Solving polynomial variational inequality problems via Lagrange multiplier expressions and Moment-SOS relaxations

Jiawang Nie · Defeng Sun · Xindong Tang · Min Zhang

Abstract In this paper, we study variational inequality problems (VIPs) with involved mappings and feasible sets characterized by polynomial functions (namely, polynomial VIPs). We propose a numerical algorithm for computing solutions to polynomial VIPs based on Lagrange multiplier expressions and the Moment-SOS hierarchy of semidefinite relaxations. We also extend our approach to finding more or even all solutions to polynomial VIPs. We show that the method proposed in this paper can find solutions or detect the nonexistence of solutions within finitely many steps, under some general assumptions. In addition, we show that if the VIP is given by generic polynomials, then it has finitely many Karush-Kuhn-Tucker points, and our method can solve it within finitely many steps. Numerical experiments are conducted to illustrate the efficiency of the proposed methods.

1 Introduction

Let $\mathbb{R}^n$ be the n-dimensional Euclidean space and $X \subseteq \mathbb{R}^n$ be a nonempty closed set. For a tuple of functions $F(x) := (F_1(x), \ldots, F_n(x))$ where every $F_i : X \to \mathbb{R}$, the *variational inequality problem* (VIP) is

$$\text{Finding} \quad x \in X \quad \text{such that} \quad (y - x)^T F(x) \geq 0 \quad \text{for all} \quad y \in X. \tag{1.1}$$

Jiawang Nie

Department of Mathematics, University of California San Diego, 9500 Gilman Drive, La Jolla, CA, USA, 92093.
E-mail: njw@math.ucsd.edu

Defeng Sun

Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.
E-mail: defeng.sun@polyu.edu.hk

Xindong Tang

Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.
E-mail: xindong.tang@polyu.edu.hk

Min Zhang

Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.
E-mail: min-opt.zhang@polyu.edu.hk

Denote the problem (1.1) by $\mathrm{VI}(X, F)$ and its solution set by $\mathrm{SOL}(X, F)$. We call the $\mathrm{VI}(X, F)$ a *polynomial VIP* if $F_1, \ldots, F_n$ are polynomial functions in x and X is characterized by

$$X = \{x \in \mathbb{R}^n : g_i(x) \geq 0 \, (i \in \mathcal{I}), \ g_i(x) = 0 \, (i \in \mathcal{E})\}, \tag{1.2}$$

where $\mathcal{E}$ and $\mathcal{I}$ are two disjoint index sets and every $g_i \, (i \in \mathcal{E} \cup \mathcal{I})$ is a polynomial in x.

Variational inequalities were first proposed in the 1960s and have been in the central position of the optimization field since then. As a uniform approach to handling optimization and equilibrium problems, variational inequalities have various applications in decision-making, economy, finance, engineering, and other fields. We refer to the monograph [6] for a general introduction to VIPs.

Methods for solving VIPs have been extensively investigated in the literature (see [6,19] and references therein). Most approaches are designed for handling the $\mathrm{VI}(X, F)$ under the typical assumption that the set X is convex. For such cases, it is well known that the $\mathrm{VI}(X, F)$ can be equivalently reformulated into the following equation:

$$F^{\mathrm{nat}}(x) := x - \Pi_X(x - F(x)) = 0, \tag{1.3}$$

where $\Pi_X(\cdot)$ means the projection onto X. This reformulation leads to the popular projection-type methods, such as the basic fixed-point method [51], the extragradient method [20], and the hyperplane projection method [18]. Usually, the pseudomonotonicity of map F is required to guarantee the convergence of the projection-type methods. Besides (1.3), another equivalent equation reformulation of the $\mathrm{VI}(X, F)$ is as follows [48]:

$$F^{\mathrm{nor}}(z) := F(\Pi_X(z)) + z - \Pi_X(z) = 0. \tag{1.4}$$

With a change of variable, if $x^* \in \mathrm{SOL}(X, F)$, then $z^* := x^* - F(x^*)$ is a solution to (1.4), and conversely, if $z^* \in X$ solves (1.4), then $x^* := \Pi_X(z^*)$ is a solution to the $\mathrm{VI}(X, F)$. Numerical approaches such as the nonsmooth Newton method [44,47], the interior point method [17,53], and the smoothing Newton method [3,46], were developed for solving the $\mathrm{VI}(X, F)$ via dealing with the equivalent nonsmooth equation (1.3) or (1.4).

However, when the set X is nonconvex, the projection map Π_X may not be single-valued (e.g., consider the projection from the origin to a ring $\{x \in \mathbb{R}^n : 1 \leq \|x\|^2 \leq 2\}$), which leads the nonsmooth equations (1.3) and (1.4) to become general inclusion problems. To the best of our knowledge, no well-developed algorithm is applicable to solve general inclusion problems efficiently, especially when F lacks the monotonicity-like property. Fortunately, if the set X is finitely representable as a system of equalities and inequalities, then the $\mathrm{VI}(X, F)$ associates with the Karush-Kuhn-Tucker (KKT) system (see details in Section 2) under certain constraint qualification conditions (CQs). The KKT system can be viewed as a mixed complementarity problem (MiCP), thus methods for solving MiCPs can be applied; see [5,6,29]. Every solution of the KKT system solves the $\mathrm{VI}(X, F)$ when X is convex. However, this is not true under the nonconvex setting. Indeed, when X is nonconvex, solutions to the KKT system only provide candidate solutions to the $\mathrm{VI}(X, F)$. Furthermore, when X is bounded and convex, the existence of solutions to the $\mathrm{VI}(X, F)$ is guaranteed. However, without boundedness or convexity of X, the $\mathrm{SOL}(X, F)$ may be empty, and there is no well-known result for the existence of solutions. In fact, when X is nonconvex or unbounded, how to detect the nonexistence of solutions is still open, to the best of the authors' knowledge.

Recently, some studies on the theoretical properties of polynomial VIPs and their variations have been developed. Hieu [14] studied the solution set of the polynomial VIP with a convex feasible set. Tensor complementarity problems and tensor variational inequality problems were considered in [1,52]. Further, weakly homogeneous variational inequality problems, as extensions

of polynomial VIPs and tensor VIPs, were studied in [9,27]. Moreover, the Moment-SOS hierarchy of semidefinite relaxations was developed for solving polynomial optimization problems. It can find global minimizers for nonconvex polynomial optimization problems with theoretical guarantees under mild conditions. It was also applied to solve many optimization-related problems given by polynomials, such as bilevel optimization [15,31,32], saddle point problems and generalized Nash equilibrium problems (GNEPs) [40,41,42], tensor (eigenvalue) complementarity problems [7,54], etc. We refer to [22,23,25,26,37] for more references about polynomial optimization.

In this paper, we focus on numerical methods for solving polynomial VIPs without assuming the monotonicity of the map F or the convexity of the set X. Our main contributions are listed below.

- We propose an algorithm for solving polynomial VIPs using Lagrange multiplier expressions and the Moment-SOS relaxations. Under some general assumptions, the proposed algorithm can find a solution to the VIP or detect the nonexistence of solutions in finitely many steps. Particularly, this computational goal can be achieved in the initial loop when X is convex.
- Based on the proposed algorithm for finding a solution to a polynomial VIP, we further investigate how to find more (even all, if there are finitely many) solutions.
- We show that if the VIP is given by generic polynomial functions, then there are finitely many KKT points, and all solutions to the polynomial VIP are KKT points. In this case, our method is guaranteed to terminate within finitely many loops. Moreover, we give an upper bound for the number of KKT points using the intersection theory.
- Even when the algorithms do not have finite convergence, we show the asymptotic convergence under certain continuity assumptions. Numerical experiments are made and presented to show the efficiency of our methods.

This paper is organized as follows. Some preliminaries about VIPs and the polynomial optimization problems are given in Section 2. In Section 3, the algorithm for finding a solution to the polynomial VIP is developed. Based on the algorithm, we further introduce methods for finding more solutions to the polynomial VIP. Section 4 studies the Moment-SOS hierarchy of semidefinite relaxations. Numerical experiments are presented in Section 5. In the appendix, we show the finiteness of KKT points for solution sets of VIPs given by generic polynomials and study the algebraic degree of polynomial VIPs.

2 Preliminaries

Notation The symbol $\mathbb{N}$ (resp., $\mathbb{R}$, $\mathbb{C}$) stands for the set of nonnegative integers (resp., real numbers, complex numbers). For a positive integer k, denote the set $[k] := \{1, \ldots, k\}$. For a real number t, the $\lceil t \rceil$ represents the smallest integer not smaller than t. We use e_i to denote the vector such that the ith entry is 1 and all others are zeros. By writing $A \succeq 0$ (resp., $A \succ 0$), we mean that the matrix A is symmetric positive semidefinite (resp., positive definite).

Let $\mathbb{R}[x]$ denote the ring of polynomials with real coefficients in $x \in \mathbb{R}^n$, and $\mathbb{R}[x]_d$ denote its subset of polynomials whose degrees are not greater than d. The $\mathbb{C}[x]$ and $\mathbb{C}[x]_d$ are defined similarly. For a polynomial $p \in \mathbb{R}[x]$, the $p = 0$ means $p(x)$ is identically zero on $\mathbb{R}^n$. We say the polynomial p is nonzero if $p \neq 0$. Let $\alpha := (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, and we denote

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

For an integer $d > 0$, denote the monomial power set $\mathbb{N}_d^n := \{\alpha \in \mathbb{N}^n : |\alpha| \leq d\}$. We use $[x]_d$ to denote the vector of all monomials in x in the graded alphabetical order whose degree is at

most d. For instance, if $x = (x_1, x_2)$, then

$$[x]_3 = \left[1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, x_1^2 x_2, x_1 x_2^2, x_2^3\right]^T.$$

Throughout the paper, a property is said to hold *generically* if it holds for all points in the space of input data except for a set of Lebesgue measure zero.

2.1 Ideals and positive polynomials

Let $\mathbb{F} := \mathbb{R}$ or $\mathbb{C}$. For a polynomial $p \in \mathbb{F}[x]$ and subsets $I, J \subseteq \mathbb{F}[x]$, we define

$$p \cdot I := \{pq : q \in I\}, \quad I + J := \{a + b : a \in I, b \in J\}.$$

The subset I is an ideal if $p \cdot I \subseteq I$ for all $p \in \mathbb{F}[x]$ and $I + I \subseteq I$. For a tuple of polynomials $h = (h_1, \ldots, h_s)$, the set

$$\text{Ideal}[h] := h_1 \cdot \mathbb{F}[x] + \cdots + h_s \cdot \mathbb{F}[x]$$

is the ideal generated by h, which is the smallest ideal containing every h_i.

We review basic concepts in polynomial optimization. A polynomial $\sigma \in \mathbb{R}[x]$ is said to be a sum of squares (SOS) if $\sigma = p_1^2 + \cdots + p_k^2$ for polynomials $p_i \in \mathbb{R}[x]$ $(i \in [k])$. The set of all SOS polynomials in x is denoted as $\Sigma[x]$. For a degree d, we denote the truncation

$$\Sigma[x]_d := \Sigma[x] \cap \mathbb{R}[x]_d.$$

For a tuple $g = (g_1, \ldots, g_t)$ of polynomials in x, its quadratic module is the set

$$\text{Qmod}[g] := \Sigma[x] + g_1 \cdot \Sigma[x] + \cdots + g_t \cdot \Sigma[x].$$

Similarly, we denote the truncation of $\text{Qmod}[g]$

$$\text{Qmod}[g]_{2d} := \Sigma[x]_{2d} + g_1 \cdot \Sigma[x]_{2d-\deg(g_1)} + \cdots + g_t \cdot \Sigma[x]_{2d-\deg(g_t)},$$

where $\deg(g_i)$ means the degree of g_i. The tuple g determines the basic closed semi-algebraic set

$$\mathcal{S}(g) := \{x \in \mathbb{R}^n : g_1(x) \geq 0, \ldots, g_t(x) \geq 0\}. \tag{2.1}$$

For a tuple $h = (h_1, \ldots, h_s)$ of polynomials in $\mathbb{R}[x]$, its real zero set is

$$\mathcal{Z}(h) := \{x \in \mathbb{R}^n : h_1(x) = \cdots = h_s(x) = 0\}.$$

The set $\text{Ideal}[h] + \text{Qmod}[g]$ is said to be *archimedean* if there exists $\rho \in \text{Ideal}[h] + \text{Qmod}[g]$ such that the set $\mathcal{S}(\rho)$ is compact. If $\text{Ideal}[h] + \text{Qmod}[g]$ is archimedean, then $\mathcal{Z}(h) \cap \mathcal{S}(g)$ must be compact. Conversely, if $\mathcal{Z}(h) \cap \mathcal{S}(g)$ is compact, say, $\mathcal{Z}(h) \cap \mathcal{S}(g)$ is contained in the ball $R - \|x\|^2 \geq 0$, then $\text{Ideal}[h] + \text{Qmod}[g, R - \|x\|^2]$ is archimedean and $\mathcal{Z}(h) \cap \mathcal{S}(g) = \mathcal{Z}(h) \cap \mathcal{S}(g, R - \|x\|^2)$. Clearly, if $f \in \text{Ideal}[h] + \text{Qmod}[g]$, then $f \geq 0$ on $\mathcal{Z}(h) \cap \mathcal{S}(g)$. The reverse is not necessarily true. However, when $\text{Ideal}[h] + \text{Qmod}[g]$ is archimedean, if $f > 0$ on $\mathcal{Z}(h) \cap \mathcal{S}(g)$, then $f \in \text{Ideal}[h] + \text{Qmod}[g]$. This conclusion is referenced as Putinar's Positivstellensatz [45]. Interestingly, if $f \geq 0$ on $\mathcal{Z}(h) \cap \mathcal{S}(g)$, we also have $f \in \text{Ideal}[h] + \text{Qmod}[g]$, under some standard optimality conditions [35].

