Potential game design for a class of distributed optimisation problems

Peng Yi\textsuperscript{a}, Yanqiong Zhang\textsuperscript{a} & Yiguang Hong\textsuperscript{a}

\textsuperscript{a}Key Laboratory of Systems and Control, Institute of Systems Science, Chinese Academy of Sciences, Beijing, China.

Published online: 15 May 2014.

To cite this article: Peng Yi, Yanqiong Zhang & Yiguang Hong (2014) Potential game design for a class of distributed optimisation problems, Journal of Control and Decision, 1:2, 166-179, DOI: 10.1080/23307706.2014.899111

To link to this article: \url{http://dx.doi.org/10.1080/23307706.2014.899111}
Potential game design for a class of distributed optimization problems

Peng Yi, Yanqiong Zhang and Yiguang Hong*

Key Laboratory of Systems and Control, Institute of Systems Science, Chinese Academy of Sciences, Beijing, China

(Received 1 November 2013; accepted 30 March 2014)

The state-based potential game is discussed and a game-based approach is proposed for distributed optimization problem in this paper. A continuous-time model is employed to design the state dynamics and learning algorithms of the state-based potential game with Lagrangian multipliers as the states. It is shown that the stationary state Nash equilibrium of the designed game contains the optimal solution of the optimization problem. Moreover, the convergence and stability of the learning algorithms are obtained for both undirected and directed communication graph. Additionally, the application to plug-in electric vehicle management is also discussed.

Keywords: distributed optimization; potential game; multi-agent systems; plug-in electric vehicle

1. Introduction

In recent years, the distributed optimization problems have drawn much research attention as the rapid development of large-scale networks, including sensor networks, smart grid, and traffic networks. Discrete-time algorithms incorporated with consensus dynamics were shown to be an efficient way to migrate existing optimization methods to distributed networks, including the sub-gradient algorithms (Lou, Shi, Johansson, & Hong, 2012; Nedic & Ozdaglar, 2009; Nedic, Ozdaglar, & Parrilo, 2010; Tsitsiklis & Athans, 1984), alternating direction method of multipliers (Wei & Ozdaglar, 2012), and dual averaging methods (Duchi, Agarwal, & Wainwright, 2012). However, the discrete-time algorithms share some limitations including the sensitivity to additive noise and the diminishing step size. On the other hand, with the inspiration of the seminal works (Arrow, Hurwitz, & Uzawa, 1958), distributed optimization algorithms with continuous-time dynamics were proposed (referring to (Gharesifard & Cortes, 2014; Shi & Johansson, 2013; Shi, Johansson, & Hong, 2013; Wang & Elia, 2010, 2011)), which may overcome the above limitations in some sense. Continuous-time dynamics also provide a control perspective for the design of optimization algorithms for multi-agent networks.

In fact, game theory has also been used to facilitate the control and optimization of multi-agent systems in (Campos-Nanez, Garcia, & Li, 2008; Marden, 2012; Saad, Han, Poor, & Basar, 2012; Yi, Zhang, & Hong, 2013). With the help of game theory, the self-interest decision-makers are modelled as players under a game theoretic framework, which includes the players set, the rules of action, and the action space of each player. A proper local objective function is assigned for each agent in order to provide proper incentive for
the agents to reach the desirable equilibrium point. Then a learning algorithm is proposed as the interaction rules of agents to ensure the desirable operation point (Nash equilibrium) as a result of agents interaction. To be specific, game methods with discrete-time learning algorithms were proposed in (Li & Marden, 2013; Marden, Arslan, & Shamma, 2009) to solve the distributed optimization problem, which also showed that the distributed optimization problem could be solved with robustness to variation in clock rates, delays in information, and component failures by the inherent robustness property of potential game. Stochastic search algorithms were explored to solve the distributed optimization problem under the framework of potential game in (Marden, 2012). However, some constraints must be imposed to the communication graph to ensure the outcome of the game will be the optimal solution under the state space design in (Li & Marden 2013; Marden et al., 2009), which may limit the application of the design in (Li & Marden, 2013). Moreover, directed communication graph was not discussed in (Li & Marden 2013; Marden et al., 2009).

The object of this paper is to propose a new continuous-time distributed learning algorithm to reach the Nash equilibrium. We extend the results given in (Li & Marden, 2013) by considering directed communication graphs with the re-selection of the states and the continuous-time dynamics of the state-based potential game. We show that our game design ensures that the equilibrium of the game contains the optimal solution. Then, we redesign the potential game and the learning algorithms under the undirected communication graph. The Lyapunov method is adopted to show the convergence of the proposed learning algorithms for both the directed and undirected graphs, which also provides the stability of the optimal solution.

