Distributed Nash Equilibrium Seeking for Generalized Convex Games with Shared Constraints

Chao Sun¹ and Guoqiang Hu¹

¹ School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798.

E-mail: gqhu@ntu.edu.sg.

Abstract. In this paper, we deal with the problem of finding a Nash equilibrium for a generalized convex game. Each player is associated with a convex cost function and multiple shared constraints. Supposing that each player can exchange information with its neighbors via a connected undirected graph, the objective of this paper is to design a Nash equilibrium seeking law such that each agent minimizes its objective function in a distributed way. Consensus and singular perturbation theories are used to prove the stability of the system. A numerical example is given to show the effectiveness of the proposed algorithms.

1. Introduction

How to find a Nash equilibrium is an interesting and important problem for non-cooperative games [1]. In [2], an iterative steepest descent algorithm was proposed for numerical approximation of local Nash equilibria. In [3], an adaptive learning technique was used to iteratively compute the Nash equilibrium of a multi-player game. Extremum seeking based methods were proposed to search Nash equilibrium [4–6].

Compared with the conventional Nash equilibrium problem, the generalized Nash games assume that each player’s feasible set can be constrained by the rival players’ strategies [7–9]. A generalized convex game usually has continuous strategy spaces and the actions are coupled through both objective functions and constraints. The problem was first formally proposed by [10], followed by [11] studying economic equilibria. Since then, there have been many studies reported on the existence of a generalized Nash equilibrium, and algorithms to compute it. For instance, a method involving variational inequalities was presented in [12].

Most of the aforementioned literature require that each player can obtain the strategies of its opponents. In practice, especially in some multi-agent networks, the agent might have limited interaction (communication) with other agents. Recently, distributed computation algorithms have attracted much attention due to their reliability, security and low communication burden. For example, [13–16] addressed the distributed optimization problem using saddle point dynamics methods and [17] investigated a distributed time-varying optimization problem for quadratic objective functions. The recently proposed algorithm in [18] solved a distributed Nash equilibrium seeking problem for unconstrained non-cooperative games based on average consensus and singular perturbation theory. In [19], discrete-time adaptive algorithms were presented to solve Nash equilibrium seeking problems, the objectives and constraints are required to be neighbor-coupled.

In this paper, we present a continuous-time distributed algorithm to seek a Nash equilibrium for a generalized convex game with shared constraints. The main contributions of this paper are summarized as follows: 1) A distributed algorithm to find the normalized Nash equilibrium of a
generalized convex game with class-$C^2$ objective functions is presented. Compared with the previous work [18], constraints relying on other players' strategies are considered. Compared with [19], the objective functions and constraints can be coupled arbitrarily; 2) The convergence of the players' actions to the normalized Nash equilibrium is analyzed, by using singular perturbation based techniques, it is proven that the proposed algorithms converge into a neighborhood of the Nash equilibrium and the error bound can be arbitrarily small by selecting the control parameters.

The rest of this paper is organized as follows: In Section 2, notations and background information on graph theory are given. In Section 3, the distributed Nash equilibrium seeking problem is formulated mathematically. In Section 4, a distributed algorithm is designed, and convergence analysis is given. Section 5 gives a numerical example to illustrate the effectiveness of the proposed algorithms. Finally, Section 6 concludes the paper.

2. Notations and Preliminaries
Throughout this paper, $R$ ($R^{\mathbb{N}}$), $R^n$ and $R^{n \times n}$ denote the set of reals (nonnegative reals), $n$-dimensional column vectors, and $n \times n$ matrices, respectively. I and 0 are column vectors with appropriate dimension. The symbol $[a_i] \in R^N$ represents a column vector defined as $[a_i]=[a_{i1},...,a_{iN}]^T$. A function $f: [0,a) \rightarrow [0,\infty)$ is said to belong to class-$K$ if it is continuous, zero at zero, and strictly increasing. It is said to belong to class-$K_\infty$ if it belongs to class-$K$, $a=\infty$ and $\lim f(r)=\infty$. A function $f(x)$ is said to belong to class-$C^2$ if the derivatives $f^{(1)}(x), f^{(2)}(x)$ exist and are continuous.

