Designing Games for Distributed Optimization with a time varying communication graph

Na Li and Jason R. Marden

Abstract— **The central goal in multi-agent systems is to engineer a decision making architecture where agents make independent decisions in response to local information while ensuring that the emergent global behavior is desirable with respect to a given system level objective. Our previous work identified a systematic methodology for such a task using the framework of state based games. One core advantage of the approach is that it provides a two step process that can be decoupled by utilizing specific classes of games. Exploiting this decomposition could lead to a rich class of distributed learning algorithm. However, a drawback of our previous approach is the dependence on a time-invariant and connected communication graph. These conditions are not practical for a wide variety of multi-agent systems. In this paper we propose a new game theoretical approach for addressing distributed optimization problems that permits relaxations in the structure of the communication graph.**

I. INTRODUCTION

The goal in multi-agent systems is to establish a local interaction framework where agents make independent decisions in response to local information while ensuring that the emergent global behavior is desirable with respect to a given system level objective, e.g., [1]–[5]. These architectures possess several desirable attributes including real-time adaptation and robustness to dynamic uncertainties. However, realizing these benefits requires addressing the underlying complexity associated with a potentially large number of interacting agents and the analytical difficulties of dealing with incomplete information. Furthermore, time-varying architectures resulting from communication constraints inherent to many multi-agent systems brings additional complexity to the control algorithm design.

Game theory is a powerful tool for the design and control of multiagent systems [5]–[8]. Utilizing game theory for this purpose requires (i) modeling the agents as self-interested decision makers in a game theoretic environment with a set of choices and a local objective function; (ii) specifying a behavior rule for each agent. The goal is to complete both steps to ensure that the emergent global behavior is desirable with respect to the system level objective. One of the core advantages of game theory is that it provides a hierarchical decomposition between the distribution of the optimization problem (*game design*) and the specific local decision rules (*learning design*) [7], [9], [10]. For example, if the game is designed as a potential game [11] then there is an inherent robustness to decision making rules as a wide class of distributed learning algorithms can achieve convergence to a pure Nash equilibrium under a variety of informational dependencies [12]–[15]. Therefore, if the Nash equilibria of the designed game are also efficient with respect to the system objective, then there is a rich class of robust distributed learning algorithms for this multi-agent system.

In [10] we identified a systematic methodology for the design of local agent objective functions that satisfies virtually any degree of locality while ensuring all resulting Nash equilibria represent optimal solutions to a global optimization problem. That design paralleled the theme of distributed optimization algorithm design which can be considered as a concatenation between a designed game and a distributed learning algorithm. Well-known distributed optimization algorithms include sub-gradient methods [16]–[22], consensus based methods [1], [2], or two-step consensus based approaches [8], [23]. The core difference between our game theoretical approaches and those algorithm design approaches is that our focus is on the decomposition as opposed to a particular algorithm. Exploiting this decomposition could lead to a rich set of tools for both game design and learning design that permits a broad class of distributed learning algorithms.

However, a drawback of our previous proposed game design methodology is the dependence on a connected, undirected, and time-invariant communication graph. These conditions are not practical for a wide variety of multi-agent systems. In this paper we propose a new methodology for addressing this task that permits relaxations in the structure of the communication graph while still ensuring the efficiency of the resulting equilibria. The communication graph is allowed to be time-varying and even unconnected at frequent times.

The key enabler for this result is the same as in [10], i.e. the addition of local state variables to the game environment. These state variables are utilized as a coordinating entity to decouple the system level objective into agent specific objectives of the desired interdependence. The difference between this work and our previous work relies on the design of local objective functions. Here, the resulting game is a state based potential game [24] with a state based potential function possessing a property which is invariant to the structure of the communication graph. This is in contrast to the design in [10] where the state based potential function is dependent on the structure communication graph. This invariant property of the state based potential function allows our proposed methodology to distributively solve the global optimization problem under almost any practical setting for the time-varying communication graph. Therefore, there is no specific time-varying rule modeled in this paper for the communication graph. Our results shows that as long as the communication graph is sufficiently connected over the time, the distributed algorithm we propose will converge to the optimal solution of the global optimization problem. More

Na Li is a graduate student of Control and Dynamical Systems, California Institute of Technology, Pasadena, CA 91125, USA nali@caltech.edu

J. R. Marden is with the Department of Electrical, Computer & Energy Engineering, University of Colorado at Boulder, Boulder, CO 80309, USA jason.marden@colorado.edu

rigorous arguments will follow in the later part of the paper.

The structure of the paper is as follows. Section II introduces the problem setup and basic concepts for strategic games. Section III provides a state based game design and analyzes the properties of the designed game. Section IV proposes a distributed learning algorithm to reach the equilibria in the designed state based game which represent the optimal solution for the global optimization problem. Lastly Section V provides a simple example to illustrate our methodology and Section VI concludes the paper.