This paper is organised as follows. In Section 2, we introduce some preliminaries and the problem formulation. Then we design a game to solve the distributed optimization problem with directed communication graphs in Section 3. We redesign the game to achieve better utility of information in the undirected graph case in Section 4. Following that we apply the game algorithm to the Plug-in Electric Vehicles (PIEVs) charge station management with a simulation example in Section 5. Finally, we give concluding remarks in Section 6.

Notations Let $\mathbb{R}$, $\mathbb{R}_0$ be the sets of real, nonnegative real numbers, respectively; vector $x$ denotes a column vector, and $x^T$ denotes its transpose; denote $\|x\|_2 = \sqrt{x^T x}$ as the Euclidean norm of vector $x$; denote $1_d = (1, \ldots, 1)^T \in \mathbb{R}^d$ and $0_d = (0, \ldots, 0)^T \in \mathbb{R}^d$; let $I_d$ be the identity matrix in $\mathbb{R}^{d \times d}$; $A \otimes B$ denotes the Kronecker product of matrices $A$ and $B$. Denote the vector $(x_1^T, x_2^T, \ldots, x_n^T)^T$ as $(x_1, x_2, \ldots, x_n)^T$ if there is no confusion.

2. Preliminaries and problem formulation

In this section, we give preliminary knowledge for the following analysis.

We first review some useful concepts in algebraic graph theory (Godsil, Royle, & Godsil, 2001). A graph is a pair $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, \ldots, n\}$ is a finite set called vertex set, and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is called edge set. A weighted graph is a triplet $\mathcal{G} = (\mathcal{N}, \mathcal{E}, A)$, where $(\mathcal{N}, \mathcal{E})$ is a graph and $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ is the weighted adjacency matrix of $\mathcal{G}$ with $a_{ij} > 0$ if and only if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The weighted out-degree matrix $D$ is the diagonal matrix defined by $D = \text{diag}(d_1, \ldots, d_n)$, where $d_i = \sum_{j=1}^{n} a_{ij}$. The Laplacian matrix $L := D - A$ with $L 1_n = 0_n$. A path is an ordered sequence of vertices such that any ordered pair of vertices appearing consecutively is an edge of the graph. A graph is strongly connected if there is a path between any pair of distinct vertices. When $\mathcal{G}$ is strongly connected, zero is a simple eigenvalue of $L$. The graph $\mathcal{G}$ is undirected if $A = A^T$ or equivalently $L = L^T$. $\mathcal{G}$ is weighted balanced if $1_n^T L = 0_n$. 


Consider a network of agents labeled by \( \mathcal{N} := \{1, \ldots, n\} \) to cooperatively optimise the problem as follows:

\[
\min_v \phi(v) \tag{1}
\]

where \( v \in \mathbb{R}^m \) is collectively decided by all agents in \( \mathcal{N} \), and \( \phi : \mathbb{R}^m \to \mathbb{R} \) is a continuously differentiable convex function. The goal of the multi-agents group is to cooperatively solve the optimization problem in a distributed way because agent \( i \) can only use the information of its neighbors from \( \mathcal{N}_i = \{j | (i, j) \in \mathcal{E}\} \) to find the optimal solution. The problem (1) is said to be solvable if there exists \( v^* \in \mathbb{R}^m \) such that \( \phi(v^*) = \min_v \phi(v) \).

Suppose that each agent has an estimation of the optimal solution \( v^* \), denoted as \( v^i \in \mathbb{R}^m \). Then we need to design an algorithm to ensure \( v^i \to v^* \). At first, we convert the considered optimization problem to the following constrained optimization problem:

\[
\min_{v^i, i=1,\ldots,n} \sum_{i=1}^n \phi(v^i), \quad \text{subject to} \quad \mathbf{L} \bar{v} = \mathbf{0}_{mn}, \tag{2}
\]

where \( \mathbf{L} = L \otimes I_m \in \mathbb{R}^{mn \times mn} \) with \( L \) as the Laplacian matrix of graph \( \mathcal{G} \), and \( \bar{v} = (v^1, v^2, \ldots, v^n)^T \in \mathbb{R}^{mn} \). Then we have the following result:

**Lemma 1** If graph \( \mathcal{G} \) is strongly connected and \( \bar{v}^* \) is the optimal solution of the problem (2), then the first \( m \) components of \( \bar{v}^* \) is the optimal solution of the problem (1).

**Proof** When the graph \( \mathcal{G} \) is strongly connected, \( \mathbf{L} \bar{v} = \mathbf{0}_{mn} \), i.e. \( \bar{v} = 1_n \otimes v \). Then the optimization problem (2) becomes

\[
\min_{v^i} \sum_{i} \phi(v^i) = \min_{v} n \phi(v).
\]

So \( \bar{v}^* = 1_n \otimes v^* \) is the optimal solution of (2) if and only if \( v^* \) is the optimal solution of (1). \( \square \)

This transformation brings the consensus of the agents’ estimation of the optimal solution with the solution of a constrained optimization problem. Thus, we can just deal with the constrained optimization problem later on. Similar discussion can also be found in (Gharesifard & Cortes, 2014).