2.2 Localizing and moment matrices

Let $\mathbb{R}^{\mathbb{N}^n_{2d}}$ denote the space of all real vectors that are labeled by $\alpha \in \mathbb{N}^n_{2d}$. A vector $y \in \mathbb{R}^{\mathbb{N}^n_{2d}}$ is labeled as $y = (y_\alpha)_{\alpha \in \mathbb{N}^n_{2d}}$. Such y is called a *truncated multi-sequence* (tms) of degree $2d$. For a polynomial $f = \sum_{\alpha \in \mathbb{N}^n_{2d}} f_\alpha x^\alpha \in \mathbb{R}[x]_{2d}$, define the operation

$$\langle f, y \rangle := \sum_{\alpha \in \mathbb{N}^n_{2d}} f_\alpha y_\alpha. \tag{2.2}$$

The operation $\langle f, y \rangle$ is a bilinear function in (f, y). For a polynomial $q \in \mathbb{R}[x]$ with $\deg(q) \leq 2d$, and the integer $t = d - \lceil \deg(q)/2 \rceil$, the outer product $q \cdot [x]_t([x]_t)^T$ is a symmetric matrix polynomial in x, with length $\binom{n+t}{t}$. We write the expansion as

$$q \cdot [x]_t([x]_t)^T = \sum_{\alpha \in \mathbb{N}^n_{2d}} x^\alpha Q_\alpha,$$

for some symmetric matrices Q_α. Then we define the matrix function

$$L_q^{(d)}[y] := \sum_{\alpha \in \mathbb{N}^n_{2d}} y_\alpha Q_\alpha. \tag{2.3}$$

It is called the dth *localizing matrix* of q generated by y. For a given q, the matrix $L_q^{(d)}[y]$ is linear in y. Localizing and moment matrices are important for getting semidefinite relaxations of polynomial optimization problems [21, 33, 34]. We refer to [22, 23, 25, 26, 37] for more references about polynomial optimization and moment problems.

2.3 The KKT conditions and Lagrange multiplier expressions

Let $x \in \mathrm{SOL}\,(X, F)$ with the set X given by (1.2). Suppose $\mathcal{E} \cup \mathcal{I} = [m]$. Under the linear independence constraint qualification condition (LICQ, see [2]), there exists a Lagrange multiplier vector $\lambda = (\lambda_1, \ldots, \lambda_m)$ such that

$$\begin{cases} F(x) = \sum_{i=1}^m \lambda_i \nabla g_i(x), \ g_i(x) = 0 \, (i \in \mathcal{E}), \\ \lambda_i \geq 0, \ g_i(x) \geq 0, \ \lambda_i \cdot g_i(x) = 0 \, (i \in \mathcal{I}). \end{cases} \tag{2.4}$$

This is called the *KKT conditions* for the VI (X, F). A solution (x, λ) to (2.4) is called a *KKT pair*, and x is called a *KKT point*. Particularly, when the set X is convex, x solves the VI (X, F) if and only if it is a KKT point. For the given VI (X, F), every KKT pair (x, λ) satisfies

$$\underbrace{\begin{bmatrix} \nabla g_1(x) & \nabla g_2(x) & \cdots & \nabla g_m(x) \\ g_1(x) & 0 & \cdots & 0 \\ 0 & g_2(x) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_m(x) \end{bmatrix}}_{G(x)} \underbrace{\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_m \end{bmatrix}}_{\lambda} = \underbrace{\begin{bmatrix} F(x) \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{\hat{F}(x)}. \tag{2.5}$$

We say the polynomial tuple $g = (g_1, \ldots, g_m)$ is nonsingular if the matrix of polynomials $G(x)$ has full column rank for all $x \in \mathbb{C}^n$. By [39, Proposition 5.1], the tuple g is nonsingular if and only if there exists a matrix of real coefficient polynomials $L(x)$ such that

$$L(x)G(x) = I_m. \tag{2.6}$$

When constraining polynomials $g_1, \ldots, g_m$ are generic, $\operatorname{rank} G(x) = m$ for all $x \in \mathbb{C}^n$ [39, Proposition 5.7], and there exists an $L(x)$ satisfying (2.6). In this case, the following identity holds at all KKT pairs (x, λ):

$$\lambda = L(x) \cdot \hat{F}(x). \tag{2.7}$$

Denote the ith component of $L(x) \cdot \hat{F}(x)$ by $\lambda_i(x)$, and let $\lambda(x) := (\lambda_1(x), \ldots, \lambda_m(x))$. Then $\lambda(x)$ is a tuple of real polynomials in x and $\lambda_i = \lambda_i(x)$ for all $i \in [m]$ at all KKT pairs (x, λ). The $\lambda(x)$ is called a *polynomial Lagrange multiplier expression* (LME) for the VI (X, F).

For the polynomial VI (X, F), suppose the LME $\lambda(x)$ exists. Then the KKT system (2.4) can be equivalently written as

$$\begin{cases} F(x) = \sum_{i=1}^m \lambda_i(x) \nabla g_i(x), \\ g_i(x) = 0 \, (i \in \mathcal{E}), \\ \lambda_i(x) \geq 0, g_i(x) \geq 0, \lambda_i(x) \cdot g_i(x) = 0 \, (i \in \mathcal{I}). \end{cases} \tag{2.8}$$

This is a system of polynomial equalities and inequalities in x. If we denote by $\mathcal{K}$ the set of points satisfying (2.8), then SOL $(X, F) \subseteq \mathcal{K}$. For the convenience of our discussion, we denote

$$E := \left\{ F(x) - \sum_{i=1}^m \lambda_i(x) \nabla g_i(x) \right\} \cup \{\lambda_i(x) \cdot g_i(x) \mid i \in \mathcal{I}\} \cup \{g_i(x) \mid i \in \mathcal{E}\},$$

$$I := \{\lambda_i(x) \mid i \in \mathcal{I}\} \cup \{g_i(x) \mid i \in \mathcal{I}\}.$$

Then, $\mathcal{K} = \mathcal{Z}(E) \cap \mathcal{S}(I)$.

3 Finding solutions for polynomial VIPs

In this section, we first propose an algorithm for finding a solution to the polynomial VI (X, F) by solving a sequence of polynomial optimization problems. Then, based on the proposed algorithm, we provide a method to find more solutions to the VI (X, F). Throughout this section, we assume the VI (X, F) is given by polynomials and the constraining tuple $g = (g_1, \ldots, g_m)$ is nonsingular, thus LMEs exist.

3.1 Finding one solution for the polynomial VIP

For the given polynomial VI (X, F), let Θ be a generic positive definite matrix. Consider the following polynomial optimization problem ($[x]_1 := (1, x_1, x_2, \ldots, x_n)$)

$$\begin{cases} \min_{x \in \mathbb{R}^n} & [x]_1^T \Theta [x]_1, \\ s.t. & x \in \mathcal{K}. \end{cases} \tag{3.1}$$

For the VI (X, F) with a convex feasible set X, by solving (3.1), one gets a solution to the VI (X, F) if (3.1) has a nonempty feasible set, or detect the emptiness of SOL (X, F) otherwise. This is shown in the following proposition.

Proposition 3.1 *Consider the polynomial $VI(X, F)$, where X is a convex set that is given by (1.2) with a nonsingular polynomial tuple g. If the problem (3.1) is infeasible, then $VI(X, F)$ does not have any solution. Otherwise, the solution set of (3.1) is nonempty, and every minimizer solves the $VI(X, F)$.*

Proof. By [36, Proposition 5.1], the nonsingularity of the polynomial tuple g yields the existence of the LMEs, which also infers that the LICQ holds. Thus $\mathrm{SOL}\,(X, F) \subseteq \mathcal{K}$, and the infeasibility of the problem (3.1) implies the emptiness of $\mathrm{SOL}\,(X, F)$.

Suppose the problem (3.1) is feasible, i.e., $\mathcal{K} \neq \emptyset$. Since Θ is a generic positive definite matrix, the quadratic polynomial $[x]_1^T \Theta [x]_1$ is strongly convex, which implies the existence of minimizers for (3.1). In addition, under the nonsingularity condition of g and the convexity of X, we have $\mathrm{SOL}\,(X, F) = \mathcal{K}$, thus the minimizer of (3.1) solves $\mathrm{VI}\,(X, F)$. $\square$

However, when the set X is nonconvex, a minimizer of (3.1) may not solve the $\mathrm{VI}\,(X, F)$. Consider the gap function

$$\nu(x) := \inf_{y \in X} (y - x)^T F(x).$$

It is clear that $\nu(u) \leq 0$ for all $u \in X$, and $u \in \mathrm{SOL}\,(X, F)$ if and only if $\nu(u) = 0$. If $u \notin \mathrm{SOL}\,(X, F)$, the gap function can be used to construct an extra constraint to shrink the feasible set of (3.1). That is, for a minimizer u of (3.1) such that $u \notin \mathrm{SOL}\,(X, F)$, we consider

$$\begin{cases} \varepsilon := \min_{y \in \mathbb{R}^n} & (y - u)^T F(u) \\ s.t. & y \in X. \end{cases} \tag{3.2}$$

To our main interest, assume (3.2) has a minimizer v (otherwise, we may add some extra boundedness constraints, see Section 4.2). Let $\mathcal{K}' := \mathcal{K} \cap \{x \in \mathbb{R}^n : (v - x)^T F(x) \geq 0\}$, then we have $\mathrm{SOL}\,(X, F) \subseteq \mathcal{K}'$ and $u \notin \mathcal{K}'$. Therefore, if we solve (3.1) with a replacement of $\mathcal{K}$ by the shrinked feasible set $\mathcal{K}'$, then another candidate solution to the $\mathrm{VI}\,(X, F)$ will be obtained. Indeed, we can repeat this procedure to remove KKT points that are not solutions, and get tighter approximations to the solution set $\mathrm{SOL}\,(X, F)$, until an actual solution is obtained or the nonexistence of solutions is detected. Based on such a scheme, we propose the following algorithm for finding a solution to the $\mathrm{VI}\,(X, F)$:

Algorithm 3.2 For $\mathrm{VI}\,(X, F)$ such that X is given by (1.2) with nonsingular g, do the following:

Step 0 Choose a generic positive definite matrix Θ of length $n + 1$. Set $k := 1$ and $V := \emptyset$.
Step 1 Solve the following polynomial optimization problem

$$\begin{cases} \min_{x \in \mathbb{R}^n} & [x]_1^T \Theta [x]_1 \\ s.t. & x \in \mathcal{K}, \\ & (v - x)^T F(x) \geq 0 \ (v \in V). \end{cases} \tag{3.3}$$

If (3.3) is infeasible, then $\mathrm{SOL}\,(X, F)$ is empty and stop; otherwise, solve (3.3) for a minimizer $u^{(k)}$.
Step 2 Solve the polynomial optimization problem (3.2) with $u := u^{(k)}$ for a set of minimizers $T^{(k)}$.
Step 3 If $\varepsilon = 0$, then $u^{(k)} \in \mathrm{SOL}\,(X, F)$ and stop; otherwise, let $V := V \cup T^{(k)}$ and $k := k + 1$, then go back to Step 1.

Remark 3.3 (i) In Section 4, we will study how to solve the occurring polynomial optimization subproblems in Algorithm 3.2. (ii) In Step 1, all feasible points of (3.3) are KKT points. Thus, the objective function in (3.3) can be replaced by any polynomial function in x. However, using $[x]_1^T \Theta [x]_1$ gives us better convergence results for Algorithm 3.2, and makes (3.3) easier to solve; see Theorem 3.4 and Section 4.1. (iii) For (3.3), we may also replace the constraint $x \in \mathcal{K}$ by the KKT system (2.4), which means we treat all λ_i as new variables without using LMEs. However, this is not computationally efficient, as shown in Examples 5.1 and 5.2 in Section 5.1. (iv) We do not need the set $T^{(k)}$ in Step 2 to contain all minimizers of (3.2). Indeed, one minimizer is

enough to get another solution candidate. However, we can usually get a tighter approximation to SOL (X, F) if there are more points in $T^{(k)}$. When (3.2) has more than one minimizer, the *flat truncation* can be applied to find them; see Section 4 for more details.

We say Algorithm 3.2 terminates within finitely many steps if the algorithm finds a solution to VI (X, F) or detects the nonexistence of solutions at some iteration loop k. In the following, we establish the convergence property for Algorithm 3.2.

Theorem 3.4 *Consider the* $VI(X, F)$ *where* X *is given by (1.2) with a nonsingular polynomial tuple* g. *Let* $\mathfrak{s} := |\mathcal{K} \setminus SOL(X, F)|$.

  (i) *If* $SOL(X, F) = \emptyset$ *and* $\mathfrak{s} < \infty$, *then Algorithm 3.2 will detect nonexistence of solutions to the* $VI(X, F)$ *with* $k \leq \mathfrak{s} + 1$.
  (ii) *If* $SOL(X, F) \neq \emptyset$ *and* $\mathfrak{s} < \infty$, *then Algorithm 3.2 will find a solution to the* $VI(X, F)$ *with* $k \leq \mathfrak{s} + 1$.
  (iii) *Suppose* $\mathfrak{s} = \infty$ *and Algorithm 3.2 does not terminate within finitely many steps. If* u^* *is an accumulation point of* $\{u^{(k)}\}_{k=1}^{\infty}$, *the gap function* $\nu(u)$ *is continuous at* u^*, *and* $\limsup_{k \to \infty} T^{(k)}$ *is bounded, then* $u^* \in SOL(X, F)$.

Proof. We assume $\mathcal{K}$ is nonempty. Otherwise, the conclusion follows directly from the fact that $|\mathcal{K}| = 0$ implies the infeasibility for (3.1). Denote by $\mathcal{K}_k$ the feasible set of (3.3) at the kth iteration. Then, $\mathcal{K}_{k+1} \subseteq \mathcal{K}_k$ for all $k \in \mathbb{N}_+$. In addition, if Algorithm 3.2 does not terminate at the kth loop, then $u^{(k)} \notin \mathcal{K}_{k+1}$. Thus, we have the following strictly descent chain:

$$\mathcal{K} = \mathcal{K}_1 \supsetneq \mathcal{K}_2 \cdots \supsetneq \mathcal{K}_k \supsetneq \cdots \supsetneq SOL(X, F).$$

Since $\mathfrak{s} = |\mathcal{K} \setminus SOL(X, F)| < \infty$, there exists $k_1 \in \mathbb{N}_+$ such that $k_1 \leq \mathfrak{s} + 1$ and $\mathcal{K}_{k_1} = SOL(X, F)$.

