Mathematical Programming (2023) 198:1485–1518
https://doi.org/10.1007/s10107-021-01739-7

FULL LENGTH PAPER

Series B

# Convex generalized Nash equilibrium problems and polynomial optimization

Jiawang Nie[1] · Xindong Tang[2]

Received: 16 January 2021 / Accepted: 4 November 2021 / Published online: 7 December 2021

**Abstract**
This paper studies convex generalized Nash equilibrium problems that are given by polynomials. We use rational and parametric expressions for Lagrange multipliers to formulate efficient polynomial optimization for computing generalized Nash equilibria (GNEs). The Moment-SOS hierarchy of semidefinite relaxations are used to solve the polynomial optimization. Under some general assumptions, we prove the method can find a GNE if there exists one, or detect nonexistence of GNEs. Numerical experiments are presented to show the efficiency of the method.

**Keywords** Generalized Nash equilibrium problem · Convex polynomial · Polynomial optimization · Moment-SOS relaxation · Lagrange multiplier expression

**Mathematics Subject Classification** 90C33 · 91A10 · 90C22 · 65K05

## 1 Introduction

The generalized Nash equilibrium problem (GNEP) is a kind of game to find strategies for a group of players such that each player's objective function is optimized, for given other players' strategies. Suppose there are $N$ players and the $i$th player's strategy is a vector $x_i \in \mathbb{R}^{n_i}$ (the $n_i$-dimensional real Euclidean space). We write that

$$x_i := (x_{i,1}, \dots, x_{i,n_i}), \quad x := (x_1, \dots, x_N).$$

✉ Jiawang Nie
njw@math.ucsd.edu

Xindong Tang
xindong.tang@polyu.edu.hk

[1] Department of Mathematics, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA

[2] Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

The total dimension of all strategies is $n := n_1 + \cdots + n_N$. The main task of the GNEP is to find a tuple $u = (u_1, \ldots, u_N)$ of strategies such that each $u_i$ is a minimizer of the $i$th player's optimization

$$\mathrm{F}_i(u_{-i}): \begin{cases} \min\limits_{x_i \in \mathbb{R}^{n_i}} & f_i(u_1, \ldots, u_{i-1}, x_i, u_{i+1}, \ldots, u_N) \\ \ \ s.t. & g_{i,j}(u_1, \ldots, u_{i-1}, x_i, u_{i+1}, \ldots, u_N) = 0\,(j \in \mathcal{E}_i), \\ & g_{i,j}(u_1, \ldots, u_{i-1}, x_i, u_{i+1}, \ldots, u_N) \geq 0\,(j \in \mathcal{I}_i), \end{cases} \tag{1.1}$$

where $u_{-i} := (u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_N)$, the $f_i$ and $g_{i,j}$ are continuously differentiable functions in $x_i$, and the $\mathcal{E}_i, \mathcal{I}_i$ are disjoint finite (possibly empty) labeling sets. The point $u$ satisfying the above is called a generalized Nash equilibrium (GNE). For notational convenience, when the $i$th player's strategy is considered, we use $x_{-i}$ to denote the subvector of all players' strategies except the $i$th one, i.e.,

$$x_{-i} := (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N),$$

and write $x = (x_i, x_{-i})$ accordingly.

This paper focuses on the generalized Nash equilibrium problem of Polynomials (GNEPP), i.e., all the functions $f_i$ and $g_{i,j}$ are polynomials in $x$. For each $i = 1, \ldots, N$, let $X_i$ be the point-to-set map such that

$$X_i(x_{-i}) := \left\{ x_i \in \mathbb{R}^{n_i} \;\middle|\; \begin{array}{l} g_{i,j}(x_i, x_{-i}) = 0,\ j \in \mathcal{E}_i, \\ g_{i,j}(x_i, x_{-i}) \geq 0,\ j \in \mathcal{I}_i \end{array} \right\}. \tag{1.2}$$

The $X_i(x_{-i})$ is the feasible strategy set of $\mathrm{F}_i(x_{-i})$. The domain of $X_i$ is

$$\mathrm{dom}(X_i) := \{x_{-i} \in \mathbb{R}^{n-n_i} : X_i(x_{-i}) \neq \emptyset\}.$$

The tuple $x$ is said to be a feasible point of the GNEP if $x_i \in X_i(x_{-i})$ for all $i$. Denote the set

$$X := \left\{ x \in \mathbb{R}^{n} \;\middle|\; \begin{array}{l} g_{i,j}(x_i, x_{-i}) = 0,\ j \in \mathcal{E}_i,\ i = 1, \ldots, N, \\ g_{i,j}(x_i, x_{-i}) \geq 0,\ j \in \mathcal{I}_i,\ i = 1, \ldots, N \end{array} \right\}. \tag{1.3}$$

Then $x$ is a feasible point for the GNEP if and only if $x \in X$.

**Definition 1.1** The GNEP given by (1.1) is called *convex* [1] if for all $i = 1, \ldots, N$ and for all given $x_{-i} \in \mathrm{dom}(X_i)$, the objective $f_i(x_i, x_{-i})$ is convex in $x_i$ on $X_i(x_{-i})$, all $g_{i,j}(x_i, x_{-i})$ $(j \in \mathcal{E}_i)$ are affine linear in $x_i$, and all $g_{i,j}(x_i, x_{-i})$ $(j \in \mathcal{I}_i)$ are concave in $x_i$.

[1] In some literature, this is also called player-convex, to distinguish from jointly-convex GNEPs; see [14].

For instance, consider the 2-player GNEPP

$$\left.\begin{array}{ll} \min\limits_{x_1 \in \mathbb{R}^3} & \sum\limits_{j=1}^{3} (x_{1,j} - x_{2,j})^2 \\ s.t. & x_2^T x_1 - 1 = 0, \\ & (x_{11}, x_{12}, x_{13}) \geq 0; \end{array} \right| \begin{array}{ll} \min\limits_{x_2 \in \mathbb{R}^3} & \sum\limits_{j=1}^{3} \Big( (x_{2,j})^4 - x_{2,j} \prod\limits_{k=1}^{3} x_{1,k} \Big) \\ s.t. & \|x_1\|^2 - \|x_2\|^2 \geq 0. \\ & \end{array} \tag{1.4}$$

In the above, the $\| \cdot \|$ denotes the Euclidean norm. For each $i$, the Hessian of $f_i$ with respect to $x_i$ is positive semidefinite for all $x_{-i} \in \mathrm{dom}(X_i)$. All players have convex optimization problems, so this is a convex GNEP. One can directly check that it has a unique GNE $u = (u_1, u_2)$ with

$$u_1 = \left( \frac{\sqrt[3]{2}}{\sqrt{3}}, \frac{\sqrt[3]{2}}{\sqrt{3}}, \frac{\sqrt[3]{2}}{\sqrt{3}} \right), \; u_2 = \left( \frac{1}{\sqrt[6]{108}}, \frac{1}{\sqrt[6]{108}}, \frac{1}{\sqrt[6]{108}} \right).$$

GNEPs originated from economics in [4,10]. Recently, it has been widely used in many areas, such as economics, transportation, telecommunications and pollution control. Convex GNEPs often appear in applications. We refer to [1,3,8,52] for recent work on applications of GNEPs. Some application examples are shown in Sect. 6.

For the classical Nash equilibrium problems (NEPs) of polynomials, there exist semidefinite relaxation methods [2,48,50]. Convex GNEPs can be reformulated as variational inequality (VI) or quasi-variational inequality (QVI) problems [15,23,24, 36,51]. The Karush-Kuhn-Tucker (KKT) system for all player's optimization problems is considered in [13]. The penalty functions are used to solve convex GNEPs in [18, 19,22]. Some methods using the Nikaido-Isoda function are given in [14,57,58]. The Lemke's method is used to solve affine GNEPs [54]. For general nonconvex GNEPs, we refer to [5,12,16,28,47]. It is generally quite difficult to solve GNEPs, even if they are convex. This is because the KKT system of a convex GNEP may still be difficult to solve. The set of GNEs may be nonconvex, even for convex NEPs (see [48]). We refer to [17,20] for surveys on GNEPs.

## Contributions

This paper focuses on convex GNEPPs. Under some constraint qualifications, a feasible point is a GNE if and only if it satisfies the KKT conditions. We introduce rational and parametric expressions for Lagrange multipliers and formulate polynomial optimization for computing GNEs. Our major results are:

- For GNEPPs, we introduce the rational expression for Lagrange multipliers and study their properties. We prove the existence of rational expressions and give a sufficient and necessary condition for positivity of denominators. Moreover, we give parametric expressions for Lagrange multipliers for several cases. For all GNEPs, parametric expressions always exist.
- Using rational and parametric expressions, we formulate polynomial optimization and propose an algorithm for computing GNEs. Under some general assumptions,

we prove that the algorithm can compute a GNE if it exists, or detect nonexistence of GNEs. This is the first numerical method that has these properties, to the best of the authors' knowledge.

- The Moment-SOS semidefinite relaxations are used to solve polynomial optimization for finding and verifying GNEs. Numerical experiments are presented to show the efficiency of the method.

The paper is organized as follows. Some preliminaries about polynomial optimization are given in Sect. 2. We introduce rational expressions for Lagrange multipliers in Sect. 3. The parametric expressions for Lagrange multipliers are given in Sect. 4. We formulate polynomial optimization problems for computing GNEs and show how to solve them using the Moment-SOS hierarchy in Sect. 5. Numerical experiments and applications are given in Sect. 6. Conclusions and some discussions are given in Sect. 7.

## 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 $[k] := \{1, \dots, k\}$. For a real number $t$, $\lceil t \rceil$ (resp., $\lfloor t \rfloor$) denotes the smallest integer not smaller than $t$ (resp., the biggest integer not bigger than $t$). We use $e_i$ to denote the vector such that the $i$th 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). For the $i$th player's strategy vector $x_i \in \mathbb{R}^{n_i}$, the $x_{i,j}$ denotes the $j$th entry of $x_i$, for $j = 1, \dots, n_i$. When we write $(y, x_{-i})$, it means that the $i$th player's strategy is $y \in \mathbb{R}^{n_i}$, while the vector of all other players' strategy is fixed to be $x_{-i}$. Let $\mathbb{R}[x]$ denote the ring of polynomials with real coefficients in $x$, and $\mathbb{R}[x]_d$ denote its subset of polynomials whose degrees are not greater than $d$. For the $i$th player's strategy vector $x_i$, the notation $\mathbb{R}[x_i]$ and $\mathbb{R}[x_i]_d$ are defined in the same way. For $i$th player's objective $f_i(x)$, the notation $\nabla_{x_i} f_i$, $\nabla^2_{x_i} f_i$ respectively denote its gradient and Hessian with respect to $x_i$.

In the following, we use the letter $z$ to represent either $x$, $x_i$ or $(x, \omega)$ for some new variables $\omega$, for convenience of discussion. Suppose $z := (z_1, \dots, z_l)$. For a polynomial $p(z) \in \mathbb{R}[z]$, the $p = 0$ means $p(z)$ is identically zero on $\mathbb{R}^l$. We say the polynomial $p$ is nonzero if $p \neq 0$. Let $\alpha := (\alpha_1, \dots, \alpha_l) \in \mathbb{N}^l$, and we denote

$$z^\alpha := z_1^{\alpha_1} \cdots z_l^{\alpha_l}, \quad |\alpha| := \alpha_1 + \cdots + \alpha_l.$$

For an integer $d > 0$, denote the monomial power set

$$\mathbb{N}^l_d := \{\alpha \in \mathbb{N}^l : \ |\alpha| \le d\}.$$

We use $[z]_d$ to denote the vector of all monomials in $z$ whose degree is at most $d$, ordered in the graded alphabetical ordering. For instance, if $z = (z_1, z_2)$, then

$$[z]_3 = (1, z_1, z_2, z_1^2, z_1z_2, z_2^2, z_1^3, z_1^2z_2, z_1z_2^2, z_2^3).$$

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

### 2.1 Ideals and positive polynomials

Let $\mathbb{F} := \mathbb{R}$ or $\mathbb{C}$. For a polynomial $p \in \mathbb{F}[z]$ and subsets $I, J \subseteq \mathbb{F}[z]$, define the product and Minkowski sum

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

The subset $I$ is an ideal if $p \cdot I \subseteq I$ for all $p \in \mathbb{F}[z]$ and $I + I \subseteq I$. For a tuple of polynomials $q = (q_1, \dots, q_m)$, the set

$$\text{Ideal}[q] := q_1 \cdot \mathbb{F}[z] + \cdots + q_m \cdot \mathbb{F}[z]$$

is the ideal generated by $q$, which is the smallest ideal containing each $q_i$.

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

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

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

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

Similarly, we denote the truncation of Qmod$[g]$

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

The tuple $g$ determines the basic closed semi-algebraic set

$$\mathcal{S}(g) \ := \ \{z \in \mathbb{R}^l : g_1(z) \geq 0, \dots, g_t(z) \geq 0\}. \tag{2.1}$$

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

$$\mathcal{Z}(h) := \{z \in \mathbb{R}^l : h_1(z) = \ldots = h_s(z) = 0\}.$$

The set Ideal$[h]$ + Qmod$[g]$ is said to be *archimedean* if there exists $\rho \in$ Ideal$[h]$ + Qmod$[g]$ such that the set $\mathcal{S}(\rho)$ is compact. If Ideal$[h]$ + 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 $R - \|z\|^2 \geq 0$, then $\text{Ideal}[h] + \text{Qmod}[g, R - \|z\|^2]$ is archimedean and $\mathcal{Z}(h) \cap \mathcal{S}(g) = \mathcal{Z}(h) \cap \mathcal{S}(g, R - \|z\|^2)$. 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 [53]. 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 [40].

### 2.2 Localizing and moment matrices

Let $\mathbb{R}^{\mathbb{N}^l_{2d}}$ denote the space of all real vectors that are labeled by $\alpha \in \mathbb{N}^l_{2d}$. A vector $y \in \mathbb{R}^{\mathbb{N}^l_{2d}}$ is labeled as

$$y = (y_\alpha)_{\alpha \in \mathbb{N}^l_{2d}}.$$

Such $y$ is called a *truncated multi-sequence* (tms) of degree $2d$. For a polynomial $f = \sum_{\alpha \in \mathbb{N}^l_{2d}} f_\alpha z^\alpha \in \mathbb{R}[z]_{2d}$, define the operation

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

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

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

for some symmetric matrices $Q_\alpha$. Then we define the matrix function

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

It is called the $d$th *localizing matrix* of $q$ generated by $y$. For given $q$, the matrix $L_q^{(d)}[y]$ is linear in $y$. Localizing and moment matrices are important for getting semidefinite relaxations of solving polynomial optimization [29,38,39]. They are also useful for solving truncated moment problems [21,43] and tensor decompositions [44,45]. We refer to [31,32,34,35,37,42,50] for more references about polynomial optimization and moment problems.