Recall some basic game notations in (Fudenberg & Tirole, 1991) before our design of potential game. \( \mathbf{G} := (\mathcal{N}, (a_i)_{i \in \mathcal{N}}, (s_i)_{i \in \mathcal{N}}, (u_i)_{i \in \mathcal{N}}) \) is called a finite strategic form game, where \( \mathcal{N} = \{1, \ldots, n\} \) is the set of players, \( a_i \in A_i \) specifies what kind of action player \( i \) can take, \( s_i \in S_i \) is a non-empty set of available strategy for player \( i \), and \( u_i : A \to \mathbb{R} \) is the payoff function of player \( i \), where \( A = \prod_{i \in \mathcal{N}} A_i \) denotes the joint action profile space. Denote \( a = (a_1, \ldots, a_n) \in A \) as an action profile, and \( a_{-i} \) as the action profile of all the agents except \( i \), thus \( a = (a_i, a_{-i}) \).

Define a state-based game as

\[
\mathbf{G} := (\mathcal{N}, (a_i)_{i \in \mathcal{N}}, (s_i)_{i \in \mathcal{N}}, x, (u_i)_{i \in \mathcal{N}})
\]

where \( x \in X \) is the state variable, and \( X \) is the state space. Furthermore, we have the state dynamics to be continuous-time, that is \( \dot{x} = f(x; a) \), to be designed later. With continuous-time state dynamics, we can design continuous-time learning algorithms for each agent later. Payoff function \( u_i : X \times A \to \mathbb{R} \) is the function of state and action profile. More details of such a game will be covered in Section 3.

In fact, potential game has been extensively studied in (Fudenberg & Levine, 1998; Monderer & Shapley, 1996; Young, 2004), where the existence of Nash equilibrium can
be guaranteed with some mild assumptions. Many learning algorithms have been proposed for the convergence to the Nash equilibrium in the potential game.

**Definition 1** (State-based Potential Game) A state-based game $G$ is a state-based potential game if there exists a potential function $\Phi : X \times A \rightarrow \mathbb{R}$ satisfying:

$$u_i(x; \hat{a}_i, a_{-i}) - u_i(x; a_i, a_{-i}) = \Phi(x; \hat{a}_i, a_{-i}) - \Phi(x; a_i, a_{-i}),$$

for all $i \in N$, $x \in X$, $\forall \hat{a}_i, a_i \in A_i$, $\forall a_{-i} \in \prod_{j \neq i} A_j$.

The state-based potential game has been investigated in (Marden, 2012), as a simplification of stochastic game or Markov game proposed in (Shapley, 1953). Then we define the stationary state Nash equilibrium as the proper predication of the outcome of a state-based potential game.

**Definition 2** (SSNE) A state action pair $[x^*, a^*]$ is a stationary state Nash Equilibrium (SSNE) of a state-based potential game $G = (N, (a_i)_{i \in N}, (s_i)_{i \in N}, x, (u_i)_{i \in N})$ if

- $a^*_i \in \arg\min_{a_i \in A_i} u_i(x^*; a_i, \hat{a}^*_{-i})$, $\forall i \in N$.
- The state $x^*$ is an equilibrium state of the state dynamics.

### 3. Design potential game under directed graph

In this section, we first give our design of the state-based potential game for each agent to play, and prove that the SSNE of the designed game guarantees the optimal solution in the first subsection. Then, we propose a learning algorithm for each agent to find the SSNE by local communication and interaction, and analyse the convergence of our algorithm and the stability of optimal solution by Lyapunov method in the second subsection.

#### 3.1. Design state-based potential game

We design a state-based potential game with the payoff function for each agent $i$ only depending on local information, such that each agent can evaluate its payoff by local communication and computation. We also need to ensure that the SSNE of the designed game contains the optimal solution of the problem defined in (1).

To be specific, we design the state-based game $G_1$ as follows.

- **Players**: $N = \{1, \ldots, n\}$.
- **Action space**: $a_i \in A_i$ is the estimation of the optimal solution for agent $i$. Clearly, $a_i = v_i, v_i \in \mathbb{R}^m, \forall a_i \in A_i$. For convenience, we use $a_i$ and $v_i$ exchangeably in the following analysis, and we use $\bar{v} = (v^1, v^2, \ldots, v^n)^T$ or $(v^i, v^{-i})$ to denote an action profile.
- **State space**: $\bar{x} = (x_1, x_2, \ldots, x_n)^T, x_i \in \mathbb{R}^m$. Here, we introduce state $x_i$ as the Lagrange multiplier associated with the constraint in the optimization problem (2).
- **State dynamics**: State dynamics describes how the underlying state evolves, which is designed according to the classical Lagrange multiplier theory (Boyd & Vandenberghe, 2004) and the continuous-time primal-dual algorithm in (Arrow, Hurwitz, & Uzawa, 1958): $\dot{x}_i = \sum_{j \in N_i} a_{ij} (v^j - v^i)$ or $\dot{\bar{x}} = L \bar{\nu}$, where $L = L \otimes I_m$.
- **Strategy space**: Strategy $s_i \in S_i$ is the full plan of agent $i$, which describes how agent $i$ reacts to the change of the states and the actions of other agents. For limitations in communication, the strategy can only depend on local information specified by communication graph $G$. We will design the strategy or the learning algorithm...
in Section 3.2 to ensure that the agents achieve the desired point by the local interactions.

