Math. Program., Ser. B (2010) 125:353–383
DOI 10.1007/s10107-010-0409-z

FULL LENGTH PAPER

Approximation algorithms for homogeneous polynomial optimization with quadratic constraints

Simai He · Zhening Li · Shuzhong Zhang

Received: 30 April 2009 / Accepted: 30 August 2010 / Published online: 17 September 2010

Abstract In this paper, we consider approximation algorithms for optimizing a generic multi-variate homogeneous polynomial function, subject to homogeneous quadratic constraints. Such optimization models have wide applications, e.g., in signal processing, magnetic resonance imaging (MRI), data training, approximation theory, and portfolio selection. Since polynomial functions are non-convex, the problems under consideration are all NP-hard in general. In this paper we shall focus on polynomial-time approximation algorithms. In particular, we first study optimization of a multi-linear tensor function over the Cartesian product of spheres. We shall propose approximation algorithms for such problem and derive worst-case performance ratios, which are shown to be dependent only on the dimensions of the model. The methods are then extended to optimize a generic multi-variate homogeneous polynomial function with spherical constraint. Likewise, approximation algorithms are proposed with provable approximation performance ratios. Furthermore, the constraint set is relaxed to be an intersection of co-centered ellipsoids; namely, we consider maximization of a homogeneous polynomial over the intersection of ellipsoids centered at the origin, and propose polynomial-time approximation algorithms with provable worst-case performance ratios. Numerical results are reported, illustrating the effectiveness of the approximation algorithms studied.

S. He
Department of Management Sciences, City University of Hong Kong, Kowloon Tong, Hong Kong
e-mail: simaihe@cityu.edu.hk

Z. Li · S. Zhang (✉)
Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong, Shatin, Hong Kong
e-mail: zhang@se.cuhk.edu.hk

Z. Li
e-mail: zheningli@gmail.com

Keywords Multi-linear tensor form · Polynomial function optimization · Approximation algorithm

Mathematics Subject Classification (2000) 15A69 · 90C26 · 90C59

1 Introduction

Maximizing (or minimizing) a polynomial function, subject to some suitable polynomial constraints, is a fundamental model in optimization. As such, it is widely used in practice—just to name a few examples: signal processing [26,41], speech recognition [28], biomedical engineering [1,8], material science [44], investment science [4, 15,25,27,34,37], quantum mechanics [3,11], and numerical linear algebra [33,39,40]. It is basically impossible to list, even very partially, the success stories of polynomial optimization, simply due to its sheer size in the literature. To motivate our study, below we shall nonetheless mention a few sample applications to illustrate the usefulness of polynomial optimization. Polynomial optimization has immediate applications in signal and image processing, e.g. Magnetic Resonance Imaging (MRI). As an example, Ghosh et al. [8] formulated a fiber detection problem in Diffusion MRI by maximizing a homogenous polynomial function, subject to a spherical constraint. In this particular case, the order of the polynomial may be high, and the problem is non-convex. Barmpoutis et al. [1] presented a case for the 4th order tensor approximation in Diffusion Weighted MRI. In statistics, Micchelli and Olsen [28] considered a maximum-likelihood estimation model in speech recognition. In Maricic et al. [26], a quartic polynomial model was proposed for blind channel equalization in digital communication, and in Qi and Teo [41], a study of global optimization was conducted for high order polynomial minimization models arising from signal processing. Polynomial functions also have wide applications in material sciences. As an example, Soare, Yoon, and Cazacu [44] proposed some 4th, 6th and 8th order homogeneous polynomials to model the plastic anisotropy of orthotropic sheet metal. In quantum physics, for example, Dahl et al. [3] proposed a polynomial optimization model to verify whether a physical system is entangled or not, which is an important problem in quantum physics. Gurvits [11] showed that the entanglement verification is NP-hard in general. In fact, the model discussed in [3] is related to the nonnegative quadratic mappings studied by Luo, Sturm and Zhang [23]. Homogeneous polynomials, which we shall focus on in this paper, play an important role in approximation theory; see e.g. two recent papers by Kroó and Szabados [17] and Varjú [47]. Essentially their results state that the homogeneous polynomial functions are fairly 'dense' among continuous functions in a certain well-defined sense. One interesting application of homogeneous polynomial optimization is related to the so-called eigenvalues of tensors; see Qi [39,40], and Ni et al. [33]. Investment models involving more than the first two moments (for instance to include the skewness and the kurtosis of the investment returns) have been another source of inspiration underlying polynomial optimization. Mandelbrot and Hudson [25] made a strong case against a 'normal view' of the investment returns. The use of higher moments in portfolio selection becomes quite necessary. Along that line, several authors proposed investment models incorporating the higher moments;

e.g. De Athayde and Flôre [4], Prakash, Chang and Pactwa [37], Jondeau and Rockinger [15]. Moreover, Parpas and Rustem [34] and Maringer and Parpas [27] proposed diffusion-based methods to solve the non-convex polynomial optimization models arising from portfolio selection involving higher moments.

On the front of solution methods, the search for general and efficient algorithms for polynomial optimization has been a priority for many mathematical optimizers. Indeed, generic solution methods based on nonlinear programming and global optimization have been studied and tested; see e.g. Qi [38], and Qi et al. [42] and the references therein. An entirely different (and systematic) approach based on the so-called *Sum of Squares* (SOS) was proposed by Lasserre [18,19], and Parrilo [35,36]. The SOS approach has a strong theoretical appeal, since it can in principle solve any general polynomial optimization model to any given accuracy, by resorting to a (possibly large) Semidefinite Program (SDP). For univariate polynomial optimization, Nesterov [31] showed that the SOS approach in combination with the SDP solution has a polynomial-time complexity. In general, however, the SDP problems required to be solved by the SOS approach may grow very large. At any rate, thanks to the recently developed efficient SDP solvers (cf. e.g. SeDuMi of Jos Sturm [45], SDPT3 of Toh et al. [46], and SDPA of Fujisawa et al. [7]) the SOS approach appears to be attractive. Henrion and Lasserre [13] developed a specialized tool known as GloptiPoly (the latest version, GloptiPoly 3, can be found in Henrion et al. [14]) for finding a global optimal solution for a polynomial function based on the SOS approach. For an overview on the recent theoretical developments, we refer to the excellent survey by Laurent [20].

In most cases, polynomial optimization is NP-hard, even for very special ones, such as maximizing a cubic polynomial over a sphere (cf. Nestorov [32]). The reader is referred to De Klerk [5] for a survey on the computational complexity issues of polynomial optimization over some simple constraint sets. In the case that the constraint set is a simplex and the polynomial has a fixed degree, it is possible to derive Polynomial-Time Approximation Schemes (PTAS); see De Klerk et al. [6], albeit the result is viewed mostly as a theoretical one. Almost in all practical situations, the problem is difficult to solve, theoretically as well as numerically. The intractability of general polynomial optimization therefore motivates the search for approximate solutions. Luo and Zhang [24] proposed an approximation algorithm for optimizing a homogenous quartic polynomial under ellipsoidal constraints. That approach is similar, in its spirit, to the seminal SDP relaxation and randomization method of Goemans and Williamson [9], although the objective function in [9] is quadratic. Note that the approach in [9] has been generalized subsequently by many authors, including Nesterov [30], Ye [48,49], Nemirovski et al. [29], Zhang [50], Zhang and Huang [51], Luo et al. [22], and He et al. [12]. All these works deal with quadratic objective functions. Luo and Zhang [24] considered quartic optimization, and showed that optimizing a quartic polynomial over the intersection of some co-centered ellipsoids is essentially equivalent to its (quadratic) SDP relaxation problem, which is itself also NP-hard; however, this gives a handle on the design of approximation algorithms with provable worst-case approximation ratios. Ling et al. [21] considered a special quartic optimization model. Basically, the problem is to minimize a biquadratic function over two spherical constraints. In [21], approximate solutions as well as exact solutions using the SOS approach are considered. The approximation bounds in [21] are indeed comparable to the bound in [24], although they are dealing with two different models. The current paper is concerned with general homogeneous polynomial optimization models, and we shall focus on approximate solutions. Our goal is to present a rather general scheme which will enable us to obtain approximate solutions with guaranteed worst-case performance ratios. To present the results, we shall start in the next section with some technical preparations.

2 Models, notations, and the organization of the paper

Consider the following multi-linear function

$$F(x^1, x^2, \ldots, x^d) = \sum_{1 \le i_1 \le n_1, 1 \le i_2 \le n_2, \ldots, 1 \le i_d \le n_d} a_{i_1 i_2 \ldots i_d} x^1_{i_1} x^2_{i_2} \cdots x^d_{i_d},$$

where $x^k \in \Re^{n_k}, k = 1, 2, \ldots, d$. In the shorthand notation we shall denote $M = (a_{i_1 i_2 \ldots i_d}) \in \Re^{n_1 \times n_2 \times \cdots \times n_d}$ to be a d-th order tensor. Closely related to the tensor form M is a general homogeneous polynomial function $f(x)$ of degree d, where $x \in \Re^n$. We call the tensor form M *super-symmetric* (see [16]) if $a_{i_1 i_2 \ldots i_d}$ is invariant under all permutations of $\{i_1, i_2, \ldots, i_d\}$. As any homogeneous quadratic function uniquely determines a symmetric matrix, a given homogeneous polynomial function $f(x)$ of degree d also uniquely determines a super-symmetric tensor form. In particular, suppose that

$$f(x) = \sum_{1 \le i_1 \le i_2 \le \cdots \le i_d \le n} b_{i_1 i_2 \ldots i_d} x_{i_1} x_{i_2} \cdots x_{i_d}.$$

Let the super-symmetric tensor form be $M = (a_{i_1 i_2 \ldots i_d}) \in \Re^{n^d}$, with $a_{i_1 i_2 \ldots i_d} \equiv b_{i_1 i_2 \ldots i_d} / |P(i_1, i_2, \ldots, i_d)|$, where $|P(i_1, i_2, \ldots, i_d)|$ is the number of distinctive permutations of the indices $\{i_1, i_2, \ldots, i_d\}$. Let F be the multi-linear function defined by the super-symmetric tensor M. Then $f(x) = F(\underbrace{x, x, \ldots, x}_{d})$, and this super-symmetric tensor representation is indeed unique. The Frobenius norm of the tensor form M is naturally defined as

$$\|M\| := \sqrt{\sum_{1 \le i_1 \le n_1, 1 \le i_2 \le n_2, \ldots, 1 \le i_d \le n_d} a^2_{i_1 i_2 \ldots i_d}}.$$

Throughout this paper, we shall denote F to be a multi-linear function defined by a tensor form, and f to be a homogenous polynomial function; without loss of generality we assume that $n_1 \le n_2 \le \cdots \le n_d$.

In this paper we shall study optimization of a generic polynomial function, subject to two types of constraints: **(A)** (Euclidean) spherical constraints; **(B)** general ellipsoidal constraints. To be specific, we consider the following models:

Table 1 Organization of the paper and the approximation results

Subsection	Model	Approximation performance ratio
3.1	$(A^1_{\max})$	$(n_1 n_2 \cdots n_{d-2})^{-\frac{1}{2}}$
3.2	$(A^2_{\max})$	$d!d^{-d}n^{-\frac{d-2}{2}}$
4.1	$(B^1_{\max})$	$\Omega\left(\left(\sqrt{n_1 n_2 \cdots n_{d-2}}\ (\log \max_{1\le k\le d} m_k)^{d-1}\right)^{-1}\right)$
4.2	$(B^2_{\max})$	$\Omega\left(d!d^{-d}\left(n^{\frac{d-2}{2}}\log^{d-1} m\right)^{-1}\right)$

$$
\begin{array}{lll}
(A^1_{\max}) & \max & F(x^1, x^2, \ldots, x^d) \\
 & \text{s.t.} & \|x^k\| = 1,\ x^k \in \Re^{n_k}, \quad k = 1, 2, \ldots, d; \\
(A^2_{\max}) & \max & f(x) = F(\underbrace{x, x, \ldots, x}_{d}) \\
 & \text{s.t.} & \|x\| = 1,\ x \in \Re^n; \\
(B^1_{\max}) & \max & F(x^1, x^2, \ldots, x^d) \\
 & \text{s.t.} & (x^k)^{\mathrm{T}} Q^k_{i_k} x^k \le 1, \quad k = 1, 2, \ldots, d, \quad i_k = 1, 2, \ldots, m_k, \\
 & & x^k \in \Re^{n_k}, \quad k = 1, 2, \ldots, d; \\
(B^2_{\max}) & \max & f(x) = F(\underbrace{x, x, \ldots, x}_{d}) \\
 & \text{s.t.} & x^{\mathrm{T}} Q_i x \le 1, \quad i = 1, 2, \ldots, m, \\
 & & x \in \Re^n.
\end{array}
$$

The models and results of type **(A)** are presented in Sect. 3; the models and results of type **(B)** are presented in Sect. 4. To put the matters in perspective, Table 1 summarizes the organization of the paper and the approximation results.

