Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 21, No. 2, pp. 195–214, 2025

POLYNOMIAL SEMIDEFINITE COMPLEMENTARITY PROBLEMS*

Xin Zhao, Anwa Zhou† and Jinyan Fan

Abstract: In this paper, we introduce the polynomial semidefinite complementarity problem. We formulate it equivalently as a polynomial optimization problem with both scalar polynomial constraints and polynomial matrix inequality constraints. The solutions for the problem can be computed sequentially, if there are finite ones. Each of them can be obtained by Lasserre's hierarchy of matrix-type semidefinite relaxations. Under suitable assumptions, the asymptotic and finite convergences for such sequence of semidefinite relaxations are also proved. Numerical experiments show that the proposed method is efficient for test problems.

Key words: *semidefinite complementarity problem, polynomial optimization, matrix-type semidefinite relaxation, finite convergence*

Mathematics Subject Classification: *65K10, 90C22, 90C26, 90C33*

1 Introduction

Polynomial matrices play an important role in control systems. Most synthesis problems for linear systems can be formulated as polynomial matrix inequality (PMI) optimization problems in controller parameters. For example, many robust fixed-order controller design problems can be formulated as optimization problems, where the feasible solution set is modeled by parametrized PMIs (cf. [8]). In some practical problems, suitable changes in variables or subspace projections have been found to convexify the design problem and derive equivalent linear matrix inequality (LMI) formulations (cf. [5, 37]).

Generally, PMI problems are nonconvex and difficult to solve. Kojima [23] proposed sums-of-squares relaxations for PMI optimization problems, with an additional constraint for the variables to be in the unit ball. It was shown that the optimal values of the relaxations converge to the optimal value of the problem. Hol and Scherer [19, 20] handled the matrix-type inequality constraints directly and derived a hierarchy of specific linear matrix inequality (LMI) relaxations, whose associated sequence of optimal values converge to the global optimum. Henrion and Lasserre [14] proposed a hierarchy of convex LMI relaxations to solve nonconvex PMI optimization problems, which generate a monotone sequence of

*Xin Zhao was partially supported by Shanghai Sailing Program grant 22YF1400900. Anwa Zhou was partially supported by the National Natural Science Foundation of China grant 12271336. Jinyan Fan was partially supported by the National Natural Science Foundation of China grants 11971309 and 12371307, and the Fundamental Research Funds for the Central Universities.

†Corresponding author.

 DOI: https:// doi.org/10.61208/pjo-2023-040

lower bounds that converge to the global optimum of the problem. In [30], Nie gave the explicit constructions of lifted LMI representations for the set defined by PMIs. Henrion and Lasserre [15] constructed a hierarchy of inner approximations of feasible sets defined by parametrized or uncertain PMIs, which are essential for controller design purposes because they correspond to sufficient conditions and guarantee stability or robust stability [12, 21].

In recent years, the polynomial complementarity problem (PCP) has received much attention. Given two polynomial maps $F, G : \mathbb{R}^n \to \mathbb{R}^n$, the polynomial complementarity problem (PCP(F,G)) is the nonlinear complementarity problem of finding a vector $x \in \mathbb{R}^n$ such that

$$F(x) \geq 0, \quad G(x) \geq 0, \quad \text{and} \quad \langle F(x), G(x) \rangle = 0, \tag{1.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product in real Euclidean space. The PCP(F,G) includes the tensor complementarity problem (TCP) as a special case if $F(x) = x$ and $G(x) = f(x) + q$ with a given homogeneous polynomial map $f(x) : \mathbb{R}^n \to \mathbb{R}^n$ and a vector $q \in \mathbb{R}^n$. TCP has wide applications in n-person noncooperative game[17], hypergraph clustering problem[18] and so on. Interested readers are referred to [2, 3, 18, 39] for theories and methods of TCP. Obviously, all methods for the nonlinear complementarity problem (NCP) can be applied to solve the PCP. One may expect specialized methods and results that make use of the polynomial nature of the problem [10, 27, 40].

Inspired by the wide applications of PMI optimization problems and the PCP, in this paper, we introduce the polynomial semidefinite complementarity problem (PSDCP), which is to find a vector $x \in \mathbb{R}^n$, such that

$$Q_1(x) \succeq 0, \quad Q_2(x) \succeq 0, \quad \text{and} \quad \langle Q_1(x), Q_2(x) \rangle = 0, \tag{1.2}$$

where $Q_i(x)$ represents the given $m \times m$ polynomial matrices, with each entry being a polynomial in x; $Q_i(x) \succeq 0$ means that $Q_i(x)$ is symmetric positive semidefinite ($i = 1, 2$); $\langle A, B \rangle := trace(AB)$ denotes the standard inner product in real symmetric matrix space. Clearly, when $Q_1(x)$ and $Q_2(x)$ are diagonal, the problem (1.2) is reduced to the PCP (1.1). When one of the two matrix is zero, (1.2) becomes the PMI problem. Hence the PSDCP includes both the PCP and PMI problem as special cases. They have wide applications in control systems, noncooperative game, hypergraph clustering problem and so on.

Indeed, the constraint $Q_i(x) \succeq 0$ defines a semi-algebraic set that can be described explicitly in terms of finite scalar polynomial inequalities such as the principal minors of $Q_i(x)$. Thus Lasserre's semidefinite relaxation method for the scalar polynomial optimization problem [25] can be used to solve (1.2). By increasing the relaxation orders, Lasserre's approach results in a sequence of lower bounds of the global optimum, and the asymptotic convergence can be obtained under the Archimedean condition. However, note that $Q_i(x)$ may have exponentially many principal minors with high degrees, so the method may encounter numerical difficulties. The representation directly on $Q_i(x)$ is preferable. In [14], Henrion and Lasserre gave matrix-type relaxations to solve nonconvex PMI optimization problems.

Contributions. This paper proposes a matrix-type semidefinite relaxation method for computing all real solutions of the PSDCP (1.2) if there are finitely many ones. We formulate (1.2) as polynomial optimization with scalar polynomial and PMI constraints. Especially, we replace the complementarity condition $\langle Q_1(x), Q_2(x) \rangle = 0$ with equivalent m^2 algebraic equations. By doing this, tighter relaxation problems can be obtained when applying Lasserre's relaxation method. The solutions can be computed in order by choosing a random sum of squares polynomial objective. Each of them can be computed by a sequence of semidefinite relaxations. Under suitable assumptions, we prove that such sequence of semidefinite relaxations has asymptotic convergence and finite convergence.

The paper is organized as follows. In section 2, we review the basics of polynomial optimization. In section 3, we study properties of the PSDCP and reformulate it as a polynomial optimization problem with scalar polynomial and PMI constraints. A hierarchy of semidefinite relaxations is constructed to compute all real solutions of the PSDCP if there are finitely many ones. Convergence properties of the proposed algorithm are also studied. In section 4, some numerical experiments are presented. Finally, we conclude the paper in section 5.

Notation. The symbol $\mathbb{R}$ (resp., $\mathbb{N}$) denotes the set of real numbers (resp., nonnegative integers). $\mathbb{R}^n$ (resp., $\mathbb{N}^n$) denotes the set of all real (resp., nonnegative integers) n-dimensional vectors. For integer $n > 0$, $[n]$ denotes the set $\{1, \ldots, n\}$. For $x := (x_1, \ldots, x_n)$ and $\alpha := (\alpha_1, \ldots, \alpha_n)$, denote the monomial power $x^\alpha := x_1^{\alpha_1} \cdots x_n^{\alpha_n}$. The symbols $[x]$ and $[x]_d$ denote the following vectors of monomials, respectively,

$$[x] := [1, x_1, \ldots, x_n, x_1^2, x_1x_2, \ldots, x_1^d, x_1^{d-1}x_2, \ldots, x_n^d, \ldots]^T$$

and

$$[x]_d := [1, x_1, \ldots, x_n, x_1^2, x_1x_2, \ldots, x_1^d, x_1^{d-1}x_2, \ldots, x_n^d]^T.$$

We denote $s(d) := \binom{n+d}{d}$, which is the length of $[x]_d$. The symbol $\mathbb{R}[x] = \mathbb{R}[x_1, \ldots, x_n]$ denotes the ring of polynomials in $x := (x_1, \ldots, x_n)$ over the real field. The symbol $deg(p)$ denotes the degree of a polynomial $p \in \mathbb{R}[x]$. $\mathbb{R}[x]_k$ denotes the set of polynomials in $\mathbb{R}[x]$ with a degree of at most k. Its dimension is $s(k)$. $\mathbb{R}[x]^m$ denotes the set of m-dimensional real polynomial vectors. $\mathbb{S}[x]^m$ denotes the set of $m \times m$ real symmetric polynomial matrices in x. For a polynomial matrix $Q(x) \in \mathbb{S}[x]^m$, $deg(Q(x))$ denotes the maximal degree of all its polynomial entries. $\mathbb{S}[x]^m_k$ (resp. $\mathbb{R}[x]^m_k$) denotes the set of polynomial matrices (resp. vectors) in $\mathbb{S}[x]^m$ (resp. $\mathbb{R}[x]^m$) with a degree of at most k. For two polynomial matrices, $A(x), B(x) \in \mathbb{S}[x]^m$, $\langle A(x), B(x)\rangle := \mathrm{trace}(A(x)B(x))$ denotes the inner product of $A(x)$ and $B(x)$. Especially, we denote by $\mathbb{S}^m := \mathbb{S}[x]^m_0$ the set of all $m \times m$ real symmetric matrices. For a symmetric matrix $X \in \mathbb{S}^m$, $X \succeq 0$ means X is positive semidefinite. The set of all positive semidefinite matrix in $\mathbb{S}^m$ is denoted as $\mathbb{S}^m_+$. For $k \in \mathbb{R}$, $\lceil k \rceil$ denotes the smallest integer not smaller than k. Throughout this paper, we use the words "generic" and "generically" as conditions on the input data for some property to hold, and they shall mean for all but a set of Lebesgue measure zero in the space of data.