- Payoff function:
\[
  u_i(\bar{x}; \bar{v}) = \phi(v^i) + v^i T \sum_{j \in N_i} a_{ij}(x_i - x_j),
\]
where \( A = [a_{ij}] \) is the weighted adjacency matrix of graph \( \mathcal{G} \). Clearly, the payoff function of agent \( i \) only relies on local neighbour information which could be achieved by local communication.

- Potential function:
\[
  \Phi(\bar{x}; \bar{v}) = \sum_{i \in \mathcal{N}} \phi(v^i) + \sum_{i \in \mathcal{N}} v^i T \sum_{j \in N_i} a_{ij}(x_i - x_j).
\]

**Remark 1** We design a continuous-time state dynamics, which is different from the state-based potential game in (Li & Marden 2013; Marden et al., 2009), and still use control theory to analyse the behaviour of agents (Basar & Olsder, 1999). Furthermore, we distinguish the concepts between action and strategy by viewing the strategy as the control of the action, and use a quiet different kind of states to achieve the goal. Additionally, we can deal with quite general class of distributed optimization problems by relaxing the constraints on the communication graphs in (Li & Marden 2013; Marden et al., 2009).

**Lemma 2** With \( \Phi \) defined in (5) as the potential function, the state-based game \( \mathbf{G}_1 \) is a state-based potential game.

Lemma 2 is easy to be verified, whose proof is omitted for space limitation. Then one main result of this section is given as follows:

**Theorem 1** Suppose the graph \( \mathcal{G} \) is directed weighted balanced and strongly connected. Then \([\bar{x}^*, \bar{v}^*]\) is SSNE of the state-based potential game \( \mathbf{G}_1 \) if and only if
\[
  x_1 = x_2 = \cdots = x_n = x^*, \quad v^1 = v^2 = \cdots = v^n = v^*
\]
namely,
\[
  \bar{x}^* = 1_n \otimes x^*, \quad \bar{v}^* = 1_n \otimes v^*
\]
where \( v^* \) is the optimal solution of the optimization problem (1).

**Proof**
Sufficiency: By
\[
  v^1 = v^2 = \cdots = v^n = v^*, \quad \bar{v} = 1_n \otimes v^*,
\]
we have
\[
  \dot{\bar{x}} = L\bar{v} = (L \otimes I_m)(1_n \otimes v^*) = (L1_n) \otimes (I_m v^*) = 0.
\]
Therefore, \( \bar{x}^* \) is an equilibrium state. Then we prove
\[
  u_i(x^*; v^i, v^{-i}) = \min_{v^i} u_i(x^*; v^i, v^{-i}), \quad \forall i \in \mathcal{N}.
\]
Because \( u_i(x^*; v^i, v^{-i}) \) is continuously differentiable convex function of \( v^i \), we have
\[
  \nabla_{v^i} u_i(x^*; v^i, v^{-i})|_{v^i=v^*} = \sum_{j \in N_i} a_{ij}(x_i - x_j) + \nabla_{v^i} \phi(v^i)|_{v^i=v^*}.
\]
As $v^*$ is the optimal solution of the problem (1), we get $\nabla_v \phi(v)|_{v=v^*} = 0_m$. By
\[ x_1 = \cdots = x_n = x^*, \quad v^1 = \cdots = v^n = v^*, \]
We have
\[ \nabla_v u_i(x^*; v^i, v^{-i})|_{v^i=v^*} = 0_m. \]
Since $u_i(x^*; v^i, v^{-i})$ is continuously differentiable convex function in $v^i$,
\[ u_i(x^*; v^i, v^{-i}) = \min_{v^i} u_i(x^*; v^i, v^{-i}), \quad \forall i \in \mathcal{N}. \]