As a convention, the notation $\Omega(\lambda)$ should be read as: "at least in the order of λ". Since Table 1 is concerned with approximation ratios, we shall understand $\Omega(\infty)$ as a universal constant in the interval $(0, 1]$.

In case $d = 2$, Problem $(B^2_{\max})$ is precisely the same QCQP problem considered by Nemirovski et al. [29], and our approximation ratio reduces to that of [29]. For $d > 2$, there are unfortunately not many results in the literature on approximation algorithms for optimizing higher degree (larger than 2) polynomial functions with quadratic constraints. Among the existing ones, the most noticeable recent papers include Ling et al. [21], and Luo and Zhang [24]. Both papers consider optimization of a quartic polynomial function subject to one or two quadratic constraints, and (quadratic) semidefinite programming relaxation is proposed and analyzed in proving the approximation performance ratios. The relative ratios in [21] and [24] are in the order of $\Omega(1/n^2)$. The algorithms in the current paper solve (approximately) general homogenous polynomials of degree d, with arbitrary number of constraints. If $d = 4$ and there is only one quadratic constraint, our relative approximation ratio is $\Omega(1/n)$, which is better than the results in [21] and [24]. Very recently, in a working paper

Zhang et al. [52] study the cubic spherical optimization problems, which is a special case of our Problem $(A^1_{\max})$ with $d = 3$. Their approximation ratio is $\Omega(1/\sqrt{n})$, which is the same as ours, when specialized to the case $d = 3$.

3 Polynomial optimization with spherical constraints

3.1 Multi-linear function optimization with spherical constraints

Let us first consider the problem

$$\begin{array}{lll}(A^1_{\max}) & \max & F(x^1, x^2, \ldots, x^d) \\ & \text{s.t.} & \|x^k\| = 1, \quad x^k \in \Re^{n_k}, \quad k = 1, 2, \ldots, d,\end{array}$$

where $n_1 \le n_2 \le \cdots \le n_d$. Suppose that M is the tensor form associated with the multi-linear function F. It is clear that the optimal value of the above problem, $v(A^1_{\max})$, is positive, unless M is a zero-tensor. A special case of Problem $(A^1_{\max})$ is worth noting, and we shall come back to this point later.

Proposition 1 *If $d = 2$, then Problem $(A^1_{\max})$ can be solved in polynomial-time, with $v(A^1_{\max}) \ge \|M\|/\sqrt{n_1}$.*

Proof The problem is essentially $\max_{\|x\|=\|y\|=1} x^{\mathrm{T}} M y$. For any fixed y, the corresponding optimal x must be $My/\|My\|$ due to the Cauchy-Schwartz inequality, and accordingly,

$$x^{\mathrm{T}} M y = \left(\frac{My}{\|My\|}\right)^{\mathrm{T}} My = \|My\| = \sqrt{y^{\mathrm{T}} M^{\mathrm{T}} M y}.$$

Thus the problem is equivalent to $\max_{\|y\|=1} y^{\mathrm{T}} M^{\mathrm{T}} M y$, whose solution is the largest eigenvalue and a corresponding eigenvector of the positive semidefinite matrix $M^{\mathrm{T}} M$. Denote $\lambda_{\max}(M^{\mathrm{T}} M)$ to be the largest eigenvalue of $M^{\mathrm{T}} M$, and we have

$$\lambda_{\max}(M^{\mathrm{T}} M) \ge \mathrm{tr}\,(M^{\mathrm{T}} M)/\mathrm{rank}\,(M^{\mathrm{T}} M) \ge \|M\|^2/n_1,$$

which implies $v(A^1_{\max}) = \sqrt{\lambda_{\max}(M^{\mathrm{T}} M)} \ge \|M\|/\sqrt{n_1}$. □

However, for any $d \ge 3$, Problem $(A^1_{\max})$ becomes NP-hard.

Proposition 2 *If $d = 3$, then Problem $(A^1_{\max})$ is NP-hard.*

Proof We first quote a result of Nesterov [32], which states that

$$\begin{array}{ll}\max & \sum_{k=1}^{m} (x^{\mathrm{T}} A_k x)^2 \\ \text{s.t.} & \|x\| = 1, \quad x \in \Re^n\end{array}$$

is NP-hard. Now, in a special case $d = 3$ and $n_1 = n_2 = n_3 = n$, the objective function of Problem $(A^1_{\max})$ can be written as

$$F(x, y, z) = \sum_{i,j,k=1}^{n} a_{ijk} x_i y_j z_k = \sum_{k=1}^{n} z_k \left(\sum_{i,j=1}^{n} a_{ijk} x_i y_j \right) = \sum_{k=1}^{n} z_k (x^{\mathrm{T}} A_k y),$$

where matrix $A_k \in \Re^{n \times n}$ with its (i, j)-th entry being a_{ijk} for $k = 1, 2, \ldots, n$. By the Cauchy-Schwartz inequality, Problem $(A^1_{\max})$ is equivalent to

$$\begin{aligned} &\max \sum_{k=1}^{n} (x^{\mathrm{T}} A_k y)^2 \\ &\text{s.t.} \quad \|x\| = \|y\| = 1, \; x, \; y \in \Re^n. \end{aligned}$$

We need only to show that the optimal value of the above problem is always attainable at $x = y$. To see why, denote $(\bar{x}, \bar{y})$ to be any optimal solution pair, with optimal value v^*. If $\bar{x} = \pm\bar{y}$, then the claim is true; otherwise, we may suppose that $\bar{x} + \bar{y} \neq 0$. Let us denote $\bar{w} := (\bar{x} + \bar{y})/\|\bar{x} + \bar{y}\|$. Since $(\bar{x}, \bar{y})$ must be a KKT point, there exist (λ, μ) such that

$$\begin{cases} \sum_{k=1}^{n} \bar{x}^{\mathrm{T}} A_k \bar{y} \; A_k \bar{y} = \lambda \bar{x} \\ \sum_{k=1}^{n} \bar{x}^{\mathrm{T}} A_k \bar{y} \; A_k \bar{x} = \mu \bar{y}. \end{cases}$$

Pre-multiplying $\bar{x}^{\mathrm{T}}$ to the first equation and $\bar{y}^{\mathrm{T}}$ to the second equation yield $\lambda = \mu = v^*$. Summing up the two equations, pre-multiplying $\bar{w}^{\mathrm{T}}$, and then scaling, lead us to

$$\sum_{k=1}^{n} \bar{x}^{\mathrm{T}} A_k \bar{y} \; \bar{w}^{\mathrm{T}} A_k \bar{w} = v^*.$$

By applying the Cauchy-Schwartz inequality to the above equality, we have

$$v^* \leq \left(\sum_{k=1}^{n} (\bar{x}^{\mathrm{T}} A_k \bar{y})^2 \right)^{1/2} \left(\sum_{k=1}^{n} (\bar{w}^{\mathrm{T}} A_k \bar{w})^2 \right)^{1/2} = \sqrt{v^*} \left(\sum_{k=1}^{n} (\bar{w}^{\mathrm{T}} A_k \bar{w})^2 \right)^{1/2},$$

which implies that $(\bar{w}, \bar{w})$ is also an optimal solution. The problem is then reduced to Nesterov's quartic model, and its NP-hardness thus follows. □

We remark that the above hardness result is also shown independently in [52]. In the remainder of this subsection, we shall focus on approximation algorithms for general Problem $(A^1_{\max})$. To illustrate the main idea of the algorithms, let us first work with the case $d = 3$, i.e.

$$\begin{aligned} (\bar{A}^1_{\max}) \; \max \quad & F(x, y, z) = \sum_{1 \leq i \leq n_1, 1 \leq j \leq n_2, 1 \leq k \leq n_3} a_{ijk} x_i y_j z_k \\ \text{s.t.} \quad & \|x\| = \|y\| = \|z\| = 1, \\ & x \in \Re^{n_1}, \quad y \in \Re^{n_2}, \quad z \in \Re^{n_3}. \end{aligned}$$

Denote $W = xy^{\mathrm{T}}$, and we have

$$\|W\|^2 = \mathrm{tr}\,(WW^{\mathrm{T}}) = \mathrm{tr}\,(xy^{\mathrm{T}}yx^{\mathrm{T}}) = \mathrm{tr}\,(x^{\mathrm{T}}xy^{\mathrm{T}}y) = \|x\|^2\|y\|^2 = 1.$$

Problem $(\bar{A}^1_{\max})$ can now be relaxed to

$$\begin{array}{ll} \max & F(W,z) = \sum_{1\le i\le n_1, 1\le j\le n_2, 1\le k\le n_3} a_{ijk} W_{ij} z_k \\ \text{s.t.} & \|W\| = \|z\| = 1, \\ & W \in \Re^{n_1\times n_2}, \quad z \in \Re^{n_3}. \end{array}$$

Notice that the above problem is exactly Problem $(A^1_{\max})$ with $d = 2$, which can be solved in polynomial-time by Proposition 1. Denote its optimal solution to be $(\hat{W}, \hat{z})$. Clearly $F(\hat{W}, \hat{z}) \ge v(\bar{A}^1_{\max})$. The key step is to recover solution $(\hat{x}, \hat{y})$ from the matrix $\hat{W}$. Below we shall introduce two basic decomposition routines: one is based on randomization and the other on eigen-decomposition. They play a fundamental role in our proposed algorithms; all solution methods to be developed later rely on these two routines as a basis.

DR (Decomposition Routine) 1

- *Input: matrices* $M, W \in \Re^{n_1\times n_2}$ *with* $\|W\| = 1$.
- *Construct*

$$\tilde{W} = \begin{bmatrix} I_{n_1} & W \\ W^{\mathrm{T}} & W^{\mathrm{T}}W \end{bmatrix} \succeq 0.$$

- *Randomly generate*

$$\binom{\xi}{\eta} \sim \mathcal{N}(0_{n_1+n_2}, \tilde{W})$$

and repeat if necessary, until $\xi^{\mathrm{T}}M\eta \ge M \bullet W$ *and* $\|\xi\|\|\eta\| \le O(\sqrt{n_1})$.
- *Output:* $(x, y) = (\xi/\|\xi\|, \eta/\|\eta\|)$.

Now, let $M = F(\cdot, \cdot, \hat{z})$ and $W = \hat{W}$ in applying the above decomposition routine. For the randomly generated (ξ, η), we have

$$\mathsf{E}[F(\xi, \eta, \hat{z})] = \mathsf{E}[\xi^{\mathrm{T}}M\eta] = M \bullet W = F(\hat{W}, \hat{z}).$$

He et al. [12] establish that if $f(x)$ is an homogeneous quadratic function and x is drawn from a zero-mean multi-variate normal distribution, then there is a universal constant $\theta \ge 0.03$ such that

$$\mathrm{Prob}\,\{f(x) \ge \mathsf{E}[f(x)]\} \ge \theta.$$

Since $\xi^{\mathrm{T}} M \eta$ is a homogeneous quadratic function of the normal random vector $(\xi^{\mathrm{T}}, \eta^{\mathrm{T}})^{\mathrm{T}}$, we know

$$\operatorname{Prob}\{\xi^{\mathrm{T}} M \eta \geq M \bullet W\} = \operatorname{Prob}\{F(\xi, \eta, \hat{z}) \geq \mathsf{E}[F(\xi, \eta, \hat{z})]\} \geq \theta. \tag{1}$$

Moreover, by using a property of normal random vectors (see Lemma 3.1 of [24]) we have

$$\mathsf{E}\left[\|\xi\|^2\|\eta\|^2\right] = \mathsf{E}\left[\sum_{i=1}^{n_1}\sum_{j=1}^{n_2} \xi_i^2 \eta_j^2\right] = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\left(\mathsf{E}[\xi_i^2]\mathsf{E}[\eta_j^2] + 2\mathsf{E}[\xi_i\eta_j]^2\right)$$
$$= \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\left[(\hat{W}^{\mathrm{T}}\hat{W})_{jj} + 2\hat{W}_{ij}^2\right] = (n_1+2)\operatorname{tr}(\hat{W}^{\mathrm{T}}\hat{W}) = n_1 + 2. \tag{2}$$

By applying the Markov inequality, it follows that

$$\operatorname{Prob}\{\|\xi\|^2\|\eta\|^2 \geq t\} \leq \mathsf{E}\left[\|\xi\|^2\|\eta\|^2\right]/t = (n_1+2)/t,$$

for any $t > 0$. Therefore, by the so-called union inequality for the probability of joint events, we have

$$\begin{aligned} &\operatorname{Prob}\left\{F(\xi, \eta, \hat{z}) \geq F(\hat{W}, \hat{z}),\ \|\xi\|^2\|\eta\|^2 \leq t\right\} \\ \geq\ & 1 - \operatorname{Prob}\left\{F(\xi, \eta, \hat{z}) < F(\hat{W}, \hat{z})\right\} - \operatorname{Prob}\left\{\|\xi\|^2\|\eta\|^2 > t\right\} \\ \geq\ & 1 - (1-\theta) - (n_1+2)/t = \theta/2, \end{aligned}$$

where we let $t = 2(n_1+2)/\theta$. Thus we have

$$F(x, y, \hat{z}) \geq \frac{F(\hat{W}, \hat{z})}{\sqrt{t}} \geq v(\bar{A}^1_{\max})\sqrt{\frac{\theta}{2(n_1+2)}},$$

obtaining an $\Omega(1/\sqrt{n_1})$ approximation ratio.

