SIAM J. OPTIM.
Vol. 28, No. 4, pp. 2902–2921

A COMPLETE SEMIDEFINITE ALGORITHM FOR DETECTING COPOSITIVE MATRICES AND TENSORS*

JIAWANG NIE†, ZI YANG†, AND XINZHEN ZHANG‡

Abstract. A real symmetric tensor is said to be copositive if the associated homogeneous form is greater than or equal to zero over the nonnegative orthant. The problem of detecting tensor copositivity is NP-hard. This paper proposes a complete semidefinite relaxation algorithm for detecting the copositivity of a symmetric tensor. If it is copositive, the algorithm can get a certificate for the copositivity. If it is not, the algorithm can get a point that refutes the copositivity. We show that the detection can be done by solving a finite number of semidefinite relaxations for all tensors. As a special case, the algorithm can also be applied to detect matrix copositivity.

Key words. copositivity, matrix, tensor, semidefinite relaxation

AMS subject classifications. 15A69, 15B48, 90C22

DOI. 10.1137/17M115308X

1. Introduction.

A real symmetric matrix $A \in \mathbb{R}^{n\times n}$ is said to be *copositive* if

$$x^T Ax \geq 0 \quad \forall\, x \in \mathbb{R}^n_+,$$

where $\mathbb{R}^n_+$ is the nonnegative orthant (i.e., the set of nonnegative vectors). If $x^T Ax > 0$ for all $0 \neq x \in \mathbb{R}^n_+$, then A is said to be *strictly copositive*. The set of all $n\times n$ copositive matrices is a cone in $\mathbb{R}^{n\times n}$, which is denoted by $\mathcal{COP}_n$. Copositive matrices were introduced in [39]. They have broad applications, e.g., in quadratic programming [8], dynamical systems and control theory [34, 38], graph theory [13, 21], complementarity problems and variational inequalities [24]. We refer the reader to [4, 22] for surveys on copositive optimization.

A basic problem in optimization is the detection of copositive matrices. Let $\mathcal{S}^n_+$ be the cone of $n \times n$ real symmetric positive semidefinite (PSD) matrices and $\mathcal{N}^n_+$ be the cone of $n \times n$ real symmetric matrices whose entries are all nonnegative. Clearly, it holds that

$$\mathcal{S}^n_+ + \mathcal{N}^n_+ \subseteq \mathcal{COP}_n. \tag{1.1}$$

For $n \leq 4$, the above inclusion is an equality; for $n \geq 5$, the equality does not hold any more [16]. For instance, the Horn matrix [27] is copositive, but it is not a sum of PSD and nonnegative matrices. Checking membership of the cone $\mathcal{COP}_n$ is NP-hard [19, 41]. As shown in [36], a matrix A is copositive if and only if it does not have a principal submatrix that has a negative eigenvalue with a positive eigenvector. To apply this testing, one needs to check eigenvalues for all principal submatrices, which grow exponentially in the dimension. For the $n = 5$ case, when the diagonal entries

*Received by the editors October 20, 2017; accepted for publication (in revised form) June 18, 2018; published electronically October 9, 2018.
http://www.siam.org/journals/siopt/28-4/M115308.html
Funding: Jiawang Nie was partially supported by the NSF Grants DMS-1417985 and DMS-1619973. Xinzhen Zhang was partially supported by the National Natural Science Foundation of China (Grant 11471242).
†Department of Mathematics, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093 (njw@math.ucsd.edu, ziy109@ucsd.edu).
‡School of Mathematics, Tianjin University, Tianjin 300072, People's Republic of China (xzzhang@tju.edu.cn).

are all ones, A is copositive if and only if the polynomial $\|x\|^2(\sum_{i,j=1}^5 A_{ij}x_i^2x_j^2)$ is a sum of squares [17]. When off-diagonal entries are nonpositive, A is copositive if and only if A is positive semidefinite [31]. When a matrix is tridiagonal or acyclic, its copositivity can be detected in linear time [3, 33]. For testing copositivity for general matrices, there exist methods based on simplicial partition [6, 58]. Another approach for testing copositivity is to use the difference of convexity [5, 23]. A survey about existing results and open problems for copositive matrices can be found in [2].

The matrix copositivity can be detected by solving a linear program (LP). Gaddum [25] showed that a symmetric matrix $A \in \mathbb{R}^{n\times n}$ is copositive if and only if for every subset $\emptyset \neq J \subseteq \{1,\dots,n\}$, there exists a vector $x_J \in \mathbb{R}^J$ such that

$$A_{JJ}x_J \geq 0, \quad x_J \geq 0, \quad e_{|J|}^T x_J = 1.$$

In the above, A_{JJ} denotes the principal submatrix of A whose row and column indices are from J, and $e_{|J|}$ denotes the vector of all ones with length $|J|$. Based on this, De Klerk and Pasechnik [15] proposed an LP reformulation for testing matrix copositivity. Let $\underline{\mathrm{v}}$ be the optimal value of the LP:

$$\left\{\begin{array}{cl} \max & \lambda \\ \text{subject to (s.t.)} & A_{JJ}x_J - \lambda e_{|J|} \geq 0, \\ & x_J \geq 0,\ e_{|J|}^T x_J = 1 \ \big(J \subseteq \{1,\dots,n\}\big). \end{array}\right. \tag{1.2}$$

Then A is copositive if and only if the optimal value $\underline{\mathrm{v}} \geq 0$. The linear program (1.2) can detect matrix copositivity exactly, because LP problems can be solved exactly in computation [26]. The size of (1.2) grows exponentially in n. Recently, Dickinson [18] proposed a new certificate for matrix copositivity. It gives a new algorithm to carry out the detection by checking finitely many linear inequalities. The size of linear inequalities also grows exponentially. Testing matrix copositivity is equivalent to solving a nonconvex quadratic program. The finite branch-and-bound algorithm in [7] can be applied to test matrix copositivity. It requires one to solve finitely many semidefinite relaxations.

1.1. Copositive tensors. The concept of copositivity can be naturally generalized to tensors, as in Qi [48]. An n-dimensional tensor of order-m is an array

$$\mathcal{A} := (\mathcal{A}_{i_1\dots i_m}),$$

with indices in the range $1 \leq i_1,\dots,i_m \leq n$. The entries of the form $\mathcal{A}_{jj\dots j}$ are called *diagonal*, while the other entries are called *off-diagonal.* Such an $\mathcal{A}$ is called an n-dimensional tensor of order m. Clearly, vectors are tensors of order 1 and matrices are tensors of order 2. In some applications, we often have symmetric tensors. The tensor $\mathcal{A}$ is *symmetric* if $\mathcal{A}_{i_1i_2\dots i_m} = \mathcal{A}_{j_1j_2\dots j_m}$ whenever $(i_1,i_2,\dots,i_m)$ is a permutation of $(j_1,j_2,\dots,j_m)$. We denote by $\mathtt{S}^m(\mathbb{R}^n)$ the space of symmetric tensors of order m over the vector space $\mathbb{R}^n$. For $\mathcal{A} \in \mathtt{S}^m(\mathbb{R}^n)$ and $x \in \mathbb{R}^n$, define the polynomial

$$\mathcal{A}(x) := \sum_{1\leq i_1,i_2,\cdots,i_m\leq n} \mathcal{A}_{i_1i_2\cdots i_m} x_{i_1}x_{i_2}\cdots x_{i_m}. \tag{1.3}$$

Clearly, $\mathcal{A}(x)$ is a form (i.e., a homogeneous polynomial) of degree m in the variable $x := (x_1,\dots,x_n)$. If $\mathcal{A}(x) \geq 0$ for all $x \in \mathbb{R}^n$, $\mathcal{A}$ is said to be *positive semidefinite* (PSD). If $\mathcal{A}(x) \geq 0$ for all $x \in \mathbb{R}^n_+$, $\mathcal{A}$ is said to be *copositive.* Similarly, if $\mathcal{A}(x) > 0$

for all $0 \neq x \in \mathbb{R}^n_+$, $\mathcal{A}$ is said to be *strictly* copositive. Denote by $\mathcal{COP}_{m,n}$ the cone of all copositive tensors in $\mathtt{S}^m(\mathbb{R}^n)$. Clearly, when the order $m = 2$, positive semidefinite (resp., copositive) tensors are the same as positive semidefinite (resp., copositive) matrices. To be PSD, a tensor must have even order. An odd order nonzero tensor can never be PSD, but it is possibly copositive. For instance, every nonzero tensor with zero diagonal entries and nonnegative off-diagonal ones is copositive, but not PSD.

Copositive tensors have broad applications. For instance, some complementarity problems can be formulated by using copositive tensors [9, 56, 57]. The coclique number of a hypergraph can be bounded by tensor copositivity [10]; see Example 4.6. Copositive tensors are useful in vacuum stability [35]. The spectral radius of a nonnegative tensor can be obtained by copositive tensor optimization [61]; see Example 4.7. Moreover, some polynomial optimization problems can be formulated as linear conic programs about copositive tensors [46]. We refer the reader to [11, 48, 55, 56] for more applications of copositive tensors.

Detecting tensor copositivity is also a mathematically challenging problem. It is also NP-hard, because testing matrix copositivity is a special case. If the off-diagonal entries of a symmetric tensor $\mathcal{A}$ are nonpositive, then $\mathcal{A}$ is copositive if and only if $\mathcal{A}$ is PSD [48]. There also exists a characterization of copositive tensors by the eigenpairs of its principal subtensors [55]. Like the matrix case, tensor copositivity can also be tested by algorithms based on simplicial partition. Typically, when a tensor lies in the interior of the copositive cone, the copositivity can be detected by these kinds of algorithms. However, if it lies on the boundary, they usually have difficulties. We refer the reader to [6, 10, 11, 58] for related work.