### 2.3 Lagrange multiplier expressions

We study optimality conditions for generalized Nash equilibrium problems. Consider the $i$th player's optimization. For convenience, suppose $\mathcal{E}_i \cup \mathcal{I}_i = [m_i]$ and

$g_i := (g_{i,1}, \dots, g_{i,m_i})$. For a given $x_{-i}$, under some suitable constraint qualifications (e.g., the linear independence constraint qualification (LICQ), Mangasarian-Fromovite constraint qualification (MFCQ), or the Slater's Condition; see [7] for them), if $x_i$ is a minimizer of $\mathrm{F}_i(x_{-i})$, then there exists a Lagrange multiplier vector $\lambda_i := (\lambda_{i,1}, \dots, \lambda_{i,m_i})$ such that

$$\begin{cases} \nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \lambda_{i,j} \nabla_{x_i} g_{i,j}(x) = 0, \\ \lambda_i \perp g_i(x),\ g_{i,j}(x) = 0\,(j \in \mathcal{E}_i), \\ \lambda_{i,j} \geq 0\,(j \in \mathcal{I}_i),\ g_{i,j}(x) \geq 0\,(j \in \mathcal{I}_i). \end{cases} \tag{2.4}$$

This is called the first order Karush-Kuhn-Tucker system for $\mathrm{F}_i(x_{-i})$. Such $(x_i, \lambda_i)$ is called a critical pair of $\mathrm{F}_i(x_{-i})$. Therefore, if $x$ is a GNE, under constraint qualifications, then (2.4) holds for all $i \in [N]$, i.e., there exist Lagrange multiplier vectors $\lambda_1, \dots, \lambda_N$ such that

$$\begin{cases} \nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \lambda_{i,j} \nabla_{x_i} g_{i,j}(x) = 0\,(i \in [N]), \\ \lambda_i \perp g_i(x)\,(i \in [N]),\ g_{i,j}(x) = 0\,(i \in [N],\, j \in \mathcal{E}_i), \\ \lambda_{i,j} \geq 0\,(i \in [N],\, j \in \mathcal{I}_i),\ g_{i,j}(x) \geq 0\,(i \in [N],\, j \in \mathcal{I}_i). \end{cases} \tag{2.5}$$

A point $x$ satisfying (2.5) is called a KKT point for the GNEP. For convex GNEPs, each KKT point is a GNE [17, Theorem 4.6].

For each critical pair $(x_i, \lambda_i)$ of $\mathrm{F}_i(x_{-i})$, the Eq. (2.4) implies that

$$\underbrace{\begin{bmatrix} \nabla_{x_i} g_{i,1}(x) & \nabla_{x_i} g_{i,2}(x) & \cdots & \nabla_{x_i} g_{i,m_i}(x) \\ g_{i,1}(x) & 0 & \cdots & 0 \\ 0 & g_{i,2}(x) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_{i,m_i}(x) \end{bmatrix}}_{G_i(x)} \underbrace{\begin{bmatrix} \lambda_{i,1} \\ \lambda_{i,2} \\ \vdots \\ \lambda_{i,m_i} \end{bmatrix}}_{\lambda_i} = \underbrace{\begin{bmatrix} \nabla_{x_i} f_i(x) \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{\hat{f}_i(x)}. \tag{2.6}$$

If there exists a matrix polynomial $L_i(x)$ such that

$$L_i(x) G_i(x) = I_{m_i}, \tag{2.7}$$

then the Lagrange multipliers $\lambda_i$ can be expressed as

$$\lambda_i = L_i(x) \hat{f}_i(x).$$

The vector of polynomials $\lambda_i(x) := (\lambda_{i,1}(x), \dots, \lambda_{i,m_i}(x))$ is called a *polynomial expression for Lagrange multipliers* [46], where $\lambda_{i,j}(x)$ is the $j$th component of $L_i(x)\hat{f}_i(x)$. The matrix polynomial $G_i(x)$ is said to be nonsingular if it has full column rank for all $x \in \mathbb{C}^n$. It was shown that $G_i(x)$ is nonsingular if and only if there exists $L_i(x) \in \mathbb{R}[x]^{(m_i+n_i)\times m_i}$ such that (2.7) holds [46, Proposition 5.1]. The nonsingularity of $G_i(x)$ is independent of objective functions or other player's constraints.

For example, consider the GNEP given by (1.4). The first player's optimization has a polynomial expression of Lagrange multipliers

$$\lambda_{1,1} = x_1^T \nabla_{x_1} f_1,\ \lambda_{1,j+1} = \frac{\partial f_1(x)}{\partial x_{1,j}} - \lambda_{1,1} x_{2,j}\ (j = 1, 2, 3). \tag{2.8}$$

For the second player, the matrix polynomial $G_2(x)$ is not nonsingular, and polynomial expressions do not exist. In Sect. 6, we give a rational expression for the second player's Lagrange multipliers.

## 3 Rational expressions for Lagrange multipliers

In Sect. 2.3, a polynomial expression for the $i$th player's Lagrange multipliers exists if and only if the matrix $G_i(x)$ is nonsingular. For classical NEPs of polynomials, the nonsingularity holds generically [46,48]. However, this is often not the case for GNEPs. Let $g_i = (g_{i,1}, \ldots, g_{i,m_i})$ be the tuple of constraining polynomials in $\mathrm{F}_i(x_{-i})$ and $G_i(x)$ be the matrix polynomial as in (2.7). If there exists a matrix polynomial $\hat{L}_i(x)$ and a nonzero scalar polynomial $q_i(x)$ such that

$$\hat{L}_i(x) G_i(x) \;=\; q_i(x) \cdot I_{m_i}, \tag{3.1}$$

then $q_i(x)\lambda_i = \hat{L}_i(x)\hat{f}_i(x)$ for all critical pairs $(x_i, \lambda_i)$ of $\mathrm{F}_i(x_{-i})$. Let

$$\hat{\lambda}_i(x) \;:=\; \hat{L}_i(x)\hat{f}_i(x). \tag{3.2}$$

Denote by $\hat{\lambda}_{i,j}(x)$ the $j$th entry of $\hat{\lambda}_i(x)$.

**Definition 3.1** For the $i$th player's optimization $\mathrm{F}_i(x_{-i})$, if there exist polynomials $\hat{\lambda}_{i,1}, \ldots, \hat{\lambda}_{i,m_i}$ and a nonzero polynomial $q_i$ such that $q_i(x) \geq 0$ for all $x \in X$, and $\hat{\lambda}_{i,j}(x) = q_i(x)\lambda_{i,j}$ holds for all critical pairs $(x_i, \lambda_i)$, then we call the tuple

$$\hat{\lambda}_i/q_i := (\hat{\lambda}_{i,1}(x)/q_i(x), \ldots, \hat{\lambda}_{i,m_i}(x)/q_i(x))$$

a rational expression for Lagrange multipliers.

The following is an example of rational expression.

***Example 3.2*** Consider the 2-player convex GNEP

$$\left.\begin{array}{ll} \min\limits_{x_1 \in \mathbb{R}^2} & f_1(x_1, x_2) \\ s.t. & 2 - x_1^T x_1 - x_2 \geq 0; \end{array}\right| \begin{array}{ll} \min\limits_{x_2 \in \mathbb{R}^1} & f_2(x_1, x_2) \\ s.t. & 3x_2 - x_1^T x_1 \geq 0,\ 1 - x_2 \geq 0. \end{array} \tag{3.3}$$

The matrices of polynomials $G_1(x)$ and $G_2(x)$ are

$$G_1(x) := \begin{bmatrix} -2x_{1,1} \\ -2x_{1,2} \\ 2 - x_1^T x_1 - x_2 \end{bmatrix}, \quad G_2(x) := \begin{bmatrix} 3 & -1 \\ 3x_2 - x_1^T x_1 & 0 \\ 0 & 1 - x_2 \end{bmatrix}.$$

For $x_1 = (0, 0)$ and $x_2 = 2$, the $G_1(x)$ is the zero vector. For $x_1 = (\sqrt{3}, 0)$ and $x_2 = 1$, $\operatorname{rank}(G_2(x)) = 1$. Both $G_1(x), G_2(x)$ are not nonsingular, so there are no polynomial expressions for Lagrange multipliers. However, (3.1) holds for

$$\begin{gathered} q_1(x) = 2 - x_2,\ q_2(x) = 1 - \tfrac{1}{3}x_1^T x_1, \\ \hat{L}_1(x) = \begin{bmatrix} -\frac{x_{1,1}}{2} & -\frac{x_{1,2}}{2} & 1 \end{bmatrix},\ \hat{L}_2(x) = \begin{bmatrix} \frac{1}{3} - \frac{1}{3}x_2 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3}x_1^T x_1 - x_2 & 1 & 1 \end{bmatrix}. \end{gathered} \tag{3.4}$$

The Lagrange multiplier expressions are

$$\lambda_1 = \frac{-x_1^T \nabla_{x_1} f_1}{2q_1},\ \lambda_{2,1} = \frac{(1 - x_2)}{3q_2} \cdot \frac{\partial f_2}{\partial x_2},\ \lambda_{2,2} = \frac{x_1^T x_1 - 3x_2}{3q_2} \cdot \frac{\partial f_2}{\partial x_2}. \tag{3.5}$$

In Sect. 3.2, we show that if none of the $g_{i,j}$ is identically zero, then a rational expression for $\lambda_i$ always exists.

### 3.1 Optimality conditions and rational expressions

Suppose for each $i$, there exists a rational expression $\hat{\lambda}_i / q_i$ for the $i$th player's Lagrange multiplier vector. Since $q_i(x)\lambda_{i,j} = \hat{\lambda}_i(x)$ and $q_i(x) \geq 0$ for all $x \in X$, the following holds for all KKT points

$$\begin{cases} q_i(x)\nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(x)\nabla_{x_i} g_{i,j}(x) = 0\ (i \in [N]), \\ \hat{\lambda}_i(x) \perp g_i(x),\ g_{i,j}(x) = 0\ (j \in \mathcal{E}_i, i \in [N]), \\ g_{i,j}(x) \geq 0,\ \hat{\lambda}_{i,j}(x) \geq 0\ (j \in \mathcal{I}_i, i \in [N]). \end{cases} \tag{3.6}$$

Under some constraint qualifications, if $x$ is a GNE, then it satisfies (3.6). For convex GNEPs, if $x$ satisfies (3.6) and $q_i(x) > 0$, then $x$ must be a GNE, since it satisfies (2.5) with $\lambda_{i,j}$ given by $\lambda_{i,j} = \hat{\lambda}_{i,j}(x)/q_i(x)$. This leads us to consider the following optimization problem

$$\begin{cases} \min\limits_{x \in X} \ [x]_1^T \Theta [x]_1 \\ \ s.t.\ q_i(x)\nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(x)\nabla_{x_i} g_{i,j}(x) = 0\ (i \in [N]), \\ \qquad \hat{\lambda}_{i,j}(x) \perp g_{i,j}(x)\ (j \in \mathcal{E}_i \cup \mathcal{I}_i, i \in [N]), \\ \qquad \hat{\lambda}_{i,j}(x) \geq 0\ (j \in \mathcal{I}_i, i \in [N]). \end{cases} \tag{3.7}$$

In the above, $\Theta$ is a generically chosen positive definite matrix. The following proposition is straightforward.

**Proposition 3.3** *For the GNEPP given by (1.1), suppose for each $i \in [N]$, the Lagrange multiplier vector $\lambda_i$ has the rational expression as in* Definition 3.1.

(i) *If (3.7) is infeasible, then the GNEP has no KKT points. Therefore, if every GNE is a KKT point, then the infeasibility of (3.7) implies the nonexistence of GNEs.*
(ii) *Assume the GNEP is convex. If $u$ is a feasible point of (3.7) and $q_i(u) > 0$ for all $i \in [N]$, then $u$ must be a GNE.*

In Proposition 3.3 (ii), if $q_i(u) = 0$, then $u$ may not be a GNE. The following is such an example.

***Example 3.4*** [18, Example A.8] Consider the 3-player convex GNEP

$$\left.\begin{array}{cl} \min\limits_{x_1\in\mathbb{R}^1} & -x_1 \\ s.t. & x_3 \le x_1 + x_2 \le 1, \\ & x_1 \ge 0; \end{array}\right| \begin{array}{cl} \min\limits_{x_2\in\mathbb{R}^1} & (x_2 - 0.5)^2 \\ s.t. & x_3 \le x_1 + x_2 \le 1, \\ & x_2 \ge 0; \end{array} \left| \begin{array}{cl} \min\limits_{x_3\in\mathbb{R}^1} & (x_3 - 1.5x_1)^2 \\ s.t. & 0 \le x_3 \le 2. \\ & \end{array}\right.$$

For the first two players ($i = 1, 2$), the Eq. (3.1) holds for

$$\hat{L}_i(x) := \begin{bmatrix} x_i(1 - x_1 - x_2) & x_i & x_i & x_1 + x_2 - 1 \\ x_i(x_3 - x_1 - x_2) & x_i & x_i & x_1 + x_2 - x_3 \\ 0 & 0 & 0 & 1 - x_3 \end{bmatrix}, \quad q_i(x) := x_i(1 - x_3).$$

For the third player ($i = 3$), the Eq. (3.1) holds for

$$\hat{L}_3(x) := \frac{1}{2} \cdot \begin{bmatrix} 2 - x_3 & 1 & 1 \\ -x_3 & 1 & 1 \end{bmatrix}, \quad q_3 := 1.$$

The Lagrange multiplier expressions can be obtained by letting $\hat{\lambda}_i(x) := \hat{L}_i(x)\hat{f}_i(x)$. It is clear that $u_1 = 0, u_2 = 0.5, u_3 = 0$ satisfy (2.5) with $q_1(u) = 0$. However, $u_1 = 0$ is not a minimizer for the first player's optimization $\mathrm{F}_1(u_{-1})$. It is interesting to note that for $u_1 = \frac{2}{3}, u_2 = \frac{1}{3}, u_3 = 1$, the tuple $u = (u_1, u_2, u_3)$ satisfies (2.5) with $q_1(u) = q_2(u) = 0$, but $u$ is still a GNE [18].

We would like to remark that for some special GNEPs, the equality $q_i(u) = 0$ may imply that $u_i$ is a minimizer of $\mathrm{F}_i(u_{-i})$. See Example 3.8 for such a case.

### 3.2 Existence of rational expressions

We study the existence of rational expressions with nonnegative $q_i(x)$. The following is a useful lemma.

