Achieving Pareto Optimality Through Distributed Learning

Jason R. Marden, H. Peyton Young, and Lucy Y. Pao

Abstract

We propose a simple payoff-based learning rule that is completely decentralized, and that leads to an efficient configuration of actions in any n-person game with generic payoffs. The algorithm requires no communication. Agents respond solely to changes in their own realized payoffs, which are affected by the actions of other agents in the system in ways that they do not necessarily understand. The method can be applied to the optimization of complex systems with many distributed components, such as the routing of information in networks and the design and control of wind farms.

I. Introduction

Game theory has important applications to the design and control of multiagent systems [1]–[9]. This design choice requires two steps. First, the system designer must model the system components as "agents" embedded in an interactive, game-theoretic environment. This step involves defining a set of choices and a local objective function for each agent. Second, the system designer must specify the agents' behavioral rules, i.e., the way in which they react to local conditions and information. The goal is to complete both steps in such a way that the agents' behavior leads to desirable system wide behavior even though the agents themselves do not have access to the information needed to determine the state of the system.

The existing literature primarily focuses on distributed learning algorithms that are suitable for implementation in large scale engineering systems [2], [3], [10]–[13]. Accordingly, most of

This research was supported by AFOSR grant #FA9550-09-1-0538 and by ONR grant #N00014-09-1-0751.

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

H. Peyton Young is with the Department of Economics, University of Oxford, Manor Road, Oxford OX1 3UQ, United Kingdom, peyton.young@nuffield.ox.ac.uk.

Lucy Y. Pao is with the Department of Electrical, Computer, and Energy Engineering, University of Colorado, Boulder, CO 80309, pao@colorado.edu.

the results focus on particular classes of games, notably potential games [14], that are pertinent to distributed engineering systems. The motivation for this previous work stems from the fact that the interaction framework for a distributed engineering system can often be represented as a potential game. Consequently, these distributed learning algorithms can be utilized as distributed control algorithms that provide strong asymptotic guarantees on the emergent global behavior [5]–[7], [15], [16]. This approach provides a hierarchical decomposition in the design (*game design*) and control (*learning rule*) of a multiagent system where the intermediate layer is constrained by the potential game structure [5].

There are two limitations to this framework however. First, most results in this domain focus on convergence to Nash equilibrium, which may be very inefficient is achieving the system level objective. Characterizing this inefficiency is a highly active research area in algorithmic game theory [17]. The second limitation of this framework is that it is frequently impossible to represent the interaction framework of a given system as a potential game. This stems from the fact that a given engineering system possesses inherent constraints on the types of objective functions that can be assigned to the agents. These constraints are a byproduct of the information available to different components of the system. Furthermore, in many complex systems the relationship between the behavior of the components and the overall system performance is not known with any precision.

One example of a system that exhibits these challenges is the control of a wind farm to maximize total power production. Controlling an array of turbines in a wind farm is fundamentally more challenging than controlling a single turbine. The reason is the aerodynamic interactions amongst the turbines, which render many of the single turbine control algorithms *highly inefficient* for optimizing total power production [18]. Here the goal is to establish a *distributed* control algorithm that enables the individual turbines to adjust their behavior based on local conditions, so as to maximize total system performance. One way to handle this large-scale coordination problem is to model the interactions of the turbines in a game theoretic environment. However, the space of admissible utility functions for the individual turbines is limited because of the following informational limitations:

(i) No turbine has access to the actions[1] of other turbines, due to the lack of a suitable communication system;

(ii) No turbine has access to the functional relationship between the total power generated and the action of the other turbines. The reason is that the aerodynamic interaction between the turbines is poorly understood from an engineering standpoint.

These limitations restrict the ability of the designer to represent the interaction framework as a potential game. For example, one of the common design approaches is to assign each turbine an objective function that measures the turbine's marginal contribution to the power production of the wind farm, that is, the difference between the total power produced when the turbine is active and the total power produced when the turbine is inactive [6], [15]. This assignment ensures that the resulting interaction framework is a potential game and that the action profile which optimizes the potential function also optimizes the total power production of the wind farm. Calculating this marginal contribution may not be possible due to lack of knowledge about the aerodynamic interactions, hence the existing literature does not provide suitable control algorithms for this situation.

The contribution of this paper is to demonstrate the existence of simple, completely decentralized learning algorithms that lead to efficient system-wide behavior *irrespective* of the game structure. We measure the efficiency of an action profile by the sum of the agents' utility functions. In a wind farm this sum is precisely equal to the total power generated. Our main result is the development of a simple payoff-based learning algorithm that guarantees convergence to an efficient action profile whenever the underlying game has generic payoffs. This result holds whether or not this efficient action profile is a Nash equilibrium. It therefore differs from the approach of [13], which shows how to achieve constrained efficiency *within* the set of Nash equilibrium outcomes.