(i) If $SOL(X, F) = \emptyset$, then problem (3.3) with $k = k_1$ is infeasible, which means the VI (X, F) does not have any solution. Therefore, Algorithm 3.2 can detect the nonexistence of solutions to VI (X, F) in at most k_1 loops.

(ii) If $SOL(X, F) \neq \emptyset$, then every feasible point of (3.3) with $k = k_1$ is a solution to VI (X, F). Therefore, Algorithm 3.2 can find a solution to VI (X, F) in at most k_1 loops.

(iii) Assume that Algorithm 3.2 does not terminate within finitely many steps, and there exists an accumulation point u^* of sequence $\{u^{(k)}\}_{k=1}^{\infty}$. Without loss of generality, we assume $u^{(k)} \to u^*$ as $k \to \infty$. Note that u^* is a solution to VI (X, F) if and only if $\nu(u^*) = 0$. For each $i = 1, 2, \ldots$, let $v^{(i)}$ be a minimizer of (3.2) for $u := u^{(i)}$. Since $u^{(i)} \notin SOL(X, F)$, we have

$$(v^{(i)} - u^{(i)})^T F(u^{(i)}) = \nu(u^{(i)}) < 0.$$

For each $j > i$, based on the feasibility of (3.3) at $u^{(j)}$, we have

$$(v^{(i)} - u^{(j)})^T F(u^{(j)}) \geq 0.$$

Due to the continuity of F, we have the $(v^{(i)} - x)^T F(x)$ is continuous in x for every given $v^{(i)}$. Hence,

$$(v^{(i)} - u^*)^T F(u^*) \geq 0 \quad \text{for all} \quad i \in \mathbb{N}_+.$$

Moreover, we have the following for all $i \in \mathbb{N}_+$.

$$\begin{aligned}
\nu(u^*) &= \nu(u^*) - \nu(u^{(i)}) + \nu(u^{(i)}) \\
&\geq \nu(u^*) - \nu(u^{(i)}) + (v^{(i)} - u^{(i)})^T F(u^{(i)}) - (v^{(i)} - u^*)^T F(u^*).
\end{aligned}$$

From the assumption that $\limsup_{k \to \infty} T^{(k)}$ is bounded, there exists a subsequence of $\{v^{(j)}\}_{j=1}^{\infty}$, say $\{v^{(k_j)}\}_{j=1}^{\infty}$, being convergent. Denote by v^* the limit point of $\{v^{(k_j)}\}_{j=1}^{\infty}$. Then, we have

$$\nu(u^*) \geq \nu(u^*) - \nu(u^{(k_j)}) + (v^{(k_j)} - u^{(k_j)})^T F(u^{(k_j)}) - (v^{(k_j)} - u^*)^T F(u^*) \tag{3.4}$$

hold for all $j \in \mathbb{N}_+$. Let $j \to \infty$, the right hand side of (3.4) equals to 0, because ν is continuous at u^*, and $(v-u)^T F(u)$ is continuous in variable of (u, v). So, we have that $\nu(u^*) \geq 0$. Besides, $\nu(u) \leq 0$ for all $u \in X$ based on the definition. Therefore, we conclude that $\nu(u^*) = 0$, which means $u^* \in \mathrm{SOL}\,(X, F)$. $\quad\square$

We remark that if the sequence $\{u^{(k)}\}_{k=1}^{\infty}$ generated by Algorithm 3.2 has no accumulation point, then $\mathrm{SOL}\,(X, F) = \emptyset$. To see this, if $\{u^{(k)}\}_{k=1}^{\infty}$ does not have any accumulation point, then it must be unbounded. Note that the objective function $\theta(x) := [x]_1^T \Theta [x]_1$ is a quadratic positive definite polynomial. The $\theta(u^{(k)})$ goes to ∞ as $k \to \infty$. Suppose $x^* \in \mathrm{SOL}\,(X, F)$. Then, we have $\infty > \theta(x^*) \geq \theta(u^{(1)})$, otherwise $u^{(1)}$ cannot be the minimizer for (3.1). Since $\lim_{k \to \infty} \theta(u^{(k)}) = \infty$, there exists $k_1 \in \mathbb{N}$ such that $\theta(u^{(k_1)}) \leq \theta(x^*) < \theta(u^{(k_1+1)})$. Because $x^* \in \mathrm{SOL}\,(X, F)$, x^* satisfies the KKT condition and for any $v \in X$ the following holds

$$(v - x^*)^T F(x^*) \geq 0.$$

It means that x^* is feasible for (3.3) at the $(k_1 + 1)$th loop, which contradicts to $u^{(k_1+1)}$ being a minimizer of (3.3). So such x^* does not exist and we conclude the emptiness of $\mathrm{SOL}\,(X, F)$ if the sequence $\{u^{(k)}\}_{k=1}^{\infty}$ has no accumulation point.

The continuity condition of $\nu(x)$ holds under certain conditions, for example, the *restricted inf-compactness* (RIC) condition; see [10, Definition 3.13]. However, to implement Algorithm 3.2 does not require any *a priori* knowledge on the finiteness of $\mathfrak{s}$ or the continuity of ν. Indeed, even if $\mathfrak{s} = \infty$, Algorithm 3.2 may still terminate in finitely many steps. In all numerical experiments in Section 5, our method finds solutions to the VIPs or detects the nonexistence of solutions in finitely many steps. To end this section, we demonstrate the following result saying that if the $\mathrm{VI}\,(X, F)$ is given by generic polynomials F and g, then $\mathcal{K}$ is a finite set, which guarantees the finite termination for Algorithm 3.2. The proof is given in the appendix.

Theorem 3.5 *Let $a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m \in \mathbb{N}_+$ be degrees. Suppose the $VI(X, F)$ is defined by polynomials such that $F_i(x)$ is a generic polynomial in $\mathbb{C}[x]_{a_i}$ for all $i \in [n]$, and $g_j(x)$ is a generic polynomial in $\mathbb{C}[x]_{b_j}$ for all $j \in [m]$. Then, every $x \in SOL\,(X, F)$ is a KKT point of the $VI(X, F)$, and we have $|\mathcal{K}| < \infty$.*

3.2 Finding more solutions to the polynomial VIP

In this subsection, we further investigate how to find more solutions or check the completeness of the computed solution set for the $\mathrm{VI}\,(X, F)$, based on Algorithm 3.2.

Suppose Algorithm 3.2 terminates at a solution x^*, and x^* is an isolated point in the feasible region at the last iteration of Algorithm 3.2. Since the matrix Θ in the objective function of (3.3) is generic and positive definite, by [36, Proposition 5.2], there exists $\delta > 0$ such that for all $x \in \mathrm{SOL}\,(X, F) \setminus \{x^*\}$,

$$[x]_1^T \Theta [x]_1 \geq [x^*]_1^T \Theta [x^*]_1 + \delta. \tag{3.5}$$

Therefore, if we add (3.5) as an extra constraint to (3.3), then we will get a new candidate solution for the $\mathrm{VI}\,(X, F)$ other than x^*. That is, we solve the following polynomial optimization problem:

$$\begin{cases} \min_{x \in \mathbb{R}^n} & [x]_1^T \Theta [x]_1 \\ s.t. & x \in \mathcal{K}_{x^*}, \end{cases} \tag{3.6}$$

where

$$\mathcal{K}_{x^*} := \{x \in \mathcal{K} \mid (v - x)^T F(x) \geq 0 \ (v \in V), \ [x]_1^T \Theta [x]_1 \geq [x^*]_1^T \Theta [x^*]_1 + \delta\}.$$

If (3.6) is infeasible, then $\mathrm{SOL}\,(X, F) = \{x^*\}$. Otherwise, (3.6) must have a minimizer u due to the positive definiteness of Θ, and we can check whether $u \in \mathrm{SOL}\,(X, F)$ or not by solving (3.2).

In practical implementation, one needs a suitable δ to formulate (3.6). Given x^* and $\mathcal{K}_{x^*}$, the following trick is useful for finding the δ. By [40, Proposition 3.5] (see also [37]), if x^* is isolated in $\mathcal{K}_{x^*}$, then there exists $\hat{\delta} > 0$ such that

$$\mathcal{K}_{x^*} \cap \{x \in \mathbb{R}^n : [x]_1^T \Theta [x]_1 \le [x^*]_1^T \Theta [x^*]_1 + \hat{\delta}\} = \{x^*\}.$$

Moreover, since $\mathrm{SOL}\,(X, F) \subseteq \mathcal{K}_{x^*}$, all δ satisfying $0 < \delta \le \hat{\delta}$ is available for (3.6). For *a priori* $\delta_0 > 0$, let $\delta = \delta_0$ and consider the following maximization problem

$$
\begin{cases}
\gamma^* := \max\limits_{x \in \mathbb{R}^n} & [x]_1^T \Theta [x]_1 \\
\quad\ s.t. & x \in \mathcal{K},\ (v - x)^T F(x) \ge 0\,(v \in V), \\
& [x]_1^T \Theta [x]_1 \le [x^*]_1^T \Theta [x^*]_1 + \delta.
\end{cases}
\tag{3.7}
$$

If $\gamma^* = [x^*]_1^T \Theta [x^*]_1$, then δ is applicable for (3.6). Otherwise, we update $\delta := \rho \cdot \delta$ for some $\rho < 1$ and solve (3.7) again. Repeating these steps, we may find more solutions to the $\mathrm{VI}\,(X, F)$.

Algorithm 3.6 Under the same assumptions as in Algorithm 3.2, suppose we have found a solution x^* to the $\mathrm{VI}\,(X, F)$. Do the following:

Step 0 Choose a $\delta_0 > 0$ and a positive $\rho < 1$. Let $\delta := \delta_0$.
Step 1 Solve the polynomial optimization (3.7) for the maximal value γ^*.
Step 2 If $\gamma^* = [x^*]_1^T \Theta [x^*]_1$, then proceed to the next step; otherwise, let $\delta := \delta \cdot \rho$ and go back to Step 1.
Step 3 Solve the polynomial optimization (3.6). If it is infeasible, then there does not exist any other solution for $\mathrm{VI}\,(X, F)$. Otherwise, solve (3.6) for a minimizer u.
Step 4 Solve the linear optimization problem (3.2) with u for a set of minimizers T.
Step 5 If $\varepsilon = 0$, then $u \in \mathrm{SOL}\,(X, F)$ is another solution and stop; otherwise, let $V := V \cup T$, then go back to Step 3.

When $x^* \in \mathrm{SOL}\,(X, F)$ is isolated in $\mathcal{K}_{x^*}$, we can always find a suitable δ for (3.3) by solving (3.7) for finitely many times. Therefore, if $|\mathcal{K} \setminus \mathrm{SOL}\,(X, F)|$ is finite, Algorithm 3.6 will surely produce a new solution within finitely many steps, by Theorem 3.4. Indeed, when Θ is generic and all points in $\mathrm{SOL}\,(X, F)$ are isolated, we have the following hierarchy of solutions

$$[x^{(1)}]_1^T \Theta [x^{(1)}]_1 < [x^{(2)}]_1^T \Theta [x^{(2)}]_1 < [x^{(3)}]_1^T \Theta [x^{(3)}]_1 < \cdots$$

$$\text{with}\quad \mathrm{SOL}\,(X, F) = \{x^{(1)}, x^{(2)}, x^{(3)}, \dots\}. \tag{3.8}$$

As listed in (3.8), the solution first obtained by Algorithm 3.2 is $x^{(1)}$, and Algorithm 3.6 finds $x^{(k)}$ with $x^* := x^{(k-1)}$. Particularly, when $|\mathrm{SOL}\,(X, F)| =: r < \infty$, such a scheme finds all solutions to $\mathrm{VI}\,(X, F)$, as long as $x^{(k)}$ is isolated in $\mathcal{K}_{x^{(k-1)}}$ for each $k < r$. Furthermore, if we apply Algorithm 3.6 with $x^* := x^{(r)}$, then the algorithm terminates at Step 3 because the feasible set $\mathcal{K}_{x^*}$ of (3.6) must be empty for some V, and we conclude that a complete solution set is obtained.

Theorem 3.7 *Consider the $VI(X, F)$ where X is given by (1.2) with a nonsingular polynomial tuple g. If $|\mathcal{K}| < \infty$, then all solutions to $VI(X, F)$ can be obtained by running Algorithm 3.6 finitely many times.*

It is clear that when the $\mathrm{VI}\,(X, F)$ is given by generic polynomials, we can find all solutions to the $\mathrm{VI}\,(X, F)$ because $|\mathcal{K}| < \infty$, by Theorems 3.5 and 3.7.

4 Solving the polynomial optimization subproblems

In this section, we study the method for handling polynomial optimization subproblems appearing in Algorithms 3.2 and 3.6. In general, we consider a polynomial optimization problem in the following form

$$\begin{cases} \vartheta := \min_{x \in \mathbb{R}^n} & \theta(x) \\ \quad s.t. & p(x) = 0 \, (p \in \Phi), \\ & q(x) \geq 0 \, (q \in \Psi), \end{cases} \tag{4.1}$$

where $\theta(x)$ is a polynomial function in x, and Φ, Ψ are tuples of polynomials in x.