**Lemma 3.5** *For the $i$th player's optimization $F_i(x_{-i})$, if every $g_{i,j}(x)$ is not identically zero, then a rational expression exists for $\lambda_i$.*

***Proof*** Let $H_i(x) = G_i(x)^T G_i(x)$, where $G_i(x)$ is the matrix polynomial in (2.6). If every $g_{i,j}(x)$ is not identically zero, then the determinant $\det H_i(x)$ is also not identically zero. Let $\operatorname{adj} H_i(x)$ denote the adjoint matrix of $H_i(x)$, then

$$H_i(x) \cdot \operatorname{adj} H_i(x) = \det H_i(x) \cdot I_{m_i}.$$

For $\hat{L}_i(x) := \operatorname{adj} H_i(x) \cdot G_i(x)^T$, we get the rational expression

$$\lambda_{i,j}(x) = \frac{1}{\det H_i(x)} \hat{L}_i(x) \cdot \hat{f}_i(x). \tag{3.8}$$

Moreover, $q_i(x) \geq 0$ for all $x$, since $H_i(x)$ is positive semidefinite everywhere. □

The rational expression in (3.8) may not be very practical, because the determinantal polynomials often have high degrees. In practice, we usually have rational expressions with low degrees. If each $q_i(x) > 0$ for all $x \in X$, then every solution of (3.7) is a GNE. One wonders when a rational expression exists with $q_i(x) > 0$ on $X$. The matrix polynomial $G_i$ is said to be *nonsingular on* $X$ if $G_i(x)$ has full column rank for all $x \in X$. For the GNEP given in Example 3.2, both $G_1(x)$ and $G_2(x)$ are nonsingular on $X$. The following proposition is useful.

**Proposition 3.6** *The matrix $G_i(x)$ is nonsingular on $X$ if and only if there exists a matrix polynomial $\hat{L}_i(x)$ satisfying* (3.1) *with $q_i(x) > 0$ on $X$.*

***Proof*** First, if the matrix polynomial $G_i(x)$ has full column rank for all $x \in X$, let $H_i(x) := G_i(x)^T G_i(x)$, then $H_i(x)$ is positive definite and the determinant $\det H_i(x) > 0$ for all $x \in X$. Therefore, for $\hat{L}_i(x) := \operatorname{adj} H_i(x)$, the Eq. (3.8) is satisfied with $q_i(x) := \det H_i(x) > 0$ over $X$. Second, if (3.1) holds with $q_i(x) > 0$ on $X$, then $G_i(x)$ is clearly nonsingular on $X$. □

***Remark 1*** If $G_i(x)$ is nonsingular on $X$, then the LICQC must hold for the $i$th player's optimization. Furthermore, if this holds for all $i \in [N]$, then all GNEs are KKT points.

### 3.3 A numerical method for finding rational expressions

We give a numerical method for finding rational expressions for Lagrange multipliers. It was introduced in [49] for solving bilevel optimization problems. Let $G_i(x)$ be the matrix polynomial defined in (2.6). For convenience, denote the tuples

$$g_{\mathcal{E}} := (g_{i,j})_{i\in[N],j\in\mathcal{E}_i}, \quad g_{\mathcal{I}} := (g_{i,j})_{i\in[N],j\in\mathcal{I}_i}.$$

For a priori degree $d$, consider the following linear convex optimization:

$$\begin{cases} \max\limits_{\hat{L}_i,q_i,\gamma} \ \gamma \\ \quad s.t. \ \hat{L}_i \cdot G_i = q_i \cdot I_{m_i}, \ q_i(v) = 1, \\ \qquad\quad q_i - \gamma \in \operatorname{Ideal}[g_{\mathcal{E}}]_{2d} + \operatorname{Qmod}[g_{\mathcal{I}}]_{2d}, \\ \qquad\quad \hat{L}_i \in (\mathbb{R}[x]_{2d-\deg G_i})^{m_i \times (m_i+n_i)}. \end{cases} \tag{3.9}$$

In the above, the first equality is the same as (3.1). The second equality ensures that $q_i$ is not identically zero, where $v$ is a priori point in $X$. The constraint $q_i - \gamma \in \operatorname{Ideal}[g_{\mathcal{E}_i}] + \operatorname{Qmod}[g_{\mathcal{E}_i}]$ forces the $q_i(x) \geq \gamma$ on $X$. Therefore, if the maximum $\gamma$ is positive, then $q_i(x) > 0$ on $X$. By Lemma 3.5, one can always find a feasible $\gamma \geq 0$ satisfying (3.9), for some $d \leq \deg(H(x))$, if none of $g_{i,j}(x)$ is identically zero. By Proposition 3.6, if each $G_i(x)$ is nonsingular on $X$ and the archimedeanness holds for $X$, then there must exist $\gamma > 0$ satisfying (3.9) for some $d$. If $(\hat{L}_i, q_i, \gamma)$ is a feasible point of (3.9), then one can get a rational expression for Lagrange multipliers by letting $\hat{\lambda}_{i,j}(x) = \hat{L}_i(x)\hat{f}_i(x)$.

***Example 3.7*** Consider the GNEP in Example 3.2. We have

$$g_{\mathcal{E}} = \emptyset, \;\; g_{\mathcal{I}} = (2 - x_1^T x_1 - x_2, 3x_2 - x_1^T x_1, 1 - x_2).$$

Let $\hat{L}_1(x)$ and $\hat{L}_2(x)$ be the matrix polynomials in (3.4), and $q_1(x) = 2 - x_2$, $q_2(x) = 1 - \frac{1}{3}x_1^T x_1$. Let $v := (0, 0, 1)$ for both players, and $\gamma_1 = 1$, $\gamma_2 = 1/2$. Then, the $(\hat{L}_i(x), q_i(x), \gamma_i)$ is a feasible point of (3.11), for each $i = 1, 2$. In fact, we have

$$\begin{array}{l} q_1(v) = q_2(v) = 1, \quad q_1(x) - \gamma_1 = 1 - x_2 = 1 \cdot (1 - x_2) \in \operatorname{Qmod}[g_{\mathcal{I}}]_2, \\ q_2(x) - \gamma_2 = \frac{1}{2} - \frac{1}{3}x_1^T x_1 = \frac{1}{4}(2 - x_1^T x_1 - x_2) + \frac{1}{12}(3x_2 - x_1^T x_1) \in \operatorname{Qmod}[g_{\mathcal{I}}]_2. \end{array}$$

The rational expressions for Lagrange multipliers are given by (3.5).

***Example 3.8*** Consider the following GNEP

$$\begin{array}{ll|ll} \min\limits_{x_1 \in \mathbb{R}^3} & f_1(x_1, x_2) & \min\limits_{x_2 \in \mathbb{R}^3} & f_2(x_1, x_2) \\ s.t. & 1 - x_1^T x_1 - x_2^T x_2 \ge 0; & s.t. & 1 - x_1^T x_1 - x_2^T x_2 \ge 0. \end{array}$$

The constraining tuples $g_{\mathcal{E}} := \emptyset$, $g_{\mathcal{I}} := (1 - x_1^T x_1 - x_2^T x_2)$. Let $v := (0, 0, 0)$, $\gamma_1 = \gamma_2 = 0$, $q_1(x) = 1 - x_2^T x_2$, $q_2(x) = 1 - x_1^T x_1$, and

$$\hat{L}_1 = \left[ -\frac{1}{2}x_{1,1}, \; -\frac{1}{2}x_{1,2}, \; -\frac{1}{2}x_{1,3}, \; 1 \right], \;\; \hat{L}_2 = \left[ -\frac{1}{2}x_{2,1}, \; -\frac{1}{2}x_{2,2}, \; -\frac{1}{2}x_{2,3}, \; 1 \right].$$

One can verify that $q_1(v) = q_2(v) = 1$ and

$$\begin{array}{l} q_1(x) - \gamma_1 = 1 - x_2^T x_2 = x_1^T x_1 + 1 \cdot (1 - x_1^T x_1 - x_2^T x_2) \in \operatorname{Qmod}[g_{\mathcal{I}}]_2, \\ q_2(x) - \gamma_2 = 1 - x_1^T x_1 = x_2^T x_2 + 1 \cdot (1 - x_1^T x_1 - x_2^T x_2) \in \operatorname{Qmod}[g_{\mathcal{I}}]_2. \end{array}$$

By Proposition 3.6, we know $(\hat{L}_1(x), q_1(x), \gamma_1)$ and $(\hat{L}_2(x), q_2(x), \gamma_2)$ are minimizers of (3.9) for $i = 1, 2$ respectively. Therefore, we get the rational expression

$$\lambda_1 = \frac{-x_1^T \nabla_{x_1} f_1}{2 \cdot q_1(x)}, \;\; \lambda_2 = \frac{-x_2^T \nabla_{x_2} f_2}{2 \cdot q_2(x)}. \tag{3.10}$$

For each $i = 1, 2$, if $q_i(x) = 0$, then $0 \le {x_i}^T x_i \le 1 - {x_{-i}}^T x_{-i} = 0$. This implies $x_i = (0, 0, 0)$ is the only feasible point of the $i$th player's optimization and hence it is the minimizer. Therefore, each feasible point of (3.7) is a GNE.

One can solve (3.9) numerically for getting rational expressions. This is done in Example 6.6.

## 4 Parametric expressions for Lagrange multipliers

For some GNEPs, it may be difficult to find convenient rational expressions for Lagrange multipliers. Sometimes, the denominators may have high degrees. This is the case especially when $m_i > n_i$. If some $q_i$ has high degree, the polynomial optimization (3.6) also has a high degree, which makes the result moment SDP relaxations (see Sects. 5.1 and 5.2) very difficult to be solved. To fix such issues, we introduce parametric expressions for Lagrange multipliers.

**Definition 4.1** For the $i$th player's optimization $\mathrm{F}_i(x_{-i})$, a parametric expression for the Lagrange multipliers is a tuple of polynomials

$$\hat{\lambda}_i(x, \omega_i) := (\hat{\lambda}_{i,1}(x, \omega_i), \ldots, \hat{\lambda}_{i,m_i}(x, \omega_i)),$$

in $x$ and in a parameter $\omega_i := (\omega_{i,1}, \ldots, \omega_{i,s_i})$ with $s_i \le m_i$, such that $(x_i, \lambda_i)$ is a critical pair if and only if there is a value of $\omega_i$ such that (2.4) is satisfied for $\lambda_{i,j} = \hat{\lambda}_{i,j}(x, \omega_i)$ with $j \in [m_i]$.

The following is an example of parametric expressions.

***Example 4.2*** Consider the 2-player convex GNEP

$$\begin{array}{ll|ll}
\min\limits_{x_1 \in \mathbb{R}^2} & f_1(x_1, x_2) & \min\limits_{x_2 \in \mathbb{R}^2} & f_2(x_1, x_2) \\
s.t. & x_{1,1} - 2x_{1,2} + x_{2,2} \ge 0, & s.t. & x_{1,2} + x_{2,2} - x_{2,1}^2 + 1 \ge 0, \\
 & 1 - x_{2,1} \cdot x_1^T x_1 \ge 0, & & 2 - x_{2,2} \ge 0,\ 1 + x_{2,2} \ge 0, \\
 & x_{1,1} \ge 0,\ x_{1,2} \ge 0; & & x_{2,1} \ge 0.
\end{array}$$

The Lagrange multipliers can be expressed as

$$\begin{cases}
\lambda_{1,1} = \omega_{1,1}, \\
\lambda_{1,2} = \frac{1}{2}x_{1,1}(\frac{\partial f_1}{\partial x_{1,1}} - \omega_{1,1}) + \frac{1}{2}x_{1,2}(\frac{\partial f_1}{\partial x_{1,2}} + 2\omega_{1,1}), \\
\lambda_{1,3} = \frac{\partial f_1}{\partial x_{1,1}} - \omega_{1,1} + 2x_{2,1}x_{1,1}\lambda_{1,2}, \\
\lambda_{1,4} = \frac{\partial f_1}{\partial x_{1,2}} + 2\omega_{1,1} + 2x_{2,1}x_{1,2}\lambda_{1,2}; \\
\lambda_{2,1} = \omega_{2,1}, \\
\lambda_{2,2} = -\frac{1}{3} \cdot \left[(\frac{\partial f_2}{\partial x_{2,1}} + 2x_{2,1}\omega_{2,1})x_{2,1} + (\frac{\partial f_2}{\partial x_{2,2}} - \omega_{2,1})(x_{2,2} + 1)\right], \\
\lambda_{2,3} = \frac{\partial f_2}{\partial x_{2,2}} + \lambda_{2,2} - \omega_{2,1}, \\
\lambda_{2,4} = \frac{\partial f_2}{\partial x_{2,1}} + 2x_{2,1}\omega_{2,1}.
\end{cases} \tag{4.1}$$

Parametric expressions are quite useful for solving the GNEPs. The following are some useful cases.