Below we shall present an alternative (and deterministic) decomposition routine.

DR (Decomposition Routine) 2

- *Input: matrix* $M \in \Re^{n_1 \times n_2}$.
- *Find an eigenvector* $\hat{y}$ *corresponding to the largest eigenvalue of* $M^{\mathrm{T}}M$.
- *Compute* $\hat{x} = M\hat{y}$.
- *Output:* $(x, y) = (\hat{x}/\|\hat{x}\|, \hat{y}/\|\hat{y}\|)$.

This decomposition routine literally follows the proof of Proposition 1, which tells us that $x^{\mathrm{T}}My \geq \|M\|/\sqrt{n_1}$. Thus we have

$$F(x, y, \hat{z}) = x^{\mathrm{T}}My \geq \frac{\|M\|}{\sqrt{n_1}} = \max_{\|Z\|=1} \frac{M \bullet Z}{\sqrt{n_1}} \geq \frac{M \bullet \hat{W}}{\sqrt{n_1}} = \frac{F(\hat{W}, \hat{z})}{\sqrt{n_1}} \geq \frac{v(\bar{A}^1_{\max})}{\sqrt{n_1}}.$$

The complexity for DR 1 is $O(n_1 n_2)$ (with high probability), and for DR 2 it is $O(\max\{n_1^3, n_1 n_2\})$. However DR 2 is indeed very easy to implement, and is deterministic. Both DR 1 and DR 2 lead to the following approximation result in terms of the order of the approximation ratio.

Theorem 1 *If $d = 3$, then Problem $(A^1_{\max})$ admits a polynomial-time approximation algorithm with approximation ratio $1/\sqrt{n_1}$.*

Now we proceed to the case for general d. Let $X = x^1(x^d)^{\mathrm{T}}$, and Problem $(A^1_{\max})$ can be relaxed to

$$\begin{array}{lll} (\tilde{A}^1_{\max}) & \max & F(X, x^2, x^3, \ldots, x^{d-1}) \\ & \text{s.t.} & \|x^k\| = 1, \quad x^k \in \Re^{n_k}, \quad k = 2, 3, \ldots, d-1, \\ & & \|X\| = 1, \quad X \in \Re^{n_1 \times n_d}. \end{array}$$

Clearly it is a type of Problem $(A^1_{\max})$ with degree $d - 1$. Suppose Problem $(\tilde{A}^1_{\max})$ can be solved approximately in polynomial-time with approximation ratio τ, i.e. we find $(\hat{X}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1})$ with

$$F(\hat{X}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}) \geq \tau v(\tilde{A}^1_{\max}) \geq \tau v(A^1_{\max}).$$

Observing that $F(\cdot, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \cdot)$ is an $n_1 \times n_d$ matrix, using DR 2 we shall find $(\hat{x}^1, \hat{x}^d)$ such that

$$F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^{d-1}, \hat{x}^d) \geq F(\hat{X}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1})/\sqrt{n_1} \geq (\tau/\sqrt{n_1}) v(A^1_{\max}).$$

By induction this leads to the following:

Theorem 2 *Problem $(A^1_{\max})$ admits a polynomial-time approximation algorithm with approximation ratio $\tau^A_1 := 1/\sqrt{n_1 n_2 \cdots n_{d-2}}$.*

Below we summarize the above recursive procedure to solve Problem $(A^1_{\max})$ as in Theorem 2. Remark that the approximation performance ratio of this algorithm is tight. In a special case $F(x^1, x^2, \ldots, x^d) = \sum_{i=1}^n x^1_i x^2_i \cdots x^d_i$, the algorithm can be made to return a solution with approximation ratio being exactly τ^1_A.

Algorithm 1

- *Input: d-th order tensor $M^d \in \Re^{n_1 \times n_2 \times \cdots \times n_d}$ with $n_1 \le n_2 \le \cdots \le n_d$.*
- *Rewrite M^d as $(d-1)$-th order tensor M^{d-1} by combing its first and last components into one, and place the combined component into the last one in M^{d-1}, i.e.,*

$$M^d_{i_1,i_2,\ldots,i_d} = M^{d-1}_{i_2,i_3,\ldots,i_{d-1},(i_1-1)n_d+i_d}, \quad \forall\, 1 \le i_1 \le n_1, \ldots, 1 \le i_d \le n_d.$$

- *For Problem $(A^1_{\max})$ with $(d-1)$-th order tensor M^{d-1}: if $d-1=2$, then use DR 2, with input $M = M^{d-1}$ and output $(\hat{x}^2, \hat{x}^{1,d}) = (x,y)$; otherwise obtain a solution $(\hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \hat{x}^{1,d})$ by recursion.*
- *Compute matrix $M_2 = F(\cdot, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \cdot)$ and rewrite vector $\hat{x}^{1,d}$ as a matrix $X \in \Re^{n_1 \times n_d}$.*
- *Apply either DR 1 or DR 2, with input $(M, W) = (M_2, X)$ and output $(\hat{x}^1, \hat{x}^d) = (x,y)$.*
- *Output: a feasible solution $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$.*

3.2 Homogenous polynomial optimization with spherical constraint

Suppose that $f(x)$ is a homogenous polynomial function of degree d, and consider the problem

$$\begin{array}{lll}(A^2_{\max}) & \max & f(x) \\ & \text{s.t.} & \|x\| = 1, \quad x \in \Re^n.\end{array}$$

Let F be the super-symmetric multi-linear function satisfying $F(\underbrace{x, x, \ldots, x}_{d}) = f(x)$.

Then the above polynomial optimization problem can be *relaxed* to multi-linear function optimization, as follows:

$$\begin{array}{lll}(\bar{A}^2_{\max}) & \max & F(x^1, x^2, \ldots, x^d) \\ & \text{s.t.} & \|x^k\| = 1,\ x^k \in \Re^n, \quad k = 1, 2, \ldots, d.\end{array}$$

Theorem 2 asserts that Problem $(\bar{A}^2_{\max})$ can be solved approximately with an approximation ratio $n^{-\frac{d-2}{2}}$. To establish a link between Problems $(A^2_{\max})$ and $(\bar{A}^2_{\max})$, we note the following relationship:

Lemma 1 *Suppose that $x^1, x^2, \ldots, x^d \in \Re^n$, and $\xi_1, \xi_2, \ldots, \xi_d$ are i.i.d. random variables, each takes values 1 and -1 with equal probability $1/2$. For any super-symmetric multi-linear function F of order d and function $f(x) = F(x, x, \ldots, x)$, it holds that*

$$\mathsf{E}\left[\prod_{i=1}^d \xi_i f\left(\sum_{k=1}^d \xi_k x^k\right)\right] = d!F(x^1, x^2, \ldots, x^d).$$

Proof First we observe that

$$\mathsf{E}\left[\prod_{i=1}^{d}\xi_i f\left(\sum_{k=1}^{d}\xi_k x^k\right)\right] = \mathsf{E}\left[\prod_{i=1}^{d}\xi_i \sum_{1\le k_1,k_2\dots,k_d\le d} F\left(\xi_{k_1}x^{k_1},\xi_{k_2}x^{k_2},\dots,\xi_{k_d}x^{k_d}\right)\right]$$
$$= \sum_{1\le k_1,k_2\dots,k_d\le d}\mathsf{E}\left[\prod_{i=1}^{d}\xi_i\prod_{j=1}^{d}\xi_{k_j}F\left(x^{k_1},x^{k_2},\dots,x^{k_d}\right)\right].$$

If $\{k_1, k_2, \dots, k_d\}$ is a permutation of $\{1, 2, \dots, d\}$, then

$$\mathsf{E}\left[\prod_{i=1}^{d}\xi_i\prod_{j=1}^{d}\xi_{k_j}\right] = \mathsf{E}\left[\prod_{i=1}^{d}\xi_i^2\right] = 1;$$

otherwise, there must be an index k_0 with $1 \le k_0 \le d$ and $k_0 \ne k_j$ for all $j = 1, 2, \dots, d$. In the latter case,

$$\mathsf{E}\left[\prod_{i=1}^{d}\xi_i\prod_{j=1}^{d}\xi_{k_j}\right] = \mathsf{E}\left[\xi_{k_0}\right]\mathsf{E}\left[\prod_{1\le i\le d, i\ne k_0}\xi_i\prod_{j=1}^{d}\xi_{k_j}\right] = 0.$$

Since the number of different permutations of $\{1, 2, \dots, d\}$ is $d!$, by taking into account of the super-symmetric property of F, the claimed relation follows. □

When d is odd, the identity in Lemma 1 can be rewritten as

$$d!F(x^1, x^2, \dots, x^d) = \mathsf{E}\left[\prod_{i=1}^{d}\xi_i f\left(\sum_{k=1}^{d}\xi_k x^k\right)\right] = \mathsf{E}\left[f\left(\sum_{k=1}^{d}\left(\prod_{i\ne k}\xi_i\right)x^k\right)\right].$$

Since $\xi_1, \xi_2, \dots, \xi_d$ are i.i.d. random variables taking values 1 or -1, by randomization we may find a particular binary vector $\beta = (\beta_1, \beta_2, \dots, \beta_d)$, with $\beta_i^2 = 1$ for $i = 1, 2, \dots, d$, such that

$$f\left(\sum_{k=1}^{d}\left(\prod_{i\ne k}\beta_i\right)x^k\right) \ge d!F(x^1, x^2, \dots, x^d). \tag{3}$$

(Remark that d is considered a constant parameter in this paper. Therefore, searching over all the combinations can be done, in principle, in constant time.)

Let $x' = \sum_{k=1}^{d}\left(\prod_{i\ne k}\beta_i\right)x^k$, and $\hat{x} = x'/\|x'\|$. By the triangle inequality, we have $\|x'\| \le d$, and thus

$$f(\hat{x}) \ge d!d^{-d}F(x^1, x^2, \dots, x^d).$$

Combining with Theorem 2, we have

Theorem 3 *For odd d, Problem* $(A^2_{\max})$ *admits a polynomial-time approximation algorithm with approximation ratio* $\tau_2^A := d!d^{-d}n^{-\frac{d-2}{2}}$.

If n is even, then evidently we can only speak of *relative approximation ratio*. The following algorithm applies for Problem $(A^2_{\max})$ when d is even. It is one typical case of our method for solving homogeneous polynomial optimization from multi-linear function optimization.

Algorithm 2

- *Input: d-th order super-symmetric tensor* $M^d \in \Re^{n^d}$ *and a vector* $x^0 \in \Re^n$ *with* $\|x^0\| = 1$.
- *Apply Algorithm 1 to solve Problem* $(\tilde{A}^2_{\max})$ *where function H is defined by (4), with input* M^d *and output* $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$.
- *Output: a feasible solution* $\operatorname{argmax}\left\{f(x^0);\ f\left(\frac{\sum_{i=1}^d \xi_i \hat{x}^i}{\left\|\sum_{i=1}^d \xi_i \hat{x}^i\right\|}\right), \xi_i \in \{1, -1\}\right\}$.

Theorem 4 *For even* $d \geq 4$, *Problem* $(A^2_{\max})$ *admits a polynomial-time approximation algorithm with relative approximation ratio* τ_2^A, *i.e. there exists a feasible solution* $\hat{x}$ *such that*

$$f(\hat{x}) - v(A^2_{\min}) \geq \tau_2^A \left(v(A^2_{\max}) - v(A^2_{\min})\right),$$

where $v(A^2_{\min}) := \min_{\|x\|=1} f(x)$.