Let $G=\{V,E\}$ denote an undirected graph, where $V=\{1,\ldots,N\}$ indicates the vertex set and $E \subseteq V \times V$ indicates the edge set. $N_i=\{j \in V \mid (j,i) \in E\}$ denotes the neighborhood set of vertex $i$. A path is referred by the sequence of its vertices. Path $P$ between $v_0$ and $v_k$ is the sequence $\{v_0,v_1,\ldots,v_k\}$ where $(v_{i-1},v_i) \in E$ for $i=1,\ldots,k$ and the vertices are distinct. The number $k$ is defined as the length of path $P$. Graph $G$ is connected if for any two vertices, there is a path in $G$. A matrix $A=[a_{ij}] \in R^{N \times N}$ denotes the adjacency matrix of $G$, where $a_{ij} > 0$ if and only if $(j,i) \in E$ else $a_{ij} = 0$. In this paper, we suppose $d_{ii} = 0$. A matrix $L \cdot D - A \in R^{N \times N}$ is called the Laplacian matrix of $G$, where $D=\text{diag}(d_{ij}) \in R^{N \times N}$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{N} a_{ij}$.

3. Problem Formulation
Consider the set of players $V \cap \{1,\ldots,N\}$ where the action of player $i$ is denoted as $x_i \in R$. Each player $i$ minimizes its objective function $f_i(x): R^N \rightarrow R$ where $x=[x_1,\cdots,x_N]^T \in R^N$ is the action vector of $N$ players. Suppose that each player is subject to $k$ shared constraints that depend on the players' actions, i.e., $g_j(x) \leq 0$, $j \in K \cap \{1,\ldots,K\}$. If agent $j$ is not a neighbor of agent $i$, then player $i$ has no access to player $j$'s action directly. Otherwise, player $i$ can get the information of player $j$ via a connected undirected graph topology. Our objective is to design a distributed Nash equilibrium seeking law for the players such that their actions converge to a Nash equilibrium of the game.

Denote $x_{-i}=[x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_N]^T \in R^{N-1}$. Then, for player $i \in V$, its optimization problem can be described as

$$\min_{x_i} f_i(x_i,x_{-i}), \text{ such that } g_j(x_i,x_{-i}) \leq 0, \quad j \in K.$$

(1)
Throughout this paper, suppose that the following assumptions on the constraints always hold.

**Assumption 1** $g_j(x): R^N \to R$ is a class-$C^2$ function and is convex with respect to $x$ for every $j \in K$.

**Assumption 2** The constraint set $C = \{ x \in R^N \mid g_j(x) \leq 0, \ j \in K \}$ is nonvoid and bounded. Furthermore, the Slater’s conditions hold.

4. Distributed Generalized Nash Equilibrium Seeking Law Design and Convergence Analysis

In this section, we aim to find a Nash equilibrium of a generalized convex game, where the objective functions satisfy the following assumption.

**Assumption 3** $f_j(x_i, x_{-i}): R^N \to R$ is a class-$C^2$ function and is convex with respect to $x_i$ for every $i \in V$.

4.1. Existence and Uniqueness of the Normalized Nash Equilibrium

Under Assumptions 1-3, the players’ optimization problems are convex and Nash equilibria exist [7]. Moreover, a point $x^* = [x^*_1, \cdots, x^*_N]^T \in R^N$ is a Nash equilibrium of the game if and only if the following KKT conditions hold for all $i \in V$ and some $\lambda^*_j \geq 0, j \in K$:

$$
\nabla_{x_i} f_i(x^*) + \sum_{j \in K} \lambda^*_j \nabla_{x_i} g_j(x^*) = 0, \lambda^*_j g_j(x^*) = 0, g_j(x^*) \leq 0, j \in K.
$$

(2)

Usually, the problem in (1) has multiple Nash equilibria. In this paper, we do not assume the uniqueness of the Nash equilibrium. Instead, we consider a special kind of Nash equilibrium, namely the normalized equilibrium point [7], which is defined according to the KKT condition (2), such that for all $i \in V$,

$$
\nabla_{x_i} f_i(x^*) + \sum_{j \in K} \lambda^*_j \nabla_{x_i} g_j(x^*) = 0, \lambda^*_j g_j(x^*) = 0, g_j(x^*) \leq 0, j \in K,
$$

(3)

where $\lambda^*_j \geq 0$ is the common Lagrange multiplier for each shared constraint $g_j(x)$.