2 Preliminaries

2.1 Polynomial optimization

In this section, we review some basics in polynomial optimization. A polynomial $p \in \mathbb{R}[x]$ is said to be sum-of-squares if $p = \sigma_1^2 + \cdots + \sigma_t^2$ for some $\sigma_1, \ldots, \sigma_t \in \mathbb{R}[x]$. The set of all sum-of-squares polynomials in x is denoted as $\Sigma[x]$. The k-th truncation of $\Sigma[x]$ is $\Sigma[x]_k = \Sigma[x] \cap \mathbb{R}[x]_k$. A polynomial matrix $R(x) \in \mathbb{S}[x]^m_{2k}$ is said to be the sum-of-squares if there exist polynomial vectors $L_j(x) \in \mathbb{R}[x]^m_k$, such that

$$R(x) = \sum_j L_j(x)L_j(x)^T.$$

Let $\Sigma[x]^{m\times m}$ denote the set of all $m \times m$ sum-of-square polynomial matrices in x. The k-th truncation of $\Sigma[x]^{m\times m}$ is $\Sigma[x]^{m\times m}_k = \Sigma[x]^{m\times m} \cap \mathbb{S}[x]^m_k$.

An ideal I in $\mathbb{R}[x]$ is a subset of $\mathbb{R}[x]$ such that $I \cdot \mathbb{R}[x] \subseteq I$ and $I + I \subseteq I$. Given $h_1, \ldots, h_m, g_1, \ldots, g_t \in \mathbb{R}[x]$, the ideal generated by $h = (h_1, \ldots, h_m)$ is

$$\mathcal{I}(h) = h_1 \cdot \mathbb{R}[x] + \cdots + h_m \cdot \mathbb{R}[x],$$

and the quadratic module of $g = (g_1, \ldots, g_t)$ is

$$\mathcal{Q}(g) = \Sigma[x] + g_1 \cdot \Sigma[x] + \cdots + g_t \cdot \Sigma[x].$$

The k-th truncation of $\mathcal{I}(h)$ is the set

$$\mathcal{I}_k(h) = h_1 \cdot \mathbb{R}[x]_{k-deg(h_1)} + \cdots + h_m \cdot \mathbb{R}[x]_{k-deg(h_m)}, \tag{2.1}$$

and the k-th truncation of $\mathcal{Q}(g)$ is

$$\mathcal{Q}_k(g) = \Sigma[x]_{2k} + g_1 \cdot \Sigma[x]_{2k-deg(g_1)} + \cdots + g_t \cdot \Sigma[x]_{2k-deg(g_t)}. \tag{2.2}$$

Given $G(x) \in \mathbb{S}[x]^m$, we define the quadratic module of $G(x)$ as

$$\mathcal{Q}(G) := \Sigma[x] + \{\langle R, G\rangle : R \in \Sigma[x]^{m\times m}\} = \Sigma[x] + \langle \Sigma[x]^{m\times m}, G\rangle,$$

and its k-th order truncation as

$$\mathcal{Q}_k(G) := \Sigma[x]_{2k} + \{\langle R, G\rangle : R \in \Sigma[x]^{m\times m}, \deg(R_{(i,j)}G_{(i,j)}) \le 2k, \forall i, j \in [m]\}.$$

For $G^{(1)}(x), \ldots, G^{(s)}(x) \in \mathbb{S}[x]^m$ and $\bar{G}(x) = (G^{(1)}(x), G^{(2)}(x), \ldots, G^{(s)}(x))$, the quadratic module of $\bar{G}(x)$ is defined as

$$\mathcal{Q}(\bar{G}) := \Sigma[x] + \langle \Sigma[x]^{m\times m}, G^{(1)}\rangle + \langle \Sigma[x]^{m\times m}, G^{(2)}\rangle + \cdots + \langle \Sigma[x]^{m\times m}, G^{(s)}\rangle,$$

and its k-th order truncation is the set

$$\mathcal{Q}_k(\bar{G}) := \Sigma[x]_{2k} + \langle R^{(1)}, G^{(1)}\rangle + \langle R^{(2)}, G^{(2)}\rangle + \cdots + \langle R^{(s)}, G^{(s)}\rangle,$$

where $R^{(r)} \in \Sigma[x]^{m\times m}, \deg(R^{(r)}_{(i,j)}G^{(r)}_{(i,j)}) \le 2k, \forall i, j \in [m], r \in [s]$.

$\boxed{2.2}$ Moment and localizing matrices

For $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, denote $|\alpha| = \alpha_1 + \cdots + \alpha_n$. Let

$$\mathbb{N}^n_d = \{\alpha \in \mathbb{N}^n : |\alpha| \le d\}.$$

Let $\mathbb{R}^{\mathbb{N}^n_d}$ be the space of real vectors indexed by $\alpha \in \mathbb{N}^n_d$. A vector in $\mathbb{R}^{\mathbb{N}^n_d}$ is called a truncated moment sequence (tms) of degree d. For $y \in \mathbb{R}^{\mathbb{N}^n_d}$, we define a Riesz functional, $\mathcal{F}_y$, acting on $\mathbb{R}[x]_d$ as

$$\mathcal{F}_y(q) := \sum_{\alpha\in\mathbb{N}^n_d} q_\alpha y_\alpha, \quad \text{for all } q(x) = \sum_{\alpha\in\mathbb{N}^n_d} q_\alpha x^\alpha,$$

where each q_α is a coefficient of the polynomial q. For convenience, we also denote $\langle q, y\rangle := \mathcal{F}_y(q)$. We say that y admits a representing measure supported on a set T if there exists a Borel measure, μ, such that its support $supp(\mu)$ is contained in T and

$$y_\alpha = \int_T x^\alpha d\mu, \quad \forall \alpha \in \mathbb{N}^n_d.$$

For a polynomial $g(x) \in \mathbb{R}[x]_{2k}$ and a tms $y \in \mathbb{R}^{\mathbb{N}^n_{2k}}$, denote $d_g = \lceil deg(g)/2 \rceil$. Define

$$[L_g^{(k)}(y)]_{\alpha,\beta} = \mathcal{F}_y([g(x)[x]_{k-d_g}[x]_{k-d_g}^T]_{\alpha,\beta}) = \sum_{\gamma \in \mathbb{N}^n} g_\gamma y_{\alpha+\beta+\gamma}, \quad \forall \alpha, \beta \in \mathbb{N}^n_{k-d_g}.$$

We call $L_g^{(k)}(y)$ the k-th order localizing matrix of $g(x)$, generated by a tms $y \in \mathbb{R}^{\mathbb{N}^n_{2k}}$. We also define $\langle \cdot, \cdot \rangle_{gy} : \mathbb{R}[x]_{k-d_g} \times \mathbb{R}[x]_{k-d_g} \mapsto \mathbb{R}$ by

$$\langle p, q \rangle_{gy} = \mathcal{F}_y(gpq) = \langle vec(p), L_g^{(k)}(y) vec(q) \rangle, \quad \forall p, q \in \mathbb{R}[x]_{k-d_g}.$$

In the above, $vec(p)$ denotes the coefficient vector of the polynomial p. When $g = 1$ (the constant one polynomial), $L_g^{(k)}(y)$ becomes a k-th order moment matrix and is denoted as

$$M_k(y) := L_1^{(k)}(y).$$

For convenience, we also denote

$$L_g^{(k)}(y) := \operatorname{diag}\left(L_{g_1}^{(k)}(y), \ldots, L_{g_m}^{(k)}(y)\right) \tag{2.3}$$

for a tuple of polynomials $g = (g_1, \ldots, g_m)$.

For a polynomial matrix $G(x) \in \mathbb{S}[x]^m$, we write

$$G(x) = \sum_{\gamma \in \mathbb{N}^n} G_\gamma x^\gamma$$

for some finite family of $\{G_\gamma\} \subset \mathbb{S}^m$. Recall that $s(k) := \binom{n+k}{k}$ is the dimension of the space $\mathbb{R}[x]_k$. We define an (m-block) $s(k)$-vector Gy by

$$(Gy)_\alpha = \sum_{\gamma \in \mathbb{N}^n} G_\gamma y_{\alpha+\gamma}, \forall \alpha \in \mathbb{N}^n_k,$$

which is an $m \times m$ matrix. Then the localizing matrix of the polynomial matrix $G(x)$, generated by $y \in \mathbb{R}^{\mathbb{N}^n_{2k}}$, has the block structure $\{(Gy)_{\alpha+\beta}\}_{\alpha,\beta}$ with $\forall \alpha, \beta \in \mathbb{N}^n_d$, where $d := k - \lceil \deg(G)/2 \rceil$. We denote the k-th localizing matrix of the polynomial matrix $G(x)$ as $L_G^{(k)}(y)$. Its entry has the representation

$$[L_G^{(k)}(y)]_{\alpha,\beta} = \mathcal{F}_y([[x]_d[x]_d^T \otimes G(x)]_{\alpha,\beta}), \forall \alpha, \beta \in \mathbb{N}^n_d,$$

where $\otimes$ stands for the Kronecker product.

