $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{1}{n_i}}\right)$ proposed by He, Li, and Zhang [35].

2. $\Theta\left(n^{-\frac{d}{2}}\right)$-approximation ratio for

$$\begin{array}{ll} \max & f(x) := \mathcal{F}(\underbrace{x, x, \ldots, x}_{d}) \\ \text{s.t.} & x \in \mathbb{B}^n, \end{array}$$

   where $f(x)$ is a homogeneous polynomial function with the tensor $\mathcal{F}$ being square-free. This ratio is new. In the literature, when $d \geq 4$ and is even, the only previous approximation ratio for this model was in He, Li, and Zhang [35]; however, the ratio there is a relative one.

3. $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{\ln n_i}{n_i}}\right)$-approximation ratio for

$$\begin{array}{ll} \max & \mathcal{F}(x^1, x^2, \ldots, x^d) \\ \text{s.t.} & x^i \in \mathbb{SH}^{n_i},\ i = 1, 2, \ldots, d. \end{array}$$

This improves the $\prod_{i=1}^{d-2}\sqrt{\frac{1}{n_i}}$ approximation ratio in [33], and achieves the same theoretical bound as in So [90]. However, the algorithm proposed here is straightforward to implement, while the one in [90] is very technical involved.

4. $\Theta\left(n^{-\frac{d}{2}}\right)$-approximation ratio for

$$\begin{array}{ll} \max & f(x) := \mathcal{F}(\underbrace{x, x, \ldots, x}_{d}) \\ \text{s.t.} & x \in \mathbb{SH}^n, \end{array}$$

where $f(x)$ is a homogeneous polynomial function with the tensor $\mathcal{F}$ being square-free. This ratio is new when $d \geq 4$ and is even, since previous approximation ratios in [33, 90] are all relative ones.

5. $\Theta\left(\prod_{n\in N}\sqrt{\frac{\ln n}{n}}\right)$-approximation ratio for

$$\begin{array}{ll} \max & \mathcal{F}(x^1, x^2, \ldots, x^d, y^1, y^2, \ldots, y^{d'}) \\ \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = 1, 2, \ldots, d, \\ & y^j \in \mathbb{SH}^{m_j},\ j = 1, 2, \ldots, d', \end{array}$$

where N is the set of the $d + d' - 2$ smallest numbers in $\{n_1, \ldots, n_d, m_1, \ldots, m_{d'}\}$. This ratio improves that of $\Theta\left(\prod_{i=1}^{d-1}\sqrt{\frac{1}{n_i}}\prod_{j=1}^{d'-1}\sqrt{\frac{1}{m_j}}\right)$ proposed in [35].

6.2 Polynomial Optimization in Binary Variables

The general unconstrained binary polynomial optimization model is $\max_{x\in\mathbb{B}^n} p(x)$, where $p(x)$ is a multivariate polynomial function. He, Li, and Zhang [35] proposed a polynomial-time randomized approximation algorithm with a relative performance ratio. When the polynomial $p(x)$ is homogeneous, this problem has many applications in graph theory; e.g. the max-cut problem [80] and the matrix cut-norm problem [69]. In particular we shall discuss two models in this section:

$$\begin{array}{lll} (B_1) & \max & \mathcal{F}(x^1, x^2, \ldots, x^d) \\ & \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = 1, 2, \ldots, d; \\ (B_2) & \max & f(x) = \mathcal{F}(\underbrace{x, x, \ldots, x}_{d}) \\ & \text{s.t.} & x \in \mathbb{B}^n. \end{array}$$

When $d = 2$, (B_1) is to compute the matrix $\infty \mapsto 1$ norm, which is related to so called matrix cut-norm problem. The current best approximation ratio is 0.56, due to Alon and Naor [69]. When $d = 3$, (B_1) is a slight generalization of the model considered by Khot and Naor [83], where $\mathcal{F}$ was assumed to be super-symmetric and square-free. The approximation ratio estimated in [83] is $\Theta\left(\sqrt{\frac{\ln n_1}{n_1}}\right)$, which is currently the best. Recently, He, Li, and Zhang [35] proposed a polynomial-time randomized approximation algorithm for (B_1) for any fixed degree d, with approximation performance ratio $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{1}{n_i}}\right)$. The results in this subsection will improve this approximation ratio for fixed d, thanks to Theorem 5.2.1.

Algorithm B_1 (Randomized Algorithm for (B_1))

1. Sort and rename the dimensions if necessary, so as to satisfy $n_1 \le n_2 \le \cdots \le n_d$.
2. Randomly and independently generate $\xi^i \sim \mathbb{B}^{n_i}$ for $i = 1, 2, \ldots, d-2$.
3. Solve the following bilinear form optimization problem

$$\begin{array}{ll} \max & \mathcal{F}(\xi^1, \xi^2, \ldots, \xi^{d-2}, x^{d-1}, x^d) \\ \text{s.t.} & x^{d-1} \in \mathbb{B}^{n_{d-1}}, x^d \in \mathbb{B}^{n_d} \end{array}$$

   using the deterministic algorithm of Alon and Naor [69], and get its approximate solution (ξ^{d-1}, ξ^d).
4. Compute the objective value $\mathcal{F}(\xi^1, \xi^2, \ldots, \xi^d)$.
5. Repeat the above procedures $\frac{\prod_{i=1}^{d-2} n_i^\delta}{0.03\,(c_1(\delta))^{d-2}} \ln\frac{1}{\epsilon}$ times for any constant $\delta \in \left(0, \frac{1}{2}\right)$ and choose a solution whose objective function is the largest.

We remark that Algorithm B_1 was already mentioned in [83] for odd d, where a same order of approximation bound as of Theorem 6.2.1 was suggested. However no explicit polynomial-time algorithm and the proof for approximation guarantee were offered there. The approximation ratio for Algorithm B_1 and its proof are as follows.

Theorem 6.2.1. *Algorithm B_1 solves (B_1) in polynomial-time with probability at least $1 - \epsilon$, and approximation performance ratio $\delta^{\frac{d-2}{2}} \prod_{i=1}^{d-2}\sqrt{\frac{\ln n_i}{n_i}}$.*

The proof is based on mathematical induction. Essentially, if an algorithm solves

(B_1) of order $d-1$ approximately with an approximation ratio τ, then there is an algorithm solves (B_1) of order d approximately with an approximation ratio $\tau\sqrt{\frac{\delta \ln n}{n}}$, where n is the dimension of the additional order.

Proof. For given problem degree d, the proof is based on induction on $t = 2, 3, \ldots d$. Suppose $(\xi^1, \xi^2, \ldots, \xi^d)$ is the approximate solution generated by Algorithm B_1. For $t = 2, 3, \ldots, d$, we treat $(\xi^1, \xi^2, \ldots, \xi^{d-t})$ as given parameters and define the following problems

$$\begin{array}{lll} (D_t) & \max & \mathcal{F}(\xi^1, \xi^2, \ldots, \xi^{d-t}, x^{d-t+1}, x^{d-t+2} \ldots, x^d) \\ & \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = d-t+1, d-t+2, \ldots, d, \end{array}$$

whose optimal value is denoted by $v(D_t)$. By applying Algorithm B_1 to (D_t), $(\xi^{d-t+1}, \xi^{d-t}, \ldots, \xi^d)$ can be viewed as an approximate solution generated. In the remaining, we shall prove by induction that for each $t = 2, 3, \ldots, d$,

$$\underset{(\xi^{d-t+1}, \xi^{d-t+2}, \ldots, \xi^d)}{\text{Prob}} \left\{ \mathcal{F}(\xi^1, \xi^2, \ldots, \xi^d) \geq \delta^{\frac{t-2}{2}} \prod_{i=d-t+1}^{d-2} \sqrt{\frac{\ln n_i}{n_i}} v(D_t) \right\} \geq \frac{0.03\,(c_1(\delta))^{t-2}}{\prod_{i=d-t+1}^{d-2} n_i^{\delta}}. \tag{6.1}$$

In other words, $(\xi^{d-t+1}, \xi^{d-t+2}, \ldots, \xi^d)$ has a non-trivial probability to be a $\delta^{\frac{t-2}{2}} \prod_{i=d-t+1}^{d-2} \sqrt{\frac{\ln n_i}{n_i}}$-approximate solution of (D_t).

For the initial case $t = 2$, the deterministic algorithm by Alon and Naor [69] (Step 3 of Algorithm B_1) guarantees a constant ratio, i.e., $\mathcal{F}(\xi^1, \xi^2, \ldots, \xi^d) \geq 0.03\, v(D_2)$, implying (6.1). Suppose now (6.1) holds for $t-1$. To prove that (6.1) holds for t, we notice that $(\xi^1, \xi^2, \ldots, \xi^{d-t})$ are given fixed parameters. Denote $(z^{d-t+1}, z^{d-t+2} \ldots, z^d)$ to be an optimal solution of (D_t), and define the following events

$$\begin{aligned} E_3 &= \left\{ z \in \mathbb{B}^{n_{d-t+1}} \,\middle|\, \mathcal{F}(\xi^1, \ldots, \xi^{d-t}, z, z^{d-t+2} \ldots, z^d) \geq \sqrt{\frac{\delta \ln n_{d-t+1}}{n_{d-t+1}}} v(D_t) \right\}; \\ E_4 &= \left\{ \xi^{d-t+1} \in E_3, \xi^{d-t+2} \in \mathbb{B}^{n_{d-t+2}}, \ldots, \xi^d \in \mathbb{B}^{n_d} \,\middle|\, \right. \\ &\qquad \left. \mathcal{F}(\xi^1, \ldots, \xi^d) \geq \delta^{\frac{t-3}{2}} \prod_{i=d-t+2}^{d-2} \sqrt{\frac{\ln n_i}{n_i}} \mathcal{F}(\xi^1, \ldots, \xi^{d-t}, \xi^{d-t+1}, z^{d-t+2} \ldots, z^d) \right\}. \end{aligned}$$

Then we have

$$\begin{aligned}
&\underset{(\xi^{d-t+1},\ldots,\xi^d)}{\text{Prob}}\left\{\mathcal{F}(\xi^1,\ldots,\xi^d) \geq \delta^{\frac{t-2}{2}}\prod_{i=d-t+1}^{d-2}\sqrt{\tfrac{\ln n_i}{n_i}}\,v(D_t)\right\} \\
\geq\ &\underset{(\xi^{d-t+1},\ldots,\xi^d)}{\text{Prob}}\left\{(\xi^{d-t+1},\ldots,\xi^d)\in E_4\,\middle|\,\xi^{d-t+1}\in E_3\right\}\cdot\underset{\xi^{d-t+1}}{\text{Prob}}\left\{\xi^{d-t+1}\in E_3\right\}. \quad (6.2)
\end{aligned}$$

To lower bound (6.2), first note that $(z^{d-t+2},\ldots,z^d)$ is a feasible solution of (D_{t-1}), and so we have

$$\begin{aligned}
&\underset{(\xi^{d-t+1},\ldots,\xi^d)}{\text{Prob}}\left\{(\xi^{d-t+1},\ldots,\xi^d)\in E_4\,\middle|\,\xi^{d-t+1}\in E_3\right\} \\
\geq\ &\underset{(\xi^{d-t+1},\ldots,\xi^d)}{\text{Prob}}\left\{\mathcal{F}(\xi^1,\ldots,\xi^d) \geq \delta^{\frac{t-3}{2}}\prod_{i=d-t+2}^{d-2}\sqrt{\frac{\ln n_i}{n_i}}\,v(D_{t-1})\,\middle|\,\xi^{d-t+1}\in E_3\right\} \\
\geq\ &\frac{0.03\,(c_1(\delta))^{t-3}}{\prod_{i=d-t+2}^{d-2} n_i^\delta},
\end{aligned}$$

where the last inequality is due to the induction assumption on $t-1$, and Proposition 5.2.6 for joint conditional probability with $\xi = (\xi^{d-t+2},\ldots,\xi^d)$ and $\eta = \xi^{d-t+1}$. Secondly, we have

$$\begin{aligned}
&\underset{\xi^{d-t+1}}{\text{Prob}}\left\{\xi^{d-t+1}\in E_3\right\} \\
=\ &\underset{\xi^{d-t+1}}{\text{Prob}}\left\{\mathcal{F}(\xi^1,\ldots,\xi^{d-t+1},z^{d-t+2},\ldots,z^d) \geq \sqrt{\frac{\delta\ln n_{d-t+1}}{n_{d-t+1}}}\mathcal{F}(\xi^1,\ldots,\xi^{d-t},z^{d-t+1},\ldots,z^d)\right\} \\
=\ &\underset{\xi^{d-t+1}}{\text{Prob}}\left\{\mathcal{F}(\xi^1,\ldots,\xi^{d-t+1},z^{d-t+2},\ldots,z^d) \geq \sqrt{\frac{\delta\ln n_{d-t+1}}{n_{d-t+1}}}\|\mathcal{F}(\xi^1,\ldots,\xi^{d-t},\cdot,z^{d-t+2},\ldots,z^d)\|_1\right\} \\
\geq\ &\frac{c_1(\delta)}{n_{d-t+1}^\delta},
\end{aligned}$$

where the last inequality is due to Theorem 5.2.1 for the case $d=1$. With the above two facts, we can lower bound the right hand side of (6.2), and conclude

$$\underset{(\xi^{d-t+1},\ldots,\xi^d)}{\text{Prob}}\left\{\mathcal{F}(\xi^1,\ldots,\xi^d) \geq \delta^{\frac{t-2}{2}}\prod_{i=d-t+1}^{d-2}\sqrt{\tfrac{\ln n_i}{n_i}}\,v(D_t)\right\} \geq \frac{0.03\,(c_1(\delta))^{t-3}}{\prod_{i=d-t+2}^{d-2} n_i^\delta\cdot\frac{c_1(\delta)}{n_{d-t+1}^\delta}} = \frac{0.03\,(c_1(\delta))^{t-2}}{\prod_{i=d-t+1}^{d-2} n_i^\delta}.$$

As (D_d) is exactly (B_1), Algorithm B_1 solves (B_1) approximately with probability at least $1-\epsilon$. □

We remark that theoretically we may get a better approximate solution, using the 0.56-randomized algorithm in [69] to replace the subroutine in Step 3 of Algorithm B_1, though that algorithm is quite complicated. In a similar vein, we obtain approximation results for (B_2).

Algorithm B_2 (Randomized Algorithm for (B_2))

1. Randomly generate $\xi \sim \mathbb{B}^n$ and compute $f(\xi)$.
2. Repeat this procedure $\frac{9d^2(d!)^2 36^d}{2\sqrt{3}-3} \ln \frac{1}{\epsilon}$ times and choose a solution whose objective function is the largest.

The model (B_2) has been studied extensively in the quadratic cases, i.e., $d = 2$. Goemans and Williamson [80] gave a 0.878-approximation ratio for the case $\mathcal{F}$ being the Laplacian of a given graph. Later, Nesterov [81] gave a 0.63-approximation ratio for the case $\mathcal{F}$ being positive semidefinite. For diagonal-free matrix, the best possible approximation ratio is $\Theta(1/\ln n)$, due to Charikar and Wirth [91], which is also known to be tight. For $d = 3$ and $\mathcal{F}$ is square-free, Knot and Naor [83] gave an $\Theta\left(\sqrt{\frac{\ln n}{n}}\right)$-approximation bound. They also pointed out an iterative procedure to get an $\Theta\left(\frac{\ln^{d/2-1} n}{n^{d/2-1}}\right)$-approximation bound for odd d, which requires a linkage between multilinear tensor function and homogeneous polynomial of any degree (see Lemma 1 of [33]). For general d, He, Li, and Zhang [35] proposed polynomial-time randomized approximation algorithms with approximation ratio $\Theta\left(n^{-\frac{d-2}{2}}\right)$ when $\mathcal{F}$ is square-free for odd d; however for even d, they can only propose a relative approximation ratio $\Theta\left(n^{-\frac{d-2}{2}}\right)$. Now, by virtue of Theorem 5.3.1 (more precisely Proposition 5.3.3), since $\|\mathcal{F}\|_1$ is an upper bound for the optimal value of (B_2), *absolute* approximation ratios are also established when d is even, as shown below.

Theorem 6.2.2. *When d is even and $\mathcal{F}$ is square-free, Algorithm B_2 solves (B_2) in polynomial-time with probability at least $1 - \epsilon$, and approximation performance ratio $\sqrt{\frac{d!}{16n^d}}$.*

6.3 Polynomial Optimization over Hyperspheres

Polynomial function optimization over hyperspheres have much applications in biomedical engineering, material sciences, numerical linear algebra, among many others. Readers are referred to [33, 59] and references therein for more information. Let us consider:

$$\begin{array}{lll} (S_1) & \max & \mathcal{F}(x^1, x^2, \ldots, x^d) \\ & \text{s.t.} & x^i \in \mathbb{SH}^{n_i},\ i = 1, 2, \ldots, d; \\ (S_2) & \max & f(x) = \mathcal{F}(\underbrace{x, x, \ldots, x}_{d}) \\ & \text{s.t.} & x \in \mathbb{SH}^n. \end{array}$$

When $d = 2$, (S_1) and (S_2) reduce to computing matrix spectrum norms and can be solved in polynomial-time. However they are NP-hard when $d \geq 3$. For general d, (S_2) is to compute the largest eigenvalue of the tensor $\mathcal{F}$. As far as approximation algorithms are concerned, He, Li, and Zhang [33] proposed polynomial-time approximation algorithms for (S_1) with approximation ratio $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{1}{n_i}}\right)$. In [33], a generic linkage relating (S_2) and (S_1) is established. This linkage enables one to get a solution with the same approximation ratio (relative ratio for even d though) for (S_2) whenever a solution with an approximation ratio for (S_1) is available. Therefore, let us now focus on (S_1). For (S_1), recently So [90] improved the result of [33] from $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{1}{n_i}}\right)$ to $\Theta\left(\prod_{i=1}^{d-2}\sqrt{\frac{\ln n_i}{n_i}}\right)$. Unfortunately, the method in [90] relies on the equivalence (polynomial-time reduction) between convex optimization and membership oracle queries using the ellipsoid method, and it is computationally impractical. On the other hand, the algorithm that we present below is straightforward, while retaining the same quality of approximation as the result in [90].

Algorithm S_1 (Randomized Algorithm for (S_1))

1. Sort and rename the dimensions if necessary, so as to satisfy $n_1 \leq n_2 \leq \cdots \leq n_d$.
2. Randomly and independently generate $\eta^i \sim \mathbb{SH}^{n_i}$ for $i = 1, 2, \ldots, d-2$.
3. Solve the largest singular value problem

$$\begin{array}{ll} \max & \mathcal{F}(\eta^1, \eta^2, \ldots, \eta^{d-2}, x^{d-1}, x^d) \\ \text{s.t.} & x^{d-1} \in \mathbb{SH}^{n_{d-1}}, x^d \in \mathbb{SH}^{n_d}, \end{array}$$

   and get its optimal solution (η^{d-1}, η^d).
4. Compute the objective value $\mathcal{F}(\eta^1, \eta^2, \ldots, \eta^d)$.
5. Repeat the above procedures $\frac{\prod_{i=1}^{d-2} n_i^{2\gamma} \sqrt{\ln n_i} (c_2(\gamma))^{d-2}}{\ln} \frac{1}{\epsilon}$ times for any constant $\gamma \in (0, \frac{n_1}{\ln n_1})$ and choose a solution whose objective function is the largest.

Theorem 6.3.1. *Algorithm S_1 solves (S_1) in polynomial-time with probability at least $1 - \epsilon$, and approximation performance ratio $\gamma^{\frac{d-2}{2}} \prod_{i=1}^{d-2} \sqrt{\frac{\ln n_i}{n_i}}$.*

The proof is similar to that for Theorem 6.2.1, and is omitted here.

In a similar manner, we obtain approximation results for (S_2).

Algorithm S_2 (Randomized Algorithm for (S_2))

1. Randomly generate $\xi \sim \mathbb{SH}^n$ and compute $f(\xi)$.
2. Repeat this procedure $\frac{9d^2(d!)^2 108^d}{\sqrt{3}-1} \ln \frac{1}{\epsilon}$ times and choose a solution whose objective function is the largest.

The approximation method of model (S_2) has been studied in [33, 90]. As far as we know, the best approximation ratio of this problem is $\Theta\left(\sqrt{\frac{\ln n}{n}}\right)$, but which becomes a relative ratio when d is even. Now notice that $\|\mathcal{F}\|_2$ is an upper bound for the optimal value of (S_2), so Proposition 5.3.6 enables us to obtain an *absolute* approximation ratio of this problem when d is even.

Theorem 6.3.2. *When d is even, $\mathcal{F}$ is square-free and n is sufficiently large, Algorithm S_2 solves (S_2) in polynomial-time with probability at least $1 - \epsilon$, and approximation performance ratio $\sqrt{\frac{d!}{48(4n)^d}}$.*

6.4 Polynomial Function Mixed Integer Programming

This last section of this chapter deals with optimization of polynomial functions under binary variables and variables with spherical constraints mixed up. Such problems have applications in matrix combinatorial problem, vector-valued maximum cut problem; see e.g. [35]. In [35], the authors considered

$$
\begin{array}{lll}
(M_1) & \max & \mathcal{F}(x^1, x^2, \ldots, x^d, y^1, y^2, \ldots, y^{d'}) \\
 & \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = 1, 2, \ldots, d;\ y^j \in \mathbb{SH}^{m_j},\ j = 1, 2, \ldots, d'; \\
(M_2) & \max & \mathcal{F}(\underbrace{x, x, \ldots, x}_{d}, \underbrace{y, y, \ldots, y}_{d'}) \\
 & \text{s.t.} & x \in \mathbb{B}^n, y \in \mathbb{SH}^m; \\
(M_3) & \max & \mathcal{F}(\underbrace{x^1, x^1, \ldots, x^1}_{d_1}, \ldots, \underbrace{x^s, x^s, \ldots, x^s}_{d_s}, \underbrace{y^1, y^1, \ldots, y^1}_{d'_1}, \ldots, \underbrace{y^t, y^t, \ldots, y^t}_{d'_t}) \\
 & \text{s.t.} & x^i \in \mathbb{B}^{n_i},\ i = 1, 2, \ldots, s;\ y^j \in \mathbb{SH}^{m_j},\ j = 1, 2, \ldots, t;
\end{array}
$$

and proposed polynomial-time randomized approximation algorithms when the tensor $\mathcal{F}$ is square-free in x (the binary part). In fact, (M_3) is a generalization of (M_1) and (M_2), and it can also be regarded as a generalization of (B_1), (B_2), (S_1) and (S_2) as well. Essentially the approximative results can be applied by using the linkage we mentioned earlier (see [33]) if approximation result for (M_1) can be established. In fact, (M_1) plays the role as a cornerstone for the whole construction. The approximation ratio for (M_1) derived in [35] is $\Theta\left(\prod_{i=1}^{d-1}\sqrt{\frac{1}{n_i}}\prod_{j=1}^{d'-1}\sqrt{\frac{1}{m_j}}\right)$. The results in Section 5.2 lead to the following improvements:

Theorem 6.4.1. *Denote N to be the set of the $d+d'-2$ smallest numbers in $\{n_1, \ldots, n_d, m_1, \ldots, m_{d'}\}$. (M_1) admits a polynomial-time randomized approximation algorithm with approximation performance ratio $\Theta\left(\prod_{n\in N}\sqrt{\frac{\ln n}{n}}\right)$.*

The method for solving (M_1) is similar to that for solving (B_1) and (S_1), and we shall not repeat the detailed discussions. Basically we shall do multiple trials to get a solution with high probability. For the $d + d' - 2$ numbers in N, if it is the dimension of binary constraints, the algorithm uniformly picks a vector in the discrete hypercube; and if it is the dimension of spherical constraints, the algorithms uniformly pick a vector in the hypersphere. All the randomized procedures will be done independent from each other.

As the first inductive step, we will then come across a bilinear function optimization problem in either of the three possible cases:

- $\max_{x \in \mathbb{B}^n, y \in \mathbb{B}^m} x^\top F y$, which can be solved by the algorithm proposed in Alon and Naor [69] to get a solution with the guaranteed constant approximation ratio;
- $\max_{x \in \mathbb{B}^n, y \in \mathbb{SH}^m} x^\top F y$, which can be solved by the algorithm proposed in He, Li, and Zhang [35] to get a solution with the guaranteed constant approximation ratio;
- $\max_{x \in \mathbb{SH}^n, y \in \mathbb{SH}^m} x^\top F y$, which can be solved by computing the largest singular value of matrix F.

Chapter 7

Approximation Algorithms for Complex Polynomial Optimization

7.1 Introduction

Hitherto, polynomial optimization models under investigation are mostly in the domain of real numbers. Motivated by applications from signal processing, in this chapter we set out to study several new classes of discrete and continuous polynomial optimization models in the complex domain. The detailed descriptions of these models will be presented later. As a matter of fact, there are scattered results on complex polynomial optimization in the literature. When the objective function is quadratic, the MAX-3-CUT problem is a typical instance for the 3rd roots of unity constraint. Unity circle constrained complex optimization arises from the study of robust optimization as well as control theory [92, 38]. In particular, complex quadratic form optimization over unity constraints studied by Toker and Ozbay [92] are called complex programming. If the degree of complex polynomial is beyond quadratic, say quartic, several applications in signal processing can be found in the literature. Maricic et al. [4] proposed a quartic polynomial model for blind channel equalization in digital communication. Aittomäki

and Koivunen [9] discussed the problem of beam-pattern synthesis in array signal processing problem and formulated it to be a complex quartic minimization problem. Chen and Vaidyanathan [93] studied MIMO radar waveform optimization with prior information of the extended target and clutter, by relaxing a quartic complex model. Most recently, Aubry et al. [94] managed to design a radar waveform sharing an ambiguity function behavior by resorting to a complex optimization problem. In quantum entanglement, Hilling and Sudbery [95] formulated a typical problem as a complex form optimization problem under spherical constraint, which is one of the three classes of models studied in this chapter. Inspired by their work, Zhang and Qi [96] discussed the quantum eigenvalue problem, which arises from the geometric measure of entanglement of a multipartite symmetric pure state, in the complex tensor space. In fact, complex polynomial and complex tensor are interesting on their own. Eigenvalue and eigenvectors in the complex domain were already proposed and studied by Qi [97], whereas the name E-eigenvalue was coined. Moreover, in Chapter 10, we shall see the complex tensors and complex polynomials established in this chapter can be used the Radar waveform design.

Like its real-case counterpart, complex polynomial optimization is also NP-hard in general. Therefore, approximation algorithms for complex models are on high demand. However, in the literature approximation algorithms are mostly considered for quadratic models only. Ben-Tal et al. [38] first studied complex quadratic optimization whose objective function is restricted nonnegative by using complex matrix cube theorem. Zhang and Huang [39], So et al. [40] considered complex quadratic form maximization under the m-th roots of unity constraints and unity constraints. Later, Huang and Zhang [41] also considered bilinear form complex optimization models under similar constraints. For real valued polynomial optimization problems, Luo and Zhang [22] first considered approximation algorithms for quartic optimization. At the same time, Ling et al. [32] considered a special quartic optimization model. Basically, the problem is to maximize a biquadratic form over two spherical constraints. Significant progresses have recently been made by He et al. [33, 34, 35], where the authors derived a series of approximation methods for optimization of any fixed degree polynomial function under various constrains. So [36] further considered spherically constrained homogeneous polynomial optimization and proposed a deterministic algorithm with an improved approximation

ratio. For most recent development on approximation algorithms for homogeneous polynomial optimization, we refer the interested readers to [98, 99].

To the best of our knowledge, there is no result on approximation methods for general degree complex polynomial optimization as such, except for the practice of transforming a general high degree complex polynomial to the real case by doubling the problem dimension, and then resorting to the existing approximation algorithms for the real-valued polynomials [33, 34, 35, 36, 98, 99]. The latter approach, however, may lose the handle on the structure of the problem, hence misses nice properties of the complex polynomial functions. As a result, the computational costs may increase while the solution qualities may deteriorate. Exploiting the special structure of the complex model, it is often possible to get better approximation bounds, e.g, [39]. With this in mind, in this chapter we shall study the complex polynomial optimization in its direct form. Let us start with some preparations next.

Given a d-th order complex tensor $\mathcal{F} = (\mathcal{F}_{i_1 i_2 \cdots i_d}) \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_d}$, its associated multilinear form is defined as

$$L(x^1, x^2, \cdots, x^d) := \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_d=1}^{n_d} \mathcal{F}_{i_1 i_2 \cdots i_d} x^1_{i_1} x^2_{i_2} \cdots x^d_{i_d},$$

where the variables $x^k \in \mathbb{C}^{n_k}$ for $k = 1, 2, \ldots, d$, with 'L' standing for 'multilinearity'.

Closely related to multilinear form is homogeneous polynomial function, or, more explicitly

$$H(x) := \sum_{1 \le i_1 \le i_2 \le \cdots \le i_d \le n} a_{i_1 i_2 \cdots i_d} x_{i_1} x_{i_2} \cdots x_{i_d},$$

where the variable $x \in \mathbb{C}^n$, with 'H' standing for 'homogeneous polynomial'. As we mentioned in Chapter 1, associated with any homogeneous polynomial is a super-symmetric complex tensor $\mathcal{F} \in \mathbb{C}^{n^d}$. In this sense,

$$\mathcal{F}_{i_1 i_2 \cdots i_d} = \frac{a_{i_1 i_2 \cdots i_d}}{|\Pi(i_1 i_2 \cdots i_d)|} \quad \forall\, 1 \le i_1 \le i_2 \le \cdots \le i_d \le n,$$

where $\Pi(i_1 i_2 \cdots i_d)$ is the set of all distinct permutations of the indices $\{i_1, i_2, \cdots, i_d\}$.

In light of multilinear form L associated with a super-symmetric tensor, homogeneous polynomial H is obtained by letting $x^1 = x^2 = \cdots = x^d$; i.e., $H(x) = L(\underbrace{x, x, \cdots, x}_{d})$. Furthermore, He et al. [33] established an essential linkage between multilinear forms and homogeneous polynomials in the real domain.

Lemma 7.1.1. *Suppose $x^1, x^2, \cdots, x^d \in \mathbb{R}^n$, and $\xi_1, \xi_2, \cdots, \xi_d$ are i.i.d. symmetric Bernoulli random variables (taking 1 and -1 with equal probability). For any super-symmetric tensor $\mathcal{F} \in \mathbb{R}^{n^d}$ with its associated multilinear form L and homogeneous polynomial H, it holds that*

$$\mathsf{E}\left[\prod_{i=1}^{d} \xi_i H\left(\sum_{k=1}^{d} \xi_k x^k\right)\right] = d! L(x^1, x^2, \cdots, x^d).$$

With Lemma 7.1.1 in place, tensor relaxation [33] is proposed to solve homogeneous polynomial optimization problems, by relaxing the objective function to a multilinear form.

In terms of the optimization, the real part of the above functions (multilinear form and homogeneous polynomial) is usually considered. We introduced conjugate partial-symmetric complex tensors, which are extended from Hermitian matrices.

Definition 7.1.2. *An even order complex tensor $\mathcal{F} = (\mathcal{F}_{i_1 i_2 \cdots i_{2d}}) \in \mathbb{C}^{n^{2d}}$ is called conjugate partial-symmetric if*
(1) $\mathcal{F}_{i_1 \cdots i_d \, i_{d+1} \cdots i_{2d}} = \overline{\mathcal{F}_{i_{d+1} \cdots i_{2d} \, i_1 \cdots i_d}}$ and
(2) $\mathcal{F}_{i_1 \cdots i_d \, i_{d+1} \cdots i_{2d}} = \mathcal{F}_{j_1 \cdots j_d \, j_{d+1} \cdots j_{2d}} \quad \forall\, (j_1 \cdots j_d) \in \Pi(i_1 \cdots i_d),\ (j_{d+1} \cdots j_{2d}) \in \Pi(i_{d+1} \cdots i_{2d})$.

Associated with any conjugate partial-symmetric tensor, we shall show in Section 7.4 that the following conjugate form

$$C(\overline{x}, x) := L(\underbrace{\overline{x}, \cdots, \overline{x}}_{d}, \underbrace{x, \cdots, x}_{d}) = \sum_{1 \le i_1, \cdots, i_d, j_1, \cdots, j_d \le n} \mathcal{F}_{i_1 \cdots i_d j_1 \cdots j_d} \, \overline{x_{i_1} \cdots x_{i_d}} x_{j_1} \cdots x_{j_d}$$

always takes real value for any $x \in \mathbb{C}^n$. Besides, any conjugate form C uniquely determines a conjugate partial-symmetric complex tensor. In the above expression, 'C' signifies 'conjugate'.

The following commonly encountered constraint sets for complex polynomial optimization are considered in this chapter:

- The m-th roots of unity constraint: $\mathbf{\Omega}_m = \left\{1, \omega_m, \cdots, \omega_m^{m-1}\right\}$, where $\omega_m = e^{\boldsymbol{i}\frac{2\pi}{m}} = \cos\frac{2\pi}{m} + \boldsymbol{i}\sin\frac{2\pi}{m}$. Denote $\mathbf{\Omega}_m^n = \{x \in \mathbb{C}^n \,|\, x_i \in \mathbf{\Omega}_m,\ i = 1, 2, \cdots, n\}$.
- The unity constraint: $\mathbf{\Omega}_\infty = \{z \in \mathbb{C} \,|\, |z| = 1\}$. And we denote $\mathbf{\Omega}_\infty^n = \{x \in \mathbb{C}^n \,|\, x_i \in \mathbf{\Omega}_\infty,\ i = 1, 2, \cdots, n\}$.

- The complex spherical constraint: $\mathbf{CS}^n = \{x \in \mathbb{C}^n \mid \|x\| = 1\}$.

Throughout this chapter we assume $m \geq 3$, to ensure that the decision variables being considered are essentially complex.

In this chapter, we shall discuss various complex polynomial optimization models. The objective function will be one of the three afore-mentioned complex polynomial functions (L, H, and C), or their real parts whenever is applicable; the constraint set is one of the three kinds as discussed above. The organization of the chapter is as follows. Maximizing multilinear form over three types of constraint sets will be discussed in Section 7.2, i.e., models (L_m), (L_∞) and (L_S), with the subscription indicating the constraint for: the m-th roots of unity, the unity, and the complex sphere, respectively. Section 7.3 deals with maximization of homogeneous polynomial over three types of constraints, i.e., models $(H_m), (H_\infty)$ and (H_S). To study the conjugate form optimization, in Section 7.4, we present the necessary and sufficient conditions for a complex conjugate polynomial function to always take real values. Finally, Section 7.5 discusses maximization of conjugate form over three types of constraints, i.e., models $(C_m), (C_\infty)$ and (C_S).

As a matter of notation, for any maximization problem $(P) : \max_{x\in X} p(x)$, we denote $v(P)$ to be the optimal value, and $\underline{v}(P)$ to be the optimal value of its minimization counterpart $(\min_{x\in X} p(x))$.