(i) Suppose the $i$th player's optimization $\mathrm{F}_i(x_{-i})$ contains the nonnegative constraints, i.e., its constraints are

$$x_{i,1} \ge 0, \ldots, x_{i,n_i} \ge 0, \quad g_{i,j}(x) \ge 0\,(j = n_i + 1, \ldots, m_i).$$

Let $s_i := m_i - n_i$, then a parametric expression is

$$\boxed{\begin{array}{l} (\lambda_{i,1}, \ldots, \lambda_{i,n_i}) = \nabla_{x_i} f_i - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \nabla_{x_i} g_{i,k+n_i}, \\ (\lambda_{i,n_i+1}, \ldots, \lambda_{i,m_i}) = (\omega_{i,1}, \ldots, \omega_{i,s_i}). \end{array}} \tag{4.2}$$

(ii) Suppose the $i$th player's optimization $\mathrm{F}_i(x_{-i})$ contains box constraints, i.e., its constraints are

$$\begin{array}{r} x_{i,j} - a_{i,j} \geq 0,\ b_{i,j} - x_{i,j} \geq 0,\ j = 1, \ldots, n_i \\ g_{i,j}(x) \geq 0.\ j = n_i + 1, \ldots, m_i \end{array}$$

Let $s_i := m_i - 2n_i$, then a parametric expression is

$$\boxed{\begin{array}{ll} \lambda_{i,j} = \frac{b - x_{i,j}}{b-a} \cdot \left( \frac{\partial f_i}{\partial x_{i,j}} - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \frac{\partial g_{i,k+2n_i}}{\partial x_{i,j}} \right), & j = 1, 3, \ldots, 2n_i - 1 \\ \lambda_{i,j} = \frac{a - x_{i,j}}{b-a} \cdot \left( \frac{\partial f_i}{\partial x_{i,j}} - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \frac{\partial g_{i,k+2n_i}}{\partial x_{i,j}} \right), & j = 2, 4, \ldots, 2n_i \\ \lambda_{i,j} = \omega_{i,j-2n_i}. & j = 2n_i + 1, \ldots, m_i \end{array}} \tag{4.3}$$

(iii) Suppose the $i$th player's optimization $\mathrm{F}_i(x_{-i})$ contains simplex constraints, i.e., its constraints are

$$1 - e^T x_i \geq 0, x_{i,1} \geq 0, \ldots, x_{i,n_i} \geq 0,\ g_{i,j}(x) \geq 0,\ j = n_i + 2, \ldots, m_i.$$

Let $s_i := m_i - n_i - 1$, then a parametric expression is

$$\boxed{\begin{array}{ll} \lambda_{i,j} = (\nabla_{x_i} f_i - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \nabla_{x_i} g_{i,k+n_i+1})^T x_i, & j = 1 \\ \lambda_{i,j} = \frac{\partial f_i}{\partial x_{i,j-1}} - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \frac{\partial g_{i,k+n_i+1}}{\partial x_{i,j-1}} - \lambda_{i,1}, & j = 2, \ldots, n_i + 1 \\ \lambda_{i,j} = \omega_{i,j-n_i-1}. & j = n_i + 2, \ldots, m_i \end{array}} \tag{4.4}$$

(iv) Suppose the $i$th player's optimization $\mathrm{F}_i(x_{-i})$ contains linear constraints, i.e., its constraints are

$$a_j^T x_i - b_j(x_{-i}) \geq 0,\ j = 1, \ldots, r, \quad g_{i,j}(x) \geq 0,\ j = r + 1, \ldots, m_i,$$

where each $b_j$ is a polynomial in $x_{-i}$. Let $A = \begin{bmatrix} a_1 & \cdots & a_r \end{bmatrix}^T$. Assume $\mathrm{rank} A = r$. If we let $s_i := m_i - r$, then a parametric expression is

$$\boxed{\begin{array}{l} (\lambda_{i,1}, \ldots, \lambda_{i,r}) = (AA^T)^{-1} A (\nabla_{x_i} f_i - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \nabla_{x_i} g_{i,k+r}), \\ (\lambda_{i,r+1}, \ldots, \lambda_{i,m_i}) = (\omega_{i,1}, \ldots, \omega_{i,s_i}). \end{array}}$$

(v) Suppose there exists a labeling subset $\mathcal{T}_i := (t_1, \dots, t_r) \subseteq [m_i]$ such that

$$\hat{G}_i(x) := \begin{bmatrix} \nabla_{x_i} g_{i,t_1}(x) & \dots & \nabla_{x_i} g_{i,t_r}(x) \\ g_{i,t_1}(x) & & \\ & \ddots & \\ & & g_{i,t_r}(x) \end{bmatrix}$$

is nonsingular for all $x \in \mathbb{C}^n$. By [46, Proposition 5.1], there exists a matrix polynomial $D_i(x)$ such that $D_i(x) \cdot \hat{G}_i(x) = I_r$. Let $s_i := m_i - r$, then a parametric expression is

$$\boxed{\begin{array}{l} (\lambda_{i,1}, \dots, \lambda_{i,r}) = D_i(x)(\nabla_{x_i} f_i - \sum_{k=1}^{s_i} \omega_{i,k} \cdot \nabla_{x_i} g_{i,k+r}), \\ (\lambda_{i,r+1}, \dots, \lambda_{i,m_i}) = (\omega_{i,1}, \dots, \omega_{i,s_i}). \end{array}}$$

We would like to remark that parametric expressions for Lagrange multipliers always exist. For instance, one can get a parametric expression by letting $\omega_{i,j} = \lambda_{i,j}$ for all $j$. Such expression is called a *trivial parametric expression*. However, it is preferable to have small $s_i$, to save computational costs.

### 4.1 Optimality conditions and parametric expressions

Suppose all players have parametric expressions for their Lagrange multipliers as in Definition 4.1. Let $s := s_1 + \cdots + s_N$, and denote

$$\mathbf{x} := (x, \omega_1, \dots, \omega_N).$$

The optimality conditions (2.5) can be equivalently expressed as

$$\begin{cases} \nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(\mathbf{x}) \nabla_{x_i} g_{i,j}(x) = 0\,(i \in [N]), \\ \hat{\lambda}_i(\mathbf{x}) \perp g_i(x),\, g_{i,j}(x) = 0\,(j \in \mathcal{E}_i, i \in [N]), \\ g_{i,j}(x) \geq 0,\, \hat{\lambda}_{i,j}(\mathbf{x}) \geq 0\,(j \in \mathcal{I}_i, i \in [N]). \end{cases} \tag{4.5}$$

For convex GNEPs, a point $x$ is a GNE if there exists $\omega := (\omega_1, \dots, \omega_N)$ such that $\mathbf{x}$ satisfies (4.5). Therefore, we consider the optimization

$$\begin{cases} \min\limits_{\mathbf{x} \in X \times \mathbb{R}^s} & [\mathbf{x}]_1^T \, \Theta \, [\mathbf{x}]_1 \\ \quad s.t. & \nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(\mathbf{x}) \nabla_{x_i} g_{i,j}(x) = 0\,(i \in [N]), \\ & \hat{\lambda}_{i,j}(\mathbf{x}) \perp g_{i,j}(x)\,(j \in \mathcal{E}_i \cup \mathcal{I}_i, i \in [N]), \\ & \hat{\lambda}_{i,j}(\mathbf{x}) \geq 0\,(j \in \mathcal{I}_i, i \in [N]). \end{cases} \tag{4.6}$$

In the above, the $\Theta$ is a generically chosen positive definite matrix. The following proposition is straightforward.

**Proposition 4.3** *For the GNEPP given by* (1.1)*, suppose each player's optimization has a parametric expression for their Lagrange multipliers as in* Definition 4.1.

(i) *If* (4.6) *is infeasible, then the GNEP has no KKT points. If every GNE is a KKT point, then the infeasibility of* (4.6) *implies nonexistence of GNEs.*
(ii) *Assume the GNEP is convex. If* $(u, w)$ *is a feasible point of* (4.6)*, then* $u$ *is a GNE.*

## 5 The polynomial optimization reformulation

In this section, we give an algorithm for solving convex GNEPs. We assume each $\lambda_i$ has either a rational or parametric expression, as in Definition 3.1 or 4.1. If $\lambda_i$ has a polynomial or parametric expression, we let $q_i(x) := 1$. If $\lambda_i$ has a polynomial or rational expression, then we let $s_i = 0$. Recall the notation

$$\mathbf{x} \; := \; (x, \omega_1, \ldots, \omega_N).$$

Choose a generic positive definite matrix $\Theta$. Then solve the following polynomial optimization

$$\begin{cases} \min\limits_{\mathbf{x}} \; [\mathbf{x}]_1^T \, \Theta \, [\mathbf{x}]_1 \\ \; s.t. \; q_i(x)\nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(\mathbf{x})\nabla_{x_i} g_{i,j}(x) = 0 \, (i \in [N]), \\ \qquad \hat{\lambda}_{i,j}(\mathbf{x}) \perp g_{i,j}(x) \, (j \in \mathcal{E}_i \cup \mathcal{I}_i, i \in [N]), \\ \qquad g_{i,j}(x) = 0 \, (j \in \mathcal{E}_i, i \in [N]), \\ \qquad g_{i,j}(x) \geq 0 \, (j \in \mathcal{I}_i, i \in [N]), \\ \qquad \hat{\lambda}_{i,j}(\mathbf{x}) \geq 0 \, (j \in \mathcal{I}_i, i \in [N]). \end{cases} \tag{5.1}$$

If (5.1) is infeasible, then there are no KKT points. Since $\Theta$ is positive definite, if (5.1) is feasible, then it must have a minimizer, say, $(u, w) \in X \times \mathbb{R}^s$. For convex GNEPs, if $q_i(u) > 0$ for all $i$, then $u$ must be a GNE. If $q_i(u) \leq 0$ for some $i$, then $u$ may or may not be a GNE. To check this, we solve the following optimization problem for those $i$ with $q_i(u) \leq 0$

$$\begin{cases} \delta_i := \min\limits_{x_i} \; f_i(x_i, u_{-i}) - f_i(u_i, u_{-i}) \\ \qquad\quad s.t. \;\; g_{i,j}(x_i, u_{-i}) = 0 \, (j \in \mathcal{E}_i), \;\; g_{i,j}(x_i, u_{-i}) \geq 0 \, (j \in \mathcal{I}_i). \end{cases} \tag{5.2}$$

This is a polynomial optimization in $x_i$. Since $u \in X$, the point $u_i$ is feasible for (5.2), so $\delta_i \leq 0$. If $\delta_i \geq 0$ for all $i$, then $u$ must be a GNE. The following is an algorithm for solving the GNEP.

**Algorithm 5.1** *For the convex GNEP given by* (1.1)*, do the following:*

*Step 0* *Choose a generic positive definite matrix* $\Theta$ *of length* $n + s + 1$.
*Step 1* *Solve the polynomial optimization* (5.1)*. If it is infeasible, then there are no KKT points and stop; otherwise, solve it for a minimizer* $(u, w)$.

*Step 2* *If all $q_i(u) > 0$, then $u$ is a GNE. Otherwise, for those $i$ with $q_i(u) \leq 0$, solve the optimization* (5.2) *for the minimum value $\delta_i$. If $\delta_i \geq 0$ for all such $i$, then $u$ is a GNE; otherwise, it is not.*

In Step 0, we can choose $\Theta = R^T R$ for a randomly generated square matrix $R$ of length $n + s + 1$. When $\Theta$ is a generic positive definite matrix, the optimization (5.1) must have a unique minimizer, if its feasible set is nonempty. This is shown in Theorem 5.4(ii). Since the objective $f_i(x_i, u_{-i})$ is assumed to be convex in $x_i$, if it is bounded from below on $X_i(u_{-i})$, then (5.2) must have a minimizer (see [6, Theorem 3]). In applications, we are mostly interested in cases that (5.2) has a minimizer, for the existence of a GNE. In the Sects. 5.1 and 5.2, we will discuss how to solve polynomial optimization problems in Algorithm 5.1, by the Moment-SOS hierarchy of semidefinite relaxations. The convergence of Algorithm 5.1 is shown as follows.

**Theorem 5.2** *For the convex GNEPP given by* (1.1)*, suppose each Lagrange multiplier vector $\lambda_i$ has a rational expression as in* Definition 3.1 *or a parametric expression as in* Definition 4.1

(i) *If $(u, w)$ is a feasible point of* (5.1) *such that $q_i(u) > 0$ for all $i$, then $u$ is a GNE.*

(ii) *Assume every GNE is a KKT point. If* (5.1) *is infeasible, then the GNEP has no GNEs. If $\Theta$ is positive definite and every $q_i(x) > 0$ for all feasible $x$ of* (5.1)*, then* Algorithm 5.1 *will find a GNE if it exists.*

***Proof*** (i) This is directly implied by Propositions 3.3 and 4.3.

(ii) If (5.1) is infeasible, then there is no GNE, because every GNE is assumed to be a KKT point and it must be feasible for (5.1). Next, assume (5.1) is feasible. Since $\Theta$ is positive definite, the optimization (5.1) has a minimizer, say, $(u, w)$. By the given assumption, we have $q_i(u) > 0$ for all $i$. So $u$ is a GNE, by (i). □

***Remark 2*** For convex GNEPs, we can choose not to use nontrivial expressions for Lagrange multipliers, i.e., we consider the polynomial optimization (5.1) with $s_i = m_i$ and $\lambda_{i,j} = \omega_{i,j}$ for all $i$ and $j$. By doing this, we can get an algorithm like Algorithm 5.1 to get GNEs. However, this approach is usually very inefficient computationally, because it results in more variables for the polynomial optimization (5.1). Note that when Lagrange multiplier expressions (LMEs) are not used, each Lagrange multiplier is treated as a new variable. Moreover, solving (5.1) without LMEs may require higher order Moment-SOS relaxations. This is shown in numerical experiments in Sect. 5.1. In Example 6.1(i-ii), we compare the performance of Algorithm 5.1 with and without LMEs. Computational results show the advantage of using them.

In Theorem 5.2(ii), if $q_i(x) > 0$ for all $x \in X$, then we must have $q_i(x) > 0$ for all feasible $x$ of (5.1). Suppose $(u, w)$ is a computed minimizer of (5.1). If $u$ is not a GNE, i.e., $\delta_i < 0$ for some $i$, we can let $\mathcal{N} \subseteq [N]$ be the labeling set of $i$ with $\delta_i < 0$. By Theorem 5.2, we know $q_i(u) = 0$ for all $i \in \mathcal{N}$. For a priori small $\varepsilon > 0$, we can add the inequalities $q_i(x) \geq \varepsilon$ $(i \in \mathcal{N})$ to the optimization (5.1), to exclude $u$ from the feasible set. Then we solve the following new optimization

$$\begin{cases} \min\limits_{\mathbf{x}\in X\times\mathbb{R}^s} & [\mathbf{x}]_1^T\,\Theta\,[\mathbf{x}]_1 \\ s.t. & q_i(x)\nabla_{x_i} f_i(x) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(\mathbf{x})\nabla_{x_i} g_{i,j}(x) = 0\,(i\in[N]), \\ & \hat{\lambda}_{i,j}(\mathbf{x}) \perp g_{i,j}(x)\,(j\in\mathcal{E}_i\cup\mathcal{I}_i,\, i\in[N]), \\ & \hat{\lambda}_{i,j}(\mathbf{x}) \ge 0\,(j\in\mathcal{I}_i,\, i\in[N]), \\ & q_i(x)\ge\varepsilon\,(i\in\mathcal{N}). \end{cases} \tag{5.3}$$

If $\varepsilon > 0$ is not small enough, the constraint $q_i(x) \ge \varepsilon$ may also exclude some GNEs. If the new optimization (5.3) is infeasible, one can heuristically get a candidate GNE by choosing a different generic positive definite $\Theta$ in (5.1). In computational practice, when a GNE exists, it is very likely that we can get one by doing this. However, how to detect nonexistence of GNEs when (5.1) is feasible can be theoretically difficult. The theoretical side of this problem is mostly open, to the best of the authors' knowledge.

## 5.1 The optimization for all players

We discuss how to solve the polynomial optimization problems in Algorithm 5.1, by using the Moment-SOS hierarchy of semidefinite relaxations [29,31,32,34,35]. We refer to the notation in Sects. 2.1 and 2.2.