Necessity: Because $\dot{x} = 0$, $L\ddot{v} = (L \otimes I_m)\ddot{v} = 0$. For the graph $G$ is strongly connected, $1_n$ is the only eigenvector associated with the simple eigenvalue 0 of Laplacian matrix $L$. Therefore, $L\ddot{v} = (L \otimes I_m)\ddot{v} = 0$ if and only if $\ddot{v} \in \text{span}\{1_n \otimes v\}, \forall v \in \mathbb{R}^m$, which implies
\[ v^1 = v^2 = \cdots = v^n. \]
Denote $v^1 = v^2 = \cdots = v^n = v^*$. By
\[ u_i(x^*; v^i, v^{-i}) = \min_{v^i} u_i(x^*; v^i, v^{-i}), \quad \forall i \in \mathcal{N}, \]
we get
\[ \nabla_{v^i} u_i(x^*; v^i, v^{-i})|_{v^i=v^*} = \sum_{j \in N_i} a_{ij}(x_i - x_j) + \nabla_{v^i} \phi(v^i)|_{v^i=v^*} = 0_m. \]
Rewritten in a vector form, we have:
\[ (\nabla_{v^1} \phi(v^1)|_{v^1=v^*}, \ldots, \nabla_{v^n} \phi(v^n)|_{v^n=v^*})^T = -L\ddot{x}. \]
For any vector $v \in \mathbb{R}^m$,
\[ (1_n \otimes v)^T L\ddot{x} = (1_n^T L) \otimes (v^T I_m)\ddot{x} = 0, \]
because the graph $G$ is weighted balanced and strongly connected. Hence,
\[ (1_n \otimes v)^T (\nabla_{v^1} \phi(v^1)|_{v^1=v^*}, \ldots, \nabla_{v^n} \phi(v^n)|_{v^n=v^*})^T = 0. \]
Therefore, $nv^T \nabla_v \phi(v)|_{v=v^*} = 0_m$ is satisfied for any $v \in \mathbb{R}^m$. As a result, $\nabla_v \phi(v)|_{v=v^*} = 0$. We have that $v^*$ is the optimal solution of the optimization problem (1) for $\phi(v)$ is continuously differentiable convex function. Consequently,
\[ -L\ddot{x} = (1_n \otimes 0_m). \]
Thus,
\[ x_1 = x_2 = \cdots = x_n = x^* \]
for the graph $G$ is strongly connected. The proof is completed.

### 3.2. Learning algorithm and convergence analysis

For a state-based potential game, many learning algorithms can guarantee the convergence to its Nash equilibrium (Marden, 2012). This also follows directly from (Young, 2004) that any “reasonable” distributed learning algorithm will converge to a pure Nash equilibrium in any finite potential game or finite state-based potential game. In fact, we employ game theory in the distributed optimization problem by applying those results of learning algorithms in
The result on the convergence and stability analysis of the proposed learning algorithm is given as follows.

**Theorem 2** Consider the state-based potential game $G_1$ and learning algorithm (6). If $\phi(v)$ is continuously differentiable convex function and bounded below, then all agents converge to SSNE of the state-based potential game $G_1$ and the SSNE is stable.

**Proof** Denote $G(\bar{v}) = (\nabla_{v^1} \phi(v^1), \ldots, \nabla_{v^n} \phi(v^n))^T$. Writing the dynamics of the states and the actions of all the agents in vector forms yields

$$\dot{x} = L\bar{v}, \quad \dot{\bar{v}} = -L\dot{x} - G(\bar{v}).$$

The SSNE $[\bar{x}^*, \bar{v}^*]$ satisfies $\dot{x}^* = 0$, and $\dot{\bar{v}}^* = 0$. Then we need to prove $[\bar{x}^*, \bar{v}^*]$ is an asymptotically stable equilibrium of (7).

Define a function $\Psi(x; \bar{v}) := \sum_{i \in N} \phi(v^i) + \bar{x}^T L\bar{v}$. Notice that stationary state Nash equilibrium $[\bar{x}^*, \bar{v}^*]$ is the saddle point of the function $\Psi(x; \bar{v})$, i.e.

$$\Psi(x; \bar{v}) = \Psi(\bar{x}^*; \bar{v}^*) \leq \Psi(\bar{x}^*; \bar{v}) \leq \Psi(x; \bar{v}),$$

because $\Psi(x; \bar{v})$ is convex in $\bar{v}$ and concave in $\bar{x}$, and

$$\nabla_{\bar{v}} \Psi(x; \bar{v}) |_{(\bar{x}, \bar{v}) = (\bar{x}^*, \bar{v}^*)} = \nabla_{\bar{x}} \Psi(x, \bar{v}) |_{(\bar{x}, \bar{v}) = (\bar{x}^*, \bar{v}^*)} = 0.$$

Consider the function $V(\bar{x}, \bar{v}) : \mathbb{R}^{mn} \times \mathbb{R}^{mn} \to \mathbb{R}_+$:

$$V(\bar{x}, \bar{v}) = \frac{1}{2}(\bar{x} - \bar{x}^*)^T(\bar{x} - \bar{x}^*) + \frac{1}{2}(\bar{v} - \bar{v}^*)^T(\bar{v} - \bar{v}^*).$$