Proof Denote $H(\underbrace{x, x, \ldots, x}_{d})$ to be the super-symmetric tensor form with respect to the homogeneous polynomial $h(x) = \|x\|^d = (x^{\mathrm{T}}x)^{d/2}$. Explicitly, if we denote Π to be the set of all distinctive permutations of $\{1, 2, \ldots, d\}$, then

$$H(x^1, x^2, \ldots, x^d) = \frac{1}{|\Pi|} \sum_{\{i_1, i_2, \ldots, i_d\} \in \Pi} \left((x^{i_1})^{\mathrm{T}} x^{i_2}\right)\left((x^{i_3})^{\mathrm{T}} x^{i_4}\right) \cdots \left((x^{i_{d-1}})^{\mathrm{T}} x^{i_d}\right). \tag{4}$$

For any x^k with $\|x^k\| = 1$ $(1 \leq k \leq d)$, we have $|H(x^1, x^2, \ldots, x^d)| \leq 1$ by applying the Cauchy-Schwartz inequality termwise.

Our algorithm starts by picking any fixed x^0 with $\|x^0\| = 1$. Consider the following problem

$$\begin{array}{lll} (\tilde{A}^2_{\max}) \max & F(x^1, x^2, \ldots, x^d) - f(x^0)H(x^1, x^2, \ldots, x^d) \\ \text{s.t.} & \|x^k\| = 1, \quad x \in \Re^n, \quad k = 1, 2, \ldots, d. \end{array}$$

Applying Theorem 2 we obtain a solution $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$ in polynomial-time, with

$$F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) - f(x^0)H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \geq \tilde{\tau}_1^A v(\tilde{A}^2_{\max}),$$

where $\tilde{\tau}_1^A := n^{-\frac{d-2}{2}}$. Let us first work on the case that

$$f(x^0) - v(A^2_{\min}) \le (\tilde{\tau}_1^A/4)\left(v(A^2_{\max}) - v(A^2_{\min})\right). \tag{5}$$

Since $|H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)| \le 1$, we have

$$\begin{aligned}
& F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) - v\left(A^2_{\min}\right) H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \\
&= F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) - f(x^0) H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \\
&\quad + \left(f(x^0) - v\left(A^2_{\min}\right)\right) H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \\
&\ge \tilde{\tau}_1^A v\left(\tilde{A}^2_{\max}\right) - \left(f(x^0) - v\left(A^2_{\min}\right)\right) \\
&\ge \tilde{\tau}_1^A \left(v\left(A^2_{\max}\right) - f(x^0)\right) - (\tilde{\tau}_1^A/4)\left(v\left(A^2_{\max}\right) - v\left(A^2_{\min}\right)\right) \\
&\ge \left(\tilde{\tau}_1^A(1 - \tilde{\tau}_1^A/4) - \tilde{\tau}_1^A/4\right)\left(v\left(A^2_{\max}\right) - v\left(A^2_{\min}\right)\right) \\
&\ge \left(\tilde{\tau}_1^A/2\right)\left(v\left(A^2_{\max}\right) - v\left(A^2_{\min}\right)\right),
\end{aligned}$$

where the second inequality is due to the fact that the optimal solution of Problem $(A^2_{\max})$ is feasible to Problem $(\tilde{A}^2_{\max})$. On the other hand, let $\xi_1, \xi_2, \ldots, \xi_d$ be i.i.d. random variables, each taking values 1 and -1 with equal probability 1/2. By symmetricity, we have $\text{Prob}\left\{\prod_{i=1}^d \xi_i = 1\right\} = \text{Prob}\left\{\prod_{i=1}^d \xi_i = -1\right\} = 1/2$. Applying Lemma 1 we know

$$\begin{aligned}
& d!\left(F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) - v(A^2_{\min}) H(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)\right) \\
&= \mathsf{E}\left[\prod_{i=1}^d \xi_i \left\{ f\left(\sum_{k=1}^d \xi_k \hat{x}^k\right) - v(A^2_{\min}) h\left(\sum_{k=1}^d \xi_k \hat{x}^k\right)\right\}\right] \\
&= \mathsf{E}\left[f\left(\sum_{k=1}^d \xi_k \hat{x}^k\right) - v(A^2_{\min}) \left\|\sum_{k=1}^d \xi_k \hat{x}^k\right\|^d \,\middle|\, \prod_{i=1}^d \xi_i = 1\right] \text{Prob}\left\{\prod_{i=1}^d \xi_i = 1\right\} \\
&\quad - \mathsf{E}\left[f\left(\sum_{k=1}^d \xi_k \hat{x}^k\right) - v(A^2_{\min}) \left\|\sum_{k=1}^d \xi_k \hat{x}^k\right\|^d \,\middle|\, \prod_{i=1}^d \xi_i = -1\right] \text{Prob}\left\{\prod_{i=1}^d \xi_i = -1\right\} \\
&\le \frac{1}{2}\mathsf{E}\left[f\left(\sum_{k=1}^d \xi_k \hat{x}^k\right) - v(A^2_{\min}) \left\|\sum_{k=1}^d \xi_k \hat{x}^k\right\|^d \,\middle|\, \prod_{i=1}^d \xi_i = 1\right],
\end{aligned}$$

where the last inequality is due to the fact that

$$f\left(\sum_{k=1}^d \xi_k \hat{x}^k\right) - v(A^2_{\min}) \left\|\sum_{k=1}^d \xi_k \hat{x}^k\right\|^d \ge 0,$$

since $\sum_{k=1}^{d} \xi_k \hat{x}^k \Big/ \left\| \sum_{k=1}^{d} \xi_k \hat{x}^k \right\|$ is feasible to Problem $(A^2_{\min})$.

Thus by randomization, we can find a binary vector $\beta = (\beta_1, \beta_2, \ldots, \beta_d)$ with $\beta_i^2 = 1$ and $\prod_{i=1}^{d} \beta_i = 1$, such that

$$\frac{1}{2}\left(f\left(\sum_{k=1}^{d} \beta_k \hat{x}^k\right) - v(A^2_{\min}) \left\| \sum_{k=1}^{d} \beta_k \hat{x}^k \right\|^d \right) \ge d! \left(\tilde{\tau}_1^A/2\right) \left(v(A^2_{\max}) - v(A^2_{\min}) \right).$$

By letting $\hat{x} = \sum_{k=1}^{d} \beta_k \hat{x}^k \Big/ \left\| \sum_{k=1}^{d} \beta_k \hat{x}^k \right\|$, and noticing $\| \sum_{k=1}^{d} \beta_k \hat{x}^k \| \le d$, we have

$$f(\hat{x}) - v(A^2_{\min}) \ge \frac{d! \tilde{\tau}_1^A \left(v(A^2_{\max}) - v(A^2_{\min}) \right)}{\| \sum_{k=1}^{d} \beta_k \hat{x}^k \|^d} \ge \tau_2^A \left(v(A^2_{\max}) - v(A^2_{\min}) \right).$$

Recall that the above inequality is derived under the condition that (5) holds. In case (5) does not hold, then we shall have

$$f(x^0) - v(A^2_{\min}) > \left(v(A^2_{\max}) - v(A^2_{\min}) \right) \ge \tau_2^A \left(v(A^2_{\max}) - v(A^2_{\min}) \right). \quad (6)$$

By picking $\hat{x}' = \operatorname{argmax}\{f(\hat{x}), f(x^0)\}$, regardless whether (5) or (6) holds, we shall uniformly have

$$f(\hat{x}') - v(A^2_{\min}) \ge \tau_2^A \left(v(A^2_{\max}) - v(A^2_{\min}) \right).$$

□

4 Polynomial optimization with quadratic constraints

In this section, we shall consider a further generalization of the optimization models to include general ellipsoidal constraints.

4.1 Multi-linear function optimization with quadratic constraints

Consider the following model:

$$\begin{array}{lll} (B^1_{\max}) & \max & F(x^1, x^2, \ldots, x^d) \\ & \text{s.t.} & (x^k)^{\mathrm{T}} Q^k_{i_k} x^k \le 1, \quad k = 1, 2, \ldots, d, \quad i_k = 1, 2, \ldots, m_k, \\ & & x^k \in \Re^{n_k}, \quad k = 1, 2, \ldots, d, \end{array}$$

where F is a d-th order multi-linear function with M being its associated d-th order tensor form, and the matrices $Q^k_{i_k} \succeq 0$ and $\sum_{i_k=1}^{m_k} Q^k_{i_k} \succ 0$ for all $1 \le k \le d$, $1 \le i_k \le m_k$.

Let us start with the case $d = 2$, and suppose $F(x^1, x^2) = (x^1)^{\mathrm{T}} M x^2$ with $M \in \Re^{n_1 \times n_2}$. Denote $y = \binom{x^1}{x^2}$, $\bar{M} = \begin{bmatrix} 0_{n_1 \times n_1} & M/2 \\ M^{\mathrm{T}}/2 & 0_{n_2 \times n_2} \end{bmatrix}$, $Q_i = \begin{bmatrix} Q_i^1 & 0_{n_1 \times n_2} \\ 0_{n_2 \times n_1} & 0_{n_2 \times n_2} \end{bmatrix}$ for all $1 \le i \le m_1, and Q_i = \begin{bmatrix} 0_{n_1 \times n_1} & 0_{n_1 \times n_2} \\ 0_{n_2 \times n_1} & Q_{i-m_1}^2 \end{bmatrix}$ for all $m_1 + 1 \le i \le m_1 + m_2$. Problem $(B_{\max}^1)$ is equivalent to

$$
\begin{aligned}
(QP) \max \quad & y^{\mathrm{T}} \bar{M} y \\
\text{s.t.} \quad & y^{\mathrm{T}} Q_i y \le 1, \quad i = 1, 2, \ldots, m_1 + m_2, \\
& y \in \Re^{n_1 + n_2}.
\end{aligned}
$$

It is well known that this model can be solved approximately by a polynomial-time randomized algorithm with approximation ratio $\Omega(1/\log(m_1 + m_2))$ (see e.g. Nemirovski, Roos, and Terlaky [29], and He et al. [12]).

We now proceed to the higher order cases. To illustrate the essential ideas, we shall focus on the case $d = 3$. The extension to any higher order can be done by induction. In case $d = 3$ we may explicitly write Problem $(B_{\max}^1)$ as:

$$
\begin{aligned}
(\bar{B}_{\max}^1) \max \quad & F(x, y, z) \\
\text{s.t.} \quad & x^{\mathrm{T}} Q_i x \le 1, \quad i = 1, 2, \ldots, m_1, \\
& y^{\mathrm{T}} P_j y \le 1, \quad j = 1, 2, \ldots, m_2, \\
& z^{\mathrm{T}} R_k z \le 1, \quad k = 1, 2, \ldots, m_3, \\
& x \in \Re^{n_1}, \quad y \in \Re^{n_2}, \quad z \in \Re^{n_3},
\end{aligned}
$$

where $Q_i \succeq 0$ for all $1 \le i \le m_1$, $P_j \succeq 0$ for all $1 \le j \le m_2$, $R_k \succeq 0$ for all $1 \le k \le m_3$, and $\sum_{i=1}^{m_1} Q_i \succ 0$, $\sum_{j=1}^{m_2} P_j \succ 0$, $\sum_{k=1}^{m_3} R_k \succ 0$.

Combining the constraints of x and y, we have

$$
\operatorname{tr}(Q_i x y^{\mathrm{T}} P_j y x^{\mathrm{T}}) = \operatorname{tr}(x^{\mathrm{T}} Q_i x y^{\mathrm{T}} P_j y) = x^{\mathrm{T}} Q_i x \cdot y^{\mathrm{T}} P_j y \le 1.
$$

Denoting $W = x y^{\mathrm{T}}$, Problem $(\bar{B}_{\max}^1)$ can be relaxed to

$$
\begin{aligned}
(\tilde{B}_{\max}^1) \max \quad & F(W, z) \\
\text{s.t.} \quad & \operatorname{tr}(Q_i W P_j W^{\mathrm{T}}) \le 1, \quad i = 1, 2, \ldots, m_1, \quad j = 1, 2, \ldots, m_2, \\
& z^{\mathrm{T}} R_k z \le 1, \quad k = 1, 2, \ldots, m_3, \\
& W \in \Re^{n_1 \times n_2}, \quad z \in \Re^{n_3}.
\end{aligned}
$$

Observe that for any $W \in \Re^{n_1 \times n_2}$,

$$
\operatorname{tr}(Q_i W P_j W^{\mathrm{T}}) = \operatorname{tr}(Q_i^{1/2} W P_j^{1/2} P_j^{1/2} W^{\mathrm{T}} Q_i^{1/2}) = \left\| Q_i^{1/2} W P_j^{1/2} \right\|^2 \ge 0,
$$

and that for any $W \neq 0$,

$$\sum_{1 \le i \le m_1, 1 \le j \le m_2} \operatorname{tr}(Q_i W P_j W^{\mathrm{T}}) = \operatorname{tr}\left(\left(\sum_{i=1}^{m_1} Q_i\right) W \left(\sum_{j=1}^{m_2} P_j\right) W^{\mathrm{T}}\right)$$
$$= \left\| \left(\sum_{i=1}^{m_1} Q_i\right)^{1/2} W \left(\sum_{j=1}^{m_2} P_j\right)^{1/2} \right\|^2 > 0.$$

Indeed, it is easy to verify that $\operatorname{tr}(Q_i W P_j W^{\mathrm{T}}) = (\operatorname{vec}(W))^{\mathrm{T}}(Q_i \otimes P_j)\operatorname{vec}(W)$, which implies that $\operatorname{tr}(Q_i W P_j W^{\mathrm{T}}) \le 1$ is actually a convex quadratic constraint for W. Thus, Problem $(\tilde{B}^1_{\max})$ is exactly in the form of Problem $(B^1_{\max})$ with $d = 2$. Therefore we are able to find a feasible solution $(\hat{W}, \hat{z})$ of Problem $(\tilde{B}^1_{\max})$ in polynomial-time, such that

$$F(\hat{W}, \hat{z}) \ge \Omega(1/\log(m_1 m_2 + m_3))\, v(\tilde{B}^1_{\max}) \ge \Omega(1/\log m)\, v(\bar{B}^1_{\max}),$$

where $m = \max\{m_1, m_2, m_3\}$. Let us fix $\hat{z}$, and then $F(\cdot, \cdot, \hat{z})$ is a matrix. Our next step is to generate $(\hat{x}, \hat{y})$ from $\hat{W}$. For this purpose, we first introduce the following lemma.