Definition 7.1.3. *(1) A maximization problem $(P) : \max_{x\in X} p(x)$ admits a polynomial-time approximation algorithm with approximation ratio $\tau \in (0, 1]$, if $v(P) \geq 0$ and a feasible solution $\hat{x} \in X$ can be found in polynomial-time, such that $p(\hat{x}) \geq \tau v(P)$.*
(2) A maximization problem $(P) : \max_{x\in X} p(x)$ admits a polynomial-time approximation algorithm with relative *approximation ratio $\tau \in (0, 1]$, if a feasible solution $\hat{x} \in X$ can be found in polynomial-time, such that $p(\hat{x}) - \underline{v}(P) \geq \tau\,(v(P) - \underline{v}(P))$.*

In this chapter, we reserve τ to denote the approximation ratio. All the optimization models considered in this chapter are NP-hard in general, even restricting the domain to be real. We shall propose polynomial-time approximation algorithms with worse-case performance ratios for the models concerned, when the degree of these polynomial functions, d or $2d$, is fixed. These approximation ratios are depended only on the dimensions of the problems, or data-independent. We shall start off by presenting

Table 7.1 which summarizes the approximation results and the organization of the chapter.

Section	Model	Theorem	Approximation performance ratio
7.2.1	(L_m)	7.2.4	$\tau_m^{d-2}\,(2\tau_m-1)\Big(\prod_{k=1}^{d-2} n_k\Big)^{-\frac{1}{2}}$ where $\tau_m = \frac{m^2}{4\pi}\sin^2\frac{\pi}{m}$
7.2.2	(L_∞)	7.2.6	$0.7118\,\big(\frac{\pi}{4}\big)^{d-2}\,\Big(\prod_{k=1}^{d-2} n_k\Big)^{-\frac{1}{2}}$
7.2.3	(L_S)	7.2.7	$\Big(\prod_{k=1}^{d-2} n_k\Big)^{-\frac{1}{2}}$
7.3.1	(H_m)	7.3.3, 7.3.4	$\tau_m^{d-2}\,(2\tau_m-1)\,d!d^{-d}n^{-\frac{d-2}{2}}$
7.3.2	(H_∞)	7.3.5	$0.7118(\frac{\pi}{4})^{d-2}d!d^{-d}n^{-\frac{d-2}{2}}$
7.3.3	(H_S)	7.3.6	$d!d^{-d}n^{-\frac{d-2}{2}}$
7.5.1	(C_m)	7.5.3, 7.5.4	$\tau_m^{2d-2}(2\tau_m-1)(d!)^2(2d)^{-2d}n^{-(d-1)}$
7.5.2	(C_∞)	7.5.5	$0.7118\,\big(\frac{\pi}{4}\big)^{2d-2}\,(d!)^2(2d)^{-2d}n^{-(d-1)}$
7.5.2	(C_S)	7.5.6	$(d!)^2(2d)^{-2d}n^{-(d-1)}$

Table 7.1: Organization of the chapter and the approximation results

7.2 Complex Multilinear Form Optimization

Let us consider optimization of complex multilinear forms, under three types of constraints described in Section 7.1. Specifically, the models under consideration are:

$$\begin{array}{lll}
(L_m) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\
 & \text{s.t.} & x^k \in \mathbf{\Omega}_m^{n_k},\ k = 1, 2, \ldots, d; \\
(L_\infty) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\
 & \text{s.t.} & x^k \in \mathbf{\Omega}_\infty^{n_k},\ k = 1, 2, \ldots, d; \\
(L_S) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\
 & \text{s.t.} & x^k \in \mathbf{CS}^{n_k},\ k = 1, 2, \ldots, d.
\end{array}$$

Associated with multilinear form objective is a d-th order complex tensor $\mathcal{F} \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_d}$. Without loss of generality, we assume that $n_1 \le n_2 \le \cdots \le n_d$ and $\mathcal{F} \neq 0$. The multilinear form optimization models are interesting on their own. For example, typical

optimization problem in quantum entanglement problem [95] is in the formulation of (L_S).

7.2.1 Multilinear Form in the m-th Roots of Unity

When $d = 2$, (L_m) is already NP-hard, even for $m = 2$. In this case, (L_m) is to compute $\infty \mapsto 1$-norm of a matrix, and the best approximation bound is $\frac{2\ln(1+\sqrt{2})}{\pi} \approx 0.56$ due to Alon and Naor [69]. Huang and Zhang [41] studied general m when $d = 2$, and proposed polynomial-time randomized approximation algorithm with constant worst-case performance ratio. Specifically the ratio is $\frac{m^2}{4\pi}(1-\cos\frac{2\pi}{m})-1 = 2\tau_m - 1$ for $m \geq 3$, where $\tau_m := \frac{m^2}{8\pi}(1-\cos\frac{2\pi}{m}) = \frac{m^2}{4\pi}\sin^2\frac{\pi}{m}$ throughout this chapter.

To proceed to the general degree d, let us start with the case $d = 3$.

$$\begin{array}{lll} (L_m^3) & \max & \mathrm{Re}\ L(x,y,z) \\ & \text{s.t.} & x \in \mathbf{\Omega}_m^{n_1},\ y \in \mathbf{\Omega}_m^{n_2},\ z \in \mathbf{\Omega}_m^{n_3}. \end{array}$$

Denote $W = xy^\top$. It is easy to observe that $W_{ij} = x_i y_j \in \mathbf{\Omega}_m$ for all (i,j), implying $W \in \mathbf{\Omega}_m^{n_1 \times n_2}$. The above problem can be relaxed to

$$\begin{array}{lll} (L_m^2) & \max & \mathrm{Re}\ \hat{L}(W,z) := \mathrm{Re}\ \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \mathcal{F}_{ijk} W_{ij} z_k \\ & \text{s.t.} & W \in \mathbf{\Omega}_m^{n_1 \times n_2},\ z \in \mathbf{\Omega}_m^{n_3}. \end{array}$$

This is exactly (L_m) with $d = 2$, which admits a polynomial-time approximation algorithm with approximation ratio $2\tau_m - 1$ in [41]. Denote the approximate solution of (L_m^2) to be $(\hat{W}, \hat{z})$, i.e.,

$$\mathrm{Re}\ \hat{L}(\hat{W}, \hat{z}) \geq (2\tau_m - 1) v(L_m^2) \geq (2\tau_m - 1) v(L_m^3). \tag{7.1}$$

The key step is to recover (x, y) from $\hat{W}$. For this purpose, we introduce the following decomposition routine (DR).

DR (Decomposition Routine) 7.2.1.

- *Input:* $\hat{W} \in \mathbf{\Omega}_m^{n_1 \times n_2}$.
- *Construct*

$$\tilde{W} = \begin{bmatrix} I & \hat{W}/\sqrt{n_1} \\ \hat{W}^\dagger/\sqrt{n_1} & \hat{W}^\dagger \hat{W}/n_1 \end{bmatrix} \succeq 0 \ \textit{(Hermitian positive semidefinite).}$$

- *Randomly generate*
$$\begin{pmatrix}\xi\\ \eta\end{pmatrix} \sim \mathcal{N}(0, \tilde{W}).$$
- *For* $i = 1, 2, \ldots, n_1$, *let*
$$\hat{x}_i := \omega_m^{\ell} \text{ if } \arg\xi_i \in \left[\frac{\ell}{m}2\pi, \frac{\ell+1}{m}2\pi\right) \text{ for some } \ell \in \mathbb{Z};$$
and for $j = 1, 2, \ldots, n_2$, *let*
$$\hat{y}_j := \omega_m^{-\ell} \text{ if } \arg\eta_j \in \left[\frac{\ell}{m}2\pi, \frac{\ell+1}{m}2\pi\right) \text{ for some } \ell \in \mathbb{Z}.$$
- *Output:* $(\hat{x}, \hat{y}) \in \mathbf{\Omega}_m^{n_1+n_2}$.

It was shown in [39] that

$$\mathsf{E}[\hat{x}_i\hat{y}_j] = \frac{m(2-\omega_m-\omega_m^{-1})}{8\pi^2}\sum_{\ell=0}^{m-1}\omega_m^{\ell}\left(\arccos\left(-\text{Re }\omega_m^{-\ell}\tilde{W}_{i,n_1+j}\right)\right)^2. \tag{7.2}$$

There are some useful properties regarding (7.2) as shown below; the proofs can be found in the appendix.

Lemma 7.2.2. *Define* $F_m : \mathbb{C} \mapsto \mathbb{C}$ *with*

$$F_m(x) := \frac{m(2-\omega_m-\omega_m^{-1})}{8\pi^2}\sum_{\ell=0}^{m-1}\omega_m^{\ell}\left(\arccos\left(-\text{Re }\omega_m^{-\ell}x\right)\right)^2.$$

(1) If $a \in \mathbb{C}$ *and* $b \in \mathbf{\Omega}_m$, *then* $F_m(ab) = bF_m(a)$.
(2) If $a \in \mathbb{R}$, *then* $F_m(a) \in \mathbb{R}$.

As $(\hat{W}, \hat{z})$ is a feasible solution of (L_m^2), $\hat{W}_{ij} \in \mathbf{\Omega}_m$. By Lemma 7.2.2, we have for all (i, j)

$$\mathsf{E}[\hat{x}_i\hat{y}_j] = F_m(\tilde{W}_{i,n_1+j}) = F_m(\hat{W}_{ij}/\sqrt{n_1}) = \hat{W}_{ij}F_m(1/\sqrt{n_1}) \text{ and } F_m(1/\sqrt{n_1}) \in \mathbb{R}. \tag{7.3}$$

We are now able to evaluate the objective value of $(\hat{x}, \hat{y}, \hat{z})$:

$$\begin{aligned}
\mathsf{E}\left[\text{Re } L(\hat{x}, \hat{y}, \hat{z})\right] &= \mathsf{E}\left[\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \text{Re } \mathcal{F}_{ijk}\, \hat{x}_i \hat{y}_j \hat{z}_k\right] \\
&= \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \text{Re } \mathcal{F}_{ijk}\, \mathsf{E}\left[\hat{x}_i \hat{y}_j\right] \hat{z}_k \\
&= \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \text{Re } \mathcal{F}_{ijk}\, \hat{W}_{ij} F_m(1/\sqrt{n_1}) \hat{z}_k \\
&= F_m\left(1/\sqrt{n_1}\right) \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \text{Re } \mathcal{F}_{ijk}\, \hat{W}_{ij} \hat{z}_k \\
&= F_m\left(1/\sqrt{n_1}\right) \text{Re } \hat{L}(\hat{W}, \hat{z}).
\end{aligned}$$

Furthermore, according to the appendix of [39], we have

$$F_m\left(1/\sqrt{n_1}\right) \geq \frac{m^2(1-\cos\frac{2\pi}{m})}{8\pi\sqrt{n_1}} = \frac{\tau_m}{\sqrt{n_1}}. \tag{7.4}$$

Combined with (7.1), we finally get

$$\mathsf{E}\left[\text{Re } L(\hat{x}, \hat{y}, \hat{z})\right] = F_m\left(1/\sqrt{n_1}\right) \text{Re } \hat{L}(\hat{W}, \hat{z}) \geq \frac{\tau_m}{\sqrt{n_1}}(2\tau_m - 1) v(L_m^3).$$

Theorem 7.2.3. *When $d = 3$, (L_m) admits a polynomial-time randomized approximation algorithm with approximation ratio $\frac{\tau_m(2\tau_m-1)}{\sqrt{n_1}}$.*

By a similar method and using induction, the above discussion is readily extended to any fixed degree d.

Theorem 7.2.4. *(L_m) admits a polynomial-time randomized approximation algorithm with approximation ratio $\tau(L_m) := \tau_m^{d-2}\,(2\tau_m - 1)\Big(\prod_{k=1}^{d-2} n_k\Big)^{-\frac{1}{2}}$, i.e., a feasible solution $(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)$ can be found in polynomial-time, such that*

$$\mathsf{E}\left[\text{Re } L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^{d-1})\right] \geq \tau(L_m) v(L_m).$$

Proof. The proof is based on induction on the degree d. The case for $d = 2$ or $d = 3$ is known to be true. The inductive step can be similarly derived from Theorem 7.2.3.

For general d, denote $W = x^1(x^d)^\top$ and (L_m) is then relaxed to

$$\begin{array}{lll} (L_m^{d-1}) & \max & \mathrm{Re}\ \hat{L}(W, x^2, \cdots, x^{d-1}) := \mathrm{Re}\ \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} \mathcal{F}_{i_1 i_2 \cdots i_d} W_{i_1 i_d} x_{i_2}^2 \cdots x_{i_{d-1}}^{d-1} \\ & \text{s.t.} & W \in \mathbf{\Omega}_m^{n_1 \times n_d},\ x^k \in \mathbf{\Omega}_m^{n_k},\ k = 2, 3, \cdots, d-1. \end{array}$$

By induction we are able to find $(\hat{W}, \hat{x}^2, \cdots, \hat{x}^{d-1})$, such that

$$\begin{aligned} \mathsf{E}\left[\mathrm{Re}\ \hat{L}(\hat{W}, \hat{x}^2, \cdots, \hat{x}^{d-1})\right] &\geq \tau_m^{d-3}\,(2\tau_m - 1)\left(\prod_{k=2}^{d-2} n_k\right)^{-\frac{1}{2}} v(L_m^{d-1}) \\ &\geq \tau_m^{d-3}\,(2\tau_m - 1)\left(\prod_{k=2}^{d-2} n_k\right)^{-\frac{1}{2}} v(L_m). \end{aligned}$$

Applying DR 7.2.1 with input $\hat{W}$ and output $(\hat{x}^1, \hat{x}^d)$, and using (7.3) and (7.4), we conclude that

$$\begin{aligned} \mathsf{E}\left[\mathrm{Re}\ L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)\right] &= \mathsf{E}\left[\mathrm{Re}\ \hat{L}\left(\hat{x}^1(\hat{x}^d)^\top, \hat{x}^2, \cdots, \hat{x}^{d-1}\right)\right] \\ &= \mathsf{E}\left[\mathrm{Re}\ \hat{L}\left(\mathsf{E}\left[\hat{x}^1(\hat{x}^d)^\top | \hat{W}\right], \hat{x}^2, \cdots, \hat{x}^{d-1}\right)\right] \\ &= \mathsf{E}\left[\mathrm{Re}\ \hat{L}\left(\hat{W} F_m\left(1/\sqrt{n_1}\right), \hat{x}^2, \cdots, \hat{x}^{d-1}\right)\right] \\ &= F_m\left(1/\sqrt{n_1}\right) \mathsf{E}\left[\mathrm{Re}\ \hat{L}(\hat{W}, \hat{x}^2, \cdots, \hat{x}^{d-1})\right] \\ &\geq \frac{\tau_m}{\sqrt{n_1}} \cdot \tau_m^{d-3}\,(2\tau_m - 1)\left(\prod_{k=2}^{d-2} n_k\right)^{-\frac{1}{2}} v(L_m) \\ &= \tau(L_m) v(L_m). \end{aligned}$$

□

7.2.2 Multilinear Form with Unity Constraints

Let us now turn to the optimization model with unity constraint (L_∞), which can be taken as the model (L_m) when $m \to \infty$:

$$\begin{array}{lll} (L_\infty) & \max & \mathrm{Re}\ L(x^1, x^2, \cdots, x^d) \\ & \text{s.t.} & x^k \in \mathbf{\Omega}_\infty^{n_k},\ k = 1, 2, \ldots, d. \end{array}$$

When $d = 2$, (L_∞) was studied in [41] and a polynomial-time approximation algorithm with approximation ratio 0.7118 was presented. To treat the high degree case,

one may again apply induction in the proof of Theorem 7.2.4. However, DR 7.2.1 should be slightly modified in order to apply the decomposition procedure for $\mathbf{\Omega}_\infty$.

DR (Decomposition Routine) 7.2.5.

- *Input:* $\hat{W} \in \mathbf{\Omega}_\infty^{n_1 \times n_2}$.
- *Construct* $\tilde{W} = \begin{bmatrix} I & \hat{W}/\sqrt{n_1} \\ \hat{W}^\dagger/\sqrt{n_1} & \hat{W}^\dagger \hat{W}/n_1 \end{bmatrix} \succeq 0.$
- *Randomly generate* $\binom{\xi}{\eta} \sim \mathcal{N}(0, \tilde{W})$.
- *Let* $\hat{x}_i = e^{\boldsymbol{i} \arg \xi_i}$ *for* $i = 1, 2, \ldots, n_1$, *and let* $\hat{y}_j = e^{-\boldsymbol{i} \arg \eta_j}$ *for* $j = 1, 2, \ldots, n_2$.
- *Output:* $(\hat{x}, \hat{y}) \in \mathbf{\Omega}_\infty^{n_1+n_2}$.

The estimation of $(\hat{x}, \hat{y})$ is then

$$\mathsf{E}[\hat{x}_i \hat{y}_j] = F_\infty(\tilde{W}_{i,n_1+j}) = F_\infty(\hat{W}_{ij}/\sqrt{n_1}) \quad \forall\, (i,j).$$

It was calculated in [39] that

$$F_\infty(a) := \lim_{m \to \infty} F_m(a) = \frac{\pi}{4} a + \frac{\pi}{2} \sum_{k=1}^{\infty} \frac{((2k)!)^2}{2^{4k+1}(k!)^4(k+1)} |a|^{2k} a.$$

Similar as in Lemma 7.2.2:

$$\begin{aligned} &F_\infty(ab) = b F_\infty(a) \quad \forall\, a \in \mathbb{C},\, b \in \mathbf{\Omega}_\infty, \\ &F_\infty(a) \in \mathbb{R} \quad \forall\, a \in \mathbb{R}, \\ &F_\infty(a) \geq \frac{\pi}{4} a \quad \forall\, a > 0. \end{aligned}$$

By applying the result in [41] for case $d = 2$ and using a similar argument as Theorem 7.2.4, we have the following main result of this subsection.

Theorem 7.2.6. (L_∞) *admits a polynomial-time randomized approximation algorithm with approximation ratio* $\tau(L_\infty) := 0.7118 \left(\frac{\pi}{4}\right)^{d-2} \left(\prod_{k=1}^{d-2} n_k\right)^{-\frac{1}{2}}$.

7.2.3 Multilinear Form with Spherical Constraints

Let us turn to our last model for multilinear form optimization:

$$\begin{array}{lll} (L_S) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\ & \text{s.t.} & x^k \in \mathbf{CS}^{n_k},\ k = 1, 2, \ldots, d. \end{array}$$

Model (L_S) is also known as computing the largest singular value (the real part) of a d-th order complex tensor $\mathcal{F}$. The case when $\mathcal{F}$ is real was widely studied [33, 36, 79, 59]. In particular, He et al. [33] introduced the recursive procedure and eigen-decomposition based approximation algorithm with approximation ratio $\left(\prod_{k=1}^{d-2} n_k\right)^{-\frac{1}{2}}$. Using a similar argument, we have the following result.

Theorem 7.2.7. *(L_S) admits a deterministic polynomial-time approximation algorithm with approximation ratio $\tau(L_S) := \left(\prod_{k=1}^{d-2} n_k\right)^{-\frac{1}{2}}$.*

When $d = 2$, (L_S) is to compute the largest singular value of a complex matrix, and is therefore solvable in polynomial-time, which also follows as a consequence of Theorem 7.2.7. The proof of Theorem 7.2.7 is similar to that of [33] for the real case. The main ingredients include establishing the initial step for the case $d = 2$, and then establishing a decomposition routine, which is shown as follows, to enable the induction.

DR (Decomposition Routine) 7.2.8.

- *Input:* $\hat{W} \in \mathbb{C}^{n_1 \times n_2}$.
- *Find the left singular vector* $\hat{x} \in \mathbf{CS}^{n_1}$ *and the right singular vector* $\hat{y} \in \mathbf{CS}^{n_2}$ *corresponding to the largest singular value of* $\hat{W}$.
- *Output:* $\hat{x} \in \mathbf{CS}^{n_1}, \hat{y} \in \mathbf{CS}^{n_2}$.

Remark that if we directly apply the result for the real case in [33] by treating tensor $\mathcal{F} \in \mathbb{R}^{2n_1 \times 2n_2 \times \cdots \times 2n_d}$, then the approximation ratio will be $\left(\prod_{k=1}^{d-2} 2n_k\right)^{-\frac{1}{2}}$, which is certainly worse than $\tau(L_S)$.

7.3 Complex Homogeneous Polynomial Optimization

This section is concerned with the optimization of complex homogeneous polynomial $H(x)$, associated with super-symmetric complex tensor $\mathcal{F} \in \mathbb{C}^{n^d}$. Specifically, the models under considerations are:

$$\begin{array}{lll} (H_m) & \max & \text{Re } H(x) \\ & \text{s.t.} & x \in \mathbf{\Omega}_m^n; \\ (H_\infty) & \max & \text{Re } H(x) \\ & \text{s.t.} & x \in \mathbf{\Omega}_\infty^n; \\ (H_S) & \max & \text{Re } H(x) \\ & \text{s.t.} & x \in \mathbf{CS}^n. \end{array}$$

Denote L to be multilinear form associated with $\mathcal{F}$, and then $H(x) = L(\underbrace{x, x, \cdots, x}_{d})$. By applying the tensor relaxation method established in [33], the above models are then relaxed to the following multilinear form optimization models discussed in Section 7.2:

$$\begin{array}{lll} (LH_m) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\ & \text{s.t.} & x^k \in \mathbf{\Omega}_m^n,\ k = 1, 2, \ldots, d; \\ (LH_\infty) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\ & \text{s.t.} & x^k \in \mathbf{\Omega}_\infty^n,\ k = 1, 2, \ldots, d; \\ (LH_S) & \max & \text{Re } L(x^1, x^2, \cdots, x^d) \\ & \text{s.t.} & x^k \in \mathbf{CS}^n,\ k = 1, 2, \ldots, d. \end{array}$$

The approximation results in Section 7.2 can return good approximation solutions for these relaxed models. The key next step is to obtain good solutions for the original homogeneous polynomial optimizations. Similar to Lemma 7.1.1, we establish a linkage between functions L and H in the complex domain. The proof of Lemma 7.3.1 can be found in the appendix.

Lemma 7.3.1. *Let $m \in \{3, 4, \cdots, \infty\}$. Suppose $x^1, x^2, \cdots, x^d \in \mathbb{C}^n$, and $\mathcal{F} \in \mathbb{C}^{n^d}$ is a super-symmetric complex tensor with its associated multilinear form L and homogeneous polynomial H. If $\xi_1, \xi_2, \cdots, \xi_d$ are i.i.d. uniform distribution on $\mathbf{\Omega}_m$, then*

$$\mathsf{E}\left[\prod_{i=1}^{d} \overline{\xi_i} H\left(\sum_{k=1}^{d} \xi_k x^k\right)\right] = d! L(x^1, x^2 \cdots, x^d) \text{ and } \mathsf{E}\left[\prod_{i=1}^{d} \xi_i H\left(\sum_{k=1}^{d} \xi_k x^k\right)\right] = 0.$$

7.3.1 Homogeneous Polynomial in the m-th Roots of Unity

Let us now focus on the model (H_m) : $\max_{x\in\mathbf{\Omega}_m^n} \mathrm{Re}\ H(x)$. By Lemma 7.3.1, for any fixed $\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d \in \mathbb{C}^n$, we can find $\beta_1, \beta_2, \cdots, \beta_d \in \mathbf{\Omega}_m$ in polynomial-time, such that

$$\mathrm{Re}\ \prod_{i=1}^{d} \overline{\beta_i} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) \geq \mathrm{Re}\ d^{-d} d! L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d). \tag{7.5}$$

For any $1 \leq i \leq n$, if $\hat{x}_i^k \in \mathbf{\Omega}_m$ for all $1 \leq k \leq d$, then $\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}_i^k \in \mathrm{conv}(\mathbf{\Omega}_m)$. As shown below, we are able to get a solution from $\mathrm{conv}(\mathbf{\Omega}_m)$ to one of its vertices $(\mathbf{\Omega}_m)$.

Lemma 7.3.2. *Suppose $m \in \{3, 4, \cdots, \infty\}$, and $x \in \mathbb{C}^n$ with $x_i \in \mathrm{conv}(\mathbf{\Omega}_m)$ for all $1 \leq i \leq n$.*
(1) If $H(x)$ is a complex homogeneous polynomial associated with square-free (meaning that its entry is zero whenever two of its indices are identical) super-symmetric tensor $\mathcal{F} \in \mathbb{C}^{n^d}$, then $y, z \in \mathbf{\Omega}_m^n$ can be found in polynomial-time, such that $\mathrm{Re}\ H(y) \leq \mathrm{Re}\ H(x) \leq \mathrm{Re}\ H(z)$.
(2) If $\mathrm{Re}\ H(x)$ *is convex, then $z \in \mathbf{\Omega}_m^n$ can be found in polynomial-time, such that* $\mathrm{Re}\ H(x) \leq \mathrm{Re}\ H(z)$.

Proof. If $H(x)$ is square-free, by fixing $x_2, x_3, \cdots, x_n$ as constants and taking x_1 as the only decision variable, we may write

$$\mathrm{Re}\ H(x) = \mathrm{Re}\ h_1(x_2, x_3, \cdots, x_n) + \mathrm{Re}\ x_1 h_2(x_2, x_3, \cdots, x_n) =: \mathrm{Re}\ h(x_1).$$

Since $\mathrm{Re}\ h(x_1)$ is a linear function of x_1, its optimal value over $\mathrm{conv}(\mathbf{\Omega}_m)$ is attained at one of its vertices. For instance, $z_1 \in \mathbf{\Omega}_m$ can be found easily such that $\mathrm{Re}\ h(z_1) \geq \mathrm{Re}\ h(x_1)$. Now, repeat the same procedures for $x_2, x_3, \cdots, x_n$, and let them be replaced by $z_2, z_3, \cdots, z_n$ respectively. Then $z \in \mathbf{\Omega}_m^n$ satisfies $\mathrm{Re}\ H(z) \geq \mathrm{Re}\ H(x)$. Using the same argument, we may find $y \in \mathbf{\Omega}_m^n$, such that $\mathrm{Re}\ H(y) \leq \mathrm{Re}\ H(x)$. The case that $\mathrm{Re}\ H(x)$ is convex can be proven similarly. □

Now we are ready to prove the main results in this subsection.

Theorem 7.3.3. *Suppose $H(x)$ is square-free or* $\mathrm{Re}\ H(x)$ *is convex.*
(1) If $m \mid (d-1)$, then (H_m) admits a polynomial-time randomized approximation

algorithm with approximation ratio $\tau(H_m) := \tau_m^{d-2}\,(2\tau_m - 1)\,d!d^{-d}n^{-\frac{d-2}{2}}$.
(2) If $m \nmid 2d$, *then* (H_m) *admits a polynomial-time randomized approximation algorithm with approximation ratio* $\frac{1}{2}\tau(H_m)$.

Proof. Relaxing (H_m) to (LH_m), we find a feasible solution $(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)$ of (LH_m) in polynomial-time with approximation ratio $\tau_m^{d-2}\,(2\tau_m - 1)\,n^{-\frac{d-2}{2}}$ by Theorem 7.2.4. Then by (7.5), we further find $\beta \in \mathbf{\Omega}_m^d$, such that

$$\text{Re}\ \prod_{i=1}^{d}\overline{\beta_i}H\left(\frac{1}{d}\sum_{k=1}^{d}\beta_k\hat{x}^k\right) \geq \text{Re}\ d!d^{-d}L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) \geq \tau(H_m)v(LH_m) \geq \tau(H_m)v(H_m).$$

Let us denote $\hat{x} := \frac{1}{d}\sum_{k=1}^{d}\beta_k\hat{x}^k$. Clearly we have $\hat{x}_i \in \text{conv}(\mathbf{\Omega}_m)$ for $i = 1, 2, \ldots, n$.

(1) If $m \mid (d-1)$, then $d = 1 + mp$ for some $p \in \mathbb{Z}$. As $\beta_i \in \mathbf{\Omega}_m$, we have

$$H\left(\hat{x}\prod_{i=1}^{d}\overline{\beta_i}\right) = \left(\prod_{i=1}^{d}\overline{\beta_i}\right)^d H(\hat{x}) = \prod_{i=1}^{d}\overline{\beta_i}^{1+mp}H(\hat{x}) = \prod_{i=1}^{d}\overline{\beta_i}H(\hat{x}).$$

Since $\hat{x}_j\prod_{i=1}^{d}\overline{\beta_i} \in \text{conv}(\mathbf{\Omega}_m)$ for $j = 1, 2, \ldots, n$, noticing $H(x)$ is square-free or Re $H(x)$ is convex, and applying Lemma 7.3.2, we are able to find $y \in \mathbf{\Omega}_m^n$ in polynomial-time, such that

$$\text{Re}\ H(y) \geq \text{Re}\ H\left(\hat{x}\prod_{i=1}^{d}\overline{\beta_i}\right) = \text{Re}\ \prod_{i=1}^{d}\overline{\beta_i}H(\hat{x}) \geq \tau(H_m)v(H_m).$$

(2) Let $\Phi = \left\{H(\omega_m^\ell\hat{x})\,|\,\ell = 0, 1, \ldots, m-1\right\}$. As $H(\omega_m^\ell\hat{x}) = \omega_m^{d\ell}H(\hat{x})$ for $\ell = 0, 1, \ldots, m-1$, the elements of Φ is evenly distributed on the unity circle with radius $|H(\hat{x})|$ in the complex plane. Since $\omega_m^{d\ell} = e^{\boldsymbol{i}\frac{2d\ell\pi}{m}}$ and $m \nmid 2d$, it is easy to verify that $|\Phi| \geq 3$. Let ϕ be the minimum angle between Φ and the real axis, or equivalently $|H(\hat{x})|\cos\phi = \max_{x\in\Phi}\text{Re}\ x$. Clearly $0 \leq \phi \leq \frac{\pi}{3}$ by $|\Phi| \geq 3$. Let $H(\omega_m^t\hat{x}) = \arg\max_{x\in\Phi}\text{Re}\ x$. As $\omega_m^t\hat{x}_j \in \text{conv}(\mathbf{\Omega}_m)$ for $j = 1, 2, \ldots, n$, again by Lemma 7.3.2, we are able to find $y \in \mathbf{\Omega}_m^n$ in polynomial-time, such that

$$\text{Re}\ H(y) \geq \text{Re}\ H(\omega_m^t\hat{x}) = |H(\hat{x})|\cos\phi \geq \frac{1}{2}|H(\hat{x})| \geq \frac{1}{2}\text{Re}\ \prod_{i=1}^{d}\overline{\beta_i}H(\hat{x}) \geq \frac{1}{2}\tau(H_m)v(H_m).$$

□

Remark that condition (1) in Theorem 7.3.3 is a special case of (2); however in that special case a better approximation ratio than (2) is obtained. When $d \geq 4$ is even, almost all of the optimization models of homogeneous polynomials in the real domain (e.g., [33, 35, 36, 59]) only admit *relative* approximation ratios. Interestingly, in the complex domain, as Theorem 7.3.3 suggests, absolute approximation ratios are possible for some m when d is even.

When $m \mid 2d$, the approach in (2) of Theorem 7.3.3 may not work, since $|\Phi| \leq 2$. The worst case performance of the approximate solution cannot be guaranteed any more. However a relative approximation bound is possible for any m, as long as $H(x)$ is square-free.

Theorem 7.3.4. *If $H(x)$ is square-free, then (H_m) admits a polynomial-time randomized approximation algorithm with relative approximation ratio $\frac{1}{4}\tau(H_m)$.*

Proof. Relaxing (H_m) to (LH_m), we may find a feasible solution $(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)$ of (LH_m) in polynomial-time with approximation ratio $\tau_m^{d-2}\,(2\tau_m - 1)\,n^{-\frac{d-2}{2}}$ by Theorem 7.2.4, such that

$$\begin{aligned} d!d^{-d}\text{Re } L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) &\geq d!d^{-d}\tau_m^{d-2}\,(2\tau_m - 1)\,n^{-\frac{d-2}{2}}\,v(LH_m) \\ &= \tau(H_m)v(LH_m) \geq \tau(H_m)v(H_m). \end{aligned}$$

Let $\xi_1, \xi_2, \cdots, \xi_d$ be i.i.d. uniform distribution on $\mathbf{\Omega}_m$, and we have $\frac{1}{d}\sum_{k=1}^{d} \xi_k \hat{x}_i^k \in \text{conv}(\mathbf{\Omega}_m)$ for $i = 1, 2, \ldots, n$. As $H(x)$ is square-free, by Lemma 7.3.2, there exists $y \in \mathbf{\Omega}_m^n$, such that

$$\text{Re } H\left(\frac{1}{d}\sum_{k=1}^{d} \xi_k \hat{x}^k\right) \geq \text{Re } H(y) \geq \underline{v}(H_m). \tag{7.6}$$

According to Lemma 7.3.1, it follows that

$$\mathsf{E}\left[\text{Re } \prod_{i=1}^{d} \overline{\xi_i} H\left(\sum_{k=1}^{d} \xi_k \hat{x}^k\right)\right] = \text{Re } d!L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) \text{ and } \mathsf{E}\left[\text{Re } \prod_{i=1}^{d} \xi_i H\left(\sum_{k=1}^{d} \xi_k \hat{x}^k\right)\right] = 0.$$

Combining the above two identities leads to

$$\begin{aligned}
\text{Re } d!d^{-d}L(\hat{x}^1,\hat{x}^2,\cdots,\hat{x}^d) &= \mathsf{E}\left[\text{Re }\prod_{i=1}^{d}\overline{\xi_i}H\left(\frac{1}{d}\sum_{k=1}^{m}\xi_k\hat{x}^k\right)\right]+\mathsf{E}\left[\text{Re }\prod_{i=1}^{d}\xi_i H\left(\frac{1}{d}\sum_{k=1}^{m}\xi_k\hat{x}^k\right)\right]\\
&= \mathsf{E}\left[\text{Re }\left(\prod_{i=1}^{d}\overline{\xi_i}+\prod_{i=1}^{d}\xi_i\right)H\left(\frac{1}{d}\sum_{k=1}^{d}\xi_k\hat{x}^k\right)\right]\\
&= \mathsf{E}\left[\left(\prod_{i=1}^{d}\overline{\xi_i}+\prod_{i=1}^{d}\xi_i\right)\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\xi_k\hat{x}^k\right)\right]\\
&= \mathsf{E}\left[\left(\prod_{i=1}^{d}\overline{\xi_i}+\prod_{i=1}^{d}\xi_i\right)\left(\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\xi_k\hat{x}^k\right)-\underline{v}(H_m)\right)\right]\\
&\le \mathsf{E}\left[\left|\prod_{i=1}^{d}\overline{\xi_i}+\prod_{i=1}^{d}\xi_i\right|\cdot\left|\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\xi_k\hat{x}^k\right)-\underline{v}(H_m)\right|\right]\\
&\le 2\,\mathsf{E}\left[\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\xi_k\hat{x}^k\right)-\underline{v}(H_m)\right],
\end{aligned}$$

where the last step is due to (7.6). By randomizing, we are able to find $\beta\in\mathbf{\Omega}_m^d$, such that

$$\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\beta_k\hat{x}^k\right)-\underline{v}(H_m)\ge\frac{1}{2}\text{Re }d!d^{-d}L(\hat{x}^1,\hat{x}^2,\cdots,\hat{x}^d)\ge\frac{1}{2}\tau(H_m)v(H_m).$$

Let us now separately discuss two cases. In the first case, if $v(H_m)\ge\frac{1}{2}\left(v(H_m)-\underline{v}(H_m)\right)$, then the above further leads to

$$\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\beta_k\hat{x}^k\right)-\underline{v}(H_m)\ge\frac{1}{2}\tau(H_m)v(H_m)\ge\frac{1}{4}\tau(H_m)\left(v(H_m)-\underline{v}(H_m)\right).$$

Otherwise, we have $v(H_m)\le\frac{1}{2}\left(v(H_m)-\underline{v}(H_m)\right)$, which implies

$$-\underline{v}(H_m)\ge\frac{1}{2}\left(v(H_m)-\underline{v}(H_m)\right),$$

and this leads to

$$\text{Re }H(0)-\underline{v}(H_m)=0-\underline{v}(H_m)\ge\frac{1}{2}\left(v(H_m)-\underline{v}(H_m)\right)\ge\frac{1}{4}\tau(H_m)\left(v(H_m)-\underline{v}(H_m)\right).$$

Combing these two cases, we shall uniformly get

$$\hat{x}=\arg\max\left\{\text{Re }H\left(\frac{1}{d}\sum_{k=1}^{d}\beta_k\hat{x}^k\right),\text{Re }H(0)\right\}$$

satisfying Re $H(\hat{x}) - \underline{v}(H_m) \geq \frac{1}{4}\tau(H_m)\,(v(H_m) - \underline{v}(H_m))$. Finally, by noticing $\hat{x}_i \in \mathrm{conv}(\boldsymbol{\Omega}_m)$ for $i = 1, 2, \ldots, n$ and $H(x)$ is square-free, and applying Lemma 7.3.2, we are able to find $z \in \boldsymbol{\Omega}_m^n$ in polynomial-time, such that

$$\text{Re } H(z) - \underline{v}(H_m) \geq \text{Re } H(\hat{x}) - \underline{v}(H_m) \geq \frac{1}{4}\tau(H_m)\,(v(H_m) - \underline{v}(H_m))\,.$$

□

Before concluding this subsection, we remark that (H_m) can be equivalently transferred to polynomial optimization over discrete variables in the real case, which was discussed in [35]. Essentially, by letting $x = y + \boldsymbol{i}z$ with $y, z \in \mathbb{R}^n$, Re $H(x)$ can be rewritten as a homogeneous polynomial of (y, z), where for each $i = 1, 2, \ldots, n$, $(y_i, z_i) = \left(\cos\frac{2k\pi}{m}, \sin\frac{2k\pi}{m}\right)$ for some $k \in \mathbb{Z}$. By applying the Lagrange polynomial interpolation technique, the problem can then be transferred to an inhomogeneous polynomial optimization with binary constraints, which will yield a worst case relative approximation ratio as well. However, comparing to the bounds obtained in Theorem 7.3.4, the direct transformation to the real case is much worse and more costly to implement.