We introduce the Moment-SOS hierarchy of semidefinite relaxations for (4.1). Denote

$$d_0 := \max\{\lceil \deg(p)/2 \rceil : p \in \Phi \cup \Psi \cup \{\theta\}\}. \tag{4.2}$$

For a degree $k \geq d_0$, the kth order moment relaxation for solving (4.1) is

$$\begin{cases} \vartheta_k := \min_y & \langle \theta, y \rangle \\ \quad s.t. & y_0 = 1, \, L_p^{(k)}[y] = 0 \, (p \in \Phi), \\ & M_k[y] \succeq 0, \, L_q^{(k)}[y] \succeq 0 \, (q \in \Psi), \\ & y \in \mathbb{R}^{\mathbb{N}_{2k}^n}, \end{cases} \tag{4.3}$$

where $M_k[y]$ is the kth order moment matrix generated by y, and $L_p^{(k)}[y]$, $L_q^{(k)}[y]$ are kth localizing matrices of p, q generated by y. Its dual problem is the kth order SOS relaxation for (4.1)

$$\begin{cases} \gamma_k := \max \gamma \\ \quad s.t. \quad \theta - \gamma \in \text{Ideal}[\Phi]_{2k} + \text{Qmod}[\Psi]_{2k}. \end{cases} \tag{4.4}$$

For relaxation orders $k = d_0, d_0 + 1, \ldots$, we get the Moment-SOS hierarchy of semidefinite relaxations (4.3)-(4.4). The optimization (4.3) is a relaxation of (4.1). In fact, if x is a feasible point of (4.1), then $y = [x]_{2k}$ is feasible for (4.3), while the infeasibility of (4.3) implies the infeasibility of (4.1). On the other hand, ϑ_k and γ_k provide lower bounds for ϑ for every k, that

$$\gamma_k \leq \vartheta_k \leq \vartheta.$$

Let e_i be the labeling vector such that its ith entry is 1 while all other entries are 0s, and y_{e_i} be the subvector of the tms y whose entries are labeled by e_i (for instance, when $n = 4$, $y_{e_3} = y_{0010}$). For a given k, let $y^{(k)}$ be a minimizer of the kth moment relaxation (4.3), and $u^{(k)} := (y_{e_1}^{(k)}, \ldots, y_{e_n}^{(k)})$. Then $u^{(k)}$ must be a minimizer of (3.3) if (we refer to Section 2 for definitions of $\mathcal{Z}$ and $\mathcal{S}$)

$$u^{(k)} \in \mathcal{Z}(\Phi) \cap \mathcal{S}(\Psi), \quad \vartheta_k = \theta(u^{(k)}). \tag{4.5}$$

However, (4.5) usually does not hold when there is more than one minimizer. For such cases, the *flat truncation* can be applied to certify global optimality and obtain minimizers. To conclude, we summarize the following algorithm, namely, the Moment-SOS hierarchy of semidefinite relaxations, for solving polynomial optimization problems:

Algorithm 4.1 For the polynomial optimization problem (4.1). Initialize $k := d_0$.

Step 1 Solve the moment relaxation (4.3) for the minimum value ϑ_k. If (4.3) is infeasible, then the polynomial optimization problem is infeasible. Otherwise, solve (4.3) for a minimizer y^* and let $t := d_0$.

Step 2 Let $u^* := (y_{e_1}^*, \ldots, y_{e_n}^*)$. If u^* satisfies (4.5), then u^* is the minimizer for (4.1) and stop.

Step 3 If y^* satisfies the rank condition

$$\operatorname{rank} M_t[y^*] \; = \; \operatorname{rank} M_{t-d_0}[y^*], \tag{4.6}$$

then extract a set U_i of $r := \operatorname{rank} M_t[y^*]$ minimizers for (4.1) and stop.

Step 4 If (4.6) fails to hold and $t < k$, let $t := t+1$ and then go to Step 3; otherwise, let $k := k+1$ and go to Step 1.

For Algorithm 4.1, we say it has asymptotic convergence if $\vartheta_k \to \vartheta$ as $k \to \infty$, and we say it has finite convergence if there exists $K \in \mathbb{N}$ such that $\vartheta_k = \vartheta$ for all $k \geq K$. In Step 3, the rank condition (4.6) is called *flat truncation* [33]. It is a sufficient (and almost necessary) condition to check the finite convergence of moment relaxations. When (4.6) holds, the method in [12] can be used to extract r minimizers for (4.1). This method and Algorithms 4.1 are implemented in the software `GloptiPoly 3` [13]. In the following subsections, we discuss how to use Algorithm 4.1 for solving polynomial optimization problems (3.2-3.3) and (3.6-3.7), respectively.

4.1 Finding solution candidates to polynomial VIPs

In this subsection, we discuss how to solve the polynomial optimization for obtaining candidate solutions to polynomial VIPs, particularly for solving the problem (3.3). Note that (3.3) can be rewritten as (4.1) with

$$\begin{cases} \Phi := E, \quad \theta(x) := [x]_1^T \Theta[x]_1, \\ \Psi := I \cup \{(v - x)^T F(x) \,:\, v \in V\}. \end{cases} \tag{4.7}$$

Since the $\theta(x)$ is a generic positive definite quadratic polynomial, the problem (3.3) has a unique minimizer, as we show in Theorem 4.2. In practice, we usually let $\theta(x) := [x]_1^T R^T R[x]_1$ with R being a randomly generated $(n+1) \times (n+1)$ matrix. As we mentioned before, the infeasibility of (4.3) infers the infeasibility of (3.3), which further implies the nonexistence of the KKT points. Moreover, when (3.3) is feasible, the optimal value ϑ_k of (4.3) is a lower bound for the minimum value of (3.3), i.e., $\vartheta_k \leq [x]_1^T \Theta[x]_1$ for all x that is feasible for (3.3).

The convergence of Algorithm 4.1 for solving (3.3) is shown as follows.

Theorem 4.2 *Assume the set* $\operatorname{Ideal}[\Phi] + \operatorname{Qmod}[\Psi] \subseteq \mathbb{R}[x]$ *is archimedean.*

(i) If (3.3) is infeasible, then the moment relaxation (4.3) is infeasible when k is big enough.

(ii) Suppose (3.3) is feasible and Θ is a generic positive definite polynomial. Then (3.3) has a unique minimizer u, and $u^{(k)} \to u$ as $k \to \infty$. Furthermore, if $\Phi(x) = 0$ have finitely many real solutions, then Algorithm 4.1 has finite convergence with (4.6) satisfied.

Proof. (i) If (3.3) is infeasible, the constant polynomial -1 can be viewed as a positive polynomial on the feasible set of (3.3). Since $\operatorname{Ideal}[\Phi] + \operatorname{Qmod}[\Psi]$ is archimedean, we have $-1 \in \operatorname{Ideal}[\Phi]_{2k} + \operatorname{Qmod}[\Psi]_{2k}$, for k big enough, by the Putinar Positivstellensatz [45]. For such a big k, the SOS relaxation (4.4) is unbounded from above, hence the moment relaxation (4.3) must be infeasible.

(ii) When the problem (3.3) is feasible and Θ is generic positive definite, it must have a unique minimizer, by [36, Proposition 5.2]. Let u be this minimizer. The convergence of $u^{(k)}$ to u is shown in [49] or [33, Theorem 3.3]. For the special case that $\Phi(x) = 0$ has finitely many real solutions, the point $u^{(k)}$ must be equal to u and (4.6) holds, when k is large enough. This is shown in [24] (see also [34]). □

Remark 4.3 (i) By Theorem 4.2, when $\operatorname{Ideal}[\Phi] + \operatorname{Qmod}[\Psi] \subseteq \mathbb{R}[x]$ is archimedean, $u^{(k)}$ converges to the minimizer of (3.3). If the point $u^{(k)}$ is feasible and $\vartheta_k = \theta(u^{(k)})$, then $u^{(k)}$ must be

a minimizer of (3.3), regardless of the archimedeaness holds or not. Moreover, the infeasibility of (4.3) always implies the infeasibility of (3.3), regardless of the archimedeaness holds or not.

(ii) We would like to remark that if flat truncation (4.6) is satisfied, then (4.5) holds and rank $M_t[y^*] = 1$. However, it is possible that the $u^{(k)}$ is a minimizer for (3.3), while (4.6) does not hold (see [34]). When $\Phi(x) = 0$ has finitely many real solutions (even when archimedeaness does not hold), flat truncation holds for all k that is big enough with rank $M_t[y^*] = 1$ [24]. In the actual implementation of solving (3.3), we usually skip Step 3 in Algorithm 4.1.

(iii) By Theorem 3.5, the $\Phi(x) = 0$ defines a finite set in $\mathbb{R}^n$ when the VI (X, F) is given by general polynomials. For such cases, finite convergence of Algorithm 4.1 is guaranteed. Moreover, in our computational practice, Algorithm 4.1 always has finite convergence.

The polynomial optimization subproblem (3.6) can be solved in the same way by the Moment-SOS hierarchy like (4.3)-(4.4). The convergence property is the same. For the cleanness of the paper, we omit the details.

4.2 Verifying solution candidates for polynomial VIPs

In this subsection, we study how to solve the polynomial subproblem (3.2) for verifying whether a given solution candidate u solves VI (X, F) or not. When X is compact, problem (3.2) exists at least one minimizer. Moreover, if (3.2) is unbounded from below, then u cannot be a solution. In these cases, it suffices to choose an appropriate positive R such that (3.2) with the extra boundedness constraint $\|x - u\|^2 \leq R$ has a negative minimum. For the rest of this subsection, we suppose (3.2) has at least one minimizer.

Under the assumption that X is given by (1.2) with nonsingular tuple g, the LICQ holds for every $x \in X$. Thus, if $v \in X$ is a minimizer of (3.2), then it must be a KKT point of (3.2). Introducing dual variables $\lambda(x)$, problem (3.2) can be equivalently reformulated to (4.1), with

$$
\begin{cases}
\theta(x) := (x - u)^T F(u), \\
\Phi(x) := \{F(u) - \sum_{i=1}^{m} \lambda_i(x) \nabla g_i(x)\} \\
\qquad\quad \cup \{\lambda_i(x) g_i(x) : (i \in \mathcal{I})\} \cup \{g_i(x) : (i \in \mathcal{E})\}, \\
\Psi(x) := \{g_i(x)\,(i \in \mathcal{I})\} \cup \{\lambda_i(x)\,(i \in \mathcal{I})\},
\end{cases}
\tag{4.8}
$$

Therefore, Algorithm 4.1 can be applied to solve (3.2). Note that under our assumption, (3.2) must be feasible, so there is no need to check the feasibility in Step 1. The following convergence result for Algorithm 4.1 is straightforward by [39, Theorems 3.3] and [42, Theorems 4.4]:

Theorem 4.4 *Suppose (3.2) has a minimizer. Let θ, Φ, Ψ be given as in (4.8), and let (4.1) be the polynomial optimization reformulation for (3.2). If either one of the following holds:*

  (i) *Ideal$[\Phi]$ + Qmod$[\Psi]$ is archemedean; or*
  (ii) *Ideal$[(g_i)_{i \in \mathcal{E}}]$ + Qmod$[(g_i)_{i \in \mathcal{I}}]$ is archemedean; or*
  (iii) *The $q(x) = 0\,(q \in \Phi)$ has finitely many real solutions;*

then Algorithm 4.1 has finite convergence. Moreover, if we further assume that every minimizer of (4.1) is an isolated critical point of (3.2), then (4.6) holds at each minimizer of (4.1) for all k that is big enough.

4.3 Solving the maximization problem (3.7)

For the maximization problem (3.7) in Algorithm 3.6, we let

$$\begin{cases} \theta(x) := -[x]_1^T \Theta[x]_1, \quad \Phi := E, \\ \Psi := I \cup \{(v-x)^T F(x) : v \in V\} \cup \{\Theta(x^*) + \delta - [x]_1^T \Theta[x]_1\}. \end{cases} \tag{4.9}$$

Then the maximization problem (3.7) can be equivalently rewritten as the polynomial mini-mization problem (4.1), and their kth order Moment-SOS relaxations are given by (4.3-4.4). Algorithm 4.1 can be applied to implement the Moment-SOS hierarchy to solve (3.7).

For (3.7), its feasible set is always nonempty since x^* is a feasible point, so there is no need to check its feasibility. Moreover, if the optimal value of (3.7) is nonnegative, then the δ which defines (3.7) is applicable for (3.6). For the kth ordered moment relaxation (4.3) of (3.7), if its minimum value ϑ_k satisfies $\vartheta_k \geq \Theta(x^*)$, then the optimal value of (3.7) is nonnegative and we may terminate Algorithm 4.1 directly. This is because the ϑ_k provides a lower bound for the maximum of (3.7). However, if $\vartheta_k < 0$, then we still need (4.6) to hold for the guaranteed optimality. Moreover, the archimedeanness is always satisfied by (3.7), since it has the inequality constraint $[x]_1^T \Theta[x]_1 \geq [x^*]_1^T \Theta[x^*]_1 + \delta$.

The convergence of Algorithm 4.1 is as follows. If (4.3) is infeasible, then (4.1) has an empty feasible set. This is because the $[x]_d$ is feasible for (4.3) for every feasible point x of (4.1). Suppose (4.1) has a nonempty feasible set and Ideal$[\Phi]$ + Qmod$[\Psi]$ is archimedean. Then $\vartheta_k \to \vartheta_{\min}$ as $k \to \infty$ [21]. Furthermore, under some classical optimality conditions, the finite convergence is guaranteed [35]. Lastly, if $\Phi = 0$ has finitely many real solutions, then Algorithm 4.1 has finite convergence, even if the archimedeanness is not satisfied. Note that by Theorem 3.5, when the VI (X, F) is given by general polynomials, the $\Phi = 0$ has at most finitely many complex solutions.

5 Numerical Experiments

In this section, we apply our methods to solve polynomial VIPs. We apply the software `GloptiPoly 3` [13] and `Mosek` [28] to implement Moment-SOS relaxations for solving polynomial optimization problems. The computation is implemented in a Dell OptiPlex 7090 desktop, with an Intel® Core(TM) i9-10900 CPU at 2.80GHz×10 cores and 64GB of RAM, in a Windows 10 operating system. For the neatness of the paper, only four decimal digits are shown for computational results.

5.1 Explicit polynomial VIPs

First, we apply Algorithm 3.2 and Algorithm 3.6 to solve polynomial VIPs. In Step 3 of Al-gorithm 3.2, if the optimal value $\varepsilon = 0$, then $u^{(k)}$ is a solution to the VI (X, F). In numerical computations, we may not have $\varepsilon = 0$ exactly due to round-off errors. Typically, we accept the computed candidate as a solution when $|\varepsilon| \leq 10^{-6}$. Similarly, in Step 2 of Algorithm 3.6, if the optimal value γ^* satisfying $|\gamma^* - [x^*]_1^T \Theta[x^*]_1| \leq 10^{-6}$, then we regard the value δ being applicable for (3.6). Unless specifically mentioned, we compute a complete set of solutions if it is nonempty, for every example in this subsection.