1.2. Contributions. This paper focuses on the detection of copositive tensors. In the prior existing methods for detecting copositivity, most of them complete the detection if a tensor lies in the interior of the copsitive cone $\mathcal{COP}_{m,n}$, or if it lies outside $\mathcal{COP}_{m,n}$. If a tensor lies on the boundary of $\mathcal{COP}_{m,n}$, then these methods typically have difficulty in detecting copositivity. Moreover, when a tensor is close to the boundary of $\mathcal{COP}_{m,n}$, these methods often rapidly become more expensive for carrying out detection.

In this paper, we propose a new algorithm for detecting tensor copositivity. It is based on Lasserre-type semidefinite programming (SDP) relaxations and optimality conditions of polynomial optimization. To be precise, we construct a hierarchy of semidefinite relaxations for checking copositivity. The construction uses semidefinite relaxation techniques that are developed in the recent work [45]. If a tensor $\mathcal{A}$ is copositive, we can get a certificate for the copositivity. If it is not, we can compute a point $u \in \mathbb{R}^n_+$ such that $\mathcal{A}(u) < 0$; such a point u refutes the copositivity of $\mathcal{A}$. This is implemented in Algorithm 3.1. No matter whether a tensor is copositive or not, the copositivity can be detected by Algorithm 3.1 in finitely many iterations. Even if a tensor lies on the boundary of the copositive cone, Algorithm 3.1 can also do the detection in finitely many iterations. In other words, for every tensor, its copositivity can be detected by solving a finite number of semidefinite programming relaxations. This is why we call Algorithm 3.1 a *complete semidefinite* algorithm for detecting copositivity. As a special case, matrix copositivity can also be detected by Algorithm 3.1 in finitely many iterations. We would like to remark that there already exists an LP reformulation, e.g., (1.2), which can detect matrix copositivity exactly [15]. However, to the best of the authors' knowledge, for tensors of order 3 or higher, we give the first semidefinite programming algorithm that can detect tensor copositivity in finitely many iterations.

The paper is organized as follows. Section 2 reviews some preliminaries in polynomial optimization. Section 3 gives the complete semidefinite algorithm and proves its properties. Section 4 presents numerical experiments using the algorithm. Section 5 makes conclusions and discussions.

2. Preliminaries. The symbol $\mathbb{N}$ stands for the set of nonnegative integers, and $\mathbb{R}$ for the real field. For $x := (x_1, \ldots, x_n) \in \mathbb{R}^n$ and $\alpha := (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, define

$$x^\alpha := x_1^{\alpha_1} \cdots x_n^{\alpha_n}, \quad |\alpha| := \alpha_1 + \cdots + \alpha_n.$$

For an integer $m > 0$, define the set

$$\mathbb{N}_m^n := \{\alpha \in \mathbb{N}^n |\ |\alpha| \le m\}.$$

The symbol $\mathbb{R}[x]$ denotes the ring of polynomials in x with real coefficients, and $\mathbb{R}[x]_k$ denotes the space of polynomials in $\mathbb{R}[x]$ with degrees at most k. For a symmetric matrix X, the inequality $X \succeq 0$ means that X is positive semidefinite. The superscript T denotes the transpose of a matrix or vector. We use $[x]_m$ to denote the column vector of all monomials in x and of degrees at most m (they are ordered in the graded lexicographical ordering), i.e.,

$$[x]_m := [1,\, x_1, \ldots, x_n,\, x_1^2,\, x_1x_2,\, \ldots,\, x_{n-1}x_n^{m-1},\, x_n^m\,]^T.$$

For a vector x, $\|x\|$ denotes its Euclidean norm. In the space $\mathbb{R}^n$, e denotes the vector of all ones, while e_i denotes the ith unit vector in the canonical basis. For a real number t, $\lceil t \rceil$ (resp., $\lfloor t \rfloor$) denotes the smallest integer not smaller than t (resp., the biggest integer not bigger than t).

The set $\mathbb{R}^{\mathbb{N}_d^n}$ is the space of all real vectors that are labeled by $\alpha \in \mathbb{N}_d^n$. That is, every $y \in \mathbb{R}^{\mathbb{N}_d^n}$ can be labeled as

$$y = (y_\alpha)_{\alpha \in \mathbb{N}_d^n}.$$

Such a y is called a *truncated multisequence* of degree d [44]. For a polynomial $f \in \mathbb{R}[x]_r$ that is written as

$$f = \sum_{|\alpha| \le \mathbb{N}_r^n} f_\alpha x^\alpha,$$

with $r \le d$, we define the operation

$$\langle f, y \rangle = \sum_{|\alpha| \le \mathbb{N}_r^n} f_\alpha y_\alpha. \tag{2.1}$$

Note that $\langle f, y \rangle$ is linear in y for fixed f, and is linear in f for fixed y. For a polynomial $q \in \mathbb{R}[x]_{2k}$ and the integer $t = k - \lceil \deg(q)/2 \rceil$, the outer product $q(x)[x]_t[x]_t^T$ is a symmetric matrix of length $\binom{n+t}{t}$. It can be expanded as

$$q(x)[x]_t[x]_t^T = \sum_{\alpha \in \mathbb{N}_{2k}^n} x^\alpha Q_\alpha$$

for constant symmetric matrices Q_α. For $y \in \mathbb{R}^{\mathbb{N}_{2k}^n}$, define the symmetric matrix

$$L_q^{(k)}[y] := \sum_{\alpha \in \mathbb{N}_{2k}^n} y_\alpha Q_\alpha. \tag{2.2}$$

It is called the kth *localizing matrix* of q and is generated by y. For given q, $L_q^{(k)}[y]$ is linear in y. Clearly, if $q(u) \geq 0$ and $y = [u]_{2k}$, then

$$L_q^{(k)}[y] = q(u)[u]_t[u]_t^T \succeq 0.$$

For instance, if $n = k = 2$ and $q = 1 - x_1 - x_1x_2$, then

$$L_q^{(2)}[y] = \begin{bmatrix} y_{00} - y_{10} - y_{11} & y_{10} - y_{20} - y_{21} & y_{01} - y_{11} - y_{12} \\ y_{10} - y_{20} - y_{21} & y_{20} - y_{30} - y_{31} & y_{11} - y_{21} - y_{22} \\ y_{01} - y_{11} - y_{12} & y_{11} - y_{21} - y_{22} & y_{02} - y_{12} - y_{13} \end{bmatrix}.$$

When $q = 1$ (the constant one polynomial), the localizing matrix $L_1^{(k)}[y]$ reduces to a moment matrix, which we denote by

$$M_k[y] \,:=\, L_1^{(k)}[y].$$

For instance, when $n = 2$, $k = 3$, the matrix $M_3[y]$ is

$$M_3[y] = \begin{bmatrix} y_{00} & y_{10} & y_{01} & y_{20} & y_{11} & y_{02} & y_{30} & y_{21} & y_{12} & y_{03} \\ y_{10} & y_{20} & y_{11} & y_{30} & y_{21} & y_{12} & y_{40} & y_{31} & y_{22} & y_{13} \\ y_{01} & y_{11} & y_{02} & y_{21} & y_{12} & y_{03} & y_{31} & y_{22} & y_{13} & y_{04} \\ y_{20} & y_{30} & y_{21} & y_{40} & y_{31} & y_{22} & y_{50} & y_{41} & y_{32} & y_{23} \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & y_{13} & y_{41} & y_{32} & y_{23} & y_{14} \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} & y_{32} & y_{23} & y_{14} & y_{05} \\ y_{30} & y_{40} & y_{31} & y_{50} & y_{41} & y_{32} & y_{60} & y_{51} & y_{42} & y_{33} \\ y_{21} & y_{31} & y_{22} & y_{41} & y_{32} & y_{23} & y_{51} & y_{42} & y_{33} & y_{24} \\ y_{12} & y_{22} & y_{13} & y_{32} & y_{23} & y_{14} & y_{42} & y_{33} & y_{24} & y_{15} \\ y_{30} & y_{13} & y_{04} & y_{23} & y_{14} & y_{05} & y_{33} & y_{24} & y_{15} & y_{06} \end{bmatrix}.$$

In the following, we review semidefinite relaxations of semialgebraic sets. Consider the semialgebraic set

$$S \,:=\, \{x \in \mathbb{R}^n : \, g_1(x) \geq 0, \ldots, g_t(x) \geq 0\} \tag{2.3}$$

for polynomials $g_1, \ldots, g_t \in \mathbb{R}[x]$. Define the degrees

$$d_j := \lceil \deg(g_j)/2 \rceil, \quad d := \max_j d_j. \tag{2.4}$$

For all $k \geq d$ and for all $x \in S$, we have

$$g_j(x)\big([x]_{k-d_j}\big)\big([x]_{k-d_j}\big)^T \succeq 0, \quad j = 1, \ldots, t.$$