For simplicity, we write the k-th order moment matrix and localizing matrix associated with $q(x) \in \mathbb{R}[x]$ and $G(x) \in \mathbb{S}[x]^m$ as

$$M_k(y) = \mathcal{F}_y([x]_k[x]_k^T), \tag{2.4}$$

$$L_q^k(y) = \mathcal{F}_y(q(x)[x]_{d_1}[x]_{d_1}^T), \tag{2.5}$$

$$L_G^k(y) = \mathcal{F}_y([x]_{d_2}[x]_{d_2}^T \otimes G(x)), \tag{2.6}$$

where $d_1 := k - \lceil \deg(q)/2 \rceil$, $d_2 := k - \lceil \deg(G)/2 \rceil$, and $\mathcal{F}_y$ is applied entrywise to the polynomial matrices.

Example 2.1. Let $G(x) = \begin{bmatrix} 1-4x_1x_2 & x_1 \\ x_1 & 4-x_1^2-x_2^2 \end{bmatrix}$. The second-order moment matrix is

$$M_2(y) = \mathcal{F}_y([x]_2[x]_2^T) = \begin{bmatrix} y_{00} & y_{10} & y_{01} & y_{20} & y_{11} & y_{02} \\ y_{10} & y_{20} & y_{11} & y_{30} & y_{21} & y_{12} \\ y_{01} & y_{11} & y_{02} & y_{21} & y_{12} & y_{03} \\ y_{20} & y_{30} & y_{21} & y_{40} & y_{31} & y_{22} \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & y_{13} \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} \end{bmatrix}.$$

The second-order localizing matrix of G, generated by y, is

$$L_G^{(2)} = \mathcal{F}_y([x]_1[x]_1^T \otimes G(x)) := \begin{bmatrix} L_{00} & L_{10} & L_{01} \\ L_{10} & L_{20} & L_{11} \\ L_{01} & L_{11} & L_{02} \end{bmatrix},$$

where

$$L_{00} = \begin{bmatrix} y_{00}-4y_{11} & y_{10} \\ y_{10} & 4y_{00}-y_{20}-y_{02} \end{bmatrix}, L_{10} = \begin{bmatrix} y_{10}-4y_{21} & y_{20} \\ y_{20} & 4y_{10}-y_{30}-y_{12} \end{bmatrix},$$

$$L_{01} = \begin{bmatrix} y_{01}-4y_{12} & y_{11} \\ y_{11} & 4y_{01}-y_{21}-y_{03} \end{bmatrix}, L_{20} = \begin{bmatrix} y_{20}-4y_{31} & y_{30} \\ y_{30} & 4y_{20}-y_{40}-y_{22} \end{bmatrix},$$

$$L_{11} = \begin{bmatrix} y_{11}-4y_{22} & y_{21} \\ y_{21} & 4y_{11}-y_{31}-y_{13} \end{bmatrix}, L_{02} = \begin{bmatrix} y_{02}-4y_{13} & y_{12} \\ y_{12} & 4y_{02}-y_{22}-y_{04} \end{bmatrix}.$$

In fact, for such $G(x)$ with $n = 2$ and $k = 2$, the 2-block 6-vector Gy is given by

$$Gy = \begin{bmatrix} L_{00} & L_{10} & L_{01} & L_{20} & L_{11} & L_{02} \end{bmatrix}^T,$$

and for $\alpha = (0,0)^T, \beta = (1,0)^T \in \mathbb{N}_1^2$, we have $(Gy)_{\alpha+\beta} = L_{10}$.

$\boxed{2.3}$ Flatness

It was shown in [6, 11] that a necessary condition for $y \in \mathbb{R}^{\mathbb{N}_{2k}^n}$ to admit a representing measure in the set

$$E(h) \cap S(g) := \{x \in \mathbb{R}^n : h(x) = 0\} \cap \{x \in \mathbb{R}^n : g(x) \geq 0\}$$

is that

$$L_h^{(k)}(y) = 0, \quad M_k(y) \succeq 0, \quad L_g^{(k)}(y) \succeq 0. \tag{2.7}$$

However, it is typically not sufficient. Let $d' = \max\{1, \lceil deg(h)/2 \rceil, \lceil deg(g)/2 \rceil\}$. If y also satisfies the rank condition

$$\text{rank } M_{k-d'}(y) = \text{rank } M_k(y), \tag{2.8}$$

then y admits a unique finitely atomic measure on $E(h) \cap S(g)$ (cf. [6]). We call y *flat* with respect to $h = 0$ and $g \geq 0$ if both (2.7) and (2.8) are satisfied.

We say $y \in \mathbb{R}^{\mathbb{N}_{2k}^n}$ is flat with respect to $G(x) \succeq 0$ if it satisfies not only the semidefinite constraints

$$M_k(y) \succeq 0, \ L_G^{(k)}(y) \succeq 0, \tag{2.9}$$

but also the rank condition

$$\mathrm{rank} M_{k-d_G}(y) = \mathrm{rank} M_k(y), \tag{2.10}$$

where $d_G = \max\{1, \lceil \deg(G)/2 \rceil\}$.

According to Nie [34, Proposition 2.1], if $y \in \mathbb{R}^{\mathbb{N}^n_{2k}}$ is flat with respect to polynomial matrix inequality $G(x) \succeq 0$, then y admits a unique r-atomic measure on the semi-algebraic set

$$T(G) := \{x \in \mathbb{R}^n : G(x) \succeq 0\},$$

with $r = \mathrm{rank} M_k(y)$.

Based on the above, we say that tms $y \in \mathbb{R}^{\mathbb{N}^n_{2k}}$ is flat with respect to $h = 0$, $g \geq 0$ and $G(x) \succeq 0$ if it satisfies

$$L_h^{(k)}(y) = 0, \quad M_k(y) \succeq 0, \quad L_g^{(k)}(y) \succeq 0, \quad L_G^{(k)}(y) \succeq 0 \tag{2.11}$$

and the rank condition

$$\mathrm{rank} M_{k-\bar{d}}(y) = \mathrm{rank} M_k(y), \tag{2.12}$$

where $\bar{d} = \max\{1, \lceil \deg(h)/2 \rceil, \lceil \deg(g)/2 \rceil, \lceil \deg(G)/2 \rceil\}$. In such case, y admits a unique r-atomic measure, μ^*, on $E(h) \cap S(g) \cap T(G)$, with $r = \mathrm{rank} M_k(y)$.

For $y \in \mathbb{R}^{\mathbb{N}^n_d}$ and $t \leq d$, denote the truncation of y as

$$y|_t = (y_\alpha)_{\alpha \in \mathbb{N}^n_t}.$$

For two tms $y \in \mathbb{R}^{\mathbb{N}^n_k}$ and $z \in \mathbb{R}^{\mathbb{N}^n_l}$ with $k < l$, we say that y is a truncation of z (equivalently, z is an extension of y), if $y = z|_k$. For such a case, y is called a flat truncation of z if y is flat, and z is a flat extension of y if z is flat.

3 Solving the PSDCP

In this section, we study how to find all real solutions of the PSDCP, if there are a finite number of them. We formulate it as a sequence of polynomial optimization problems and compute the solutions in order. A matrix-type semidefinite relaxation method is proposed.

3.1 Reformulation as polynomial optimization

Recall that $x \in \mathbb{R}^n$ is a solution of the PSDCP if and only if

$$Q_1(x) \succeq 0, \quad Q_2(x) \succeq 0 \quad \text{and} \quad \langle Q_1(x), Q_2(x) \rangle = 0, \tag{3.1}$$

where $Q_1(x)$, $Q_2(x) \in \mathbb{S}[x]^m$. Denote the set of all real solutions to (3.1) by

$$S(Q_1, Q_2) = \{x \in \mathbb{R}^n : Q_1(x), Q_2(x) \in \mathbb{S}^m_+, \mathrm{trace}(Q_1(x)Q_2(x)) = 0\}. \tag{3.2}$$

The following lemma gives an equivalent condition for the orthogonality of two positive semidefinite matrices.

Lemma 3.1. *Let A, B be two $m \times m$ symmetric positive semidefinite matrices. Then,* $\mathrm{trace}(AB) = 0$ *if and only if* $AB = 0$.

Proof. Obviously, $AB = 0$ implies $\mathrm{trace}(AB) = 0$. Next, we prove that the converse also holds true.