II. Preliminaries

A. Problem setup

We consider a multiagent system consisting of n agents denoted by the set $N := \{1, \cdots, n\}$. Each agent $i \in N$ is endowed with a set of possible decisions (or values) denoted by V_i which is a convex subset of $\mathbb{R}^{d_i}$, i.e. $V_i \subset \mathbb{R}^{d_i}$.[1] We denote a joint decision by the tuple $(v_1, \cdots, v_n) \in V := \prod_{i \in N} V_i$ where V is referred to as the set of joint decisions. There is a global objective of the form $\phi : \mathbb{R}^n \to \mathbb{R}$ that a system designer seeks to minimize. We assume throughout that the objective function ϕ is differentiable convex unless otherwise noted. More formally, the optimization problem takes on the form:

$$\begin{array}{ll} \min_{v_i} & \phi(v_1, v_2, \ldots, v_n) \\ s.t. & v_i \in V_i, \forall i \in N \end{array} \tag{1}$$

The goal of this paper is to establish a distributed interaction framework for this optimization problem where each agent $i \in N$ makes its decision independently in response to local information. The agents' decisions interact with each other through local communication which is defined by a communication graph. The difference between the problem considered in this paper and the one in [10] is that we now allow the communication graph between agents to be time varying. We represent the information available to each agent at time $t \in \{0, 1, ...\}$ by an undirected communication graph $\mathcal{G}(t) = \{N, \mathcal{E}(t)\}$ with nodes N and edges $\mathcal{E}(t)$. By convention, we let $(i, i) \in \mathcal{E}(t)$ for all $i \in N$ and $t \geq 0$. Define the neighbors of agent i at time t as $N_i(t) := \{j \in N : (i, j) \in \mathcal{E}(t)\}$. The distributed learning framework produces a sequence of decision $v(0), v(1), v(2), \ldots$ where at each iteration $t \in \{0, 1, \ldots\}$ the decision of each agent i is chosen independently according to a local control law of the following form

$$v_i(t) = F_i\left(\{v_j(t-1)\}_{j \in N_i(t-1)}\right).$$

Our goal is to design the local controllers $\{F_i(\cdot)\}_{i \in N}$ within the desired information constraints such that the collective behavior converges to a joint decision v^* that solves the optimization problem in (1).

B. Strategic Form Games

A strategic form game consists of a set of players $N \triangleq \{1, 2, \cdots, n\}$ where each player $i \in N$ has an action set $\mathcal{A}_i$ and a cost function $J_i : \mathcal{A} \to \mathbb{R}$ where $\mathcal{A} \triangleq \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$ is referred to as the set of joint action profiles. For an action profile $a = (a_1, \ldots, a_n)$, let a_{-i} denotes the action profile of players other than player i, i.e., $a_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$. An action profile $a^* \in \mathcal{A}$ is called a *pure Nash equilibrium* if for all $i \in N$, $J_i(a_i^*, a_{-i}^*) = \min_{a_i \in \mathcal{A}_i} J_i(a_i, a_{-i}^*)$.

III. State Based Game Design

State based games represent an extension of strategic form games where an underlying state space is introduced to the game theoretic environment [9], [10], [24]. In [10], we provided a systematic methodology for attaining a distributed solution to (1) under the requirement of a fixed communication graph. In this section we modify the state based game design to incorporate a time varying communication graph. To facilitate the understanding of our method, We begin by discussing our game design using the framework of state based game. We then proceed to define the relevant equilibrium concepts followed by analyzing the properties of our designed game.

A state based game consists of the following elements:

- an agent set N;
- a state space X;
- state dependent action sets of the form $\mathcal{A}_i(x)$ for each agent $i \in N$ and state $x \in X$;
- state dependent cost functions of the form $J_i(x, a) \in \mathbb{R}$ for each agent $i \in N$, $x \in X$, $a \in \mathcal{A}(x) \triangleq \prod_i \mathcal{A}_i(x)$;
- a state transition rule in the form of $P_t(x, a) \in \Delta(X)$ for each time t. Here $\Delta(X)$ denotes the set of probability distributions over the state space X. We express a state transition as $x(t+1) \sim P_t(x(t), a(t))$ if the probability for reaching $x(t+1)$ from $[x(t), a(t)]$ at time t is strictly positive.

To establish a distributed interaction framework for the optimization problem in (1), the objectives of our state based game design are as follows:

(i) The state represents a compilation of local state variables, i.e., the state x can be represented as $x := (x_1, \ldots, x_n)$ where each x_i represents the state of agent i.
(ii) The objective function of each agent i is local and of the form $J_i\left(\{x_j, a_j\}_{j \in N_i}\right)$
(iii) All "appropriate" Nash equilibria of the resulting game represent solutions to the optimization problem in (1).
(iv) The resulting game possesses an underlying structure, e.g. potential game [11]–[15] that can be exploited by distributed learning algorithm.

A. A state based game design for distributed optimization

State Space: The starting point of our design is an underlying state space X where each state $x \in X$ is defined as a tuple $x = (v, e, \mathcal{G})$ with the following elements:

- $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ is the profile of values.
- $e = (e_1, \ldots, e_n)$ is the profile of estimation terms where $e_i = (e_i^1, \cdots, e_i^n) \in \mathbb{R}^n$ is player i's estimation for the joint action profile v. The term e_i^k captures player i's estimate of player k's actual value v_k. The estimation terms are introduced as a means to relax the degree of information available to each agent.
- $\mathcal{G}$ is the undirected communication graph. We represent the communication graph as $\mathcal{G} = (N_1, N_2, \ldots, N_n)$ where N_i is the neighbor sets of agent i.