This implies that if $y = [u]_{2k}$ and $u \in S$, then

$$L_{g_j}^{(k)}[y] \succeq 0, \quad j = 1, \ldots, t.$$

Clearly, $[x]_k[x]_k^T \succeq 0$ for all $x \in \mathbb{R}^n$, so

$$M_k[y] \succeq 0$$

for all $y = [u]_{2k}$. So, S is always contained in the set

$$S_k := \left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{c} \exists y \in \mathbb{R}^{\mathbb{N}_{2k}^n},\, y_0 = 1,\, M_k[y] \succeq 0, \\ x = (y_{e_1}, \ldots, y_{e_n}), \\ L_{g_j}^{(k)}[y] \succeq 0 \,(j = 0, 1, \ldots, t) \end{array} \right\} \tag{2.5}$$

for all $k \geq d$. Each S_k is the projection of a set in $\mathbb{R}^{\mathbb{N}_{2k}^n}$ that is defined by linear matrix inequalities. It is a *semidefinite relaxation* of S because $S \subseteq S_k$ for all $k \geq d$. The following nested containment relation holds:

$$S \subseteq \cdots \subseteq S_{k+1} \subseteq S_k \subseteq \cdots \subseteq S_d. \tag{2.6}$$

3. A complete semidefinite algorithm. We discuss how to detect copositivity of a given tensor. For a symmetric tensor $\mathcal{A} \in \mathtt{S}^m(\mathbb{R}^n)$, let $\mathcal{A}(x)$ be the homogeneous polynomial defined as in (1.3). Clearly, $\mathcal{A}$ is copositive if and only if $\mathcal{A}(x) \geq 0$ for all x belonging to the standard simplex

$$\Delta \, = \, \{x \in \mathbb{R}^n : \, e^T x = 1, \; x \geq 0\}.$$

Consider the optimization problem

$$\left\{ \begin{array}{rl} v^* := \min & \mathcal{A}(x) \\ \text{s.t.} & e^T x = 1, \, (x_1, \ldots, x_n) \geq 0. \end{array} \right. \tag{3.1}$$

Clearly, $\mathcal{A}$ is copositive if and only if the minimum value $v^* \geq 0$. Therefore, testing the copositivity of $\mathcal{A}$ is the same as determining the sign of v^*. Problem (3.1) is a polynomial optimization problem. A standard approach for solving it is to apply classical Lasserre relaxations [37]. Since the feasible set is compact and the Archimedean condition holds, its asymptotic convergence is always guaranteed. However, there are still some issues in computation.

- The convergence of classical Lasserre relaxations may be slow for some tensors. Since the computational cost grows rapidly as the relaxation order increases, people often want faster convergence in practice.
- For some tensors $\mathcal{A}$, the classical hierarchy of Lasserre relaxations might fail to have finite convergence. In other words, it may require one to solve infinitely many semidefinite relaxations to detect copositivity. This is not practical in some applications.
- Certifying convergence of Lasserre relaxations is a critical issue in detecting copositivity. The flat extension or truncation condition is usually used for certifying convergence [43]. However, it does not hold for all tensors, especially when (3.1) has infinitely many minimizers. For such cases, certifying convergence is mostly an open question.

In this section, we construct a new hierarchy of semidefinite relaxations that can address all the above issues.

As was recently proposed in [45], there exist tight relaxations for solving polynomial optimization whose constructions are based on optimality conditions and Lagrange multiplier expressions. Since its feasible set is compact and nonempty, problem (3.1) must have a global minimizer, say, u. The constraints of (3.1) are all affine linear functions. One can see that the linear independence constraint qualification condition holds at u. So we have the following optimality conditions (the notation ∇ denotes the gradient):

$$\left\{ \begin{array}{c} \nabla \mathcal{A}(u) \, = \, \lambda_0 e + \sum_{i=1}^n \lambda_i e_i, \\ \lambda_1 u_1 = \cdots = \lambda_n u_n = 0, \; \lambda_1 \geq 0, \ldots, \lambda_n \geq 0, \end{array} \right. \tag{3.2}$$

where $\lambda_0, \lambda_1, \ldots, \lambda_n$ are the Lagrange multipliers. By a simple algebraic computation (also see [45]), one can show that (note that $x^T \nabla f(x) = m f(x)$ for all homogeneous polynomials $f(x)$ of degree m, because $x^T\nabla x^\alpha = |\alpha| x^\alpha$)

$$\begin{cases} \lambda_0 = u^T\nabla\mathcal{A}(u) = m\mathcal{A}(u), \\ \lambda_i = \dfrac{\partial\mathcal{A}(u)}{\partial x_i} - m\mathcal{A}(u) \quad (i=1,2,\dots,n). \end{cases} \tag{3.3}$$

Because of the above expressions, we define new polynomials:

$$p_i := \frac{\partial\mathcal{A}(x)}{\partial x_i} - m\mathcal{A}(x) \quad (i=1,2,\dots,n). \tag{3.4}$$

Since every optimizer u must satisfy (3.2) and its norm $\|u\| \le 1$, the optimization problem (3.1) is equivalent to

$$\begin{cases} \min & \mathcal{A}(x) \\ \text{s.t} & e^Tx - 1 = p_1(x)x_1 = \cdots = p_n(x)x_n = 0, \\ & 1-\|x\|^2 \ge 0,\ x_i \ge 0,\ p_i(x) \ge 0\ (i=1,\dots,n). \end{cases} \tag{3.5}$$

Then we apply Lasserre's relaxations to solve (3.5). For the orders $k = 1, 2, \dots$, solve the semidefinite relaxation problem:

$$\begin{cases} v_k := \min & \langle \mathcal{A}(x), y\rangle \\ \text{s.t} & y_0 = 1,\ L^{(k)}_{e^Tx-1}[y] = 0,\ L^{(k)}_{x_ip_i}[y] = 0\ (i=1,\dots,n), \\ & L^{(k)}_{x_i}[y] \succeq 0,\ L^{(k)}_{p_i}[y] \succeq 0\ (i=1,\dots,n), \\ & L^{(k)}_{1-\|x\|^2}[y] \succeq 0,\ M_k[y] \succeq 0,\ y \in \mathbb{R}^{\mathbb{N}^n_{2k}}. \end{cases} \tag{3.6}$$

The ball constraint $1-\|x\|^2 \ge 0$ is redundant in (3.5). There are two major advantages for using it: (i) Adding the ball constraint results in tighter relaxations, i.e., (3.6) is stronger than the one without using $1-\|x\|^2 \ge 0$. (ii) If $1-\|x\|^2 \ge 0$ is not used, there exist numerical difficulties in solving the semidefinite relaxation (3.6). Example 4.9 shows the benefits of adding the ball constraint.

Note that v^* is also the optimal value of (3.5). From the nested relation (2.6), the feasible set of (3.5) is contained in the projection of that of (3.6), so the optimal value v_k of (3.6) satisfies

$$v_1 \le v_2 \le \cdots \le v^*.$$

Clearly, if $v_k \ge 0$ for some k, then $\mathcal{A}$ is copositive. Combining the above, we can get the following algorithm.

Algorithm 3.1. For a given tensor $\mathcal{A} \in \mathtt{S}^m(\mathbb{R}^n)$, let $m_0 := \lceil m/2 \rceil$ and $k := m_0$. Choose a generic vector $\xi \in \mathbb{R}^{\mathbb{N}^n_m}$. Test the copositivity of $\mathcal{A}$ as follows.

Step 1: Solve the semidefinite relaxation (3.6). If its optimal value $v_k \ge 0$, then $\mathcal{A}$ is copositive and stop. If $v_k < 0$, go to Step 2.

Step 2: Solve the following semidefinite program

$$\begin{cases} \min & \langle \xi^T[x]_m, y\rangle \\ \text{s.t} & L^{(k)}_{e^Tx-1}[y] = 0,\ L^{(k)}_{x_i}[y] \succeq 0\ (i \in [n]), \\ & L^{(k)}_{1-\|x\|^2}[y] \succeq 0,\ L^{(k)}_{v_k-\mathcal{A}(x)}[y] \succeq 0, \\ & y_0 = 1,\ M_k[y] \succeq 0,\ y \in \mathbb{R}^{\mathbb{N}^n_{2k}}. \end{cases} \tag{3.7}$$

If it is feasible, compute an optimizer $\hat{y}$. If it is infeasible, let $k := k+1$ and go to Step 1.

Step 3: Let $u = ((\hat{y})_{e_1}, \dots, (\hat{y})_{e_n})$. If $\mathcal{A}(u) < 0$, then $\mathcal{A}$ is not copositive and stop; otherwise, let $k := k + 1$ and go to Step 1.

Remark. (i) In Algorithm 3.1, the vector ξ can be chosen as a random vector obeying the normal distribution. In MATLAB, we can use the function `randn` to generate each entry of ξ. In Step 1, the copositivity of $\mathcal{A}$ is justified by the relationship $v^* \geq v_k$ for all $k \geq m_0$. In Step 3, the point u must belong to the simplex Δ. This is because of the constraints $L^{(k)}_{e^Tx-1}[y] = 0$ and $L^{(k)}_{x_i}[y] \succeq 0$.

(ii) The method of Algorithm 3.1 is not the same as the one in [45]. The major difference is Step 2, where the optimization problem (3.7) does not appear in [45]. The reason for solving (3.7) is to resolve the difficulty of detecting convergence of the hierarchy of (3.6). In Theorem 3.2, we show that $v_k = v^*$ when k is sufficiently large, but do not know how big such a k is. In practice, the optimal value v^* is usually not known. In [45, section 3.2], the flat extension/truncation condition (FETC)[1] is used to detect the value of k such that $v_k = v^*$. However, the FETC might not hold for some tensors. For instance, it fails to hold when $\mathcal{A}$ is the Horn matrix. A major advantage of solving (3.7) is that we can avoid using FETC to detect convergence. This is a key reason that Algorithm 3.1 can detect copositivity for *all* tensors in finitely many iterations, even for cases in which the FETC fails to hold.