Since $A \in \mathbb{S}^m_+$, there exists an orthogonal matrix $P \in \mathbb{R}^{m\times m}$ such that $A = P\Sigma P^T$, where $\Sigma = \mathrm{diag}(\lambda_1, \ldots, \lambda_r, 0, \ldots, 0)$ with $r = \mathrm{rank}(A)$ and $\lambda_i > 0 (i = 1, \ldots, r)$. Let $C = P^T BP$. Then, $C \in \mathbb{S}^m_+$ due to $B \in \mathbb{S}^m_+$. Note that

$$AB = P\Sigma P^T B = P\Sigma CP^T. \tag{3.3}$$

We have

$$\mathrm{trace}(\Sigma C) = \mathrm{trace}(AB) = 0. \tag{3.4}$$

Write $C = \begin{bmatrix} C_1 & C_2 \\ C_2^T & C_3 \end{bmatrix}$, where $C_1 \in \mathbb{S}^r$. Then,

$$\Sigma C = \begin{bmatrix} \Sigma_1 C_1 & \Sigma_1 C_2 \\ 0 & 0 \end{bmatrix}.$$

So, by (3.4),

$$\mathrm{trace}(\Sigma_1 C_1) = 0. \tag{3.5}$$

Since $C \in \mathbb{S}^m_+$, we have $c_{ii} \geq 0 (i = 1, \ldots, r)$. This, together with (3.5) and $\lambda_i > 0 (i = 1, \ldots, r)$, gives $c_{ii} = 0$. Hence, $C_1 = 0$, which further implies that $C_2 = 0$. So $\Sigma C = 0$. It then follows from (3.3) that $AB = 0$. The proof is completed. $\square$

By Lemma 3.1, (3.2) can be written equivalently as

$$S(Q_1, Q_2) = \{x \in \mathbb{R}^n : Q_1(x), Q_2(x) \in \mathbb{S}^m_+, Q_1(x)Q_2(x) = 0\}. \tag{3.6}$$

Let $H(x) = Q_1(x)Q_2(x)$. Denote the polynomial tuple

$$h(x) := (H_{11}(x), \ldots, H_{1m}(x), H_{21}(x), \cdots, H_{mm}(x)) \tag{3.7}$$

and the set

$$V_{\mathbb{R}}(h) = \{x \in \mathbb{R}^n : h(x) = 0\}. \tag{3.8}$$

Then, (3.2) can be written as

$$S(Q_1, Q_2) = \{x \in \mathbb{R}^n : Q_1(x), Q_2(x) \in \mathbb{S}^m_+\} \cap V_{\mathbb{R}}(h). \tag{3.9}$$

Let

$$d = \max\{1, deg(h), deg(Q_1(x)), deg(Q_2(x))\} \tag{3.10}$$

and $f \in \Sigma[x]_{2k_0}$ be a generic sum-of-squares polynomial with $k_0 = \lceil d/2 \rceil$. Consider the optimization problem

$$\begin{cases} \min & f(x) \\ \text{s.t.} & h(x) = 0, \\ & Q_1(x) \succeq 0, \\ & Q_2(x) \succeq 0. \end{cases} \tag{3.11}$$

Clearly, $x \in S(Q_1, Q_2)$ if and only if x is feasible for (3.11). Note that we consider (3.11) instead of the optimization problem

$$\begin{cases} \min & f(x) \\ \text{s.t.} & \mathrm{trace}(Q_1(x)Q_2(x)) = 0, \\ & Q_1(x) \succeq 0, \\ & Q_2(x) \succeq 0. \end{cases} \tag{3.12}$$

This is because (3.11) has more equations than (3.12) and tighter relaxation problems can be obtained when using Lasserre's relaxation method.

3.2 Compute all solutions of the PSDCP

Suppose $S(Q_1, Q_2)$ is nonempty and finite. $f(x)$ achieves different values at different $x \in S(Q_1, Q_2)$ when it is randomly chosen in $\Sigma[x]_{2k_0}$, where $k_0 = \lceil d/2 \rceil$ with d given in (3.10). We order them monotonically as

$$f_1 < f_2 < \cdots < f_N.$$

Let

$$\begin{aligned} S_i &= S(Q_1, Q_2) \cap \{x \in \mathbb{R}^n : f(x) = f_i\} \\ &= \{x \in \mathbb{R}^n : Q_1(x) \succeq 0, Q_2(x) \succeq 0, h(x) = 0, f(x) = f_i\}, \quad i = 1, \cdots, N. \end{aligned} \tag{3.13}$$

Then,

$$S(Q_1, Q_2) = S_1 \cup S_2 \cup \cdots \cup S_N.$$

3.2.1 The first set S_1

Note that S_1 is the set of optimal solutions of the polynomial optimization problem:

$$\left\{ \begin{array}{rl} f_1 := \min & f(x) \\ \text{s.t.} & h(x) = 0, \\ & Q_1(x) \succeq 0, \\ & Q_2(x) \succeq 0. \end{array} \right. \tag{3.14}$$

The feasible set of problem (3.14) is a semi-algebraic set and can be represented by a finite number of scalar polynomial inequalities. Denote the principal minors of $Q_1(x)$ and $Q_2(x)$ as $(g_{I_1}(x))$ and $(g_{I_2}(x))$ with all $I_1, I_2 \subseteq [m]$, respectively. Then, the constraints $\{Q_1(x) \succeq 0,\ Q_2(x) \succeq 0\}$ can be described explicitly by finite scalar polynomial inequalities $\{g_{I_j}(x) \geq 0,\ \forall I_j \subseteq [m], j = 1, 2\}$. So, one might consider to apply Lasserre's semidefinite relaxation method for the case of the scalar-type polynomial optimization problem to solve problem (3.14), and the k-th order scalar-type semidefinite relaxation of (3.14) is

$$\left\{ \begin{array}{rl} \tilde{r}_1^k := \min & \langle f, y \rangle \\ \text{s.t.} & \langle 1, y \rangle = 1, \\ & M_k(y) \succeq 0, \\ & L_h^{(k)}(y) = 0, \\ & L_{g_{I_j}}^{(k)}(y) \succeq 0, \forall I_j \subseteq [m], j = 1, 2, \\ & y \in \mathbb{R}^{\mathbb{N}_{2k}^n}, \end{array} \right. \tag{3.15}$$

where $M_k(y), L_h^{(k)}(y), L_{g_{I_j}}^{(k)}(y)$ are the moment matrix and localizing matrices defined by (2.4) and (2.5), respectively. However, a general $m \times m$ polynomial matrix $Q(x)$ has exponentially many $2^m - 1$ principal minors typical in this scalar-type representation, and they have much higher degrees. This is a big disadvantage for using them in practice.

Based on the above observation, we apply matrix-type semidefinite relaxations to solve (3.14) in this paper. For $k = k_0, k_0 + 1, \ldots$, the k-th order semidefinite relaxation of (3.14) is

$$\left\{ \begin{array}{rl} r_1^k := \min & \langle f, y \rangle \\ \text{s.t.} & \langle 1, y \rangle = 1, \\ & M_k(y) \succeq 0, \\ & L_h^{(k)}(y) = 0, \\ & L_{Q_j}^{(k)}(y) \succeq 0, j = 1, 2, \\ & y \in \mathbb{R}^{\mathbb{N}_{2k}^n}, \end{array} \right. \tag{3.16}$$

where $M_k(y), L_h^{(k)}(y), L_{Q_j}^{(k)}(y)$ are the moment matrix and localizing matrices defined by (2.4)–(2.6).

Lemma 3.2. *For* (3.14) *and* (3.16)*, we have* $r_1^k \le f_1$ *for all* k.

Proof. Suppose $x \in \mathbb{R}^n$ is feasible for (3.14), and $y = [x]_{2k}$. By the definition, we have $\langle f, y\rangle = \mathcal{F}_y(f) = f(x)$, $M_k(y) = [x]_k[x]_k^T \succeq 0$, and $L_{h_i}^{(k)}(y) = 0, i = 1, \cdots, m^2$.

In addition, $L_{Q_1}^{(k)}(y) \succeq 0$ because $L_{Q_1}^{(k)}(y)$ is the Kronecker product of $Q_1(x) \succeq 0$ and $M_{k-d_{Q_1}}(y) \succeq 0$. So does $L_{Q_2}^{(k)}(y) \succeq 0$. Hence $y = [x]_{2k}$ is feasible for the SDP problem (3.16). Thus, the problem (3.16) is a relaxation of problem (3.14), and $r_1^k \le f_1$ for any relaxation order k. $\square$

Consider the sum-of-square optimization problem

$$\begin{cases} \lambda_1^k := \max & \lambda \\ \quad\quad\quad \text{s.t.} & f - \lambda \in \mathcal{I}_{2k}(h) + \mathcal{Q}_k(Q_1, Q_2), \end{cases} \tag{3.17}$$

where $\mathcal{I}_{2k}(h)$ is the $2k$-th truncation of the ideal generated by $h(x)$, and $\mathcal{Q}_k(Q_1, Q_2)$ is the k-th truncation of the quadratic module generated by (Q_1, Q_2). We have the following result.