Lemma 2 *Suppose $Q_i \in \mathcal{S}^n_+$ for all $1 \le i \le m$, and $\sum_{i=1}^m Q_i \in \mathcal{S}^n_{++}$, the following SDP problem*

$$\begin{array}{lll} (P) \min & \sum_{i=1}^{m} t_i & \\ \text{s.t.} & \operatorname{tr}(U Q_i) \le 1, \quad i = 1, 2, \ldots, m, \\ & t_i \ge 0, \quad i = 1, 2, \ldots, m, \\ & \begin{bmatrix} U & I_n \\ I_n & \sum_{i=1}^m t_i Q_i \end{bmatrix} \succeq 0 \end{array}$$

has an optimal solution with optimal value equal to n.

Proof Straightforward computation shows that the dual of (P) is

$$\begin{array}{lll} (D) \max & -\sum_{i=1}^{m} s_i - 2\operatorname{tr}(Z) & \\ \text{s.t.} & \operatorname{tr}(X Q_i) \le 1, \quad i = 1, 2, \ldots, m, \\ & s_i \ge 0, \quad i = 1, 2, \ldots, m, \\ & \begin{bmatrix} X & Z \\ Z^{\mathrm{T}} & \sum_{i=1}^m s_i Q_i \end{bmatrix} \succeq 0. \end{array}$$

Observe that (D) indeed resembles (P). Since $\sum_{i=1}^m Q_i \in \mathcal{S}^n_{++}$, both (P) and (D) satisfy the Slater condition, and thus both of them have attainable optimal solutions satisfying the strong duality relationship, i.e. $v(P) = v(D)$. Let (U^*, t^*) be an optimal solution of (P). Clearly $U^* \succ 0$, and by the Schur complement relationship we have $\sum_{i=1}^m t_i^* Q_i \succeq (U^*)^{-1}$. Therefore,

$$v(P) = \sum_{i=1}^m t_i^* \geq \sum_{i=1}^m t_i^* \mathrm{tr}\,(U^* Q_i) \geq \mathrm{tr}\,(U^*(U^*)^{-1}) = n. \tag{7}$$

Observe that for any dual feasible solution (X, Z, s) we always have $-\sum_{i=1}^m s_i \leq -\mathrm{tr}\,(X \sum_{i=1}^m s_i Q_i)$. Hence the following problem is a relaxation of (D), to be called (RD) as follows:

$$\begin{array}{ll} (RD) \max & -\mathrm{tr}\,(XY) - 2\,\mathrm{tr}\,(Z) \\ \text{s.t.} & \begin{bmatrix} X & Z \\ Z^{\mathrm{T}} & Y \end{bmatrix} \succeq 0. \end{array}$$

Consider any feasible solution (X, Y, Z) of (RD). Let $X = P^{\mathrm{T}} D P$ be an orthonormal decomposition with $D = \mathrm{Diag}\,(d_1, d_2, \ldots, d_n)$ and $P^{-1} = P^{\mathrm{T}}$. Notice that $(D, Y', Z') := (PXP^{\mathrm{T}}, PYP^{\mathrm{T}}, PZP^{\mathrm{T}})$ is also a feasible solution of (RD) with the same objective value. By the feasibility, it follows that $d_i Y'_{ii} - (Z'_{ii})^2 \geq 0$, for $i = 1, 2, \ldots, n$. Therefore,

$$\begin{aligned} -\,\mathrm{tr}\,(XY) - 2\,\mathrm{tr}\,(Z) &= -\mathrm{tr}\,(DY') - 2\,\mathrm{tr}\,(Z') = -\sum_{i=1}^n d_i Y'_{ii} - 2\sum_{i=1}^n Z'_{ii} \\ \leq -\sum_{i=1}^n (Z'_{ii})^2 - 2\sum_{i=1}^n Z'_{ii} &\leq -\sum_{i=1}^n (Z'_{ii} + 1)^2 + n \leq n. \end{aligned}$$

This implies that $v(D) \leq v(RD) \leq n$. By combining this with (7), and noticing the strong duality relationship, it follows that $v(P) = v(D) = n$. □

We then have the following decomposition method, to be called DR 3, as a further extension of DR 1. It plays a similar role in Algorithm 3 as DR 2 does in Algorithm 1.

DR (Decomposition Routine) 3

- *Input:* $Q_i \in \mathcal{S}^{n_1}_+$ *for all* $1 \leq i \leq m_1$ *with* $\sum_{i=1}^{m_1} Q_i \in \mathcal{S}^{n_1}_{++}$, $P_j \in \mathcal{S}^{n_2}_+$ *for all* $1 \leq j \leq m_2$ *with* $\sum_{j=1}^{m_2} P_j \in \mathcal{S}^{n_2}_{++}$, $W \in \Re^{n_1 \times n_2}$ *with* $\mathrm{tr}\,(Q_i W P_j W^{\mathrm{T}}) \leq 1$ *for all* $1 \leq i \leq m_1$ *and* $1 \leq j \leq m_2$, *and* $M \in \Re^{n_1 \times n_2}$.
- *For matrices* $Q_1, Q_2, \ldots, Q_{m_1}$, *solve the SDP problem* (P) *in Lemma 2 to get an optimal solution of matrix* U *and scalars* $t_1, t_2, \ldots, t_{m_1}$.
- *Construct*

$$\tilde{W} = \begin{bmatrix} U & W \\ W^{\mathrm{T}} & W^{\mathrm{T}} \left(\sum_{i=1}^{m_1} t_i Q_i\right) W \end{bmatrix} \succeq 0. \tag{8}$$

– *Randomly generate*

$$\begin{pmatrix} \xi \\ \eta \end{pmatrix} \sim \mathcal{N}(0_{n_1+n_2}, \tilde{W}) \tag{9}$$

and repeat if necessary, until $\xi^{\mathrm{T}} M \eta \geq M \bullet W, \xi^{\mathrm{T}} Q_i \xi \leq O(\log m_1)$ *for all* $1 \leq i \leq m_1$*, and* $\eta^{\mathrm{T}} P_j \eta \leq O(n_1 \log m_2)$ *for all* $1 \leq j \leq m_2$.

– *Output:* $(x, y) = \left(\frac{\xi}{\sqrt{\max_i\{\xi^{\mathrm{T}} Q_i \xi\}}}, \frac{\eta}{\sqrt{\max_j\{\eta^{\mathrm{T}} P_j \eta\}}} \right)$.

The complexity of DR 3 depends on the solution for the SDP problem (P), which has $O(n_1^2)$ variables and $O(m_1)$ constraints. The current best interior point method has a computational complexity of $O\left((m_1 + n_1^2)^3 \sqrt{n_1} \log(1/\epsilon)\right)$ to get an ϵ-solution (cf. Chapter 6 of [2]), and it needs $O(\max\{n_1 n_2 m_1, n_2^2 m_2\})$ other operations to get the quality assured solution with high probability.

Lemma 3 *Under the input of DR 3, we can find* $x \in \Re^{n_1}$ *and* $y \in \Re^{n_2}$ *by a polynomial-time randomized algorithm, satisfying* $x^{\mathrm{T}} Q_i x \leq 1$ *for all* $1 \leq i \leq m_1$ *and* $y^{\mathrm{T}} P_j y \leq 1$ *for all* $1 \leq j \leq m_2$*, such that*

$$x^{\mathrm{T}} M y \geq \Omega\left(\frac{1}{\sqrt{n_1} \log m}\right) M \bullet W,$$

where $m = \max\{m_1, m_2\}$.

Proof Following the randomization procedure (9) in DR 3, by Lemma 2 we have, for any $1 \leq i \leq m_1$ and $1 \leq j \leq m_2$,

$$\begin{aligned}
&\mathsf{E}[\xi^{\mathrm{T}} Q_i \xi] = \operatorname{tr}(Q_i U) \leq 1, \\
&\mathsf{E}[\eta^{\mathrm{T}} P_j \eta] = \operatorname{tr}\left(P_j W^{\mathrm{T}} \left(\sum_{i=1}^{m_1} t_i Q_i\right) W\right) = \sum_{i=1}^{m_1} t_i \operatorname{tr}(P_j W^{\mathrm{T}} Q_i W) \leq \sum_{i=1}^{m_1} t_i = n_1.
\end{aligned}$$

So et al. [43] have established that if ξ is a normal random vector and $Q \succeq 0$, then for any $\alpha > 0$

$$\text{Prob}\,\{\xi^{\mathrm{T}} Q \xi \geq \alpha \mathsf{E}[\xi^{\mathrm{T}} Q \xi]\} \leq 2e^{-\alpha/2}.$$

Using this result we have

$$\begin{aligned}
\text{Prob}\,\{\xi^{\mathrm{T}} Q_i \xi \geq \alpha_1\} &\leq \text{Prob}\,\{\xi^{\mathrm{T}} Q_i \xi \geq \alpha_1 \mathsf{E}[\xi^{\mathrm{T}} Q_i \xi]\} \leq 2e^{-\alpha_1/2}, \\
\text{Prob}\,\{\eta^{\mathrm{T}} P_j \eta \geq \alpha_2 n_1\} &\leq \text{Prob}\,\{\eta^{\mathrm{T}} P_j \eta \geq \alpha_2 \mathsf{E}[\eta^{\mathrm{T}} P_j \eta]\} \leq 2e^{-\alpha_2/2}.
\end{aligned}$$

Moreover, $\mathsf{E}[\xi^{\mathrm{T}} M \eta] = M \bullet W$. Now let $\hat{x} = \xi/\sqrt{\alpha_1}$ and $\hat{y} = \eta/\sqrt{\alpha_2 n_1}$, and we have

$$\begin{aligned}
&\text{Prob}\left\{\hat{x}^{\mathrm{T}} M \hat{y} \geq \frac{M \bullet W}{\sqrt{\alpha_1 \alpha_2 n_1}}, \quad \hat{x}^{\mathrm{T}} Q_i \hat{x} \leq 1\,\forall\, 1 \leq i \leq m_1, \quad \hat{y}^{\mathrm{T}} P_j \hat{y} \leq 1\,\forall\, 1 \leq j \leq m_2\right\} \\
&\geq 1 - \text{Prob}\,\{\xi^{\mathrm{T}} M \eta < M \bullet W\} - \sum_{i=1}^{m_1} \text{Prob}\,\{\xi^{\mathrm{T}} Q_i \xi > \alpha_1\} - \sum_{j=1}^{m_2} \text{Prob}\,\{\eta^{\mathrm{T}} P_j \eta > \alpha_2 n_1\} \\
&\geq 1 - (1-\theta) - m_1 \cdot 2e^{-\alpha_1/2} - m_2 \cdot 2e^{-\alpha_2/2} = \theta/2,
\end{aligned}$$

where we let $\alpha_1 = 2\log(8m_1/\theta)$ and $\alpha_2 = 2\log(8m_2/\theta)$. Since $\sqrt{\alpha_1\alpha_2 n_1} = O(\sqrt{n_1}\log m)$, the desired $(\hat{x}, \hat{y})$ can be found with high probability in multiple trials. □

Let us turn back to Problem $(\tilde{B}^1_{\max})$. If we pick $W = \hat{W}$ and $M = F(\cdot, \cdot, \hat{z})$ in applying Lemma 3, then in polynomial-time we can find $(\hat{x}, \hat{y})$, satisfying the constraints of Problem $(\bar{B}^1_{\max})$, such that

$$\begin{aligned} F(\hat{x}, \hat{y}, \hat{z}) = \hat{x}^{\mathrm{T}} M \hat{y} \geq \Omega\left(\frac{1}{\sqrt{n_1}\log m}\right) M \bullet W \\ = \Omega\left(\frac{1}{\sqrt{n_1}\log m}\right) F(\hat{W}, \hat{z}) \geq \Omega\left(\frac{1}{\sqrt{n_1}\log^2 m}\right) v(\bar{B}^1_{\max}). \end{aligned}$$

Thus we have shown the following result.