Example 5.1 Consider the nonlinear complementarity problem (NCP) in [43]:

$$0 \leq F(x) := \begin{bmatrix} 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ 2x_1^2 + x_1 + x_2^2 + 10x_3 + 2x_4 - 2 \\ 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 9x_4 - 9 \\ x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3 \end{bmatrix} \perp x \geq 0.$$

This NCP is equivalent to $\mathrm{VI}(\mathbb{R}_+^4, F)$, whose feasible set $X := \mathbb{R}_+^4$. There are two solutions $(\sqrt{6}/2, 0, 0, 1/2)$, $(1, 0, 3, 0)$ to the NCP. Let $F_i(x)$ be the ith entry of $F(x)$. Then, the LMEs are

$$\lambda_i = F_i(x) \quad \text{for all} \quad i = 1, 2, 3, 4.$$

We ran Algorithm 3.6 and it took around 0.55 second to find solutions $u^{(1)}, u^{(2)}$ as follows:

$$u^{(1)} = (1.2247, 0.0000, 0.0000, 0.5000), \quad \varepsilon = -1.5499 \cdot 10^{-8};$$
$$u^{(2)} = (1.0000, 0.0000, 3.0000, 0.0000), \quad \varepsilon = -3.7262 \cdot 10^{-8}.$$

We also tested solving the KKT system (2.4) without LMEs. Treating $(\lambda_1, \ldots, \lambda_4)$ as additional variables, we formulated the polynomial optimization problem similar to (3.1) with a feasible set given by (2.4). Then, we found solutions to $\mathrm{VI}(\mathbb{R}_+^4, F)$ under the same framework as in Algorithms 3.2 and 3.6. We presented the comparison of the time consumption for approaches with or without LMEs in Table 1. In the table, the first and the second row show the time consumption for finding the first and second solution, respectively, and the last row is for checking the completeness of the solution set $\{u^{(1)}, u^{(2)}\}$. It should be pointed out that the time consumption of solving (3.7) and (3.6) for finding each solution or checking the completeness is included.

	with LMEs	without LMEs
the first solution	0.12s	2.97s
the second solution	0.20s	8.49s
the completeness of $\{u^{(1)}, u^{(2)}\}$	0.23s	10.23s

Table 1 Computational results for Example 5.1

Example 5.2 Consider the VI (X, F) with

$$F(x) = \begin{bmatrix} -x_1 + 4x_1 x_2 + x_2^2 + x_3 - x_4 + 1 \\ 2x_1^2 + x_1 - x_2^3 - 10x_3 + 2x_4 \\ 3x_1^3 + x_1 x_2 + 2x_2^2 - 2x_3 + 9x_4 \\ x_1^2 - 3x_2^2 + 2x_3 - 3x_4 - 4 \end{bmatrix}, \quad X = \{x \in \mathbb{R}_+^4 : x_1 x_2 x_3 x_4 = 2\}.$$

In this example, the set X is unbounded and nonconvex. Let $F_i(x)$ be the ith entry of $F(x)$. Then, the LMEs are

$$\lambda_0 = \frac{x^T F(x)}{-8}, \quad \lambda_i = F_i(x) + \lambda_0 \ (i = 1, 2, 3, 4).$$

We ran Algorithm 3.2 and the infeasibility of (3.1) was detected at the 3rd ordered moment relaxation, hence this VIP has no solution. It took around 0.31 seconds.

Similarly to Example 5.1, we tested solving the KKT system (2.4) without LMEs under the framework of Algorithms 3.2. For this VIP, since we only need to check the infeasibility of (3.1) to detect the nonexistence of solutions, we presented the comparison for solving moment relaxations (4.3) with various relaxation orders d. In Table 2, the first column is the relaxation order; and the second and third columns represent the time consumption and whether the dth order moment relaxation successfully detects the nonexistence of solutions or not. When LMEs are applied, the 2nd order moment relaxation is not well-defined, and the 4th order moment relaxation is unnecessary for checking the emptiness of $\mathrm{SOL}(X, F)$, since the 3rd order moment relaxation is already infeasible.

d	with LMEs		without LMEs	
	time	nonexistence of solutions	time	nonexistence of solutions
2	n.a.	n.a.	4.52s	not detected
3	0.31s	detected	32.45s	not detected
4	1.28s	detected	2031.27s	detected

Table 2 Computational results for Example 5.2

Example 5.3 Consider the two-player jointly convex generalized Nash equilibrium problem in [41]:

$$\text{1st player:} \begin{cases} \min\limits_{x_1 \in \mathbb{R}^3} 10x_1^T x_2 - \sum_{j=1}^3 x_{1,j} \\ s.t. \quad x_1^T x_1 + x_2^T x_2 \leq 1; \end{cases}$$

$$\text{2nd player:} \begin{cases} \min\limits_{x_2 \in \mathbb{R}^3} \sum_{j=1}^3 (x_{1,j} x_{2,j})^2 + (3\prod_{j=1}^3 x_{1,j} - 1)\sum_{j=1}^3 x_{2,j} \\ s.t. \quad x_1^T x_1 + x_2^T x_2 \leq 1, \end{cases}$$

where $x_1 := (x_{1,1}, x_{1,2}, x_{1,3})$ and $x_2 := (x_{2,1}, x_{2,2}, x_{2,3})$ are decision variables of the first and the second player respectively. Let $x := (x_1, x_2)$ and denote the first (resp., second) player's objective functions by $f_1(x)$ (resp., $f_2(x)$). Then, finding the *normalized equilibria* for this GNEP is equivalent to solving VI (X, F) with

$$F := \left(\frac{\partial f_1}{\partial x_{1,1}}, \frac{\partial f_1}{\partial x_{1,2}}, \frac{\partial f_1}{\partial x_{1,3}}, \frac{\partial f_2}{\partial x_{2,1}}, \frac{\partial f_2}{\partial x_{2,2}}, \frac{\partial f_2}{\partial x_{2,3}} \right)^T,$$

$$X := \{x \in \mathbb{R}^4 : 1 - x^T x \geq 0\}.$$

We refer to [6,41] for more details about GNEPs. One can check that the map F is nonmonotone on X. For this polynomial VIP, the LME is

$$\lambda = -\frac{x^T F(x)}{2}.$$

We ran Algorithm 3.6 and obtained the unique solution $u := (u_1, u_2)$ with

$$u_1 = (-0.4934, -0.4934, -0.4934), \ u_2 = (0.2998, 0.2998, 0.2998),$$

and $\varepsilon = -2.5461 \cdot 10^{-8}$. It took around 4.21 seconds.

Example 5.4 (i) Consider the VI (X, F) with

$$F(x) = \begin{bmatrix} x_1 + x_2 + x_3 + x_4 \\ x_1 - x_2^2 + x_3 - x_4 \\ -x_3 - x_1 x_2 \\ x_4 - x_1 x_2 \end{bmatrix}, \quad X = \left\{ x \in \mathbb{R}^4 : x^T x - 1 \geq 0, \ 2 - x^T x \geq 0 \right\}, \qquad (5.1)$$

where $F(x)$ is nonmonotone and X is nonconvex. For this problem, the LMEs are

$$\lambda_1 = (2 - x^T x) \cdot \frac{x^T F(x)}{2}, \quad \lambda_2 = (1 - x^T x) \cdot \frac{x^T F(x)}{4}.$$

We ran Algorithm 3.6 and obtained all solutions to the VI (X, F), which are

$$\begin{aligned} u &= (-0.2639, 1.3073, -0.4537, -0.1250), &\text{with} &\quad \varepsilon = -2.7460 \cdot 10^{-9}; \\ u &= (0.4365, -1.0536, 0.7694, -0.3279), &\text{with} &\quad \varepsilon = -7.2553 \cdot 10^{-9}; \\ u &= (-0.4108, -0.4710, 1.2655, 0.0899), &\text{with} &\quad \varepsilon = -1.1356 \cdot 10^{-8}; \\ u &= (-0.8126, 0.7417, 0.7227, -0.5169), &\text{with} &\quad \varepsilon = -9.7235 \cdot 10^{-9}. \end{aligned}$$

It took around 9.56 seconds.

(ii) In (5.1), if we change $F(x)$ to $\begin{bmatrix} -x_1 - x_2 - x_3 - x_4 \\ x_1 - x_2 + x_3 - x_4 \\ x_3 - x_1 x_2 \\ x_4 - x_1 x_2 \end{bmatrix}$, then Algorithm 3.2 took around 1.52

seconds to detect the emptiness of the SOL (X, F).

Example 5.5 Let $A, B \in \mathbb{R}^{n \times n}$ be a pair of matrices such that B is symmetric positive definite, and let C be a closed convex cone in $\mathbb{R}^n$. Consider the reformulation VI (X, F) for the *constrained eigenvalue problem* (see [6]) of A over the cone C, where

$$F(x) := Ax, \quad X := \{x \in C : x^T B x = 1\}. \tag{5.2}$$

(i) Suppose

$$C := \{x \in \mathbb{R}^n_+ : x_1 \geq x_2 + \cdots + x_n\}. \tag{5.3}$$

Then $X = \{x \in \mathbb{R}^n : g_0(x) = 0, \ g_1(x) \geq 0, \ \ldots, \ g_n(x) \geq 0\}$, where

$$g_0(x) = x^T B x - 1, \ g_1(x) = x_1 - x_2 - \cdots - x_n, \ g_i = x_i \ (i = 2, \ldots, n).$$

The LMEs for VI (X, F) are

$$\lambda_0 = \frac{x^T F(x)}{2}, \ \lambda_1 = F_1(x) - 2\lambda_0 \cdot \sum_{j=1}^n B_{1,j} x_j,$$
$$\lambda_i = F_i(x) - 2\lambda_0 \cdot \sum_{j=1}^n B_{i,j} x_j + \lambda_1 \ (i = 2, \ldots, n),$$

where $F_i(x)$ is the ith entry of $F(x)$ and $B_{i,j}$ means the element in the ith row and jth column of B. If we let

$$A = \begin{bmatrix} -8 & -4 & 8 & -6 \\ -8 & -4 & 4 & -9 \\ -7 & -6 & 1 & 9 \\ -6 & -5 & -7 & 4 \end{bmatrix}, \qquad B = \begin{bmatrix} 4 & 0 & 3 & -1 \\ 0 & 4 & -1 & -2 \\ 3 & -1 & 4 & 0 \\ -1 & -2 & 0 & 2 \end{bmatrix}, \tag{5.4}$$

then we obtained the unique solution u to VI (X, F) by Algorithm 3.6 with

$$u = (0.5534, 0.2372, 0.0000, 0.3162), \quad \varepsilon = -1.3994 \cdot 10^{-8}.$$

It took around 0.50 seconds.
(ii) Suppose C is the second order cone, i.e.,

$$C := \{x \in \mathbb{R}^n : \|\bar{x}\|_2 \leq x_n\}, \quad \bar{x} = (x_1, x_2, \ldots, x_{n-1}). \tag{5.5}$$

Then $X = \{x \in \mathbb{R}^n : g_0(x) = 0, \ g_1(x) \geq 0\}$, where $g_0(x) = x^T B x - 1$, $g_1(x) = x_n^2 - \bar{x}^T \bar{x}$. We remark that in this case, polynomial LMEs do not exist since the constraints are not nonsingular. However, there exist rational Lagrange multiplier expressions

$$\lambda_0 = \frac{x^T F(x)}{2}, \quad \lambda_1 = \frac{F_n(x) - 2\lambda_0 \sum_{i=1}^n B_{n,i} x_i}{2x_n}.$$

One may check that the denominator x_n is strictly positive over X, thus we may cancel it in the KKT system. We refer to [41] for more details about rational LMEs and how to reformulate the KKT system with them. We ran Algorithm 3.6 with matrices A and B given as in (5.4). It took around 4.24 seconds and find the unique solution

$$u = (0.6906, 0.5866, -0.3661, 0.9773), \quad \varepsilon = -5.8198 \cdot 10^{-7}.$$

Example 5.6 Consider the invariant capital stock problems [16]. Let $A \in \mathbb{R}^{n_2 \times n_1}$, $B \in \mathbb{R}_+^{n_2 \times n_1}$, $b \in \mathbb{R}^{n_2}$, $\rho \in (0,1)$ be the discount rate, and let $f(x)$ be a convex loss function. For a given vector $b_0 \in \mathbb{R}^{n_2}$, we denote by $P(b_0)$ the problem of finding a sequence of activity levels $\{x^{(t)}\}_{t=1}^\infty$ to solve

$$
\begin{cases}
\min\limits_{x^{(1)}, x^{(2)}, x^{(3)}, \ldots} & \sum_{t=1}^\infty \rho^{t-1} f(x^{(t)}) \\
\quad s.t. & Ax^{(1)} \le b_0 + b, \\
& Ax^{(t)} \le Bx^{(t-1)} + b, \quad t = 2, 3, \ldots \\
& x^{(t)} \in \mathbb{R}_+^{n_1}. \quad t = 1, 2, \ldots
\end{cases}
\tag{5.6}
$$

Then, the invariant capital stock problem is to find $u \in \mathbb{R}^n$ such that $\{x^{(t)}\}_{t=1}^\infty$ with every $x^{(t)} = u$ solves $P(Bu)$. By [6, Proposition 1.4.5], if there exists $v \in \mathbb{R}^m$ such that (u, v) solves $\mathrm{VI}(\mathbb{R}_+^{n_1 + n_2}, F)$, where

$$
F(x, y) = \begin{bmatrix} \nabla f(x) + (A^T - \rho B^T) y \\ b + (B - A) x \end{bmatrix},
\tag{5.7}
$$

then u is an invariant capital stock.

We ran Algorithm 3.6 with $n_1 = 4$, $n_2 = 3$, $\rho = 0.7$, and

$$
f(x) = \sum_{1 \le i \le j \le 4} x_i^2 x_j^2 + 2x_1^3 x_2 + 2x_1 x_2^3 + x_3^3 + x_4^3 - \sum_{i=1}^4 x_i,
$$

$$
A = \begin{bmatrix} -3 & 1 & 1 & -3 \\ 1 & -1 & 3 & 1 \\ 1 & 0 & -3 & 2 \end{bmatrix}, \quad
B = \begin{bmatrix} 5 & 4 & 1 & 1 \\ 0 & 2 & 0 & 5 \\ 2 & 5 & 4 & 4 \end{bmatrix}, \quad
b = \begin{bmatrix} 1 \\ -3 \\ 2 \end{bmatrix}.
$$

We obtained the unique solution

$$
u = (0.1861, 0.5845, 0.1715, 0.4868), \quad v = (-0.0000, 0.2270, 0.0000).
$$

The error $\varepsilon = -7.3551 \cdot 10^{-9}$. It took around 5.54 seconds.