Theorem 3.3. (3.16) *and* (3.17) *are dual to each other.*

Proof. Let $\langle f, y\rangle = \sum_{\alpha\in\mathbb{N}^n} f_\alpha y_\alpha, M_k(y) = \sum_{\alpha\in\mathbb{N}^n} B_\alpha y_\alpha, L_{h_i}^{(k)}(y) = \sum_{\alpha\in\mathbb{N}^n} H_\alpha^{(i)} y_\alpha, i = 1, \ldots, m^2$, $L_{Q_j}^{(k)}(y) = \sum_{\alpha\in\mathbb{N}^n} C_\alpha^{(j)} y_\alpha, j = 1, 2$, where $B_\alpha, H_\alpha^{(i)}, C_\alpha^{(j)}$ are real symmetric. The Lagrangian function of (3.16) is

$$L(y, X, Y, Z) = \sum_{\alpha\in\mathbb{N}^n} f_\alpha y_\alpha - \langle X, M_k(y)\rangle - \sum_{j=1}^{2} \langle Y_j, L_{Q_j}^{(k)}(y)\rangle - \sum_{i=1}^{m^2} \langle Z_i, L_{h_i}^{(k)}(y)\rangle,$$

where $Y = (Y_1, Y_2)$ and $Z = (Z_1, \ldots, Z_{m^2})$. Thus,

$$\begin{aligned} g(X, Y, Z) =& \inf_y L(y, X, Y, Z) \\ =& \inf_y \Bigg[\sum_{\alpha\in\mathbb{N}^n} f_\alpha y_\alpha - \langle X, B_0 + \sum_{\alpha\neq 0} B_\alpha y_\alpha\rangle - \sum_{j=1}^{2} \langle Y_j, C_0^{(j)} + \sum_{\alpha\neq 0} C_\alpha^{(j)} y_\alpha\rangle \\ & - \sum_{i=1}^{m^2} \langle Z_i, H_0^{(i)} + \sum_{\alpha\neq 0} H_\alpha^{(i)} y_\alpha\rangle \Bigg] \\ =& f_0 + \langle X, -B_0\rangle + \sum_{j=1}^{2} \langle Y_j, -C_0^{(j)}\rangle + \sum_{i=1}^{m^2} \langle Z_i, -H_0^{(i)}\rangle := \lambda, \end{aligned}$$

and

$$f_\alpha = \langle X, B_\alpha\rangle + \sum_{j=1}^{2} \langle Y_j, C_\alpha^{(j)}\rangle + \sum_{i=1}^{m^2} \langle Z_i, H_\alpha^{(i)}\rangle, \quad \forall \alpha \neq 0. \tag{3.18}$$

Then, the dual problem of (3.16) is

$$\left\{\begin{array}{ll} \max & \lambda \\ \text{s.t.} & f_0 - \lambda = \langle X, B_0\rangle + \sum_{j=1}^{2}\langle Y_j, C_0^{(j)}\rangle + \sum_{i=1}^{m^2}\langle Z_i, H_0^{(i)}\rangle \\ & f_\alpha = \langle X, B_\alpha\rangle + \sum_{j=1}^{2}\langle Y_j, C_\alpha^{(j)}\rangle + \sum_{i=1}^{m^2}\langle Z_i, H_\alpha^{(i)}\rangle, 0 \neq |\alpha| \leq 2k, \\ & X, Y_1, Y_2 \succeq 0. \end{array}\right. \tag{3.19}$$

Suppose that (λ, X, Y, Z) is a feasible point of (3.19). Multiplying both sides of the equality constraint in problem (3.19) by x^α and adding them, we obtain

$$f(x) - \lambda = \langle X, \sum_{\alpha\in\mathbb{N}^n} B_\alpha x^\alpha\rangle + \sum_{j=1}^{2}\langle Y_j, \sum_{\alpha\in\mathbb{N}^n} C_\alpha^{(j)} x^\alpha\rangle + \sum_{i=1}^{m^2}\langle Z_i, \sum_{\alpha\in\mathbb{N}^n} H_\alpha^{(i)} x^\alpha\rangle. \tag{3.20}$$

According to the definitions of the moment matrix and localizing matrix associated with the scalar polynomial and polynomial matrix given in (2.4)–(2.6), we let $y = [x]_{2k}$ and get

$$M_k(y) = \sum_{\alpha\in\mathbb{N}^n} B_\alpha y_\alpha = [x]_k [x]_k^T, \tag{3.21}$$

$$L_{h_i}^{(k)}(y) = \sum_{\alpha\in\mathbb{N}^n} H_\alpha^{(i)} y_\alpha = [x]_{k-d_{h_i}} [x]_{k-d_{h_i}}^T h_i(x), \quad i = 1, 2, \ldots, m^2, \tag{3.22}$$

$$L_{Q_j}^{(k)}(y) = \sum_{\alpha\in\mathbb{N}^n} C_\alpha^{(j)} y_\alpha = [x]_{k-d_{Q_j}} [x]_{k-d_{Q_j}}^T \otimes Q_j(x), \quad j = 1, 2. \tag{3.23}$$

Since X, Y_1, Y_2 are positive semidefinite matrices, there exist vectors such that $X = \sum_l q_{0l} q_{0l}^T$, $Y_1 = \sum_s q_{1s} q_{1s}^T, Y_2 = \sum_r q_{2r} q_{2r}^T$. Combining (3.20)–(3.23), we have

$$\begin{aligned} f(x) - \lambda = & \sum_l \langle q_{0l}, [x]_k\rangle^2 + \sum_s \langle q_{1s} q_{1s}^T, [x]_{k-d_{Q_1}} [x]_{k-d_{Q_1}}^T \otimes Q_1(x)\rangle \\ & + \sum_r \langle q_{2r} q_{2r}^T, [x]_{k-d_{Q_2}} [x]_{k-d_{Q_2}}^T \otimes Q_2(x)\rangle \\ & + \sum_{i=1}^{m^2} \langle Z_i, [x]_{k-d_{h_i}} [x]_{k-d_{h_i}}^T h_i(x)\rangle, \end{aligned}$$

where $p_0 = \sum_l \langle q_{0l}, [x]_k\rangle^2 \in \Sigma[x]_{2k}$, and

$$\langle Z_i, [x]_{k-d_{h_i}} [x]_{k-d_{h_i}}^T h_i(x)\rangle \in h_i(x) \cdot \mathbb{R}[x]_{2k-\deg(h_i)}, i = 1, 2, \cdots, m^2.$$

Let $R_1(x) = \sum_s L_s^{(1)}(x) L_s^{(1)}(x)^T$, where $L_s^{(1)}(x) = \hat{L}_s^{(1)} [x]_{k-d_{Q_1}} \in \mathbb{R}[x]^m$ with $\hat{L}_s^{(1)} \in \mathbb{R}^{m\times s(k-d_{Q_1})}$. Arrange its column vectors in order to get the vector $q_{1s} \in \mathbb{R}^{m\cdot s(k-d_{Q_1})}$. We can prove

$$\begin{aligned} \langle R_1, Q_1\rangle = & \sum_s \langle \hat{L}_s^{(1)} [x]_{k-d_{Q_1}} [x]_{k-d_{Q_1}}^T \hat{L}_s^{(1)T}, Q_1(x)\rangle \\ = & \sum_s \langle [x]_{k-d_{Q_1}} [x]_{k-d_{Q_1}}^T \otimes Q_1(x), q_{1s} q_{1s}^T\rangle. \end{aligned}$$

Similarly, we have

$$\langle R_2, Q_2\rangle = \sum_r \langle [x]_{k-d_{Q_2}} [x]_{k-d_{Q_2}}^T \otimes Q_2(x), q_{2r} q_{2r}^T\rangle.$$

Thus, $f(x) - \lambda \in \mathcal{I}_{2k}(h) + \mathcal{Q}_k(Q_1, Q_2)$. Based on the above analysis, we know (3.16) and (3.17) are dual to each other. The proof is completed. □

Remark 3.4. It follows from Theorem 3.3 that $\lambda_1^k \le r_1^k$ for any relaxation order k by weak duality. So both r_1^k and λ_1^k are lower bounds of f_1. Moreover, the sequences $\{r_1^k\}$ and $\{\lambda_1^k\}$ monotonically non-decrease as k increases, i.e.,

$$r_1^{k_0} \le r_1^{k_0+1} \le \ldots \le f_1,$$

and

$$\lambda_1^{k_0} \le \lambda_1^{k_0+1} \le \ldots \le f_1.$$

On the other hand, we randomly choose $f(x) \in \Sigma[x]_{2k_0}$ in practice. Generically, it holds that $f(x) \in int(\Sigma[x]_{2k_0})$. This means that $\lambda = 0$ is an interior point of (3.17). Hence, the strong duality also holds if (3.16) is feasible, i.e., (3.16) has a minimizer and (3.16)–(3.17) has the same optimum value $r_1^k = \lambda_1^k$, for all $k \ge k_0$ (cf. [1, Section 2.4]).

Further, by the results given in [19, 20, 23], we have the following asymptotic convergence result.

Theorem 3.5. *Suppose that the Archimedean condition holds for the feasible set of* (3.14)*, i.e., there exists a polynomial* $p \in \mathcal{I}(h)+\mathcal{Q}(Q_1, Q_2)$ *such that the level set* $\{x \in \mathbb{R}^n : p(x) \ge 0\}$ *is compact. Then,* $\lim\limits_{k\to\infty} r_1^k = f_1$ *and* $\lim\limits_{k\to\infty} \lambda_1^k = f_1$.

Proof. By Lemma 3.2 and Theorem 3.3, we have $\lambda_1^k \le r_1^k \le f_1$ for any relaxation order k. Under the assumption, Hol and Scherer [19, 20] and Kojima [23] have proved that $\lim\limits_{k\to\infty} \lambda_1^k = f_1$. From what precedes, the result follows. □

Remark 3.6. The assumption of Theorem 3.5 is not very restrictive. For instance, suppose that there exist a global minimizer contained in the ball $\{x \in \mathbb{R}^n : \rho - \|x\|^2 \ge 0\}$, where $\rho > 0$ is a priori bound. Then, we can add $\rho - \|x\|^2 \ge 0$ to the constraints so that the Archimedean condition holds for such feasibility set.