7.3.2 Homogeneous Polynomial with Unity Constraints

Let us now turn to the case $m \to \infty$. In that case, (H_m) becomes

$$\begin{array}{lll} (H_\infty) & \max & \text{Re } H(x) \\ & \text{s.t.} & x \in \boldsymbol{\Omega}_\infty^n. \end{array}$$

It is not hard to verity (see the proof of Theorem 7.3.5) that (H_∞) is actually equivalent to

$$\begin{array}{ll} \max & |H(x)| \\ \text{s.t.} & x \in \boldsymbol{\Omega}_\infty^n. \end{array}$$

For the case $d = 2$, the above problem was studied by Toker and Ozbay [92], and was termed complex programming. Unlike the case of the m-th roots of unity, where certain conditions on m and d are required to secure approximation ratios, model (H_∞) actually always admits a polynomial-time approximation ratio for any fixed d.

Theorem 7.3.5. *If $H(x)$ is square-free or* Re $H(x)$ *is convex, then (H_∞) admits a polynomial-time randomized approximation algorithm with approximation ratio $\tau(H_\infty) := 0.7118(\frac{\pi}{4})^{d-2}d!d^{-d}n^{-\frac{d-2}{2}}$.*

Proof. Relaxing (H_∞) to (LH_∞), we may find a feasible solution $(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)$ of (LH_∞) in polynomial-time with approximation ratio $0.7118\left(\frac{\pi}{4}\right)^{d-2} n^{-\frac{d-2}{2}}$ by Theorem 7.2.6, i.e.,

$$\text{Re } L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) \geq 0.7118 \left(\frac{\pi}{4}\right)^{d-2} n^{-\frac{d-2}{2}} v(LH_\infty).$$

Then by Lemma 7.3.1, we further find $\beta \in \mathbf{\Omega}_\infty^d$ by randomization, such that

$$\text{Re } \prod_{i=1}^{d} \overline{\beta_i} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) \geq \text{Re } d^{-d} d! L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) \geq \tau(H_\infty) v(LH_\infty) \geq \tau(H_\infty) v(H_\infty).$$

Let $\phi = \arg H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right)$, and we get

$$\begin{aligned} H\left(\frac{e^{-\boldsymbol{i}\phi/d}}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) = e^{-\boldsymbol{i}\phi} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) &= \left| H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right)\right| \\ &\geq \text{Re } \prod_{i=1}^{d} \overline{\beta_i} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right). \end{aligned}$$

Finally, by noticing that each component of $\frac{e^{-\boldsymbol{i}\phi/d}}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k$ is in $\text{conv}(\mathbf{\Omega}_\infty)$, and applying Lemma 7.3.2, we are able to find $y \in \mathbf{\Omega}_\infty^n$ in polynomial-time, such that

$$\text{Re } H(y) \geq \text{Re } H\left(\frac{e^{-\boldsymbol{i}\phi/d}}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) \geq \text{Re } \prod_{i=1}^{d} \overline{\beta_i} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) \geq \tau(H_\infty) v(H_\infty).$$

□

7.3.3 Homogeneous Polynomial with Spherical Constraint

Our last model in this section is spherically constrained homogeneous polynomial optimization in the complex domain

$$\begin{array}{lll} (H_S) & \max & \text{Re } H(x) \\ & \text{s.t.} & x \in \mathbf{CS}^n. \end{array}$$

The model is equivalent to $\max_{x \in \mathbf{CS}^n} |H(x)|$, which is also equivalent to computing the largest eigenvalue of a super-symmetric complex tensor $\mathcal{F} \in \mathbb{C}^{n^d}$.

The real counterpart of (H_S) is studied in the literature; see [33, 36, 59]. The problem is related to computing the largest Z-eigenvalue of a super-symmetric tensor, or equivalently, finding the best rank-one approximation of a super-symmetric tensor [79, 96]. Again, in principle, the complex case can be transformed to the real case by letting $x = y + \boldsymbol{i}z$ with $y, z \in \mathbb{R}^n$, which however increases the number of the variables as well as the dimension of the data tensor $\mathcal{F}$. As a result, this will cause a deterioration in the approximation quality. Moreover, in the real case, (H_S) only admits a *relative* approximation ratio when d is even. Interestingly, for any fixed d, an absolute approximation ratio is possible for the complex case.

Theorem 7.3.6. (H_S) *admits admits a deterministic polynomial-time approximation algorithm with approximation ratio* $\tau(H_S) := d!d^{-d}n^{-\frac{d-2}{2}}$.

Proof. Like in the proof of Theorem 7.3.5, by relaxing (H_S) to (LH_S), we first find a feasible solution $(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d)$ of (LH_S) with approximation ratio $n^{-\frac{d-2}{2}}$ (Theorem 7.2.7). Then by Lemma 7.3.1, we further find $\beta \in \mathbf{\Omega}^d_\infty$, such that

$$\text{Re} \prod_{i=1}^{d} \overline{\beta_i} H\left(\frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k\right) \geq \text{Re}\ d^{-d}d!L(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^d) \geq \tau(H_S)v(LH_S) \geq \tau(H_S)v(H_S).$$

Let $\hat{x} = \frac{1}{d}\sum_{k=1}^{d} \beta_k \hat{x}^k$ and $\phi = \arg H(\hat{x})$. By triangle inequality we have $\|\hat{x}\| \leq \frac{1}{d}\sum_{k=1}^{d} \|\beta_k \hat{x}^k\| = 1$. Finally, $e^{-\boldsymbol{i}\phi/d}\hat{x}/\|\hat{x}\|$ is a feasible solution of (H_S), satisfying

$$\begin{aligned} H\left(e^{-\boldsymbol{i}\phi/d}\frac{\hat{x}}{\|\hat{x}\|}\right) = e^{-\boldsymbol{i}\phi}\|\hat{x}\|^{-d}H(\hat{x}) &= \|\hat{x}\|^{-d}|H(\hat{x})| \geq |H(\hat{x})| \\ &\geq \text{Re} \prod_{i=1}^{d} \overline{\beta_i} H(\hat{x}) \geq \tau(H_S)v(H_S). \end{aligned}$$

□

We remark that the above result does not require $H(x)$ to be square-free or Re $H(x)$ to be convex, which is a condition for Theorems 7.3.3 and 7.3.5.

7.4 Necessary and Sufficient Conditions for Real Valued Complex Polynomials

Recall the conjugate polynomial function is defined as

$$C(\overline{x},x)=\sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} a_{i_1\cdots i_d,j_1\cdots j_d}\overline{x_{i_1}\cdots x_{i_d}}x_{j_1}\cdots x_{j_d}. \tag{7.7}$$

In this section, we also consider a more general complex polynomial called general conjugate polynomial function

$$f_G(\bar{x},x)=\sum_{\substack{1\le i_1\le i_2\cdots\le i_k\le d\\ 1\le j_1\le j_2\le\cdots\le j_{d-k}\le d,\,0\le k\le n}} b_{i_1\cdots i_k,j_1\cdots j_{d-k}}\overline{x_{i_1}\cdots x_{i_k}}x_{j_1}\cdots x_{j_{d-k}}. \tag{7.8}$$

We will study the conditions for these two types of complex polynomials to always take real values and the main conclusion is summarized in the following theorem.

Theorem 7.4.1. *Conjugate polynomial function $C(\overline{x},x)$ defined in* (7.7) *is real valued if and only if*

$$a_{i_1\cdots i_d,j_1\cdots j_d}=\bar{a}_{j_1\cdots j_d,i_1\cdots i_d}\ \forall\ 1\le i_1\le i_2\cdots\le i_d\le n,\ 1\le j_1\le j_2\le\cdots\le j_d\le n, \tag{7.9}$$

General Conjugate polynomial function $f_G(\bar{x},x)$ defined in (7.15) *is real valued if and only if*

$$b_{i_1\cdots i_k,\,j_1\cdots j_d-k}=\bar{b}_{j_1\cdots j_{d-k},\,i_1\cdots i_k}\ \forall\ 1\le i_1\le i_2\cdots\le i_k\le n,\ 0\le k\le d,\\ 1\le j_1\le j_2\le\cdots\le j_{d-k}\le n. \tag{7.10}$$

Before going to the technical proof of Theorem 7.4.1, we would like to present an alternative representation of real valued conjugate polynomials as a corollary of this theorem.

Corollary 7.4.2. *Complex function*

$$C(\overline{x},x)=\sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} a_{i_1\cdots i_d,j_1\cdots j_d}\overline{x_{i_1}\cdots x_{i_d}}x_{j_1}\cdots x_{j_d}$$

is real valued if and only if

$$C(\overline{x},x)=\sum_k \alpha_k|g_k(x)|^2,$$

where $g_k = \sum_{1\le i_1\le i_2\cdots\le i_d\le n} c^k_{i_1\cdots i_d} x_{i_1}\cdots x_{i_d}$ *and* α_k *is a real scaler.*

Proof.Sufficient part is trivial. Suppose now complex function f_C is real valued. Then by Theorem 7.4.1 $a_{i_1\cdots i_d,j_1\cdots j_d} = \bar{a}_{j_1\cdots j_d,i_1\cdots i_d}$. Thus it can be checked that

$$\begin{aligned} & a_{i_1\cdots i_d,j_1\cdots j_d}\overline{x_{i_1}\cdots x_{i_d}}x_{j_1}\cdots x_{j_d} + a_{j_1\cdots j_d,i_1\cdots i_d}\overline{x_{j_1}\cdots x_{j_d}}x_{i_1}\cdots x_{i_d} \\ = \;& |x_{i_1}\cdots x_{i_d} + a_{i_1\cdots i_d,j_1\cdots j_d}x_{j_1}\cdots x_{j_d}|^2 - |x_{i_1}\cdots x_{i_d}|^2 - |a_{i_1\cdots i_d,j_1\cdots j_d}x_{j_1}\cdots x_{j_d}|^2. \end{aligned}$$

Consequently,

$$\begin{aligned} C(\overline{x},x) = \sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} & (|x_{i_1}\cdots x_{i_d} + a_{i_1\cdots i_d,j_1\cdots j_d}x_{j_1}\cdots x_{j_d}|^2 \\ & -|x_{i_1}\cdots x_{i_d}|^2 - |a_{i_1\cdots i_d,j_1\cdots j_d}x_{j_1}\cdots x_{j_d}|^2) \end{aligned}$$

and conclusion follows. □

In the rest of this section, we shall focus on the proof of Theorem 7.4.1. We first notice that the sufficiency is obvious. To see this, let's suppose a complex function $C(\overline{x},x)$ is in form of (7.7) and its conjugate counterpart is

$$\begin{aligned} \overline{C(\overline{x},x)} &= \sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} \bar{a}_{i_1\cdots i_d,j_1\cdots j_d}x_{i_1}\cdots x_{i_d}\overline{x_{j_1}\cdots x_{j_d}} \\ &= \sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} \bar{a}_{j_1\cdots j_d,i_1\cdots i_d}x_{j_1}\cdots x_{j_d}\overline{x_{i_1}\cdots x_{i_d}}. \end{aligned}$$

If the coefficients of $C(\overline{x},x)$ satisfy condition (7.9), i.e. $a_{i_1\cdots i_d,j_1\cdots j_d} = \bar{a}_{j_1\cdots j_d,i_1\cdots i_d}$. Then

$$C(\overline{x},x) - \overline{C(\overline{x},x)} = \sum_{\substack{1\le i_1\le i_2\cdots\le i_d\le n\\ 1\le j_1\le j_2\le\cdots\le j_d\le n}} (a_{i_1\cdots i_d,j_1\cdots j_d} - \bar{a}_{j_1\cdots j_d,i_1\cdots i_d})\,\overline{x_{i_1}\cdots x_{i_d}}x_{j_1}\cdots x_{j_d} = 0$$

implying $C(\overline{x},x)$ is real valued. Similarly, we can also prove the sufficiency of (7.10).

To proceed our discussion on the necessary part, we first consider some easy case: a univariate general conjugate polynomial $\sum_{m=0}^{d}\sum_{j=0}^{m} b_{j,m-j}(\bar{x})^j(x)^{m-j}$, whose property is shown in the following lemma.

Lemma 7.4.3. *Suppose* $\sum_{m=0}^{d}\sum_{j=0}^{m} b_{j,m-j}(\bar{x})^j(x)^{m-j} = 0$ *for all* $x\in\mathbb{C}^1$. *Then we have* $b_{j,m-j} = 0$ *for any* $m = 0,1,\cdots,d$ *and* $j = 0,1,\cdots,m$.

Proof. By letting $x = \rho e^{\boldsymbol{i}\theta}$, the polynomial identity can be rewritten as

$$\sum_{m=0}^{d}\left(\sum_{j=0}^{m} b_{j,m-j} e^{\boldsymbol{i}(m-2j)\theta}\right)\rho^m = 0,\ \forall \rho \in (0,\infty),\ \theta \in (0, 2\pi], \tag{7.11}$$

the left hand side of which can be viewed as a polynomial function with respect to ρ for any fixed θ. Thus the coefficient associated with the highest order monomial ρ^d should be 0, i.e.

$$\sum_{j=0}^{d} b_{j,d-j} e^{\boldsymbol{i}(d-2j)\theta} = 0, \forall \theta \in (0, 2\pi].$$

Consequently,

$$\sum_{j=0}^{d} \text{Re }(b_{j,d-j}) \cos((d-2j)\theta) - \sum_{j=0}^{d} \text{Im }(b_{j,d-j}) \sin((d-2j)\theta) = 0 \tag{7.12}$$

and

$$\sum_{j=0}^{d} \text{Im }(b_{j,d-j}) \cos((d-2j)\theta) + \sum_{j=0}^{d} \text{Re }(b_{j,d-j}) \sin((d-2j)\theta) = 0. \tag{7.13}$$

The first and second summation terms of (7.12) can be respectively simplified as

$$\begin{aligned} &\sum_{j=0}^{d} \text{Re }(b_{j,d-j}) \cos((d-2j)\theta) \\ = &\begin{cases} \sum_{j=0}^{\lfloor \frac{d}{2} \rfloor - 1} \text{Re }(b_{j,d-j} + b_{d-j,j}) \cos((d-2j)\theta), & \text{when d is odd} \\ \sum_{j=0}^{\frac{d}{2}-1} \text{Re }(b_{j,d-j} + b_{d-j,j}) \cos((d-2j)\theta) + \text{Re }(b_{\frac{d}{2},\frac{d}{2}}), & \text{when d is even;} \end{cases} \end{aligned}$$

and

$$\sum_{j=0}^{d} \text{Im }(b_{j,d-j}) \sin((d-2j)\theta) = \sum_{j=0}^{\lfloor \frac{d}{2} \rfloor - 1} \text{Im }(b_{j,d-j} - b_{d-j,j}) \sin((d-2j)\theta).$$

By orthogonality of trigonometric functions, it holds that

$$\text{Re }(b_{j,d-j}) = -\text{Re }(b_{d-j,j}) \text{ and } \text{Im }(b_{j,d-j}) = \text{Im }(b_{d-j,j})\ \forall\ j = 0, 1, \cdots, d.$$

Similarly, (7.13) implies

$$\text{Im } (b_{j,d-j}) = -\text{Im } (b_{d-j,j}) \text{ and Re } (b_{j,d-j}) = \text{Re } (b_{d-j,j}) \; \forall \; j = 0, 1, \cdots, d.$$

Combining the above two identities yields

$$b_{j,d-j} = 0 \; \forall \; j = 0, 1, \cdots, d.$$

and we can reduce the order of polynomial in (7.11) by 1. By repeating the above procedure $d-1$ times, the conclusion follows. □

Actually the above property holds for the generic complex polynomial.

Lemma 7.4.4. *Suppose that for all $x = (x_1, \cdots, x_n)^T \in \mathbb{C}^n$, the d-th order complex polynomial function*

$$\sum_{\ell=1}^{d} \sum_{k=1}^{\ell-1} \sum_{\substack{1 \le i_1 \le \cdots \le i_k \le n \\ 1 \le j_1 \le \cdots \le j_{\ell-k} \le n}} b_{i_1 \cdots i_k, j_1 \cdots j_{\ell-k}} \overline{x_{i_1} \cdots x_{i_k}} x_{j_1} \cdots x_{j_{\ell-k}} +$$

$$\sum_{\ell=1}^{d} \sum_{1 \le i_1 \le \cdots \le i_\ell \le n} b_{i_1 \cdots i_\ell, 0} \overline{x_{i_1} \cdots x_{i_\ell}} + \sum_{\ell=1}^{d} \sum_{1 \le j_1 \le \cdots \le j_\ell \le n} b_{0, j_1 \cdots j_\ell} x_{j_1} \cdots x_{j_\ell} + b_{0,0} = 0 \quad (7.14)$$

Then the coefficients associated to different monomials are all equal to 0. That is $b_{0,0} = 0$, $b_{i_1 \cdots i_\ell, 0} = 0$, $b_{0, j_1 \cdots j_\ell} = 0$ and $b_{i_1 \cdots i_k, j_1 \cdots j_{m-k}} = 0$, $\forall \; \ell = 1, \cdots, d$, $k = 1, \cdots, \ell - 1$, $1 \le i_1 \le \cdots \le i_k \le n$, $1 \le j_1 \le \cdots \le j_{d-k} \le n$.

Proof.We shall prove this result by mathematical induction on the dimension n of the variable. First of all, by letting $x = 0$ we get $b_{0,0} = 0$. Then the case $n = 1$ is already implied by Lemma 7.4.3.

In the following, let's assume the conclusion holds for any complex polynomial with variable dimension no more than $n-1$. Now suppose the variable $x = (x_1, \cdots, x_n)^T \in \mathbb{C}^n$. By fixing $x_2, \cdots, x_n$ as constants and take x_1 as independent variable, equality (7.14) can be rewritten as

$$\sum_{p=1}^{d} \sum_{q=0}^{p} \left(g_{q,p-q}(x_2, \cdots, x_n) \right) \bar{x}_1^q x_1^{p-q} + g_{0,0}(x_2, \cdots, x_n) = 0,$$

where

$$g_{q,p-q}(x_2,\cdots,x_n):=$$
$$\sum_{\ell=p}^{d}\sum_{k=0}^{\ell-p}\sum_{\substack{2\le i_1\le\cdots\le i_k\le n\\ 2\le j_1\le\cdots\le j_{\ell-p-k}\le n}} b_{\underbrace{1\cdots1}_{q}i_1\cdots i_k,\underbrace{1\cdots1}_{p-q}j_1\cdots j_{\ell-p-k}}\overline{x_{i_1}\cdots x_{i_k}}x_{j_1}\cdots x_{j_{\ell-p-k}}$$

and

$$g_{0,0}(x_2,\cdots,x_n):=\sum_{\ell=1}^{d}\sum_{k=1}^{\ell-1}\sum_{\substack{2\le i_1\le\cdots\le i_k\le n\\ 2\le j_1\le\cdots\le j_{\ell-k}\le n}} b_{i_1\cdots i_k,j_1\cdots j_{\ell-k}}\overline{x_{i_1}\cdots x_{i_k}}x_{j_1}\cdots x_{j_{\ell-k}}$$
$$+\sum_{\ell=1}^{d}\sum_{2\le i_1\le\cdots\le i_\ell\le n} b_{i_1\cdots i_\ell,0}\overline{x_{i_1}\cdots x_{i_\ell}}+\sum_{\ell=1}^{d}\sum_{2\le j_1\le\cdots\le j_\ell\le n} b_{0,j_1\cdots j_\ell}x_{j_1}\cdots x_{j_\ell}$$

Due to lemma 7.4.3, we have

$$g_{q,p-q}(x_2,\cdots,x_n)=0,\ \forall\ 1\le p\le d,\ 0\le q\le p,\ \text{and}\ \ g_{0,0}(x_2,\cdots,x_n)=0$$

hold for every $(x_2,\cdots,x_n)^T\in\mathbb{C}^{n-1}$. Notice $g_{q,p-q}(x_2,\cdots,x_n)$ and $g_{0,0}(x_2,\cdots,x_n)$ are all complex polynomial with at most $n-1$ variables. Thus by induction

$$b_{(\underbrace{1\cdots1}_{q}i_1\cdots i_k,\underbrace{1\cdots1}_{p-q}j_1\cdots j_{\ell-p-k})}=0,\ \forall\ 1\le p\le d,\ 0\le q\le p,\ p\le\ell\le d.$$

and

$$b_{i_1\cdots i_\ell,0}=0,\ b_{0,j_1\cdots j_\ell}=0,\ b_{i_1\cdots i_k,j_1\cdots j_{m-k}}=0,\ \forall\ \ell=1,\cdots,d,\ k=1,\cdots,\ell-1,$$
$$2\le i_1\le\cdots\le i_k\le n,\ 2\le j_1\le\cdots\le j_{d-k}\le n.$$

□

Now thanks to Lemma 7.4.4, we can complete the necessary part of Theorem 7.4.1. Suppose $f_G(\bar{x},x)$ in form of (7.15) is real for all x, then $f_G(\bar{x},x)-\overline{f_G(\bar{x},x)}=0$. This is to say

$$\sum_{\substack{1\le i_1\le i_2\cdots\le i_k\le n\\ 1\le j_1\le j_2\le\cdots\le j_{d-k}\le n,\,0\le k\le n}}\left(b_{i_1\cdots i_k,j_1\cdots j_{d-k}}-\bar{b}_{j_1\cdots j_{d-k},i_1\cdots i_k}\right)\overline{x_{i_1}\cdots x_{i_k}}x_{j_1}\cdots x_{j_{d-k}}=0.$$

Then condition (7.10) immediately follows from Lemma 7.4.4. Since $f_C(\bar{x},x)$ is a special case of $f_G(\bar{x},x)$, condition (7.9) is automatically implied by condition (7.10).

Now we can consider a type of complex tensor, which has stronger symmetric property than the conjugate partial-symmetric tensor in Definition 7.1.2

Definition 7.4.5. *A d-th order $2n$ dimensional complex tensor $\mathcal{G} = (\mathcal{G}_{i_1 i_2 \cdots i_d})$ is called conjugate supper-symmetric if*
(i) $\mathcal{G}$ is super-symmetric, i.e. $\mathcal{G}_{i_1 \cdots i_d} = \mathcal{G}_{j_1 \cdots j_d}$, $\forall\ (j_1 \cdots j_d) \in \Pi(i_1 \cdots i_d)$;
(ii) $\mathcal{G}_{i_1 \cdots i_d} = \bar{\mathcal{G}}_{j_1 \cdots j_d}$ if $i_k - j_k = n$ or $j_k - i_k = n$ holds for any $1 \le k \le d$.

$$f_G(\bar{x}, x) = \sum_{\substack{1 \le i_1 \le i_2 \cdots \le i_k \le d \\ 1 \le j_1 \le j_2 \le \cdots \le j_{d-k} \le d,\, 0 \le k \le n}} b_{i_1 \cdots i_k, j_1 \cdots j_{d-k}} \overline{x_{i_1} \cdots x_{i_k}} x_{j_1} \cdots x_{j_{d-k}}. \tag{7.15}$$

Due to the conjugate property discussed above, the conjugate supper-symmetric tensors are also bijectively related to general conjugate polynomials in (7.15). More specifically, given a complex function in form of (7.15), we can construct tensor $\mathcal{G}$ by letting $\mathcal{G}_{k_1 \cdots k_d} = b_{i_1 \cdots i_k, j_1 \cdots j_{d-k}} / |\Pi(n + i_1 \cdots n + i_k j_1 \cdots j_{d-k})|$, when $(k_1 \cdots k_d) \in \Pi(n + i_1 \cdots n + i_k j_1 \cdots j_{d-k})$. Then we can check that $\mathcal{G}$ is hermitian-supper-symmetric and

$$f_G(\bar{x}, x) = \mathcal{G}\underbrace{\left(\begin{pmatrix} x \\ \bar{x} \end{pmatrix}, \begin{pmatrix} x \\ \bar{x} \end{pmatrix}, \cdots, \begin{pmatrix} x \\ \bar{x} \end{pmatrix}\right)}_{d}.$$

7.5 Conjugate Form Optimization

Our last set of optimization models involve the so-called conjugate forms:

$$\begin{array}{lll} (C_m) & \max & C(\overline{x}, x) \\ & \text{s.t.} & x \in \mathbf{\Omega}^n_m; \\ (C_\infty) & \max & C(\overline{x}, x) \\ & \text{s.t.} & x \in \mathbf{\Omega}^n_\infty; \\ (C_S) & \max & C(\overline{x}, x) \\ & \text{s.t.} & x \in \mathbf{CS}^n. \end{array}$$

Recall that the conjugate form $C(\overline{x}, x) = L(\underbrace{\overline{x}, \cdots, \overline{x}}_{d}, \underbrace{x, \cdots, x}_{d})$ is associated with a conjugate partial-symmetric tensor $\mathcal{F} \in \mathbb{C}^{n^{2d}}$ (cf. Section 7.1 for details).

These models are known to have wide applications as well. For instance, (C_m) and (C_∞) with degree 4 are used in the design of radar waveforms sharing an ambiguity function (see Chapter 10, [94] for details). (C_∞) includes (H_∞) as its special case, since (H_∞) is equivalent to $\max_{x\in\mathbf{\Omega}^n_\infty}|H(x)|$, where $|H(x)|^2$ is a special class for $C(\overline{x},x)$. Therefore, complex programming ((H_∞) with $d=2$) studied by Toker and Ozbay [92] also belongs to (C_∞). Similarly, (C_S) also includes (H_S) as its special case.

Let us now focus on approximation algorithms. Observe that for any conjugate partial-symmetric tensor $\mathcal{F}$ with its associated conjugate form $C(\overline{x},x)$:

$$C(\overline{x},x) = \mathrm{Re}\ L(x^1,\cdots,x^d,x^{d+1},\cdots,x^{2d})$$

when $x^1=\cdots=x^d=\overline{x}$ and $x^{d+1}=\cdots=x^{2d}=x$. Therefore, (C_m), (C_∞) and (C_S) can be relaxed to the following multilinear optimization models:

$$\begin{array}{lll}
(LC_m) & \max & \mathrm{Re}\ L(x^1,\cdots,x^d,x^{d+1},\cdots,x^{2d}) \\
 & \text{s.t.} & x^k\in\mathbf{\Omega}^n_m,\ k=1,2,\ldots,2d; \\
(LC_\infty) & \max & \mathrm{Re}\ L(x^1,\cdots,x^d,x^{d+1},\cdots,x^{2d}) \\
 & \text{s.t.} & x^k\in\mathbf{\Omega}^n_\infty,\ k=1,2,\ldots,2d; \\
(LC_S) & \max & \mathrm{Re}\ L(x^1,\cdots,x^d,x^{d+1},\cdots,x^{2d}) \\
 & \text{s.t.} & x^k\in\mathbf{CS}^n,\ k=1,2,\ldots,2d.
\end{array}$$

By the approximation results established in Section 7.2, we are able to find good approximate solutions for these multilinear form optimization models. In order to generate good approximate solutions for the original conjugate form optimizations, we need the following new linkage between the conjugate form and the multilinear form.

Lemma 7.5.1. *Let $m\in\{3,4,\cdots,\infty\}$. Suppose $x^1,x^2,\cdots,x^{2d}\in\mathbb{C}^n$, and $\mathcal{F}\in\mathbb{C}^{n^{2d}}$ is a conjugate partial-symmetric tensor with its associated multilinear form L and conjugate form C. If $\xi_1,\xi_2,\cdots,\xi_{2d}$ are i.i.d. uniform distribution on $\mathbf{\Omega}_m$, then*

$$\begin{array}{rl}
 & \mathsf{E}\left[\left(\prod_{i=1}^{d}\xi_i\right)\left(\prod_{i=d+1}^{2d}\overline{\xi_i}\right)C\left(\sum_{k=1}^{d}\overline{\xi_k}x^k+\sum_{k=d+1}^{2d}\overline{\xi_k x^k},\sum_{k=1}^{d}\xi_k\overline{x^k}+\sum_{k=d+1}^{2d}\xi_k x^k\right)\right] \\
= & (d!)^2L(x^1,x^2,\cdots,x^{2d}).
\end{array}$$

The proof of Lemma 7.5.1 can be found in the appendix. By randomization we find

$\beta \in \mathbf{\Omega}_m^{2d}$ in polynomial-time, such that

$$\text{Re } \left(\prod_{i=1}^{d} \beta_i\right)\left(\prod_{i=d+1}^{2d} \overline{\beta_i}\right) C\left(\overline{x_\beta}, x_\beta\right) \geq (d!)^2 2d^{-2d} \text{Re } L(x^1, x^2, \cdots, x^{2d}), \tag{7.16}$$

where

$$x_\beta := \frac{1}{2d}\sum_{k=1}^{d} \beta_k \overline{x^k} + \frac{1}{2d}\sum_{k=d+1}^{2d} \beta_k x^k. \tag{7.17}$$

7.5.1 Conjugate Form in the m-th Roots of Unity

For (C_m), by relaxing to (LC_m) and generating its approximate solution $(x^1, x^2, \cdots, x^{2d})$ from Theorem 7.2.4, we know $x^k \in \mathbf{\Omega}_m^n$ for $k = 1, 2, \ldots, 2d$. Observe that each component of x_β defined by (7.17) is a convex combination of the elements in $\mathbf{\Omega}_m$, and is thus in $\text{conv}(\mathbf{\Omega}_m)$. Though x_β may not be feasible to (C_m), a vertex solution (in $\mathbf{\Omega}_m$) can be found under certain conditions.

Lemma 7.5.2. *Let $m \in \{3, 4, \cdots, \infty\}$. Suppose $x \in \mathbb{C}^n$ with $x_i \in \text{conv}(\mathbf{\Omega}_m)$ for all $1 \leq i \leq n$.*
(1) If $C(\overline{x}, x)$ is a square-free conjugate form, then $y, z \in \mathbf{\Omega}_m^n$ can be found in polynomial-time, such that $C(\overline{y}, y) \leq C(\overline{x}, x) \leq C(\overline{z}, z)$.
(2) If $C(\overline{x}, x)$ is convex, then $z \in \mathbf{\Omega}_m^n$ can be found in polynomial-time, such that $C(\overline{x}, x) \leq C(\overline{z}, z)$.

The proof is similar to that of Lemma 7.3.2, and is thus omitted. Basically, the algorithm optimizes one variable x_i over $\mathbf{\Omega}_m$ while fixing other $n-1$ variables, alternatively for $i = 1, 2, \ldots, n$. The condition of square-free or convexity guarantees that each step of optimization can be done in polynomial-time. With all these preparations in place, we are ready to present the first approximation result for conjugate form optimization.

Theorem 7.5.3. *If $C(\overline{x}, x)$ is convex, then (C_m) admits a polynomial-time randomized approximation algorithm with approximation ratio $\tau(C_m) := \tau_m^{2d-2}(2\tau_m - 1)(d!)^2(2d)^{-2d}n^{-(d-1)}$.*

Proof. By relaxing (C_m) to (LC_m) and getting its approximate solution $(x^1, x^2, \cdots, x^{2d})$, we have

$$\text{Re } L(x^1, x^2, \cdots, x^{2d}) \geq \tau_m^{2d-2}(2\tau_m - 1)n^{-(d-1)}v(LC_m) \geq \tau_m^{2d-2}(2\tau_m - 1)n^{-(d-1)}v(C_m). \tag{7.18}$$

Applying Lemma 7.5.1, we further get x_β defined by (7.17), satisfying (7.16), i.e.,

$$\text{Re } \left(\prod_{i=1}^{d}\beta_i\right)\left(\prod_{i=d+1}^{2d}\overline{\beta_i}\right) C\left(\overline{x_\beta}, x_\beta\right) \geq (d!)^2 2d^{-2d}\text{Re } L(x^1, x^2, \cdots, x^{2d}) \geq \tau(C_m)v(C_m).$$

Next we notice that any convex conjugate form is always nonnegative [100], i.e., $C(\overline{x}, x) \geq 0$ for all $x \in \mathbb{C}^n$. This further leads to

$$C\left(\overline{x_\beta}, x_\beta\right) \geq \text{Re } \left(\prod_{i=1}^{d}\beta_i\right)\left(\prod_{i=d+1}^{2d}\overline{\beta_i}\right) C\left(\overline{x_\beta}, x_\beta\right) \geq \tau(C_m)v(C_m).$$

Finally, as each component of x_β belongs to $\text{conv}(\mathbf{\Omega}_m)$, applying Lemma 7.5.2, we find $z \in \mathbf{\Omega}_m^n$ with $C(\overline{z}, z) \geq C\left(\overline{x_\beta}, x_\beta\right) \geq \tau(C_m)v(C_m)$. □

As seen from the proof in Theorem 7.5.3, the nonnegativity of convex conjugate form plays an essential role in preserving approximation guarantee. For the general case, this approximation is not possible, since a conjugate form may be negative definite. However under the square-free condition, relative approximation is doable.