First, we discuss how to solve the optimization (5.1). Denote the polynomial tuples

$$\begin{aligned} \Phi_i := & \left\{q_i(x)\nabla_{x_i} f_i(\mathbf{x}) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(\mathbf{x})\nabla_{x_i} g_{i,j}(x)\right\} \cup \left\{g_{i,j}(x) : j\in\mathcal{E}_i\right\} \\ & \cup\left\{\hat{\lambda}_{i,j}(\mathbf{x})\cdot g_{i,j}(x) : j\in\mathcal{I}_i\right\}, \end{aligned} \tag{5.4}$$

$$\Psi_i := \left\{g_{i,j}(x) : j\in\mathcal{I}_i\right\} \cup \left\{\hat{\lambda}_{i,j}(\mathbf{x}) : j\in\mathcal{I}_i\right\}. \tag{5.5}$$

For notational convenience, for a vector $p = (p_1, \dots, p_s)$, the set $\{p\}$ stands for $\{p_1, \dots, p_s\}$, in the above. Denote the unions

$$\Phi := \bigcup_{i=1}^{N} \Phi_i, \quad \Psi := \bigcup_{i=1}^{N} \Psi_i.$$

They are both finite sets of polynomials. Then, the optimization (5.1) can be equivalently written as

$$\begin{cases} \vartheta_{\min} := \min\limits_{\mathbf{x}} \theta(\mathbf{x}) := [\mathbf{x}]_1^T\Theta[\mathbf{x}]_1 \\ \qquad\qquad s.t.\ p(\mathbf{x}) = 0\ (\forall\, p\in\Phi), \\ \qquad\qquad\quad q(\mathbf{x}) \ge 0\ (\forall\, q\in\Psi). \end{cases} \tag{5.6}$$

Denote the degree

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

For a degree $k \geq d_0$, consider the $k$th order moment relaxation for solving (5.6)

$$\begin{cases} \vartheta_k := \min\limits_{y} \ \langle \theta, y\rangle \\ \qquad\quad s.t.\ y_0 = 1,\ L_p^{(k)}[y] = 0\,(p \in \Phi), \\ \qquad\qquad\quad M_k[y] \succeq 0,\ L_q^{(k)}[y] \succeq 0\,(q \in \Psi), \\ \qquad\qquad\quad y \in \mathbb{R}^{\mathbb{N}^{n+s}_{2k}}. \end{cases} \tag{5.7}$$

Its dual optimization problem is the $k$th order SOS relaxation

$$\begin{cases} \max\ \gamma \\ s.t.\ \ \theta - \gamma \in \mathrm{Ideal}[\Phi]_{2k} + \mathrm{Qmod}[\Psi]_{2k}. \end{cases} \tag{5.8}$$

For relaxation orders $k = d_0, d_0 + 1, \ldots$, we get the Moment-SOS hierarchy of semidefinite relaxations (5.7)–(5.8). This produces the following algorithm for solving the polynomial optimization problem (5.6).

**Algorithm 5.3** *Let $\theta$, $\Phi$, $\Psi$ be as in* (5.6)*. Initialize $k := d_0$.*

*Step 1* *Solve the semidefinite relaxation* (5.7)*. If it is infeasible, then* (5.6) *has no feasible points and stop; otherwise, solve it for a minimizer $y^*$.*

*Step 2* *Let $\mathbf{u} = (u, w) := (y^*_{e_1}, \ldots, y^*_{e_{n+s}})$. If $\mathbf{u}$ is feasible for* (5.6) *and $\vartheta_k = \theta(u)$, then $\mathbf{u}$ is a minimizer of* (5.6)*. Otherwise, let $k := k + 1$ and go to Step 1.*

In the Step 2, $e_i$ denotes the labeling vector such that its $i$th entry is 1 while all other entries are 0. For instance, when $n = s = 2$, $y_{e_3} = y_{0010}$. The optimization (5.7) is a relaxation of (5.6). This is because if $\mathbf{x}$ is a feasible point of (5.6), then $y = [\mathbf{x}]_{2k}$ must be feasible for (5.7). Hence, if (5.7) is infeasible, then (5.6) must be infeasible, which also implies the nonexistence of KKT points. Moreover, the optimal value $\vartheta_k$ of (5.7) is a lower bound for the minimum value of (5.6), i.e., $\vartheta_k \leq \theta(\mathbf{x})$ for all $\mathbf{x}$ that is feasible for (5.6). In the Step 2, if $\mathbf{u}$ is feasible for (5.6) and $\vartheta_k = \theta(\mathbf{u})$, then $\mathbf{u}$ must be a minimizer of (5.6). The Algorithm 5.3 can be implemented in `GloptiPoly 3` [27]. The convergence of Algorithm 5.3 is shown as follows.

**Theorem 5.4** *Assume the set Ideal$[\Phi]$ + Qmod$[\Psi] \subseteq \mathbb{R}[\mathbf{x}]$ is archimedean.*

(i) *If* (5.6) *is infeasible, then the moment relaxation* (5.7) *must be infeasible when the order $k$ is big enough.*

(ii) *Suppose* (5.6) *is feasible and $\Theta$ is a generic positive definite matrix. Then* (5.6) *has a unique minimizer. Let $\mathbf{u}^{(k)}$ be the point $\mathbf{u}$ produced in the Step 2 of* Algorithm 5.3 *in the kth loop. Then $\mathbf{u}^{(k)}$ converges to the unique minimizer of* (5.6)*. In particular, if the real zero set of $\Phi$ is finite, then $\mathbf{u}^{(k)}$ is the unique minimizer of* (5.6)*, when $k$ is sufficiently large.*

***Proof*** (i) If (5.6) is infeasible, the constant polynomial $-1$ can be viewed as a positive polynomial on the feasible set of (5.6). Since $\text{Ideal}[\Phi]+\text{Qmod}[\Psi]$ is archimedean, we have $-1 \in \text{Ideal}[\Phi]_{2k} + \text{Qmod}[\Psi]_{2k}$, for $k$ big enough, by the Putinar Positivstellensatz [53]. For such a big $k$, the SOS relaxation (5.8) is unbounded from above, hence the moment relaxation (5.7) must be infeasible.

(ii) When the optimization (5.6) is feasible, it must have a unique minimizer, say, $\mathbf{x}^*$. To see this, let $\theta$ be defined as in (5.6), $K$ be the feasible set of (5.6), and $\mathcal{R}_2(K)$ be the set of tms's in $\mathbb{R}^{\mathbb{N}_2^n}$ admitting K-representing measures. Consider the linear conic optimization problem

$$\begin{cases} \min \ \langle \theta, y \rangle \\ s.t. \quad y_0 = 1, y \in \mathcal{R}_2(K). \end{cases} \tag{5.9}$$

If $\Theta$ is generic in the cone of positive definite matrice, the objective $\langle \theta, y \rangle$ is a generic linear function in $y$. By [41, Proposition 5.2], the optimization (5.9) has a unique minimizer. The minimum value of (5.9) is equal to $\vartheta_{\min}$. Therefore, (5.6) has a unique minimizer when $\Theta$ is generic. The convergence of $\mathbf{u}^{(k)}$ to $\mathbf{x}^*$ is shown in [55] or [38, Theorem 3.3]. For the special case that $\Phi(\mathbf{x}) = 0$ has finitely many real solutions, the point $\mathbf{u}^{(k)}$ must be equal to $\mathbf{x}^*$, when $k$ is large enough. This is shown in [33] (also see [39]). □

The archimedeaness of the set $\text{Ideal}[\Phi]+\text{Qmod}[\Psi]$ is essentially requiring that the feasible set of (5.6) is compact. The archimedeaness is sufficient but not necessary for Algorithm 5.3 to converge. Even if the archimedeaness fails to hold, Algorithm 5.3 is still applicable for solving (5.1). If the point $\mathbf{u}^{(k)}$ is feasible and $\vartheta_k = \theta(\mathbf{u}^{(k)})$, then $\mathbf{u}^{(k)}$ must be a minimizer of (5.1), regardless of the archimedeaness holds or not. Moreover, without archimedeaness, the infeasibility of (5.7) still implies that (5.1) is infeasible. In our computational practice, Algorithm 5.3 almost always has finite convergence.

The polynomial optimization (5.3) can be solved in the same way by the Moment-SOS hierarchy of semidefinite relaxations. The convergence property is the same. For the cleanness of this paper, we omit the details.

### 5.2 Checking generalized Nash equilibria

Suppose $\mathbf{u} = (u, w) \in \mathbb{R}^n \times \mathbb{R}^s$ is a minimizer of (5.1). For convex GNEPPs, if all $q_i(u) > 0$, then $u$ is a GNE, by Theorem 5.2(i). If $q_i(u) \leq 0$ for some $i$, we need to solve the optimization (5.2), to check if $u = (u_i, u_{-i})$ is a GNE or not, Note that (5.2) is a convex polynomial optimization problem in $x_i$. For given $u_{-i}$, if it is bounded from below, then (5.2) achieves its optimal value at a minimizer.

Consider the $i$th player's optimization with $q_i(u) \le 0$. For notational convenience, we denote the polynomial tuples

$$\begin{aligned} H_i(u) := & \left\{g_{i,j}(x_i, u_{-i}) : j \in \mathcal{E}_i\right\} \cup \left\{\hat{\lambda}_{i,j}(x_i, u_{-i}) \cdot g_{i,j}(x_i, u_{-i}) : j \in \mathcal{I}_i\right\} \\ & \cup \Big\{q_i(x_i, u_{-i}) \nabla_{x_i} f_i(x_i, u_{-i}) - \sum_{j=1}^{m_i} \hat{\lambda}_{i,j}(x_i, u_{-i}) \nabla_{x_i} g_{i,j}(x_i, u_{-i})\Big\}, \end{aligned} \tag{5.10}$$

$$J_i(u) := \left\{g_{i,j}(x_i, u_{-i}) : j \in \mathcal{I}_i\right\} \cup \left\{\hat{\lambda}_{i,j}(x_i, u_{-i}) : j \in \mathcal{I}_i\right\}. \tag{5.11}$$

Like in (5.4)-(5.5), the set $\{p\}$ stands for $\{p_1, \ldots, p_s\}$, when $p = (p_1, \ldots, p_s)$ is a vector of polynomial. The sets $H_i(u)$, $J_i(u)$ are finite collections.

Under some suitable constraint qualification conditions (e.g., the Slater's Condition), when (5.2) has a minimizer, it is equivalent to

$$\begin{cases} \eta_i := \min\limits_{x_i \in \mathbb{R}^{n_i}} & \zeta_i(x_i) := f_i(x_i, u_{-i}) - f_i(u_i, u_{-i}) \\ \quad\quad s.t. & p(x_i) = 0 \,(p \in H_i(u)), \\ & q(x_i) \ge 0 \,(q \in J_i(u)). \end{cases} \tag{5.12}$$

Denote the degree in variables $x_i$ for its constraining polynomials

$$\begin{aligned} d_i := \max \big\{ & \lceil \deg(\zeta_i(x_i, u_{-i}))/2, \deg(p(x_i))/2, \\ & \deg(q(x_i))/2 : p \in H_i(u), q \in J_i(u) \rceil \big\}. \end{aligned} \tag{5.13}$$

For a degree $k \ge d_i$, the $k$th order moment relaxation for (5.6) is

$$\begin{cases} \eta_i^{(k)} := \min\limits_{y} \langle \zeta_i(x_i), y \rangle \\ \quad\quad s.t. \; y_0 = 1, \; L_p^{(k)}[y] = 0 \,(p \in H_i(u)), \\ \quad\quad\quad M_k[y] \succeq 0, \; L_q^{(k)}[y] \succeq 0 \,(q \in J_i(u)), \\ \quad\quad\quad y \in \mathbb{R}^{\mathbb{N}_{2k}^{n_i}}. \end{cases} \tag{5.14}$$

The dual optimization problem of (5.14) is the $k$th order SOS relaxation

$$\begin{cases} \max \; \gamma \\ s.t. \;\; \zeta_i(x_i) - \gamma \in \text{Ideal}[H_i(u)]_{2k} + \text{Qmod}[J_i(u)]_{2k}. \end{cases} \tag{5.15}$$

By solving the above relaxations for $k = d_i, d_i + 1, \ldots$, we get the Moment-SOS hierarchy of relaxations (5.14)-(5.15). This gives the following algorithm.

**Algorithm 5.5** *For a minimizer $u = (u_i, u_{-i})$ of (5.1) with $q_i(u) \le 0$, solve the $i$th player's optimization (5.12). Initialize $k := d_i$.*

*Step 1* *Solve the moment relaxation (5.14) for the minimum value $\eta_i^{(k)}$ and a minimizer $y^*$. If $\eta_i^{(k)} \ge 0$, then $\eta_i = 0$ and stop; otherwise, go to the next step.*

*Step 2* *Let $t := d_i$ as in (5.13). If $y^*$ satisfies the rank condition*

$$\operatorname{rank} M_t[y^*] = \operatorname{rank} M_{t-d_i}[y^*], \tag{5.16}$$

*then extract a set $U_i$ of $r := \operatorname{rank} M_t(y^*)$ minimizers for (5.12) and stop.*

*Step 3* *If (5.16) fails to hold and $t < k$, let $t := t + 1$ and then go to Step 2; otherwise, let $k := k + 1$ and go to Step 1.*

We would like to remark that the optimization (5.12) is always feasible, because $u_i$ is a feasible point since $u$ is a minimizer of (5.1). The moment relaxation (5.14) is also feasible. Because $\eta_i^{(k)}$ is a lower bound for $\eta_i$, and $\eta_i \le \zeta_i(u_i, u_{-i}) = 0$, if $\eta_i^{(k)} \ge 0$, then $\eta_i$ must be 0. In Step 2, the rank condition (5.16) is called *flat truncation* [38]. It is a sufficient (and almost necessary) condition to check convergence of moment relaxations. When (5.16) holds, the method in [26] can be used to extract $r$ minimizers for (5.12). The Algorithm 5.5 can also be implemented in `GloptiPoly 3` [27]. If $\operatorname{Ideal}[H_i(u)] + \operatorname{Qmod}[J_i(u)]$ is archimedean, then $\eta_i^{(k)} \to \eta_i$ as $k \to \infty$ [29]. It is interesting to remark that

$$I_1 := \operatorname{Ideal}[g_{i,j}(x_i, u_{-i}) : j \in \mathcal{E}_i] \subseteq \operatorname{Ideal}[H_i(u)],$$
$$I_2 := \operatorname{Qmod}[g_{i,j}(x_i, u_{-i}) : j \in \mathcal{I}_i] \subseteq \operatorname{Qmod}[J_i(u)].$$

If $I_1 + I_2$ is archimedean, then $\operatorname{Ideal}[H_i(u)] + \operatorname{Qmod}[J_i(u)]$ must also be archimedean. Furthermore, we have the following convergence theorem for Algorithm 5.5.