Before presenting the algorithm, we show that under some assumptions, the normalized Nash equilibrium satisfying (3) exists and is unique. Firstly, we introduce the concept of monotonicity for a single-valued mapping $F: R^N \to R^N$.

**Definition 1 (On monotonicity of a single-valued mapping)** [19]

(1) A mapping $F: R^N \to R^N$ is said to be monotone on $R^N$ if for all $x, \bar{x} \in R^N, x \neq \bar{x}, (F(x) - F(\bar{x}))^T (x - \bar{x}) \geq 0$.

(2) A mapping $F: R^N \to R^N$ is said to be strictly monotone on $R^N$ if for all $x, \bar{x} \in R^N, x \neq \bar{x}, (F(x) - F(\bar{x}))^T (x - \bar{x}) > 0$.

**Assumption 4** [7] The function $\nabla_{x} f(x) = [\nabla_{x_1} f_1(x), \cdots, \nabla_{x_N} f_N(x)]^T$ is strictly monotone on $R^N$.

**Lemma 1** [7] Under Assumptions 1-4, the normalized Nash equilibrium $x^*$ satisfying (3) exists and is unique.

The following assumption will be used in the stability analysis.

**Assumption 5** Under Assumptions 1-4, the pair $(x^*, \lambda^*_1, \cdots, \lambda^*_N)$ satisfying (3) is unique.

4.2. Control Design and Stability Analysis

We now propose a distributed control law using the estimation of neighboring actions. Let $Y_i = [y_{i1}, \cdots, y_{in}]^T$ be player $i$’s estimation on all the players’ actions, which is produced by the following consensus based protocol

$$
y_{ii} = x_i, i \in V, y_{ij} = -w_i \sum_{k=1}^{N} a_{ik} (y_{jk} - y_{ik}), j \in V \setminus \{i\},
$$

(4)
with \( w_j \) being a positive constant.

Based on (4), the updating law for play \( i \) is designed as
\[
\dot{x}_i = -\bar{k}_i(\nabla_x f(Y_i) + \sum_{j \in \mathcal{K}} \nabla_x g_j(Y_i)),
\]
\[
\dot{\lambda}_{ij} = -\gamma_{ij} \sum_{k=1}^{N} a_{ik}(\lambda_{ij} - \lambda_{kj}), \quad j \in \mathcal{K}, i \neq 1,
\]
where player 1 is selected to calculate the common multipliers \( \lambda_{ij} \) and consensus based control laws are used to broadcast them to all the other players. In (5), \( \lambda_{ij}(0) > 0, \gamma_{ij} > 0, \) and \( \bar{k}_i = \varepsilon k_i \), where \( \varepsilon \) is a small positive constant and \( k_i > 0 \) is a positive constant.

The distributed Nash equilibrium seeking algorithm in (4) and (5) uses leader-following consensus to estimate unknown information and saddle point dynamics to achieve convergence. In the following analysis, we use singular perturbation theory to prove the stability of the system. First, an auxiliary system is designed. Then, based on the stability of this auxiliary system, we prove the convergence of the original system using Lyapunov based methods.