Theorem 7.5.4. *If $C(\overline{x}, x)$ is square-free, then (C_m) admits a polynomial-time randomized approximation algorithm with relative approximation ratio $\frac{1}{2}\tau(C_m)$.*

Proof. The main structure of the proof is similar to that of Theorem 7.3.4, based on two complementary cases: $v(C_m) \geq \frac{1}{2}\left(v(C_m) - \underline{v}(C_m)\right)$ and $-\underline{v}(C_m) \geq \frac{1}{2}\left(v(C_m) - \underline{v}(C_m)\right)$. For the latter case, it is obvious that

$$C(\overline{0}, 0) - \underline{v}(C_m) = 0 - \underline{v}(C_m) \geq \frac{1}{2}\left(v(C_m) - \underline{v}(C_m)\right) \geq \frac{1}{2}\tau(C_m)\left(v(C_m) - \underline{v}(C_m)\right). \tag{7.19}$$

For the former case, we relax (C_m) to (LC_m) and get its approximate solution $(x^1, x^2, \cdots, x^{2d})$. By (7.18) it follow that

$$\begin{aligned}(d!)^2(2d)^{-2d}\text{Re } L(x^1, x^2, \cdots, x^{2d}) &\geq (d!)^2(2d)^{-2d}\tau_m^{2d-2}(2\tau_m - 1)n^{-(d-1)}v(C_m) \\ &\geq \frac{1}{2}\tau(C_m)\left(v(C_m) - \underline{v}(C_m)\right). \end{aligned} \tag{7.20}$$

Assume $\xi \in \mathbf{\Omega}_m^{2d}$, whose components are i.i.d. uniform distribution on $\mathbf{\Omega}_m$. As each component of x_ξ defined by (7.17) belongs to $\text{conv}(\mathbf{\Omega}_m)$, by Lemma 7.5.2, there exists $y \in \mathbf{\Omega}_m^n$ such that

$$C(\overline{x_\xi}, x_\xi) \geq C(\overline{y}, y) \geq \underline{v}(C_m). \tag{7.21}$$

Applying Lemma 7.5.1, (7.20) further leads to

$$\begin{aligned}
\frac{1}{2}\tau(C_m)\left(v(C_m)-\underline{v}(C_m)\right) &\leq (d!)^2(2d)^{-2d}\,\mathrm{Re}\;L(x^1,x^2,\cdots,x^{2d}) \\
&= \mathsf{E}\left[\mathrm{Re}\;\left(\prod_{i=1}^{d}\xi_i\right)\left(\prod_{i=d+1}^{2d}\overline{\xi_i}\right)C(\overline{x_\xi},x_\xi)\right] \\
&= \mathsf{E}\left[\mathrm{Re}\;\left(\prod_{i=1}^{d}\xi_i\right)\left(\prod_{i=d+1}^{2d}\overline{\xi_i}\right)(C(\overline{x_\xi},x_\xi)-\underline{v}(C_m))\right] \\
&\leq \mathsf{E}\left[\left|\left(\prod_{i=1}^{d}\xi_i\right)\left(\prod_{i=d+1}^{2d}\overline{\xi_i}\right)\right|\cdot|C(\overline{x_\xi},x_\xi)-\underline{v}(C_m)|\right] \\
&= \mathsf{E}\left[C(\overline{x_\xi},x_\xi)-\underline{v}(C_m)\right],
\end{aligned}$$

where the third step is due to $\mathsf{E}\left[\left(\prod_{i=1}^{d}\xi_i\right)\left(\prod_{i=d+1}^{2d}\overline{\xi_i}\right)\right]=0$, and the last step is due to (7.21). Therefore by randomization, we are able to find $\beta\in\mathbf{\Omega}_m^{2d}$, such that

$$C(\overline{x_\beta},x_\beta)-\underline{v}(C_m)\geq\mathsf{E}\left[C(\overline{x_\xi},x_\xi)-\underline{v}(C_m)\right]\geq\frac{1}{2}\tau(C_m)\left(v(C_m)-\underline{v}(C_m)\right).$$

Combining (7.19), if we let $x'=\arg\max\left\{C(\overline{0},0),C(\overline{x_\beta},x_\beta)\right\}$, then we shall uniformly have $C(\overline{x'},x')-\underline{v}(C_m)\geq\frac{1}{2}\tau(C_m)\left(v(C_m)-\underline{v}(C_m)\right)$. Finally, as each component of x' belongs to $\mathrm{conv}(\mathbf{\Omega}_m)$ and $C(\overline{x},x)$ is square-free, by Lemma 7.5.2, we are able to find $z\in\mathbf{\Omega}_m^n$ in polynomial-time, such that

$$C(\overline{z},z)-\underline{v}(C_m)\geq C(\overline{x'},x')-\underline{v}(C_m)\geq\frac{1}{2}\tau(C_m)\left(v(C_m)-\underline{v}(C_m)\right).$$

□

7.5.2 Conjugate form with Unity Constraints or Spherical Constraint

The discussion in Section 7.5.1 can be extended to conjugate form optimization over unity constraints, and the complex spherical constraint: (C_∞) and (C_S). Due to its similar nature, here we shall skip the details and only provide the main approximation results; the details can be easily supplemented by the interested reader. Essentially, the main steps are: (1) relax to multilinear form optimization models and find their approximate solutions as discussed in Section 7.2; (2) conduct randomization based on the link provided in Lemma 7.5.1; (3) search for the best vertex solution. For the

complex unity constrained (C_∞), a vertex solution is guaranteed by Lemma 7.5.2, and for the spherically constrained (C_S), a vertex solution is obtained by scaling to $\mathbf{CS}^n$: $x_\beta/\|x_\beta\|$.

Theorem 7.5.5. *(1) If $C(\overline{x}, x)$ is convex, then (C_∞) admits a polynomial-time randomized approximation algorithm with approximation ratio*

$$\tau(C_\infty) := 0.7118 \left(\frac{\pi}{4}\right)^{2d-2} (d!)^2 (2d)^{-2d} n^{-(d-1)}.$$

(2) If $C(\overline{x}, x)$ is square-free, then (C_∞) admits a polynomial-time randomized approximation algorithm with relative approximation ratio $\frac{1}{2}\tau(C_\infty)$.

Theorem 7.5.6. *(1) If $C(\overline{x}, x)$ is nonnegative (including convex as its special case), then (C_S) admits a deterministic polynomial-time approximation algorithm with approximation ratio $\tau(C_S) := (d!)^2(2d)^{-2d}n^{-(d-1)}$.*
(2) For general $C(\overline{x}, x)$, (C_S) admits a deterministic polynomial-time approximation algorithm with relative approximation ratio $\frac{1}{2}\tau(C_S)$.

Chapter 8

Tensor Principal Component Analysis via Convex Optimization

8.1 Introduction

Principal component analysis (PCA) plays an important role in applications arising from data analysis, dimension reduction and bioinformatics etc. PCA finds a few linear combinations of the original variables. These linear combinations, which are called principal components (PCs), are orthogonal to each other and explain most of the variance of the data. PCs provide a powerful tool to compress data along the direction of maximum variance to reach the minimum information loss. Specifically, let $\xi = (\xi_1, \ldots, \xi_m)$ be an m-dimensional random vector. Then for a given data matrix $A \in \mathbb{R}^{m\times n}$ which consists of n samples of the m variables, finding the PC that explains the largest variance of the variables $(\xi_1, \ldots, \xi_m)$ corresponds to the following optimization problem:

$$(\lambda^*, x^*, y^*) := \min_{\lambda\in\mathbb{R}, x\in\mathbb{R}^m, y\in\mathbb{R}^n} \|A - \lambda x y^\top\|. \tag{8.1}$$

Problem (8.1) is well known to be reducible to computing the largest singular value (and corresponding singular vectors) of A, and can be equivalently formulated as:

$$\begin{array}{ll} \max_{x,y} & \begin{pmatrix} x \\ y \end{pmatrix}^\top \begin{pmatrix} 0 & A \\ A^\top & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{s.t.} & \left\| \begin{pmatrix} x \\ y \end{pmatrix} \right\| = 1. \end{array} \tag{8.2}$$

Note that the optimal value and the optimal solution of Problem (8.2) correspond to the largest eigenvalue and the corresponding eigenvector of the symmetric matrix $\begin{pmatrix} 0 & A \\ A^\top & 0 \end{pmatrix}$.

Although the PCA and eigenvalue problem for matrix have been well studied in the literature, the research of PCA for tensors is still lacking. Nevertheless, the tensor PCA is of great importance in practice and has many applications in computer vision [101], diffusion Magnetic Resonance Imaging (MRI) [2, 102, 103], quantum entanglement problem [95], spectral hypergraph theory [104] and higher-order Markov chains [105]. This is mainly because in real life we often encounter multidimensional data, such as images, video, range data and medical data such as CT and MRI. A color image can be considered as 3D data with row, column, color in each direction, while a color video sequence can be considered as 4D data, where time is the fourth dimension. Moreover, it turns out that it is more reasonable to treat the multidimensional data as a tensor instead of unfolding it into a matrix. For example, Wang and Ahuja [101] reported that the images obtained by tensor PCA technique have higher quality than that by matrix PCA. Similar to its matrix counterpart, the problem of finding the PC that explains the most variance of a tensor $\mathcal{A}$ (with degree m) can be formulated as:

$$\begin{array}{ll} \min & \|\mathcal{A} - \lambda x^1 \otimes x^2 \otimes \cdots \otimes x^m\| \\ \text{s.t.} & \lambda \in \mathbb{R},\ \|x^i\| = 1, i = 1, 2, \ldots, m, \end{array} \tag{8.3}$$

which is equivalent to

$$\begin{array}{ll} \max & \mathcal{A}(x^1, x^2, \cdots, x^m) \\ \text{s.t.} & \|x^i\| = 1, i = 1, 2, \ldots, m. \end{array} \tag{8.4}$$

Once the most leading PC has been found, other leading PCs can be computed sequentially via the so-called "deflation" technique. For instance to find the second

leading PC, this technique works as first subtracting a rank-one tensor that is formed by the first leading PC, and then computing the most leading PC of the resulting tensor. Of course, this procedure may not be well justified for tensor eigenvalue problem, although it is valid in the matrix case. However, it still provides a way to compute multiple principal components of a tensor approximately and heuristically. Thus in the rest of this chapter, we focus on finding the most leading PC of a tensor.

Problem (8.4) is also known as the best rank-one approximation of tensor $\mathcal{A}$; cf. [76, 106]. As we shall find out later, Problem (8.4) can be reformulated as

$$\begin{array}{ll} \max & \mathcal{F}(x, x, \cdots, x) \\ \text{s.t.} & \|x\| = 1, \end{array} \tag{8.5}$$

where $\mathcal{F}$ is a super-symmetric tensor. Problem (8.5) is NP-hard and is known as the maximum Z-eigenvalue problem. Note that a variety of eigenvalues and eigenvectors of a real symmetric tensor are introduced by Lim [63] and Qi [64] independently in 2005. Since then, various methods have been proposed to find the Z-eigenvalues [79, 65, 106, 76, 107], which possibly correspond to some local optimums. In this chapter, we shall focus on finding the global optimal solution of (8.5).

In the subsequent analysis, for convenience we assume m to be even, i.e., $m = 2d$ in (8.5), where d is a positive integer. As we will see later, this assumption is essentially not restrictive. Therefore, we will focus on the following problem of largest eigenvalue of an even order super-symmetric tensor:

$$\begin{array}{ll} \max & \mathcal{F}(\underbrace{x, \cdots, x}_{2d}) \\ \text{s.t.} & \|x\| = 1, \end{array} \tag{8.6}$$

where $\mathcal{F}$ is a $2d$-th order super-symmetric tensor. In particular, problem (8.6) can be equivalently written as

$$\begin{array}{ll} \max & \mathcal{F} \bullet \underbrace{x \otimes \cdots \otimes x}_{2d} \\ \text{s.t.} & \|x\| = 1. \end{array} \tag{8.7}$$

In this chapter, given any $2d$-th order super-symmetric tensor form $\mathcal{F}$, we call it *rank one* if its real symmetric CP rank (see Definition 4.4.1) equals to one, i.e., $\mathcal{F} = \lambda \underbrace{a \otimes \cdots \otimes a}_{2d}$ for some $a \in \mathbb{R}^n$ and $\lambda \neq 0 \in \mathbb{R}^1$.

In the following, to simplify the notation, we denote

$$\mathbb{K}(n,d) = \left\{ k = (k_1, \cdots, k_n) \in \mathbb{Z}_+^n \;\middle|\; \sum_{j=1}^{n} k_j = d \right\}$$

and

$$\mathcal{X}_{1^{2k_1}2^{2k_2}\ldots n^{2k_n}} := \mathcal{X}_{\underbrace{1\ldots1}_{2k_1}\underbrace{2\ldots2}_{2k_2}\ldots\underbrace{n\ldots n}_{2k_n}}.$$

By letting $\mathcal{X} = \underbrace{x \otimes \cdots \otimes x}_{2d}$ we can further convert problem (8.7) into:[1]

$$\begin{array}{ll} \max & \mathcal{F} \bullet \mathcal{X} \\ \text{s.t.} & \sum\limits_{k\in\mathbb{K}(n,d)} \frac{d!}{\prod_{j=1}^n k_j!} \mathcal{X}_{1^{2k_1}2^{2k_2}\ldots n^{2k_n}} = 1, \\ & \mathcal{X} \in \mathbf{S}^{n^{2d}},\ \text{rank}_{RCP}(\mathcal{X}) = 1, \end{array} \tag{8.8}$$

where the first equality constraint is due to the fact that

$$\sum_{k\in\mathbb{K}(n,d)} \frac{d!}{\prod_{j=1}^n k_j!} \prod_{j=1}^{n} x_j^{2k_j} = \|x\|^{2d} = 1.$$

The difficulty of the above problem lies in the dealing of the rank constraint rank $(\mathcal{X}) = 1$. Not only the rank function itself is difficult to deal with, but, as we already mentioned earlier, determining the rank of a specific given tensor is NP-hard in general [25]. One way to deal with the difficulty is to convert the tensor optimization problem (8.8) into a matrix optimization problem. A typical matricization technique is the so-called mode-n matricization [76], which is also dicussed in Chapter 4.1. Most recently, Liu *et al.* [49] and Gandy *et al.* [77] have used this notion to study the low-n-rank tensor recovery problem. Along with this line, Tomioka *et al.* [108] analyzed the statistical performance of nuclear norm relaxation of the tensor n-rank minimization problem. However, up till now, the relationship between the n-rank and CP rank is still unclear. Chandrasekaran *et al.* [109] propose another interesting idea, in particular they directly apply convex relaxation to the tensor rank and obtain a new norm called tensor nuclear norm, which is numerically intractable. Thus, a further semidefinite representable relaxation is introduced. However, the authors did not provide any numerical results for this relaxation.

[1] See Definition 4.4.1 for the detailed description of the function rank $_{RCP}(\cdot)$.

Therefore, in the following we apply the new operator $\boldsymbol{M}(\cdot)$ of matrix unfolding introduced in 4.1.2, and transform the tensor problem (8.8) into a matrix problem. To this end, we denote $X = \boldsymbol{M}(\mathcal{X})$, and so

$$\mathrm{tr}\,(X) = \sum_{\ell} X_{\ell,\ell} \text{ with } \ell = \sum_{j=1}^{d} (i_j - 1) n^{d-j} + 1.$$

If we assume $\mathcal{X}$ to be of rank one, then

$$\mathrm{tr}\,(X) = \sum_{i_1,\cdots,i_d} \mathcal{X}_{i_1\cdots i_d i_1 \cdots i_d} = \sum_{i_1,\cdots,i_d} \mathcal{X}_{i_1^2 \cdots i_d^2}.$$

In the above expression, $(i_1, \cdots, i_d)$ is a subset of $(1, 2, \ldots, n)$. Suppose that j appears k_j times in $(i_1, \cdots, i_d)$ with $j = 1, 2, \ldots, n$ and $\sum_{j=1}^{n} k_j = d$. Then for a fixed outcome $(k_1, k_2, \cdots, k_n)$, the total number of permutations $(i_1, \cdots, i_d)$ to achieve such outcome is

$$\binom{d}{k_1}\binom{d-k_1}{k_2}\binom{d-k_1-k_2}{k_3}\cdots\binom{d-k_1-\cdots-k_{n-1}}{k_n} = \frac{d!}{\prod_{j=1}^{n} k_j!}.$$

Consequently,

$$\mathrm{tr}\,(X) = \sum_{i_1,\cdots,i_d} \mathcal{X}_{i_1^2 \cdots i_d^2} = \sum_{k \in \mathbb{K}(n,d)} \frac{d!}{\prod_{j=1}^{n} k_j!} \mathcal{X}_{1^{2k_1} 2^{2k_2} \cdots n^{2k_n}}. \tag{8.9}$$

In light of the above discussion, if we further denote $F = \boldsymbol{M}(\mathcal{F})$, then the objective in (8.8) is $\mathcal{F} \bullet \mathcal{X} = \mathrm{tr}\,(FX)$, while the first constraint $\sum_{k \in \mathbb{K}(n,d)} \frac{d!}{\prod_{j=1}^{n} k_j!} \mathcal{X}_{1^{2k_1} 2^{2k_2} \cdots n^{2k_n}} = 1 \iff \mathrm{tr}\,(X) = 1$. The hard constraint in (8.8) is $\mathrm{rank}_{RCP}(\mathcal{X}) = 1$. According to Theorem 4.4.7, we know that a super-symmetric tensor is of rank one, if and only if its matrix correspondence obtained via the matricization operation defined in Definition 4.1.2, is also of rank one. As a result, we can reformulate Problem (8.8) equivalently as the following matrix optimization problem:

$$\begin{array}{ll} \max & \mathrm{tr}\,(FX) \\ \text{s.t.} & \mathrm{tr}\,(X) = 1,\ \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}, \\ & X \in \mathbf{S}^{n^d \times n^d},\ \mathrm{rank}\,(X) = 1, \end{array} \tag{8.10}$$

where $X = \boldsymbol{M}(\mathcal{X})$, $F = \boldsymbol{M}(\mathcal{F})$, and $\mathbf{S}^{n^d \times n^d}$ denotes the set of $n^d \times n^d$ symmetric matrices. Note that the constraints $\boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}$ requires the tensor correspondence

of X to be super-symmetric, which essentially correspond to $O(n^{2d})$ linear equality constraints. The rank constraint $\text{rank}\,(X) = 1$ makes the problem intractable. In fact, Problem (8.10) is NP-hard in general, due to its equivalence to problem (8.6). So the tasks of followings sections are dedicated to how to solve Problem (8.10).

8.2 A Nuclear Norm Penalty Approach

There have been a large amount of work that deal with the low-rank matrix optimization problems. Research in this area was mainly ignited by the recent emergence of compressed sensing [110, 111], matrix rank minimization and low-rank matrix completion problems [112, 113, 114]. The matrix rank minimization seeks a matrix with the lowest rank satisfying some linear constraints, i.e.,

$$\min_{X\in\mathbb{R}^{n_1\times n_2}} \ \text{rank}\,(X),\ \text{s.t.},\ \mathcal{C}(X) = b, \tag{8.11}$$

where $b \in \mathbb{R}^p$ and $\mathcal{C} : \mathbb{R}^{n_1\times n_2} \to \mathbb{R}^p$ is a linear operator. The works of [112, 113, 114] show that under certain randomness hypothesis of the linear operator $\mathcal{C}$, the NP-hard problem (8.11) is equivalent to the following nuclear norm minimization problem, which is a convex programming problem, with high probability:

$$\min_{X\in\mathbb{R}^{n_1\times n_2}} \ \|X\|_*,\ \text{s.t.},\ \mathcal{C}(X) = b. \tag{8.12}$$

In other words, the optimal solution to the convex problem (8.12) is also the optimal solution to the original NP-hard problem (8.11).

Motivated by the convex nuclear norm relaxation, one way to deal with the rank constraint in (8.10) is to introduce the nuclear norm term of X, which penalizes high-ranked X's, in the objective function. This yields the following convex optimization formulation:

$$\begin{array}{ll} \max & \text{tr}\,(FX) - \rho\|X\|_* \\ \text{s.t.} & \text{tr}\,(X) = 1,\ \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}, \\ & X \in \mathbf{S}^{n^d\times n^d}, \end{array} \tag{8.13}$$

where $\rho > 0$ is a penalty parameter. It is easy to see that if the optimal solution of (8.13) (denoted by $\tilde{X}$) is of rank one, then $\|\tilde{X}\|_* = \text{tr}\,(\tilde{X}) = 1$, which is a constant. In this case, the term $-\rho\|X\|_*$ added to the objective function is a constant, which leads

to the fact the solution is also optimal with the constraint that X is rank-one. In fact, Problem (8.13) is the convex relaxation of the following problem

$$\begin{array}{ll} \max & \mathrm{tr}\,(FX) - \rho\|X\|_* \\ \text{s.t.} & \mathrm{tr}\,(X) = 1,\ \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}, \\ & X \in \mathbf{S}^{n^d \times n^d},\ \mathrm{rank}\,(X) = 1, \end{array}$$

which is equivalent to the original problem (8.10) since $\rho\|X\|_* = \rho\mathrm{tr}\,(X) = \rho$.

After solving the convex optimization problem (8.13) and obtaining the optimal solution $\tilde{X}$, if $\mathrm{rank}\,(\tilde{X}) = 1$, we can find $\tilde{x}$ such that $\boldsymbol{M}^{-1}(\tilde{X}) = \underbrace{\tilde{x} \otimes \cdots \otimes \tilde{x}}_{2d}$, according to Theorem 4.4.7. In this case, $\tilde{x}$ is the optimal solution to Problem (8.6). The original tensor PCA problem, or the Z-eigenvalue problem (8.6), is thus solved to optimality.

Interestingly, we found from our extensive numerical tests that the optimal solution to Problem (8.13) is a rank-one matrix almost all the time. In the following, we will show this interesting phenomenon by some concrete examples. The first example is taken from [106].

Example 8.2.1. *We consider a super-symmetric tensor $\mathcal{F} \in \mathbf{S}^{3^4}$ defined by*

$$\begin{array}{lllll} \mathcal{F}_{1111} = 0.2883, & \mathcal{F}_{1112} = -0.0031, & \mathcal{F}_{1113} = 0.1973, & \mathcal{F}_{1122} = -0.2485, & \mathcal{F}_{1123} = -0.2939, \\ \mathcal{F}_{1133} = 0.3847, & \mathcal{F}_{1222} = 0.2972, & \mathcal{F}_{1223} = 0.1862, & \mathcal{F}_{1233} = 0.0919, & \mathcal{F}_{1333} = -0.3619, \\ \mathcal{F}_{2222} = 0.1241, & \mathcal{F}_{2223} = -0.3420, & \mathcal{F}_{2233} = 0.2127, & \mathcal{F}_{2333} = 0.2727, & \mathcal{F}_{3333} = -0.3054. \end{array}$$

We want to compute the largest Z-eigenvalue of $\mathcal{F}$.

Since the size of this tensor is small, we used CVX [115] to solve Problem (8.13) with $F = \boldsymbol{M}(\mathcal{F})$ and $\rho = 10$. It turned out that CVX produced a rank-one solution $\tilde{X} = aa^\top \in \mathbb{R}^{3^2 \times 3^2}$, where

$$a = (0.4451, 0.1649, -0.4688, 0.1649, 0.0611, -0.1737, -0.4688, -0.1737, 0.4938)^\top.$$

Thus we get the matrix correspondence of a by reshaping a into a square matrix A:

$$A = [a(1:3), a(4:6), a(7:9)] = \begin{bmatrix} 0.4451 & 0.1649 & -0.4688 \\ 0.1649 & 0.0611 & -0.1737 \\ -0.4688 & -0.1737 & 0.4938 \end{bmatrix}.$$

It is easy to check that A is a rank-one matrix with the nonzero eigenvalue being 1. This further confirms our theory on the rank-one equivalence, i.e., Theorem 4.4.7. The eigenvector that corresponds to the nonzero eigenvalue of A is given by

$$\tilde{x} = (-0.6671, -0.2472, 0.7027)^\top,$$

which is the optimal solution to Problem (8.6).

The next example is from a real Magnetic Resonance Imaging (MRI) application studied by Ghosh et al. in [2]. In [2], Ghosh et al. studied a fiber detection problem in diffusion Magnetic Resonance Imaging (MRI), where they tried to extract the geometric characteristics from an antipodally symmetric spherical function (ASSF), which can be described equivalently in the homogeneous polynomial basis constrained to the sphere. They showed that it is possible to extract the maxima and minima of an ASSF by computing the stationary points of a problem in the form of (8.6) with $d = 2$ and $n = 4$.

Example 8.2.2. *The objective function $\mathcal{F}(x, x, x, x)$ in this example is given by*

$$\begin{array}{rrrrr} 0.74694x_1^4 & - 0.435103x_1^3x_2 & + 0.454945x_1^2x_2^2 & + 0.0657818x_1x_2^3 & + x_2^4 \\ + 0.37089x_1^3x_3 & - 0.29883x_1^2x_2x_3 & - 0.795157x_1x_2^2x_3 & + 0.139751x_2^3x_3 & + 1.24733x_1^2x_3^2 \\ + 0.714359x_1x_2x_3^2 & + 0.316264x_2^2x_3^2 & - 0.397391x_1x_3^3 & - 0.405544x_2x_3^3 & + 0.794869x_3^4. \end{array}$$

Again, we used CVX to solve problem (8.13) with $F = \boldsymbol{M}(\mathcal{F})$ and $\rho = 10$, and a rank-one solution was found with $\tilde{X} = aa^\top$, with

$$a = (0.0001, 0.0116, 0.0004, 0.0116, 0.9984, 0.0382, 0.0004, 0.0382, 0.0015)^\top.$$

By reshaping vector a, we get the following expression of matrix A:

$$A = [a(1:3), a(4:6), a(7:9)] = \begin{bmatrix} 0.0001 & 0.0116 & 0.0004 \\ 0.0116 & 0.9984 & 0.0382 \\ 0.0004 & 0.0382 & 0.0015 \end{bmatrix}.$$

It is easy to check that A is a rank-one matrix with 1 being the nonzero eigenvalue. The eigenvector corresponding to the nonzero eigenvalue of A is given by

$$\tilde{x} = (0.0116, 0.9992, 0.0382)^\top,$$

which is also the optimal solution to the original problem (8.6).

We then conduct some numerical tests on randomly generated examples. We construct 4-th order tensor $\mathcal{T}$ with its components drawn randomly from i.i.d. standard normal distribution. The super-symmetric tensor $\mathcal{F}$ in the tensor PCA problem is obtained by symmetrizing $\mathcal{T}$. All the numerical experiments in this chapter were conducted on an Intel Core i5-2520M 2.5GHz computer with 4GB of RAM, and all the default settings of Matlab 2012b and CVX 1.22 were used for all the tests. We choose $d = 2$ and the dimension of $\mathcal{F}$ in the tensor PCA problem from $n = 3$ to $n = 9$. We choose $\rho = 10$. For each n, we tested 100 random instances. In Table 8.1, we report the number of instances that produced rank-one solutions. We also report the average CPU time (in seconds) using CVX to solve the problems.

n	rank-1	CPU
3	100	0.21
4	100	0.56
5	100	1.31
6	100	6.16
7	100	47.84
8	100	166.61
9	100	703.82

Table 8.1: Frequency of nuclear norm penalty problem (8.13) having a rank-one solution

Table 8.1 shows that for these randomly created tensor PCA problems, the nuclear norm penalty problem (8.13) *always* gives a rank-one solution, and thus *always* solves the original problem (8.6) to optimality.

8.3 Semidefinite Programming Relaxation

In this section, we study another convex relaxation for Problem (8.10). Note that the constraint

$$X \in \mathbf{S}^{n^d \times n^d}, \operatorname{rank}(X) = 1$$

in (8.10) actually implies that X is positive semidefinite. To get a tractable convex problem, we drop the rank constraint and impose a semidefinite constraint to (8.10) and consider the following SDP relaxation:

$$\begin{array}{lll} (SDR) & \max & \mathrm{tr}\,(FX) \\ & \text{s.t.} & \mathrm{tr}\,(X) = 1, \\ & & \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}},\ X \succeq 0. \end{array} \tag{8.14}$$

Remark that replacing the rank-one constraint by SDP constraint is by now a common and standard practice; see, e.g., [116, 80, 117]. Next theorem shows that the SDP relaxation (8.14) is actually closely related to the nuclear norm penalty problem (8.13).

Theorem 8.3.1. *Let X^*_{SDR} and $X^*_{PNP}(\rho)$ be the optimal solutions of problems* (8.14) *and* (8.13) *respectively. Suppose $Eig^+(X)$ and $Eig^-(X)$ are the summations of non-negative eigenvalues and negative eigenvalues of X respectively, i.e.,*

$$Eig^+(X) := \sum_{i:\,\lambda_i(X)\geq 0} \lambda_i(X), \quad Eig^-(X) := \sum_{i:\,\lambda_i(X)<0} \lambda_i(X).$$

It holds that

$$2(\rho - v)\left|Eig^-(X^*_{PNP}(\rho))\right| \leq v - F_0,$$

where $F_0 := \max\limits_{1\leq i\leq n} \mathcal{F}_{i^{2d}}$ and v is the optimal value of the following optimization problem

$$\begin{array}{ll} \max & \mathrm{tr}\,(FX) \\ \text{s.t.} & \|X\|_* = 1, \\ & X \in \mathbf{S}^{n^d\times n^d}. \end{array} \tag{8.15}$$

As a result, $\lim\limits_{\rho\to+\infty} \mathrm{tr}\,(FX^*_{PNP}(\rho)) = \mathrm{tr}\,(FX^*_{SDR})$.

Proof. Observe that $\boldsymbol{M}(\underbrace{e^i\otimes\cdots\otimes e^i}_{2d})$, where e^i is the i-th unit vector, is a feasible solution for problem (8.13) with objective value $\mathcal{F}_{i^{2d}} - \rho$ for all $1 \leq i \leq n$. Moreover, by denoting $r(\rho) = |Eig^-(X^*_{PNP}(\rho))|$, we have

$$\begin{aligned} \|X^*_{PNP}(\rho)\|_* &= Eig^+(X^*_{PNP}(\rho)) + \left|Eig^-(X^*_{PNP}(\rho))\right| \\ &= \left(Eig^+(X^*_{PNP}(\rho)) + Eig^-(X^*_{PNP}(\rho))\right) + 2\left|Eig^-(X^*_{PNP}(\rho))\right| \\ &= 1 + 2r(\rho). \end{aligned}$$

Since $X^*_{PNP}(\rho)$ is optimal to problem (8.13), we have

$$\mathrm{tr}\,(FX^*_{PNP}(\rho)) - \rho\,(1 + 2r(\rho)) \geq \max_{1\leq i\leq n} \mathcal{F}_{i^{2d}} - \rho \geq F_0 - \rho. \tag{8.16}$$

Moreover, since $X^*_{PNP}(\rho)/\|X^*_{PNP}(\rho)\|_*$ is feasible to problem (8.15), we have

$$\mathrm{tr}\,(FX^*_{PNP}(\rho)) \leq \|X^*_{PNP}(\rho)\|_*\, v = (1 + 2r(\rho))\, v. \tag{8.17}$$

Combining (8.17) and (8.16) yields

$$2(\rho - v)\, r(\rho) \leq v - F_0. \tag{8.18}$$

Notice that $\|X\|_* = 1$ implies $\|X\|_\infty$ is bounded for all feasible $X \in \mathbf{S}^{n^d\times n^d}$, where $\|X\|_\infty$ denotes the largest entry of X in magnitude. Thus the set $\{X^*_{PNP}(\rho) \mid \rho > 0\}$ is bounded. Let X^*_{PNP} be one cluster point of sequence $\{X^*_{PNP}(\rho) \mid \rho > 0\}$. By taking the limit $\rho \to +\infty$ in (8.18), we have $r(\rho) \to 0$ and thus $X^*_{PNP} \succeq 0$. Consequently, X^*_{PNP} is a feasible solution to problem (8.14) and $\mathrm{tr}\,(FX^*_{SDR}) \geq \mathrm{tr}\,(FX^*_{PNP})$. On the other hand, it is easy to check that for any $0 < \rho_1 < \rho_2$,

$$\mathrm{tr}\,(FX^*_{SDR}) \leq \mathrm{tr}\,(FX^*_{PNP}(\rho_2)) \leq \mathrm{tr}\,(FX^*_{PNP}(\rho_1)),$$

which implies $\mathrm{tr}\,(FX^*_{SDR}) \leq \mathrm{tr}\,(FX^*_{PNP})$. Therefore, $\lim_{\rho\to+\infty} \mathrm{tr}\,(FX^*_{PNP}(\rho)) = \mathrm{tr}\,(FX^*_{PNP}) = \mathrm{tr}\,(FX^*_{SDR})$. □

Theorem (8.3.1) shows that when ρ goes to infinity in (8.13), the optimal solution of the nuclear norm penalty problem (8.13) converges to the optimal solution of the SDP relaxation (8.14). As we have shown in Table 8.1 that the nuclear norm penalty problem (8.13) returns rank-one solutions for all the randomly created tensor PCA problems that we tested, it is expected that the SDP relaxation (8.14) will also be likely to give rank-one solutions. In fact, this is indeed the case as shown through the numerical results in Table 8.2. As in Table 8.1, we tested 100 random instances for each n. In Table 8.2, we report the number of instances that produced rank-one solutions for $d = 2$. We also report the average CPU time (in seconds) using CVX to solve the problems. As we see from Table 8.2, for these randomly created tensor PCA problems, the SDP relaxation (8.14) *always* gives a rank-one solution, and thus *always* solves the original problem (8.6) to optimality.

n	rank-1	CPU
3	100	0.14
4	100	0.25
5	100	0.55
6	100	1.16
7	100	2.37
8	100	4.82
9	100	8.89

Table 8.2: Frequency of SDP relaxation (8.14) having a rank-one solution

8.4 Alternating Direction Method of Multipliers

The computational times reported in Tables 8.1 and 8.2 suggest that it can be time consuming to solve the convex problems (8.13) and (8.14) when the problem size is large (especially for the nuclear norm penalty problem (8.13)). In this section, we propose an alternating direction method of multipliers (ADMM) for solving (8.13) and (8.14) that fully takes advantage of the structures. ADMM is closely related to some operator-splitting methods, known as Douglas-Rachford and Peaceman-Rachford methods, that were proposed in 1950s for solving variational problems arising from PDEs (see [118, 119]). These operator-splitting methods were extensively studied later in the literature for finding the zeros of the sum of monotone operators and for solving convex optimization problems (see [120, 121, 122, 123, 124]). The ADMM we will study in this section was shown to be equivalent to the Douglas-Rachford operator-splitting method applied to convex optimization problem (see [125]). ADMM was revisited recently as it was found to be very efficient for many sparse and low-rank optimization problems arising from the recent emergence of compressed sensing [126], compressive imaging [127, 128], robust PCA [129], sparse inverse covariance selection [130, 131], sparse PCA [132] and SDP [133] etc. For a more complete discussion and list of references on ADMM, we refer to the recent survey paper by Boyd et al. [134] and the references therein.