II. Background

Let G be a finite strategic-form game with n agents. The set of agents is denoted by $N := \{1, ..., n\}$. Each agent $i \in N$ has a finite action set $\mathcal{A}_i$ and a utility function $U_i : \mathcal{A} \rightarrow \mathbb{R}$,

[1]A turbine's action is called an *axial induction factor*. The axial induction factor indicates the amount of power the turbine extracts from the wind.

where $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ denotes the joint action set. We shall henceforth refer to a finite strategic-form game simply as "a game." Given an action profile $a = (a_1, a_2, ..., a_n) \in \mathcal{A}$, let a_{-i} denote the profile of agent actions *other than* agent i, that is., $a_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$. With this notation, we shall sometimes denote a profile a of actions by (a_i, a_{-i}) and $U_i(a)$ by $U_i(a_i, a_{-i})$. We shall also let $\mathcal{A}_{-i} = \prod_{j \neq i} \mathcal{A}_j$ denote the set of possible collective actions of all agents other than agent i. The *welfare* of an action profile $a \in \mathcal{A}$ is defined as

$$W(a) = \sum_{i \in N} U_i(a).$$

An action profile that optimizes the welfare will be denoted by $a^{\text{opt}} \in \arg\max_{a \in \mathcal{A}} W(a)$.

A. Repeated Games

We shall assume that a given game G is repeated one each period $t \in \{0, 1, 2, \ldots\}$. In period t, the agents simultaneously choose actions $a(t) = (a_1(t), ..., a_n(t))$ and receive payoffs $U_i(a(t))$. Agent $i \in N$ chooses the action $a_i(t)$ according to a probability distribution $p_i(t) \in \Delta(\mathcal{A}_i)$, which is the simplex of probability distributions over $\mathcal{A}_i$. We shall refer to $p_i(t)$ as the *strategy* of agent i at time t. We adopt the convention that $p_i^{a_i}(t)$ is the probability that agent i selects action a_i at time t according to the strategy $p_i(t)$. An agent's strategy at time t relies only on observations from times $\{0, 1, 2, ..., t-1\}$.

Different learning algorithms are specified by the agents' information and the mechanism by which their strategies are updated as information is gathered. Suppose, for example, that an agent knows his own utility function and is capable of observing the actions of all other agents at every time step but does not know their utility functions. Then the strategy adjustment mechanism of a given agent i can be written in the form

$$p_i(t) = F_i\big(a(0), ..., a(t-1); U_i\big).$$

Such an algorithm is said to be *uncoupled* [19], [20].

In this paper we ask whether agents can learn to play the welfare maximizing action profile under even more restrictive observational conditions. In particular, we shall assume that agents *only* have access to: (i) the action they played and (ii) the payoff they received. In this setting, the strategy adjustment mechanism of agent i takes the form

$$p_i(t) = F_i\left(\{a_i(\tau), U_i(a(\tau))\}_{\tau=0,\ldots,t-1}\right). \tag{1}$$

Such a learning rule is said to be *completely uncoupled* or *payoff-based* [21]. Recent work has shown that for finite games with generic payoffs there exist completely uncoupled learning rules that lead to Pareto optimal Nash equilibria [13]; see also [12], [22], [23]. Here we exhibit a different class of learning procedures that lead to Pareto optimal outcomes whether or not they are Nash equilibria.[2]

III. A Payoff Based Algorithm for Maximizing Welfare

Our proposed algorithm is a variant of the approach in [13], where each agent possesses an internal state variable which impacts the agent's behavior rule. The key difference between our algorithm and the one in [13] is the asymptotic guarantees. In particular, [13] guarantees convergence to a Pareto Nash equilibrium, whereas our proposed algorithm converges to a Pareto (efficient) action profile irrespective of whether or not this action profile is a Nash equilibrium. Furthermore, our algorithm uses fewer state variables than the method in [13].

At each point in time an agent's *state* can be represented as a triple $[\bar{a}_i, \bar{u}_i, m_i]$, where

- The ***benchmark action*** is $\bar{a}_i \in \mathcal{A}_i$.
- The ***benchmark payoff*** is $\bar{u}_i$, which is in the range of $U_i(\cdot)$.
- The ***mood*** is m_i, which can take on two values: *content* (C) and *discontent* (D).

The learning algorithm produces a sequence of action profiles $a(1)$, ..., $a(t)$, where the behavior of an agent i in each period $t = 1, 2, ...,$ is conditioned on agent i's underlying benchmark payoff $\bar{u}_i(t)$, benchmark action $\bar{a}_i(t)$, and mood $m_i(t) \in \{C, D\}$.