For agent \( i \), define a subgraph \( \mathcal{G}_i = \{ \mathcal{V}_i, \mathcal{E}_i \} \), where \( \mathcal{V} = \mathcal{V} / \{ i \} \) and \( \mathcal{E}_i \subset \mathcal{V} \times \mathcal{V} \) indicate the set of vertices and edges, respectively. Let \( L_i \) be the Laplacian matrix of \( \mathcal{G}_i \) and \( B_i = \text{diag} \{ a_{i,1}, \ldots, a_{i(i-1)}, \ldots, a_{iN} \} \in \mathbb{R}^{(N-1) \times (N-1)} \) is the information exchange matrix of agent \( i \). Then, we can rewrite (4) and (5) as:
\[
\dot{x} = \bar{k}(\nabla_x f(Y) + \sum_{j \in \mathcal{K}} \text{diag}(\lambda_{ij}) \nabla_x g_j(Y)), \quad \dot{\lambda}_{ij} = \bar{k}_i \lambda_{ij} g_j(Y_i),
\]
where \( Y = [Y_1^T, \ldots, Y_N^T]^T \in \mathbb{R}^{N^2}, \nabla_x f(Y) = [\nabla_x f_1(Y_1), \ldots, \nabla_x f_N(Y_N)]^T, \nabla_x g(Y) = [\nabla_x g_1(Y_1), \ldots, \nabla_x g_N(Y_N)]^T, \lambda = [\lambda_2, \ldots, \lambda_N]^T, \bar{y}_i = [y_{i1}, \ldots, y_{i(i-1)}, \ldots, y_{iN}]^T \in \mathbb{R}^{N-1}, \bar{k} = \text{diag} \{ \bar{k}_2, \ldots, \bar{k}_N \} \in \mathbb{R}^{N^2}, \lambda = [\lambda_2, \ldots, \lambda_N]^T, \bar{y}_i = [y_{i1}, \ldots, y_{i(i-1)}, \ldots, y_{iN}]^T \in \mathbb{R}^{N-1} \), and \( \alpha_i = \text{diag} \{ \alpha_{i2}, \ldots, \alpha_{iN} \} \). Define an error variable \( \Delta(t) = [(x(t) - x^*)]^T, [\lambda(t) - \lambda^*]^T, [\bar{y}_i(t) - \bar{y}_{iN}]^T \) \( \in \mathbb{R}^{(N-1) \times (N-1)} \). The following theorem presents the main result of this section.

**Theorem 1** Suppose that Assumptions 1-5 hold and let (4) and (5) be the updating law. Then, for each pair of positive constants \( (r, \tilde{r}) \), there exists a positive constant \( \varepsilon^*(r, \tilde{r}) \) such that for every \( 0 < \varepsilon < \varepsilon^*(r, \tilde{r}) \), there exists a time \( T \) such that \( \| \Delta(t) \| \leq \tilde{r}, \quad \forall t \geq T \) for every \( \| \Delta(0) \| \leq r \).

**Proof:** Firstly, introduce the following auxiliary system
\[
\dot{x} = -k(\nabla_x f(x) + \sum_{j \in \mathcal{K}} \lambda_{ij} \nabla_x g_j(x)), \quad \dot{\lambda}_{ij} = k_{ij} \lambda_{ij} g_j(x), \quad j \in \mathcal{K},
\]
where \( k = \text{diag} \{ k_1, \ldots, k_N \} \in \mathbb{R}^{N \times N} \).

For (7), define a Lyapunov candidate function as
\[
V_0 = \frac{1}{2} (x - x^*)^T k^{-1}(x - x^*) + \sum_{j \in \mathcal{K}} \frac{1}{k_{ij}} (\lambda_{ij} - \lambda^*_i - \lambda^*_j \log(\lambda_{ij}) + \lambda^*_j \log(\lambda^*_j)),
\]
where $[x^T, \lambda_1, \ldots, \lambda_N]^T$ is the point satisfying (3). Taking the derivative of $V$ along (7), we have

$$V'_0 = (x - x^*)^T (-\nabla_x f(x) - \sum_{j \in K} \lambda_j \nabla_x g_j(x)) + \sum_{j \in K} (\lambda_j - \lambda_j^*) g_j(x). \tag{8}$$

Based on (3), if $\lambda_j^* > 0$, we have $(\lambda_j - \lambda_j^*) g_j(x^*) = 0$; otherwise $\lambda_j^* = 0$, which implies that $(\lambda_j - \lambda_j^*) g_j(x^*) \leq 0$. Under Assumption 4, we have $(x - x^*)^T (\nabla_x f(x) - \nabla_x f(x^*)) \geq 0$. Using (3), equation (8) can be rewritten as

$$V'_0 \leq -(x - x^*)^T \nabla_x f(x^*) - \sum_{j \in K} (x - x^*)^T \lambda_j \nabla_x g_j(x) + \sum_{j \in K} (\lambda_j - \lambda_j^*) (g_j(x) - g_j(x^*))$$