[1]For easa of exposition, we let $d_i = 1$ for all $i \in N$. The results in this paper hold for cases where $d_i > 1$. Moreover, d_i can be different from d_j if $i \neq j$.

Action Sets: Each agent i is assigned an action set $\mathcal{A}_i$ that permits agents to change their value and change their estimation through communication with neighboring agents. Specifically, an action for agent i is defined as a tuple $a_i = (\hat{v}_i, \hat{e}_i)$ where

- $\hat{v}_i \in \mathbb{R}$ indicates a change in the agent's value v_i
- $\hat{e}_i := \{\hat{e}^k_{i\to j}\}^{k\in N}_{j\in N}$ indicates a change in the agent's estimation terms e_i where $\hat{e}^k_{i\to j} \in \mathbb{R}$ represents the estimation value that player i passes to player j regarding to the value of player k.

Since a player is only allowed to communicate with its neighbors, the admissible actions for $\hat{e}_i$ given the state x is

$$\mathcal{A}^e_i(x) := \left\{\{\hat{e}^k_{i\to j}\}^{k\in N}_{j\in N} : \hat{e}^k_{i\to j} = 0, \forall j \notin N_i, k \in N\right\}.$$

Here 0 means that player i does not pass any estimation to player j.

State Transition Rules: We now describe how the state involves.

- The evolution of the value profile v is captured by a time-invariant, deterministic, and local state transition rule of the form:
$$P^v_t(x,a) = P^v(x,a) = \{v_i + \hat{v}_i\}_{i\in N} \quad (2)$$
- The evolution of the estimation profile e is also captured by a time-invariant, deterministic, and local state transition rules of the form:
$$P^e_t(x,a) = P^e(x,a) = \{e^k_i + n\delta^k_i \hat{v}_i + \check{e}^k_i\}_{i,k\in N} \quad (3)$$
where $\check{e}^k_i \triangleq \sum_{j\in N_i} \hat{e}^k_{j\to i} - \sum_{j\in N_i} \hat{e}^k_{i\to j}$ and δ^k_i is an indicator function, i.e., $\delta^i_i = 1$ and $\delta^k_i = 0$ for all $k \neq i$.
- The state transition for the communication graph $\mathcal{G}$ is given as $P^{\mathcal{G}}_t : X\times\mathcal{A} \to \Delta(\bar{\mathcal{G}})$ at each time t. Here $\bar{\mathcal{G}}$ denotes the set of all undirected communication graph and $\Delta(\bar{\mathcal{G}})$ denotes the set of probability distributions over this set. In practice different applications would have different evolution rule $P^{\mathcal{G}}_t$. To ensure the generality of our model, we do not assign any specific evolution rule for $P^{\mathcal{G}}_t$ and later we will show that as long as the undirected $\mathcal{G}(t)$ is connected sufficiently over the time, our approach can solve the optimization problem (1). Notice that since the state transition rule $P^{\mathcal{G}}_t$ is allowed to be time-dependent/variant, the evolution rule $P^{\mathcal{G}}_t$ can also model the situation where the graph transition is determined/affected by exogenous disturbances.

Notice that each agent i can update its own local state (v_i, e_i) using local state and action information through Equation (2,3). Since the optimization problem in (1) imposes the requirement that $v_i \in V_i$, we condition the available actions for $\hat{v}_i$ to an agent i on the current state $x = (v, e)$ as:

$$\mathcal{A}^v_i(x) := \{\hat{v}_i : v_i + \hat{v}_i \in V_i\} \quad (4)$$

The admissible action set is defined as $\mathcal{A}_i(x) \triangleq \mathcal{A}^v_i(x) \times \mathcal{A}^e_i(x)$.

Invariance Property of State Dynamics: Let $v(0) = (v_1(0), ..., v_n(0))$ be the initial values of the agents. Define the initial estimation terms $e(0)$ to satisfy $\sum_{i\in N} e^k_i(0) = n\cdot v_k(0)$ for each agent $k \in N$; hence, the initial estimation values are contingent on the initial values. Note that satisfying this condition is trivial as we can set $e^i_i(0) = n\cdot v_i(0)$ and $e^j_i(0) = 0$ for all agents $i, j \in N$ where $i \neq j$. Define the initial state as $x(0) = (v(0), e(0), \mathcal{G}(0))$. It is straightforward to show that for any action trajectory $a(0), a(1), \cdots$,, the resulting state realization $x(t+1) \sim P_t(x(t), a(t))$ satisfies the following equalities for all times $t \geq 1$ and agents $k \in N$:

$$\sum_{i=1}^{n} e^k_i(t) = n \cdot v_k(t) \quad (5)$$

Agent Cost Functions: The introduced cost functions possess two distinct components and take on the form

$$J_i(x,a) = J^{\phi}_i(x,a) + \alpha \cdot J^e_i(x,a) \quad (6)$$

where $J^{\phi}_i(\cdot)$ represents the component centered on the objective function ϕ; $J^e_i(\cdot)$ represents the component centered on the state x; and α is a positive constant representing the trade-off between the two components.[2] We define each of these components as follows: for any state $x \in X$ and admissible action profile $a \in \prod_{i\in N} \mathcal{A}_i(x)$ we define