Generally speaking, ADMM solves the following convex optimization problem,

$$\begin{array}{ll} \min_{x\in\mathbb{R}^n, y\in\mathbb{R}^p} & f(x)+g(y) \\ \text{s.t.} & Ax+By=b \\ & x\in\mathcal{C},\ y\in\mathcal{D}, \end{array} \tag{8.19}$$

where f and g are convex functions, $A\in\mathbb{R}^{m\times n}$, $B\in\mathbb{R}^{m\times p}$, $b\in\mathbb{R}^m$, $\mathcal{C}$ and $\mathcal{D}$ are some simple convex sets. A typical iteration of ADMM for solving (8.19) can be described as follows:

$$\left\{\begin{array}{lcl} x^{k+1} & := & \operatorname{argmin}_{x\in\mathcal{C}}\ \mathcal{L}_\mu(x, y^k; \lambda^k) \\ y^{k+1} & := & \operatorname{argmin}_{y\in\mathcal{D}}\ \mathcal{L}_\mu(x^{k+1}, y; \lambda^k) \\ \lambda^{k+1} & := & \lambda^k - (Ax^{k+1}+By^{k+1}-b)/\mu, \end{array}\right. \tag{8.20}$$

where the augmented Lagrangian function $\mathcal{L}_\mu(x,y;\lambda)$ is defined as

$$\mathcal{L}_\mu(x,y;\lambda) := f(x)+g(y)-\langle\lambda, Ax+By-b\rangle+\frac{1}{2\mu}\|Ax+By-b\|^2,$$

λ is the Lagrange multiplier and $\mu>0$ is a penalty parameter. The following theorem gives the global convergence of (8.20) for solving (8.19), and this has been well studied in the literature (see, e.g., [121, 123]).

Theorem 8.4.1. *Assume both A and B are of full column rank, the sequence $\{(x^k, y^k, \lambda^k)\}$ generated by* (8.20) *globally converges to a pair of primal and dual optimal solutions (x^*, y^*) and λ^* of* (8.19) *from any starting point.*

Because both the nuclear norm penalty problem (8.13) and SDP relaxation (8.14) can be rewritten as the form of (8.19), we can apply ADMM to solve them.

8.4.1 ADMM for Nuclear Penalty Problem (8.13)

Note that the nuclear norm penalty problem (8.13) can be rewritten equivalently as

$$\begin{array}{ll} \min & -\operatorname{tr}(FY)+\rho\|Y\|_* \\ \text{s.t.} & X-Y=0, \\ & X\in\mathcal{C}, \end{array} \tag{8.21}$$

where $\mathcal{C} := \{X \in \mathbf{S}^{n^d \times n^d} \mid \mathrm{tr}\,(X) = 1,\ \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}\}$. A typical iteration of ADMM for solving (8.21) can be described as

$$\left\{\begin{array}{lcl} X^{k+1} & := & \mathrm{argmin}_{X\in\mathcal{C}} -\mathrm{tr}\,(FY^k) + \rho\|Y^k\|_* - \langle \Lambda^k, X - Y^k\rangle + \frac{1}{2\mu}\|X - Y^k\|_F^2 \\ Y^{k+1} & := & \mathrm{argmin}\ -\mathrm{tr}\,(FY) + \rho\|Y\|_* - \langle \Lambda^k, X^{k+1} - Y\rangle + \frac{1}{2\mu}\|X^{k+1} - Y\|_F^2 \\ \Lambda^{k+1} & := & \Lambda^k - (X^{k+1} - Y^{k+1})/\mu, \end{array}\right. \tag{8.22}$$

where Λ is the Lagrange multiplier associated with the equality constraint in (8.21) and $\mu > 0$ is a penalty parameter. Following Theorem 8.4.1, we know that the sequence $\{(X^k, Y^k, \Lambda^k)\}$ generated by (8.22) globally converges to a pair of primal and dual optimal solutions (X^*, Y^*) and Λ^* of (8.21) from any starting point.

Next we show that the two subproblems in (8.22) are both easy to solve. The first subproblem in (8.22) can be equivalently written as

$$X^{k+1} := \underset{X\in\mathcal{C}}{\mathrm{argmin}} \frac{1}{2}\|X - (Y^k + \mu\Lambda^k)\|_F^2, \tag{8.23}$$

i.e., the solution of the first subproblem in (8.22) corresponds to the projection of $Y^k + \mu\Lambda^k$ onto convex set $\mathcal{C}$. We will elaborate how to compute this projection in Section 8.4.2.

The second subproblem in (8.22) can be reduced to:

$$Y^{k+1} := \underset{Y}{\mathrm{argmin}} \quad \mu\rho\|Y\|_* + \frac{1}{2}\|Y - (X^{k+1} - \mu(\Lambda^k - F))\|_F^2. \tag{8.24}$$

This problem is known to have a closed-form solution that is given by the following so-called matrix shrinkage operation (see, e.g., [135]):

$$Y^{k+1} := U\mathrm{Diag}\,(\max\{\sigma - \mu\rho, 0\})V^\top,$$

where $U\mathrm{Diag}\,(\sigma)V^\top$ is the singular value decomposition of matrix $X^{k+1} - \mu(\Lambda^k - F)$.

8.4.2 The Projection

In this subsection, we study how to solve (8.23), i.e., how to compute the following projection for any given matrix $Z \in \mathbf{S}^{n^d \times n^d}$:

$$\begin{array}{ll} \min & \|X - Z\|_F^2 \\ \text{s.t.} & \mathrm{tr}\,(X) = 1, \\ & \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}}. \end{array} \tag{8.25}$$

For the sake of discussion, in the following we consider the equivalent tensor representation of (8.25):

$$\begin{array}{ll} \min & \|\mathcal{X}-\mathcal{Z}\|_F^2 \\ \text{s.t.} & \sum\limits_{k\in\mathbb{K}(n,d)} \frac{d!}{\prod_{j=1}^n k_j!}\mathcal{X}_{1^{2k_1}2^{2k_2}\cdots n^{2k_n}} = 1, \\ & \mathcal{X}\in\mathbf{S}^{n^{2d}}, \end{array} \tag{8.26}$$

where $\mathcal{X}=\boldsymbol{M}^{-1}(X)$, $\mathcal{Z}=\boldsymbol{M}^{-1}(Z)$, and the equality constraint is due to (8.9). Now we denote index set

$$\mathbf{I}=\left\{(i_1\cdots i_{2d})\in\pi(1^{2k_1}\cdots n^{2k_n}) \;\middle|\; k=(k_1,\cdots,k_n)\in\mathbb{K}(n,d)\right\}.$$

Then the first-order optimality conditions of Problem (8.26) imply

$$\begin{cases} 2\left(|\pi(i_1\cdots i_{2d})|\,\mathcal{X}_{i_1\cdots i_{2d}} - \sum\limits_{j_1\cdots j_{2d}\in\pi(i_1\cdots i_{2d})}\mathcal{Z}_{j_1\cdots j_{2d}}\right)=0, & \text{if } (i_1\cdots i_{2d})\notin\mathbf{I}, \\ 2\left(\frac{(2d)!}{\prod_{j=1}^n(2k_j)!}\mathcal{X}_{1^{2k_1}\cdots n^{2k_n}} - \sum\limits_{j_1\cdots j_{2d}\in\pi(1^{2k_1}\cdots n^{2k_n})}\mathcal{Z}_{j_1\cdots j_{2d}}\right)-\lambda\frac{(d)!}{\prod_{j=1}^n(k_j)!}=0, & \text{otherwise.} \end{cases}$$

Denote $\hat{\mathcal{Z}}$ to be the super-symmetric counterpart of tensor $\mathcal{Z}$, i.e.

$$\hat{\mathcal{Z}}_{i_1\cdots i_{2d}} = \sum_{j_1\cdots j_{2d}\in\pi(i_1\cdots i_{2d})}\frac{\mathcal{Z}_{j_1\cdots j_{2d}}}{|\pi(i_1\cdots i_{2d})|}$$

and $\alpha(k,d):=\left(\frac{(d)!}{\prod_{j=1}^n(k_j)!}\right)/\left(\frac{(2d)!}{\prod_{j=1}^n(2k_j)!}\right)$. Then due to the first-order optimality conditions of (8.26), the optimal solution $\mathcal{X}^*$ of Problem (8.26) satisfies

$$\begin{cases} \mathcal{X}^*_{i_1\cdots i_{2d}} &=& \hat{\mathcal{Z}}_{i_1\cdots i_{2d}}, & \text{if } (i_1\cdots i_{2d})\notin\mathbf{I}, \\ \mathcal{X}^*_{1^{2k_1}\cdots n^{2k_n}} &=& \frac{\lambda}{2}\,\alpha(k,d)+\hat{\mathcal{Z}}_{1^{2k_1}\cdots n^{2k_n}}, & \text{otherwise .} \end{cases} \tag{8.27}$$

Multiplying the second equality of (8.27) by $\frac{(d)!}{\prod_{j=1}^n(k_j)!}$ and summing the resulting equality over all $k=(k_1,\cdots,k_n)$ yield

$$\sum_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\mathcal{X}^*_{1^{2k_1}\cdots n^{2k_n}} = \frac{\lambda}{2}\sum_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\alpha(k,d)+\sum_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\hat{\mathcal{Z}}_{1^{2k_1}\cdots n^{2k_n}}.$$

It remains to determine λ. Noticing that $\mathcal{X}^*$ is a feasible solution for problem (8.26), we have $\sum\limits_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\mathcal{X}^*_{1^{2k_1}\cdots n^{2k_n}}=1$. As a result,

$$\lambda = 2\left(1-\sum_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\hat{\mathcal{Z}}_{1^{2k_1}\cdots n^{2k_n}}\right)\Big/\sum_{k\in\mathbb{K}(n,d)}\frac{(d)!}{\prod_{j=1}^n(k_j)!}\alpha(k,d),$$

and thus we derived $\mathcal{X}^*$ and $X^*=\boldsymbol{M}(\mathcal{X}^*)$ as the desired optimal solution for (8.25).

8.4.3 ADMM for SDP Relaxation (8.14)

Note that the SDP relaxation problem (8.14) can be formulated as

$$\begin{array}{ll} \min & -\mathrm{tr}\,(FY) \\ \text{s.t.} & \mathrm{tr}\,(X) = 1, \quad \boldsymbol{M}^{-1}(X) \in \mathbf{S}^{n^{2d}} \\ & X - Y = 0, \quad Y \succeq 0. \end{array} \tag{8.28}$$

A typical iteration of ADMM for solving (8.28) is

$$\left\{ \begin{array}{lcl} X^{k+1} & := & \mathrm{argmin}_{X \in \mathcal{C}} -\mathrm{tr}\,(FY^k) - \langle \Lambda^k, X - Y^k \rangle + \frac{1}{2\mu}\|X - Y^k\|_F^2 \\ Y^{k+1} & := & \mathrm{argmin}_{Y \succeq 0} -\mathrm{tr}\,(FY) - \langle \Lambda^k, X^{k+1} - Y \rangle + \frac{1}{2\mu}\|X^{k+1} - Y\|_F^2 \\ \Lambda^{k+1} & := & \Lambda^k - (X^{k+1} - Y^{k+1})/\mu, \end{array} \right. \tag{8.29}$$

where $\mu > 0$ is a penalty parameter. Following Theorem 8.4.1, we know that the sequence $\{(X^k, Y^k, \Lambda^k)\}$ generated by (8.29) globally converges to a pair of primal and dual optimal solutions (X^*, Y^*) and Λ^* of (8.28) from any starting point.

It is easy to check that the two subproblems in (8.29) are both relatively easy to solve. Specifically, the solution of the first subproblem in (8.29) corresponds to the projection of $Y^k + \mu\Lambda^k$ onto $\mathcal{C}$. The solution of the second problem in (8.29) corresponds to the projection of $X^{k+1} + \mu F - \mu\Lambda^k$ onto the positive semidefinite cone $Y \succeq 0$, i.e.,

$$Y^{k+1} := U\mathrm{Diag}\,(\max\{\sigma, 0\})U^\top,$$

where $U\mathrm{Diag}\,(\sigma)U^\top$ is the eigenvalue decomposition of matrix $X^{k+1} + \mu F - \mu\Lambda^k$.

8.5 Numerical Results

8.5.1 The ADMM for Convex Programs (8.13) and (8.14)

In this subsection, we report the results on using ADMM (8.22) to solve the nuclear norm penalty problem (8.13) and ADMM (8.29) to solve the SDP relaxation (8.14). For the nuclear norm penalty problem (8.13), we choose $\rho = 10$. For ADMM, we choose $\mu = 0.5$ and we terminate the algorithms whenever

$$\frac{\|X^k - X^{k-1}\|_F}{\|X^{k-1}\|_F} + \|X^k - Y^k\|_F \leq 10^{-6}.$$

We shall compare ADMM and CVX for solving (8.13) and (8.14), using the default solver of CVX – SeDuMi version 1.21. We report in Table 8.3 the results on randomly created problems with $d = 2$ and $n = 6, 7, 8, 9$. For each pair of d and n, we test ten randomly created examples. In Table 8.3, we use 'Inst.' to denote the number of the instance and use 'Iter.' to denote the number of iterations for ADMM to solve a random instance. We use 'Sol.Dif.' to denote the relative difference of the solutions obtained by ADMM and CVX, i.e., Sol.Dif. $= \frac{\|X_{ADMM} - X_{CVX}\|_F}{\max\{1, \|X_{CVX}\|_F\}}$, and we use 'Val.Dif.' to denote the relative difference of the objective values obtained by ADMM and CVX, i.e., Val.Dif. $= \frac{|v_{ADMM} - v_{CVX}|}{\max\{1, |v_{CVX}|\}}$. We use T_{ADMM} and T_{CVX} to denote the CPU times (in seconds) of ADMM and CVX, respectively. From Table 8.3 we see that, ADMM produced comparable solutions compared to CVX; however, ADMM were much faster than CVX, i.e., the interior point solver, especially for nuclear norm penalty problem (8.13). Note that when $n = 10$, ADMM was about 500 times faster than CVX for solving (8.13), and was about 8 times faster for solving (8.14).

In Table 8.4, we report the results on larger problems, i.e., $n = 14, 16, 18, 20$. Because it becomes time consuming to use CVX to solve the nuclear norm penalty problem (8.13) (our numerical tests showed that it took more than three hours to solve one instance of (8.13) for $n = 11$ using CVX), we compare the solution quality and objective value of the solution generated by ADMM for solving (8.13) with solution generated by CVX for solving SDP problem (8.14). From Table 8.4 we see that, the nuclear norm penalty problem (8.13) and the SDP problem (8.14) indeed produce the same solution as they are both close enough to the solution produced by CVX. We also see that using ADMM to solve (8.13) and (8.14) were much faster than using CVX to solve (8.14).

Inst. #	Nuclear Norm Penalty (8.13)					SDP (8.14)				
	Sol.Dif.	Val.Dif.	T_{ADMM}	Iter.	T_{CVX}	Sol.Dif.	Val.Dif.	T_{ADMM}	Iter.	T_{CVX}
Dimension $n = 6$										
1	1.77e-04	3.28e-06	1.16	464	18.50	1.01e-04	2.83e-06	0.50	367	1.98
2	1.25e-04	3.94e-07	0.71	453	13.43	4.99e-05	3.78e-06	0.38	355	1.68
3	1.56e-04	2.36e-07	0.89	478	12.20	4.59e-05	3.51e-06	0.39	370	1.33
4	3.90e-05	6.91e-07	0.59	475	14.10	8.00e-05	9.57e-07	0.44	364	2.63
5	1.49e-04	3.69e-06	0.58	459	15.08	4.74e-05	3.18e-06	0.60	355	1.98
6	8.46e-05	3.92e-06	1.07	463	13.23	1.02e-04	2.68e-07	0.76	362	1.46
7	5.59e-05	4.12e-06	0.86	465	12.62	4.91e-05	4.75e-06	0.37	344	1.54
8	5.24e-05	3.95e-06	0.61	462	14.07	1.63e-05	2.97e-06	0.55	368	1.90
9	9.30e-05	3.05e-06	0.85	471	11.41	1.05e-04	2.90e-06	0.39	380	1.39
10	1.36e-04	3.89e-08	0.56	465	11.04	3.38e-05	3.11e-06	0.30	319	1.69
Dimension $n = 7$										
1	1.59e-04	4.62e-07	1.23	600	65.73	1.14e-04	4.09e-06	0.82	453	2.60
2	9.11e-05	3.93e-07	1.02	593	68.65	8.24e-05	2.87e-09	0.79	474	2.51
3	2.61e-04	4.19e-06	1.07	609	66.08	6.83e-05	4.01e-06	0.78	480	2.53
4	1.12e-04	4.44e-06	1.07	590	65.21	6.02e-05	3.88e-06	0.86	480	2.50
5	1.22e-04	4.34e-06	1.10	614	57.40	9.15e-05	4.15e-07	0.81	487	2.57
6	1.44e-04	8.81e-08	1.06	599	60.89	4.51e-05	4.46e-06	0.77	466	2.44
7	1.93e-04	3.81e-06	1.08	590	66.09	1.19e-04	2.82e-07	0.62	389	2.54
8	1.53e-04	4.59e-06	1.09	594	59.98	2.76e-05	3.73e-06	0.75	463	2.61
9	1.41e-04	4.29e-08	1.06	616	78.20	3.29e-04	4.21e-06	0.69	443	2.57
10	1.51e-04	3.94e-06	0.83	501	75.58	1.23e-04	3.52e-06	0.78	454	2.63
Dimension $n = 8$										
1	2.86e-04	5.10e-06	2.15	728	342.25	1.12e-04	4.52e-06	1.59	592	5.34
2	2.76e-04	3.95e-07	2.07	739	303.75	8.17e-05	4.78e-06	1.81	591	5.02
3	9.29e-05	4.78e-06	7.74	2864	333.46	2.57e-05	5.00e-06	7.20	2746	4.75
4	3.21e-04	4.65e-06	2.01	715	337.57	9.86e-05	4.01e-06	1.47	512	5.00
5	1.26e-04	7.05e-07	1.92	746	335.63	7.41e-05	4.36e-06	1.68	607	4.92
6	1.32e-04	1.63e-07	2.12	745	336.35	7.80e-05	5.00e-06	1.44	550	5.29
7	3.49e-04	7.19e-07	2.00	739	309.76	6.33e-05	4.55e-07	1.54	582	5.03
8	4.55e-05	4.72e-07	2.13	744	316.74	3.59e-05	7.27e-07	1.59	600	5.02
9	5.60e-04	4.99e-06	2.06	759	336.10	4.19e-05	4.97e-06	1.46	569	6.00
10	2.65e-04	1.36e-07	2.46	746	382.20	8.00e-05	4.14e-06	1.86	606	5.98
Dimension $n = 9$										
1	1.41e-04	1.35e-07	4.35	910	1370.60	7.29e-05	4.78e-06	3.26	715	12.61
2	1.83e-04	5.77e-06	3.60	872	1405.46	1.77e-04	4.72e-06	2.86	732	9.63
3	4.00e-04	4.85e-06	3.24	807	1709.30	3.12e-04	8.28e-07	2.73	702	9.99
4	3.34e-04	1.36e-07	3.06	747	1445.57	6.13e-05	3.19e-07	2.91	707	10.19
5	2.63e-04	5.43e-06	3.62	904	1307.60	2.34e-05	4.68e-06	2.82	729	10.20
6	8.01e-05	9.01e-08	3.78	906	1353.45	9.33e-05	5.37e-06	2.49	597	9.31
7	2.30e-04	5.16e-06	3.77	900	1434.71	8.14e-05	5.68e-06	2.75	676	9.52
8	3.27e-04	5.45e-06	3.71	908	1314.14	1.98e-05	5.10e-06	2.91	730	9.98
9	9.53e-05	5.56e-06	3.66	888	1575.16	1.69e-04	4.82e-06	2.85	714	9.64
10	2.73e-04	2.16e-07	4.50	1136	1628.80	2.73e-05	4.98e-06	3.39	882	9.90

Table 8.3: Comparison of CVX and ADMM for small-scale problems

Inst. #	NNP				SDP				
	Sol.Dif.$_{DS}$	Val.Dif.$_{DS}$	T_{ADMM}	Iter.	Sol.Dif.	Val.Dif.	T_{ADMM}	Iter.	T_{CVX}
Dimension $n = 14$									
1	4.61e-04	8.41e-06	36.85	1913	4.61e-04	8.35e-06	37.00	1621	158.21
2	4.02e-04	2.94e-07	39.52	1897	4.02e-04	7.93e-06	39.65	1639	167.89
3	1.62e-04	2.68e-08	37.21	1880	1.62e-04	8.23e-06	34.36	1408	213.04
4	4.92e-04	7.74e-06	45.15	1918	4.92e-04	4.70e-07	59.84	1662	202.95
5	8.56e-04	8.15e-06	34.93	1674	8.56e-04	8.14e-06	38.15	1588	194.01
6	3.99e-05	4.05e-07	34.41	1852	4.08e-05	7.48e-06	32.28	1411	186.99
7	7.98e-05	7.90e-06	38.11	1839	7.94e-05	3.76e-08	40.81	1555	191.76
8	1.50e-04	8.10e-06	38.29	1990	1.50e-04	8.30e-06	34.10	1543	164.13
9	1.35e-04	8.54e-06	34.58	1874	1.35e-04	2.62e-07	30.33	1387	171.77
10	5.50e-04	8.59e-06	37.28	1825	5.50e-04	7.71e-06	35.85	1567	169.51
Dimension $n = 16$									
1	5.22e-05	9.00e-06	125.24	2359	5.21e-05	9.45e-06	102.85	2035	582.19
2	1.02e-04	3.37e-07	92.37	2244	1.02e-04	9.11e-06	63.02	1427	606.70
3	2.02e-05	5.97e-07	96.21	2474	2.01e-05	4.40e-07	83.92	1910	566.92
4	8.53e-05	9.27e-06	90.83	2323	8.54e-05	9.59e-06	93.44	2048	560.54
5	2.14e-04	9.19e-06	86.22	2359	2.14e-04	2.19e-07	80.06	1961	523.15
6	3.12e-04	9.29e-06	88.82	2304	3.12e-04	8.58e-06	88.31	2042	498.55
7	9.69e-05	9.12e-06	88.29	2431	9.65e-05	2.86e-07	88.05	2067	520.82
8	3.34e-04	1.00e-05	85.32	2271	3.34e-04	8.53e-06	85.04	2043	515.85
9	2.61e-04	9.01e-06	93.13	2475	2.61e-04	9.12e-06	88.85	2034	505.71
10	2.06e-04	3.45e-07	103.92	2813	2.05e-04	1.01e-05	94.41	2269	527.50
Dimension $n = 18$									
1	2.70e-04	1.01e-05	172.97	2733	2.70e-04	1.87e-07	168.91	2323	1737.94
2	8.17e-04	1.11e-05	184.70	2970	8.17e-04	1.99e-07	168.83	2365	1549.10
3	1.07e-04	3.22e-08	183.72	2920	1.07e-04	1.14e-05	169.64	2456	1640.04
4	5.16e-04	1.01e-05	182.40	2958	5.16e-04	1.02e-05	174.72	2442	1636.86
5	9.48e-04	1.03e-05	184.69	3039	9.48e-04	1.04e-05	170.68	2441	1543.41
6	1.67e-04	1.03e-05	171.71	2845	1.67e-04	9.96e-06	182.37	2553	1633.55
7	4.87e-05	3.77e-07	180.64	2883	4.87e-05	2.79e-07	187.56	2545	1638.38
8	8.28e-05	1.07e-05	178.35	2904	8.28e-05	1.04e-05	181.57	2542	1641.56
9	2.45e-04	1.06e-07	174.82	2902	2.45e-04	9.97e-06	152.58	2127	1735.26
10	9.58e-05	7.61e-07	191.06	2872	9.66e-05	1.11e-05	183.29	2480	1642.33
Dimension $n = 20$									
1	1.23e-03	6.98e-08	414.62	3415	1.23e-03	4.21e-08	388.36	2810	6116.02
2	7.93e-04	1.24e-05	401.54	3383	7.93e-04	1.14e-05	347.27	2689	6182.56
3	3.11e-04	1.21e-05	426.91	3498	3.11e-04	1.21e-05	399.92	2845	6808.99
4	7.16e-05	6.99e-07	397.69	3312	7.40e-05	1.18e-05	366.82	2758	7701.91
5	6.24e-04	1.19e-05	435.05	3564	6.25e-04	1.20e-05	419.23	2903	7419.43
6	1.09e-04	1.20e-05	393.25	3376	1.09e-04	1.15e-05	397.43	2869	8622.19
7	4.58e-04	3.21e-05	429.38	3536	4.58e-04	3.20e-05	422.72	2938	9211.37
8	6.15e-04	1.11e-05	273.33	2330	6.15e-04	7.14e-07	205.49	1511	5166.66
9	4.92e-04	1.16e-05	344.99	3017	4.92e-04	2.32e-07	259.18	1896	5063.00
10	3.45e-004	2.56e-004	395.63	3357	1.14e-005	4.36e-007	359.13	2713	6559.39

Table 8.4: Comparison of CVX and ADMM for large-scale problems

8.5.2 Comparison with SOS and MBI

Based on the results of the above tests, we may conclude that it is most efficient to solve the SDP relaxation by ADMM. In this subsection, we compare this approach with two competing methods: one is based on the Sum of Squares (SOS) approach (Lasserre [27, 28] and Parrilo [29, 30]), and the other one is the Maximum Block Improvement

(MBI) method proposed by Chen *et al.* [79].

Theoretically speaking, the SOS can solve any general polynomial problems to any given accuracy, but it requires to solve a sequence of (possibly large) semidefinite programs, which limits the applicability of the method to solve large size problems. Henrion et al. [136] developed a specialized Matlab toolbox known as GloptiPoly 3 based on SOS approach, which will be used in our tests. The MBI is tailored for multi-block optimization problem, and the polynomial optimization can be treated as multi-block problems, to which MBI can be applied. As we mentioned before, MBI aims to finding a stationary point, which may or may not be globally optimal.

In Table 8.5 we report the results using ADMM to solve SDP relaxation of PCA problem and compare them with the results of applying the SOS method as well as the MBI method for the same problem. When using the MBI, as suggested in [79], we actually work on an equivalent problem of (8.6): $\max_{\|x\|=1} \mathcal{F}(\underbrace{x, \cdots, x}_{2d}) + 6(x^\top x)^d$, where the equivalence is due to the constraint $\|x\| = 1$. This transformation can help the MBI avoid getting trapped in a local minimum.

We use 'Val.' to denote the objective value of the solution, 'Status' to denote optimal status of GloptiPoly 3, i.e., Status = 1 means GloptiPoly 3 successfully identified the optimality of current solution, 'Sol.R.' to denote the solution rank returned by SDP relaxation and thanks to the previous discussion 'Sol.R.=1' means the current solution is already optimal. From Table 8.5, we see that the MBI is the fastest among all the methods but usually cannot guarantee global optimality, while GloptiPoly 3 is very time consuming but can globally solve most instances. Note that when $n = 20$, our ADMM was about 30 times faster than GloptiPoly 3. Moreover, for some instances GloptiPoly 3 cannot identify the optimality even though the current solution is actually already optimal (see instance 9 with $n = 16$ and instance 3 with $n = 18$).

Inst. #	MBI		GLP			SDP by ADMM		
	Val.	Time	Val.	Time	Status	Val.	Time	Sol.R.
Dimension $n = 14$								
1	5.17	0.23	5.28	143.14	1	5.28	14.29	1
2	5.04	0.22	5.65	109.65	1	5.65	32.64	1
3	5.08	0.13	5.80	119.48	1	5.80	34.30	1
4	5.94	0.16	5.95	100.39	1	5.95	30.64	1
5	4.74	0.48	5.88	122.19	1	5.88	33.13	1
6	5.68	0.54	6.38	122.44	1	6.38	33.30	1
7	4.61	0.12	5.91	104.68	1	5.91	30.17	1
8	5.68	0.23	6.31	141.52	1	6.31	41.73	1
9	5.93	0.22	6.40	102.73	1	6.40	37.32	1
10	5.09	0.36	6.03	114.35	1	6.03	35.68	1
Dimension $n = 16$								
1	6.52	0.45	6.74	420.10	1	6.74	91.80	1
2	5.51	1.21	5.93	428.10	1	5.93	83.90	1
3	5.02	0.30	6.44	393.16	1	6.44	90.16	1
4	5.60	0.32	6.48	424.07	1	6.48	90.67	1
5	5.78	0.36	6.53	431.44	1	6.53	95.48	1
6	5.23	0.26	6.42	437.58	1	6.42	98.19	1
7	6.11	0.24	6.23	406.16	1	6.23	89.21	1
8	5.92	0.51	6.39	416.58	1	6.39	89.75	1
9	5.47	0.28	6.00	457.29	0	6.00	77.56	1
10	4.95	0.35	6.32	367.26	1	6.32	80.38	1
Dimension $n = 18$								
1	6.16	0.57	7.38	1558.00	1	7.38	199.44	1
2	5.94	0.25	6.65	1388.45	1	6.65	190.52	1
3	7.42	0.22	7.42	1500.05	0	7.42	193.27	1
4	5.85	0.94	7.21	1481.34	1	7.21	195.02	1
5	7.35	0.43	7.35	1596.00	1	7.35	117.44	1
6	5.91	1.05	6.79	1300.82	1	6.78	193.36	1
7	5.80	0.85	6.84	1433.50	1	6.84	182.58	1
8	5.72	0.54	6.96	1648.63	1	6.96	231.88	1
9	6.15	0.17	7.07	1453.82	1	7.07	212.50	1
10	6.01	1.11	6.89	1432.06	1	6.89	199.26	1
Dimension $n = 20$								
1	5.95	0.39	7.40	8981.97	1	7.40	429.64	1
2	6.13	2.14	6.93	9339.06	1	6.93	355.25	1
3	6.37	2.49	6.68	9629.04	1	6.68	418.11	1
4	6.23	1.14	6.87	10148.21	1	6.87	404.18	1
5	6.62	1.66	7.72	11079.94	1	7.72	326.44	1
6	6.81	1.26	7.46	10609.65	1	7.46	415.69	1
7	7.80	1.02	7.80	9723.37	1	7.80	430.76	1
8	6.03	0.95	7.02	12755.35	1	7.02	416.00	1
9	7.80	0.61	7.80	12353.47	1	7.80	430.45	1
10	7.47	0.89	7.47	11629.12	1	7.47	375.52	1

Table 8.5: Comparison SDP Relaxation by ADMM with GloptiPoly 3 and MBI.

8.6 Extensions

In this section, we show how to extend the results in the previous sections for super-symmetric tensor PCA problem to tensors that are not super-symmetric.

8.6.1 Biquadratic Tensor PCA

A closely related problem to the tensor PCA problem (8.6) is the following biquadratic PCA problem:

$$\begin{array}{ll} \max & \mathcal{G}(x,y,x,y) \\ \text{s.t.} & x \in \mathbb{R}^n, \|x\| = 1, \\ & y \in \mathbb{R}^m, \|y\| = 1, \end{array} \tag{8.30}$$

where $\mathcal{G}$ is a *partial-symmetric* tensor defined in Definition 4.4.8. Various approximation algorithms for biquadratic PCA problem have been studied in [32]. Problem (8.30) arises from the strong ellipticity condition problem in solid mechanics and the entanglement problem in quantum physics; see [32] for more applications of biquadratic PCA problem.

We can unfold a partial-symmetric tensor $\mathcal{G}$ in the manner of Definition 4.1.2. It is easy to check that for given vectors $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, $a \otimes b \otimes a \otimes b \in \overrightarrow{\mathbf{S}}^{(nm)^2}$, and it is of rank-one in the sense of partial symmetric CP rank.

Since $\operatorname{tr}(xy^\top yx^\top) = x^\top xy^\top y = 1$, by letting $\mathcal{X} = x \otimes y \otimes x \otimes y$, problem (8.30) is equivalent to

$$\begin{array}{ll} \max & \mathcal{G} \bullet \mathcal{X} \\ \text{s.t.} & \sum\limits_{i,j} \mathcal{X}_{ijij} = 1, \\ & \mathcal{X} \in \overrightarrow{\mathbf{S}}^{(nm)^2},\ \operatorname{rank}_{PCP}(\mathcal{X}) = 1. \end{array}$$

In the following, we group variables x and y together and treat $x \otimes y$ as a long vector by stacking its columns. Denote $X = \boldsymbol{M}(\mathcal{X})$ and $G = \boldsymbol{M}(\mathcal{G})$. Then, we end up with a matrix version of the above tensor problem:

$$\begin{array}{ll} \max & \operatorname{tr}(GX) \\ \text{s.t.} & \operatorname{tr}(X) = 1, \quad X \succeq 0, \\ & \boldsymbol{M}^{-1}(X) \in \overrightarrow{\mathbf{S}}^{(nm)^2},\ \operatorname{rank}(X) = 1. \end{array} \tag{8.31}$$

According to the rank-one equivalence theorem (4.4.10), the above two problems are actually equivalent. Moreover, by using the similar argument in Theorem 8.3.1, we can show that the following SDP relaxation of (8.31) has a good chance to get a low rank solution.

$$\begin{array}{ll} \max & \operatorname{tr}(GX) \\ \text{s.t.} & \operatorname{tr}(X) = 1, \quad X \succeq 0, \\ & \boldsymbol{M}^{-1}(X) \in \overrightarrow{\mathbf{S}}^{(nm)^2}. \end{array} \tag{8.32}$$

Therefore, we used the same ADMM to solve (8.32). The frequency of returning rank-one solutions for randomly created examples is reported in Table 8.6. As in Table 8.1 and Table 8.2, we tested 100 random instances for each (n, m) and report the number of instances that produced rank-one solutions. We also report the average CPU time (in seconds) using ADMM to solve the problems. Table 8.6 shows that the SDP relaxation (8.32) can give a rank-one solution for *most* randomly created instances, and thus is likely to solve the original problem (8.30) to optimality.