$$= -(x - x^*)^T (-\sum_{j \in K} \lambda_j^* \nabla_x g_j(x)) - \sum_{j \in K} (x - x^*)^T \lambda_j \nabla_x g_j(x) + \sum_{j \in K} (\lambda_j - \lambda_j^*) (g_j(x) - g_j(x^*))$$

$$\leq \sum_{j \in K} \lambda_j^* (g_j(x) - g_j(x^*)) + \sum_{j \in K} \lambda_j^* (g_j(x^*) - g_j(x^*)) + \sum_{j \in K} (\lambda_j - \lambda_j^*) (g_j(x) - g_j(x^*))$$

$$= 0,$$

where we utilize the convexity of $g_j(x)$.

Based on Assumption 4, letting $V'_0 \equiv 0$ gives $x = x^*$. Then, the largest invariant set can be written as

$$S = \{(x, \lambda_1, \ldots, \lambda_N) \in \mathbb{R}^N \times \mathbb{R}^{2^N} \times \cdots \times \mathbb{R}^{2^N} : 0 = \nabla_x f(x) + \sum_{j \in K} \lambda_j \nabla_x g_j(x), x = x^*, \lambda_j g_j(x^*) = 0, j \in K\}.$$

It can be seen that all the points in $S$ satisfy the condition in (3). According to Lemma 1 and Assumption 5, such a point is unique. Define $z = [(x - x^*)^T, [\lambda_j - \lambda_j^*]^T]^T$. Thus, for each $z \neq 0$, there exists a positive definite function $\beta(z)$ such that

$$(x - x^*)^T (-\nabla_x f(x) - \sum_{j \in K} \lambda_j \nabla_x g_j(x)) + \sum_{j \in K} (\lambda_j - \lambda_j^*) g_j(x) < -\beta(z). \tag{9}$$

For notational convenience, $\hat{Y}_i = \tilde{Y}_i - x_i^*$ and $\hat{k}_j = \lambda_j - \lambda_j^*$. Define a Lyapunov candidate $V(z, [\hat{Y}_i], [\hat{k}_j]) = c V'_0 + (1 - c) V_1$, where $c > 0$ is a positive constant and $V_1 = \frac{1 - c}{2} \sum_{i=1}^N \hat{y}_{i}^T \hat{Y}_i + \sum_{i=1}^N \hat{k}_{i}^T \hat{k}_j$. It can be obtained that there exist two class-$K_\infty$ functions $\alpha_i$ and $\alpha_N$ such that $\alpha_i(z, [\hat{Y}_i], [\hat{k}_j]) \leq V(z, [\hat{Y}_i], [\hat{k}_j]) \leq \alpha_N(z, [\hat{Y}_i], [\hat{k}_j]).$

Thus, taking the derivative of $V$ along (6) gives

$$\dot{V} = -c e (x - x^*)^T (\nabla_x f(Y) - \nabla_x f(x)) - c e (x - x^*)^T \nabla_x f(x) - c e (x - x^*)^T$$

$$\times (\sum_{j \in K} \text{diag} \{\lambda_j, \lambda_j\} \nabla_x g_j(Y) - \sum_{j \in K} \lambda_j \nabla_x g_j(Y)) - c e (x - x^*)^T$$

$$\times (\sum_{j \in K} \lambda_j \nabla_x g_j(Y) - \sum_{j \in K} \lambda_j \nabla_x g_j(x)) - c e (x - x^*)^T \sum_{j \in K} \lambda_j \nabla_x g_j(x)$$

$$- c e \sum_{j \in K} (\lambda_j - \lambda_j^*) (g_j(Y) - g_j(x)) - c e \sum_{j \in K} (\lambda_j - \lambda_j^*) g_j(x)$$

$$-(1 - c) \sum_{i=1}^N \hat{y}_{i}^T W_i (L_j + B_j) \hat{Y}_i - (1 - c) \sum_{i=1}^N \hat{y}_{i}^T \frac{dx}{dt} 1 - (1 - c) \sum_{j \in K} \hat{k}_{j}^T \hat{k}_j$$