In the following, we show that Algorithm 3.1 must terminate within finitely many iterations for all tensors $\mathcal{A}$. In other words, the copositivity of every $\mathcal{A}$ can be detected correctly by solving finitely many semidefinite relaxations. This is why we call Algorithm 3.1 a complete semidefinite algorithm for detecting tensor copositivity.

THEOREM 3.2. *For all symmetric tensors $\mathcal{A} \in \mathtt{S}^m(\mathbb{R}^n)$, Algorithm* 3.1 *has the following properties:*

(i) *For all $k \geq m_0$, the semidefinite relaxation* (3.6) *is feasible and achieves its optimal value v_k; moreover, $v_k = v^*$ for all k sufficiently large.*
(ii) *For all $k \geq m_0$, the semidefinite program* (3.7) *has an optimizer if it is feasible.*
(iii) *If $\mathcal{A}$ is copositive, then Algorithm* 3.1 *must stop with $v_k \geq 0$ when k is sufficiently large.*
(iv) *If $\mathcal{A}$ is not copositive, then, for almost all $\xi \in \mathbb{R}^{\mathbb{N}^n_m}$ (i.e., $\xi \in \mathbb{R}^{\mathbb{N}^n_m} \backslash \Theta$ for a subset $\Theta \subseteq \mathbb{R}^{\mathbb{N}^n_m}$ of zero Lebesgue measure), Algorithm* 3.1 *must return a point $u \in \Delta$ with $f(u) < 0$ when k is sufficiently large.*

Proof. (i) The feasible set of (3.1) is compact, so it must have a minimizer, say, u^*. Then, u^* satisfies (3.2), and hence u^* is a feasible point for (3.5). So, the feasible set of (3.5) is nonempty. This implies that the semidefinite relaxation (3.6) is always feasible. By the constraint $L^{(k)}_{1-\|x\|^2}[y] \succeq 0$, we can show that the feasible set of (3.6) is compact as follows. First, we can see that

$$1 = y_0 \geq y_{2e_1} + \cdots + y_{2e_n}.$$

So, $0 \leq y_{2e_i} \leq 1$ since each $y_{2e_i} \geq 0$ (because $M_k[y] \succeq 0$). Second, for all $0 < |\alpha| \leq k-1$, the (α, α)th diagonal entry of $L^{(k)}_{1-\|x\|^2}[y]$ is nonnegative, so

$$y_{2\alpha} \geq y_{2\alpha+2e_1} + \cdots + y_{2\alpha+2e_n}. \tag{3.8}$$

By choosing $\alpha = e_1, \dots, e_n$, the same argument can show that $0 \leq y_{2\beta} \leq 1$ for all $|\beta| \leq 2$. By repeatedly applying (3.8), one can further get that $0 \leq y_{2\beta} \leq 1$ for all

[1]See the rank condition (3.16) in [45].

$|\beta| \le k$. Third, note that the diagonal entries of $M_k[y]$ are precisely $y_{2\beta}$ with $|\beta| \le k$. Since $M_k[y] \succeq 0$, all the entries of $M_k[y]$ must be between -1 and 1. This means that y is bounded, and hence the feasible set of (3.6) is compact. Therefore, (3.6) must achieve its optimal value v_k.

To prove $v_k = v^*$ for all k sufficiently large, note that (3.5) is the same as the optimization

$$\left\{ \begin{array}{ll} \min & \mathcal{A}(x) \\ \text{s.t} & e^T x - 1 = p_1(x)x_1 = \cdots = p_n(x)x_n = 0, \\ & x_i \ge 0,\ p_i(x) \ge 0,\ i = 1, \ldots, n. \end{array} \right. \tag{3.9}$$

Its corresponding Lasserre relaxations are

$$\left\{ \begin{array}{rl} v_k' := \min & \langle \mathcal{A}(x), y \rangle \\ \text{s.t} & L^{(k)}_{e^T x - 1}[y] = 0,\ L^{(k)}_{x_i p_i}[y] = 0 \ (1 \le i \le n), \\ & L^{(k)}_{x_i}[y] \succeq 0,\ L^{(k)}_{p_i}[y] \succeq 0 \ (1 \le i \le n), \\ & y_0 = 1,\ M_k[y] \succeq 0,\ y \in \mathbb{R}^{\mathbb{N}^n_{2k}} \end{array} \right. \tag{3.10}$$

for the orders $k = 1, 2, \ldots$. The optimal value of (3.9) is also v^*. The feasible set of (3.6) is contained in that of (3.10), so

$$v_k' \le v_k \le v^*, \quad k = m_0, m_0 + 1, \ldots. \tag{3.11}$$

Next, we show that the set of polynomials

$$F := \left\{ (1 - e^T x)\phi + \sum_{j=1}^n x_j \left(\sum_\ell s_{j,\ell}^2 \right) : \phi \in \mathbb{R}[x],\ s_{j,\ell} \in \mathbb{R}[x] \right\}$$

is Archimedean, i.e., there exists $f \in F$ such that the inequality $f(x) \ge 0$ defines a compact set in $\mathbb{R}^n$. This is true for $f = 1 - \|x\|^2$, because of the identity

$$1 - \|x\|^2 = (1 - e^T x)(1 + \|x\|^2) + \sum_{i=1}^n x_i (1 - x_i)^2 + \sum_{i \ne j} x_i^2 x_j. \tag{3.12}$$

By Theorem 3.3 of [45], we know that $v_k' = v^*$ when k is sufficiently large. Hence, the relation (3.11) implies that $v_k = v^*$ for all k sufficiently large.

(ii) The semidefinite program (3.7) also has the constraint $L^{(k)}_{1 - \|x\|^2}[y] \succeq 0$. By the same argument as in (i), we know that the feasible set of (3.7) is compact. So, it must have an optimizer if it is feasible.

(iii) Clearly, $\mathcal{A}$ is copositive if and only if $v^* \ge 0$. By item (i), $v_k = v^*$ for all k big enough. Therefore, if $\mathcal{A}$ is copositive, we must have $v_k \ge 0$ for all k large enough.

(iv) If $\mathcal{A}$ is not copositive, then $v^* < 0$. By item (i), there exists $k_1 \in \mathbb{N}$ such that $v_k = v^*$ for all $k \ge k_1$. Hence, for all $k \ge k_1$, (3.7) is the same as

$$\left\{ \begin{array}{ll} \min & \langle \xi^T [x]_m, y \rangle \\ \text{s.t} & L^{(k)}_{e^T x - 1}[y] = 0,\ L^{(k)}_{x_i}[y] \succeq 0 \ (i \in [n]), \\ & L^{(k)}_{1 - \|x\|^2}[y] \succeq 0,\ L^{(k)}_{v^* - \mathcal{A}(x)}[y] \succeq 0, \\ & (y)_0 = 1,\ M_k[y] \succeq 0,\ y \in \mathbb{R}^{\mathbb{N}^n_{2k}}. \end{array} \right. \tag{3.13}$$

It is the kth Lasserre relaxation for the polynomial optimization

$$\left\{ \begin{array}{ll} \min & \xi^T [x]_m \\ \text{s.t} & e^T x - 1 = 0,\ x \ge 0,\ v^* - \mathcal{A}(x) \ge 0. \end{array} \right. \tag{3.14}$$

The feasible set of (3.14) is clearly compact. There exists a subset $\Theta \subseteq \mathbb{R}^{\mathbb{N}^n_m}$ of zero Lebesgue measure [53, section 2.2] such that for all $\xi \in \mathbb{R}^{\mathbb{N}^n_m} \backslash \Theta$ problem (3.14) has a unique optimizer, say, u^*. Hence, for almost all $\xi \in \mathbb{R}^{\mathbb{N}^n_m}$, u^* is the unique optimizer. For notational convenience, use $\hat{y}^k$ to denote the optimizer of (3.7) with the relaxation order k. Let $u^k = ((\hat{y}^k)_{e_1}, \ldots, (\hat{y}^k)_{e_n})$. By Corollary 3.5 of [54] or Theorem 3.3 of [43], the sequence $\{u^k\}_{k=m_0}^{\infty}$ must converge to u^*, the unique optimizer of (3.14). Since $\mathcal{A}(u^*) \leq v^* < 0$, we must have $\mathcal{A}(u^k) < 0$ when k is sufficiently large. Moreover, the constraints $L^{(k)}_{x_i}[y] \succeq 0$ imply that $u^k \geq 0$, and $L^{(k)}_{e^Tx-1}[y] = 0$ implies that $e^T u^k = 1$. Therefore, $u^k \in \Delta$. □

Remark 3.3. (i) In Step 1 of Algorithm 3.1, we need to test whether or not $v_k \geq 0$. When the absolute value of v_k is big, this test is easy. However, if its absolute value is very small, then testing its sign might be difficult. Note that the semidefinite relaxation (3.6) is often solved numerically, i.e., v_k is accurate up to a tiny round-off error. This difficulty is not because of theoretical properties of Algorithm 3.1, but due to round-off errors, which occur in all numerical methods. In practice, if v_k is positive or close to zero (say, $v_k > -10^{-6}$), then we can reasonably claim the copositivity of $\mathcal{A}$.

(ii) The semidefinite relaxation programs (3.6) and (3.7) can be solved exactly by quantifier elimination methods [49, 50, 51]. This is because they can be equivalently reformulated as semialgebraic feasibility problems with quantifiers. We refer the reader to [29, 47] for such exact methods and their complexity. Generally, such methods are very expensive to use.