3.2.2 The second and other sets S_j

Suppose that f_t and S_t are known. We investigate how to compute f_{t+1} and S_{t+1}, if they exist. For a small $\varepsilon > 0$, we consider the optimization problem

$$\left\{\begin{array}{rl} f_t^+ = \min & f(x) \\ \text{s.t.} & h(x) = 0, \\ & Q_1(x) \succeq 0, \\ & Q_2(x) \succeq 0, \\ & f(x) \ge f_t + \varepsilon. \end{array}\right. \tag{3.24}$$

Clearly, $f_{t+1} = f_t^+$ if and only if

$$0 < \varepsilon < f_{t+1} - f_t. \tag{3.25}$$

Let $g(x) = f(x) - f_t - \varepsilon$. We construct the optimization problem

$$\left\{\begin{array}{rl} r_{t+1}^k := \min & \langle f, y \rangle \\ \text{s.t.} & \langle 1, y \rangle = 1, \\ & M_k(y) \succeq 0, \\ & L_h^{(k)}(y) = 0, \\ & L_g^{(k)}(y) \succeq 0, \\ & L_{Q_1}^{(k)}(y) \succeq 0, \\ & L_{Q_2}^{(k)}(y) \succeq 0, \\ & y \in \mathbb{R}^{\mathbb{N}_{2k}^n} \end{array}\right. \tag{3.26}$$

and a sum-of-squares problem

$$\left\{\begin{array}{rl} \lambda_{t+1}^k := \max & \lambda \\ \text{s.t.} & f - \lambda \in \mathcal{I}_{2k}(h) + \mathcal{Q}_k(g, Q_1, Q_2), \end{array}\right. \tag{3.27}$$

where $\mathcal{I}_{2k}(h)$ is the $2k$-th truncation of the ideal generated by $h(x)$, and $\mathcal{Q}_k(g, Q_1, Q_2)$ is the k-th truncation of the quadratic module of (g, Q_1, Q_2)

$$\mathcal{Q}_k(g, Q_1, Q_2) = \Sigma[x]_{2k} + g \cdot \Sigma[x]_{2k-\deg(g)} + \mathcal{Q}_k(Q_1, Q_2).$$

Similarly to Lemma 3.2 and Theorem 3.3, we have the following result.

Lemma 3.7. *For* (3.26) *and* (3.27), *we have* $r_{t+1}^k \le f_t^+$ *for all* k. *Moreover,* (3.26) *and* (3.27) *are dual to each other.*

Next, we discuss how to determine the value of ε that satisfies $0 < \varepsilon < f_{t+1} - f_t$. Note that we typically do not know whether or not f_{t+1} exists in practice. Consider the optimization problem

$$\left\{\begin{array}{rl} f_t^- = \max & f(x) \\ \text{s.t.} & h(x) = 0, \\ & Q_1(x) \succeq 0, \\ & Q_2(x) \succeq 0, \\ & f(x) \le f_t + \varepsilon. \end{array}\right. \tag{3.28}$$

Its optimal value, f_t^-, can be computed by semidefinite relaxations similar to (3.26) and (3.16).

Lemma 3.8. *Suppose that* $S(Q_1, Q_2)$ *is nonempty and finite. Let* f_t *be the value of* $f(x)$ *on the set* S_t. *For* $t = 1, 2, \ldots, N-1$ *and* $\varepsilon > 0$, *we have*

1. *If* f_{t+1} *exists, then* $f_t^- = f_t$ *if and only if* ε *satisfies* (3.25).
2. *If* $f_t^- = f_t$ *for some* $\varepsilon > 0$, *and* (3.24) *is infeasible, then* f_{t+1} *does not exist; that is,* f_t *is the maximum value on* $S(Q_1, Q_2)$.
3. *If* f_t *is the maximum value of* $f(x)$ *on* $S(Q_1, Q_2)$, *then* $f_t^- = f_t$ *for any* $\varepsilon > 0$.

Proof. (i) If ε satisfies (3.25), then $f_{t+1} > f_t + \varepsilon$. Because f_t^- is the maximum feasible value on $S(Q_1, Q_2)$ less than or equal to $f_t + \varepsilon$, the result is obvious. Conversely, if $f_t^- = f_t$, then there are no feasible values in the interval $(f_t, f_t + \varepsilon)$. So $f_{t+1} > f_t + \varepsilon$ if f_{t+1} exists.

(ii) By (i), $f_t^- = f_t$ means $f_{t+1} > f_t + \varepsilon$, if f_{t+1} exists. Since (3.24) is infeasible, there are no feasible values larger than f_t on $S(Q_1, Q_2)$. Thus, f_t is the maximum value on $S(Q_1, Q_2)$.

(iii) If $f_t = f_N$ is the maximum value on $S(Q_1, Q_2)$, then for any $x \in S(Q_1, Q_2)$, we have $f(x) \le f_t$. Hence, $f_t^- = f_t$ for any $\varepsilon > 0$. □

For numerical reasons, the value $\varepsilon > 0$ cannot be too small. A typical value like 0.05 is preferable in computations. Let $\varepsilon = 0.05$ in (3.28). If $f_t^- > f_t$, we decrease ε by half and solve (3.28) again. After repeating this process several times, we can obtain $f_t^- = f_t$.

3.2.3 An algorithm for computing all real solutions of PSDCP

Based on the above analysis, we propose a semidefinte relaxation algorithm to compute all real solutions of the PSDCP, if there are finitely many ones. First, we solve (3.14) to get S_1, if $S(Q_1, Q_2)$ is nonempty. After obtaining S_1, we turn to solve (3.24). If (3.24) is infeasible, then $S_1 = S(Q_1, Q_2)$ is the solution set of the $PSDCP$, and we stop. Otherwise, we determine f_2 and S_2 by solving problem (3.24). Repeating this procedure, we can get the solution set $S(Q_1, Q_2) = S_1 \cup S_2 \cup \cdots \cup S_N$. The algorithm is presented as follows.

Algorithm 3.9 (Computing all real solutions of PSDCP)**.**

Step 0. Let $k_0 = \lceil d/2 \rceil$ with d given in (3.10). Choose a random polynomial $f(x) \in \Sigma[x]_{2k_0}$. Set $t := 0$ and $k := k_0$.

Step 1. If (3.16) is infeasible, then $S(Q_1, Q_2) = \emptyset$, and stop. Otherwise, solve (3.16) to get an optimal solution $y^{1,k}$ and the optimal value $r_{1,k}$.

Step 2. If the rank condition (2.12) is satisfied for some $s \in [k_0, k]$, i.e., it holds that $\text{rank}M_{s-k_0}(y^{1,k}) = \text{rank}M_s(y^{1,k})$, then set $S(Q_1, Q_2) := S_1$, where S_1 is the set of minimizers of (3.14). Let $\mathcal{F} = \{r_{1,k}\}$, $k := k_0$, $t := t + 1$, and go to Step 3. If such s does not exist, let $k := k + 1$ and go to Step 1.

Step 3. Let $\varepsilon = 0.05$. Compute the optimal value f_t^- of (3.28). If $f_t^- > f_t$, let $\varepsilon = \varepsilon/2$ and solve (3.28) again. Repeat this procedure until $f_t^- = f_t$.

Step 4. Solve (3.26). If it is infeasible, then the PSDCP has no more real solutions, and stop. Otherwise, compute an optimal solution $y^{t+1,k}$ and the optimal value $r_{t+1,k}$.

Step 5. If the rank condition (2.12) is satisfied for some $s \in [k_0, k]$, i.e., it holds that $\text{rank}M_{s-k_0}(y^{t+1,k}) = \text{rank}M_s(y^{t+1,k})$, then update $S(Q_1, Q_2) := S(Q_1, Q_2) \cup S_{t+1}$, where S_{t+1} is the set of minimizers of (3.24). Set $\mathcal{F} := \mathcal{F} \cup \{r_{t+1,k}\}, k := k_0, t := t + 1$, and go to Step 3. If such s does not exist, let $k := k + 1$ and go to Step 4.

Remark 3.10. Algorithm 3.9 can be implemented by the software `GloptiPoly 3` [16], which solves the generalized problem of moments. In Step 0, we choose $F = [x]_{k_0}^T R^T R[x]_{k_0}$, where R is a random square matrix obeying Gaussian distribution. In Steps 1 and 4, the semidefinite relaxation problems (3.16) and (3.26) are solved by the semidefinite programming solvers `YALMIP` [28] and `SeDuMi` [38]. In Steps 2 and 5, we evaluate the rank of a matrix as the number of its singular values that are not smaller than 10^{-6}, which is a standard procedure in numerical linear algebra (see [7, 9]). If the rank condition (2.12) is satisfied, Henrion and Lasserre's method [13] is used to extract the optimal solutions.

Now, we prove that Algorithm 3.9 converges in finitely many steps if the set $V_{\mathbb{R}}(h)$ as in (3.8) is finite.