$$- (L_j + B_j) \hat{k}_j - (1 - c) \sum_{j \in K} \hat{k}_{j}^T \frac{d\lambda_{j}}{dt} 1.$$
\[
\dot{V} \leq -c\varepsilon \beta(z) + c \varepsilon \left( \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 + \varepsilon \left( \sum_{i=1}^{N} \left\| \nabla g_j(Y) \right\| \right) \right)
\]
\[
+ c \varepsilon \left( \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 \right)^{\frac{1}{2}} \max \{ \lambda_{ij} \} + c \varepsilon l_3 \sum_{j,k} \left\| \lambda_{ij} - \lambda_{kj} \right\| \left( \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 \right)^{\frac{1}{2}}
\]
\[
-(1-c)l_4 \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 - (1-c) \sum_{i=1}^{N} \dot{y}_i \frac{dx}{dt} - (1-c)l_5 \sum_{j,k} \left\| \hat{\lambda}_{ij} \right\|^2
\]
\[
-(1-c) \sum_{j,k} \frac{d\lambda_{ij}}{dt} 1,
\]
where \(l_1, \ldots, l_5\) are some positive constants.

Let \(\delta_1, \ldots, \delta_8\) be positive constants that can be arbitrarily chosen. Using Young’s Inequality gives
\[
\dot{V} \leq -c\varepsilon \beta(z) - [1-(1-c)] \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 \left[ -l_5(1-c) - \frac{\varepsilon}{\delta_2} - \frac{\varepsilon}{\delta_6} \right] + \frac{c^2\varepsilon^2 \delta_3}{4} \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 \left( \sum_{j,k} \left\| \lambda_{ij} - \lambda_{kj} \right\|^2 \right)
\]
\[
+ \frac{c^2\varepsilon^2 \delta_3^2}{4} \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 \left( \sum_{j,k} \left\| \lambda_{ij} - \lambda_{kj} \right\|^2 \right)
\]
\[
+ \frac{c^2\varepsilon^2 \delta_3^2}{4} \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 + \frac{c\varepsilon \delta_6^2 (1-c)^2}{4} \sum_{j,k} \left\| \frac{d\lambda_{ij}}{dt} 1 \right\|^2.
\]

Let \(\varepsilon\) be chosen such that \(0 < \varepsilon < \min \{\varepsilon_1, \varepsilon_2\}\), where \(\varepsilon_1 = \left( \frac{1}{\delta_1} + \frac{1}{\delta_4} + \frac{1}{\delta_5} + \frac{1}{\delta_7} \right)^{-1}(1-c)l_4\),

and \(\varepsilon_2 = \left( \frac{1}{\delta_2} + \frac{1}{\delta_5} + \frac{1}{\delta_8} \right)^{-1}(1-c)l_5\). Then (10) can be rewritten as
\[
\dot{V} \leq -c\varepsilon \beta(z) + \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 + \sum_{j,k} \left\| \hat{\lambda}_{ij} \right\|^2 + o(\varepsilon \delta_1)
\]
\[
+ o(\varepsilon \delta_2) + o(\varepsilon \delta_3) + o(\varepsilon \delta_4) + o(\varepsilon \delta_5) + o(\varepsilon \delta_6)
\]
\[
\leq -c\varepsilon \rho(\|\Delta\|) + \varepsilon \sum_{i=1}^{6} o(\delta_i),
\]
where \(c_\rho = \min \{c, \frac{1}{\delta_1}, \frac{1}{\delta_8} \}\) and \(\rho(\|\Delta\|) = \beta(z) + \sum_{i=1}^{N} \left\| \dot{y}_i \right\|^2 + \sum_{j,k} \left\| \hat{\lambda}_{ij} \right\|^2 \).