Then

$$\frac{dV}{dt} = (\bar{x} - \bar{x}^*)^T(L\bar{v}) - (\bar{v} - \bar{v}^*)^T(L\bar{x} + (\nabla_{v^1} \phi(v^1)|_{v^1}, \ldots, \nabla_{v^n} \phi(v^n)|_{v^n}))^T.$$

Since

$$\dot{x} = -\nabla_{\bar{v}} \Psi(x; \bar{v}), \quad \dot{\bar{v}} = \nabla_{\bar{x}} \Psi(x, \bar{v}),$$

we have

$$\frac{dV}{dt} = (\bar{x} - \bar{x}^*)(\nabla_{\bar{x}} \Psi(x; \bar{v})) - (\bar{v} - \bar{v}^*)^T(\nabla_{\bar{v}} \Psi(x, \bar{v})).$$

From the property of convex function,

$$(\bar{x} - \bar{x}^*)(\nabla_{\bar{x}} \Psi(x; \bar{v})) = \Psi(x, \bar{v}) - \Psi(\bar{x}^*, \bar{v}),$$

and

$$-(\bar{v} - \bar{v}^*)^T(\nabla_{\bar{v}} \Psi(x, \bar{v})) \leq \Psi(x, \bar{v}^*) - \Psi(x, \bar{v}).$$

Thus,

$$\frac{dV}{dt} \leq \Psi(x, \bar{v}) - \Psi(\bar{x}^*, \bar{v}) + \Psi(\bar{x}, \bar{v}^*) - \Psi(x, \bar{v})$$

$$= \Psi(\bar{x}^*, \bar{v}^*) - \Psi(\bar{x}^*, \bar{v}) + \Psi(\bar{x}, \bar{v}^*) - \Psi(\bar{x}^*, \bar{v}^*) < 0$$
for $\bar{x} \neq \bar{x}^*$, while $\frac{dV}{dt} = 0$ on the set $\varepsilon := \{(\bar{x}, \bar{v})|\bar{v} = \mathbf{1}_n \otimes v^*\}$. To show that the equilibrium point is globally asymptotically stable, we invoke LaSalle Invariant Principle to show that there are no trajectories in $\varepsilon$ other than

$$\bar{x} = \bar{x}^*, \quad \bar{v} = \mathbf{1}_n \otimes v^*. $$

We have $\dot{x} = \mathbf{0}$ and $\dot{v} = -\mathbf{L}\bar{x} = 0$. Because $\mathbf{G}$ is connected, $x = \mathbf{1}_n \otimes x^*$ is the only eigenvector of $\mathbf{L}$ corresponding to the zero eigenvalue. Therefore, $x_1 = x_2 = \cdots = x_n$. Because only $(\bar{x}^*, \bar{v}^*)$ stays in the invariant set $\varepsilon$, it is globally asymptotically stable. 

The benefits of applying game methods for distributed optimization problem can be found in (Li & Marden 2013; Marden 2012; Marden et al., 2009; Young 2004). In fact, the algorithm provides a decomposition between the game design and learning algorithm design so that we can use the game learning algorithms in the optimization problems. Moreover, the inherent robustness of the game learning algorithms provides the robustness of the proposed method.

4. Design game for undirected graph

In this section, we give further discussions in the undirected communication graph case. Clearly, the design in Section 3 can also be applied to solve the optimization problem with undirected graphs. However, we can make better use of the bi-directional information flow to improve the performance of the game dynamics as it is shown in (Wang & Elia, 2009a,b).

Clearly, the payoff function provides the incentive for the agents to interact with each other. Therefore, we only need to redesign the payoff function to make agents change the behaviours when confronted with more structured information. The state-based game $\mathbf{G}_2$ is designed with the same players set, action space, state space, state dynamics, and strategy space as those for the state-based game $\mathbf{G}_1$ in Section 3, but the payoff function of each agent $i$ is redesigned as follows:

$$u_i(\bar{x}; \bar{v}) = \phi(v^i) + v^T \sum_{j \in N_i} a_{ij} (x_j - x_i) + \frac{1}{2} \sum_{j \in N_i} a_{ij} \|v^i - v^j\|^2_2. $$

(8)

Therefore, we can find the potential function for the state-based game $\mathbf{G}_2$:

$$\Phi(\bar{x}; \bar{v}) = \sum_{i \in N} \phi(v^i) + \sum_{i \in N} v^i \sum_{j \in N_i} a_{ij} (x_j - x_i) + \frac{1}{4} \sum_{i \in N} \sum_{j \in N_i} a_{ij} \|v^i - v^j\|^2_2. $$

(9)

We can verify that

$$u_i(\bar{x}; v^i, v^{-i}) - u_i(\bar{x}; \bar{v}^i, v^{-i}) = \Phi(\bar{x}; v^i, v^{-i}) - \Phi(\bar{x}; \bar{v}^i, v^{-i}), \quad \forall i \in N$$

is satisfied $\forall \bar{x} \in X, \forall v^i, \bar{v}^i \in \mathbb{R}^n$, and $\forall v^{-i} \in \prod_{j \neq i} \mathbb{R}^n$. Similar to Lemma 2 and Theorem 1, we have the following two results, whose proofs are omitted for space limitations.