***Remark 3*** To check the flat truncation (5.16), we need to evaluate the ranks of $M_t[y^*]$ and $M_{t-d_i}[y^*]$. Evaluating matrix ranks is a classical problem in numerical linear algebra. When a matrix is near to be singular, it may be difficult to determine its rank accurately, due to round-off errors. In computational practice, we often determine the rank of a matrix as the number of its singular values larger than a tolerance (say, $10^{-6}$). We refer to [11] for determining matrix ranks numerically. Moreover, when (5.12) has a unique minimizer, the ranks of $M_t[y^*]$ and $M_{t-d_i}[y^*]$ are one, the flat truncation (5.16) is relatively easy to check by looking at the largest singular value.

**Theorem 5.6** *For the convex polynomial optimization (5.2), assume its optimal value is achieved at a KKT point. If either one of the following conditions hold,*

(i) *The set $I_1 + I_2$ is archimedean, and the Hessian $\nabla^2_{x_i} \zeta_i(x_i^*, u_{-i}) \succ 0$ for a minimizer $x_i^*$ of (5.12); **or***

(ii) *The real zero set of polynomials in $H_i(u)$ is finite,*

*then* Algorithm 5.5 *must terminate within finitely many loops.*

***Proof*** Since its optimal value is achieved at a KKT point, the optimization problem (5.2) is equivalent to (5.12).

(i) If $I_1 + I_2$ is archimedean and $\nabla^2_{x_i} \zeta_i(x_i^*, u_{-i}) \succ 0$ if $x_i^*$ is a minimizer of (5.12), then $\zeta_i(x_i) - \eta_i \in I_1 + I_2$, by [9, Corollary 3.3]. Since

$$I_1 + I_2 \subseteq \operatorname{Ideal}[H_i(u)] + \operatorname{Qmod}[J_i(u)],$$

we have $\zeta_i(x_i) - \eta_i \in \mathrm{Ideal}[H_i(u)]_{2k} + \mathrm{Qmod}[J_i(u)]_{2k}$ for all $k$ big enough. Therefore, Algorithm 5.5 must terminate within finitely many loops, by the duality theory.

(ii) If the real zero set of polynomials in $H_i(u)$ is finite, then the conclusion is implied by [39, Theorem 1.1] and [38, Theorem 2.2]. □

***Remark 4*** If the objective polynomial in (5.2) is SOS-convex and its constraining ones are SOS-concave (see [25] for the definition of SOS-convex polynomials), then Algorithm 5.5 must terminate in the first loop (see [30]). If the optimal value of (5.2) is not achieved at a KKT point, the classical Moment-SOS hierarchy of semidefinite relaxations can be used to solve it. We refer to [9,29–32,34,35] for the work for solving general polynomial optimization.

## 6 Numerical experiments

In this section, we apply Algorithm 5.1 to solve convex GNEPs. To use it, we need Lagrange multiplier expressions. This can be done as follows.

- When polynomial expressions exist, we always use them. In particular, we use polynomial expressions for the first player of the GNEP given by (1.4), the first player in Example 6.1(ii), the third player in Examples 3.4 and 6.7(i-ii), the production unit and market players in Example 6.9.
- We use rational expressions for all players in Examples 6.3, 6.4 and 6.6. Moreover, rational expressions are used for the second player of the GNEP given by (1.4), the first two players in Examples 3.4 and 6.7(i-ii), and the consumer players in Example 6.9. For Example 6.6, the rational expression is obtained by solving (3.9) numerically.
- When it is difficult to find convenient polynomial or rational expressions, we use parametric expressions for Lagrange multipliers. For all players in Examples 6.5, 6.8, we use parametric expressions.

We apply the software `GloptiPoly 3` [27] and `SeDuMi` [56] to solve the Moment-SOS relaxations for the polynomial optimization (5.6) and (5.12). We use the software `YALMIP` for solving (3.9). The computation is implemented in an Alienware Aurora R8 desktop, with an Intel® Core(TM) i7-9700 CPU at 3.00GHz×8 and 16GB of RAM, in a Windows 10 operating system. For neatness of the paper, only four decimal digits are shown for the computational results.

In Step 2 of Algorithm 5.1, if the optimal values $\delta_i \geq 0$ for each $i$ such that $q_i(u) \leq 0$, then the computed minimizer of (5.1) is a GNE. In numerical computations, we may not have $\delta_i \geq 0$ exactly due to round-off errors. Typically, when $\delta_i$ is near zero, say, $\delta_i \geq -10^{-6}$, we regard the computed solution as an accurate GNE. In the following, all the GNEPs are convex.

***Example 6.1*** (i) For the GNEP given by (1.4), the first player has a polynomial expression for Lagrange multipliers given by (2.8), and the second player has a rational

**Table 1** The computational results for Example 6.1(i)

| | Algorithm 5.1 | | Without LME | |
|---|---|---|---|---|
| | Time | GNE | Time | GNE |
| $k = 2$ | n.a. | n.a. | 4.03 | No |
| $k = 3$ | 2.83 | Yes | 1350.09 | No |

expression given as

$$\lambda_{2,1} = \frac{-x_2^T \nabla_{x_2} f_2}{2q_2(x)}, \quad q_2(x) = x_1^T x_1.$$

For each $i$, the $q_i(x) > 0$ for all $x \in X$. We ran Algorithm 5.1 and obtained the GNE $u = (u_1, u_2)$ with

$$u_1 \approx (0.7274, 0.7274, 0.7274), \quad u_2 \approx (0.4582, 0.4582, 0.4582). \qquad (6.1)$$

It took around 2.83 seconds.
(ii) If the first player's objective is changed to

$$f_1(x) = (x_{2,1} + x_{2,2} - 2x_{2,3})(x_{1,1} + x_{1,2} - 2x_{1,3})^2 + x_{1,1} + x_{1,2} - 2x_{1,3},$$

then the GNEP has no GNE, detected by Algorithm 5.1. It took around 70.31 seconds to detect the nonexistence. The matrix polynomials $G_1(x)$ and $G_2(x)$ are nonsingular on $X$, so all GNEs must be KKT points if they exist.

In the following, we compare the performance of Algorithm 5.1 with the method of solving the optimization (5.1) without using Lagrange multiplier expressions, i.e., each Lagrange multiplier is treated as a new variable for polynomials. The comparison for Example 6.1(i) is given in Table 1. The computational results for the method using Lagrange multiplier expressions (i.e., for Algorithm 5.1) are given in the column labeled "Algorithm 5.1". The results for the method without using Lagrange multiplier expressions are given in the column labeled "Without LME". In the rows, the value $k$ is the relaxation order for solving (5.1). The subcolumn "time" lists the consumed time (in seconds) for solving the moment relaxation of order $k$, and the subcolumn "GNE" shows if a GNE is obtained or not. When $k = 2$ for Algorithm 5.1, the degree of relaxation is less than appearing polynomials, so we display that "not applicable (n.a.)". For Example 6.1(ii), the comparison is given in Table 2. This GNEP does not have a GNE. When no LMEs are used, the 4th order moment relaxation cannot be solved due to out of memory. However this can be done by using nontrivial LMEs.

***Example 6.2*** Consider the GNEP in Example 3.4. We use Lagrange multiplier expressions given there. By Algorithm 5.1, we obtained a feasible point $\hat{u} \approx 10^{-4} \cdot (0.1274, 0.4102, 0.3219)$ of (5.1) with $q_1(\hat{u}) \approx 0.1274 \cdot 10^{-4}$ and $q_2(\hat{u}) \approx 0.4102 \cdot 10^{-4}$. We solved (5.2), for $i = 1, 2$, to check if $\hat{u}$ is a GNE or not, and got

**Table 2** The computational results for Example 6.1(ii)

| | Algorithm 5.1 | | Without LME | |
|---|---|---|---|---|
| | Time | Nonexistence of GNE | Time | Nonexistence of GNE |
| $k = 2$ | n.a. | n.a. | 3.67 | Not detected |
| $k = 3$ | 2.77 | Not detected | 1201.75 | Not detected |
| $k = 4$ | 70.31 | Detected | Out of memory | |

$\delta_1 \approx -1.0000$, $\delta_2 \approx -1.8996 \cdot 10^{-10}$. Therefore, we solved (5.3) with $\mathcal{N} = \{1\}$ and $\varepsilon = 0.1$, and obtained a GNE $u = (u_1, u_2, u_3)$ with

$$u_1 \approx 0.5000,\ u_2 \approx 0.5000,\ u_3 \approx 0.7500,\ q_1(u) \approx q_2(u) \approx 0.1250.$$

It took around 0.89 second.

***Example 6.3*** Consider the GNEP in Example 3.2 with objectives

$$f_1(x) = \sum_{j=1}^{2}(x_{1,j} - 1)^2 + x_2(x_{1,1} - x_{1,2}),\ f_2(x) = (x_2)^3 - x_{1,1}x_{1,2}x_2 - x_2.$$

The rational expressions for both players are given by (3.5). For each $i$, the $q_i(x) > 0$ for all $x \in X$. We ran Algorithm 5.1 and got the GNE $u = (u_1, u_2)$ with

$$u_1 \approx (0.4897, 1.0259),\ u_2 \approx 0.7077.$$

It took around 0.20 second.

***Example 6.4*** Consider the GNEP in Example 3.8 with objectives

$$f_1(x) = 10x_1^T x_2 - \sum_{j=1}^{3} x_{1,j},\ f_2(x) = \sum_{j=1}^{3}(x_{1,j}x_{2,j})^2 + (3\prod_{j=1}^{3} x_{1,j} - 1)\sum_{j=1}^{3} x_{2,j}.$$

We use rational expressions as in (3.10). From Example 3.8, we know all feasible points of (5.1) are GNEs. By Algorithm 5.1, we got the GNE $u = (u_1, u_2)$ with

$$u_1 \approx (0.9864, 0.0088, 0.0088), \quad u_2 \approx (0.0836, 0.0999, 0.0999).$$

It took around 2.03 seconds.

***Example 6.5*** Consider the GNEP in Example 4.2 with objectives

$$\begin{aligned} f_1(x) &= x_{2,1}(x_{1,1})^3 + (x_{1,2})^3 - \textstyle\sum_{j=1}^{2} x_{1,j} \cdot \sum_{j=1}^{2} x_{2,j}, \\ f_2(x) &= (x_{1,1} + x_{1,2})(x_{2,1})^3 - 3x_{2,1} + (x_{2,2})^2 + x_{1,1}x_{1,2}x_{2,2}. \end{aligned}$$

We use parametric expressions as in (4.1). For each $i$, the $q_i(x) > 0$ for all $x \in X$. By Algorithm 5.1, we got the GNE $u = (u_1, u_2)$ with

$$u_1 \approx (0.6475, 0.2786), \quad u_2 \approx (1.0391, -0.0902).$$

It took around 63.97 seconds.

***Example 6.6*** Consider the 2-player GNEP

$$\left.\begin{array}{cl} \min\limits_{x_1 \in \mathbb{R}^2} & (x_{1,1})^2 + 2(x_{1,2})^2 \\ & \qquad +3\sum_{j=1}^{2} x_{1,j}(x_{2,j})^2 \\ s.t. & x_{1,1} + 2x_{1,2} - x_{2,1} \le 1, \\ & (x_{1,2})^2 + (x_{2,1})^2 \le 3, \\ & x_{1,1} \ge 0, \end{array}\right| \begin{array}{cl} \min\limits_{x_2 \in \mathbb{R}^2} & \|x_1\|^2 \cdot \|x_2\|^2 + 3x_1^T x_2 \\ & \qquad +x_{2,1} - x_{2,2} \\ s.t. & (x_{2,1})^2 + x_{1,2}x_{2,1} \le 2, \\ & (x_{1,1})^2 + (x_{2,2})^2 \le 3, \\ & x_{2,2} \ge 0. \end{array}$$

We solve (3.9) numerically for $i = 1, 2$ with $v = (0, 0, 0, 0)$, $d = 2$ to get rational expressions for $\lambda_i$'s. By Algorithm 5.1, we got the GNE $u = (u_1, u_2)$ with

$$\begin{array}{ll} u_1 \approx (0.0000, -1.3758), & q_1(u) \approx 6.7538; \\ u_2 \approx (-0.2641, 1.3544), & q_2(u) \approx 2.3227. \end{array}$$

It took around 0.41 second in solving (3.9) for both players, and 6.40 seconds to find the GNE. For neatness of the paper, we do not display Lagrange multiplier expressions obtained by solving (3.9).

***Example 6.7*** (i) Consider the 3-player GNEP

$$\text{1st player:} \begin{cases} \min\limits_{x_1 \in \mathbb{R}^2} & x_{2,1}(x_{1,1})^2 + x_{2,2}(x_{1,2})^2 - (x_{3,1})^2 x_{1,1} - (x_{3,2})^2 x_{1,2} \\ s.t. & x_1^T x_1 \le 1 + x_2^T x_2; \end{cases}$$

$$\text{2nd player:} \begin{cases} \min\limits_{x_2 \in \mathbb{R}^2} & (x_{2,1})^3 + (x_{2,2})^3 - x_{1,1}x_{2,1}x_{3,1} - x_{1,2}x_{2,2}x_{3,2} \\ s.t. & x_{2,1} + x_{2,2} \le 1 + x_3^T x_3, \ x_{2,1} \ge 0, \ x_{2,2} \ge 0; \end{cases}$$

$$\text{3rd player:} \begin{cases} \min\limits_{x_3 \in \mathbb{R}^2} & \left(\sum_{i=1}^{3}(x_{i,1} + x_{i,2})\right)^2 - x_{3,1} - x_{3,2} \\ s.t. & x_{3,1} \ge x_{1,1}, \ x_{3,2} \ge x_{1,2}. \end{cases}$$

The first player's Lagrange multipliers have a rational expression, that

$$\lambda_1 = \frac{-x_1^T \nabla_{x_1} f_1}{2q_1(x)}, \quad q_1(x) = 1 + x_2^T x_2.$$

For the second player, we let $q_2(x) = 1 + x_3^T x_3$, and there exists a rational expression for $\lambda_2$, that

$$\lambda_{2,1} = \frac{-x_2^T \nabla_{x_2} f_2}{q_2(x)}, \quad \lambda_{2,2} = \frac{\partial f_2}{\partial x_{2,1}} + \lambda_{2,1}, \quad \lambda_{2,3} = \frac{\partial f_2}{\partial x_{2,2}} + \lambda_{2,1}.$$

For $\lambda_3$, we use the polynomial expression that

$$\lambda_{3,1} = \frac{\partial f_3}{\partial x_{3,1}}, \quad \lambda_{3,2} = \frac{\partial f_3}{\partial x_{3,2}}.$$

For each $i$, the $q_i(x) > 0$ for all $x \in X$. By Algorithm 5.1, we got the GNE $u = (u_1, u_2, u_3)$ with

$$u_1 \approx (0.1097, 0.0750), \; u_2 \approx (0.0663, 0.0458), \; u_3 \approx (0.1205, 0.0828).$$

It took around 3.23 seconds.
(ii) If the third player's objective function becomes

$$\big(\textstyle\sum_{i=1}^{3}(x_{i,1} - x_{i,2})\big)^2 - x_{3,1} - x_{3,2},$$

then Algorithm 5.1 took around 2.86 seconds to detect nonexistence of GNEs. Note that all the matrix polynomials $G_i(x)\,(i = 1, \dots, 3)$ are nonsingular on $X$, so all GNEs must be KKT points if they exist.