Comparison with the PATH Solver. There exist many numerical algorithms for solving VIPs. Since there is no monotonicity or convexity assumption in general for the polynomial VIPs in Section 5.1, we only compared Algorithm 3.2 with the damped Newton's method for solving the MiCP (the KKT system (2.4)) via the path search [5,47], which is implemented with the well-developed Matlab software: the PATH Solver [8]. We remark that we also tried some other numerical methods introduced in [6], such as the hyperplane-projection method, interior point method, etc. However, they usually cannot solve the VIPs in Section 5.1, especially when the problem is nonmonotone or nonconvex. So we omit the numerical results of these methods for the cleanness of this paper. We used the default parameter setting, which can be found in [8, Table 6&7]. Feasible initial points are applied for these methods, that we use $(2, 1, 1, 1)$ for Example 5.2, the $(1, 0, 1, 0)$ for Examples 5.4(i-ii), the $0.5e_1$ for Example 5.5(i), and the zero vector for other problems. Besides that, we use the zero vector as the initial point corresponding to Lagrange multipliers.

Numerical results are presented in Table 3. The column 'time' is the consumed time in seconds for solving the $\mathrm{VI}(X, F)$ and verifying if a solution is obtained, and the column 'error' stands for the minimum of (3.2) at the computed solution. Here, we only compute one solution for each problem, hence the time consumption in rows 'Alg. 3.2' may be much smaller than that in the last subsection, since we do not find more solutions or check the completeness of the solution sets.

The observations are summarized as follows:

Example	PATH		Alg. 3.2	
	time	error	time	error
5.1	0.48	$2 \cdot 10^{-7}$	0.12	$1 \cdot 10^{-8}$
5.2	fail		no solution	
5.3	0.79	$6 \cdot 10^{-9}$	1.54	$2 \cdot 10^{-8}$
5.4(i)	0.40	2.98	0.46	$2 \cdot 10^{-9}$
5.4(ii)	0.13	2.41	no solution	
5.5(i)	fail		0.14	$1 \cdot 10^{-8}$
5.5(ii)	fail		1.51	$6 \cdot 10^{-7}$
5.6	0.46	$8 \cdot 10^{-9}$	5.54	$8 \cdot 10^{-9}$

Table 3 Comparison with PATH

– The software PATH failed to converge for Example 5.2 since the set of KKT points is empty. For Example 5.5(i), the PATH solver fails to solve the problem, because the Jacobian is singular at the solution. Besides that, PATH found the KKT point $(0.2301, -0.6948, 0.0715, 0.6776)$ for Example 5.4(i), and found the KKT point $(0.5345, 0.0000, -0.8018, -0.2673)$ for Example 5.4(ii). However, these KKT points obtained by the PATH solver are not solutions to the VI (X, F), since the feasible sets in these two examples are nonconvex.
– The damped Newton's method may be sensitive to the choice of initial points. For example, if we use e_1 as the initial point for Example 5.3, then the PATH Solver do not converge. In contrast, Algorithm 3.2 can efficiently find solutions for all examples in Section 5.1 if there exists any, or certify the emptiness of the SOL (X, F).

5.2 Randomly generated polynomial VIPs

In this subsection, we tested our method on VIPs that are given by polynomials with randomly generated coefficients. For all randomly generated VIPs, we only ran Algorithm 3.2, i.e., the algorithm for finding one solution, and we do not check the uniqueness of the solution.

For all instances in the following, we randomly generated 100 polynomial VIPs and solved them by Algorithm 3.2. In the experiment, a computed point is regarded as a solution to the VIP if the error $|\varepsilon| \leq 10^{-6}$. We allow 10 maximum iterations (i.e., $\ell = 10$) in Algorithm 3.2 for each VIP. We remark that for VIPs with convex feasible sets, Algorithm 3.2 guarantees to find a solution to the VIP at the initial loop (see Proposition 3.1). However, the computed solution may not satisfy $|\varepsilon| \leq 10^{-6}$ due to the numerical error. For such cases, we proceed to new loops of Algorithm 3.2 until a solution satisfying $|\varepsilon| \leq 10^{-6}$ is computed or the maximum ℓ is reached. All polynomial optimization subproblems in Algorithm 3.2 are solved by Algorithm 4.1 with the relaxation order $k = d_0, d_0 + 1, \ldots, d_0 + 4$. If the $(d_0 + 4)$th Moment-SOS relaxation cannot find a solution for the polynomial optimization, then we stop the algorithm and report a failure of solving this polynomial VIP.

5.2.1 Randomly generated ball constrained VIPs

Consider the VI$(\mathbb{B}_n, F)$ with $\mathbb{B}_n := \{x \in \mathbb{R}^n : x^T x \leq 1\}$. Since $\mathbb{B}_n$ is convex and compact, the SOL$(\mathbb{B}_n, F) \neq \emptyset$. For the given dimension n and degree d, we randomly generate the matrix $A \in \mathbb{R}^{n \times N}$ with $N = \binom{n+d}{n}$ using the Matlab function randn, and let $F(x) = A \cdot [x]_d$. We report the numerical results in Table 4. In the table, (n, d) is the pair of the dimension of x and the degree of every $F_i(x)$, 'SR' (success rate) is the rate of finding a solution for the VIP successfully, and 'time' is the average consumed time (in seconds) of finding solutions.

(n,d)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)	(4,9)
SR	100%	100%	100%	100%	100%	100%	100%	100%
time	0.25s	0.28s	0.34s	0.67s	1.03s	2.51s	3.95s	8.34s
(n,d)	(4,10)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)
SR	100%	100%	100%	99%	100%	100%	100%	100%
time	20.49s	0.28s	0.43s	1.22s	3.82s	9.26s	24.63	70.79s
(n,d)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(7,2)	(7,3)	(7,4)
SR	100%	100%	100%	99%	100%	100%	100%	100%
time	0.92s	1.52s	7.43s	35.38s	99.51s	1.58s	9.79s	41.18
(n,d)	(7,5)	(8,2)	(8,3)	(8,4)	(9,2)	(9,3)	(10,2)	(11,2)
SR	100%	100%	100%	100%	100%	100%	99%	100%
time	365.49s	4.60	43.98s	213.74s	12.44s	188.38s	47.22s	121.70s

Table 4 Computational results for randomly generated ball constrained VIPs

For this problem, we would like to remark on the following:

– When $(n,d) = (5,4)$, there is one case of VIP where Algorithm 3.2 can only find a solution whose error $|\varepsilon| = 9.04 \cdot 10^{-6} > 10^{-6}$.
– When $(n,d) = (6,5)$, there is one case of VIP where Algorithm 3.2 can only find a solution whose error $|\varepsilon| = 5.78 \cdot 10^{-6} > 10^{-6}$.
– When $(n,d) = (10,2)$, there is one case of VIP where we cannot find a minimizer for (3.3) due to the limit of memory.

5.2.2 Randomly generated constrained eigenvalue problems

Consider the constrained eigenvalue problem in Example 5.5. For the given dimension n, we randomly generate matrices $A \in \mathbb{R}^{n \times n}$ and $\hat{B} \in \mathbb{R}^{n \times n}$ using the Matlab function randn, and let $B := \hat{B}^T \hat{B}$. Then $x \in \mathbb{R}^n$ solves the constrained eigenvalue problem defined by A and B if it solves the VI (X, F) given by (5.2). We ran Algorithm 3.2 for the cone given in (5.3) and the second order cone as in (5.5), and reported numerical results in Tables 5. In the table, the column 'C' is the cone C we use for the constrained eigenvalue problem, 'n' is the dimension for x, 'SR' is the rate of solving the VI (X, F) successfully, that either a solution is obtained or the nonexistence of solutions is detected, and 'time' has the same meaning as in Table 4.

C	n	3	4	5	6	7	8
(5.3)	SR	100%	100%	98%	100%	100%	98%
	time	0.60s	0.81s	0.82s	1.46s	8.03	20.85
(5.5)	SR	100%	100%	100%	99%	98%	
	time	0.98s	1.31s	2.21s	7.04s	45.24	

Table 5 Computational results for randomly generated constrained eigenvalue problems

For the case that C is given by (5.3), we would like to remark on the following

– When $n = 5$, there is one case of VIPs where Algorithm 3.2 can only find a solution whose error $|\varepsilon| = 9.50 \cdot 10^{-6} > 10^{-6}$. Moreover, there is another VIP where the Moment-SOS hierarchy cannot find a global minimizer for (3.3).
– When $n = 7$, there are two cases of VIPs where we cannot find a minimizer for (3.3) due to the limit of memory.
– When $n = 8$, there is one VIP where we cannot find a minimizer for (3.3) due to the limit of memory, and there is another VIP where we can only find a solution whose error $|\varepsilon| = 5.04 \cdot 10^{-6} > 10^{-6}$.

For the case that C is given by (5.5), we would like to remark the following

– When $n = 6$, there is one case of VIPs where we can only find a solution whose error $|\varepsilon| = 2.65 \cdot 10^{-6} > 10^{-6}$.

– When $n = 7$, there are two cases of VIPs where we cannot find a minimizer for (3.3) due to the limit of memory.

5.2.3 Randomly generated invariant capital stock problems

Consider the invariant capital stock problems (5.6) in Example 5.6. For the given dimensions n_1, n_2, we randomly generate $A \in \mathbb{R}^{n_2 \times n_1}$, $C \in \mathbb{R}^{n_1 \times n_1}$, $b \in \mathbb{R}^{n_2}$ using the `Matlab` function `randn`, and randomly generate $B \in \mathbb{R}^{n_2 \times n_1}_+$ using `rand`, and let $\rho = 0.8$, $f(x) := [x]_1^T C^T C[x]_1$. Then, the $\mathrm{VI}(\mathbb{R}^{n_1+n_2}_+, F)$ is given by (5.7). Note that although the feasible set of this VIP is convex, it is unbounded. It is possible that $\mathrm{SOL}(\mathbb{R}^{n_1+n_2}_+, F) = \emptyset$. We reported the numerical results in Table 6. In the table, (n_1, n_2) is the pair of dimensions, 'SR' (success rate) and 'time' have the same meaning with them in Table 4. We remark that the $\mathrm{VI}(X, F)$ computed in Example 5.6 has higher degrees, and we checked the uniqueness of the computed solution, thus the time consumption is larger than that in Table 6.

(n_1, n_2)	(4,2)	(5,2)	(6,2)	(7,2)	(8,2)	(9,2)
SR	100%	100%	100%	100%	100%	99%
time	0.25s	0.26s	0.28s	0.26s	0.32s	0.29s
(n_1, n_2)	(4,3)	(5,3)	(6,3)	(7,3)	(8,3)	(9,3)
SR	100%	100%	100%	100%	100%	100%
time	0.26s	0.29s	0.30s	0.34s	0.40s	0.42s
(n_1, n_2)	(4,4)	(5,4)	(6,4)	(7,4)	(8,4)	(9,4)
SR	100%	100%	100%	100%	99%	100%
time	0.35s	0.33s	0.43s	1.03s	0.47s	51.75s

Table 6 Computational result for randomly generated invariant capital stock problems

For this problem, we would like to remark on the following:

– When $(n_1, n_2) = (9, 2)$, there is a VIP where we can only find a solution whose error $|\varepsilon| = 5.22 \cdot 10^{-6} > 10^{-6}$.

– When $(n_1, n_2) = (8, 4)$, there is a VIP where the Moment-SOS hierarchy cannot find a global minimizer for (3.3).

– When $(n_1, n_2) = (9, 4)$, there are two VIPs where Algorithm 3.2 took more than 2000 seconds to solve each one of them (2371.65s and 2740.15s respectively), and the average time consumption for solving all other VIPs is 0.65s.

Appendices

A Generic properties of polynomial VIPs

First, we review some basics of algebraic geometry. Let $\mathbb{C}$ be the complex field and $\mathbb{C}[x]$ be the ring of polynomials in the variable $x = (x_{i,j})$. An *ideal* I of $\mathbb{C}[x]$ is a subset of $\mathbb{C}[x]$ such that

$a + b \in I$ for all $a, b \in I$ and $q \cdot p \in I$ for all $p \in I$ and $q \in \mathbb{C}[x]$. Given the ideal I, we say the polynomial tuple $(p_1, \ldots, p_m)$ is a generator for I if for every $p \in I$, we have

$$p = p_1 \cdot q_1 + \cdots + p_m \cdot q_m \text{ for some } q_1, \ldots, q_m \in \mathbb{C}[x].$$

Every ideal is generated by finitely many polynomials. For an ideal I, the set $V(I) := \{x \in \mathbb{C}^n : p(x) = 0, \forall p \in I\}$ is called the variety of I. Such a set is called an *affine variety*.

Let $\tilde{x} := (x_0, x_1, \ldots, x_n)$ be the tuple of variables in $\mathbb{C}^{n+1}$. The n-dimensional complex projective space $\mathbb{P}^n$ is the set of all lines passing through the origin of $\mathbb{C}^{n+1}$. A point $\tilde{x}$ in $\mathbb{P}^n$ has the coordinate $[x_0 : x_1 : \cdots : x_n]$ such that at least one of x_i is nonzero. An ideal generated by homogeneous polynomials in $\mathbb{C}[x]$ is called a *homogeneous ideal*. Given the homogeneous ideal $\tilde{I}$, the set of all $\tilde{x} \in \mathbb{P}^n$ such that $p(\tilde{x}) = 0$ for all $p \in \tilde{I}$ is called the *projective variety* of $\tilde{I}$. Moreover, the *Zariski topology* over $\mathbb{P}^n$ (resp., $\mathbb{C}^n$) is the topology space such that all closed sets are given by projective (resp., affine) varieties, and open subsets of projective varieties are called *quasi-projective varieties*. For the given quasi-projective varieties $\mathcal{X}$, closed subsets are given by intersections of $\mathcal{X}$ with a projective variety. It is clear that both projective and affine varieties are quasi-projective. Besides that, we say a variety being *irreducible* if it is not the union of two proper closed subsets. Any variety can be decomposite as the union of finitely many irreducible varieties, which are called *irreducible components*.