**Lemma 3** With $\Phi$ defined in (9) as the potential function, the state-based game $\mathbf{G}_2$ is a state-based potential game.

**Theorem 3** Suppose the communication graph $\mathcal{G}$ is undirected and connected. Then $[\bar{x}^*, \bar{v}^*]$ is SSNE of the state-based potential game $\mathbf{G}_2$ if and only if

$$x_1 = x_2 = \cdots = x_n = x^*, \quad v^1 = v^2 = \cdots = v^n = v^*, $$

$\bar{x}^*$.
or equivalently

\[ \bar{x}^* = 1_n \otimes x^*, \quad \bar{v}^* = 1_n \otimes v^*, \]

where \( v^* \) is the optimal solution of the problem (1).

With the payoff function (8), we construct a learning algorithm for each agent \( i \) as follows:

\[
\dot{v}_i = -\nabla_{v_i} u_i(\bar{x}; v^i, v^{-i}) = -\sum_{j \in N_i} a_{ij}(v^j - v^i) - \sum_{j \in N_i} a_{ij}(x_i - x_j) - \nabla_{v_i} \phi(v^i). \tag{10}
\]

We give the following result to show the convergence and stability of the proposed algorithm, and the proof is omitted for the proof idea is similar to Theorem 2.

**Theorem 4** Consider the state-based potential game \( G_2 \) and learning algorithm (10). If \( \phi(v) \) is continuously differentiable convex function and bounded below, then all agents converge to SSNE of the state-based potential game \( G_2 \) and the SSNE is stable.

The difference between the learning dynamics of (6) and (10) is the term \(-L\bar{v}\). With the bi-directional flow of information in undirected communication graph, we can use the information \( L\bar{v} \) twice. Thus, we add proportional term to the feedback controller of the system, then the strategy becomes the well-known proportional integral (PI) controller. The signal graphs of the two dynamics are shown in the Figures 1 and 2, respectively.

Notice that the Lagrangian problem of the optimization problem (2) with Lagrange multipliers \( x_i, x_i \in \mathbb{R}^n, i = 1, \ldots, n \) is

\[
\min_{v^i, i \in \mathcal{N}} \max_{x_i, i \in \mathcal{N}} \sum_{i \in \mathcal{N}} \phi(v^i) + \bar{x}^T L \bar{v}, \tag{11}
\]

whose objective function is exactly the potential function of the game \( G_1 \). The argument Lagrangian problem of the problem (2) is

\[
\min_{v^i, i \in \mathcal{N}} \sum_{i \in \mathcal{N}} \phi(v^i) + \frac{1}{2} \bar{v}^T L \bar{v}, \quad \text{subject to} \quad L \bar{v} = 0_{mn}. \tag{12}
\]
Then the Lagrangian problem of (12) is

$$\min_{v^i, i \in N} \max_{x_i, i \in N} \sum_{i \in N} \phi(v^i) + \frac{1}{2} \bar{v}^T L \bar{v} + \bar{x}^T L \bar{v},$$

whose objective function is exactly the potential function of the game $G_2$. Thus for the undirected graph, we have applied the argument Lagrange method to redesign the algorithm.

Remark 2 Sometimes, we may have to adopt a discrete-time version for the game design and learning algorithm because of digital communication or computational concerns. One simple way to get discrete-time algorithm from our continuous-time algorithm is using Euler method with a small step-size $\beta$. For example, the discrete time algorithm for undirected case learning algorithm (10) is

$$\begin{cases} \bar{x}(k + 1) = \bar{x}(k) + \beta L \bar{v}(k), \\ \bar{v}(k + 1) = \bar{v}(k) - \beta L \bar{x}(k) - \beta L \bar{v}(k) - \beta G(\bar{v}(k)). \end{cases}$$

(14)

The choice of step-size $\beta$ will affect the stability of the system (14). Some relevant results using small gain theorem for the choice of step-size can be found in (Wang & Elia, 2010), and it was shown that it was not necessary to choose diminishing step-size to guarantee the convergence of the algorithm.

5. Application in plug-in electric vehicles and simulation example

Smart grid emerges as a new concept to meet growing challenge with the extending grid complexity and expanding renewable energy source. Plug-in electric vehicles (PIEVs) are supposed to share a great deal of future power load, which will bring benefits to the environment as well as new challenge to the future grid. In this section, we apply the proposed potential game method to the optimization of the charge station management, which might address the dynamic arrival of PIEVs, the different demands of different PIEV users, and the dynamic cost of charge station.