(iii) For computational efficiency, semidefinite programs are often solved numerically by interior-point methods. Starting from interior points, for a given $\epsilon > 0$, interior-point methods can compute a primal-dual feasible pair such that the duality gap is less than ϵ; this can be done in arithmetic operations whose number is polynomial in the size of constraints and variables of (3.6) and in $1/\epsilon$. We refer the reader to [14, 60] for interior-point methods and their complexity. Therefore, for all big k such that $v_k = v^*$ (this is guaranteed by Theorem 3.2), if interior-point methods compute a feasible $\tilde{y}$ such that the duality gap is less than ϵ, then the copositivity of $\mathcal{A}$ can be detected up to the accuracy parameter ϵ.

Remark 3.4. As shown in [15], the LP reformulation (1.2) is exact for detecting matrix copositivity. Here, we compare it with Algorithm 3.1. The major advantage of (1.2) is that it can detect copositivity for *all* matrices, because (1.2) can be solved exactly in computation [26]. However, the size of the LP reformulation (1.2) is exponential growing in n: it has $1 + \frac{1}{2}n2^n$ variables, $n2^n$ inequality constraints, and $2^n - 1$ equality constraints. Algorithm 3.1 is based on solving the hierarchy of semidefinite relaxations (3.6). Theorem 3.2 shows that (3.6) is able to detect copositivity when the relaxation order k is large enough. For each k, the relaxation (3.6) has $\binom{n+2k}{2k} - 1 = O(n^{2k})$ variables. The number of equality constraints is

$$\binom{n+2k-1}{2k-1} + n\binom{n+2k-2}{2k-2} = O(n^{2k-1}).$$

Moreover, (3.6) has $n+1$ linear matrix inequality constraints. Each of them has the length $\binom{n+k-1}{k-1} = O(n^{k-1})$. For fixed k, the size of the relaxation (3.6) is polynomial in n. The relaxation (3.7) is of similar size. However, it is not known when k is big enough to complete the detection, which is an interesting future work. On the other hand, Algorithm 3.1 is also able to detect copositivity for all tensors (including matrices), while (1.2) is only for matrices.

4. Numerical experiments. This section presents numerical experiments that apply Algorithm 3.1 to detect matrix and tensor copositivity. The computation is implemented in MATLAB R2016b, on a Lenovo Laptop with CPU@2.90 GHz and 16.0 GB RAM. Algorithm 3.1 can be implemented by using the software `Gloptipoly 3` [28], which calls the semidefinite program solver `SeDuMi` [59]. For cleanness, we only display 4 decimal digits. The computational time is reported in seconds (s). Recall that v_k is the optimal value of (3.6). We refer the reader to Remark 3.3(i) for the determination of the sign condition $v_k \geq 0$.

4.1. Testing copositive matrices. First, we consider some copositive matrices that are not a sum of PSD and nonnegative matrices.

Example 4.1. Consider the Horn matrix [27]

$$\begin{bmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 & 1 \end{bmatrix}, \tag{4.1}$$

the Hoffman–Pereira matrix [32]

$$\begin{bmatrix} 1 & -1 & 1 & 0 & 0 & 1 & -1 \\ -1 & 1 & -1 & 1 & 0 & 0 & 1 \\ 1 & -1 & 1 & -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 & 1 & -1 & 1 \\ 1 & 0 & 0 & 1 & -1 & 1 & -1 \\ -1 & 1 & 0 & 0 & 1 & -1 & 1 \end{bmatrix}, \tag{4.2}$$

and the Hildebrand matrix [30]

$$\begin{bmatrix} 1 & -\cos\psi_4 & \cos(\psi_4+\psi_5) & \cos(\psi_2+\psi_3) & -\cos\psi_3 \\ -\cos\psi_4 & 1 & -\cos\psi_5 & \cos(\psi_1+\psi_5) & \cos(\psi_3+\psi_4) \\ \cos(\psi_4+\psi_5) & -\cos\psi_5 & 1 & -\cos\psi_1 & \cos(\psi_1+\psi_2) \\ \cos(\psi_2+\psi_3) & \cos(\psi_1+\psi_5) & -\cos\psi_1 & 1 & -\cos\psi_2 \\ -\cos\psi_3 & \cos(\psi_3+\psi_4) & \cos(\psi_1+\psi_2) & -\cos\psi_2 & 1 \end{bmatrix}, \tag{4.3}$$

where each $\psi_i \geq 0$ and $\sum_{i=1}^5 \psi_i < \pi$. Here, we choose the values

$$\psi_1 = \psi_2 = \psi_3 = \psi_4 = \psi_5 = \pi/6.$$

All these matrices are copositive but are not a sum of PSD and nonnegative matrices. We apply Algorithm 3.1 to test their copositivities. The lower bounds v_k and computational times are shown in Table 1. Their copositivities are all confirmed at $k = 3$, up to tiny round-off errors.

Example 4.2. Consider the matrix

$$H_\gamma := \begin{bmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 & \gamma+1 \end{bmatrix}, \tag{4.4}$$

TABLE 1
Computational results for matrices in Example 4.1.

	Horn		Hoffman–Pereira		Hildebrand	
k	v_k	Time (s)	v_k	Time (s)	v_k	Time (s)
1	-0.7889	0.59	-0.4503	0.58	-0.2218	0.61
2	-0.0472	0.35	-0.0250	0.60	-0.0153	0.32
3	-7.0×10^{-8}	1.68	-2.2×10^{-7}	24.85	-1.2×10^{-8}	1.11

which is obtained from the Horn matrix by adding a number γ to the $(5,5)$-entry. Since $(e_1+e_5)^T H_\gamma (e_1+e_5) = \gamma$, H_γ is copositive if and only if $\gamma \geq 0$. For each γ, let $v^*(\gamma)$ denote the optimal value of (3.1) corresponding to the matrix H_γ. With thanks to an anonymous referee, it actually holds that

$$v^*(\gamma) = \begin{cases} 0 & \text{if } \gamma \geq 0, \\ \gamma/(4+\gamma) & \text{if } -2 < \gamma < 0, \\ 1+\gamma & \text{if } \gamma \leq -2. \end{cases} \tag{4.5}$$

(The above can be shown as follows. For the point $\xi = (0, 1/2, 1/2, 0, 0)$, $\xi^T H_\gamma \xi = 0$, so the formula is clearly true for $\gamma \geq 0$. For $-2 < \gamma < 0$, after an enumeration of all possible active constraints, $v^*(\gamma)$ is the smallest of $\gamma + 1$, $\gamma/(4+\gamma)$, $1/(1-\gamma)$, $1/(5-r)$, so $v^*(\gamma) = \gamma/(4+\gamma)$. Similarly, for $\gamma \leq -2$, $v^*(\gamma)$ is the smallest of $1+\gamma$, $1/(1-\gamma)$, $1/(5-r)$, so $v^*(\gamma) = 1+\gamma$.) We explore the performance of Algorithm 3.1 for testing copositivity as γ varies from -0.1 to 0.1 (the increment is 0.005). For each $k = 1,2,3$, the lower bounds v_k are plotted in Figure 1 and the line represents the exact value $v^*(\gamma)$. As we can see, v_1, v_2 do not change much as γ varies, but v_3 increases relatively faster. For $k=1$, the biggest γ for which Algorithm 3.1 returns a point $u \in \Delta$ refuting copositivity (i.e., $u^T H_\gamma u < 0$) is -0.05. For $k=3$, the biggest γ for which Algorithm 3.1 returns a refuting u is -0.005. Indeed, for $\gamma \leq -0.005$, the lower bound v_3 matches the exact value $v^*(\gamma)$, as shown in Figure 1. For $k=2$, no refuting point u is returned by Algorithm 3.1, because (3.7) is infeasible when $k=2$. When $\gamma \geq 0$, the lower bounds v_3 are bigger than -10^{-6}. When $\gamma < 0$, the lower bounds v_3 are smaller than -10^{-3}. The computational results are accurate for detecting copositivity, up to some tiny round-off errors.

Copositive matrices have applications in graph theory. Let $G = (V, E)$ be a graph, with V the set of vertices and E the set of edges. Its stability number $\alpha(G)$ is the maximum number of pairwise disjoint vertices. As shown in [13, 40], it holds that

$$\alpha(G)^{-1} = \min_{x \in \Delta} \quad x^T (A_G + I) x,$$

where A_G is the adjacency matrix of G. To determine $\alpha(G)$, it is enough to compute the minimum value v^* of (3.1) for the matrix $A := A_G + I$.

Example 4.3. For each integer $\ell > 0$, construct a graph G_ℓ as in [20, section 4.2.2] as follows. Let $K_{\ell+1,\ell+1}$ be the complete bipartite graph with vertex set $\{(-1,i), (1,i) : i = 0, 1, \ldots, \ell\}$. Its edges are $((-1,i),(1,j))$ for $i, j = 0, 1, \ldots, \ell$. For each $i = 1, \ldots, \ell$, add a vertex to the edge of the form $((-1,i),(1,i))$, which we denote by $(0,i)$, then delete the old edge $((-1,i),(1,i))$ from the graph and add two new ones $((-1,i),(0,i))$, $((0,i),(1,i))$. The resulting graph is G_ℓ. As mentioned in [20], $\alpha(G_\ell) = \ell + 1$. For the matrix $A := A_G + I$, the optimal value v^* of (3.1) is $1/(\ell+1)$. We apply the semidefinite relaxation (3.6) to compute $\alpha(G_\ell)^{-1}$. The lower bounds v_2 and their

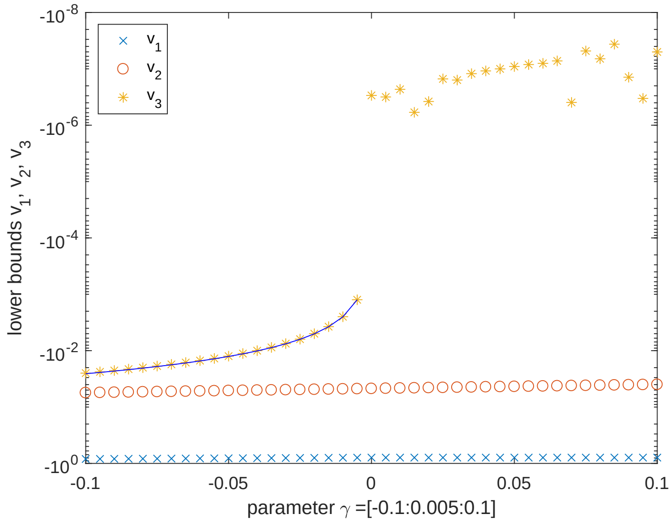

FIG. 1. *Lower bounds* v_1, v_2, v_3 *versus* γ *for* H_γ *in Example* 4.2.