Dim (n, m)	rank-1	CPU
(4,4)	100	0.12
(4,6)	100	0.25
(6,6)	100	0.76
(6,8)	100	1.35
(8,8)	98	2.30
(8,10)	100	3.60
(10,10)	96	5.77

Table 8.6: Frequency of problem (8.32) having rank-one solution

8.6.2 Trilinear Tensor PCA

Now let us consider a highly non-symmetric case: trilinear PCA.

$$\begin{array}{ll} \max & \mathcal{F}(x, y, z) \\ \text{s.t.} & x \in \mathbb{R}^n, \|x\| = 1, \\ & y \in \mathbb{R}^m, \|y\| = 1, \\ & z \in \mathbb{R}^\ell, \ \|z\| = 1, \end{array} \tag{8.33}$$

where $\mathcal{F} \in \mathbb{R}^{n \times m \times \ell}$ is any 3-rd order tensor and $n \leq m \leq \ell$.

Recently, trilinear PCA problem was found to be very useful in many practical problems. For instance, Wang and Ahuja [101] proposed a tensor rank-one decomposition algorithm to compress image sequence, where they essentially solve a sequence of trilinear PCA problems.

By the Cauchy-Schwarz inequality, the problem (8.33) is equivalent to

$$\begin{array}{lll} \max & \|\mathcal{F}(x,y,\cdot)\| & \\ \text{s.t.} & x\in\mathbb{R}^n, \|x\|=1, & \Longleftrightarrow \\ & y\in\mathbb{R}^m, \|y\|=1, & \end{array} \quad \begin{array}{lll} \max & \|\mathcal{F}(x,y,\cdot)\|^2 & \\ \text{s.t.} & x\in\mathbb{R}^n, \|x\|=1, & \\ & y\in\mathbb{R}^m, \|y\|=1. & \end{array}$$

We further notice

$$\begin{aligned} \|\mathcal{F}(x,y,\cdot)\|^2 &= \mathcal{F}(x,y,\cdot)^\top \mathcal{F}(x,y,\cdot) = \sum_{k=1}^{\ell} \mathcal{F}_{ijk}\,\mathcal{F}_{uvk}\,x_i y_j x_u y_v \\ &= \sum_{k=1}^{\ell} \mathcal{F}_{ivk}\,\mathcal{F}_{ujk}\,x_i y_v x_u y_j = \sum_{k=1}^{\ell} \mathcal{F}_{ujk}\,\mathcal{F}_{ivk}\,x_u y_j x_i y_v. \end{aligned}$$

Therefore, we can find a partial symmetric tensor $\mathcal{G}$ with

$$\mathcal{G}_{ijuv} = \sum_{k=1}^{\ell} \left(\mathcal{F}_{ijk}\,\mathcal{F}_{uvk} + \mathcal{F}_{ivk}\,\mathcal{F}_{ujk} + \mathcal{F}_{ujk}\,\mathcal{F}_{ivk}\right)/3, \quad \forall\, i,j,u,v,$$

such that $\|\mathcal{F}(x,y,\cdot)\|^2 = \mathcal{G}\,(x,y,x,y)$. Hence, the trilinear problem (8.33) can be equivalently formulated in the form of problem (8.30), which can be solved by the method proposed in the previous subsection.

8.6.3 Quadrilinear Tensor PCA

In this subsection, we consider the following quadrilinear PCA problem:

$$\begin{array}{ll} \max & \mathcal{F}(x^1,x^2,x^3,x^4) \\ \text{s.t.} & x^i\in\mathbb{R}^{n_i}, \|x^i\|=1,\ \forall\, i=1,2,3,4, \end{array} \tag{8.34}$$

where $\mathcal{F}\in\mathbb{R}^{n_1\times\cdots\times n_4}$ with $n_1\le n_3\le n_2\le n_4$. Let us first convert the quadrilinear function $\mathcal{F}(x^1,x^2,x^3,x^4)$ to a biquadratic function $\mathcal{T}\left(\begin{smallmatrix}x^1\\x^3\end{smallmatrix},\begin{smallmatrix}x^2\\x^4\end{smallmatrix},\begin{smallmatrix}x^1\\x^3\end{smallmatrix},\begin{smallmatrix}x^2\\x^4\end{smallmatrix}\right)$ with $\mathcal{T}$ being partial symmetric. To this end, we first construct $\mathcal{G}$ with

$$\mathcal{G}_{i_1,i_2,n+i_3,n+i_4} = \begin{cases} \mathcal{F}_{j_1j_2j_3j_4}, & \text{if } 1\le i_k\le n_k \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, we have $\mathcal{F}(x^1,x^2,x^3,x^4) = \mathcal{G}\left(\begin{smallmatrix}x^1\\x^3\end{smallmatrix},\begin{smallmatrix}x^2\\x^4\end{smallmatrix},\begin{smallmatrix}x^1\\x^3\end{smallmatrix},\begin{smallmatrix}x^2\\x^4\end{smallmatrix}\right)$. Then we can further partial-symmetrize $\mathcal{G}$ and the desired tensor $\mathcal{T}$ is as follows,

$$\mathcal{T}_{i_1i_2i_3i_4} = \frac{1}{4}\left(\mathcal{G}_{i_1i_2i_3i_4} + \mathcal{G}_{i_1i_4i_3i_2} + \mathcal{G}_{i_3i_2i_1i_4} + \mathcal{G}_{i_3i_4i_1i_2}\right) \quad \forall\, i_1,i_2,i_3,i_4,$$

satisfying $\mathcal{T}\left(\begin{smallmatrix} x^1, & x^2, & x^1, & x^2 \\ x^3, & x^4, & x^3, & x^4 \end{smallmatrix}\right) = \mathcal{G}\left(\begin{smallmatrix} x^1, & x^2, & x^1, & x^2 \\ x^3, & x^4, & x^3, & x^4 \end{smallmatrix}\right)$. Therefore, problem (8.34) is now reformulated as a biquadratic problem:

$$\begin{array}{ll} \max & \mathcal{T}\left(\begin{smallmatrix} x^1, & x^2, & x^1, & x^2 \\ x^3, & x^4, & x^3, & x^4 \end{smallmatrix}\right) \\ \text{s.t.} & x^i \in \mathbb{R}^{n_i}, \|x^i\| = 1, \ \forall\, i = 1, \ldots, 4. \end{array} \tag{8.35}$$

Moreover, we can show that the above problem is actually a biquadratic problem in the form of (8.30).

Proposition 8.6.1. *Suppose $\mathcal{T}$ is a fourth order partial symmetric tensor. Then problem* (8.35) *is equivalent to*

$$\begin{array}{ll} \max & \mathcal{T}\left(\begin{smallmatrix} x^1, & x^2, & x^1, & x^2 \\ x^3, & x^4, & x^3, & x^4 \end{smallmatrix}\right) \\ \text{s.t.} & \sqrt{\|x^1\|^2 + \|x^3\|^2} = \sqrt{2}, \\ & \sqrt{\|x^2\|^2 + \|x^4\|^2} = \sqrt{2}. \end{array} \tag{8.36}$$

Proof. It is obvious that (8.36) is a relaxation of (8.35). To further prove that the relaxation (8.36) is tight, we assume $(\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4)$ is optimal to (8.36). Then $\mathcal{T}\left(\begin{smallmatrix} \hat{x}^1, & \hat{x}^2, & \hat{x}^1, & \hat{x}^2 \\ \hat{x}^3, & \hat{x}^4, & \hat{x}^3, & \hat{x}^4 \end{smallmatrix}\right) = \mathcal{F}(\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4) > 0$, and so $\hat{x}^i \neq 0$ for all i. Moreover, notice that

$$\sqrt{\|\hat{x}^1\|\|\hat{x}^3\|} \leq \sqrt{\frac{\|\hat{x}^1\|^2 + \|\hat{x}^3\|^2}{2}} = 1 \text{ and } \sqrt{\|\hat{x}^2\|\|\hat{x}^4\|} \leq \sqrt{\frac{\|\hat{x}^2\|^2 + \|\hat{x}^4\|^2}{2}} = 1.$$

Thus

$$\begin{aligned} \mathcal{T}\left(\begin{smallmatrix} \frac{\hat{x}^1}{\|\hat{x}^1\|}, & \frac{\hat{x}^2}{\|\hat{x}^2\|}, & \frac{\hat{x}^1}{\|\hat{x}^1\|}, & \frac{\hat{x}^2}{\|\hat{x}^2\|} \\ \frac{\hat{x}^3}{\|\hat{x}^3\|}, & \frac{\hat{x}^4}{\|\hat{x}^4\|}, & \frac{\hat{x}^3}{\|\hat{x}^3\|}, & \frac{\hat{x}^4}{\|\hat{x}^4\|} \end{smallmatrix}\right) &= \mathcal{F}\left(\frac{\hat{x}^1}{\|\hat{x}^1\|}, \frac{\hat{x}^2}{\|\hat{x}^2\|}, \frac{\hat{x}^3}{\|\hat{x}^3\|}, \frac{\hat{x}^4}{\|\hat{x}^4\|}\right) \\ &= \frac{\mathcal{F}(\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4)}{\|\hat{x}^1\|\|\hat{x}^2\|\|\hat{x}^3\|\|\hat{x}^4\|} \\ &\geq \mathcal{F}(\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4) \\ &= \mathcal{T}\left(\begin{smallmatrix} \hat{x}^1, & \hat{x}^2, & \hat{x}^1, & \hat{x}^2 \\ \hat{x}^3, & \hat{x}^4, & \hat{x}^3, & \hat{x}^4 \end{smallmatrix}\right). \end{aligned}$$

To summarize, we have found a feasible solution $\left(\frac{\hat{x}^1}{\|\hat{x}^1\|}, \frac{\hat{x}^2}{\|\hat{x}^2\|}, \frac{\hat{x}^3}{\|\hat{x}^3\|}, \frac{\hat{x}^4}{\|\hat{x}^4\|}\right)$ of (8.35), which is optimal to its relaxation (8.36) and thus this relaxation is tight. □

By letting $y = \begin{pmatrix} x^1 \\ x^3 \end{pmatrix}$, $z = \begin{pmatrix} x^2 \\ x^4 \end{pmatrix}$ and using some scaling technique, we can see that problem (8.36) share the same solution with

$$\begin{array}{ll} \max & \mathcal{T}(y, z, y, z) \\ \text{s.t.} & \|y\| = 1, \\ & \|z\| = 1, \end{array}$$

which was studied in Subsection 8.6.1.

8.6.4 Even Order Multilinear PCA

The above discussion can be extended to the even order multilinear PCA problem:

$$\begin{array}{ll} \max & \mathcal{F}(x^1, x^2, \cdots, x^{2d}) \\ \text{s.t.} & x^i \in \mathbb{R}^{n_i}, \|x^i\| = 1, \ \forall\, i = 1, 2, \ldots, 2d, \end{array} \tag{8.37}$$

where $\mathcal{F} \in \mathbb{R}^{n^1 \times \cdots \times n^{2d}}$. An immediate relaxation of (8.37) is the following

$$\begin{array}{ll} \max & \mathcal{F}(x^1, x^2, \cdots, x^{2d}) \\ \text{s.t.} & x^i \in \mathbb{R}^{n_i}, \sqrt{\sum\limits_{i}^{2d} \|x^i\|^2} = \sqrt{2d}. \end{array} \tag{8.38}$$

The following result shows that these two problems are actually equivalent.

Proposition 8.6.2. *It holds that problem* (8.37) *is equivalent to* (8.38).

Proof. It suffices to show that relaxation (8.38) is tight. To this end, suppose $(\hat{x}^1, \cdots, \hat{x}^{2d})$ is an optimal solution of (8.38). Then $\mathcal{F}(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^{2d}) > 0$ and so $\hat{x}^i \neq 0$ for $i = 1, \ldots, 2d$. We also notice

$$\sqrt{\left(\prod_{i=1}^{2d} \|\hat{x}_i\|^2\right)^{\frac{1}{2d}}} \leq \sqrt{\sum_{i}^{2d} \|\hat{x}^i\|^2 / 2d} = 1.$$

Consequently, $\prod\limits_{i=1}^{2d} \|\hat{x}_i\| \leq 1$ and

$$\mathcal{F}\left(\frac{\hat{x}^1}{\|\hat{x}^1\|}, \frac{\hat{x}^2}{\|\hat{x}^2\|}, \cdots, \frac{\hat{x}^{2d}}{\|\hat{x}^{2d}\|}\right) = \frac{\mathcal{F}(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^{2d})}{\prod\limits_{i=1}^{2d} \|\hat{x}_i\|} \geq \mathcal{F}(\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^{2d}).$$

Therefore, we have found a feasible solution $\left(\frac{\hat{x}^1}{\|\hat{x}^1\|}, \frac{\hat{x}^2}{\|\hat{x}^2\|}, \cdots, \frac{\hat{x}^{2d}}{\|\hat{x}^{2d}\|}\right)$ of (8.37), which is optimal to (8.38) implying that the relaxation is tight. □

We now focus on (8.38). Based on $\mathcal{F}$, we can construct a larger tensor $\mathcal{G}$ as follows

$$\mathcal{G}_{i_1\cdots i_{2d}} = \begin{cases} \mathcal{F}_{j_1\cdots j_{2d}}, & \text{if } 1+\sum_{\ell=1}^{k-1} n_\ell \le i_k \le \sum_{\ell=1}^{k} n_\ell \text{ and } j_k = i_k - \sum_{\ell=1}^{k-1} n_\ell \\ 0, & \text{otherwise.} \end{cases}$$

By this construction, we have

$$\mathcal{F}(x^1, x^2, \cdots, x^{2d}) = \mathcal{G}(\underbrace{y, \cdots, y}_{2d})$$

with $y = ((x^1)^\top, (x^2)^\top, \cdots, (x^{2d})^\top)^\top$. We can further symmetrize $\mathcal{G}$ and find a super-symmetric $\mathcal{T}$ such that

$$\mathcal{T}_{i_1\cdots i_{2d}} := \frac{1}{|\pi(i_1\cdots i_{2d})|} \sum_{j_1\cdots j_{2d}\in\pi(i_1\cdots i_{2d})} \mathcal{G}_{j_1\cdots j_{2d}}, \quad \forall\, 1 \le i_1, \cdots, i_{2d} \le \sum_{\ell=1}^{2d} n_\ell,$$

and

$$\mathcal{T}(\underbrace{y, \cdots, y}_{2d}) = \mathcal{G}(\underbrace{y, \cdots, y}_{2d}) = \mathcal{F}(x^1, x^2, \cdots, x^{2d}).$$

Therefore, problem (8.38) is equivalent to

$$\begin{array}{ll} \max & \mathcal{T}(\underbrace{y, \cdots, y}_{2d}) \\ \text{s.t.} & \|y\| = \sqrt{2d}, \end{array}$$

which is further equivalent to

$$\begin{array}{ll} \max & \mathcal{T}(\underbrace{y, \cdots, y}_{2d}) \\ \text{s.t.} & \|y\| = 1. \end{array}$$

Thus the methods we developed for solving (8.6) can be applied to solve (8.37).

8.6.5 Odd Degree Tensor PCA

The last problem studied in this section is the following odd degree tensor PCA problem:

$$\begin{array}{ll} \max & \mathcal{F}(\underbrace{x, \cdots, x}_{2d+1}) \\ \text{s.t.} & \|x\| = 1, \end{array} \tag{8.39}$$

where $\mathcal{F}$ is a $(2d+1)$-th order super-symmetric tensor. As the degree is odd,

$$\max_{\|x\|=1} \mathcal{F}(\underbrace{x, \cdots, x}_{2d+1}) = \max_{\|x\|=1} |\mathcal{F}(\underbrace{x, \cdots, x}_{2d+1})| = \max_{\|x^i\|_2^2=1,\, i=1,\ldots,2d+1} |\mathcal{F}(x^1, \cdots, x^{2d+1})|,$$

where the last identity is due to Corollary 4.2 in [79]. The above formula combined with the fact that

$$\max_{\|x\|=1} |\mathcal{F}(\underbrace{x, \cdots, x}_{2d+1})| \le \max_{\|x\|=1,\, \|y\|=1} |\mathcal{F}(\underbrace{x, \cdots, x}_{2d}, y)| \le \max_{\|x^i\|=1,\, i=1,\ldots,2d+1} |\mathcal{F}(x^1, \cdots, x^{2d+1})|$$

implies

$$\max_{\|x\|=1} \mathcal{F}(\underbrace{x, \cdots, x}_{2d+1}) = \max_{\|x\|=1,\, \|y\|=1} |\mathcal{F}(\underbrace{x, \cdots, x}_{2d}, y)| = \max_{\|x\|=1,\, \|y\|=1} \mathcal{F}(\underbrace{x, \cdots, x}_{2d}, y).$$

By using the similar technique as in Subsection 8.6.2, problem (8.39) is equivalent to an even order tensor PCA problem:

$$\begin{array}{ll} \max & \mathcal{G}(\underbrace{x, \cdots, x}_{4d}) \\ \text{s.t.} & \|x\| = 1, \end{array}$$

where $\mathcal{G}$ is super-symmetric with

$$\mathcal{G}_{i_1, \cdots, i_{4d}} = \frac{1}{|\pi(i_1 \cdots i_{4d})|} \sum_{k=1}^{n} \left(\sum_{j_1 \cdots j_{4d} \in \pi(i_1 \cdots i_{4d})} \mathcal{F}_{i_1 \cdots i_{2d} k}\, \mathcal{F}_{i_{2d+1} \cdots i_{4d} k} \right).$$

Chapter 9

Low-Rank Tensor Optimization

9.1 Introduction

As we mentioned in Chapter 4.1, in practise the tensors encountered in our real life often bears some low-rank structure, although the actual data may not appear so. In this chapter, we are discussing the two most important low-rank tensor problems.

Low-rank Tensor Completion Problem

The completion problem is a missing value estimation problem arising from medical imaging and computer vision. The so-called low-rank tensor completion problem is to recover a low-rank tensor from partial information: Given a linear map $\boldsymbol{L} : \mathbb{C}^{n_1 \times n_2 \cdots \times n_d} \to \mathbb{C}^p$, find the tensor $\mathcal{X}$ that fulfills the linear measurements $\boldsymbol{L}(\mathcal{X}) = b$ and while minimizes the CP rank of $\mathcal{X}$.

$$\begin{aligned} \min_{\mathcal{X}} \quad & \operatorname{rank}_{CP}(\mathcal{X}) \\ \text{s.t.} \quad & \boldsymbol{L}(\mathcal{X}) = b. \end{aligned} \tag{9.1}$$

Robust Tensor Recovery Problem

In some practical problems, the underlining tensor data $\mathcal{X}$ is not low-rank, but the summation of a low-rank tensor $\mathcal{Y}$ and a sparse tensor $\mathcal{Z}$. The robust tensor recovery problem is to find such decomposition:

$$\begin{aligned} \min_{\mathcal{Y},\mathcal{Z}} \quad & \operatorname{rank}_{CP}(\mathcal{Y}) + \lambda \|\mathcal{Z}\|_0 \\ \text{s.t.} \quad & \mathcal{Y} + \mathcal{Z} = \mathcal{X}, \end{aligned} \tag{9.2}$$

where $\|\cdot\|_0$ is the cardinality function. Since computing the CP rank of a tensor is difficult, people may want to unfold the tensor into some matrix and using the rank of resulting matrix to replace that of tensor. The notion of n-rank has been widely used in the low-rank tensor optimization problems [49, 77, 137]. However, the drawback of this approach is that the relationship between the n-rank and CP rank is still unclear. Since we have already established various relations between CP rank and matrix-rank of a tensor, our strategy is to replace CP rank with matrix-rank in (9.1) and (9.2) and solve the resulting problems. Some numerical results will be provided to justify our approach.

9.2 Optimizing Low-Rank Tensor Problem through Matrix Rank

In this section, we consider the two low-rank tensor optimization problems (9.1) and (9.2). Since matrix-rank of a tensor is easy to compute, we replace the CP-rank by matrix-rank, and get

$$\begin{array}{ll} \min\limits_{\mathcal{X}} & \text{rank}_M(\mathcal{X}) \\ \text{s.t.} & \boldsymbol{L}(\mathcal{X}) = b, \end{array} \qquad \begin{array}{ll} \min\limits_{\mathcal{Y},\mathcal{Z}} & \text{rank}_M(\mathcal{Y}) + \lambda\|\mathcal{Z}\|_0 \\ \text{s.t.} & \mathcal{Y} + \mathcal{Z} = \mathcal{X}. \end{array} \tag{9.3}$$

By definition of matrix-rank, for any permutation $\pi \in \Pi(1, \cdots, 2d)$,

$$\text{rank}\ (\boldsymbol{M}(\mathcal{X}_\pi)) \geq \text{rank}_M(\mathcal{X}).$$

So we can further replace the objectives in (9.3) with their upper bounds:

$$\text{rank}\ (\boldsymbol{M}(\mathcal{X}_\pi)) \quad \text{and} \quad \text{rank}\ (\boldsymbol{M}(\mathcal{Y}_\pi)) + \lambda\|\mathcal{Z}\|_0 \quad \text{respectively.}$$

Without loss of generality, we choose π to be the identical permutation, denote $X = \boldsymbol{M}(\mathcal{X})$, $Y = \boldsymbol{M}(\mathcal{Y})$ and $Z = \boldsymbol{M}(\mathcal{Z})$, and consider the following matrix optimization problems

$$\begin{array}{ll} \min\limits_{X} & \text{rank}\,(X) \\ \text{s.t.} & \boldsymbol{L}'(X) = b, \end{array} \qquad \begin{array}{ll} \min\limits_{Y,Z} & \text{rank}\,(Y) + \lambda\|Z\|_0 \\ \text{s.t.} & Y + Z = X, \end{array}$$

where $\boldsymbol{L}'$ is the linear mapping associated with $\boldsymbol{L}$ such that $\boldsymbol{L}'(X) = \boldsymbol{L}(\mathcal{X})$. In the presence of noise, the equality constraints are relaxed and appear as quadratic penalty

terms in the objective function. So the problems we are looking at are:

$$\min_X \quad \text{rank}\,(X) + \tfrac{\mu}{2}\|\boldsymbol{L}'(X) - b\|_2^2 \quad \text{and} \quad \min_{Y,Z} \quad \text{rank}\,(Y) + \lambda\|Z\|_0 + \tfrac{\mu}{2}\|Y + Z - X\|_2^2. \tag{9.4}$$

Notice the problems above are still NP-hard. If all the matrices are real-valued, people often replace the rank ($\text{rank}\,(\cdot)$) and cardinality ($\|\cdot\|_0$) functions with their convex surrogates, the nuclear norm and the L_1 norm respectively. In the following, let's check that whether these two convex approximations are also appropriate for complex-valued matrices.

Let's first do the singular value decomposition for a complex matrix X, that is $X = \sum_{i=1}^r \sigma_i u^i (v^i)^\dagger$, where $U = [u^1, \cdots, u^r]$ and $V = [v^1, \cdots, v^r]$ are both unitary matrices. Denote $\sigma = (\sigma_1, \cdots, \sigma_r)^\top \in \mathbb{R}^r$; then $\text{rank}\,(X) = \|\sigma\|_0$, which is same as the real-valued case. So nuclear norm is a good convex approximation of rank function even for complex-valued matrices.

While the definition of L_1 norm for complex matrix is quite different from that for real matrix. More precisely, given a complex matrix Z, its L_1 norm is the summation of modulus of all the components, i.e. $\|Z\|_1 = \sum_{k,\ell} |Z_{k,\ell}|$, where $|Z_{k,\ell}| = \sqrt{(\text{Re } Z_{k,\ell})^2 + (\text{Im } Z_{k,\ell})^2}$. A natural question is whether L_1 norm is still a good complex approximation of cardinality function in complex domain. Notice that once $Z_{k\ell} = 0$, we have Re $Z_{k\ell} = 0$ as well as Im $Z_{k\ell} = 0$. As a result, the sparsity of Z implies the group sparsity of the matrix $[\boldsymbol{V}(\text{Re } Z), \boldsymbol{V}(\text{Im } Z)]^\top$. It's well known that the $L_{2,1}$ norm is a good convex approximation of group sparsity. Recall that for a matrix $A \in \mathbb{R}^{m\times n}$, $\|A\|_{2,1} = \sum_{j=1}^n \|A_j\|_2$, where A_j is the j-th column of A. So

$$\|[\boldsymbol{V}(\text{Re } Z), \boldsymbol{V}(\text{Im } Z)]^\top\|_{2,1} = \sum_{k,\ell} \sqrt{(\text{Re } Z_{k,\ell})^2 + (\text{Im } Z_{k,\ell})^2},$$

which happens to be the L_1 norm of Z. Therefore, we can still replace the rank and cardinality functions with nuclear norm and L_1 norm respectively (9.4), and get two convex formulations:

$$\min_X \|X\|_* + \frac{\mu}{2}\|\boldsymbol{L}'(X) - b\|_2^2 \tag{9.5}$$

and

$$\min_{Y,Z} \|Y\|_* + \lambda\|Z\|_1 + \frac{\mu}{2}\|Y + Z - X\|_2^2. \tag{9.6}$$

We remark that, since the complex matrix-rank of a tensor (i.e., Definition 4.1.3) and the real matrix-rank of a tensor (i.e., conduct rank-one decomposition in real field) are the same. When the underlining data are real, we can restrict all the matrix operations be conducted in the real field.

In the next section, we will use some numerical results to justify the convex formulations (9.5) and (9.6).

9.3 Numerical Results

We have already derived some theoretical bounds for CP-rank in terms of matrix-rank in Chapter 4, but these bounds may seem loose. In this section, we present some numerical examples to which the matrix-rank works well, and thus show that it is a good replacement of CP-rank.

9.3.1 Synthetic Examples

We generate random complex valued tensors of size $(20\times 20\times 30\times 30)$, $(30\times 30\times 30\times 30)$ with CP-rank 10, and tensors of size $(30 \times 30 \times 40 \times 40)$, $(40 \times 40 \times 40 \times 40)$ with CP-rank 18. Then only 30% of the data was randomly selected to be the observations and we use model (9.5) to recover the missing data. We report the recovery relative error, the number of iterations required and the matrix-rank of the recovered tensor $\mathcal{X}^*$ in Table 9.1. The relative error for recovery is defined as $\frac{\|\mathcal{X}^*-\mathcal{X}_0\|_F}{\|\mathcal{X}_0\|_F}$, where $\mathcal{X}_0$ is the original noiseless low CP-rank tensor.

From Table 9.1, we can see that, even we only observe 30% of the data, we can still recover a tensor with a very small relative error. Moreover, the matrix-rank of the recovered tensor is the same as the CP-rank of the original tensor.

Inst.	CP-rank	Re. Error	Time	matrix-rank
Dimension $20 \times 20 \times 30 \times 30$				
1	10	3.36e-05	11.42	10
2	10	3.10e-05	10.95	10
3	10	3.46e-05	11.00	10
4	10	3.47e-05	11.03	10
5	10	2.82e-05	10.76	10
6	10	3.28e-05	10.78	10
7	10	3.83e-05	10.95	10
8	10	3.21e-05	11.17	10
9	10	2.92e-05	10.97	10
10	10	3.34e-05	11.34	10
Dimension $30 \times 30 \times 30 \times 30$				
1	10	1.67e-05	33.88	10
2	10	1.93e-05	34.96	10
3	10	1.92e-05	33.35	10
4	10	1.80e-05	33.31	10
5	10	1.69e-05	33.31	10
6	10	1.74e-05	32.64	10
7	10	1.74e-05	33.77	10
8	10	1.73e-05	31.79	10
9	10	1.15e-05	33.07	10
10	10	1.81e-05	32.67	10
Dimension $30 \times 30 \times 40 \times 40$				
1	18	1.30e-05	59.00	18
2	18	1.58e-05	63.34	18
3	18	1.48e-05	59.98	18
4	18	1.53e-05	63.70	18
5	18	1.46e-05	55.36	18
6	18	1.50e-05	52.92	18
7	18	1.41e-05	55.83	18
8	18	1.51e-05	55.01	18
9	18	1.48e-05	54.62	18
10	18	1.47e-05	53.84	18
Dimension $40 \times 40 \times 40 \times 40$				
1	18	9.11e-06	119.79	18
2	18	1.05e-05	116.39	18
3	18	1.02e-05	119.90	18
4	18	8.68e-06	130.78	18
5	18	8.69e-06	145.50	18
6	18	1.02e-05	159.95	18
7	18	9.99e-06	152.21	18
8	18	8.80e-06	151.66	18
9	18	1.09e-05	145.05	18
10	18	9.94e-06	144.86	18

Table 9.1: CP-rank of original tensor VS matrix-rank of recovered tensor through (9.5)

9.3.2 Example in Color Videos

A color video sequence is a good example of 4D data with row, column, color and time in each direction. In the following, we collected 50 colored frames, each of which is a $128 \times 160 \times 3$ tensor, forming a $128 \times 160 \times 3 \times 50$ tensor. We implemented the algorithm provided in [135] to solve problem (9.5). The results are shown in Figure 9.1, where the first row are three frames selected from the original video, the second row are the same

three frames with 80% of the data missed and the last row are the recovered video. We can see that we successfully recovered the original video except some loss in the color.

Figure 9.1: Video Completion. The first row are the 3 frames of the original video sequence. The second row are images with 80% missing data. The last row are recovered images

We collected another 50 frames of the same color video when a person come into the picture so we may consider problem (9.6) to differentiate the dynamic foreground (the moving person) and the static background. Moreover, the underlining data is still a tensor of size $128 \times 160 \times 3 \times 50$. Here we solve the problem (9.6) by resorting the algorithm in [138] and the results are shown in Figure 9.2, where the first row are three frames selected from the original video, the second row are reconstructed background

and the third row are the reconstructed foreground. We can see that we successfully decomposed the foreground and the background from the original video.

Figure 9.2: Robust Video Recovery. The first row are the 3 frames of the original video sequence. The second row are recovered background. The last row are recovered foreground

Chapter 10

An Application in Radar Waveform Design

10.1 Introduction

In radar systems, a key role is played by the ambiguity function of the waveform used to probe the environment. Indeed, it controls both the Doppler and the range resolutions of the system and also regulates the interference power, produced by unwanted returns, at the output of the matched filter to the target signature. Many papers have addressed the problem of properly designing the ambiguity function or its zero-Doppler cut [139, 140].

In the following, we shall propose a cognitive approach to devise radar waveforms sharing a desired ambiguity function. This study is motivated observing that during the recent past, the cognitive paradigm is becoming one of the leading approaches for advanced signal processing techniques, attempting to satisfy the more and more demanding system performance requirements. We suppose that the radar system can predict the actual scattering environment, using a dynamic environmental database, including a geographical information system, meteorological data, previous scans, and some electromagnetic reflectivity and spectral clutter models. Hence, exploiting the above information, the radar can locate the range-Doppler bins where strong unwanted returns are foreseen and, consequently, transmit a suitable waveform, to test a target of interest, whose ambiguity function exhibits low values in correspondence of those bins. We shall formulate the problem as a quartic order polynomial optimization problem

with constant modulus constraints forced to ensure phase-only modulated waveforms, compatible with today amplifier's technology.

In particular, consider a monostatic radar system which transmits a coherent burst of N pulses. The N-dimensional column vector $v \in \mathbb{C}^N$ of the observations, from the range-azimuth cell under test, can be expressed as: $v = \alpha_T s \odot \mathbf{p}(v_{d_T}) + z + w$, with $s = [s(1), ..., s(N)] \in \mathbb{C}^N$ the radar code, α_T a complex parameter accounting for the response of the target, $\mathbf{p}(v_{d_T}) = [1, e^{\boldsymbol{i} 2\pi v_{d_T}}, ..., e^{\boldsymbol{i} 2\pi (N-1) v_{d_T}}]^\top$, v_{d_T} the normalized target Doppler frequency, $z \in \mathbb{C}^N$ the interfering echo samples, $w \in \mathbb{C}^N$ the noise samples, and $\odot$ denotes the Hadamard product. The interfering vector z is the superposition of the returns from different uncorrelated point like scatterers, and models clutter, nontreating or treating targets (different from the one of interest) contributions. As a consequence, the vector z can be expressed as:

$$z = \sum_{k=1}^{N_t} \rho_k J^{r_k} s \odot \mathbf{p}(v_{d_k}), \tag{10.1}$$

where N_t is the number of interfering scatterers, $r_k \in \{0, 1, ..., N-1\}$, ρ_k, and v_{d_k} are, respectively, the range position, the echo complex amplitude, and the normalized Doppler frequency of the k-th scatterer. Furthermore, $\forall r \in \{-N+1, ..., 0, ..., N-1\}$

$$J^r(\ell, m) = \begin{cases} 1, & \text{if } \ell - m = r \\ 0, & \text{if } \ell - m \neq r, \end{cases} \quad (\ell, m) \in \{1, \cdots, N\}^2$$

denotes the shift matrix, and $J^{-r} = J^{r\top}$. According to (10.1), the output of the matched filter to the target signature $s \odot \mathbf{p}(v_{d_T})$, is given by:

$$(s \odot \mathbf{p}(v_{d_T}))^H v = \alpha_T \|s\|^2 + (s \odot \mathbf{p}(v_{d_T}))^H w + \sum_{k=1}^{N_t} \rho_k \left(s \odot \mathbf{p}(v_{d_T})\right)^H J^{r_k} \left(s \odot \mathbf{p}(v_{d_k})\right).$$

We assume that the noise vector w is a zero-mean, circular white noise, i.e. $\mathsf{E}[w] = 0$, $\mathsf{E}[ww^H] = \sigma_n^2 I$. Furthermore, we denote by $\sigma_k^2 = \mathsf{E}[|\rho_k|^2]$ the echo mean power, produced by the k-th scatterer. Additionally, we model the normalized Doppler frequency v_{d_k} as a random variable uniformly distributed around a mean Doppler frequency $\hat{v}_{d_k}$, i.e. $v_{d_k} \sim \mathbb{U}(\hat{v}_{d_k} - \frac{\epsilon}{2}, \hat{v}_{d_k} + \frac{\epsilon}{2})$ Consequently, the disturbance power at the output of the

matched filter is given by

$$\sum_{k=1}^{N_t} \sigma_k^2 \|s\|^2 \mathsf{E}[g_s(r_k, v_{d_k} - v_{d_T})] + \sigma^2 \|s\|^2, \tag{10.2}$$

where

$$g_s(r, v) = \frac{1}{\|s\|^2} \left| s^H J^r (s \odot \mathbf{p}(v)) \right|^2$$

is the ambiguity function of s, with $r \in \{0, 1, ..., N-1\}$ the time-lag and $v \in [-\frac{1}{2}, \frac{1}{2}]$ the normalized Doppler frequency. Based on (10.2), in order to characterize the mean disturbance power at the output of the matched filter to the target signature, we need to know the mean power σ_k^2 as well as the Doppler parameters $\hat{v}_k$ and ϵ_k of each scatterer. This information can be obtained in a cognitive fashion. Precisely, for a point like scatterer modeling the return from a clutter range-azimuth bin, we can characterize its parameters [141] using a dynamic environmental database, including a geographical information system, meteorological data, previous scans, and some electromagnetic reflectivity and spectral clutter models. Furthermore, with reference to both nontreating and treating targets, we can obtain information about their parameters exploiting the tracking files managed by the radar system.