Theorem 5 *For $d = 3$, Problem $(B^1_{\max})$ admits a polynomial-time randomized approximation algorithm with approximation ratio $\Omega\left(\left(\sqrt{n_1}\,\log^2 m\right)^{-1}\right)$, where $m =$* $\max\{m_1, m_2, m_3\}$.

The result can be generalized to Problem $(B^1_{\max})$ of any degree d.

Theorem 6 *Problem $(B^1_{\max})$ admits a polynomial-time randomized approximation algorithm with approximation ratio $\tau_1^B := \Omega\left(\left(\sqrt{n_1 n_2 \cdots n_{d-2}}\log^{d-1} m\right)^{-1}\right)$ and* $m = \max_{1 \leq k \leq d}\{m_k\}$.

Proof We shall again take recursive steps. Denoting $W = x^1 (x^d)^{\mathrm{T}}$, Problem $(B^1_{\max})$ is relaxed to

$$\begin{aligned} (\hat{B}^1_{\max})\ \max\quad & F(W, x^2, x^3, \ldots, x^{d-1}) \\ \text{s.t.}\quad & \mathrm{tr}\,(Q^1_{i_1} W Q^d_{i_d} W^{\mathrm{T}}) \leq 1, \quad i_1 = 1, 2, \ldots, m_1, \quad i_d = 1, 2, \ldots, m_d, \\ & (x^k)^{\mathrm{T}} Q^k_{i_k} x^k \leq 1, \quad k = 2, 3, \ldots, d-1, \quad i_k = 1, 2, \ldots, m_k, \\ & W \in \Re^{n_1 \times n_d}, \quad x^k \in \Re^{n_k}, \quad k = 2, 3, \ldots, d-1. \end{aligned}$$

Notice that Problem $(\hat{B}^1_{\max})$ is exactly in the form of Problem $(B^1_{\max})$ of degree $d-1$, by treating W as a vector of dimension $n_1 n_d$. By recursion, with high probability we can find a feasible solution $(\hat{W}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1})$ of Problem $(\hat{B}^1_{\max})$ in polynomial-time, such that

$$\begin{aligned} F(\hat{W}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}) &\geq \Omega\left(\left(\sqrt{n_2 n_3 \cdots n_{d-2}}\,\log^{d-2} m\right)^{-1}\right) v(\hat{B}^1_{\max}) \\ &\geq \Omega\left(\left(\sqrt{n_2 n_3 \cdots n_{d-2}}\,\log^{d-2} m\right)^{-1}\right) v(B^1_{\max}). \end{aligned}$$

As long as we fix $(\hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1})$, and pick $M = F(\cdot, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \cdot)$ and $W = \hat{W}$ in applying Lemma 3, we shall be able to find $(\hat{x}^1, \hat{x}^d)$ satisfying the constraints of Problem $(B^1_{\max})$, such that

$$F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^{d-1}, \hat{x}^d) \geq \Omega\left(\frac{1}{\sqrt{n_1}\log m}\right) F(\hat{W}, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}) \geq \tau_1^B\, v(B^1_{\max}).$$

□

Summarizing, the recursive procedure for Problem $(B^1_{\max})$ (Theorem 6) is highlighted as follows:

Algorithm 3

- *Input: d-th order tensor $M^d \in \Re^{n_1 \times n_2 \times \cdots \times n_d}$ with $n_1 \leq n_2 \leq \cdots \leq n_d$, matrices $Q^k_{i_k} \in \mathcal{S}^{n_k}_{+}$ for all $1 \leq i_k \leq m_k$ with $\sum_{i_k=1}^{m_k} Q^k_{i_k} \in \mathcal{S}^{n_k}_{++}$ for all $1 \leq k \leq d$.*
- *Rewrite M^d as $(d-1)$-th order tensor M^{d-1} by combing its first and last components into one, and place the combined component into the last one in M^{d-1}, i.e.,*

$$M^d_{i_1,i_2,\ldots,i_d} = M^{d-1}_{i_2,i_3,\ldots,i_{d-1},(i_1-1)n_d+i_d}, \quad \forall\, 1 \leq i_1 \leq n_1, \ldots, 1 \leq i_d \leq n_d.$$

- *Compute matrices $P_{i_1,i_d} = Q^1_{i_1} \otimes Q^d_{i_d}$ for all $1 \leq i_1 \leq m_1$ and $1 \leq i_d \leq m_d$.*
- *For Problem $(B^1_{\max})$ with $(d-1)$-th order tensor M^{d-1}, matrices $Q^k_{i_k} (2 \leq k \leq d-1)$ and P_{i_1,i_d}: if $d-1=2$, then the problem is essentially Problem (QP), and admits an approximate solution $(\hat{x}^2, \hat{x}^{1,d})$; otherwise obtain a solution $(\hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \hat{x}^{1,d})$ by recursion.*
- *Compute matrix $M_2 = F(\cdot, \hat{x}^2, \hat{x}^3, \ldots, \hat{x}^{d-1}, \cdot)$ and rewrite vector $\hat{x}^{1,d}$ as a matrix $X \in \Re^{n_1 \times n_d}$.*
- *Apply DR 3, with input $(Q_i, P_j, W, M) = (Q^1_i, Q^d_j, X, M_2)$ and output $(\hat{x}^1, \hat{x}^d) = (x, y)$.*
- *Output: a feasible solution $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$.*

4.2 Homogenous polynomial optimization with quadratic constraints

Similar to the spherically constrained case, we now consider the problem

$$\begin{aligned}(B^2_{\max})\ \max\quad & f(x)\\ \text{s.t.}\quad & x^{\mathrm{T}} Q_i x \leq 1, \quad i = 1, 2, \ldots, m,\\ & x \in \Re^n,\end{aligned}$$

where $f(x)$ is a homogenous polynomial function of degree d, $Q_i \succeq 0$ for all $1 \leq i \leq m$, and $\sum_{i=1}^m Q_i \succ 0$. If we relax Problem $(B^2_{\max})$ to the multi-linear form like

Problem $(B^1_{\max})$, then we have

$$
\begin{aligned}
(\bar{B}^2_{\max})\ \max\quad & F(x^1, x^2, \ldots, x^d) \\
\text{s.t.}\quad & (x^k)^{\mathrm{T}} Q_i x^k \le 1, \quad k = 1, 2, \ldots, d, \quad i = 1, 2, \ldots, m, \\
& x^k \in \Re^n, \quad k = 1, 2, \ldots, d.
\end{aligned}
$$

Theorem 7 *For odd d, Problem $(B^2_{\max})$ admits a polynomial-time randomized approximation algorithm with approximation ratio $\tau^B_2 := \Omega\left(d! d^{-d} \left(n^{\frac{d-2}{2}} \log^{d-1} m\right)^{-1}\right)$.*

Proof According to Theorem 6 we can find a feasible solution $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$ of Problem $(\bar{B}^2_{\max})$ in polynomial-time, such that

$$
F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \ge \bar{\tau}^B_2 v(\bar{B}^2_{\max}) \ge \bar{\tau}^B_2 v(B^2_{\max}),
$$

where $\bar{\tau}^B_2 := \Omega\left(\left(n^{\frac{d-2}{2}} \log^{d-1} m\right)^{-1}\right)$.

By (3), we can find a binary vector $\beta = (\beta_1, \beta_2, \ldots, \beta_d)$ with $\beta_i^2 = 1$ for all $1 \le i \le d$, such that

$$
f\left(\sum_{i=1}^d \beta_i \hat{x}^i\right) \ge d! F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d).
$$

Notice that for any $1 \le k \le m$,

$$
\begin{aligned}
&\left(\sum_{i=1}^d \beta_i \hat{x}^i\right)^{\mathrm{T}} Q_k \left(\sum_{j=1}^d \beta_j \hat{x}^j\right) = \sum_{i,j=1}^d \beta_i (\hat{x}^i)^{\mathrm{T}} Q_k \beta_j \hat{x}^j \\
&= \sum_{i,j=1}^d \left(\beta_i Q_k^{\frac{1}{2}} \hat{x}^i\right)^{\mathrm{T}} \left(\beta_j Q_k^{\frac{1}{2}} \hat{x}^j\right) \le \sum_{i,j=1}^d \left\|\beta_i Q_k^{\frac{1}{2}} \hat{x}^i\right\| \left\|\beta_j Q_k^{\frac{1}{2}} \hat{x}^j\right\| \\
&= \sum_{i,j=1}^d \sqrt{(\hat{x}^i)^{\mathrm{T}} Q_k \hat{x}^i} \sqrt{(\hat{x}^j)^{\mathrm{T}} Q_k \hat{x}^j} \le \sum_{i,j=1}^d 1 \cdot 1 = d^2. \qquad (10)
\end{aligned}
$$

If we denote $\hat{x} = \frac{1}{d} \sum_{i=1}^d \beta_i \hat{x}^i$, then $\hat{x}$ is a feasible solution of Problem $(B^2_{\max})$, satisfying

$$
f(\hat{x}) \ge d^{-d} d! F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \ge d^{-d} d! \bar{\tau}^B_2\, v(B^2_{\max}) = \tau^B_2\, v(B^2_{\max}).
$$

□

Theorem 8 *For even d, Problem* $(B^2_{\max})$ *admits a polynomial-time randomized approximation algorithm with relative approximation ratio* τ_2^B*, i.e. there exists a feasible solution* $\hat{x}$ *such that*

$$f(\hat{x}) - v(B^2_{\min}) \geq \tau_2^B \left(v(B^2_{\max}) - v(B^2_{\min}) \right),$$

where $v(B^2_{\min}) := \min_{x^{\mathrm{T}} Q_i x \leq 1, \quad i=1,2,\ldots,m} f(x)$.

Proof First, we observe that $v(B^2_{\max}) \leq v(\bar{B}^2_{\max})$ and $v(B^2_{\min}) \geq -v(\bar{B}^2_{\max})$. Therefore, $2v(\bar{B}^2_{\max}) \geq v(B^2_{\max}) - v(B^2_{\min})$. Let $(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d)$ be the feasible solution of Problem $(\bar{B}^2_{\max})$ as in the proof of Theorem 7. By (10) it follows that $\frac{1}{d}\sum_{k=1}^d \xi_k \hat{x}^k$ is feasible to Problem $(B^2_{\max})$, where $\xi_1, \xi_2, \ldots, \xi_d$ are i.i.d. random variables, each taking values 1 and -1 with equal probability $1/2$. Therefore, by Lemma 1 we have

$$\begin{aligned} d! F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) &= \mathsf{E}\left[\prod_{i=1}^d \xi_i f\left(\sum_{k=1}^d \xi_k \hat{x}^k \right) \right] \\ &= \frac{d^d}{2} \mathsf{E}\left[f\left(\frac{1}{d}\sum_{k=1}^d \xi_k \hat{x}^k \right) - v(B^2_{\min}) \,\middle|\, \prod_{i=1}^d \xi_i = 1 \right] \\ &\quad - \frac{d^d}{2} \mathsf{E}\left[f\left(\frac{1}{d}\sum_{k=1}^d \xi_k \hat{x}^k \right) - v(B^2_{\min}) \,\middle|\, \prod_{i=1}^d \xi_i = -1 \right] \\ &\leq \frac{d^d}{2} \mathsf{E}\left[f\left(\frac{1}{d}\sum_{k=1}^d \xi_k \hat{x}^k \right) - v(B^2_{\min}) \,\middle|\, \prod_{i=1}^d \xi_i = 1 \right]. \end{aligned}$$

By randomization, we are able to find a binary vector $\beta = (\beta_1, \beta_2, \ldots, \beta_d)$ with $\beta_i^2 = 1$ and $\prod_{i=1}^d \beta_i = 1$, such that

$$\begin{aligned} f\left(\frac{1}{d}\sum_{i=1}^d \beta_i \hat{x}^i \right) - v(B^2_{\min}) &\geq 2d^{-d} d! F(\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^d) \\ &\geq 2\tau_2^B v(\bar{B}^2_{\max}) \geq \tau_2^B \left(v(B^2_{\max}) - v(B^2_{\min}) \right). \end{aligned}$$

□

We remark that whether the approximation ratios derived in this paper are tight or not is still unknown, including the case $d = 3$.