TABLE 2
Stability numbers for graphs G_ℓ.

ℓ	$n = \|G_\ell\|$	v_2	$\|v_2 - (\ell+1)^{-1}\|$	$\left\|\frac{v_2-(\ell+1)^{-1}}{(l+1)^{-1}}\right\|$	Time (s)
1	5	0.5000	9.2×10^{-8}	1.8×10^{-7}	0.53
2	8	0.3333	1.3×10^{-7}	3.9×10^{-7}	1.77
3	11	0.2500	1.5×10^{-6}	6.0×10^{-6}	10.47
4	14	0.2000	2.4×10^{-6}	1.2×10^{-5}	119.25
5	17	0.1667	5.5×10^{-6}	3.3×10^{-5}	901.23
6	20	0.1428	7.4×10^{-6}	5.2×10^{-5}	5186.80
7	23	0.1250	9.3×10^{-6}	7.4×10^{-5}	23205.84

absolute/relative errors are reported in Table 2. For $k = 2$, the lower bounds v_2 are quite accurate. For $\ell \le 4$, it took a short time; for $\ell = 5, 6, 7$, it took a while.

4.2. Testing copositive tensors.

Example 4.4. Consider three tensors $\mathcal{A} \in \mathtt{S}^3(\mathbb{R}^3)$ whose polynomials $\mathcal{A}(x)$ are respectively given as

$$
\left\{
\begin{array}{rl}
\text{Motzkin:} & \mathcal{A}(x) := x_1^2x_2 + x_1x_2^2 + x_3^3 - 3x_1x_2x_3, \\
\text{Robinson:} & \mathcal{A}(x) := x_1^3 + x_2^3 + x_3^3 - x_1^2x_2 - x_1x_2^2 - x_1^2x_3 \\
 & \qquad\qquad -x_1x_3^2 - x_2^2x_3 - x_2x_3^2 + 3x_1x_2x_3, \\
\text{Choi–Lam:} & \mathcal{A}(x) := x_1^2x_2 + x_2^2x_3 + x_3^2x_1 - 3x_1x_2x_3.
\end{array}
\right. \tag{4.6}
$$

When each x_i is replaced by x_i^2, the polynomials $\mathcal{A}(x)$ are respectively the Motzkin, Robinson, and Choi–Lam polynomials (they are all nonnegative but not sums of squares [52]). Hence, these tensors are all copositive. We detect their copositivities via Algorithm 3.1. The computational results are shown in Table 3. For all these tensors, their copositivities are confirmed for $k = 3$, up to tiny round-off errors.

Table 3
Computational results for tensors in Example 4.4.

$\mathcal{A}(x)$	Motzkin		Robinson		Choi–Lam	
k	v_k	Time (s)	v_k	Time (s)	v_k	Time (s)
2	-0.0045	0.78	-0.0208	0.76	-0.0129	0.77
3	-4.3×10^{-8}	0.45	-4.9×10^{-8}	0.23	-2.1×10^{-8}	0.37

Example 4.5. Consider the quartic tensor $\mathcal{A} \in \mathtt{S}^4(\mathbb{R}^4)$ such that

$$\mathcal{A}(x) \,=\, (x_1 + x_2 + x_3 + x_4)^4 - 16(x_1x_2 + x_2x_3 + x_3x_4)^2.$$

It is copositive, because $\mathcal{A}(x)$ has the factorization

$$\big((x_1 - x_2 + x_3 - x_4)^2 + 4x_1x_4\big) \cdot \Big((x_1 + x_2 + x_3 + x_4)^2 + 4(x_1x_2 + x_2x_3 + x_3x_4)\Big).$$

For $k = 2$, we get $v_2 \approx -0.3862$, which took about 0.8 seconds. For $k = 3$, we get $v_3 \approx -1.4 \times 10^{-7}$, which took about 0.6 seconds. The copositivity is confirmed for $k = 3$, up to a round-off error.

Copositive tensors have applications in hypergraph theory [10]. A hypergraph $G = (V, E)$ has a vertex set $V = \{1, \ldots, n\}$ and an edge set E, such that each edge in E is an unordered tuple $(i_1, \ldots, i_\ell)$, with $i_1, \ldots, i_\ell \in V$. It is m-uniform if each edge is an unordered m-tuple $(i_1, \ldots, i_m)$, for distinct $i_1, \ldots, i_m$. Tensor copositivity can be used to bound coclique numbers for hypergraphs.

Example 4.6. A coclique of an m-uniform hypergraph G is a subset $K \subseteq V$ such that any subset of K with cardinality m does not give an edge of G. The largest cardinality of a coclique of G is called the coclique number of G, which we denote by $\omega(G)$ [10]. Computing $\omega(G)$ is typically a challenging problem. However, we can get a good upper bound for it by using tensor copositivity, as shown in [10]. The adjacency tensor of an m-uniform hypergraph $G = (V, E)$ is the symmetric tensor $\mathcal{C} \in \mathtt{S}^m(\mathbb{R}^n)$ such that

$$\mathcal{C}_{i_1 \ldots i_m} = \begin{cases} 1/(m-1)!, & (i_1, \ldots, i_m) \in E, \\ 0 & \text{otherwise}. \end{cases}$$

Let $\mathcal{I}$ be the identity tensor (i.e., $\mathcal{I}_{i_1 \ldots i_m} = 1$ if $i_1 = \cdots = i_m$ and $\mathcal{I}_{i_1 \ldots i_m} = 0$ otherwise), and let $\mathcal{E}$ be the tensor of all ones. It is shown in [10] that $\omega(G)^{m-1} \le \rho$ for all ρ such that $\rho(\mathcal{I} + \mathcal{C}) - \mathcal{E}$ is copositive. To get the smallest such ρ, we need to compute the largest γ such that $(\mathcal{I} + \mathcal{C}) - \gamma\mathcal{E}$ is copositive. Such a γ equals the minimum value v^* of (3.1) for the tensor $\mathcal{A} := \mathcal{I} + \mathcal{C}$. Let v_k be the lower bound given by (3.6). Then

$$\omega(G) \le (1/v^*)^{1/(m-1)} \le (1/v_k)^{1/(m-1)}.$$

Since $\omega(G)$ is an integer, the above implies that

$$\omega(G) \le \lfloor (1/v_k)^{1/(m-1)} \rfloor \tag{4.7}$$

for all $k = m_0, m_0 + 1, \ldots$. We test the above bounds for a class of 3-uniform hypergraphs. Let $G_n = (V_n, E_n)$ be the hypergraph such that $V_n = \{1, \ldots, n\}$ and

$$E_n = \Big\{(i, i+1, i+2)\Big\}_{i=1}^{n-2}.$$

TABLE 4
Coclique numbers of hypergraphs G_n.

n	$\omega(G_n)$	$(1/v_2)^{\frac{1}{m-1}}$	$\lfloor (1/v_2)^{\frac{1}{m-1}} \rfloor$	Time (s)
3	2	2.1381	2	0.12
4	3	3.0000	3	0.13
5	4	4.0000	4	0.16
6	4	4.1631	4	0.26
7	5	5.0000	5	0.37
8	6	6.0000	6	0.63
9	6	6.2140	6	1.41
10	7	7.0041	7	3.07
11	8	8.0000	8	5.39
12	8	8.2657	8	15.61
13	9	9.0370	9	31.57
14	10	10.0000	10	72.08
15	10	10.3254	10	213.15
16	11	11.0836	11	282.55
17	12	12.0000	12	487.77

For these hypergraphs G_n, we solve the relaxation (3.6) for $k = 2$ and get v_2, which gives an upper bound for $\omega(G_n)$ by (4.7). The computational results are shown in Table 4. For G_n in the table, the upper bounds given by (4.7) are tight. Indeed, for $n \geq 3$, one can verify that $\omega(G_n) = n - \lfloor n/3 \rfloor$. A coclique with maximum cardinality for G_n $(n \geq 3)$ is the subset

$$\{1 \leq i \leq n : \operatorname{mod}(i, 3) \neq 0\}.$$

Copositive tensors are useful in spectral theory for nonnegative tensors.