$$\begin{array}{rcl} J^{\phi}_i(x,a) &=& \sum_{j\in N_i} \phi(\tilde{e}^1_j, \tilde{e}^2_j, \ldots, \tilde{e}^n_j) \\ J^x_i(x,a) &=& \sum_{j\in N_i} \sum_{k\in N} \left(\tilde{e}^k_j\right)^2 - n\left(\tilde{v}_i\right)^2 \end{array} \quad (7)$$

where $\tilde{v} = P^v(x,a)$ and $\tilde{e} = P^e(x,a)$. The local cost function in (7) is the main difference between the design in the paper and the design in [10]. The rest of the paper shows that the new local cost function design allows us to deal with time-varying communication graphs.

B. Analytical properties of designed game

Before analyzing the properties of the designed game, we introduce one core equilibrium concept that we will use in this paper. Define a state set $\bar{X}(x^0, a^0)$ as the set of all possible ensuring states from the state action pair $[x^0, a^0]$:

$$\begin{array}{rl} \bar{X}(x^0,a^0) \triangleq \{ & x = (v, e, \mathcal{G}) : \\ & v = P^v(x^0, a^0), \\ & e = P^e(x^0, a^0), \\ & \mathcal{G} \text{ is an undirected graph}\} \end{array}$$

Definition 1. *(Stationary Nash Equilibrium) A state pair $[x^*, a^*]$ is a stationary Nash equilibrium if*
(D-1): for any $x \in \bar{X}(x^*, a^*)$:

$$a^*_i \in \mathcal{A}_i(x) \textit{ and } a^*_i \in \mathrm{argmin}_{a_i\in\mathcal{A}_i(x)} J_i(x, a_i, a^*_{-i}).$$

(D-2): $x^* \in \bar{X}(x^*, a^*)$.

The first condition is similar to Nash equilibrium concept and the second condition requires that the state component v and e are stationary. As the structure of the graph transition rule $P^{\mathcal{G}}_t$ can be very general, in the definition of $\bar{X}[x^*, a^*]$ we include all the undirected graph as possible ensuing communication graph. The two conditions implies that stationary Nash equilibria represent fixed points of the better reply process for state based games under any communication graph transition rule $P^{\mathcal{G}}_t$. That is, if a state action pair at time t, i.e., $[x(t), a(t)]$ is a stationary Nash equilibrium, then

[2] We will show that as long as α is positive, all the results demonstrated in this paper hold. However, choosing the right α is important for the learning algorithm implementation.

$a(\tau) = a(t)$ for all time $\tau \geq t$ if all players adhere to a better reply process. The following theorem demonstrates that *all* stationary Nash equilibria of our designed game are solutions to the optimization problem (1).

Theorem 1. *Model the optimization problem in (1) as a state based game G as depicted in Section III-A with any positive constant α. Then a state action pair $[x, a] := [(v, e, \mathcal{G}), (\hat{v}, \hat{e})]$ is a stationary Nash equilibrium in game G if and only if the following conditions are satisfied:*

(i) The value profile v is optimal for problem (1);
(ii) The estimation profile e satisfies that $e_i^k = v_k$, $\forall i, k \in N$;
(iii) The change in value satisfies $\hat{v}_i = 0$, $\forall i \in N$;
(iv) The change in estimation satisfies $\hat{e}_{i\to j}^k = 0$, $\forall i, j, k \in N$.

The above theorem demonstrates that the resulting equilibria of our state based game coincide with the optimal solutions to the optimization problem in (1). Moreover, from this theorem, it is straightforward to derive the following corollary:

Corollary 2. *If a station action pair $[x^*, a^*] \triangleq [(v^*, e^*, \mathcal{G}^*), a^*]$ is a stationary Nash equilibrium, then any station action pair $[(v^*, e^*, \mathcal{G}), a^*]$ is a stationary Nash equilibrium for any undirected graph $\mathcal{G}$.*

The following theorem demonstrate that the designed game possesses a property similar with potential games that facilitates the design of learning rules to reach such stationary Nash equilibrium.

Theorem 3. *Model the optimization problem in (1) as a state based game G as depicted in Section III-A with any positive constant α. The following function $\Phi : X \times \mathcal{A} \to \mathbb{R}$*

$$\Phi(x, a) = \Phi^{\phi}(x, a) + \alpha \cdot \Phi^{x}(x, a) \tag{8}$$

where

$$\Phi^{\phi}(x, a) = \sum_{i \in N} \phi(\tilde{e}_i^1, \tilde{e}_i^2, ..., \tilde{e}_i^n) \tag{9}$$

$$\Phi^{x}(x, a) = \sum_{i \in N} \sum_{k \in N} \left(\tilde{e}_i^k\right)^2 - n \cdot \sum_{i \in N} \left(\tilde{v}_i\right)^2 \tag{10}$$

$$\tilde{v} = P^v(x, a) \text{ and } \tilde{e} = P^e(x, a),$$

satisfies the following two properties:

1) For every state action pair $[x, a]$ any player $i \in N$ any action $a_i' \in \mathcal{A}_i(x)$,

$$J_i(x, a_i', a_{-i}) - J_i(x, a) = \Phi(x, a_i', a_{-i}) - \Phi(x, a)$$

2) For every state action pair $[x, a]$ and any $\tilde{x} \in \bar{X}(x, a)$, we have $\Phi(x, a) = \Phi(\tilde{x}, \mathbf{0})$ where $\mathbf{0}$ is a null action given as $\hat{v}_i = 0, \hat{e}_{i\to j}^k = 0$ for any $i, k \in N$.