Theorem 3.11. *For $t = 1, 2, \ldots, N$, let f_t be the value of $f(x)$ on the set S_t if it exists. Then, we have the following properties:*

(i) *If the relaxation* (3.16) *(resp.* (3.26)*) is infeasible for some order k, then the feasible set of* (3.14) *(resp.* (3.24)*) is empty.*

(ii) *If $V_{\mathbb{R}}(h)$ as in* (3.8) *is a compact set and the feasible set of* (3.14) *(resp.* (3.24)*) is empty, then for all k big enough, the relaxation* (3.16) *(resp.* (3.26)*) is infeasible.*

(iii) *Suppose* f_{t+1} *exists and* $0 < \varepsilon < f_{t+1} - f_t$. *If* $V_{\mathbb{R}}(h)$ *as in* (3.8) *is a finite set, then for all* k *big enough, the rank condition* (2.12) *must be satisfied and*

$$r^k_{t+1} = \lambda^k_{t+1} = f_{t+1}.$$

Proof. (i) If the relaxation (3.16) (resp. (3.26)) is infeasible for some order k, then the feasible set of (3.14) (resp. (3.24)) must be empty. This is because, if otherwise (3.14) (resp. (3.24)) has a feasible point, say, u (resp. $\tilde{u}$), then the tms $[u]_{2k}$ (resp. $[\tilde{u}]_{2k}$) generated by u (resp. $\tilde{u}$) must be feasible for (3.26) (resp. (3.26)).

(ii) The ideal $I(h)$ is archimedean because $V_{\mathbb{R}}(h)$ is compact and $-\|h\|^2 \geq 0$ defines a compact set in $\mathbb{R}^n$. If the feasible set of (3.14) (resp. (3.24)) is empty, then $-Q_1(x) \nsucceq 0$ or $-Q_2(x) \nsucceq 0$ for any $x \in V_{\mathbb{R}}(h)$ (resp. for all $x \in V_{\mathbb{R}}(h) \cap \{g(x) \geq 0\}$). By Corollary 3.16 of Klep and Schweighofer [22],

$$-1 \in \mathcal{I}_{2k}(h) + \mathcal{Q}_k(Q_1, Q_2) \text{ (resp. } -1 \in \mathcal{I}_{2k}(h) + \mathcal{Q}_k(g, Q_1, Q_2)),$$

Thus, for all k big enough, (3.17) (resp. (3.27)) is unbounded from above, which implies the infeasibility of (3.16) (resp. (3.26)) by weak duality.

(iii) When the set $V_{\mathbb{R}}(h)$ is finite, the Lasserre's hierarchy (3.26)-(3.27) must have finite convergence, and the rank condition (2.12) must be satisfied, when k is sufficiently large. This can be implied by Theorem 1.1 of [33] and Proposition 4.6 of [26]. □

Remark 3.12. By item (ii) of Theorem 3.11, for any f, (3.16) (resp. (3.26)) is infeasible for some k if $V_{\mathbb{R}}(h)$ is a compact set and (3.14) (resp. (3.24)) is infeasible. This implies that our algorithm can always obtain a certificate for the infeasibility of (3.14) (resp. (3.24)).

Remark 3.13. By item (iii) of Theorem 3.11, the Lasserre's hierarchy must have finite convergence if $V_{\mathbb{R}}(h)$ is finite. Note that for generic symmetric polynomial matrices $Q_1(x), Q_2(x) \in \mathbb{S}^m$, the tuple h as in (3.8) with m^2 polynomials is also generic. Then, for the $n \leq m^2$ case, $V_{\mathbb{R}}(h)$ is a finite set. This can be implied by [29, Corollary A.2]. This means that our algorithm converges in finite steps for generic $Q_1(x), Q_2(x) \in \mathbb{S}^m$ if $n \leq m^2$.

On the other hand, the asymptotic convergence of r^k_{t+1} and λ^k_{t+1} can also be established even if $V_{\mathbb{R}}(h)$ is not a finite set. When the feasible set of (3.24) is nonempty, if we add $\rho - \|x\|^2 \geq 0$ to the constraints, where $\rho > 0$ is a sufficiently large number, r^k_{t+1} and λ^k_{t+1} have asymptotic convergence to f_{t+1}. This can be implied by the results given in [19, 20, 23] and the fact that the feasible set is Archimedean. In fact, the finite convergence occurred in all our numerical experiments.

4 Numerical experiments

In this section, we present some numerical experiments to solve the PSDCP by Algorithm 3.9. We used the softwares `YALMIP` [28], `SeDuMi` [38], and `GloptiPoly 3` [16] to solve the Lasserre's hierarchy of matrix-type semidefinite relaxations. We also give the numerical results for each example by the scalar-type method that uses the semidefinite relaxation (3.15). The experiments were implemented on a laptop with an Intel Core i7-8550U CPU (1.80 GHz) and 8 GB of RAM, using Matlab R2020a. We display 4 decimal digits for numerical numbers.

Example 4.1. Consider the $PSDCP(Q_1, Q_2)$ with

$$Q_1(x) = \begin{bmatrix} 2x_1^4 - x_1^2x_2 + 2x_2^2 & x_1^3x_2 - x_1x_2^2 - 2 \\ x_1^3x_2 - x_1x_2^2 - 2 & x_1^2 - 4x_1^2x_2 + x_2^2 + 1 \end{bmatrix}$$

and

$$Q_2(x) = \begin{bmatrix} x_1 + x_2^2 & x_2 - x_1 \\ x_2 - x_1 & x_1^2 + x_2 \end{bmatrix}.$$

We solved it by Algorithm 3.9 with $k_0 = 3$. It took about 2.74 seconds for Algorithm 3.9 to find two real solutions $(-1.0000, -1.0000)^T$ and $(0.0000, 1.0000)^T$. We also solved it by the scalar-type method with $k_0 = 4$. It took 3.39 seconds to find all real solutions.

Example 4.2. Consider the $PSDCP(Q_1, Q_2)$ with

$$Q_1(x) = \begin{bmatrix} \frac{1}{2}x_1^6 - \frac{1}{3}x_1^2x_2 + 2x_2^2 & \frac{2}{3}x_1^4x_2 - \frac{1}{2}x_1x_2^2 - 2 \\ \frac{2}{3}x_1^4x_2 - \frac{1}{2}x_1x_2^2 - 2 & 2x_1^2 - \frac{1}{4}x_1^4x_2 + x_2^2 + 1 \end{bmatrix}$$

and

$$Q_2(x) = \begin{bmatrix} x_1 + x_2^2 & x_2 - \frac{1}{2}x_1^2 \\ x_2 - \frac{1}{2}x_1^2 & x_1^2 + x_2 \end{bmatrix}.$$

We solved it by Algorithm 3.9 with $k_0 = 4$. It took about 1.36 seconds for Algorithm 3.9 to find the unique real solution $(0.0000, 1.0000)^T$. We also solved it by the scalar-type method with $k_0 = 6$. It took 2.66 seconds to find the real solution.

Example 4.3. Consider the $PSDCP(Q_1, Q_2)$ with

$$Q_1(x) = \begin{bmatrix} 2 - 2x_1^4 - 4x_1^2x_2^2 - 2x_2^4 & 3 - x_1^3x_2 - x_1x_2^3 \\ 3 - x_1^3x_2 - x_1x_2^3 & 5 - x_1^4 - 4x_1^2x_2^2 - x_2^4 \end{bmatrix}$$

and

$$Q_2(x) = \begin{bmatrix} 1 - x_1^4 - x_2^4 & x_2^4 - x_1^4 \\ x_2^4 - x_1^4 & 1 - x_1^4 - x_2^4 \end{bmatrix}.$$

We applied Algorithm 3.9 with $k_0 = 4$. It took about 0.58 seconds and stopped at Step 1, because (3.16) is infeasible. This means that the problem has no real solutions. We also solved it by the scalar-type method with $k_0 = 4$. It took about 0.44 seconds and stopped at Step 1 because (3.15) is infeasible.

Example 4.4. Consider the $PSDCP(Q_1, Q_2)$ with

$$Q_1(x) = \begin{bmatrix} 2x_2 - x_1^2 - 1 & x_1x_2 - 1 & x_2 - 1 \\ x_1x_2 - 1 & 2x_1 - x_2^2 & 1 - x_3^2 \\ x_2 - 1 & 1 - x_3^2 & 1 - x_3^2 \end{bmatrix}$$

and

$$Q_2(x) = \begin{bmatrix} x_1 + x_2x_3 + 3 & x_1 - 1 & x_1 + 2x_2 - 3 \\ x_1 - 1 & x_1x_3 - x_2^2x_3 & x_1^2x_2 + x_2 - 2 \\ x_1 + 2x_2 - 3 & x_1^2x_2 + x_2 - 2 & x_1^2 - 2x_3 \end{bmatrix}.$$

We solved it by Algorithm 3.9 with $k_0 = 3$. It took about 1.53 seconds for Algorithm 3.9 to find two real solutions $(1.0000, 1.0000, 0.5000)^T$ and $(1.0000, 1.0000, -1.0000)^T$. We also solved it by the scalar-type method with $k_0 = 4$. It took 4.36 seconds to find all real solutions.