Example 4.7. The largest H-eigenvalue[2] of a symmetric nonnegative tensor $\mathcal{A}$ is related to tensor copositivity. Let ($\mathcal{I}$ denotes the identity tensor)

$$\mathcal{B}(\eta) := \eta\mathcal{I} - \mathcal{A}.$$

As shown in [61], when $\mathcal{A}$ is nonnegative, the tensor $\mathcal{B}(\eta)$ is copositive if and only if $\eta \geq \rho(\mathcal{A})$, where $\rho(\mathcal{A})$ is the spectral radius of $\mathcal{A}$ (i.e., the largest modulus of

[2] A number λ is an H-eigenvalue of $\mathcal{A} \in \mathtt{S}^m(\mathbb{C}^n)$ if there exists a nonzero vector u such that $\frac{\partial}{\partial x_i}\mathcal{A}(u) = m\lambda(u_i)^{m-1}$ for each $i = 1, \ldots, n$; see [10].

H-eigenvalues of $\mathcal{A}$). We verify this fact for the nonnegative tensor $\mathcal{A} \in \mathtt{S}^3(\mathbb{R}^5)$:

$$\mathcal{A}(:,:,1) = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad \mathcal{A}(:,:,2) = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix},$$

$$A(:,:,3) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \quad A(:,:,4) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix},$$

$$A(:,:,5) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

For a range of values of η, the computational results are shown in Table 5.

TABLE 5
Copositivity for $\mathcal{B}(\eta)$ with various ρ.

η	v_2	Time (s)	Copositivity
8.41	1.2×10^{-3}	0.16	Yes
8.40	7.8×10^{-4}	0.17	Yes
8.39	3.5×10^{-4}	0.14	Yes
8.38	-7.8×10^{-5}	0.17	No
8.37	-5.1×10^{-4}	0.15	No
8.36	-9.4×10^{-4}	0.16	No

For all the cases, the order $k = 2$ is enough for detecting copositivity. Indeed, the spectral radius $\rho(\mathcal{A}) \approx 8.381829395789357$, which can be computed by using the method in [12]. For $\eta = \rho(\mathcal{A})$, by Algorithm 3.1, for $k = 2$, we got $v_2 \approx -1.2 \times 10^{-9}$, which took about 0.28 seconds.

Example 4.8. For every tensor $\mathcal{A} \in \mathtt{S}^m(\mathbb{R}^n)$, there always exists a number γ such that $\mathcal{A} + \gamma e^{\otimes m}$ is copositive. The smallest such γ, which we denote by $\gamma_{\min}$, is the negative of the optimal value v^* of (3.1) for the tensor $\mathcal{A}$. Clearly, $\mathcal{A}$ is copositive if and only if $\gamma_{\min} \leq 0$. This example explores the computational cost for computing $\gamma_{\min}$ for randomly generated cubic tensors $\mathcal{A} \in \mathtt{S}^3(\mathbb{R}^n)$ for various n. Here, we generate each $\mathcal{A}_{i_1 i_2 i_3}$ randomly, obeying the normal distribution (this can be done as $\mathcal{A}_{i_1 i_2 i_3} =$ `randn` in MATLAB). For all generated instances, we got $-\gamma_{\min} = v_2$, i.e., the relaxation (3.6) is tight for the order $k = 2$ (this is because rank $M_2[\hat{y}] = 1$ for the optimal solution $\hat{y}$). The computational time is reported in Table 6.

4.3. Some comparisons.

Benefits of the ball constraint. We would like to remark that adding the ball constraint $1 - \|x\|_2 \geq 0$ in (3.5) can give better lower bounds and also improve the computational efficiency. See the remark following (3.6).

TABLE 6
Computational time (in seconds) for random cubic tensors.

n	9	10	11	12	13	14
Time (s)	0.97	1.82	4.38	10.93	23.79	50.44
n	15	16	17	18	19	20
Time (s)	116.89	229.32	327.72	633.40	1109.81	2748.65

Example 4.9. Consider the Hoffman–Pereira matrix in Example 4.1. If the ball constraint is not used, the computational results are shown in Table 7. It takes more time to solve and the computed lower bounds are less accurate.

TABLE 7
Copositivity testing for the Hoffman–Pereira matrix without ball constraint.

k	2	3	4
v_k	-0.0250	-1.6×10^{-4}	-4.5×10^{-4}
Time (s)	1.26	35.31	2870.00

Classical Lasserre relaxations. Since (3.1) is a polynomial optimization problem, the classical Lasserre hierarchy of moment semidefinite relaxations [37] can be applied to solve it. They are

$$\left\{\begin{array}{rl} \min & \langle \mathcal{A}(x), y\rangle \\ \text{s.t} & L^{(k)}_{e^Tx-1}[y] = 0,\ L^{(k)}_{1-x^Tx}[y] \succeq 0, \\ & L^{(k)}_{x_i}[y] \succeq 0\ (1 \le i \le n), \\ & y_0 = 1,\ M_k[y] \succeq 0,\ y \in \mathbb{R}^{\mathbb{N}^n_{2k}} \end{array}\right. \tag{4.8}$$

for $k = m_0, m_0+1, \ldots$. Let ν_k be the optimal value of (4.8). Since the feasible set is compact and the Archimedean condition holds, one can show that $\nu_k \to v^*$ as $k \to \infty$. However, (4.8) is weaker than (3.6), because the feasible set of (3.6) is properly contained in that of (4.8). So, $\nu_k \le v_k \le v^*$ for all k. The following is an example of comparing the lower bounds ν_k and v_k.

Example 4.10. Consider the tensor in Example 4.5. The comparison is reported in Table 8. The optimal value $v^* = 0$.

TABLE 8
A comparison of relaxations (3.6) *and* (4.8) *for the tensor in Example* 4.5.

k	Relaxation (3.6)		Relaxation (4.8)	
	Time (s)	v_k	Time (s)	ν_k
2	0.83	-0.3862	0.22	-0.3862
3	0.55	-1.4×10^{-7}	0.44	-0.0010
4	1.55	-3.0×10^{-7}	1.88	-0.0002
5	8.03	-3.7×10^{-7}	10.80	-0.0001

For $k = 2$, $v_k = \nu_k$, but for $k = 3, 4, 5$, $v_k \gg \nu_k$. Indeed, Algorithm 3.1 terminates at $k = 3$, and the copositivity is detected. In contrast, the convergence of ν_k to v^* is much slower.

High-accuracy SDP solver. In our numerical experiments, the package `SeDuMi` [59] is used to solve the SDP relaxations. Generally, `SeDuMi` can solve SDPs accurately in the computational environment of double precision. However, if SDPs need to be solved highly accurately, we might use high-accuracy solvers, e.g., `SDPA-GMP` [42]. Here, we report the experiment of using `SDPA-GMP` in Algorithm 3.1 to solve the SDP relaxations. The matrices/tensors in Examples 4.1, 4.4, and 4.5 are tested. The results are shown in Table 9. For $k = 2$, `SDPA-GMP` gets similar lower bounds as `SeDuMi` does. However, for $k = 3$, `SDPA-GMP` obtains highly accurate lower bounds compared to those in Tables 1 and 3 and Example 4.5. For the Hildebrand matrix, we got $v_3 \approx -1.2 \times 10^{-17}$; for other matrices/tensors, we got v_3 in a magnitude of order 10^{-30}. We do not know why the accuracy for the Hildebrand matrix is relatively lower. A possible reason is that the Hildebrand matrix is given by cosine values, which might cause extra round-off errors in the computation. The comparison also shows that `SDPA-GMP` takes much more time to solve the SDPs. For Motzkin/Robinson/Choi–Lam tensors, the time is much less than that for others. This is because the sizes of their SDP relaxations are smaller. In some applications, if the copositivity testing needs to be highly accurate, a high-accuracy SDP solver like `SDPA-GMP` might be useful.

TABLE 9
Computational results by `SDPA-GMP`.

Matrix/tensor	$k=2$		$k=3$	
	v_2	Time (s)	v_3	Time (s)
Horn	-0.0472	7.28	-6.0×10^{-29}	303.33
Hoffman–Pereira	-0.0250	76.83	-4.6×10^{-29}	12437.55
Hildebrand	-0.0153	8.25	-1.2×10^{-17}	297.41
Motzkin	-0.0448	0.34	-7.6×10^{-31}	4.94
Robinson	-0.0208	0.37	-1.4×10^{-30}	3.90
Choi–Lam	-0.0129	0.40	-7.7×10^{-31}	4.42
Example 4.5	-0.3862	1.34	-6.4×10^{-29}	43.48

5. Conclusions and discussions. This paper gives a complete semidefinite algorithm for detecting tensor copositivity. If a tensor $\mathcal{A}$ is copositive, we can get a certificate for that, i.e., a nonnegative lower bound for the optimal value v^* of (3.1). If it is not copositive, we can get a point that refutes the copositivity, i.e., a point $u \in \Delta$ such that $\mathcal{A}(u) < 0$. For all $\mathcal{A}$, the copositivity can be detected by solving a finite number of semidefinite relaxations. This is shown in Theorem 3.2.

Algorithm 3.1 is able to detect copositivity for all symmetric tensors. It always terminates after finitely many iterations. However, at the moment, no bound on k is known for the termination. An interesting future work is to estimate the complexity of Algorithm 3.1.

Acknowledgments. The authors would like to thank the associate editor and two anonymous referees for their fruitful suggestions.

REFERENCES

[1] L. Andersson, G. Chang, and T. Elfving, *Criteria for copositive matrices using simplices and barycentric coordinates*, Linear Algebra Appl., 220 (1995), pp. 9–30.

[2] A. Berman, M. Dür, and N. Shaked-Monderer, *Open problems in the theory of completely positive and copositive matrices*, Electron. J. Linear Algebra, 29 (2015), pp. 46–58.

[3] I. M. Bomze, *Linear-time detection of copositivity for tridiagonal matrices and extension to block-tridiagonality*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 840–848.