Moreover, $\Phi(x, a)$ is a convex function over $a = (\hat{v}, \hat{e})$.

Property 1) and 2) of this theorem demonstrate that the function $\Phi(x, a)$ satisfies the properties of a state based potential function defined in our previous paper [9], [10]. Thus we call this game a state based potential game and $\Phi(x, a)$ a state based potential function.

Notice that $\Phi(x, \mathbf{0})$ is independent of the communication graph $\mathcal{G}$. Therefore, even though the communication graph $\mathcal{G}$ is time varying, Theorem 3 establishes that our state based game design possesses an underlying structure that facilitates the design of distributed algorithms to reach stationary Nash equilibria. In the next section, we provide a distributed learning algorithm to reach those stationary Nash equilibria that were characterized in Theorem 1.

IV. Gradient Play

Since the state based potential function $\Phi(x, a)$ is a convex function over $a = (\hat{v}, \hat{e})$, we can apply gradient play algorithm [9], [10] to develop a distributed learning algorithm for the state based game depicted in section III. In this section, we assume that $\mathcal{V}_i$ is a closed convex set for all $i \in N$. The gradient play algorithm is given as follows:

1) Each agent i initially random choose a value $v_i(0)$ and set $e_i^i = nv_i(0)$ and $e_i^k(0) = 0$ for all $k \neq i$. Set t=0;
2) At each time $t \geq 0$ each agent i selects an action $a_i(t) \triangleq (\hat{v}_i(t), \hat{e}_i(t))$ given the state $x(t) = (v(t), e(t), \mathcal{G}(t))$ according to:

$$\begin{aligned} \hat{v}_i(t) &= \left[-\epsilon \cdot \frac{\partial J_i\left(x(t), a\right)}{\partial \hat{v}_i}\bigg|_{a=0}\right]^+ \qquad (11) \\ &= \left[-\epsilon(n\ \phi_i|_{e_i(t)} + 2n\alpha(e_i^i(t) - v_i(t))\right]^+ \\ \hat{e}_{i\to j}^k(t) &= -\epsilon\ \frac{\partial J_i\left(x(t), a\right)}{\partial \hat{e}_{i\to j}^k}\bigg|_{a=0} \qquad (12) \\ &= \epsilon\left(\phi_k|_{e_i(t)} - \phi_k|_{e_j(t)} + 2\alpha\left(e_i^k(t) - e_j^k(t)\right)\right) \end{aligned}$$

where $[\cdot]^+$ represents the projection onto the closed convex set $\mathcal{A}_i^{\hat{v}}(x)$; and ϵ is the stepsize which is a positive constant. Notice that each agent i can selects its own action using local information since $J_i(\cdot)$ only depends on local information.
3) Each agent i updates the local state $(v_i(t+1), e_i(t+1))$ according to Equation (2) and (2) using its own local information. The communication graph $\mathcal{G}(t)$ is realized according to $P_t^{\mathcal{G}}$.
4) Increase t by 1 and return to step 2.

The following theorem establishes the convergences of the gradient play.

Theorem 4. *Suppose there exist an integer $k > 0$ such that the undirected communication graph $G(t)$ is connected for at least one time step $t \in [nk, nk + k - 1]$ for all $n \geq 0$. If the step-size is sufficiently small, and the sequence $(v(1), e(1)), (v(2), x(2)), \cdots$ produced by the gradient play algorithm is contained in a compact subset of $\mathbb{R}^{2n}$, then $[v(t), e(t), a(t)]$ in the gradient play algorithm asymptotically converges to $[(v^*, e^*), \mathbf{0}]$ where $[(v^*, e^*, \mathcal{G}), \mathbf{0}]$ is a stationary Nash equilibrium with any graph $\mathcal{G}$.*

Proof: Notice that $\Phi(x(t), \mathbf{0})$ is independent of $\mathcal{G}(t)$, therefore we can write $\Phi(x(t), \mathbf{0})$ as $\Phi(v(t), e(t), \mathbf{0})$. Then we can show that $\Phi(x(t), \mathbf{0})$ is monotonically decreasing along the gradient play algorithm. The proof of the convergence follows exactly the same as the proof for Theorem 4 in [9]. We omit the details here. □

Combining with Theorem 1, Theorem 4 demonstrates that gradient play algorithm provides a distributed learning algorithm to solve the optimization problem in (1).