***Example 6.8*** [18, Example A.3] Consider the GNEP of 3 players. For $i = 1, 2, 3$, the $i$th player aims to minimize the quadratic function

$$f_i(x) = \frac{1}{2} x_i^T A_i x_i + x_i^T (B_i x_{-i} + b_i).$$

All variables have box constraints $-10 \le x_{i,j} \le 10$, for all $i, j$. In addition to them, the first player has linear constraints $x_{1,1} + x_{1,2} + x_{1,3} \le 20$, $x_{1,1} + x_{1,2} - x_{1,3} \le x_{2,1} - x_{3,2} + 5$; the second player has $x_{2,1} - x_{2,2} \le x_{1,2} + x_{1,3} - x_{3,1} + 7$; and the third player has $x_{3,2} \le x_{1,1} + x_{1,3} - x_{2,1} + 4$. The values of parameters are set as follows

$$A_1 = \begin{bmatrix} 20 & 5 & 3 \\ 5 & 5 & -5 \\ 3 & -5 & 15 \end{bmatrix}, \; A_2 = \begin{bmatrix} 11 & -1 \\ -1 & 9 \end{bmatrix}, \; A_3 = \begin{bmatrix} 48 & 39 \\ 39 & 53 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} -6 & 10 & 11 & 20 \\ 10 & -4 & -17 & 9 \\ 15 & 8 & -22 & 21 \end{bmatrix}, \; B_2 = \begin{bmatrix} 20 & 1 & -3 & 12 & 1 \\ 10 & -4 & 8 & 16 & 21 \end{bmatrix},$$

$$B_3 = \begin{bmatrix} 10 & -2 & 22 & 12 & 16 \\ 9 & 19 & 21 & -4 & 20 \end{bmatrix}, \; b_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \; b_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \; b_3 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}.$$

We use parametric expressions for Lagrange multipliers as in (4.3). It is clear $q_i(x) = 1$ for all $x \in X$ and for all $i = 1, 2, 3$. By Algorithm 5.1, we got the GNE $u = (u_1, u_2, u_3)$ with

$$u_1 \approx (-0.3805, -0.1227, -0.9932), \quad u_2 \approx (0.3903, 1.1638),$$
$$u_3 \approx (0.0504, 0.0176).$$

It took around 8.18 seconds.

***Example 6.9*** Consider the GNEP based on the Arrow and Debreu model of a competitive economy [4,18]. The first $N_1$ players are consumers, the second $N_2$ players are production units, and the last player is the market, so $N = N_1 + N_2 + 1$. In this GNEP, each player has $n_1 = \cdots = n_N$ variables. Let $Q_i \in \mathbb{R}^{n_i \times n_i}$, $b_i \in \mathbb{R}^{n_i}$, $\xi_i \in \mathbb{R}^{n_i}_+$ and $a_{i,k} \in \mathbb{R}_+$ be parameters. These players' optimization problems are:

$$\text{The } i\text{th player (a consumer): } \begin{cases} \min\limits_{x_i \in \mathbb{R}^{n_i}_+} & \frac{1}{2} x_i^T Q_i x_i - b_i^T x_i \\ s.t. & x_N^T x_i \le x_N^T \xi_i + \sum_{k=N_1+1}^{N-1} a_{i,k} x_N^T x_k. \end{cases}$$

$$\text{The } i\text{th player (a production unit): } \begin{cases} \min\limits_{x_i \in \mathbb{R}^{n_i}_+} & -x_N^T x_i \\ s.t. & x_i^T x_i \le i - N_1. \end{cases}$$

$$\text{The } N\text{th player (the market): } \begin{cases} \min\limits_{x_N \in \mathbb{R}^{n_i}_+} & x_N^T \left( \sum_{k=N_1+1}^{N-1} x_k - \sum_{k=1}^{N_1} (x_k - \xi_k) \right) \\ s.t. & \sum_{j=1}^{n_i} x_{N,j} = 1. \end{cases}$$

For each $i \in [N_1]$, the Lagrange multipliers have rational expressions as

$$\lambda_{i,1} = \frac{-x_i^T \nabla_{x_i} f_i}{q_i(x)}, \quad \lambda_{i,j} = \frac{\partial f_i}{\partial x_{i,j}} + x_{N,j} \cdot \lambda_{i,1} \; (j = 1, \ldots, n_i),$$

where $q_i(x) = x_N^T \xi_i + \sum_{k=N_1+1}^{N-1} a_{i,k} x_N^T x_k > 0$ for all $x \in X$. For each $i = N_1 + 1, \ldots, N_1 + N_2$, the $i$th player (a production unit) has polynomial expressions

$$\lambda_{i,1} = \frac{-x_i^T \nabla_{x_i} f_i}{2(i - N_1)}, \quad \lambda_{i,j} = \frac{\partial f_i}{\partial x_{i,j}} + 2x_{i,j} \cdot \lambda_{i,1} \; (j = 1, \ldots, n_i).$$

For the last player (the market), we substitute $x_{N,n_i}$ by $1 - \sum_{j=1}^{n_i-1} x_{N,j}$, then the constraints become $1 - \sum_{j=1}^{n_i-1} x_{N,j} \ge 0$, $x_{N,1} \ge 0, \ldots, x_{N,n_i-1} \ge 0$, and hence

$$\lambda_{N,1} = -\sum_{j=1}^{n_i-1} \frac{\partial f_N}{\partial x_{N,j}} \cdot x_{N,j}, \quad \lambda_{N,j+1} = \frac{\partial f_N}{\partial x_{N,j}} + \lambda_{N,1} \; (j = 1, \ldots, n_i - 1).$$

For each $i = 1, \ldots, N_1$, when $n_i = 2$, the parameters are given as

$$Q_i = \begin{bmatrix} 0.75 + 0.25i & 1.5 - 0.5i \\ 1.5 - 0.5i & i \end{bmatrix}, \quad b_i = \begin{bmatrix} 0.4 + 0.1i \\ 0.9 + 0.1i \end{bmatrix}, \quad \xi_i = \begin{bmatrix} i \\ i \end{bmatrix},$$
$$a_{i,j} = 0.2 + 0.1i \quad (j = N_1 + 1, \ldots, N_1 + N_2).$$

When $n_i = 3$, the parameters are given as:

$$Q_i = \begin{bmatrix} -1+2i & -i & i \\ -i & 1+i & 1-i \\ i & 1-i & 1+i \end{bmatrix}, \quad b_i = \begin{bmatrix} 0.4+0.1i \\ 0.9+0.1i \\ 1.4+0.1i \end{bmatrix}, \quad \xi_i = \begin{bmatrix} i \\ i \\ i \end{bmatrix},$$
$$a_{i,j} = 0.2 + 0.1i \quad (j = N_1+1, \ldots, N_1+N_2).$$

The numerical results are presented in Table 3. The "$N$" is the total number of all players, the "$N_1$" and "$N_2$" are the number of consumers and production units respectively, the "$n$" (resp., "$n_i$") is the dimension of "$x$" (resp., "$x_i$"), the "$u$" is the GNE obtained by Algorithm 5.1, the "$q(u)$" gives the value of the denominator vector $q(u) := (q_1(u), \ldots, q_{N_1}(u))$, and "time" shows the consumed time (in seconds).

### 6.1 Comparison with other methods

We compare our method (i.e., Algorithm 5.1) with some classical methods for solving convex GNEPPs, such as the two-step method in [23] based on Quasi-variational formulation, the penalty method in [18], the exact version of interior point method based on the KKT system in [13], and the Augmented-Lagrangian method in [28]. All examples in Sect. 6 are tested for comparisions. For Example 6.9, we test for the case that $N_1 = N_2 = 1, n_i = 3$.

For a computed tuple $u := (u_1, \ldots, u_N)$, we use the value

$$\xi := \max\Big\{ \max_{i\in[N], j\in\mathcal{I}_i} \{-g_{i,j}(u)\}, \max_{i\in[N], j\in\mathcal{E}_i} \{|g_{i,j}(u)|\}\Big\}$$

to measure the feasibility violation. Clearly, the point $u$ is feasible if and only if $\xi \le 0$. If we solve (5.2) for all $i \in [N]$, the accuracy parameter of $u$ is $\delta := \max_{i\in[N]} |\delta_i|$. For these methods, we use the following stopping criterion: For each time we get a new iterate $u$, if its feasibility violation $\xi < 10^{-6}$, then we compute the accuracy parameter $\delta$. If $\delta < 10^{-6}$, then we stop the iteration.

For these classical methods, the parameters are the same as given in [13,18,23, 28,57]. When implementing the QVI method, we use Moment-SOS relaxations to find projections into given sets (the maximum number of iterations for line search is set to be 100). For the penalty method, the MATLAB function `fsolve` is used to implement the Levenberg-Marquardt Algorithm for solving all equations involved (the maximum number of iterations is set to be 100). The full penalization is used when we implement the Augmented-Lagrangian method, and a Levenberg-Marquardt type method (see [28, Algorithm 24]) is exploited to solve penalized subproblems. We let 1000 be the maximum number of iterations for the QVI method, let 1000 be the maximum number of outer iterations for the penalty method and the Augmented-Lagrangian method, and let 10, 000 be the maximum number of iterations for the interior point method. For initial points, we use $(1, 0, 0, 1, 0, 0)$ for Example 6.1(i-ii), $(0, 0, 0, 0, 0, 0, 0, 0, 1)$ for Example 6.9, and the zero vectors for other GNEPs. If the maximum number of iterations is reached but the stopping criterion is not met, we still solve the (5.2) to check if the latest iterating point is a GNE or not.

**Table 3** Numerical results of Example 6.9

| Number of players | Dimension | $u$ | $q(u)$ | Time |
|---|---|---|---|---|
| $N=5$<br>$N_1=2$<br>$N_2=2$ | $n=10$<br>$n_i=2$ | (0.0000, 1.0000, 0.2889, 0.4778,<br>0.4166, 0.9091, 0.5892, 1.2856,<br>0.3143, 0.6857) | 1.4907<br>2.6543 | 1.37 |
| $N=6$<br>$N_1=3$<br>$N_2=2$ | $n=12$<br>$n_i=2$ | (0.0000, 1.0000, 0.2889, 0.4778,<br>0.4667, 0.4000, 0.4354, 0.9002,<br>0.6157, 1.2731, 0.3260, 0.6740) | 1.5423<br>2.7230<br>3.9038 | 3.82 |
| $N=7$<br>$N_1=3$<br>$N_2=3$ | $n=14$<br>$n_i=2$ | (0.0000, 1.0000, 0.2889, 0.4778,<br>0.4667, 0.4000, 0.5587, 0.8294,<br>0.7901, 1.1729, 0.9677, 1.4365,<br>0.4025, 0.5975) | 1.8961<br>3.1948<br>4.4935 | 21.26 |
| $N=8$<br>$N_1=4$<br>$N_2=3$ | $n=16$<br>$n_i=2$ | (0.0000, 1.0000, 0.2889, 0.4778,<br>0.4667, 0.4000, 0.5704, 0.3963,<br>0.5835, 0.8121, 0.8251, 1.1485,<br>1.0106, 1.4067, 0.4181, 0.5819) | 1.8913<br>3.1884<br>4.4855<br>5.7826 | 106.78 |
| $N=9$<br>$N_1=4$<br>$N_2=4$ | $n=18$<br>$n_i=2$ | (0.0000, 1.0000, 0.2889, 0.4778,<br>0.4667, 0.4000, 0.5704, 0.3963,<br>0.6258, 0.7800, 0.8850, 1.1031,<br>1.0838, 1.3510, 1.2515, 1.5600,<br>0.4451, 0.5549) | 2.3116<br>3.7489<br>5.1861<br>6.6233 | 465.71 |
| $N=3$<br>$N_1=1$<br>$N_2=1$ | $n=9$<br>$n_i=3$ | (1.3076, 1.0871, 0.0962,<br>0.8087, 0.5882, 0.0000,<br>0.5789, 0.4211, 0.0000) | 1.2148 | 0.50 |
| $N=4$<br>$N_1=2$<br>$N_2=1$ | $n=12$<br>$n_i=3$ | (1.3696, 1.0886, 0.0652,<br>0.3500, 0.7875, 0.5625,<br>0.6245, 0.7810, 0.0000,<br>0.4443, 0.5557, 0.0000) | 1.2134<br>2.2846 | 3.76 |
| $N=5$<br>$N_1=2$<br>$N_2=2$ | $n=15$<br>$n_i=3$ | (1.7172, 1.3109, 0.0000,<br>0.3500, 0.7875, 0.5625,<br>0.7006, 0.7135, 0.0097,<br>0.9908, 1.0091, 0.0006,<br>0.4953, 0.5044, 0.0003) | 1.5121<br>2.6829 | 42.66 |
| $N=6$<br>$N_1=3$<br>$N_2=2$ | $n=18$<br>$n_i=3$ | (1.7734, 1.3398, 0.0000,<br>0.3500, 0.7875, 0.5625,<br>0.2250, 0.7958, 0.6542,<br>0.5780, 0.8160, 0.0001,<br>0.8174, 1.1541, 0.0040,<br>0.4146, 0.5854, 0.0000) | 1.5192<br>2.6923<br>3.8653 | 473.84 |

The numerical results are presented in Table 4, and the comparison is summarized in the following.