[4] I. M. Bomze, *Copositive optimization—Recent developments and applications*, European J. Oper. Res., 216 (2012), pp. 509–520.
[5] I. M. Bomze and G. Eichfelder, *Copositivity detection by difference-of-convex decomposition and ω-subdivision*, Math. Program., 138 (2013), pp. 365–400.
[6] S. Bundfuss and M. Dür, *Algorithmic copositivity detection by simplicial partition*, Linear Algebra Appl., 428 (2008), pp. 1511–1523.
[7] S. Burer and D. Vandenbussche, *A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations*, Math. Program., 113 (2008), pp. 259–282.
[8] S. Burer, *On the copositive representation of binary and continuous nonconvex quadratic programs*, Math. Program., 120 (2009), pp. 479–495.
[9] M. Che, L. Qi, and Y. Wei, *Positive-definite tensors to nonlinear complementarity problems*, J. Optim. Theory Appl., 168 (2016), pp. 475–487.
[10] H. Chen, Z. Huang, and L. Qi, *Copositive tensor detection and its applications in physics and hypergraphs*, Comput. Optim. Appl., 69 (2018), pp. 133–158.
[11] H. Chen, Z. Huang, and L. Qi, *Copositivity detection of tensors: Theory and algorithm*, J. Optim. Theory Appl., 174 (2017), pp. 746–761.
[12] C.-F. Cui, Y.-H. Dai, and J. Nie, *All real eigenvalues of symmetric tensors*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 1582–1601.
[13] E. de Klerk and D. V. Pasechnik, *Approximation of the stability number of a graph via copositive programming*, SIAM J. Optim., 12 (2002), pp. 875–892.
[14] E. de Klerk, *Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications*, Kluwer Academic Publishers, Norwell, MA, 2002.
[15] E. de Klerk and D. V. Pasechnik, *A linear programming reformulation of the standard quadratic optimization problem*, J. Global Optim., 37 (2007), pp. 75–84.
[16] P. H. Diananda, *On non-negative forms in real variables some or all of which are non-negative*, Math. Proc. Cambridge Philos. Soc., 58 (1962), pp. 17–25.
[17] P. Dickinson, M. Dür, L. Gijben, and R. Hildebrand, *Scaling relationship between the copositive cone and Parrilo's first level approximation*, Optim. Lett., 7 (2013), pp. 1669–1679.
[18] P. J. C. Dickinson, *A New Certificate for Copositivity*, preprint, http://www.optimization-online.org/DB_HTML/2017/07/6146.html, 2017.
[19] P. Dickinson and L. Gijben, *On the computational complexity of membership problems for the completely positive cone and its dual*, Comput. Optim. Appl., 57 (2014), pp. 403–415.
[20] C. Dobre and J. Vera, *Exploiting symmetry in copositive programs via semidefinite hierarchies*, Math. Program., 151 (2015), pp. 659–680.
[21] I. Dukanovic and F. Rendl, *Copositive programming motivated bounds on the stability and the chromatic numbers*, Math. Program., 121 (2010), pp. 249–268.
[22] M. Dür, *Copositive programming—A survey*, in Recent Advances in Optimization and Its Applications in Engineering, M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels, eds., Springer, Berlin, 2010, pp. 3–20.
[23] M. Dür and J. Hiriart-Urruty, *Testing copositivity with the help of difference-of-convex optimization*, Math. Program., 140 (2013), pp. 31–43.
[24] F. Facchinei and J. S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, New York, 2003.
[25] J. Gaddum, *Linear inequalities and quadratic forms*, Pacific J. Math., 8 (1958), pp. 411–414.
[26] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer, Berlin, 1988.
[27] M. Hall and M. Newman, *Copositive and completely positive quadratic forms*, Math. Proc. Cambridge Philos. Soc., 59 (1963), pp. 329–33.
[28] D. Henrion, J. Lasserre, and J. Loefberg, *GloptiPoly 3: Moments, optimization and semidefinite programming*, Optim. Methods Softw., 24 (2009), pp. 761–779.
[29] D. Henrion, S. Naldi, and M. Safey El Din, *Exact algorithms for linear matrix inequalities*, SIAM J. Optim., 26 (2016), pp. 2512–2539.
[30] R. Hildebrand, *The extreme rays of the 5×5 copositive cone*, Linear Algebra Appl., 437 (2012), pp. 1538–1547.
[31] J. B. Hiriart-Urruty and A. Seeger, *A variational approach to copositive matrices*, SIAM Rev., 52 (2010), pp. 593–629.
[32] A. Hoffman and F. Pereira, *On copositive matrices with -1, 0, 1 entries*, J. Combin. Theory Ser. A, 14 (1973), pp. 302–309.
[33] K. Ikramov, *Linear-time algorithm for verifying the copositivity of an acyclic matrix*, Comput. Math. Math. Phys., 42 (2002), pp. 1701–1703.

[34] D. H. Jacobson, ed., *Extensions of Linear-Quadratic Control, Optimization and Matrix Theory*, Math. Sci. Eng. 133, Elsevier, New York, 2000.

[35] K. Kannike, *Vacuum stability of a general scalar potential of a few fields*, Eur. Phys. J. C, 76 (2016), pp. 1–16.

[36] W. Kaplan, *A test for copositive matrices*, Linear Algebra Appl., 313 (2000), pp. 203–206.

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

[38] O. Mason and R. Shorten, *On linear copositive Lyapunov functions and the stability of switched positive linear systems*, IEEE Trans. Automat. Control, 52 (2007), pp. 1346–1349.

[39] T. S. Motzkin, *Copositive quadratic forms*, in Annual Report of the National Bureau of Standards, Washington, DC, 1952, Report 1818, pp. 11–22.

[40] T. S. Motzkin and E. Straus, *Maxima for graphs and a new proof of a theorem of Turán*, Canadian J. Math., 17 (1965), pp. 533–540.

[41] K. Murty and S. Kabadi, *Some NP-complete problems in quadratic and nonlinear programming*, Math. Program., 39 (1987), pp. 117–129.

[42] M. Nakata, *A numerical evaluation of highly accurate multiple-precision arithmetic version of semidefinite programming solver: SDPA–GMP, –QD and –DD*, in Proceedings of 2010 IEEE Multi-Conference on Systems and Control, IEEE Press, Piscataway, NJ, 2010, pp. 29–34.

[43] J. Nie, *Certifying convergence of Lasserre's hierarchy via flat truncation*, Math. Program. Ser. A, 142 (2013), pp. 485–510.

[44] J. Nie, *The $\mathcal{A}$-truncated K-moment problem*, Found. Comput. Math., 14 (2014), pp. 1243–1276.

[45] J. Nie, *Tight relaxations for polynomial optimization and Lagrange multiplier expressions*, Math. Program. (2018), https://doi.org/10.1007/s10107-018-1276-2.

[46] J. Peña, J. Vera, and L. Zuluaga, *Completely positive reformulations for polynomial optimization*, Math. Program., 151 (2014), pp. 405-431.

[47] L. Porkolab and L. Khachiyan, *On the complexity of semidefinite programs*, J. Global Optim., 10 (1997), pp. 351–365.

[48] L. Qi, *Symmetric nonnegative tensors and copositive tensors*, Linear Algebra Appl., 439 (2013), pp. 228–238.

[49] J. Renegar, *On the computational complexity and geometry of the first-order theory of the reals. Part* I*: Introduction*, J. Symbolic Comput., 13 (1992), pp. 255–299.

[50] J. Renegar, *On the computational complexity and geometry of the first-order theory of the reals. Part* II*: The general decision problem*, J. Symbolic Comput., 13 (1992), pp. 301–327.

[51] J. Renegar, *On the computational complexity and geometry of the first-order theory of the reals. Part* III*: Quantifier elimination*, J. Symbolic Comput., 13 (1992), pp. 329–352.

[52] B. Reznick, *Some concrete aspects of Hilbert's* 17*th problem*, in Real Algebraic Geometry and Ordered Structures, Contemp. Math. 253, American Mathematical Society, Providence, RI, 2000, pp. 251–272.

[53] R. Schneider, *Convex Bodies: The Brunn–Minkowski Theory*, Encyclopedia Math. Appl. 44, Cambridge University Press, Cambridge, 1993.

[54] M. Schweighofer, *Optimization of polynomials on compact semialgebraic sets*, SIAM J. Optim., 15 (2005), pp. 805–825.

[55] Y. Song and L. Qi, *Necessary and sufficient conditions for copositive tensors*, Linear Multilinear Algebra, 63 (2015), pp. 120–131.

[56] Y. Song and L. Qi, *Properties of tensor complementarity problem and some classes of structured tensors*, Ann. Appl. Math., 33 (2017), pp. 308–323.

[57] Y. Song and L. Qi, *Tensor complementarity problem and semi-positive tensors*, J. Optim. Theory Appl., 169 (2016), pp. 1069–1078.

[58] J. Sponsel, S. Bundfuss, and M. Dür, *An improved algorithm to test copositivity*, J. Global Optim., 52 (2012), pp. 537–551.

[59] J. Sturm, *Using SeDuMi* 1.02*: A MATLAB toolbox for optimization over symmetric cones*, Optim. Methods Softw., 11 (1999), pp. 625–653; available at http://sedumi.ie.lehigh.edu.

[60] M. Todd, *Semidefinite Optimization*, Acta Numer., 10 (2001), pp. 515–560.

[61] L. Zhang, L. Qi, and G. Zhang, *M-tensors and some applications*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 437–452.