Remark 1. *The theorem requires a strong condition on the undirected communication graph $\mathcal{G}(t)$, i.e. it is connected* *frequently enough, the results can be extended to more general cases. For example, it can be shown that if there exist a finite $k \geq 0$ such that $\cup_{t=\tau k}^{\tau k+k-1} \mathcal{G}(t) \triangleq (N, \cup_{t=\tau k}^{\tau k+k-1} \mathcal{E}(t))$ is connected for all $\tau \geq 0$, then the gradient play algorithm will converge to the stationary Nash equilibrium. As a non-rigorous statement, as long as the union of $\mathcal{G}(t)$ over a finite time horizon is connected frequently enough, the gradient play algorithm will converge to a stationary Nash equilibrium.*

V. Illustrations

We will use a simple abstract example to illustrate the problem and the method. Consider the following optimization problem:

$$\min_{v_1,\ldots,v_5} \quad v^T P v + q^T v$$
$$\text{s.t.} \quad v_i \in [0, i] \subset \mathbb{R}$$

where $q^T = -[9\ 9\ 9\ 9\ 9]$ and

$$P = \begin{bmatrix} 6 & 1 & 1 & 1 & -1 \\ 1 & 7 & 1 & -1 & 2 \\ 1 & 1 & 8 & 2 & -2 \\ 1 & -1 & 2 & 9 & 3 \\ -1 & 2 & -2 & 3 & 9 \end{bmatrix}$$

The goal is to establish a local control law for each agent i that converges to the optimal value v_i^*. One possibility for a distributed algorithm is to utilize a gradient decent algorithm where each agent adjust their own value according to $\frac{\partial \phi}{\partial v_i} = 2\sum_{j=1}^{5} P(i,j)v(j) + q(i)v(i)$. As P is full-itemized matrix, implementing this algorithm requires each agent to have complete information regarding the decision of all other agents.

Using the method developed in this paper, we localize the information available to each agent by allowing them to have estimates of other agents' decision value. We simulate the gradient play algorithm with a time varying communication graph. In the simulation, at each time $t \geq 0$, each the communication link (i,j) is drawn randomly with a certain probability. Figure 1 illustrates the results of the gradient play algorithm and an animation movie of the simulation is available to download on website [25] which shows the algorithm dynamics and the communication graph dynamics as well. The top figure in Figure 1 shows the evolution of the cost $\phi(v)$ using the true gradient decent algorithm (red) and our proposed gradient play algorithm (blue). The figure shows that the convergence rate is comparable to the centralized gradient descent algorithm. Also we can notice that $\phi(v(t))$ for our distributed algorithm is not monotonically decreasing. This is reasonable since the gradient play only guarantees the potential function $\Phi(x(t), \mathbf{0})$ monotonically decreasing. This is confirmed in the middle figure of Figure 1 which shows the evolution of the state based potential function $\Phi(x(t), \mathbf{0})$. The bottom figure shows the evolution of agent i's estimation error as to agent 1's true value, i.e., $e_i^1 - v_1$. Note that the error converges to 0 illustrating that the agent's estimate converge to the right values as proved in Theorem 1 and 4.

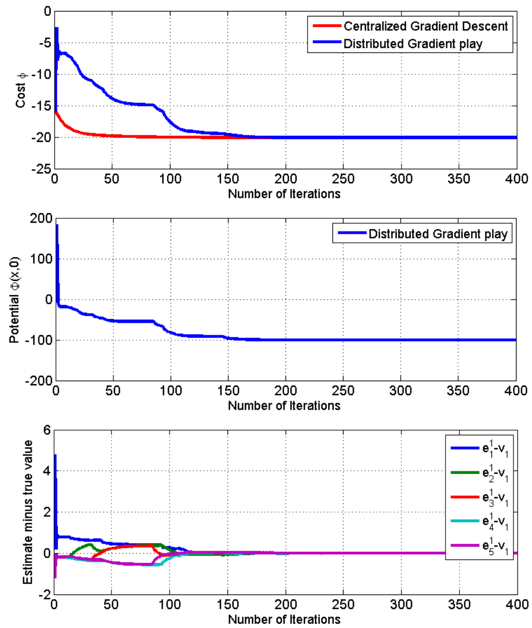

Fig. 1. Simulation results: The top figure shows the evolution of the system cost using the true gradient decent algorithm (red) and our proposed algorithm (black). The middle figure shows the evolution of the state based potential function $\Phi(x(t), \mathbf{0})$. The bottom figure shows the evolution of agent i's estimation error as to agent 1's true value, i.e., $e_i^1 - v_1$. Note that the error converges to 0 illustrating that the agent's estimate converge to the right values as proved in Theorem 1 and 4.

VI. Conclusions

We utilize the framework of state based potential games to develop a systematic methodology for distributed optimization with a time-varying communication graph. This work, along with previous work [9], [10], demonstrates the framework of state based potential games leads to a value hierarchical decomposition that can be an extremely powerful for the design and control of multiagent systems.

References

[1] J. Tsitsiklis and M. Athans. Convergence and asymptotic agreement in distributed decision problems. *Automatic Control, IEEE Transactions on*, 29(1):42–50, 1984.

[2] R. Olfati-saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. In *Proceedings of the IEEE*, volume 95, January 2007.