5 Numerical results

In this section we are going to test the performance of the approximation algorithms proposed. We shall focus on the case $d = 4$, i.e. fourth order multi-linear function or homogeneous quartic polynomial as a typical case. All the numerical computations are conducted using an Intel Pentium 4 CPU 2.80 GHz computer with 2GB of RAM. The supporting software is MATLAB 7.7.0 (R2008b), and cvx v1.2 (Grant and Boyd [10]) is called for solving the SDP problems whenever applicable.

5.1 Multi-linear function with spherical constraints

Numerical test results on Problem $(A^1_{\max})$ for $d = 4$ are reported in this subsection. In particular, the model to be tested is:

$$\begin{aligned} (E_1)\ \max\quad & F(x, y, z, w) = \sum_{1\le i,j,k,l\le n} a_{ijkl} x_i y_j z_k w_l \\ \text{s.t.}\quad & \|x\| = \|y\| = \|z\| = \|w\| = 1, \\ & x, y, z, w \in \Re^n. \end{aligned}$$

5.1.1 Randomly generated tensors

A fourth order tensor F is generated randomly, with its n^4 entries following i.i.d. normal distributions. Basically we have a choice to make in the recursion in Algorithm 1, yielding two procedures described below. Both methods will use the deterministic routine, namely DR 2.

Test Procedure 1

1. *Solve the relaxation problem*

$$\begin{aligned} \max\quad & F(X, Z) = \sum_{1\le i,j,k,l\le n} a_{ijkl} X_{ij} Z_{kl} \\ \text{s.t.}\quad & \|X\| = \|Z\| = 1,\ X, Z \in \Re^{n\times n} \end{aligned}$$

   by DR 2. Denote its optimal solution to be $(\hat X, \hat Z)$ *and optimal value to be* $\bar v_1$.
2. *Compute matrix* $M_1 = F(\cdot, \cdot, \hat Z)$*, and then solve the problem* $\max_{\|x\|=\|y\|=1} x^{\mathrm T} M_1 y$ *by DR 2. Denote its optimal solution to be* $(\hat x, \hat y)$.
3. *Compute matrix* $M_2 = F(\hat x, \hat y, \cdot, \cdot)$*, and then solve the problem* $\max_{\|z\|=\|w\|=1} z^{\mathrm T} M_2 w$ *by DR 2. Denote its optimal solution to be* $(\hat z, \hat w)$.
4. *Construct a feasible solution* $(\hat x, \hat y, \hat z, \hat w)$ *with objective value* $\hat v_1 = F(\hat x, \hat y, \hat z, \hat w)$*, and report an upper bound of optimal value* $\bar v_1$*, and the ratio* $\tau_1 := \hat v_1 / \bar v_1$.

Table 2 Numerical results (average of 10 instances for each n) of Problem (E_1)

n	2	5	10	20	30	40	50	60	70
$\hat{v}_1$	2.69	6.57	7.56	10.87	11.74	13.89	14.56	17.10	17.76
$\hat{v}_2$	2.61	5.64	8.29	9.58	12.55	13.58	15.57	17.65	18.93
$\bar{v}_1$	2.91	9.46	20.46	39.40	59.55	79.53	99.61	119.77	140.03
$\bar{v}_2$	3.84	12.70	34.81	93.38	169.08	258.94	360.89	472.15	594.13
τ_1	0.926	0.694	0.369	0.276	0.197	0.175	0.146	0.143	0.127
τ_2	0.679	0.444	0.238	0.103	0.074	0.052	0.043	0.037	0.032
$n \cdot \tau_1$	1.85	3.47	3.69	5.52	5.91	6.99	7.31	8.57	8.88
$\sqrt{n} \cdot \tau_1$	1.31	1.55	1.17	1.23	1.08	1.10	1.03	1.11	1.06
$n \cdot \tau_2$	1.36	2.22	2.38	2.05	2.23	2.10	2.16	2.24	2.23

Test Procedure 2

1. *Solve the relaxation problem*

$$\begin{aligned} \max \quad & F(Z, w) = \sum_{1 \le i,j,k,l \le n} a_{ijkl} Z_{ijk} w_l \\ \text{s.t.} \quad & \|Z\| = \|w\| = 1,\ Z \in \Re^{n \times n \times n},\ w \in \Re^n \end{aligned}$$

   by DR 2. Denote its optimal solution to be $(\hat{Z}, \hat{w})$ and optimal value to be $\bar{v}_2$.
2. *Compute third order tensor $F_3 = F(\cdot, \cdot, \cdot, \hat{w})$, and then solve the problem* $\max_{\|Y\|=\|z\|=1} F_3(Y, z)$ *by DR 2. Denote its optimal solution to be $(\hat{Y}, \hat{z})$.*
3. *Compute matrix $M_4 = F_3(\cdot, \cdot, \hat{z})$, and then solve the problem* $\max_{\|x\|=\|y\|=1} x^{\mathrm{T}} M_4 y$ *by DR 2. Denote its optimal solution to be $(\hat{x}, \hat{y})$.*
4. *Construct a feasible solution $(\hat{x}, \hat{y}, \hat{z}, \hat{w})$ with objective value $\hat{v}_2 = F(\hat{x}, \hat{y}, \hat{z}, \hat{w})$, and report an upper bound of optimal value $\bar{v}_2$, and the ratio $\tau_2 := \hat{v}_2/\bar{v}_2$.*

Test Procedure 2 is an explicit description of Algorithm 1 when $d = 4$ and $n_1 = n_2 = n_3 = n_4$. It enjoys a theoretic worst-case performance ratio of $1/n$ by Theorem 2. *Test Procedure 1* follows a similar fashion of Algorithm 1 by following a different recursion. It also enjoys a worst-case performance ratio of $1/n$, which can be proven by using exactly same argument as for Theorem 2. From the simulation results in Table 2, the objective values of the feasible solutions are indeed very similar. However, *Test Procedure 1* computes a much better upper bound for $v(E_1)$, and thus ends up with a better approximation ratio.

The numerical results in Table 2 seem to indicate that the performance ratio of *Test Procedure 1* is about $1/\sqrt{n}$, while that of *Test Procedure 2* is about $2/n$. The main reason for the difference of upper bounds of $v(E_1)$ ($\bar{v}_1$ vs. $\bar{v}_2$) is the relaxation

Table 3 CPU seconds (average of 10 instances for each n) for solving Problem (E_1)

n	5	10	20	30	40	50	60	70	80	90	100	150
Test Procedure 1	0.01	0.02	1.13	12.6	253	517	2433	9860	–	–	–	–
Test Procedure 2	0.01	0.01	0.02	0.06	0.20	0.45	0.95	1.94	3.04	5.08	8.04	58.4

methods. By Proposition 1 we may guess that $\bar{v}_1 = \Omega(\|M\|/n^2)$, while $\bar{v}_2 = \Omega(\|M\|/n)$, and this may contribute to the large gap between $\bar{v}_1$ and $\bar{v}_2$. Consequently, it is quite possible that the true value of $v(E_1)$ is closer to the solution values ($\hat{v}_1$ and $\hat{v}_2$), rather than the optimal value of the relaxed problem ($\bar{v}_1$). The real quality of the solutions produced is possibly much better than what is shown by the upper bounds.

Although *Test Procedure 1* works clearly better than *Test Procedure 2* in terms of upper bound of $v(E_1)$, it requires much more computational time. The most expensive part of *Test Procedure 1* is in Step 1, computing the eigenvalue and its corresponding eigenvector of an $n^2 \times n^2$ matrix. In comparison, for *Test Procedure 2* the corresponding part involves only an $n \times n$ matrix. Evidence in Table 3 shows that *Test Procedure 2* can find a good quality solution very fast even for large size problems. We remark here that for $n = 100$, the sizes of the input data are already in the magnitude of 10^8.

5.1.2 Examples with known optimal solutions

The upper bounds appear to be quite loose in general, as one may observe from the previous numerical results. To test how good the solutions are without referring to the computed upper bounds, in this subsection we report the test results where the problem instances are constructed in such a way that the optimal solutions are known. By this we hope to get some impression, from a different angle, on the quality of the approximate solutions produced by our algorithms. We first randomly generate an n dimensional vector a with norm 1, and generate m symmetric matrices $A_i (1 \le i \le m)$ with its eigenvalues lying in the interval $[-1, 1]$ and $A_i a = a$. Then, we randomly generate an n dimensional vector b with norm 1, and m symmetric matrices $B_i (1 \le i \le m)$ with eigenvalues in the interval $[-1, 1]$ and $B_i b = b$. Define

$$F(x, y, z, w) = \sum_{i=1}^{m} \left(x^{\mathrm{T}} A_i y \cdot z^{\mathrm{T}} B_i w \right).$$

For this particular multi-linear function $F(x, y, z, w)$, it is easy to see that Problem (E_1) has an optimal solution (a, a, b, b) and optimal value is m.

We generate such random instances with $n = 50$ for various m, and subsequently apply *Test Procedure 2* to solve them. Since the optimal values are known, it is possible to compute the *exact* performance ratios. For each m, 200 random instances are generated and tested. The results are shown in Table 4, which suggest that our algorithm works very well and the performance ratios are much better than the theoretical worst-case bounds. Indeed, whenever $m \ge 50$ our algorithm always finds optimal solutions.

Table 4 Numerical results of Problem (E_1) with known optimal when $n = 50$

m	5	10	20	30	40	50	100	150	200
Minimal ratio	0.50	0.66	0.43	0.37	0.37	1.00	1.00	1.00	1.00
Maximal ratio	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Average ratio	0.97	0.86	0.76	0.87	0.97	1.00	1.00	1.00	1.00
Percentage of optimality	7%	10%	35%	71%	94%	100%	100%	100%	100%

5.2 Homogenous polynomial function with quadratic constraints

In this subsection we shall test our solution methods for Problem $(B^2_{\max})$ when $d = 4$:

$$
\begin{aligned}
(E_2)\ \max \quad & f(x) = \sum_{1 \le i,j,k,l \le n} a_{ijkl} x_i x_j x_k x_l \\
\text{s.t.} \quad & x^{\mathrm{T}} Q_i x \le 1, \quad i = 1, 2, \ldots, m, \\
& x \in \Re^n,
\end{aligned}
$$

where $M = (a_{ijkl})$ is super-symmetric, and Q_i is positive semidefinite for all $1 \le i \le m$. First, a fourth order tensor M' is randomly generated, with its n^4 entries following i.i.d. normal distributions. We then symmetrize M' (averaging of the related entries) to form a super-symmetric tensor M. For the constraints, we generate $n \times n$ matrix R_i, whose entries also follow i.i.d. normal distributions, and then let $Q_i = R_i^{\mathrm{T}} R_i$. The following test procedure is applied to (approximately) solve Problem (E_2). For the particular nature of Problem (E_2), *Test Procedure 3* is a simplification of the algorithm proposed in proving Theorem 8. By following essentially the same proof, this procedure also has a worst case relative performance ratio of $\Omega\left(1/(n \log^3 m)\right)$, similar as Theorem 8 asserted.