In the following, without loss of generality, we center the Doppler frequency axis around the target Doppler frequency, namely all the normalized Doppler frequencies are expressed in terms of the difference with respect to v_{d_T} . Furthermore, we discretize the normalized Doppler frequency interval $[-\frac{1}{2}, \frac{1}{2})$ into N_v bins given by $v_h = -\frac{1}{2} + \frac{h}{N_v}, h = 0, ..., N_v - 1$.Thus, each statistical expectation

$$\mathsf{E}\left[\mathrm{g}_s\left(r_k, \nu_{d_k}\right)\right]$$

can be approximated with the sample mean over the Doppler bins intersecting $[\bar{\nu}_{d_k} - \frac{\epsilon_k}{2}, \bar{\nu}_{d_k} + \frac{\epsilon_k}{2}] = I_k$, namely

$$\mathsf{E}\left[\mathrm{g}_s\left(r_k, \nu_{d_k}\right)\right] \approx \frac{1}{Card\left(B_k\right)} \sum_{h \in B_k} \mathrm{g}_s\left(r_k, \nu_h\right). \tag{10.3}$$

where $B_k = \left\{ h : [\nu_h - \frac{1}{2N_\nu}, \nu_h + \frac{1}{2N_\nu}) \cap I_k \neq \emptyset \right\}$, i.e. the set of the Doppler bin indices

associated to the $k-$th scatterer.

$$\begin{aligned}
&\sum_{k=1}^{N_t}\sigma_k^2\|s\|^2\left(\frac{1}{Card\,(B_k)}\sum_{h\in B_k}\mathrm{g}_s\,(r_k,\nu_h)\right)+\sigma_n^2\|s\|^2=\\
&\sum_{r=0}^{N-1}\sum_{k=1}^{N_t}\sigma_k^2\|s\|^2\left(\frac{1}{Card\,(B_k)}\sum_{h\in B_k}\delta(r-r_k)\mathrm{g}_s\,(r,\nu_h)\right)+\sigma_n^2\|s\|^2=\\
&\sum_{r=0}^{N-1}\sum_{h=0}^{N_\nu-1}\|s\|^2\mathrm{g}_s\,(r,\nu_h)\left(\sum_{k=1}^{N_t}\delta(r-r_k)\mathbf{1}_{B_k}(h)\frac{\sigma_k^2}{Card\,(B_k)}\right)+\sigma_n^2\|s\|^2=\\
&\sum_{r=0}^{N-1}\sum_{h=0}^{N_\nu}p_{(r,h)}\|s\|^2\mathrm{g}_s\,(r,\nu_h)+\sigma_n^2\|s\|^2,
\end{aligned}\tag{10.4}$$

where, $\delta(\cdot)$ is the Kronecker delta function, $\mathbf{1}_A(x)$ denotes the indicator function related to the set A, and

$$p_{(r,h)}=\sum_{k=1}^{N_t}\delta(r-r_k)\mathbf{1}_{B_k}(h)\frac{\sigma_k^2}{Card\,(B_k)},$$

namely, it is the total interference power produced by the range-Doppler bin (r,ν_h). Notice that, in correspondence of range-Doppler regions free of interference, $p_{(r,h)}=0$.

Now, we focus on the design of a suitable radar waveform whose ambiguity function exhibits low values in correspondence of range-Doppler bins where strong unwanted returns are foreseen, reducing as much as possible the matched filter output disturbance power. To be complaint with today amplifier technology, we force a constant modulus constraint on the amplitude of the radar code, ensuring phase-only modulated waveforms. Precisely, if we assume $s\in\Omega_\infty^N$, then the design of continuous phase radar codes can be formulated as the following constrained optimization problem:

$$\mathrm{P}^\infty\ \begin{cases}\min\limits_{s} & \phi(s)\\ \text{s.t.} & s\in\Omega_\infty^N\end{cases}\tag{10.5}$$

Now, let us observe that

$$\phi(s)=\sum_{r=0}^{N-1}\sum_{h=0}^{N_\nu-1}p_{(r,h)}\left|s^\dagger J^r\mathrm{diag}\,\left(\mathbf{p}\,(\nu_h)\right)s\right|^2$$

thus, the objective function of problems P^∞ and P^M is in the form of

$$f(s)\ \ =\ \ \sum_{r_1=1}^{R_1}|s^\dagger A^{r_1}s|^2-\sum_{r_1=R_1+1}^{R_2}|s^\dagger A^{r_1}s|^2\tag{10.6}$$

with $A^{r_1} \in \mathbb{C}^{N,N}$, $r_1 = 1, \ldots, R_1, \ldots, R_2$, namely, it is a real valued conjugate homogeneous quartic function (see Section 7.4 for more details regarding this kind of functions). In fact, $\phi(s)$ can be expressed as in (10.6) taking $R_2 = 0$, $R_1 = NN_\nu$. Consequently, problem P^∞ belongs to the class of complex quartic minimization problems

$$\mathrm{CQ}^\infty \left\{ \begin{array}{ll} \min\limits_{s} & f(s) \\ \text{s.t.} & s \in \Omega_\infty^N \end{array} \right. , \tag{10.7}$$

with $f(s)$ given in (10.6).

10.2 Maximum Block Improvement Method

In this subsection, we devise three MBI type optimization algorithms, which try to locally improve the objective function in CQ^∞. The reason we only focus on local optimal solution is that the problem CQ^∞ is NP-hard as illustrated by the following theorem:

Theorem 10.2.1. *Problem CQ^∞ is NP-hard in general.*

Proof. We consider a reduction from a known NP-hard problem ([39]):

$$\mathrm{P}_1 \left\{ \begin{array}{ll} \min\limits_{y} & y^\dagger Q y \\ \text{s.t.} & y_i \in \Omega_\infty,\ i = 1, 2, \ldots, N, \end{array} \right. . \tag{10.8}$$

where Q is complex Hermitian positive semidefinite matrix. By a variable transformation $y_i \mapsto (s_i)^2$, $i = 1, 2, \cdots, N$, and the fact $y_i \in \Omega_\infty$ if and only if $s_i \in \Omega_\infty$, problem P_1 is equivalent to following complex quartic problem:

$$\mathrm{P}_2 \left\{ \begin{array}{ll} \min\limits_{s} & \sum\limits_{\ell,h} Q_{\ell h} (\bar{s}_\ell)^2 (s_h)^2 \\ \text{s.t.} & s_i \in \Omega_\infty,\ i = 1, 2, \ldots, N. \end{array} \right. . \tag{10.9}$$

The latter can be written in the form of problem CQ^∞, since for any $1 \le h, k \le N$

$$\left(|Q_{kh} \bar{x}_k x_h + x_k \bar{x}_h|^2 - |Q_{hk} \bar{x}_h x_k - x_h \bar{x}_k|^2 \right) / 2 = Q_{kh} (\bar{x}_k)^2 (x_h)^2 + Q_{hk} (\bar{x}_h)^2 (x_k)^2.$$

Therefore, the conclusion immediately follows from the NP-hardness of problem P_2. $\square$

The MBI method is an iterative algorithm known to achieve excellent performance in the maximization of real polynomial functions subject to spherical constraints [79]. Moreover, it was proved that the sequence produced by the MBI method converges to a stationary point for the relaxed multi-linear problem [79]; however, such stationary point is not ensured being a globally optimal solution.

Before proceeding further with the design of our MBI type algorithms, we point out that, for any finite value λ, problems CQ^{∞} shares the same (local) optimal solutions, respectively, of

$$\widetilde{\mathrm{CQ}}_{\lambda}^{\infty} \begin{cases} \max\limits_{s} & \lambda(s^{\dagger}s)^2 - f(s) \\ \text{s.t.} & s \in \Omega_{\infty}^{N} \end{cases}, \tag{10.10}$$

In fact, since $(s^{\dagger}s)^2 = N^2$ is a constant function whenever $s \in \Omega_{\infty}^{N}$, CQ^{∞} (resp. CQ^{M}) is equivalent to $\widetilde{\mathrm{CQ}}_{\lambda}^{\infty}$. Thus in the following, we focus on problems $\widetilde{\mathrm{CQ}}_{\lambda}^{\infty}$ and $\widetilde{\mathrm{CQ}}_{\lambda}^{M}$.

The first algorithm we propose, exploits the conjugate super-symmetric tensor representation of the complex quartic functions, see Section 7.4. Precisely, suppose that $f(s)$ is a complex quartic function in the form of (10.6) and let $\mathcal{G}^{\lambda}$ be the conjugate super-symmetric tensor form such that

$$\mathcal{G}^{\lambda}\left(\begin{pmatrix} s \\ \bar{s} \end{pmatrix}, \begin{pmatrix} s \\ \bar{s} \end{pmatrix}, \begin{pmatrix} s \\ \bar{s} \end{pmatrix}, \begin{pmatrix} s \\ \bar{s} \end{pmatrix}\right) = \lambda(s^{\dagger}s)^2 - f(s). \tag{10.11}$$

Then, we propose an MBI type method with a linear-improvement subroutine[1] for $\widetilde{\mathrm{CQ}}_{\lambda}^{\infty}$, as described in **Algorithm MBIL**.

[1] Notice that, $\mathcal{G}^{\lambda}\left(y, \begin{pmatrix} s_k^2 \\ \bar{s}_k^2 \end{pmatrix}, \begin{pmatrix} s_k^3 \\ \bar{s}_k^3 \end{pmatrix}, \begin{pmatrix} s_k^4 \\ \bar{s}_k^4 \end{pmatrix}\right) = {c^1}^{\top} y$ is a linear function in the variable y, and we denote by $c^1 \to \mathcal{G}^{\lambda}\left(\cdot, \begin{pmatrix} s_k^2 \\ \bar{s}_k^2 \end{pmatrix}, \begin{pmatrix} s_k^3 \\ \bar{s}_k^3 \end{pmatrix}, \begin{pmatrix} s_k^4 \\ \bar{s}_k^4 \end{pmatrix}\right)$ the vector associated to the linear function on the right side of the arrow. Similarly we proceed for the other variables.

Algorithm MBIL:

0 (Initialization): Generate, possibly randomly, $(s_0^1, s_0^2, s_0^3, s_0^4)$ with $s_0^m \in \Omega_\infty^N$ for $m = 1, 2, 3, 4$, and compute the initial objective value $v_0 = \mathcal{G}^\lambda\left(\binom{s_0^1}{\bar{s}_0^1}, \binom{s_0^2}{\bar{s}_0^2}, \binom{s_0^3}{\bar{s}_0^3}, \binom{s_0^4}{\bar{s}_0^4}\right)$. Set $k = 0$.

1 (Block Linear Improvement): Let

$c^1 \to \mathcal{G}^\lambda\left(\cdot, \binom{s_k^2}{\bar{s}_k^2}, \binom{s_k^3}{\bar{s}_k^3}, \binom{s_k^4}{\bar{s}_k^4}\right)$, $c^2 \to \mathcal{G}^\lambda\left(\binom{s_k^1}{\bar{s}_k^1}, \cdot, \binom{s_k^3}{\bar{s}_k^3}, \binom{s_k^4}{\bar{s}_k^4}\right)$,

$c^3 \to \mathcal{G}^\lambda\left(\binom{s_k^1}{\bar{s}_k^1}, \binom{s_k^2}{\bar{s}_k^2}, \cdot, \binom{s_k^4}{\bar{s}_k^4}\right)$, $c^4 \to \mathcal{G}^\lambda\left(\binom{s_k^1}{\bar{s}_k^1}, \binom{s_k^2}{\bar{s}_k^2}, \binom{s_k^3}{\bar{s}_k^3}, \cdot\right)$,

be the vectors associated to the linear functions on the right side of the arrows.

For $m = 1, 2, 3, 4$ let $y_{k+1}^m = \arg\max_{s \in \Omega_\infty^N} \binom{s}{\bar{s}}^\top c^m$, $w_{k+1}^m = \begin{pmatrix} y_{k+1}^m \\ \bar{y}_{k+1}^m \end{pmatrix}^\top c^m$.

2 (Maximum Improvement): Let $w_{k+1} = \max_{1 \le m \le 4} w_{k+1}^m$ and $m^* = \arg\max_{1 \le m \le 4} w_{k+1}^m$. Replace $s_{k+1}^m = s_k^m$ for all $m \neq m^*$, $s_{k+1}^{m^*} = y_{k+1}^{m^*}$ and $v_{k+1} = w_{k+1}$.

3 (Stopping Criterion): If $|\frac{v_{k+1} - v_k}{\max(1, v_k)}| < \epsilon$, stop. Otherwise, set $k = k + 1$, and go to step 1.

4 (Output): For $1 \le n \le 4$, let

$t^n = \mathcal{G}^\lambda\left(\binom{s_{k+1}^n}{\bar{s}_{k+1}^n}, \binom{s_{k+1}^n}{\bar{s}_{k+1}^n}, \binom{s_{k+1}^n}{\bar{s}_{k+1}^n}, \binom{s_{k+1}^n}{\bar{s}_{k+1}^n}\right)$,

$n^* = \arg\max_{1 \le n \le 4} t^n$. Return t^{n^*} and $s_{k+1}^{n^*}$.

Notice that the objective value in each iteration of MBIL method is generally increasing except for the last step. That is because the returned value t^{n^*} is the value of a polynomial function instead of the multi-linear form on which the MBIL algorithm is applied. However, we will show later, when the polynomial function itself is convex, monotonicity of the MBIL method can be guaranteed. In this light, it is important to study the convexity of a complex quartic function.

Theorem 10.2.2. *Suppose $f(s)$ is a complex quartic function.*

- *If $f(s) = \sum_{r_1=1}^{R_1} |s^\dagger A^{r_1} s|^2 - \sum_{r_1=R_1+1}^{R_2} |s^\dagger A^{r_1} s|^2$, then f is convex with respect to s if and only if*

$$h(y, z) = \sum_{r_1=1}^{R_1} h_{r_1}(y, z) - \sum_{r_1=R_1+1}^{R_2} h_{r_1}(y, z) \ge 0, \ \ \forall y, z \in \mathbb{C}^n \tag{10.12}$$

where

$$\begin{aligned} h_{r_1}(y,z) &= (y^\dagger A^{r_1} y z^\dagger A^{r_1\dagger} z + y^\dagger A^{r_1} z y^\dagger A^{r_1\dagger} z + y^\dagger A^{r_1} z z^\dagger A^{r_1\dagger} y + \\ &\quad z^\dagger A^{r_1} y y^\dagger A^{r_1\dagger} z + z^\dagger A^{r_1} y z^\dagger A^{r_1\dagger} y + z^\dagger A^{r_1} z y^\dagger A^{r_1\dagger} y), \ \ \forall y, z \in \mathbb{C}^N. \end{aligned}$$

- *If $\mathcal{H}$ is the conjugate partial-symmetric fourth order tensor form such that $\mathcal{H}(\bar{s}, s, \bar{s}, s) = f(s)$, then f is convex with respect to s if and only if*

$$4\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) \geq 0, \ \ \forall y, z \in \mathbb{C}^N. \tag{10.13}$$

- *If $\mathcal{G}$ is the conjugate super-symmetric fourth order tensor form such that $\mathcal{G}\left(\binom{s}{\bar{s}}, \binom{s}{\bar{s}}, \binom{s}{\bar{s}}, \binom{s}{\bar{s}}\right) = f(s)$, then f is convex with respect to s if and only if*

$$\mathcal{G}\left(\begin{pmatrix} y \\ \bar{y} \end{pmatrix}, \begin{pmatrix} y \\ \bar{y} \end{pmatrix}, \begin{pmatrix} z \\ \bar{z} \end{pmatrix}, \begin{pmatrix} z \\ \bar{z} \end{pmatrix}\right) \geq 0, \ \ \forall y, z \in \mathbb{C}^N. \tag{10.14}$$

Proof. To study the convexity of complex function, recall the following theorem in convex analysis for real valued function [142]:

Theorem 10.2.3. *Let $g : \mathbb{R}^n \to \mathbb{R}$ be a function. Given $s_0, s \in \mathbb{R}^n$, define the function $\tilde{g}_{(s_0,s)} : \mathbb{R} \to \mathbb{R}$ by $\tilde{g}_{(s_0,s)}(t) = g(s_0 + ts)$. Then, g is convex on $\mathbb{R}^n$ if and only if $\tilde{g}_{(s_0,s)}$ is convex in $\mathbb{R}$ for any $s_0, s \in \mathbb{R}^n$, if and only if $\tilde{g}''_{(s_0,s)}(t) \geq 0$ for all $t \in \mathbb{R}$ and $s_0, s \in \mathbb{R}^n$, assuming that the second order derivatives exists.*

Notice that the value of complex quartic function $f(s)$ is always real, so it can be viewed as real valued function with respect to real variables Re (s) and Im (s). Let us compute the second derivative of the real value complex functions of our interest.

Lemma 10.2.4. *Suppose $f(s)$ is a complex quartic function; given $x, y \in \mathbb{C}^N$, define $\tilde{f}_{(x,y)} : \mathbb{R} \to \mathbb{R}$ by $\tilde{f}_{(x,y)}(t) = f(x + ty)$. Then, denoting by $z = x + ty$, we have:*

- *If $f(s) = |s^\dagger A s|^2$, then*

$$\begin{aligned} \tilde{f}''_{(x,y)}(t) &= 2\Big(y^\dagger A y z^\dagger A^\dagger z + y^\dagger A z y^\dagger A^\dagger z + y^\dagger A z z^\dagger A^\dagger y & (10.15) \\ &\quad + z^\dagger A y y^\dagger A^\dagger z + z^\dagger A y z^\dagger A^\dagger y + z^\dagger A z y^\dagger A^\dagger y\Big) & (10.16) \end{aligned}$$

and it is always a real value function.

- *If $\mathcal{H}$ is the conjugate partial-symmetric fourth order tensor form such that $f(s) = \mathcal{H}(\bar{s}, s, \bar{s}, s)$, then*

$$\tilde{f}''_{(x,y)}(t) = 8\mathcal{H}(\bar{y}, y, \overline{z}, z) + 2\mathcal{H}(\bar{y}, z, \bar{y}, z) + 2\mathcal{H}(\overline{z}, y, \overline{z}, y)$$

  and it is always a real value function.

- *If $\mathcal{G}$ is the conjugate super-symmetric fourth order tensor form such that $f(s) = \mathcal{G}\left(\binom{s}{\bar{s}}, \binom{s}{\bar{s}}, \binom{s}{\bar{s}}, \binom{s}{\bar{s}}\right)$, then*

$$\tilde{f}''_{(x,y)}(t) = 12\mathcal{G}\left(\binom{y}{\bar{y}}, \binom{y}{\bar{y}}, \binom{z}{\overline{z}}, \binom{z}{\overline{z}}\right)$$

  and it is always a real value function.

Proof. We only prove the statement (ii); all other cases are almost the same. In this case,

$$\tilde{f}_{(x,y)}(t) = \mathcal{H}(\overline{x+ty}, x+ty, \overline{x+ty}, x+ty).$$

Due to conjugate-partial-symmetry,

$$\tilde{f}'_{(x,y)}(t) = 2\mathcal{H}(\overline{y}, x+ty, \overline{x+ty}, x+ty) + 2\mathcal{H}(\overline{x+ty}, y, \overline{x+ty}, x+ty)$$

and

$$\begin{aligned}
\tilde{f}''_{(x,y)}(t) &= 2\left(2\mathcal{H}(\overline{y}, y, \overline{x+ty}, x+ty) + \mathcal{H}(\overline{y}, x+ty, \overline{y}, x+ty)\right) \\
&\quad +2\left(2\mathcal{H}(\overline{y}, y, \overline{x+ty}, x+ty) + \mathcal{H}(\overline{x+ty}, y, \overline{x+ty}, y)\right) \\
&= 8\mathcal{H}(\bar{y}, y, \overline{x+ty}, x+ty) + 2\mathcal{H}(\bar{y}, x+ty, \bar{y}, x+ty) + 2\mathcal{H}(\overline{x+ty}, y, \overline{x+ty}, y).
\end{aligned}$$

Furthermore, by hermitian partial symmetry of $\mathcal{H}$, $\mathcal{H}(\bar{y}, y, \overline{x+ty}, x+ty) = \bar{\mathcal{H}}(y, \bar{y}, x+ty, \overline{x+ty}) = \overline{\mathcal{H}(\bar{y}, y, \overline{x+ty}, x+ty)}$ and $\mathcal{H}(\bar{y}, x+ty, \bar{y}, x+ty) = \bar{\mathcal{H}}(x+ty, \bar{y}, x+ty, \bar{y}) = \overline{\mathcal{H}(\overline{x+ty}, y, \overline{x+ty}, y)}$. Hence, as expected $\tilde{f}''_{(x,y)}(t)$ is always real. □

By arbitrariness of t and y in Lemma 10.2.4, the vector z, obtained through the variable transformation $x + ty \mapsto z$, is a free complex variable with respect to x. Thus, combining Lemma 10.2.4 and Theorem 10.2.3 we obtain the convexity characterizations for the various complex quartic function representations given in (10.12), (10.13), and (10.14). □

Theorem 10.2.2 indicates that the convexity of a quartic function is equivalent to the nonnegativity of a certain biquadratic function. We further notice that the biquadratic function corresponding to the quartic function $(s^\dagger s)^2$ is $2y^\dagger yz^\dagger z + (y^\dagger z + z^\dagger y)^2$, which is strictly positive whenever $y \neq 0$ and $z \neq 0$. Consequently, we can make any quartic function convex by adding $(s^\dagger s)^2$ multiplied by a large enough constant λ.

Corollary 10.2.5. *Suppose $f(s)$ is a quartic function represented in the form of* (10.6) *and let $h(y,z)$ be the function defined by* (10.12)*. Then $\lambda(s^\dagger s)^2 - f(s)$ is convex in s if and only if the scalar λ satisfies:*

$$\lambda\left(2y^\dagger yz^\dagger z + (y^\dagger z + z^\dagger y)^2\right) - h(y,z) \geq 0, \ \ \forall y, z \in \mathbb{C}^N, \tag{10.17}$$

Furthermore, letting

$$\lambda^* = \max_{\|z\|=1, \|y\|=1} h(y,z), \tag{10.18}$$

for any $\lambda \geq \lambda^*/2$, $\lambda(s^\dagger s)^2 - f(s)$ is convex in s.

Proof. For fixed λ, let $g(s) = \lambda(s^\dagger s)^2 - f(s)$. From Theorem 10.2.2, $g(s)$ is a convex function in s if and only if

$$\lambda\left(2y^\dagger yz^\dagger z + (y^\dagger z + z^\dagger y)^2\right) - h(y,z) \geq 0, \ \ \forall y, z \in \mathbb{C}^N, \tag{10.19}$$

Furthermore, let us define

$$\lambda^* = \max_{\|z\|_2=1, \|y\|_2=1} h(y,z),$$

which is a finite real value by the compactness of feasible set and the continuity of the function $h(y,z)$. Additionally, since $h(y,z)$ is a biquadratic function on z and y,

$$\lambda^* \geq h\left(\frac{y}{\|y\|_2}, \frac{z}{\|z\|_2}\right) = h(y,z)/y^\dagger yz^\dagger z \quad \forall y, z \in \mathbb{C}^N.$$

Therefore for any $\lambda \geq \frac{1}{2}\lambda^*$,

$$\lambda\left(2y^\dagger yz^\dagger z + (y^\dagger z + z^\dagger y)^2\right) - h(y,z) \geq \lambda^*\left(y^\dagger yz^\dagger z\right) - h(y,z) \geq 0.$$

Thus, condition (10.17) is satisfied and the conclusion follows. □

Once the characteristics of the convexity of complex quartic function are given, then we are in a position to extend the result obtained in [143, Ch. 5] to the complex case:

Theorem 10.2.6. *Suppose $g(s)$ is a convex complex quartic function and let $\mathcal{G}$ be the conjugate super-symmetric tensor form associated to $g(s)$; then*

$$\begin{aligned} & \mathcal{G}\left(\begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}\right) \\ \leq & \max\left\{ \mathcal{G}\left(\begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}\right), \mathcal{G}\left(\begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}\right), \right. \\ & \left. \mathcal{G}\left(\begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}\right), \mathcal{G}\left(\begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}\right)\right\}. \end{aligned}$$

Proof. The key idea here is to apply inequality (10.14) twice. First all of denote $y' = s^1 + s^2$, $y'' = s^1 - s^2$, $z' = s^3 + s^4$ and $z'' = s^3 - s^4$, by inequality (10.14)

$$\begin{aligned} 0 \leq & \mathcal{G}\left(\begin{pmatrix} y' \\ \bar{y}' \end{pmatrix}, \begin{pmatrix} y' \\ \bar{y}' \end{pmatrix}, \begin{pmatrix} z'' \\ \bar{z}'' \end{pmatrix}, \begin{pmatrix} z'' \\ \bar{z}'' \end{pmatrix}\right) + \mathcal{G}\left(\begin{pmatrix} y'' \\ \bar{y}'' \end{pmatrix}, \begin{pmatrix} y'' \\ \bar{y}'' \end{pmatrix}, \begin{pmatrix} z' \\ \bar{z}' \end{pmatrix}, \begin{pmatrix} z' \\ \bar{z}' \end{pmatrix}\right) \\ = & 2\sum_{\substack{i=1,2 \\ j=3,4}} \mathcal{G}\left(\begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}\right) - 8\mathcal{G}\left(\begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}\right). \end{aligned}$$

Secondly, again by inequality (10.14)

$$\begin{aligned} 0 \leq & \mathcal{G}\left(\begin{pmatrix} s^i + s^j \\ \bar{s}^i + \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^i + s^j \\ \bar{s}^i + \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^i - s^j \\ \bar{s}^i - \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^i - s^j \\ \bar{s}^i - \bar{s}^j \end{pmatrix}\right) \\ = & \mathcal{G}\left(\begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}\right) + \mathcal{G}\left(\begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}\right) - \\ & 2\mathcal{G}\left(\begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}, \begin{pmatrix} s^j \\ \bar{s}^j \end{pmatrix}\right) \qquad \forall i = 1, 2 \;\; j = 3, 4 \end{aligned}$$

Combining the two inequalities above yields

$$8\mathcal{G}\left(\begin{pmatrix} s^1 \\ \bar{s}^1 \end{pmatrix}, \begin{pmatrix} s^2 \\ \bar{s}^2 \end{pmatrix}, \begin{pmatrix} s^3 \\ \bar{s}^3 \end{pmatrix}, \begin{pmatrix} s^4 \\ \bar{s}^4 \end{pmatrix}\right) \leq 2\sum_{i=1}^{4} \mathcal{G}\left(\begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}, \begin{pmatrix} s^i \\ \bar{s}^i \end{pmatrix}\right)$$

proving the desired inequality. □

Thus, exploiting Corollary 10.2.5 and Theorem 10.2.6, we have that for any complex quartic function $f(s)$, we can find a λ such that the devised MBIL algorithm is monotonically increasing. Otherwise stated $\lambda^*/2$, with λ^* given in (10.18), ensures the equivalence between problems (10.10) and their relaxed multi-linear problems. Notice that, in order to compute λ^*, i.e. the maximum value of the biquadratic function $h(y, z)$,

the MBI algorithm proposed in [79] can be exploited. Finally, we explicitly point out that the quality of the solution can be improved by repeatedly using **Algorithm MBIL**, setting each time $s_0^1 = s_0^2 = s_0^3 = s_0^4 = s_{k+1}^{n^*}$ as new starting points, if further progress is still possible.

The second method we propose, exploits the conjugate partial-symmetric tensor representation of the complex quartic functions, also see Section 7.4. To this end, let $f(s)$ be a complex quartic function in the form of (10.6) and $\mathcal{H}^\lambda$ be the conjugate-partial-symmetric tensor form such that

$$\lambda(s^\dagger s)^2 - f(s) = \mathcal{H}^\lambda(\bar{s}, s, \bar{s}, s). \tag{10.20}$$

Hence, we introduce an MBI type method with a quadratic-improvement subroutine[2] for problem $\widetilde{\mathrm{CQ}}_\lambda^\infty$, as described in **Algorithm MBIQ**.

Algorithm MBIQ:

0 (Initialization): Generate, possibly randomly, (s_0^1, s_0^2) with $s_0^m \in \Omega_\infty^N$ for $m = 1, 2$ and compute the initial objective value $v_0 = \mathcal{H}^\lambda\left(\bar{s}_0^1, s_0^1, \bar{s}_0^2, s_0^2\right)$. Set $k = 0$.

1 (Block Quadratic Improvement): Let $B^1 \to \mathcal{H}^\lambda\left(\cdot, \cdot, \bar{s}_k^2, s_k^2\right)$, $B^2 \to \mathcal{H}^\lambda\left(\bar{s}_k^1, s_k^1, \cdot, \cdot\right)$, be the matrices associated to the bilinear functions on the right side of the arrows. For $m = 1, 2$ let $y_{k+1}^m = \arg\max_{s \in \Omega_\infty^N} (s)^\dagger B^m s$, $w_{k+1}^m = \left(y_{k+1}^m\right)^\dagger B^m y_{k+1}^m$.

2 (Maximum Improvement): Let $w_{k+1} = \max_{1 \le m \le 2} w_{k+1}^m$ and $m^* = \arg\max_{1 \le m \le 2} w_{k+1}^m$. Replace $s_{k+1}^m = s_k^m$ for all $m \neq m^*$, $s_{k+1}^{m^*} = y_{k+1}^{m^*}$ and $v_{k+1} = w_{k+1}$.

3 (Stopping Criterion): If $|\frac{v_{k+1} - v_k}{\max(1, v_k)}| < \epsilon$, stop. Otherwise, set $k = k + 1$, and go to step 1. **4** (Output): For $n = 1, 2$, let $t^n = \mathcal{H}^\lambda\left(\bar{s}_{k+1}^n, s_{k+1}^n, \bar{s}_{k+1}^n, s_{k+1}^n\right)$, $n^* = \arg\max_{n=1,2} t^n$. Return t^{n^*} and $s_{k+1}^{n^*}$.

Similar to Theorem 10.2.6, a sufficient condition for monotonicity of MBIQ method is given below.

Theorem 10.2.7. *Consider the complex quartic function $f(s)$ and let $\mathcal{H}$ be the associated conjugate-partial-symmetric fourth order tensor form. Then,*

$$2\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) \geq 0, \ \ \forall y, z \in \mathbb{C}^N \tag{10.21}$$

[2] Notice that, $\mathcal{H}^\lambda\left(y^1, y^2, \bar{s}_k^2, s_k^2\right) = (y^1)^\top B^1 y^2$ is a bilinear function in the variables y^1 and y^2, and we denote by $B^1 \to \mathcal{H}^\lambda\left(\cdot, \cdot, \bar{s}_k^2, s_k^2\right)$ the matrix associated to the bilinear function on the right side of the arrow. Similarly we proceed for the other pair of variables.

implies

$$\mathcal{H}(\bar{y}, y, \bar{z}, z) \leq \max\{\mathcal{H}(\bar{y}, y, \bar{y}, y), \mathcal{H}(\bar{z}, z, \bar{z}, z)\} \quad \forall y, z \in \mathbb{C}^N.$$

Proof. By applying formula (10.21) to $y = y^1 - z^1$ and $z = y^1 + z^1$, one has

$$\begin{aligned} 0 &\leq 2\mathcal{H}(\overline{y^1 + z^1}, y^1 + z^1, \overline{y^1 - z^1}, y^1 - z^1) + \mathcal{H}(\overline{y^1 + z^1}, y^1 - z^1, \overline{y^1 + z^1}, y^1 - z^1) + \\ &\quad + \mathcal{H}(\overline{y^1 - z^1}, y^1 + z^1, \overline{y^1 - z^1}, y^1 + z^1) \\ &= 4\left(\mathcal{H}(\bar{y}^1, y^1, \bar{y}^1, y^1) + \mathcal{H}(\bar{z}^1, z^1, \bar{z}^1, z^1)\right) - 8\mathcal{H}(\bar{y}^1, y^1, \bar{z}^1, z^1), \end{aligned}$$

and the conclusion follows. □

In light of Theorem 10.2.7, one may ask whether we can find a λ large enough such that conjugate-partial-symmetric fourth order tensor form $\mathcal{H}^\lambda$ associated to $\lambda(s^\dagger s)^2 - f(s)$ satisfies (10.21). Unfortunately, this is not possible. In fact, let us consider the conjugate-partial-symmetric form $\mathcal{H}$ corresponding to the quartic function $(s^\dagger s)^2$; then

$$\mathcal{H}(s, y, z, w) = \frac{1}{2}\left((s^\top y)(z^\top w) + (z^\top y)(s^\top w)\right).$$

Thus, $\mathcal{H}(\bar{y}, y, \bar{z}, z) = \frac{1}{2}\left((y^\dagger y)(z^\dagger z) + (z^\dagger y)(y^\dagger z)\right)$, $\mathcal{H}(\bar{z}, y, \bar{z}, y) = (z^\dagger y)(z^\dagger y)$, and $\mathcal{H}(\bar{y}, z, \bar{y}, z) = (y^\dagger z)(y^\dagger z)$. Moreover,

$$2\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) = y^\dagger y z^\dagger z + y^\dagger z z^\dagger y + y^\dagger z y^\dagger z + z^\dagger y z^\dagger y.$$

Simply choosing $y = \boldsymbol{i}z$ leads to $2\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) = 0$ implying that it is not strictly positive, so the technique in Corollary 2.5 does not apply here. This phenomenon lies in the fact that (10.21) is a stronger condition than the convexity requirements (10.13). To see this, let us consider a complex quartic function whose associated conjugate-partial-symmetric tensor $\mathcal{H}$ satisfies (10.21); then

$$2\mathcal{H}(\bar{y}, y, \bar{z}, z) \geq -\mathcal{H}(\bar{y}, z, \bar{y}, z) - \mathcal{H}(\bar{z}, y, \bar{z}, y), \quad \forall y, z \in \mathbb{C}^N,$$

and replacing y by $e^{\boldsymbol{i}\theta}y$ we have

$$\begin{aligned} 2\mathcal{H}(\bar{y}, y, \bar{z}, z) &= 2\mathcal{H}(\overline{e^{\boldsymbol{i}\theta}y}, e^{\boldsymbol{i}\theta}y, \bar{z}, z) \geq -\mathcal{H}(\overline{e^{\boldsymbol{i}\theta}y}, z, \overline{e^{\boldsymbol{i}\theta}y}, z) - \mathcal{H}(\bar{z}, e^{\boldsymbol{i}\theta}y, \bar{z}, e^{\boldsymbol{i}\theta}y) \\ &= -2\mathrm{Re}\left(e^{\boldsymbol{i}2\theta}\mathcal{H}(\bar{z}, y, \bar{z}, y)\right), \quad \forall y, z \in \mathbb{C}^N, \theta \in [0, 2\pi]. \end{aligned}$$

Obviously, choosing

$$2\theta = \pi - \arg\left(\mathcal{H}(\bar{z}, y, \bar{z}, y)\right),$$

it holds that $\mathcal{H}(\bar{y}, y, \bar{z}, z) \geq 0 \ \forall y, z \in \mathbb{C}^N$. Now adding $2\mathcal{H}(\bar{y}, y, \bar{z}, z)$ to the left hand side of (10.21), we have that $\forall y, z \in \mathbb{C}^N$

$$4\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) \geq 2\mathcal{H}(\bar{y}, y, \bar{z}, z) + \mathcal{H}(\bar{y}, z, \bar{y}, z) + \mathcal{H}(\bar{z}, y, \bar{z}, y) \geq 0,$$

implying that the corresponding complex quartic function is convex. However, the difference between (10.21) and condition (10.13) is very subtle, so in practice we decide to use the same λ in both **Algorithm MBIL** and **Algorithm MBIQ**. Some useful remarks on **Algorithm MBIQ** are now given:

(i) By conjugate partial-symmetry of $\mathcal{H}^{\lambda}$, B^1 and B^2 in step 1 are both Hermitian matrices.