[3] A. Tang, J. Wang, S. H. Low, and M. Chiang. Equilibrium of heterogeneous congestion control: Existence and uniqueness. *IEEE/ACM Transactions on Networking*, 15(4):824–837, October 2007.

[4] V. Mhatre, K. Papagiannaki, and F. Baccelli. Interference mitigation through power control in high density 802.11. In *Proceedings of INFORCOM*, 2007.

[5] G. Scutari, D. P. Palomar, and J. Pang. Flexible design of cognitive radio wireless systems: from game theory to variational inequality theory. *IEEE Signal Processing Magazine*, 26(5):107–123, September 2009.

[6] L. Chen, S. H. Low, and J. C. Doyle. Random access game and medium access control design. *IEEE/ACM Transactions on Networking*, 2010.

[7] R. Gopalakrishnan, J. R. Marden, and A. Wierman. An architectural view of game theoretic control. *ACM SIGMETRICS Performance Evaluation Review*, 38(3):31–36, 2011.

[8] V. Reddy, S. Shakkottai, A. Sprintson, and N. Gautam. Multipath wireless network coding: a population game perspective. In *Proceedings IEEE INFOCOM*, pages 1–9. IEEE, 2010.

[9] N. Li and J. R. Marden. Decouple coupled constraints through utility design. *Under submission*, 2011. Available on `http://www.cds.caltech.edu/~nali/GameDisCoup.pdf`.

[10] N. Li and J. R. Marden. Design games for distributed optimization. In *50th IEEE Conference on Decision and Control*, 2011.

[11] D. Monderer and L.S. Shapley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.

[12] D. Fudenberg and D. K. Levine, editors. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

[13] H. P. Young. *Strategic Learning and Its Limits*. Oxford University Press, Oxford, UK, 2004.

[14] J.R. Marden, G. Arslan, and J.S. Shamma. Joint strategy fictitious play with inertia for potential games. *Automatic Control, IEEE Transactions on*, 54(2):208–220, 2009.

[15] J. R. Marden, H. P. Young, G. Arslan, and J. S. Shamma. Payoff based dynamics for multi-player weakly acyclic games. *SIAM Journal on Control and Optimization*, 48:373–396, February 2009.
[16] M. V. Solodov. Incremental gradient algorithms with stepsizes bounded away from zero. *Computational Optimization and Applications*, 11(1):23–35, 1998.
[17] D. Blatt, A.O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2008.
[18] J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *Automatic Control, IEEE Transactions on*, 31(9):803–812, 1986.
[19] A. Nedic, A. Olshevsky, A. Ozdaglar, and J.N. Tsitsiklis. On distributed averaging algorithms and quantization effects. *Automatic Control, IEEE Transactions on*, 54(11):2506–2517, 2009.
[20] I. Lobel and A. Ozdaglar. Distributed subgradient methods for convex optimization over random networks. *IEEE Transactions on Automatic Control*, (99):1–1, 2010.
[21] I. Lobel, A. Ozdaglar, and D. Feijer. Distributed multi-agent optimization with state-dependent communication. *Mathematical Programming*, 129(2):255–284.
[22] M. Zhu and S. Martinez. On distributed convex optimization under inequality and equality constraints. *IEEE Transactions on Automatic Control*, 57(1):151–164, 2012.
[23] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *working paper, Stanford Univ*, 2010.
[24] J. R. Marden. State based potential games. *under submission*, 2011.
[25] Simulation animation movie. Available on `http://www.cds.caltech.edu/~nali/CDC2012.avi`.

APPENDIX

A. Proof of Theorem 1

Proof: Firstly we prove that the two conditions in Definition 1 of a stationary Nash equilibrium, (i) $a_i \in \mathcal{A}_i(x')$ for any $x' \in \bar{X}(x,a)$ and (ii) $x \in \bar{X}(x,a)$, are equivalent to Condition (iii) and (iv) in Theorem 1, i.e. action a is a null action $\mathbf{0}$. For one direction, it is straightforward to show that if the action a is a null action $\mathbf{0}$, then $a_i \in \mathcal{A}_i(x')$ for any $x' \in \bar{X}(x,a)$ and $x \in \bar{X}(x,a)$. For the other direction, it is also can be shown that if the action $\hat{v} \neq 0$ then $x \notin \bar{X}(x,a)$ and if the action $\hat{e} \neq 0$, $a_i \notin \mathcal{A}_i(x')$ for some x'.

Now notice that $a = \mathbf{0}$ ensures that the ensuing value profile $P^v(x,a) = v$ and the ensuing estimate profile $P^e(x,a) = e$, which means that

$$\bar{X}(x,\mathbf{0}) = \{(v,e,\mathcal{G}') : \mathcal{G}' \text{ is an undirected graph}\}.$$