Test Procedure 3

1. *Solve the relaxation problem*

$$
\begin{aligned}
\max \quad & F(X, X) = \sum_{1 \le i,j,k,l \le n} a_{ijkl} X_{ij} X_{kl} \\
\text{s.t.} \quad & \operatorname{tr}(Q_i X Q_j X^{\mathrm{T}}) \le 1, \quad i = 1, 2, \ldots, m, \quad j = 1, 2, \ldots, m, \\
& X \in \Re^{n \times n}
\end{aligned}
$$

   by SDP relaxation, and randomly sample 10 times to keep the best sampled solution (see [22]). Denote the solution to be $\hat{X}$, and the optimal value of the SDP relaxation to be $\bar{v}_3$.
2. *Solve the SDP problem (P) in Lemma 2. Apply the randomized process as described in (8) and (9), and sample 10 times to keep the best sampled $\hat{x}$ and $\hat{y}$ with maximum $F(\hat{x}, \hat{y}, \hat{x}, \hat{y})$.*
3. *Compute* $\operatorname{argmax}\left\{f(0), f(\hat{x}), f(\hat{y}), f\left((\hat{x}+\hat{y})/2\right), f\left((\hat{x}-\hat{y})/2\right)\right\}$ *with its objective value $\hat{v}_3$, and report an upper bound of optimal value $\bar{v}_3$, and the ratio $\tau_3 := \hat{v}_3/\bar{v}_3$.*

Table 5 Numerical results of Problem (E_2) when $n = 10$ and $m = 30$

Instance	1	2	3	4	5	6	7	8	9	10
$100 \cdot \hat{v}_3$	0.65	0.77	0.32	0.27	0.73	0.42	0.52	0.64	0.98	1.04
$100 \cdot \bar{v}_3$	4.96	4.53	4.75	5.05	5.86	5.32	5.00	5.19	5.07	5.92
τ_3	0.13	0.17	0.07	0.05	0.12	0.08	0.10	0.12	0.19	0.18
$n \log^3 m \cdot \tau_3$	51.5	66.5	26.9	20.9	49.1	31.1	40.6	48.2	75.6	69.4

Table 6 Absolute approximation ratios of Problem (E_2) for various n and m

	$n = 2$	$n = 5$	$n = 8$	$n = 10$	$n = 12$
$m = 1$	0.902	0.579	0.733	0.662	0.600
$m = 5$	0.656	0.283	0.225	0.291	0.171
$m = 10$	0.604	0.223	0.146	0.160	0.089
$m = 30$	0.594	0.178	0.102	0.122	0.092

Table 7 Comparison with SOS methods of Problem (E_2) when $n = 12$ and $m = 30$

Instance	1	2	3	4	5	6	7	8	9	10
$100 \cdot \hat{v}_3$	0.30	0.76	0.43	0.76	0.70	0.49	0.81	0.34	0.29	0.62
$100 \cdot \bar{v}_3$	4.75	4.47	5.21	5.20	4.59	4.81	5.23	5.12	5.89	4.78
$100 \cdot v_{sos}$	2.05	2.02	2.43	2.41	1.86	2.02	1.99	2.24	2.83	1.88
Optimality of v_{sos}	No	No	Yes	Yes	No	Yes	No	Yes	No	No
$\hat{v}_3/\bar{v}_3$	0.06	0.17	0.08	0.15	0.15	0.10	0.15	0.07	0.05	0.13
$\hat{v}_3/v_{sos}$	0.15	0.38	0.18	0.31	0.37	0.24	0.41	0.15	0.10	0.33

For $n = 10$ and $m = 30$, we randomly generate 10 instances of Problem (E_2). The solution results are shown in Table 5.

Table 6 shows the absolute approximation ratios for various n and m by following *Test Procedure 3*. Each entry is the average performance ratio of 10 randomly generated instances.

Next we compare our solution method with the so-called SOS approach for solving Problem (E_2). Due to the limitations of the current SDP solvers (constraining the size of SDP relaxation problem at Step 1 in *Test Procedure 3*), our test procedures work only for small size problems. Since the SOS approach [18,19] works quite efficiently for small size problems, it is interesting to know how the SOS methods would perform in solving these random generated instances of Problem (E_2). In particular, we shall use GloptiPoly 3 of Henrion, Lasserre, and Loefberg [14].

We randomly generated 10 instances of Problem (E_2). By using the first SDP relaxation (Lasserre's procedure [18]), GloptiPoly 3 found global optimal solutions for 4 instances, and got upper bounds of optimal values for the other 6 instances. In the latter case, however, no feasible solutions are generated, while our algorithm always finds feasible solutions with guaranteed approximation ratio, and so the two approaches are complementary to each other. Moreover, GloptiPoly 3 always yields a better upper bound than τ_3 for our test instances, which helps to yield better approximation ratios. The average ratio is 0.112 by using upper bound τ_3, and is 0.262 by using the upper bound produced by GloptiPoly 3 (see Table 7).

To conclude this section and the whole paper, we remark that the algorithms proposed are actually practical, and they produce very high quality solutions. The worst case performance analysis offers a theoretical 'safety net', which is usually far from *typical* performance. Moreover, it is of course possible to improve the solution by some local search procedure. A stable local improvement procedure is a nontrivial task for problem in high dimensions, which is one of our future research topics.

Acknowledgments This research was partially supported by Hong Kong RGC Grants CUHK418406 and CUHK419409. We also like to thank the anonymous referees for their helpful comments.

References

1. Barmpoutis, A., Jian, B., Vemuri, B.C., Shepherd, T.M.: Symmetric positive 4th order tensors and their estimation from diffusion weighted MRI. In: Karssemijer, N., Lelieveldt, B. (eds.) IPMI 2007, LNCS 4584, pp. 308–319 (2007)
2. Ben-Tal, A., Nemirovski, A.: Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications. MPS-SIAM Series on Optimization, Philadelphia (2001)
3. Dahl, G., Leinaas, J.M., Myrheim, J., Ovrum, E.: A tensor product matrix approximation problem in quantum physics. Linear. Algebra. Appl. **420**, 711–725 (2007)
4. De Athayde, G.M., Flôres, R.G. Jr.: Incorporating skewness and kurtosis in portfolio optimization: a multidimensional efficient set, 10. In: Satchell, S., Scowcroft, A. (eds.) Advances in Portfolio Construction and Implementation, pp. 243–257. Butterworth-Heinemann, UK (2003)
5. De Klerk, E.: The complexity of optimizing over a simplex, hypercube or sphere: a short survey. Central Eur J. Oper. Res. **16**, 111–125 (2008)
6. De Klerk, E., Laurent, M., Parrilo, P.A.: A PTAS for the minimization of polynomials of fixed degree over the simplex. Theor. Comput. Sci. **261**, 210–225 (2006)
7. Fujisawa, K., Kojima, M., Nakata, K., Yamashita, M.: SDPA (SemiDefinite Programming Algorithm) User's Manual—version 6.2.0, Research Report B-308, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan (1995)
8. Ghosh, A., Tsigaridas, E., Descoteaux, M., Comon, P., Mourrain, B., Deriche, R.: 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. Computational Diffusion MRI Workshop (CDMRI'08), New York (2008)
9. Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. J. ACM **42**, 1115–1145 (1995)
10. Grant, M., Boyd, S.: CVX: Matlab Software for Disciplined Convex Programming, version 1.2. http://cvxr.com/cvx (2010)
11. Gurvits, L.: Classical deterministic complexity of Edmonds' problem and quantum entanglement. In: Proceedings of the Thirty-Fifth ACM Symposium on Theory of Computing, pp. 10–19, ACM, New York (2003)
12. He, S., Luo, Z.Q., Nie, J., Zhang, S.: Semidefinite relaxation bounds for indefinite homogeneous quadratic optimization. SIAM J. Optim. **19**, 503–523 (2008)
13. Henrion, D., Lasserre, J.B.: GloptiPoly: global optimization over polynomials with Matlab and SeDuMi. ACM Tran. Math. Soft **29**, 165–194 (2003)
14. Henrion, D., Lasserre, J.B., Loefberg, J.: GloptiPoly 3: moments, optimization and semidefinite programming. Optim. Methods Softw. **24**, 761–779 (2009)
15. Jondeau, E., Rockinger, M.: Optimal portfolio allocation under higher moments. Eur. Financ. Manage. **12**, 29–55 (2006)
16. Kofidis, E., Regalia, Ph.: On the best rank-1 approximation of higher order supersymmetric tensors. SIAM J. Matrix Anal. App. **23**, 863–884 (2002)

17. Kroó, A., Szabados, J.: Joackson-type theorems in homogeneous approximation. J. Appr. Theory **152**, 1–19 (2008)
18. Lasserre, J.B.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**, 796–817 (2001)
19. Lasserre, J.B.: Polynomials nonnegative on a grid and discrete representations. Trans. Am. Math. Soc. **354**, 631–649 (2001)
20. Laurent, M.: Sums of squares, moment matrices and optimization over polynomials. In: Putinar, M., Sullivant, S. (eds.) Emerging Applications of Algebraic Geometry, Series: The IMA Volumes in Mathematics and its Applications, vol. 149, Springer, Berlin (2009)
21. Ling, C., Nie, J., Qi, L., Ye, Y.: Biquadratic optimization over unit spheres and semidefinite programming relaxations. SIAM J. Optim. **20**, 1286–1310 (2009)
22. Luo, Z.Q., Sidiropoulos, N.D., Tseng, P., Zhang, S.: Approximation bounds for quadratic optimization with homogeneous quadratic constraints. SIAM J. Optim. **18**, 1–28 (2007)
23. Luo, Z.Q., Sturm, J.F., Zhang, S.: Multivariate nonnegative quadratic mappings. SIAM J. Optim. **14**, 1140–1162 (2004)
24. Luo, Z.Q., Zhang, S.: A semidefinite relaxation scheme for multivariate quartic polynomial optimization with quadratic constraints. SIAM J. Optim. **20**, 1716–1736 (2010)
25. Mandelbrot, B., Hudson, R.L.: The (Mis)Behavior of Markets. Basic Books, New York (2004)
26. Maricic, B., Luo, Z.Q., Davidson, T.N.: Blind constant modulus equalization via convex optimization. IEEE Trans. Signal Process. **51**, 805–818 (2003)
27. Maringer, D., Parpas, P.: Global optimization of higher order moments in portfolio selection. J. Glob. Optim. **43**, 219–230 (2009)
28. Micchelli, C.A., Olsen, P.: Penalized maximum-likelihood estimation, the Baum-Welch algorithm, diagonal balancing of symmetric matrices and applications to training acoustic data. J. Comput. Appl. Math. **119**, 301–331 (2000)
29. Nemirovski, A., Roos, C., Terlaky, T.: On maximization of quadratic form over intersection of ellipsoids with common center. Math. Prog. A **86**, 463–473 (1999)
30. Nesterov, Yu.: Semidefinite relaxation and nonconvex quadratic optimization. Optim. Methods Softw. **9**, 141–160 (1998)
31. Nesterov, Yu. et al.: Squared functional systems and optimization problems. In: Frenk, J.B.G. (ed.) High Performance Optimization, pp. 405–440. Kluwer Academic Press, Dordrecht (2000)
32. Nesterov, Yu.: Random walk in a simplex and quadratic optimization over convex polytopes. CORE Discussion Paper. UCL, Louvain-la-Neuve, Belgium (2003)
33. Ni, Q., Qi, L., Wang, F.: An eigenvalue method for testing positive definiteness of a multivariate form. IEEE Trans. Automat. Contr. **53**, 1096–1107 (2008)
34. Parpas, P., Rustem, B.: Global optimization of the scenario generation and portfolio selection problems. ICCSA 2006, LNCS 3982, pp. 908–917 (2006)
35. Parrilo, P.A.: Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. PhD Dissertation, California Institute of Technology, CA (2000)
36. Parrilo, P.A.: Semidefinite programming relaxations for semialgebraic problems. Math. Prog. B **96**, 293–320 (2003)
37. Prakash, A.J., Chang, C.H., Pactwa, T.E.: Selecting a portfolio with skewness: recent evidence from US, European, and Latin American equity markets. J Banking Financ. **27**, 1375–1390 (2003)
38. Qi, L.: Extrema of a real polynomial. J. Glob. Optim. **30**, 405–433 (2004)
39. Qi, L.: Eigenvalues of a real supersymmetric tensor. J. Symbolic Comput. **40**, 1302–1324 (2005)
40. Qi, L.: Eigenvalues and invariants of tensors. J. Math. Anal. Appl. **325**, 1363–1377 (2007)
41. Qi, L., Teo, K.L.: Multivariate polynomial minimization and its applications in signal processing. J. Glob. Optim. **26**, 419–433 (2003)
42. Qi, L., Wan, Z., Yang, Y.F.: Global minimization of normal quadratic polynomials based on global descent directions. SIAM J. Optim. **15**, 275–302 (2004)
43. So, A.M.C., Ye, Y., Zhang, J.: A unified theorem on SDP rank reduction. Math. Oper. Res. **33**, 910–920 (2008)
44. Soare, S., Yoon, J.W., Cazacu, O.: On the use of homogeneous polynomials to develop anisotropic yield functions with applications to sheet forming. Int. J. Plast. **24**, 915–944 (2008)
45. Sturm, J.F.: SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. Optim. Methods Softw. **11&12**, 625–653 (1999)

46. Toh, K.C., Todd, M.J., Tutuncu, R.H.: SDPT3—a Matlab software package for semidefinite programming. Optim. Methods Softw. **11**, 545–581 (1999)
47. Varjú, P.P.: Approximation by homogeneous polynomials. Const. Appr. **26**, 317–337 (2007)
48. Ye, Y.: Approximating quadratic programming with bound and quadratic constraints. Math. Prog. **84**, 219–226 (1999)
49. Ye, Y.: Approximating global quadratic optimization with convex quadratic constraints. J. Glob. Optim. **15**, 1–17 (1999)
50. Zhang, S.: Quadratic maximization and semidefinite relaxation. Math. Prog. A **87**, 453–465 (2000)
51. Zhang, S., Huang, Y.: Complex quadratic optimization and semidefinite programming. SIAM J. Optim. **16**, 871–890 (2006)
52. Zhang, X., Qi, L., Ye, Y.: The cubic spherical optimization problems, Working Paper. The Hong Kong Polytechnic University, Hong Kong (2009)