Therefore, for each \(\|\Delta(t)\| \geq \rho^{-1}(\frac{2}{c_\rho} \sum_{i=1}^{6} o(\delta_i)))\), \(\dot{V} \leq -\frac{c}{2} c_\rho \rho(\|\Delta\|)\). Based on Theorem 4.18 of [20], there exists \(\delta_i^*\) such that for any \(0 < \delta_i < \delta_i^*\), \(i = 1, \ldots, 6\), there exists \(\varepsilon^*\) such that for any \(0 < \varepsilon < \varepsilon^*\), there exists \(T > 0\) and class-\(K\) function \(\rho_0\) [20] such that
\[
\|\Delta(t)\| \leq \rho_0(\|\Delta(0)\|, t) \leq T, \|\Delta(t)\| \leq \alpha_1^{-1}(\alpha_2(\rho^{-1}(\frac{2}{c_\rho} \sum_{i=1}^{6} o(\delta_i))))), t \geq T.
\]
Remark 1 In this paper, we consider that each player has several shared constraints, where each constraint is shared by all the players. Our algorithm can also be used to solve the following problem:

$$\min_{x_i} f_i(x_i, x_{-i}), \text{ such that } g_{ij}(x_i) \leq 0, \ j \in K,$$

where each player has multiple private constraints that depend on it's own action. The algorithm can be designed as follows:

$$y_{ij} = x_i, i \in V, \dot{y}_{ij} = -w_{ij} \sum_{k=1}^{N} a_{ik} (y_{ij} - y_{kj}), j \in V / \{i\},$$

$$\dot{x}_i = -\bar{K}_i (\nabla_y f_i(Y_i) + \sum_{j \in K} \nabla_y \lambda_{ij} g_{ij}(x_i)), \dot{\lambda}_{ij} = \bar{K}_{ij} \lambda_{ij} g_{ij}(x_i),$$

where player 1 is selected to calculate the common multipliers $\lambda_{ij}$ and consensus based control laws are used to broadcast them to all the other players. In (12), $\lambda_{ij}(0) > 0, \ \gamma_{ij} > 0$, and $\bar{K}_i = \varepsilon k_i$, where $\varepsilon$ is a small positive constant and $k_i > 0$ is a positive constant.

Remark 2 The objective functions in this paper are assumed to belong to class $C^2$. In practical applications, there are many objective functions that don’t satisfy this condition, e.g., locally Lipschitz objective functions. The proof for locally Lipschitz objective functions are more complicated. The reasons are three-fold: 1) The functions are not differentiable at some points. To deal with this problem, subgradients are used for control design, which leads to nonsmooth dynamical systems. 2) For a locally Lipschitz objective function, its subdifferential is not locally Lipschitz but upper semicontinuous (see [21]), which brings challenges to singular perturbation analysis, the main analysis method used in this paper. 3) The existence and uniqueness of the normalized Nash equilibrium for games with locally Lipschitz objective functions must be analyzed. Here, we only give the algorithm design, and the convergence analysis for the differential inclusion will be our future work. The designed algorithm for locally Lipschitz objective functions is described as follows:

$$\dot{x}_i = -\bar{K}_i (h_i(Y_i) + \sum_{j \in K} \lambda_{ij} \nabla_y g_{ij}(Y_i)), \dot{\lambda}_{ij} = \bar{K}_{ij} \lambda_{ij} g_{ij}(Y_i), \dot{\lambda}_{ij} = -\gamma_{ij} \sum_{k=1}^{N} a_{ik} (\lambda_{ij} - \lambda_{kj}), j \in K, i \neq 1,$$

where $Y_i$ is produced by (4). In (13), $h_i(Y_i)$ is the subgradient of $f_i(Y_i)$, defined as $h_i(Y_i) \in \partial_x f_i(Y_i)$, where $\partial_x f_i(Y_i)$ is the subdifferential of $f_i(x)$ at $Y_i$, and $\bar{K}_i$ and $\gamma_{ij}$ are defined the same as in (12).

5. Simulation

Example 1 A network of 5 players is shown in Fig. 1. Let $f_1(x, t) = x_1^2 + (x_1 - x_2 - 3)^2$, $f_2(x, t) = (x_1 - x_2)^2 + (x_2 - x_3)^2$, $f_3(x, t) = x_3^2 + (x_2 - x_3)^2$, $f_4(x, t) = (x_4 - x_3 - 1)^2$, and $f_5(x, t) = x_4^2 + (x_3 - x_4)^2$ be the local objective functions for players 1-5, respectively. The constraint set is $\Omega = \{x \in \mathbb{R}^5 \mid \|x - d\|^2 \leq 9, d = [1, 2, 1, 0, -1]^T\}$. Let (4) and (5) be the updating law. Fig. 2 shows the simulation result of the estimate on the optimal solutions $x_1^*(t), x_2^*(t), \ldots, x_5^*(t)$.