(ii) The complex quadratic problems in step 1 are still NP-hard, in practice we apply the randomization algorithms in [39] to get a good approximate solution.

(iii) In order to improve the quality of the solution, we could repeatedly use MBIQ approach, setting $s_0^1 = s_0^2 = s_{k+1}^{n^*}$ as new starting points, if further progress is still possible.

10.3 Performance Assessment

In this Section, we analyze the capability of the proposed MBI type algorithms to select a radar phase code with a properly shaped ambiguity function on continuous phase code design. Precisely, we devise a radar code considering the following three-step based procedure:

1. select the value of the parameter $\tilde{\lambda} > 0$, for instance such that $\tilde{\lambda}(s^{\dagger}s)^2 - \phi(s)$ is convex;

2. apply **Algorithm MBIL** and **Algorithm MBIQ**, to problem $\widetilde{\mathrm{CQ}}_{\tilde{\lambda}}^{\infty}$ with $f(s) = \phi(s)$, each starting from K_1 different initial points;

3. get a feasible solution $s^{\star}$ for P^{∞}, picking the solution which leads to the minimum objective function value among the outputs of the two algorithms.

As to the selection of the parameter $\tilde{\lambda}$, its value can be chosen by user. Notice that, Corollary 10.2.5 provides a systematic approach to compute it in order to ensure the convexity of the objective function $\tilde{\lambda}(s^\dagger s)^2 - \phi(s)$; in this way, the monotonicity of **Algorithm MBIL** is guaranteed by Theorem 10.2.6 and the monotonicity of **Algorithm MBIQ** is expectable based on Theorem 10.2.7. Nevertheless, if the value of $\tilde{\lambda}$ is too high, the original objective function $-\phi(s)$ is significantly changed with respect to the one considered in the MBI type algorithms and the numerical performance of the proposed procedures could be consequently affected (the bias term $\tilde{\lambda}N^2$ could mask, from a numerical point of view, the variations in the objective function $\phi(s)$). Based on the previous considerations, a reasonable choice is to consider the smallest $\tilde{\lambda}$ ensuring the convexity. For this reason, denoting by $\lambda_1 = \frac{\lambda^\star}{2}$ with $\lambda^\star$ defined in (10.18), the focus will be on $\tilde{\lambda} \in \{\lambda_1, \lambda_1/3, \lambda_1/6\}$.

We address the performance analysis considering the range-Doppler interference scenario reported in Figure 10.1. In this interference map, the red portions correspond to the regions of the unwanted range-Doppler returns (interference).

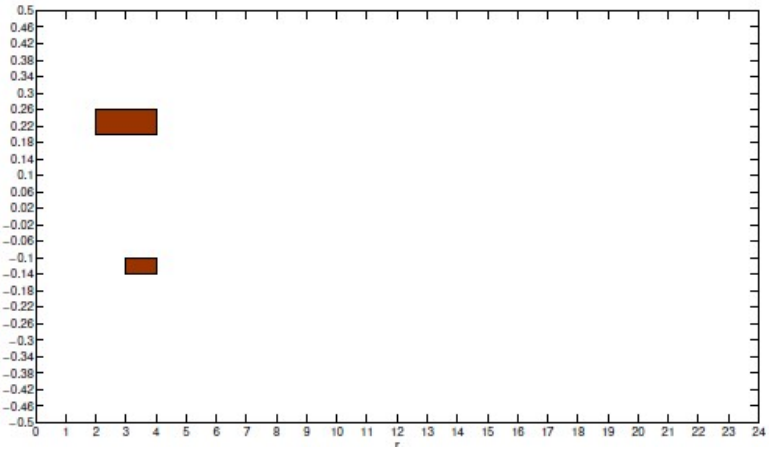

Figure 10.1: Range-Doppler inference map

In this scenario, we discretize the normalized Doppler frequency axis into $N_\nu = 50$ bins, namely the discrete Doppler frequencies are $\nu_h = -\frac{1}{2} + h\frac{1}{50}$, $h = 0, 1, \ldots, 49$. Furthermore, we assume a uniform interference power among the interference bins.

Precisely, we suppose

$$p_{(r,h)} = \begin{cases} 1 & (r,h) \in \{2,3,4\} \times \{35,36,37,38\} \\ 1 & (r,h) \in \{3,4\} \times \{18,19,20\} \\ 0 & \text{otherwise} \end{cases},$$

As to the parameters of the devised MBI type algorithms, we require a minimum iteration gain of $\epsilon = 10^{-6}$, and allow for a maximum value of 5000 iterations for **Algorithm MBIL**, and 200 iterations for **Algorithm MBIQ**. Additionally, as to the quadratic-improvement subroutine, involved in **Algorithm MBIQ**, we assume that 100 randomizations are performed for getting a good approximate solution and computing adequate improvements.

In order to assess the performance of the proposed three-step based procedure for the

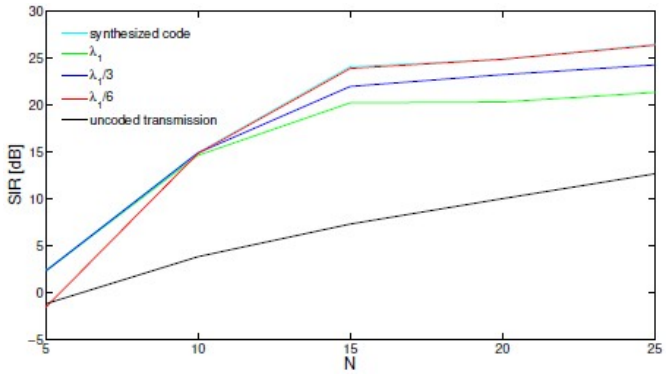

Figure 10.2: SIR versus N, for the uncoded transmission, the synthesized code, and the radar codes designed exploiting some $\tilde{\lambda}$ values.

design of a continuous phase code, we assume $K_1 = 11$; specifically, in step 2 we run ten times the MBI type algorithms with independent random initial points, as well as once with the uncoded sequence $\tilde{s}$, $\tilde{s}(i) = 1,\ i = 1, 2, \ldots, N$; the best solution is kept as the devised code. The same initial points are used for any tested $\tilde{\lambda} \in \{\lambda_1, \lambda_1/3, \lambda_1/6\}$. In Figure 10.2, we plot the Signal to Interference Ratio (SIR), defined as

$$\text{SIR} = \frac{N^2}{\sum_{r=1}^{N} \sum_{h=1}^{N_\nu} p_{(r,h)} \|s\|^2 \mathrm{g}_s\left(r, \nu_h\right)},$$

versus the length of the code, averaged over 20 independent trials of the proposed algorithm, for the devised codes $s^{\star}_{\lambda_1}, s^{\star}_{\lambda_1/3}, s^{\star}_{\lambda_1/6}$ and the uncoded sequence $\tilde{s}$. Also, the SIR achieved by the best radar code $s^{\star}$, among $\left\{s^{\star}_{\lambda_1}, s^{\star}_{\lambda_1/3}, s^{\star}_{\lambda_1/6}\right\}$ at each trial, is plotted (the synthesized code).

As expected, the synthesized code outperforms the uncoded transmission, showing the capability of the proposed algorithms to contrast and suppress the interfering returns. Furthermore, increasing N, smaller values of $\tilde{\lambda}$ allow to obtain better performances, probably reflecting the numerical problems that could affect the proposed procedures when high values of $\tilde{\lambda}N^2$ are considered. Nevertheless, $\tilde{\lambda} = \lambda_1/6$ produces the worst performances for $N = 5$; this could be due to the non-convexity of $\lambda_1/6(s^{\dagger}s)^2 - \phi(s)$ for $N = 5$. Notice that, the achieved SIR values improve as N increases, according to the higher degrees of freedom available at the design stage.
In Figure 10.3, we plot the contour map of the ambiguity function of the synthesized code for $N = 25$.

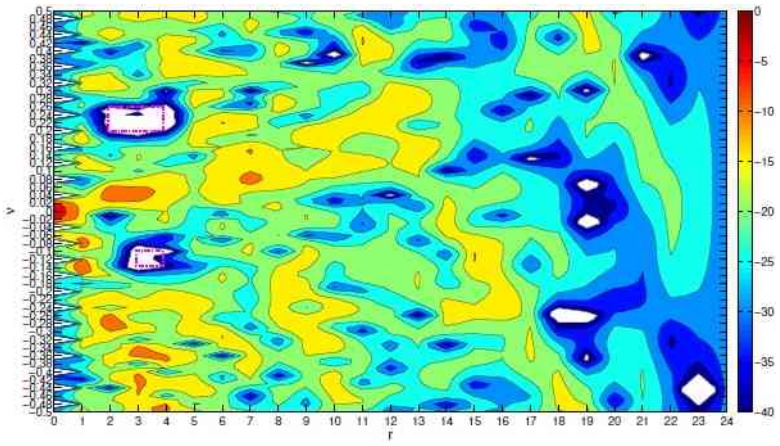

Figure 10.3: Ambiguity function, in dB, of the synthesized transmission code $s^{\star}$ for $N = 25$ (also in fuchsia the assumed interfering regions).

This figure highlights the capability of the proposed algorithm to suitably shape the ambiguity function.

Chapter 11

Conclusion and Discussion

This dissertation discusses polynomial optimization from three different perspectives: the structures of polynomial functions, the efficient algorithms to solve polynomial optimization and its application in solving some real engineering problems. On the structural part, we present a hierarchical relationship for six convex quartic forms, the polynomial sized representation of Hilbert's identity, the probability bound for polynomial function in random variables and the nice properties of matrix-rank for even order tensors. On the algorithm design part, we propose the randomized approximation algorithm, the alternating direction method of multipliers (ADMM) and the maximum block improvement (MBI) algorithm. Moreover, our theoretical analysis provides the worst case performance guarantee for all the approximation algorithms in this thesis. On the other hand, the ADMM and MBI methods are considered to be more practical, since they have been used in this thesis to solve real-sized problems arising from image processing, radar waveforms design and statistics, and the effectiveness (and in fact high performance) of these two approaches have been confirmed by our numerical results.

Furthermore, we also would like to highlight some new findings in this thesis. We have proposed two new concepts. One is the so-called k-wise zero correlation, which connects the representation of the polynomial function to some fundamental issue in probability theory, and the other one is the matrix-rank for even order tensors, which leads to the new and unexplored formulation of tensor PCA problem. Based on this new formulation, we are able to first address the polynomial optimization problem through the low rank technique, through which the global optimality of the candidate solution

is very likely and can be easily verified if it occurs. Besides, we manage to answer an open question proving that computing the matrix $2 \mapsto 4$ problem is NP-hard, which is, in fact a bi-product of the polynomial sized representation of Hilbert's identity.

Most of the results presented in this thesis are based on our research papers [94, 98, 144, 145, 52, 100, 146, 147], most of which have been submitted for publication. To be specific Chapter 2 is mainly based on [146], Chapter 3 is mainly based on [144], Chapters 4, 8, 9 are mainly based on [52, 147], Chapters 5 and 6 are mainly based on [98], Chapter 7 is mainly based on [145, 100], and Chapter 10 is mainly based on [94].

There are definitely several possible directions to advance the study in this dissertation. In Chapter 3, we find the polynomial sized presentation of Hilbert's identity when $d = 2$. It is important to find polynomial sized presentation for any d and see if this result can help to prove computing matrix $2 \mapsto 2^d$ norm is NP-hard. In Chapter 4 the bound between CP-rank and matrix-rank seems loose, which is $O(n^2)$. So a sharper bound may be possible. In Chapter 5, the probability inequalities hold only for L_1 or L_2 norm of certain tensor. It is also interesting to find similar probability inequalities in terms of the general L_p norm. The numerical results show that our approach in Chapter 8 is very likely to return a global optimal solution of tensor PCA problem. Therefore, the last one of our future plans is to provide a theoretical foundation for this approach.

References

[1] A. Barmpoutis, B. Jian, B. C. Vemuri, and T. M. Shepherd. Symmetric positive 4 th order tensors & their estimation from diffusion weighted mri. In *Information processing in medical imaging*, pages 308–319. Springer, 2007.

[2] 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 fiber directions from the orientation distribution function in diffusion mri. In *Computational Diffusion MRI Workshop (CDMRI08), New York*, 2008.

[3] S. Zhang, K. Wang, B. Chen, and X. Huang. A new framework for co-clustering of gene expression data. In *Pattern Recognition in Bioinformatics*, pages 1–12. Springer, 2011.

[4] B. Mariere, Z.-Q. Luo, and T. N. Davidson. Blind constant modulus equalization via convex optimization. *Signal Processing, IEEE Transactions on*, 51(3):805–818, 2003.

[5] L. Qi and K. L. Teo. Multivariate polynomial minimization and its application in signal processing. *Journal of Global Optimization*, 26(4):419–433, 2003.

[6] S. Weiland and F. van Belzen. Singular value decompositions and low rank approximations of tensors. *Signal Processing, IEEE Transactions on*, 58(3):1171–1182, 2010.

[7] S. Soare, J. W. Yoon, and O. Cazacu. On the use of homogeneous polynomials to develop anisotropic yield functions with applications to sheet forming. *International Journal of Plasticity*, 24(6):915–944, 2008.

[8] C. A. Micchelli and P. A. Olsen. Penalized maximum likelihood estimation methods, the baum welch algorithm and diagonal balancing of symmetric matrices for the training of acoustic models in speech recognition, April 16 2002. US Patent 6,374,216.

[9] T. Aittomaki and V. Koivunen. Beampattern optimization by minimization of quartic polynomial. In *Statistical Signal Processing, 2009. SSP'09. IEEE/SP 15th Workshop on*, pages 437–440. IEEE, 2009.

[10] P. M. Pardalos and S. A. Vavasis. Open questions in complexity theory for numerical optimization. *Mathematical Programming*, 57(1):337–339, 1992.

[11] A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and J. Tsitsiklis. Np-hardness of deciding convexity of quartic polynomials and related problems. *Mathematical Programming*, 137(1-2):453–476, 2013.

[12] J. Sturm and S. Zhang. On cones of nonnegative quadratic functions. *Mathematics of Operations Research*, 28(2):246–267, 2003.

[13] J. F Luo, Z.-Q.and Sturm and S. Zhang. Multivariate nonnegative quadratic mappings. *SIAM Journal on Optimization*, 14(4):1140–1162, 2004.

[14] D. Hilbert. Über die darstellung definiter formen als summe von formenquadraten. *Mathematische Annalen*, 32(3):342–350, 1888.

[15] J. W. Helton and J. Nie. Semidefinite representation of convex sets. *Mathematical Programming*, 122(1):21–64, 2010.

[16] B. Reznick. Uniform denominators in hilbert's seventeenth problem. *Mathematische Zeitschrift*, 220:75–97, 1995.

[17] M.B. Nathanson. *Additive Number Thoery.* The Classical Bases, Graduate Texts in Mathematics, 164. Springer-Verlag, New York, 1996.

[18] A. Barvinok. *A Course in Convexity*. Graduate Studies in Mathematics, 54. American Mathematical Society, 2002.

[19] A. Ben-Tal, A. Nemirovskii, and C. Roos. Robust solutions of uncertain quadratic and conic-quadratic problems. *SIAM Journal on Optimization*, 13:535–560, 2002.

[20] Y.-X. Yuan. A counter-example to a conjecture of ben-tal, nemirovski and roos. *Journal of the Operations Research Society of China*, 1, 2013.

[21] S. He, Z.-Q. Luo, J. Nie, and Zhang S. Semidefinite relaxation bounds for indefinite homogeneous quadratic optimization. *SIAM Journal on Optimization*, 19:503–523, 2008.

[22] Z.-Q. Luo and S. Zhang. A semidefinite relaxation scheme for multivariate quartic polynomial optimization with quadratic constraints. *SIAM Journal on Optimization*, 20:1716–1736, 2010.

[23] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. *Psychometrika*, 35(3):283–319, 1970.

[24] R. A. Harshman. Foundations of the parafac procedure: models and conditions for an" explanatory" multimodal factor analysis. 1970.

[25] J. Håstad. Tensor rank is NP-complete. *Journal of Algorithms*, 11:644–654, 1990.

[26] Y. Nesterov. Random walk in a simplex and quadratic optimization over convex polytopes. *CORE Discussion Paper, UCL, Louvain-la-Neuve*, 2003.

[27] J.B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Jounal on Optimization*, 23:863–884, 2002.

[28] J. B. Lasserre. Polynomials nonnegative on a grid and discrete representations. *Transactions of the American Mathematical Society*, 354:631–649, 2001.

[29] P. A. Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, 2000.

[30] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming, Series B*, 96:293–320, 2003.

[31] E. De Klerk, M. Laurent, and P. A. Parrilo. A ptas for the minimization of polynomials of fixed degree over the simplex. *Theoretical Computer Science*, 361(2):210–225, 2006.

[32] C. Ling, J. Nie, L. Qi, and Y. Ye. Biquadratic optimization over unit spheres and semidefinite programming relaxations. *SIAM Jounal on Optimization*, 20:1286–1310, 2009.

[33] S. He, Li Z., and Zhang S. Approximation algorithms for homogeneous polynomial optimization with quadratic constraints. *Mathematical Programming*, 125:353–383, 2010.

[34] S. He, Li Z., and Zhang S. General constrained polynomial optimization: An approximation approach. *Technical Report, Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong*, 2010.

[35] S. He, Li Z., and Zhang S. Approximation algorithms for discrete polynomial optimization. *Journal of the Operations Research Society of China*, 1, 2013.

[36] A. M.-C. So. Moment inequalities for sums of random matrices and their applications in optimization. *Mathematical Programming*, 130:125–151, 2011.

[37] Z. Li. *Polynomial Optimization Problems—Approximation Algorithms and Applications*. PhD thesis, The Chinese Univesrity of Hong Kong, 2011.

[38] A. Ben-Tal, A. Nemirovski, and C. Roos. Extended matrix cube theorems with applications to μ-theory in control. *Mathematics of Operations Research*, 28(3):497–523, 2003.

[39] S. Zhang and Y. Huang. Complex quadratic optimization and semidefinite programming. *SIAM Journal on Optimization*, 16:871–890, 2006.

[40] A. M.-C. So, J. Zhang, and Y. Ye. On approximating complex quadratic optimization problems via semidefinite programming relaxations. *Mathematical Programming*, 110(1):93–110, 2007.

[41] Y. Huang and S. Zhang. Approximation algorithms for indefinite complex quadratic maximization problems. *Science in China, Series A*, 53:2697–2708, 2010.

[42] A. Bhaskara and A. Vijayaraghavan. Approximating matrix p-norms. In *Annual ACM Symposium on Theory of Computing Discrete Algorithms (SODA)*, 2011.

[43] A. A. Ahmadi and P. A. Parrilo. A convex polynomial that is not sos-convex. *Mathematical Programming*, 135(1-2):275–292, 2012.

[44] G. Blekherman. Convex forms that are not sums of squares. *arXiv preprint arXiv:0910.0656*, 2009.

[45] B. Reznick. *Sums of Even Powers of Real Linear Formas*, volume 463. AMS Bookstore, 1992.

[46] P. M. Kleniati, P. Parpas, and B. Rustem. Partitioning procedure for polynomial optimization: Application to portfolio decisions with higher order moments. Technical report, 2009.

[47] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye. Semidefinite programming based algorithms for sensor network localization. *ACM Transactions on Sensor Networks (TOSN)*, 2(2):188–220, 2006.

[48] E. C. Chi and T. Kolda. On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272–1299, 2012.

[49] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. In *The Twelfth IEEE International Conference on Computer Vision*, 2009.

[50] M. Laurent. Sums of squares, moment matrices and optimization over polynomials. In *Emerging applications of algebraic geometry*, pages 157–270. Springer, 2009.

[51] A. A. Ahmadi and P. A. Parrilo. On the equivalence of algebraic conditions for convexity and quasiconvexity of polynomials. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 3343–3348. IEEE, 2010.

[52] B. Jiang, S. Ma, and S. Zhang. Tensor principal component analysis via convex optimization. *arXiv preprint arXiv:1212.2702*, 2012.

[53] B. Reznick. Some concrete aspects of hilbert's 17th problem. *Contemporary Mathematics*, 253:251–272, 2000.

[54] A. A. Ahmadi, G. Blekherman, and P. A. Parrilo. Convex ternary quartics are sos-convex. Technical report, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2011.

[55] P. J. C. Dickinson and L. Gijben. On the computational complexity of membership problems for the completely positive cone and its dual. *Mathematical Programming, submitted*, 2011.

[56] B. Lenore, F. Cucker, M. Shub, and S. Smale. *Complexity and real computation*. Springer, 1998.

[57] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120(2):479–495, 2009.

[58] S. Burer and H. Dong. Representing quadratically constrained quadratic programs as generalized copositive programs. *Operations Research Letters*, 40(3):203–206, 2012.

[59] Z. Li, S. He, and S. Zhang. *Approximation Methods for Polynomial Optimization: Models, Algorithms, and Applications*. SpringerBriefs in Optimization. Springer, New York, 2012.

[60] R. E. Burkard, E. Çela, and B. Klinz. On the biquadratic assignment problem. In *Quadratic Assignment and Related Problems: Dimacs Workshop May 20-21, 1993*, volume 16, page 117. American Mathematical Soc., 1994.

[61] R. E. Burkard and E. Çela. Heuristics for biquadratic assignment problems and their computational comparison. *European Journal of Operational Research*, 83(2):283–300, 1995.

[62] T. Mavridou, P. M. Pardalos, L. S. Pitsoulis, and M. G. C. Resende. A grasp for the biquadratic assignment problem. *European Journal of Operational Research*, 105(3):613–621, 1998.

[63] L. H. Lim. Singular values and eigenvalues of tensors: a variational approach. In *Computational Advances in Multi-Sensor Adaptive Processing, 2005 1st IEEE International Workshop on*, pages 129–132. IEEE, 2005.

[64] L. Qi. Eigenvalues of a real supersymmetric tensor. *Journal of Symbolic Computation*, 40:1302–1324, 2005.

[65] L. Qi, F. Wang, and Y. Wang. Z-eigenvalue methods for a global polynomial optimization problem. *Mathematical Programming, Series A*, 118:301–316, 2009.

[66] N. Alon, L. Babai, and A. Itai. A fast and simple randomized algorithm for the maximal independent set problem. *Journal of Algorithms*, 7:567–583, 1986.

[67] H. Karloff and Y. Mansour. On construction of k-wise independent random variables. *Combinatorica*, 17:91–107, 1997.

[68] A. Joffe. On a set of almost deterministic k-wise independent random variables. *Annals of Probability*, 2:161–162, 1974.

[69] N. Alon and A. Naor. Approximating the cut-norm via grothendieck's inequality. *SIAM Journal on Computing*, 35:787–803, 2006.

[70] N.J. Higham. Estimating the matrix p-norm. *Numerische Mathematik*, 62:539–555, 1992.

[71] D. Steinberg. *Computation of Matrix Norms with Applications to Robust Optimization*. PhD thesis, Thchnion-Israel University of Technology, 2005.

[72] J.M. Hendrickx and Olshevsky A. Matrix p-norms are np-hard to approximate if $p \neq 1, 2, \infty$. *SIAM Journal on Matrix Analysis and Applications*, 31:2802–2812, 2010.

[73] F. L. Hitchcock. *The expression of a tensor or a polyadic as a sum of products.* Institute of Technology, 1927.

[74] F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *Journal of Mathematical Physics*, 7(1):39–79, 1927.

[75] J. B. Kruskal. Rank, decomposition, and uniqueness for 3-way and n-way arrays. *Multiway data analysis*, pages 7–18, 1989.

[76] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51:455–500, 2009.

[77] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 2011.

[78] P. Comon, G. Golub, L. H. Lim, and B. Mourrain. Symmetric tensors and symmetric tensor rank. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1254–1279, 2008.

[79] B. Chen, S. He, Z. Li, and S. Zhang. Maximum block improvement and polynomial optimization. *SIAM Journal on Optimization*, 22:87–107, 2012.

[80] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.

[81] Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Softwares*, 9:141–160, 1998.

[82] Z.-Q. Luo, N.D. Sidiropoulos, P. Tseng, and S. Zhang. Approximation bounds for quadratic optimization with homogeneous quadratic constraints. *SIAM Journal on Optimization*, 18:1–28, 2007.

[83] S. Khot and A. Naor. Linear equations modulo 2 and the l_1 diameter of convex bodies. *SIAM Journal on Computing*, 38:1448–1463, 2008.

[84] S. He, Zhang J., and Zhang S. Bounding probability of small deviation: A fourth moment approach. *Mathematics of Operations Research*, 35:208–232, 2010.

[85] B. Grübaum. Partitions of mass-distributions and of convex bodies by hyperplanes. *Pacific Journal of Mathematics*, 10:1257–1271, 1960.

[86] A. Brieden, P. Gritzmann, R. Kannan, V. Klee, L. Lovász, and Simonovits M. Approximation of diameters: Randomization doesn't help. In *The 39th Annual IEEE Symposium on Foundations of Computer Science*, pages 244–251, 1998.

[87] A. Brieden, P. Gritzmann, R. Kannan, V. Klee, L. Lovász, and Simonovits M. Deterministic and randomized polynomial-time approximation of radii. *Mathematika*, 48:63–105, 2003.

[88] R. Paley and A. Zygmund. A note on analytic functions in the unit circle. *Mathematical Proceedings of the Cambridge Philosophical Society*, 28:266–272, 1932.

[89] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28:1302–1338, 2000.

[90] A. M.-C. So. Deterministic approximation algorithms for sphere constrained homogeneous polynomial optimization problems. *Mathematical Programming*, 129:357–382, 2011.

[91] M. Charikar and A. Wirth. Maximizing quadratic programs: Extending grothendieck's inequality. In *The 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 54–60, 2004.

[92] O. Toker and H. Ozbay. On the complexity of purely complex μ computation and related problems in multidimensional systems. *IEEE Transactions on Automatic Control*, 43:409–414, 1998.

[93] C. Chen and P. P. Vaidyanathan. Mimo radar waveform optimization with prior information of the extended target and clutter. *Signal Processing, IEEE Transactions on*, 57(9):3533–3544, 2009.

[94] A. Aubry, A. De Maio, B. Jiang, and S. Zhang. A cognitive design of the radar waveform range-doppler response. *Signal Processing, IEEE Transactions on, forthcoming*, 2013.

[95] J. J. Hilling and A. Sudbery. The geometric measure of multipartite entanglement and the singular values of a hypermatrix. *Journal of Mathematical Physics*, 51:072102, 2010.

[96] X. Zhang and L. Qi. The quantum eigenvalue problem and z-eigenvalues of tensors. *arXiv preprint arXiv:1205.1342*, 2012.

[97] L. Qi. Eigenvalues and invariants of tensors. *Journal of Mathematical Analysis and Applications*, 325(2):1363–1377, 2007.

[98] S. He, Jiang B., Li Z., and Zhang S. Probability bounds for polynomial functions in random variables. Technical report, Technical Report. Department of Industrial and Systems Engineering, University of Minnesota, Minneapolis, 2012.

[99] K. Hou and A. M.-C. So. Hardness and approximation results for *l_p*-ball constrained homogeneous polynomial optimization problems. *arXiv preprint arXiv:1210.8284*, 2012.

[100] B. Jiang, Z. Li, and S. Zhang. Conjugate symmetric complex tensors and applications. Technical report, Department of Industrial and Systems Engineering, University of Minnesota, 2013.

[101] H. Wang and N. Ahuja. Compact representation of multidimensional data using tensor rank-one decomposition. In *Proceedings of the Pattern Recognition, 17th International Conference on ICPR*, 2004.

[102] 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*, pages 1–8, 2008.

[103] L. Qi, G. Yu, and E. X. Wu. Higher order positive semi-definite diffusion tensor imaging. *SIAM Journal on Imaging Sciences*, pages 416–433, 2010.

[104] S. Hu and L. Qi. Algebraic connectivity of an even uniform hypergraph. *Journal of Combinatorial Optimization*, 24(4):564–579, 2012.

[105] W. Li and M. Ng. Existence and uniqueness of stationary probability vector of a transition probability tensor. Technical report, Department of Mathematics, The Hong Kong Baptist University, March 2011.

[106] E. Kofidis and P. A. Regalia. On the best rank-1 approximation of higher-order supersymmetric tensors. *SIAM Journal on Matrix Analysis and Applications*, 23:863–884, 2002.

[107] T. G. Kolda and J. R. Mayo. Shifted power method for computing tensor eigenpairs. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1095–1124, 2011.

[108] R. Tomioka, T. Suzuki, K. Hayashi, and H. Kashima. Statistical performance of convex tensor decomposition. *Advances in Neural Information Processing Systems (NIPS)*, page 137, 2011.

[109] V. Chandrasekaran, P. A. Recht, B.and Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.

[110] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52:489–509, 2006.

[111] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306, 2006.

[112] B. Recht, M. Fazel, and P. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[113] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9:717–772, 2009.

[114] E. J. Candès and T. Tao. The power of convex relaxation: near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2009.

[115] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. *http://cvxr.com/cvx*, May 2010.

[116] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1993.

[117] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996.

[118] J. Douglas and H. H. Rachford. On the numerical solution of the heat conduction problem in 2 and 3 space variables. *Transactions of the American Mathematical Society*, 82:421–439, 1956.

[119] D. H. Peaceman and H. H. Rachford. The numerical solution of parabolic elliptic differential equations. *SIAM Journal on Applied Mathematics*, 3:28–41, 1955.

[120] P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16:964–979, 1979.

[121] M. Fortin and R. Glowinski. *Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems*. North-Holland Pub. Co., 1983.

[122] R. Glowinski and P. Le Tallec. *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*, volume 9. SIAM, 1989.

[123] J. Eckstein. *Splitting methods for monotone operators with applications to parallel optimization*. PhD thesis, Massachusetts Institute of Technology, 1989.

[124] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55:293–318, 1992.

[125] D. Gabay. Applications of the method of multipliers to variational inequalities. In M. Fortin and R. Glowinski, editors, *Augmented Lagrangian Methods: Applications to the Solution of Boundary Value Problems*. North-Hollan, Amsterdam, 1983.

[126] J. Yang and Y. Zhang. Alternating direction algorithms for ℓ_1 problems in compressive sensing. *SIAM Journal on Scientific Computing*, 33(1):250–278, 2011.

[127] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.

[128] T. Goldstein and S. Osher. The split Bregman method for L1-regularized problems. *SIAM Journal on Imaging Sciences*, 2:323–343, 2009.

[129] M. Tao and X. Yuan. Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization*, 21:57–81, 2011.

[130] X. Yuan. Alternating direction methods for sparse covariance selection. *Journal of Scientific Computing*, 51:261–273, 2012.

[131] K. Scheinberg, S. Ma, and D. Goldfarb. Sparse inverse covariance selection via alternating linearization methods. In *NIPS*, 2010.

[132] S. Ma. Alternating direction method of multipliers for sparse principal component analysis. Technical report, 2011.

[133] Z. Wen, D. Goldfarb, and W. Yin. Alternating direction augmented Lagrangian methods for semidefinite programming. *Mathematical Programming Computation*, 2:203–230, 2010.

[134] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 2011.

[135] S. Ma, D. Goldfarb, and L. Chen. Fixed point and Bregman iterative methods for matrix rank minimization. *Mathematical Programming Series A*, 128:321–353, 2011.

[136] D. Henrion, J. B. Lasserre, and J. Loefberg. GloptiPoly 3: Moments, optimization and semidefinite programming. *Optimization Methods and Software*, 24:761–779, 2009.

[137] D. Goldfarb and Z. Qin. Robust low-rank tensor recovery: Models and algorithms. Technical report, Department of Industrial Engineering and Operations Research, Columbia University, 2012.

[138] S. Ma, D. Goldfarb, and K. Scheinberg. Fast alternating linearization methods for robust principal component analysis. Technical report, Department of Industrial Engineering and Operations Research, Columbia University, 2010.

[139] S. Sussman. Least-square synthesis of radar ambiguity functions. *Information Theory, IRE Transactions on*, 8(3):246–254, 1962.

[140] P. Stoica, H. He, and J. Li. New algorithms for designing unimodular sequences with good correlation properties. *Signal Processing, IEEE Transactions on*, 57(4):1415–1425, 2009.

[141] A. Aubry, A. De Maio, A. Farina, and M. Wicks. Knowledge-aided (potentially cognitive) transmit signal and receive filter design in signal-dependent clutter. *Aerospace and Electronic Systems, IEEE Transactions on*, 49(1):93–117, 2013.

[142] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[143] B. Chen. *Optimization with Block Variables: Theory and Applications*. PhD thesis, The Chinese Univesrity of Hong Kong, 2012.

[144] B. Jiang, S. He, Z. Li, and S. Zhang. Moments tensors, hilbert's identity, and k-wise uncorrelated random variables. Technical report, Department of Industrial and Systems Engineering, University of Minnesota, 2012.

[145] B. Jiang, Z. Li, and S. Zhang. Approximation methods for complex polynomial optimization. Technical report, Department of Industrial and Systems Engineering, University of Minnesota, 2012.

[146] B. Jiang, Z. Li, and S. Zhang. On cones of nonnegative quartic forms. Technical report, Department of Industrial and Systems Engineering, University of Minnesota, 2013.

[147] B. Jiang, S. Ma, and S. Zhang. On matrix-rank of even order tensor and low-rank tensor optimization. Technical report, Department of Industrial and Systems Engineering, University of Minnesota, 2013.