For the irreducible quasi-projective variety $\mathcal{X}$ in $\mathbb{P}^n$, its *dimension* $\dim \mathcal{X}$ is the largest non-negative integer k such that the intersection of $\mathcal{X}$ and k general hyperplanes is nonempty. The dimension of a reducible variety is defined by the maximum dimensions of all its irreducible components. The *codimension* $\operatorname{codim}(\mathcal{X})$ for the quasi-projective variety $\mathcal{X}$ in $\mathbb{P}^n$ equals $n - \dim(\mathcal{X})$. It is clear that $\dim \mathbb{P}^n = n$, and the dimension of all hypersurfaces in $\mathbb{P}^n$ equals $n - 1$. If $\mathcal{Y}$ is the hypersurface defined by a homogeneous polynomial $p(\tilde{x})$ that is not identically zero on any irreducible component of $\mathcal{X}$ and $\mathcal{X} \cap \mathcal{Y} \neq \emptyset$, then

$$\operatorname{codim}(\mathcal{X} \cap \mathcal{Y}) = \operatorname{codim} \mathcal{X} + 1.$$

A.1 Proof of Theorem 3.5

Now, we show the finiteness of $\mathcal{K}$ defined in Subsection 2.3 and the algebraic degree of the $\mathrm{VI}(X, F)$. Recall that when some constraint qualification condition holds at $x \in \mathrm{SOL}(X, F)$, x satisfies the KKT conditions (2.4) for the $\mathrm{VI}(X, F)$. Consider the following polynomial system

$$\begin{cases} F(x) = \sum_{i=1}^m \lambda_i \nabla g_i(x), \\ \lambda_1 \cdot g_1(x) = \cdots = \lambda_m \cdot g_m(x) = 0, \ g_i(x) = 0 \, (i \in \mathcal{E}). \end{cases} \tag{A.1}$$

Then, the (A.1) consists of all equalities in (2.4). We show the finiteness of complex solutions to (A.1) when all $F_1, \ldots, F_n, g_1, \ldots, g_m$ are generic polynomials. Let $\mathcal{A}$ be an active constraint labelling set. Without loss of generality, assume $\mathcal{A} = \{1, \ldots, \hat{m}\}$ with $\hat{m} \leq m$. From the generality of $g_1, \ldots, g_m$, the $\hat{m}$ cannot be greater than n. This is because the intersection of more than n hypersurfaces in $\mathbb{C}^n$ given by general polynomials in x is empty. Moreover, if we let

$$J(x) := [\, F(x) \quad \nabla g_1(x) \quad \ldots \quad \nabla g_{\hat{m}}(x) \,], \quad \mathcal{V} := \{x \in \mathbb{C}^n : \operatorname{rank} J(x) \leq \hat{m}\},$$

then $\mathcal{V}$ is a complex variety defined by vanishing all $(\hat{m}+1) \times (\hat{m}+1)$ minors of $J(x)$. Denote

$$\mathcal{U} := \{x \in \mathbb{C}^n : g_1(x) = \cdots = g_{\hat{m}}(x) = 0\}.$$

If $\mathcal{V} \cap \mathcal{U} = \emptyset$, then there does not exist x satisfying (A.1) with active labeling set $\mathcal{A}$.

Proof. (Proof of Theorem 3.5) For any given active labeling set $\mathcal{A}$, when all $(g_i(x))_{i \in \mathcal{A}}$ are general polynomials, the Jacobian matrix $[\nabla g_i(x)]_{i \in \mathcal{A}}$ is full column ranked over $\mathbb{C}^n$, by [30, Proposition 2.1] (see also [38, Theorem 3.2]). Therefore, when $g_1, \ldots, g_m$ are all general, the LICQ holds at every point in $\mathbb{C}^n$, and the $x \in \mathrm{SOL}\,(X, F)$ only if there exist $\lambda_1, \ldots, \lambda_m$ such that (2.4) holds. Moreover, the finiteness of $|\mathrm{SOL}\,(X, F)|$ is implied by the finiteness of solutions to (A.1).

If the solution set for (A.1) is finite for any given active labelling set $\mathcal{A}$, then (A.1) has finitely many solutions by enumerating all possibilities of $\mathcal{A}$. Without loss of generality, we assume $m < n$ and all constraints are active, i.e., $\mathcal{A} = [m]$, for convenience of our discussion.

For each $g_i(x)$, its homogenization is

$$\tilde{g}_i(\tilde{x}) := x_0^{b_i} \cdot g\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right). \tag{A.2}$$

Denote $\widetilde{\mathcal{U}} := \{\tilde{x} \in \mathbb{P}^n : \tilde{g}_1(\tilde{x}) = \cdots = \tilde{g}_m(\tilde{x}) = 0\}$. It is a projective variety which contains $\mathcal{U}$ up to the natural embedding from $\mathbb{C}^n$ to $\mathbb{P}^n$; see [50]. When $g_1, \ldots, g_m$ are general polynomials, we have $\dim \widetilde{\mathcal{U}} = n - m$ by Bertini's theorem [30, Theorem A.1]. Moreover, since g_i is general, we have $\nabla_x \tilde{g}(\tilde{x}) = x_0^{d_i - 1} \nabla_x g\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right)$ and the homogenization of ∇g_i equals the last n components of $\nabla_x \tilde{g}(\tilde{x})$. For each $i \in [N]$, we let

$$\tilde{F}_i(\tilde{x}) := x_0^{a_1} \cdot F_i\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right), \quad \tilde{F}(\tilde{x}) := (\tilde{F}_1(\tilde{x}), \tilde{F}_2(\tilde{x}), \ldots, \tilde{F}_n(\tilde{x})),$$

$$\widetilde{J}(\tilde{x}) := \left[\begin{array}{cccc} \tilde{F}(\tilde{x}) & \nabla \tilde{g}_1(\tilde{x}) & \ldots & \nabla \tilde{g}_m(\tilde{x}) \end{array}\right].$$

Then $\widetilde{\mathcal{V}} := \left\{\tilde{x} \in \mathbb{P}^n : \mathrm{rank} \widetilde{J}(x) \leq m\right\}$ is a projective variety that is given by homogeneous polynomials whose degrees are $c := a_1 + b_1 + \cdots + b_m - m$. Indeed, $\mathcal{V}$ is a quasi-projective variety that is given by $\widetilde{\mathcal{V}} \setminus \{x_0 \neq 0\}$ and $\dim \mathcal{V} = \dim \widetilde{\mathcal{V}}$. Let $\widetilde{\mathcal{W}} := \mathcal{V} \cap \widetilde{\mathcal{U}}$. Then, $\widetilde{\mathcal{W}}$ is an open subset of $\widetilde{\mathcal{U}}$, and the finiteness of the solution set to (A.1) follows directly if $\dim \widetilde{\mathcal{W}} = 0$.

Note that under our assumption, $\nabla \tilde{g}_1(\tilde{x}), \ldots, \nabla \tilde{g}_m(\tilde{x})$ are linearly independent for all $\tilde{x} \in \widetilde{U}$, i.e., $\widetilde{J}(\tilde{x}) = m$. For each tuple of indices $\sigma = (s_1, \ldots, s_m) \subset [n]$, if we let

$$\widetilde{J}_\sigma := \begin{bmatrix} \frac{\partial \tilde{g}_1}{\partial x_{s_1}} & \frac{\partial \tilde{g}_2}{\partial x_{s_1}} & \cdots & \frac{\partial \tilde{g}_m}{\partial x_{s_1}} \\ \frac{\partial \tilde{g}_1}{\partial x_{s_2}} & \frac{\partial \tilde{g}_2}{\partial x_{s_2}} & \cdots & \frac{\partial \tilde{g}_m}{\partial x_{s_2}} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial \tilde{g}_1}{\partial x_{s_m}} & \frac{\partial \tilde{g}_2}{\partial x_{s_m}} & \cdots & \frac{\partial \tilde{g}_m}{\partial x_{s_m}} \end{bmatrix}, \quad \mathcal{X}_\sigma = \{\tilde{x} \in \widetilde{\mathcal{U}} : \det \widetilde{J}_\sigma \neq 0\},$$

then all $\mathcal{X}_\sigma$ are quasi-projective varieties and

$$\widetilde{\mathcal{U}} = \bigcup_\sigma \mathcal{X}_\sigma, \quad \widetilde{\mathcal{W}} = \bigcup_\sigma \mathcal{X}_\sigma \cap \mathcal{V}. \tag{A.3}$$

For the given σ and constraining polynomials $g_1, \ldots, g_m$, we consider the intersection $\widetilde{\mathcal{W}}_\sigma := \mathcal{X}_\sigma \cap \mathcal{V}$. Since $\mathrm{rank} \widetilde{J}_\sigma = m$ on $\mathcal{X}_\sigma$, the $\widetilde{\mathcal{W}}_\sigma$ is given by vanishing all $(m+1) \times (m+1)$ minors of $\widetilde{J}(\tilde{x})$ containing rows indexed by σ, over the quasi-projective variety $\mathcal{X}_\sigma \setminus \{x_0 = 0\}$. There are $n - m$ such minors in total, and each one of them has exactly one row that is not indexed by entries in σ. Denote this row by s_0, and the according minor by $\mathfrak{m}_{s_0}$. Then $\mathfrak{m}_{s_0}$ is a homogeneous

polynomial parameterized by the coefficients of F_{s_0}. Let Λ be a subset of $[n] \setminus \sigma$ (possibly empty). For all $s_0 \notin \sigma \cup \Lambda$, the $\mathfrak{m}_{s_0}$ being identically zero on some irreducible components of

$$\mathcal{L} := (\bigcap_{s \in \Lambda} \{\mathfrak{m}_s = 0\}) \cap (\mathcal{X}_\sigma \setminus \{x_0 = 0\})$$

gives a proper closed subset in the space of coefficients for F_{s_0}, which can be regarded as a projective space. Therefore, by [50, Theorem 1.22], if F_{s_0} is a general polynomial in $\mathbb{C}[x]_{a_{s_0}}$, then

$$\dim \mathcal{L} \cap \{\mathfrak{m}_{s_0} = 0\} = \dim \mathcal{L} - 1.$$

Note that $\dim \mathcal{X}_\sigma = n - m$ since $\mathcal{X}_\sigma$ is open in $\mathcal{U}$. We conclude $\dim \widetilde{\mathcal{W}}_\sigma = 0$ when $F_1, \ldots, F_n$ are general polynomials by induction, and the finiteness of $\widetilde{\mathcal{W}}$ follows from (A.3). $\square$

A.2 The algebraic degree of polynomial VIPs

When $|\mathcal{K}|$ is finite, Algorithms 3.2 and 3.6 terminate within finitely many steps. Moreover, the quantity $|\mathcal{K}| + 1$ gives an upper bound for the number of iterations needed for these algorithms to terminate. In general, it is difficult to characterize the number of real solutions to the KKT system. As we mentioned before, the $\mathcal{K}$ is a subset of the intersection of quasi-projective varieties $\mathcal{U} \cap \mathcal{V}$, which is zero-dimensional when all defining polynomials for VI (X, F) are general. Therefore, we may estimate the quantity $|\mathcal{K}|$ by $|\mathcal{U} \cap \mathcal{V}|$, if the latter one is also finite.

Let $\mathcal{X}$ be a k-dimensional irreducible projective variety. The *algebraic degree* of $\mathcal{X}$, denoted by $\deg \mathcal{X}$, is the number of points in $\mathcal{X} \cap \mathcal{Z}$, where the $\mathcal{Z}$ is a general $(n - k)$-dimensional hyperplane. In particular, if $\dim \mathcal{X} = 0$, the $\deg \mathcal{X}$ counts the number of points in $\mathcal{X}$ (counting multiplicities). When $\mathcal{X}$ is reducible and $\dim \mathcal{X} = k$, the $\deg \mathcal{X}$ is the sum of degrees of all k-dimensional irreducible components for $\mathcal{X}$.

For two projective varieties $\mathcal{X}$ and $\mathcal{Y}$, we say they intersect properly if $\dim \mathcal{X} \cap \mathcal{Y} = \dim \mathcal{X} + \dim \mathcal{Y} - n$. The following result, called Bézout's Theorem [11], is useful when considering the algebraic degree for intersections. We refer to the textbook [11] for the definition of smooth varieties and transversal intersections.