Figure 1. The communication graph.
Figure 2. The estimate on the Nash equilibrium.

6. Conclusions
In this paper, we studied the distributed Nash equilibrium seeking problems for generalized convex games with multiple shared constraints. Supposing that each player can exchange information with its neighbors via a connected undirected graph, continuous-time control laws were designed such that each player minimizes its own cost function in a distributed way. In future, we will consider distributed continuous-time Nash equilibrium seeking for nonsmooth objective functions with convergence analysis. In addition, we will relax the assumptions used in the paper.

Acknowledgments
This work was supported by Singapore Economic Development Board under EIRP grant S14-1172-NRF EIRP-IHL.

References
[1] Nash J 1951 Non-cooperative games Annals of mathematics 286–295
[2] Ratliff L J, Burden S A and Sastry S S 2013 Characterization and computation of local nash equilibria in continuous games 2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton) pp 917–924
[3] Vamvoudakis K G, Lewis F L and Hudas G R 2012 Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality Automatica 48 1598–1611
[4] Frihauf P, Krstic M and Basar T 2012 Nash equilibrium seeking in noncooperative games IEEE Transactions on Automatic Control 57 1192–1207
[5] Ye M and Hu G 2015 Distributed seeking of time-varying Nash equilibrium for non-cooperative games IEEE Transactions on Automatic Control 60 3000–3005
[6] Ye M and Hu G 2016 Distributed extremum seeking for constrained networked optimization and its application to energy consumption control in smart grid IEEE Transactions on Control Systems Technology 24 2048–2058
[7] Rosen J B 1965 Existence and uniqueness of equilibrium points for concave n-person games Econometrica: Journal of the Econometric Society 520–534
[8] Pang J S and Fukushima M 2005 Quasi-variational inequalities, generalized Nash equilibria, and multi-leaderfollower games Computational Management Science 2 21–56
[9] Facchinei F and Kanzow C 2007 Generalized Nash equilibrium problems 4OR: A Quarterly Journal of Operations Research 5 173–210
[10] Debreu G 1952 A social equilibrium existence theorem Proceedings of the National Academy of Sciences 38 886–893
[11] Arrow K J and Debreu G 1954 Existence of an equilibrium for a competitive economy Econometrica: Journal of the Econometric Society 265–290
[12] Facchinei F, Fischer A and Piccialli V 2007 On generalized Nash games and variational inequalities Operations Research Letters 35 159–164

[13] Gharesifard B and Cortes J 2014 Distributed continuous-time convex optimization on weight-balanced digraphs IEEE Transactions on Automatic Control 59 781–786

[14] Yi P, Hong Y and Liu F 2015 Distributed gradient algorithm for constrained optimization with application to load sharing in power systems Systems & Control Letters 83 45–52

[15] Cherukuri A, Mallada E and Cortes J 2016 Asymptotic convergence of constrained primal–dual dynamics Systems & Control Letters 87 10–15

[16] Bai L, Ye M, Sun C and Hu G 2016 Distributed control for optimal economic dispatch via saddle point dynamics and consensus algorithms IEEE 55th Conference on Decision and Control (CDC) pp 6934–6939

[17] Sun C, Ye M and Hu G 2017 Distributed time-varying quadratic optimization for multiple agents under undirected graphs IEEE Transactions on Automatic Control 62 3687–3694

[18] Ye M and Hu G 2017 Distributed Nash equilibrium seeking by a consensus based approach IEEE Transactions on Automatic Control 62 4811–4818

[19] Zhu M and Frazzoli E 2016 Distributed robust adaptive equilibrium computation for generalized convex games Automatica 63 82–91

[20] Khalil H K 2002 Nonlinear Systems New Jersey, Prentice Hall 9

[21] Niederl’ander S K and Cortes J 2016 Distributed coordination for nonsmooth convex optimization via saddlepoint dynamics arXiv preprint arXiv:1606.09298