1. The QVI method failed to find a GNE for Example 6.1(i), because the projection set in Step 2 is empty. Therefore the line-search could not finish (see [23, Algorithm 4.1]). This is also the case for Examples 6.1(ii) and 6.7(ii), for which the GNEs do not exist. For Examples 6.2 and 6.4, the sequence generated by QVI is

**Table 4** Comparison with some methods

| Example | | QVI | Penalty | IPM | A-L | Alogrithm 5.1 |
|---|---|---|---|---|---|---|
| 6.1(i) | Time | Fail | Fail | Fail | Fail | 2.83 |
| | Error | | | | | $4 \times 10^{-9}$ |
| 6.1(ii) | Time | Fail | Fail | Fail | Fail | 70.31 |
| | Error | | | | | no GNE |
| 6.2 | Time | Fail | 3.45 | 0.19 | Fail | 0.89 |
| | Error | | $2 \times 10^{-6}$ | $3 \times 10^{-7}$ | | $7 \times 10^{-7}$ |
| 6.3 | Time | 2.63 | 8.46 | 0.12 | 0.08 | 0.20 |
| | Error | $8 \times 10^{-7}$ | $3 \times 10^{-6}$ | $2 \times 10^{-7}$ | $2 \times 10^{-7}$ | $1 \times 10^{-8}$ |
| 6.4 | Time | Fail | 4.51 | 0.29 | Fail | 2.03 |
| | Error | | $3 \times 10^{-5}$ | $8 \times 10^{-7}$ | | $4 \times 10^{-7}$ |
| 6.5 | Time | 185.29 | 4.02 | 37.7 | 0.03 | 63.97 |
| | Error | $9 \times 10^{-5}$ | $2 \times 10^{-6}$ | $5 \times 10^{-4}$ | $3 \times 10^{-7}$ | $4 \times 10^{-7}$ |
| 6.6 | Time | 7.78 | Fail | 0.17 | Fail | 6.40 |
| | Error | $6 \times 10^{-7}$ | | $3 \times 10^{-7}$ | | $1 \times 10^{-7}$ |
| 6.7(i) | Time | 72.18 | 0.39 | 0.16 | 0.05 | 3.23 |
| | Error | $4 \times 10^{-7}$ | $8 \times 10^{-8}$ | $5 \times 10^{-7}$ | $1 \times 10^{-10}$ | $7 \times 10^{-9}$ |
| 6.7(ii) | Time | Fail | Fail | Fail | Fail | 2.86 |
| | Error | | | | | no GNE |
| 6.8 | Time | Fail | 0.38 | 0.16 | 0.01 | 8.18 |
| | Error | | $9 \times 10^{-8}$ | $1 \times 10^{-8}$ | $1 \times 10^{-8}$ | $3 \times 10^{-8}$ |
| 6.9 | Time | 1.223 | 6.26 | 0.14 | Fail | 0.50 |
| | Error | $3 \times 10^{-5}$ | $8 \times 10^{-6}$ | $3 \times 10^{-7}$ | | $7 \times 10^{-7}$ |

alternating between several points and none of them is a GNE. For Example 6.8, the sequence does not converge.

2. The penalty method failed to find a GNE for Examples 6.1(i) and 6.6, because the equation $F_{\varepsilon_k}(x) = 0$ cannot be solved for some $k$ (see [18, Algorithm 3.3]). This is also the case for Examples 6.1(ii) and 6.7(ii), for which the GNEs do not exist.
3. The interior-point method failed to find a GNE for Examples 6.1(i), 6.1(ii) and 6.7(ii), because the step-length is too small to efficiently decrease the violation of KKT conditions. Note that for Examples 6.1(ii) and 6.7(ii), the GNEs do not exist, so the Newton type directions usually do not satisfy the sufficient descent conditions.
4. The Augmented-Lagrangian method failed to find a GNE for Example 6.1(i), because the maximum penalty parameter ($10^{12}$) is reached before a GNE is obtained. This is also the case for Example 6.1(ii), for which the GNEs do not exist. For Examples 6.2, 6.4, 6.6, 6.7(ii) and 6.9, the Augmented-Lagrangian method failed to find a GNE, because the penalization subproblems cannot be efficiently solved.

## 7 Conclusions and discussions

This paper studies convex GNEPs given by polynomials. The rational and parametric expressions for Lagrange multipliers are used. Based on these expressions, Algorithms 5.1 is proposed for computing a GNE. The Moment-SOS hierarchy of semidefinite relaxations are used to solve the appearing polynomial optimization problems. Under some general assumptions, we show that Algorithm 5.1 is able to find a GNE if there exists one, or detect nonexistence of GNEs if there is none.

For future work, it is interesting to solve nonconvex GNEPPs. Under some constraint qualifications, the KKT system (2.5) is necessary but not sufficient for GNEs. A solution $u$ of (2.5) may not be a GNE for nonconvex GNEPPs. If $u$ is not a GNE, one needs to find an efficient method to obtain a different candidate. Such a method is proposed for solving NEPs [48]. For GNEPs, it is not clear how to generalize the method in [48]. When the point $u$ is not a GNE, how can we exclude it and find a better candidate? When (5.1) is feasible, how do we detect nonexistence of GNEs? These questions are mostly open, to the best of the authors' knowledge.

**Acknowledgements** The authors would like to thank Christian Kanzow and Daniel Steck for sharing the code for solving GNEPs. They also thank the editors and anonymous referees for fruitful suggestions.

## References

1. Anselmi, J., Ardagna, D., Passacantando, M.: Generalized nash equilibria for saas/paas clouds. Eur. J. Oper. Res. **236**(1), 326–339 (2014)
2. Ahmadi, A.A., Zhang, J.: Semidefinite programming and Nash equilibria in bimatrix games. INFORMS J. Comput. **33**(2), 607–628 (2020)
3. Ardagna, D., Ciavotta, M., Passacantando, M.: Generalized Nash equilibria for the service provisioning problem in multi-cloud systems. IEEE Trans. Serv. Comput. **10**, 381–395 (2017)
4. Arrow, K., Debreu, G.: Existence of an equilibrium for a competitive economy. Econom. J. Econom. Soc. **22**, 265–290 (1954)
5. Ba, Q., Pang, J.: Exact penalization of generalized Nash equilibrium problems. Oper. Res. (2020). https://doi.org/10.1287/opre.2019.1942
6. Belousov, E.G., Klatte, D.: A Frank-Wolfe type theorem for convex polynomial programs. Comput. Optim. Appl. **22**(1), 37–48 (2002)
7. Bertsekas, D.: Nonlinear Programming, 2nd edn. Athena Scientific (1995)
8. Breton, M., Zaccour, G., Zahaf, M.: A game-theoretic formulation of joint implementation of environmental projects. Eur. J. Oper. Res. **168**, 221–239 (2006)
9. De Klerk, E., Laurent, M.: On the Lasserre hierarchy of semidefinite programming relaxations of convex polynomial optimization problems. SIAM J. Optim. **21**(3), 824–832 (2011)
10. Debreu, G.: A social equilibrium existence theorem. Proc. Natl. Acad. Sci. **38**, 886–893 (1952)
11. Demmel, J.: Applied Numerical Linear Algebra. SIAM (1997)

12. Dreves, A., Facchinei, F., Fischer, A., Herrich, M.: A new error bound result for Generalized Nash Equilibrium Problems and its algorithmic application. Comput. Optim. Appl. **59**, 63–84 (2014)
13. Dreves, A., Facchinei, F., Kanzow, C., Sagratella, S.: On the solution of the KKT conditions of Generalized Nash Equilibrium Problems. SIAM J. Optim. **21**, 1082–1108 (2011)
14. Dreves, A., Kanzow, C., Stein, O.: Nonsmooth optimization reformulations of player convex generalized Nash equilibrium problems. J. Glob. Optim. **53**(4), 587–614 (2012)
15. Facchinei, F., Fischer, A., Piccialli, V.: On generalized nash games and variational inequalities. Oper. Res. Lett. **35**, 159–164 (2007)
16. Facchinei, F., Fischer, A., Piccialli, V.: Generalized Nash Equilibrium Problems and Newton methods. Math. Program. **117**, 163–194 (2009)
17. Facchinei, F., Kanzow, C.: Generalized Nash equilibrium problems. Ann. Oper. Res. **175**(1), 177–211 (2010)
18. Facchinei, F., Kanzow, C.: Penalty methods for the solution of Generalized Nash Equilibrium problems. SIAM J. Optim. **20**, 2228–2253 (2010)
19. Facchinei, F., Lampariello, L.: Partial penalization for the solution of Generalized Nash Equilibrium Problems. J. Glob. Optim. **50**(1), 39–57 (2011)
20. Facchinei, F., Pang, J.: Nash equilibria: the variational approach. In: Palomar, D., Eldar, Y. (eds.) Convex Optimization in Signal Processing and Communications, pp. 443–493. Cambridge University Press, England (2010)
21. Fialkow, L., Nie, J.: The truncated moment problem via homogenization and flat extensions. J. Funct. Anal. **263**(6), 1682–1700 (2012)
22. Fukushima, M.: Restricted generalized Nash equilibria and controlled penalty algorithm. Comput. Manag. Sci. **8**, 201–208 (2010)
23. Han, D., Zhang, H., Qian, G., Xu, L.: An improved two-step method for solving Generalized Nash Equilibrium Problems. Eur. J. Oper. Res. **216**(3), 613–623 (2012)
24. Harker, P.: Generalized nash games and quasi-variational inequalities. Eur. J. Oper. Res. **54**, 81–94 (1991)
25. Helton, J.W., Nie, J.: Semidefinite representation of convex sets. Math. Program. **122**(1), 21–64 (2010)
26. Henrion, D., Lasserre, J.: Detecting global optimality and extracting solutions in GloptiPoly, Positive polynomials in control, 293.C310, Lecture Notes in Control and Inform. Sci., 312, Springer, Berlin (2005)
27. Henrion, D., Lasserre, J., Löfberg, J.: Gloptipoly 3: moments, optimization and semidefinite programming. Optim. Methods Softw. **24**(4–5), 761–779 (2009)
28. Kanzow, C., Steck, D.: Augmented Lagrangian methods for the solution of Generalized Nash Equilibrium Problems. SIAM J. Optim. **26**, 2034–2058 (2016)
29. Lasserre, J.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**, 796–817 (2001)
30. Lasserre, J.: Convexity in semialgebraic geometry and polynomial optimization. SIAM J. Optim. **19**(4), 1995–2014 (2009)
31. Lasserre, J.: An Introduction to Polynomial and Semi-algebraic Optimization, vol. 52. Cambridge University Press (2015)
32. Lasserre, J.: The Moment-SOS Hierarchy. In: Sirakov, B., Ney de Souza., P., Viana, M. (eds.) Proceedings of the International Congress of Mathematicians (ICM 2018), vol. 3, pp. 3761–3784, World Scientific (2019)
33. Lasserre, J., Laurent, M., Rostalski, P.: Semidefinite characterization and computation of zero-dimensional real radical ideals. Found. Comput. Math. **8**(5), 607–647 (2008)
34. Laurent, J.: Sums of Squares, Moment Matrices and Optimization Over Polynomials, Emerging Applications of Algebraic Geometry of IMA Volumes in Mathematics and its Applications, 149: 157–270. Springer (2009)
35. Laurent, M.: Optimization over polynomials: Selected topics. In: Jang, S., Kim, Y., Lee, D.W., Yie, I. (eds.) Proceedings of the International Congress of Mathematicians, ICM 2014, pp. 843-869 (2014)
36. Nabetani, K., Tseng, P., Fukushima, M.: Parametrized variational inequality approaches to Generalized Nash Equilibrium Problems with shared constraints. Comput. Optim. Appl. **48**, 423–452 (2011)
37. Nie, J., Sturmfels, B.: Matrix cubes parameterized by eigenvalues. SIAM J. Matrix Anal. Appl. **31**(2), 755–766 (2009)
38. Nie, J.: Certifying convergence of Lasserre‘s hierarchy via flat truncation. Math. Program. **142**(1–2), 485–510 (2013)

39. Nie, J.: Polynomial optimization with real varieties. SIAM J. Optim. **23**(3), 1634–1646 (2013)
40. Nie, J.: Optimality conditions and finite convergence of Lasserre's hierarchy. Math. Program. **146**(1–2), 97–121 (2014)
41. Nie, J.: The $\mathcal{A}$-Truncated $\mathcal{K}$-Moment Problem. Found. Comput. Math. **14**(6), 1243–1276 (2014)
42. Nie, J.: The hierarchy of local minimums in polynomial optimization. Math. Program. **151**(2), 555–583 (2015)
43. Nie, J.: Linear optimization with cones of moments and nonnegative polynomials. Math. Program. **153**(1), 247–274 (2013)
44. Nie, J.: Generating polynomials and symmetric tensor decompositions. Found. Comput. Math. **17**(2), 423–465 (2017)
45. Nie, J.: Low rank symmetric tensor approximations. SIAM J. Matrix Anal. Appl. **38**(4), 1517–1540 (2017)
46. Nie, J.: Tight relaxations for polynomial optimization and Lagrange multiplier expressions. Math. Program. **178**(1–2), 1–37 (2019)
47. Nie, J., Tang, X., Xu, L.: The Gauss-Seidel method for generalized Nash equilibrium problems of polynomials. Comput. Optim. Appl. **78**(2), 529–557 (2021)
48. Nie, J., Tang, X.: Nash equilibrium problems of polynomials, Preprint (2020). arXiv:2006.09490
49. Nie, J., Wang, L., Ye, J., Zhong, S.: A Lagrange multiplier expression method for bilevel polynomial optimization. SIAM J. Optim. **31**(3), 2368–2395 (2021)
50. Nie, J., Yang, Z., Zhou, G.: The saddle point problem of polynomials. Found. Comput. Math. (2021). https://doi.org/10.1007/s10208-021-09526-8
51. Pang, J., Fukushima, M.: Quasi-variational inequalities, generalized nash equilibria, and multi-leader-follower games. Comput. Manag. Sci. **2**, 21–56 (2005)
52. Pang, J., Scutari, G., Facchinei, F., Wang, C.: Distributed power allocation with rate constraints in Gaussian parallel interference channels. IEEE Trans. Inf. Theory **54**(8), 3471–3489 (2008)
53. Putinar, M.: Positive polynomials on compact semi-algebraic sets. Indiana Univ. Math. J. **42**(3), 969–984 (1993)
54. Schiro, D., Pang, J., Shanbhag, U.: On the solution of affine generalized Nash equilibrium problems with shared constraints by Lemke's method. Math. Program. **142**(1), 1–46 (2013)
55. Schweighofer, M.: Optimization of polynomials on compact semialgebraic sets. SIAM J. Optim. **15**(3), 805–825 (2005)
56. Sturm, J.: Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. Optim. Methods Softw. **11**(1–4), 625–653 (1999)
57. von Heusinger, A., Kanzow, C.: Relaxation methods for Generalized Nash Equilibrium Problems with inexact line search. J. Optim. Theory Appl. **143**, 159–183 (2009)
58. von Heusinger, A., Kanzow, C.: Optimization reformulations of the Generalized Nash Equilibrium Problem using Nikaido-Isoda-type functions. Comput. Optim. Appl. **43**, 353–377 (2009)