Theorem A.1 *Let $\mathcal{X}$ and $\mathcal{Y}$ be projective varieties in $\mathbb{P}^n$ with dimensions k and l respectively. If $k + l \geq n$ and $\mathcal{X}$ intersect $\mathcal{Y}$ properly, then $\deg \mathcal{X} \cap \mathcal{Y} \leq \deg \mathcal{X} \cdot \deg \mathcal{Y}$. In particular, the equality holds when $\mathcal{X}$ and $\mathcal{Y}$ intersect transversely.*

For the positive integer r, denote by S_r the rth complete symmetric function on k letters $(n_1, n_2, ..., n_k)$, that

$$S_r := \sum_{i_1 + i_2 + \cdots + i_k = r} n_1^{i_1} n_2^{i_2} \ldots n_k^{i_k}.$$

Theorem A.2 *For the VI(X, F), let $a_1, \ldots, a_n \in \mathbb{N}$ and $b_1, \ldots, b_m \in \mathbb{N}_+$ be degrees for $F_1, \ldots, F_n$ and $g_1, \ldots, g_m$ respectively. If (A.1) is zero-dimensional, and all constraints are active, then the number of complex KKT points is bounded by*

$$b_1 b_2 \ldots b_m \cdot S_{n-m} \left(\max_i \{a_i\}, b_1, \ldots, b_m \right). \tag{A.4}$$

Proof. Without loss of generality, assume $a_1 = \max_i \{a_i\}$. Given an $\epsilon > 0$, let $\Delta F_i \in (\mathbb{C}[x]_{a_1})$ be a tuple of general perturbations to $F_i(x)$ for each $i \in [n]$, $\Delta g_i \in (\mathbb{C}[x]_{b_i})$ be a vector of general perturbations to $\nabla g_i(x)$ for each $i \in [m]$. For the parameter ϵ, denote $(F_i)_\epsilon := F_i + \epsilon \Delta F_i$,

$(\nabla g_i)_\epsilon := \nabla g_i + \epsilon \Delta g_i$ for every $i \in [m]$, and $F_\epsilon := ((F_1)_\epsilon, \ldots, (F_n)_\epsilon)$. Let $(\tilde{F}_i)_\epsilon$ and $(\nabla \tilde{g}_i)_\epsilon$ be homogenizations of $(F_i)_\epsilon$ and $(\nabla g_i)_\epsilon$, and let

$$J_\epsilon(\tilde{x}) := \begin{bmatrix} F_\epsilon(\tilde{x}) & (\nabla \tilde{g}_1)_\epsilon(\tilde{x}) & \ldots & (\nabla \tilde{g}_m)_\epsilon(\tilde{x}) \end{bmatrix},$$

$$\widetilde{\mathcal{V}}_\epsilon := \left\{ (\tilde{x}, \epsilon) \in \mathbb{P}^n \times \mathbb{C} : \operatorname{rank} \widetilde{J}_\epsilon(\tilde{x}) \le m \right\}.$$

The $\widetilde{\mathcal{V}}_\epsilon$ is a quasi-projective variety with a projection $\pi : \widetilde{\mathcal{V}}_\epsilon \to \mathbb{C}$ such that for all $\hat{\epsilon} \in \mathbb{C}$, the fiber $\pi^{-1}(\hat{\epsilon}) \subseteq \mathbb{P}^n$ is a projective variety. Since the perturbations are general, there exists at most finitely many $\hat{\epsilon}$ such that $\pi^{-1}(\hat{\epsilon}) \cap \mathcal{K}$ being 1-dimensional, and all other fibers intersection $\widetilde{\mathcal{U}}$ transversely at finitely many points. Thus $\dim(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}) = 1$. Furthermore, every irreducible component of $\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}$ has a dimension equal to 1, by [11, Proposition 17.24]. Note that for every $u \in \mathcal{V} \cap \widetilde{\mathcal{U}} = \pi^{-1}(0) \cap \widetilde{\mathcal{U}}$, it is contained in an irreducible component, thus there exists a curve $u(t) \subseteq \widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}$ such that the graph of $u(t)$ maps onto $\mathbb{C}$ and $u(0) = u$. For the general $t \in \mathbb{C}$, by genericity of every ΔF_i and Δg_i and the elimination theory (see [4]), we have $u(t) = (u_1(t), \ldots, u_n(t))$ satisfies

$$a_\xi(t) u_i(t)^\xi + a_{\xi-1}(t) u_i(t)^{\xi-1} + \cdots + a_1(t) u_i(t)^1 + a_0(t) = 0,$$

for every $i \in [n]$, where $\xi = \deg(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}})$ and all $a_\xi(t)$ are rational functions of the coefficients of F_i, g_i, ΔF_i and Δg_i. This is a univariate polynomial equation which has at most ξ many solutions. So there exist at most ξ many curves $u(t)$ described in above. Therefore, the intersection $\widetilde{\mathcal{V}} \cap \widetilde{\mathcal{U}}$ has at most ξ points, and $\deg(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}_\epsilon)$ gives an upper bound for the number of complex KKT points.

For the intersection $\deg(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}})$, we have

$$\deg(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}) \le \deg \widetilde{\mathcal{V}}_\epsilon \cdot \deg \widetilde{\mathcal{U}}.$$

The $\widetilde{\mathcal{U}}$ is given by vanishing m general homogeneous polynomials $g_1, \ldots, g_m$, and the degree of each g_i is b_i. Therefore, the algebraic degree for the complete intersection $\widetilde{\mathcal{U}}$ is at most $b_1 b_2 \ldots b_m$. In $J_\epsilon(\tilde{x})$, the degree for the 1st column is a_1, and the degree for the ith column equals $b_{i-1} - 1$ when $i \ge 2$. By [30, Proposition A.6],

$$\dim \widetilde{\mathcal{V}}_\epsilon = n - m, \quad \deg \widetilde{\mathcal{V}}_\epsilon \le S_{n-m}(a_1, b_1, \ldots, b_m).$$

So by Theorem A.1, we have

$$\deg(\widetilde{\mathcal{V}}_\epsilon \cap \widetilde{\mathcal{U}}) \le d_1 d_2 \ldots d_n S_{n-m}(a_1, b_1, \ldots, b_m).$$

And we conclude the upper bound for $|\mathcal{K}|$ by noticing that $a_1 = \max_i\{a_i\}$ from our assumption. $\square$

Remark A.3 For the VI (X, F), when there exist inequality constraints, some of them may not be active, and we can get an upper bound for $\mathcal{K}$ by enumerating all possible active labeling sets.

Remark A.4 When $a_1 = a_2 = \cdots = a_n$ and every $F_i(x)$ is a general polynomial in x with degree equal to a_i, and each $g_j(x)$ is a general polynomial in x with degree equal to a_j, then the upper bound (A.4) for the algebraic degree is tight. However, the upper bound (A.4) is large and usually much greater than $|\mathcal{K}|$, since it only provides an upper bound for all complex KKT points. Moreover, when there exists some $j \in [n]$ such that $a_j < \max_i\{a_i\}$, then the number of complex KKT points is smaller than the upper bound given by (A.4).

Declaration

Conflict of interest The authors declare that they have no conflict of interest.

References

1. X. Bai, Z. Huang, and Y. Wang. Global uniqueness and solvability for tensor complementarity problems, *J. Optim. Theory Appl.*, 170(1) (2016): 72–84.
2. D. Bertsekas. *Nonlinear programming*, second edition, Athena Scientific, 1995.
3. X. Chen, L. Qi, and D. Sun. Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities, *Math. Comput.*, 67 (1998): 519–540.
4. D. Cox, J. Little, and D. O'Shea, *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*, Springer Science & Business Media, 2013.
5. S.P. Dirkse, M.C. Ferris. The path solver: A nommonotone stabilization scheme for mixed complementarity problems, *Optim. Methods Softw.*, 5(2) (1995): 123–156.
6. F. Facchinei and J.S. Pang, *Finite-dimensional Variational Inequalities and Complementarity Problems*, Springer Science & Business Media, 2007.
7. J. Fan, J. Nie, and A. Zhou, Tensor eigenvalue complementarity problems, *Math. Program.*, 170(2) (2018): 507–539.
8. M. Ferris, and T.S. Munson. Interfaces to PATH 3.0: Design, implementation and usage, *Comput. Optim. Appl.*, 12(1) (1999): 207–227.
9. M.S. Gowda, and D. Sossa. Weakly homogeneous variational inequalities and solvability of nonlinear equations over cones, *Math. Program.*, 177(1) (2019): 149–171.
10. L. Guo, G. Lin, J.J. Ye and J. Zhang. Sensitivity analysis of the value function for parametric mathematical programs with equilibrium constraints, *SIAM J. Optim.*, 24 (2014): 1206–1237.
11. J. Harris, *Algebraic Geometry A First Course.*, Springer Verlag, Berlin, 1992.
12. D. Henrion and J. Lasserre. *Detecting global optimality and extracting solutions in GloptiPoly, Positive polynomials in control*, 293–310, Lecture Notes in Control and Inform. Sci., 312, Springer, Berlin, 2005.
13. D. Henrion, J. Lasserre, and J. Löfberg. Gloptipoly 3: moments, optimization and semidefinite programming, *Optim. Methods Softw.*, 24(4-5) (2009): 761–779.
14. V.T. Hieu. Solution maps of polynomial variational inequalities, *J. Global Optim.*, 77(4) (2020): 807–824.
15. V. Jeyakumar, JB. Lasserre, G. Li, and TS. Pham. Convergent semidefinite programming relaxations for global bilevel polynomial optimization problems. *SIAM J. Optim.* 26(1) (2016): 753–780.
16. P.C. Jones. Computing an optimal invariant capital stock, *SIAM J. Algebr. Discret. Methods*, 3(2) (1982): 145–150.
17. M. Kojima, N. Megiddo, T. Noma, and A. Yoshise. A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems, *Lecture Notes in Computer Science* 538, Springer-Verlag, Berlin, 1991.
18. I.V. Konnov. A class of combined iterative methods for solving variational inequalities, *J. Optim. Theory Appl.*, 94 (1997): 677–693.
19. I.V. Konnov. *Combined Relaxation Methods for Variational Inequalities*, Springer, Berlin, 2001.
20. G.M. Korpelevich. The extragradient method for finding saddle points and other problems, *Ekonomie i Mathematik Metody*, 12 (1976): 747–756.
21. J. Lasserre. Global optimization with polynomials and the problem of moments, *SIAM J. Optim.*, 11 (2001): 796–817.
22. J. Lasserre, *An introduction to polynomial and semi-algebraic optimization*, Cambridge University Press, Volume 52, 2015.
23. J. Lasserre. *The Moment-SOS Hierarchy*, Proceedings of the International Congress of Mathematicians (ICM 2018), vol. 3, B. Sirakov, P. Ney de Souza and M. Viana (Eds.), pp. 3761–3784, World Scientific, 2019.
24. J. Lasserre, M. Laurent and P. Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals, *Found. Comput. Math.*, 8(5) (2008): 607–647.
25. M. Laurent. Sums of squares, moment matrices and optimization over polynomials, *Emerging Applications of Algebraic Geometry of IMA Volumes in Mathematics and its Applications*, vol. 149, pp. 157–270, Springer, 2009.
26. M. Laurent. Optimization over polynomials: Selected topics, *Proceedings of the International Congress of Mathematicians*, ICM 2014, S. Jang, Y. Kim, D-W. Lee, and I. Yie (eds.), pp. 843–869, 2014.
27. X. Ma, M. Zheng, and Z. Huang. A note on the nonemptiness and compactness of solution sets of weakly homogeneous variational inequalities, *SIAM J. Optim.*, 30(1) (2020): 132–148.
28. MOSEK ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 10.0.* (2022). http://docs.mosek.com/9.0/toolbox/index.html
29. S. Z. Németh, G. Zhang. Conic optimization and complementarity problems, *Preprint*, 2016. arXiv:1607.05161
30. J. Nie and R. Kristian. Algebraic degree of polynomial optimization, *SIAM J. Optim.*, 20(1) (2009): 485–502.
31. J. Nie, L. Wang, and J.J. Ye. Bilevel polynomial programs and semidefinite relaxation methods, *SIAM J. Optim.*, 27(3) (2017): 1728-1757.
32. J. Nie, L. Wang, J.J. Ye, and S. Zhong. A Lagrange multiplier expression method for bilevel polynomial optimization, *SIAM J. Optim.*, 31(3) (2021): 2368-2395.
33. J. Nie. Certifying convergence of Lasserre's hierarchy via flat truncation, *Math. Program.*, 142(1-2) (2013): 485–510.

34. J. Nie. Polynomial optimization with real varieties, *SIAM J. Optim.*, 23(3) (2013): 1634–1646.
35. J. Nie. Optimality conditions and finite convergence of Lasserre's hierarchy, *Math. Program.*, 146(1-2) (2014): 97–121.
36. J. Nie. The $\mathcal{A}$-Truncated $\mathcal{K}$-Moment Problem, *Found. Comput. Math.*, 14(6) (2014): 1243–1276.
37. J. Nie. The hierarchy of local minimums in polynomial optimization, *Math. Program.*, 151(2) (2015): 555–583.
38. J. Nie. Discriminants and nonnegative polynomials, *J. Symb. Comput.*, 47(2) (2012): 167–191.
39. J. Nie. Tight relaxations for polynomial optimization and Lagrange multiplier expressions, *Math. Program.*, 178(1-2) (2019): 1–37.
40. J. Nie and X. Tang. Nash Equilibrium problems of polynomials, *Preprint*, 2020. `arXiv:2006.09490`
41. J. Nie and X. Tang. Convex generalized Nash equilibrium problems and polynomial optimization, *Math. Program.*, 2021. `https://doi.org/10.1007/s10107-021-01739-7`
42. J. Nie, Z. Yang and G. Zhou. The saddle point problem of polynomials, *Found. Comput. Math.*, 22(4) (2022):1133–1169.
43. J.-S. Pang, and S. Gabriel. NE/SQP: A robust algorithm for the nonlinear complementarity problem, *Math. Program.*, 60(1) (1993): 295–337.
44. J.-S. Pang. Newton's method for B-differentiable equations, *Math. Oper. Res.*, 15 (1990): 311–341.
45. M. Putinar. Positive polynomials on compact semi-algebraic sets, *Indiana U. Math. J.*, 42(3) (1993): 969–984.
46. L. Qi, D.F. Sun, G.L. Zhou. A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities, *Math. Program.*, 87(1) (2000): 1–35.
47. D. Ralph. Global convergence of damped Newton's method for nonsmooth equations via the path search, *Math. Oper. Res.*, 19(2) (1994): 352–389.
48. S.M. Robinson. Normal maps induced by linear transformations, *Math. Oper. Res.*, 17(3) (1992): 691–714.
49. M. Schweighofer. Optimization of polynomials on compact semialgebraic sets, *SIAM J. Optim.*, 15(3) (2005): 805–825.
50. I. R. Shafarevich, *Basic Algebraic Geometry 1*, Springer-Verlag Berlin Heidelberg, 2013.
51. M. Sibony. Méthodes itératives pour les équations et inéquations aux dérivées partielles non linéaires de type monotone, *Calcolo*, 7(1) (1970): 65–183.
52. Y. Wang, Z. Huang, and L. Qi. Global uniqueness and solvability of tensor variational inequalities, *J. Optim. Theory Appl.*, 177(1) (2018): 137–152.
53. Y. Ye, *Interior Point Algorithms*. Theory and analysis. John Wiley & Sons, Inc., New York, 1997.
54. X. Zhao, and J. Fan. A semidefinite method for tensor complementarity problems, *Optim. Methods Softw.*, 34(4) (2019): 758–769.