We propose the following optimization model for the PIEV charge station management. Suppose that one charge station has $n$ PIEVs, and each PIEV $i$ requests energy $v_i$ from the charge station. The cost of the charge station is a function of the aggregate energy requests
of all the PIEVs, which is a function of \( v = \sum_{i=1}^{n} v_i \) denoted by \( C'(v_1, v_2, \ldots, v_n) = C(v) \). Suppose that \( C(v) \) is a convex non-decreasing function. The charge station sets the energy price according to the marginal cost, that is \( p = \frac{dC(v)}{dv} \). Suppose that each user can decide a standard utility function parameterised by some parameters, which reflect the charge urgency and demand of each agent. Denote the utility function of agent \( i \) as \( f_i(v_i) \). Then the charge station management problem is modelled as:

\[
\min_{v_i, i = 1, \ldots, n} \sum_{i=1}^{n} (v_i \frac{dC(v)}{dv} - f_i(v_i)).
\] (15)

Clearly, each PIEV has to decide an energy request, which will affect the utility of all the agents, thus each agent faces an optimization problem in a distributed network. We use the state-based potential game methods to solve the optimization problem.

Remark 3 Here each agent needs to know the global optimization objective function, which is necessary for achieving global optimal solution because of the existence of loss of efficiency phenomenon in game theory (Nisam, Roughgarden, Tardos, & Vazirani, 2007). In the application to PIEVs, each agent has to know the aggregate utility function of all the agents, which may be achieved by standardising the utility functions with some parameters. When new PIEV arrives, it will broadcast its parameters to all the agents. Thus, the aggregate utility functions of all the agents can be constructed with these parameters.

Here is an illustrative example to show the efficiency of our algorithm. Suppose that the cost function of the charge station is a quadratic function and the utility function of the PIEV is also a quadratic function with parameters chosen by the users. Then the global objective function is also a quadratic function.

Example 1 Consider six agents with the following optimization problem:

\[
\min_{v} \quad v^T P v + q^T v
\] (16)

where \( v \in \mathbb{R}^6 \), \( q^T = [17, 39, 46, 90, 72, 58]^T \) and

\[
P = \begin{pmatrix}
50 & 0.2 & 3 & 4 & 5 & 6 \\
2 & 44 & 6 & 2 & 5 & 7 \\
3 & 0.16 & 49 & 2 & 3 & 4 \\
11 & 2 & 4 & 55 & 6 & 8 \\
5 & 3 & 0.4 & 0.8 & 12 & 0.7 \\
1 & 3 & 6 & 7 & 0.5 & 31
\end{pmatrix}
\] (17)

The communication graph \( G \) is weighted balanced directed graph with Laplacian matrix \( L \) as follows:

\[
\begin{pmatrix}
-0.7986 & 0.5326 & 0.1654 & 0.0004 & 0.0002 & 0.1 \\
0.0595 & -1.1182 & 0.6676 & 0.0681 & 0.1230 & 0.2 \\
0.0213 & 0.0004 & -1.2207 & 0.5809 & 0.3181 & 0.3 \\
0.0248 & 0.2458 & 0 & -1.2293 & 0.5587 & 0.4 \\
0.593 & 0.1394 & 0.0877 & 0.1799 & -1.5 & 0.5 \\
0.1 & 0.2 & 0.3 & 0.4 & 0.5 & -1.5
\end{pmatrix}
\]

The simulation result with the discretisation learning algorithms is shown in Figure 3. Moreover, the simulation result with random communication noisy and random node failure is shown in Figure 4.
6. Conclusion

In this paper, we extended the method given in (Li & Marden, 2013) by the redesign of the state and state dynamics of the state-based potential game. We showed the distributed optimization results for both directed communication graph and undirected communication graph, which relaxed the requirement of communication graph in (Li & Marden, 2013). Then, we proposed an optimization model for PIEV charge management and applied our game method to solve it. Our result may be further extended to the case with time-varying communication graph and non-smooth objective functions, which is under our investigation.

Funding

This work was supported by the NNSF of China [grant number 61174071] and by 973 Program [grant number 2014CB845301/2/3].

References

Notes on contributors

Peng Yi, PhD, is a candidate at the Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He received his bachelor degree from University of Science and Technology of China in 2011. His research interest covers distributed optimization machine learning and smart grid.

Yanqiong Zhang, PhD, is a candidate at the Academy of Mathematics and Systems Science, Chinese Academy of Sciences. She received her bachelor degree from Wuhan University in 2010. Her research interest covers distributed optimization and multi-agent systems.

Yiguang Hong is currently a professor in the Academy of Mathematics and Systems Science, Chinese Academy of Sciences. His research interests include nonlinear dynamics and control, multi-agent systems, and software systems. He served or serves as an associate editor of some journals such as IEEE Transactions on Automatic Control and IEEE Control Systems Magazine. Moreover, he is the editor-in-chief of Control Theory and Technology and a deputy editor-in-chief of Acta Automatica Sinica.