Example 4.5. Consider the $PSDCP(Q_1, Q_2)$ with

$$Q_1(x) = \begin{bmatrix} 1 - 4x_1x_2x_3 & x_1 + x_3 - 1 & x_1x_2 - x_3 + 1 & 4 - x_2^2x_3 \\ x_1 + x_3 - 1 & 4 - x_1^2x_3 - x_2^2 & 4x_1 & 0 \\ x_1x_2 - x_3 + 1 & 4x_1 + x_1^2 & x_3 + x_1^2 & x_1x_2 \\ 4 - x_2^2x_3 & 0 & x_1x_2 & 5x_1x_2 \end{bmatrix}$$

and

$$Q_2(x) = \begin{bmatrix} 4x_3 - x_2^2 & x_1x_2 & x_1x_3 & 0 \\ x_1x_2 & x_2^2 + x_3 - 1 & 4x_1x_2 & x_1 \\ x_1x_3 & 4x_1x_2 & x_1x_3 + x_1x_2 & x_1^2 \\ 0 & x_1 & x_1^2 & x_1^2 + x_2^2 \end{bmatrix}.$$

We solved it by Algorithm 3.9 with $k_0 = 3$. It took 6.49 seconds for Algorithm 3.9 to find two real solutions $(0.0000, 2.0000, 1.0000)^T$ and $(0.0000, -2.0000, 1.0000)^T$. We also solved it by the scalar-type method with $k_0 = 6$. Unfortunately, it ran into numerical problems and no real solutions are obtained.

Almost all of the numerical results show that the scalar-type method require exponentially many principal minors typically in its representation and have much higher degrees. This often costs much more time to solve the PSDCP compared with the matrix-type methods in practice. Especially when the size of the matrix is big (even for the case $m = 4$), the scalar-type methods runs into numerical problems.

5 Conclusions

In this paper, we studied the polynomial semidefinite complementarity problem (PSDCP), which has not only the scalar polynomial constraints but also the polynomial matrix inequality (PMI) constraints. We formulate it equivalently as a polynomial optimization problem, where $h(x) = 0$ in (3.2) is used instead of $\mathrm{trace}(Q_1(x)Q_2(x)) = 0$, so that tighter relaxation problems can be obtained when using Lasserre's relaxation method. The solutions of the problem can be computed sequentially, if there are finitely many ones. The formulated polynomial optimizations are solved by Lasserre's hierarchy of matrix-type semidefinite relaxations.

Note that the optimization problems of the PSDCP are special cases of the following polynomial optimization problem with PMI constraints:

$$\left\{ \begin{array}{rl} \min & f(x) \\ \text{s.t.} & h_i(x) = 0, \forall i \in S_1, \\ & g_j(x) \geq 0, \forall j \in S_2, \\ & Q_k(x) \succeq 0, \forall k \in S_3, \end{array} \right.$$

where $f(x), h_i(x), g_j(x)$ are scalar polynomials, $Q_k(x)$ are symmetric polynomial matrices, S_1, S_2, S_3 are finite index set. Actually, the method proposed in this paper can be extended to solve the above general problem and the related ones.

References

[1] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[2] X. Bai, Z. Huang and Y. Wang, Global uniqueness and solvability for tensor complementarity problems, *J. Optim. Theory Appl.* 170 (2016) 72–84.

[3] X. Bai, Z. Huang and X. Li, Stability of solutions and continuity of solution maps of tensor complementarity problems, Asia-Pac. J. Oper. Res. 36(02) (2019): 1940002.

[4] J. Bochnak, M. Coste and M.F. Roy, *Real Algebraic Geometry*, Springer, New York, 1998.

[5] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Philadelphia, PA: SIAM, 1994.

[6] R. Curto and L. Fialkow, Truncated K-moment problems in several variables, *J. Operator Theory* (2005) 189–226.

[7] J. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[8] K. Goh, M. Safonov and J. Ly, Robust synthesis via bilinear matrix inequality, *Int. J. Robust Nonlinear Control* 6 (1996) 1079–1095.

[9] G. Golub and C. Van Loan, *Matrix Computations*, Third edition, The Johns Hopkins University Press, 1996.

[10] M. Gowda, Polynomial complementarity problems, *Pac. J. Optim.* 13 (2017) 227–241.

[11] J. Helton and J. Nie, A semidefinite approach for truncated K-moment problems, *Found. Comput. Math.* 12 (2012) 851–881.

[12] D. Henrion, D. Peaucelle, D. Arzelier and M. Sebek, Ellipsoidal approximation of the stability domain of a polynomial, *IEEE Trans. Automat. Control* 48 (2003) 2255–2259.

[13] D. Henrion and J.B. Lasserre, *Detecting global optimality and extracting solutions in GloptiPoly*, Positive polynomials in control, Lecture Notes in Control and Inform. Sci. Springer, Berlin, 312 (2005) 293–310.

[14] D. Henrion and J.B. Lasserre, Convergent relaxations of polynomial matrix inequalities and static output feedback, *IEEE Trans. Automat. Control* 51 (2006) 192–202.

[15] D. Henrion and J.B. Lasserre, Inner approximations for polynomial matrix inequalities and robust stability regions, *IEEE Trans. Automat. Control* 57 (2012) 1456–1467.

[16] D. Henrion, J.B. Lasserre and J. Löfberg, GloptiPoly 3: moments, optimization and semidefinite programming, *Optim. Methods Softw.* 24 (2009) 761–779.

[17] Z. Huang and L. Qi, Formulating an n-person noncooperative game as a tensor complementarity problem, *Comput. Optim. Appl.* 66 (2017) 557–576.

[18] Z. Huang and L. Qi, Tensor complementarity problems-part III: applications, *J. Optim. Theory Appl.* 183 (2019) 771–791.

[19] C. Hol and C. Scherer, Sum of squares relaxations for polynomial semi-definite programming, in: *Proc. Symp. Mathematical Theory of Networks and Systems (MTNS)*, Leuven, Belgium, 2004.

[20] C. Hol and C. Scherer, A sum-of-squares approach to fixed-order H_∞ synthesis, in: *Positive Polynomials in Control*, D. Henrion and A. Garulli, Eds. Berlin, Germany: Springer-Verlag, 2005.

[21] A. Karimi, H. Khatibi and R. Longchamp, Robust control of polytopic systems by convex optimization, *Automatica* 43 (2007) 1395–1402.

[22] I. Klep and M. Schweighofer, Pure states, positive matrix polynomials and sums of hermitian squares, *Indiana Univ. Math. J.* 59 (2010) 857–874.

[23] M. Kojima, *Sums of squares relaxations of polynomial semidefinite programs*, Research Re-port B-397, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, 2003

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

[25] J.B. Lasserre, *Moments, Positive Polynomials and Their Applications*, Imperial College Press, London, 2009.

[26] J.B. Lasserre, M. Laurent and P. Rostalski, Semidefinite characterization and computation of zero-dimensional real radical ideals, *Found. Comput. Math.* 8 (2008) 607–647.

[27] L. Ling, C. Ling and H. He, Properties of the solution set of generalized polynomial complementarity problems, *Pac. J. Optim.* 16 (2020) 155–174.

[28] J. Löfberg, *YALMIP : A toolbox for modeling and optimization in MATLAB*, Proceedings of the CACSD Conference, 2004.

[29] J. Nie and K. Ranestad, Algebraic degree of polynomial optimization, *SIAM J. Optim.* 20 (2009) 485–502.

[30] J. Nie, Polynomial matrix inequality and semidefinite representation, *Math. Oper. Res.* 36 (2011) 398–415.

[31] J. Nie, An exact Jacobian SDP relaxation for polynomial optimization, *Math. Program.* 137 (2013) 225–255.

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

[33] J. Nie, Polynomial optimization with real varieties, *SIAM J. Optim.* 23 (2013), 1634–1646.

[34] J. Nie, The hierarchy of local minimums in polynomial optimization, *Math. Program.* 151 (2015) 555–583.

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

[36] L. Qi, D. Sun and G. Zhou, A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities, *Math. Program.* 87 (2000) 1–35.

[37] R. Skelton, T. Iwasaki and K. Grigoriadis, *A Unified Algebraic Approach to Linear Control Design*, New York: Taylor and Francis, 1997.

[38] J. Sturm, Using SeDuMi 1.02: A MATLAB toolbox for optimization over symmetric cones, *Optim. Methods Softw.* 11 (1999) 625–653.

[39] X. Zhao and J. Fan, A semidefinite method for tensor complementarity problems, *Optim. Methods Softw.* 34 (2019) 758–769.

[40] M. Zheng, Z. Huang and X. Ma, Nonemptiness and compactness of solution sets to generalized polynomial complementarity problems, *J. Optim. Theory Appl.* 185 (2020) 80–98.

Manuscript received 7 February 2023
revised 28 August 2023
accepted for publication 26 November 2023

Xin Zhao
College of Science, Donghua University
Shanghai 201620, P.R. China
Present address of Xin Zhao
Shanghai Artificial Intelligence Networks and
Systems Engineering Technology Research Center Inc.
Shanghai 201203, P.R. China
E-mail address: zhaoxin922@126.com

Anwa Zhou
Department of Mathematics, Shanghai University
Shanghai 200444, P.R. China
Newtouch Center for Mathematics of Shanghai University
Shanghai 200444, P.R. China
E-mail address: zhouanwa@shu.edu.cn

Jinyan Fan
School of Mathematical Sciences, and MOE-LSC
Shanghai Jiao Tong University
Shanghai 200240, P.R. China
E-mail address: jyfan@sjtu.edu.cn