Therefore the rest of the proof only need to show that the condition in Definition 1, $0 \in \text{argmin}_{a_i' \in \mathcal{A}_i(x')} J_i(x', a_i', 0)$ for any $x' \in \bar{X}(x,\mathbf{0})$, is equivalent to that (v,e) satisfies Condition (i) and (ii) in Theorem 1. Given a state $x' \in \bar{X}(x,\mathbf{0})$, the condition $0 \in \text{argmin}_{a_i' \in \mathcal{A}_i(x')} J_i(x', a_i', 0)$ is equivalent to:

$$\left[\frac{\partial J_i(x,a_i',a_{-i}=0)}{\partial \hat{v}_i}\bigg|_{a_i'=0}\right] \cdot (\hat{v}_i' - 0) \geq 0, \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x)$$
$$\frac{\partial J_i(x,a_i',a_{-i}=0)}{\partial \hat{e}_i^k}\bigg|_{a_i'=0} = 0, \forall i,k \in N,$$

The two equations are equivalent to

$$\left[\phi_i|_{e_i} + 2\alpha \cdot n e_i^i - 2\alpha \cdot n v_i\right] \cdot (v_i' - v_i) \geq 0, \quad \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x). \tag{13}$$
$$\phi_k|_{e_i} - \phi_k|_{e_j} - \left(2\alpha(e_j^k - e_i^k)\right) = 0, \quad \forall i,k \in N,\ j \in N_i, \tag{14}$$

Therefore, the rest of the proof only need to show that two Equations (14, 13) are equivalent to Condition (i) and (ii) in this theorem.

($\Leftarrow$) If (v,e) satisfies conditions (i) and (ii), we have:

$$[\phi_i|_v] \cdot (v_i' - v_i) \geq 0, \quad \forall i \in N,\ v_i' \in \mathcal{V}_i, \tag{15}$$
$$e_i^k = v_k, \qquad \forall i,k \in N. \tag{16}$$

Equation (16) tells that equation (14) is satisfied. Subsitituting equation (16) into equation (15), we know that equation (13) is satisfied. Therefore, both Equation (13) and Equation (14) are satisfied.

($\Rightarrow$) Now we prove the other direction. Suppose (v,e) satisfy Equation (14, 13). Focus on equation (14) first. Applying Lemma 5 in the following, equation (14) coupled with the fact that ϕ is a convex function implies that for any pair $i \in N, j \in N_i, \tilde{e}_i = \tilde{e}_j$.

Given an connected and undirected graph $\mathcal{G}$, we know that $e_i = e_j$ for all $i,j \in N$. Applying equality (5), we have $e_i^k = v_k, \forall i,k \in N$, i.e. (v,e) satisfies Condition (i) listed in the theorem. Subsitituting this equality into equation (13), we have

$$[\phi_i|_v] \cdot (\tilde{v}_i' - v_i) \geq 0, \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x) \tag{17}$$

Since ϕ is a convex function, this tells us that v is an optimal solution for problem (1). □

Lemma 5. *Given a convex function $\phi(x_1, x_2, \ldots, x_n)$ and two vectors $x := (x_1, \ldots, x_n)$ and $y := (y_1, \ldots, y_n)$, if for any $k = 1, 2, \ldots n$, we have $\phi_k|_x - \phi_k|_y = \alpha_k(y_k - x_k)$ where $\alpha_k > 0$, then $x = y$.*

In our previous paper [10], we have proved this lemma. However, there is a mistake in the proof. We provide a correct proof here.

Proof: Since ϕ is a convex function,

$$\begin{aligned} \phi(x) &\geq \phi(y) + (x-y)^T \nabla\phi|_y \\ \phi(y) &\geq \phi(x) + (y-x)^T \nabla\phi|_x \end{aligned}$$

Adding up the two inequalities gives $0 \geq (x-y)^T(\nabla\phi|_y - \nabla\phi|_x)$. Since $\phi_k|_x - \phi_k|_y = \alpha(y_k - x_k)$ for all k,

$$0 \geq \sum_k \alpha_k (x_k - y_k)^2 \geq 0$$

Therefore $x = y$. □

B. Proof of Theorem 3

Proof: It is straightforward to verify that $\Phi(x,a)$ defined in Equation (8) satisfies Properties 1) and 2) in Theorem 3. So we only need to prove that $\Phi(x,a)$ is a convex function over $a = (\hat{v}, \hat{e})$. Substituting equality (5) into $\Phi^x(x,a)$ which is defined in equation (8),

$$\begin{aligned} \Phi^x(x,a) &= \sum_{k\in N}\sum_{i \in N}\left(\tilde{e}_i^k\right)^2 - n\sum_{k \in N}\left(\frac{\sum_{i\in N}\tilde{e}_i^k}{n}\right)^2 \\ &= \sum_{k\in N}\left(\sum_{i\in N}\left(\tilde{e}_i^k\right)^2 - \frac{1}{n}\left(\sum_{i\in N}\tilde{e}_i^k\right)^2\right) \\ &= \frac{1}{n}\sum_{k\in N}\left(\sum_{i,j\in N, j<i}\left(\tilde{e}_i^k - \tilde{e}_j^k\right)^2\right) \end{aligned}$$

Therefore $\Phi^x(x,a)$ is a convex function of $\tilde{e}$. Since Φ^ϕ is also a convex function of $\tilde{e}$, $\Phi(x,a)$ is a convex function of $\tilde{e}$ as well. Thus $\Phi(x,\hat{a})$ is a convex function over $a = (\hat{v}, \hat{e})$ for $\tilde{e}$ is a linear function of $(\hat{v}, \hat{e})$. □