We divide the dynamics into the following two parts: the agent dynamics and the state dynamics. Without loss of generality we shall focus on the case where agent utility functions are strictly bounded between 0 and 1, i.e., for any agent $i \in N$ and action profile $a \in \mathcal{A}$ we have $1 > U_i(a) \geq 0$. Consequently, for any action profile $a \in \mathcal{A}$, the welfare function satisfies $n > W(a) \geq 0$.

[2]Such a result might seem reminiscent of the Folk Theorem, which specifies conditions under which an efficient action profile can be implemented as an equilibrium of a repeated game (see among others [23], [24]). In the present context, however, we are interested in whether agents can learn to play an efficient action profile without having any information about the game as a whole or what the other agents are doing. Hence they cannot condition their behavior on the observed behavior of others, which is a key requirement of most repeated game equilibria.

Agent Dynamics: Fix an experimentation rate $\epsilon > 0$ and constant $c > n$. Let $[\bar{a}_i, \bar{u}_i, m_i]$ be the current state of agent i.

- **Content** $(m_i = C)$: In this state, the agent chooses an action a_i according to the following probability distribution

$$p_i^{a_i} = \begin{cases} \frac{\epsilon^c}{|\mathcal{A}_i|-1} & \text{for } a_i \neq \bar{a}_i \\ 1 - \epsilon^c & \text{for } a_i = \bar{a}_i \end{cases} \tag{2}$$

  where $|\mathcal{A}_i|$ represents the cardinality of the set $\mathcal{A}_i$.

- **Discontent** $(m_i = D)$: In this state, the agent chooses an action a_i according to the following probability distribution:

$$p_i^{a_i} = \frac{1}{|\mathcal{A}_i|} \quad \text{for every} \quad a_i \in \mathcal{A}_i \tag{3}$$

  Note that the benchmark action and utility play no role in the agent dynamics when the agent is discontent.

State Dynamics: Once the agent selects an action $a_i \in \mathcal{A}_i$ and receives the payoff $u_i = U_i(a_i, a_{-i})$, where a_{-i} is the action selected by all agents other than agent i, the state is updated as follows:

- **Content** $(m_i = C)$: If $[a_i, u_i] = [\bar{a}_i, \bar{u}_i]$ the new state is determined by the transition

$$[\bar{a}_i, \bar{u}_i, C] \overset{[\bar{a}_i, \bar{u}_i]}{\longrightarrow} [\bar{a}_i, \bar{u}_i, C]. \tag{4}$$

  If $[a_i, u_i] \neq [\bar{a}_i, \bar{u}_i]$ the new state is determined by the transition

$$[\bar{a}_i, \bar{u}_i, C] \overset{[a_i, u_i]}{\longrightarrow} \begin{cases} [a_i, u_i, C] & \text{with prob } \epsilon^{1-u_i} \\ [a_i, u_i, D] & \text{with prob } 1 - \epsilon^{1-u_i}. \end{cases}$$

- **Discontent** $(m_i = D)$: If the selected action and received payoff are $[a_i, u_i]$, the new state is determined by the transition

$$[\bar{a}_i, \bar{u}_i, D] \overset{[a_i, u_i]}{\longrightarrow} \begin{cases} [a_i, u_i, C] & \text{with prob } \epsilon^{1-u_i} \\ [a_i, u_i, D] & \text{with prob } 1 - \epsilon^{1-u_i}. \end{cases}$$

To ensure that the dynamics converge to an efficient action profile, we require the following notion of interdependence in the game structure [12].

Definition 1 (Interdependence)**.** *An n-person game G on the finite action space $\mathcal{A}$ is interdependent if, for every $a \in \mathcal{A}$ and every proper subset of agents $J \subset N$, there exists an agent $i \notin J$ and a choice of actions $a'_J \in \prod_{j \in J} \mathcal{A}_j$ such that $U_i(a'_J, a_{-J}) \neq U_i(a_J, a_{-J})$.*

Roughly speaking, the interdependence condition states that it is not possible to divide the agents into two distinct subsets that do not mutually interact with one another.

These dynamics induce a Markov process over the finite state space $Z = \prod_{i \in N} (\mathcal{A}_i \times \mathcal{U}_i \times M)$, where $\mathcal{U}_i$ denotes the finite range of $U_i(a)$ over all $a \in \mathcal{A}$ and $M = \{C, D\}$ is the set of moods. We shall denote the transition probability matrix by P^ϵ for each $\epsilon > 0$. Computing the stationary distribution of this process is challenging because of the large number of states and the fact that the underlying process is not reversible. Accordingly, we shall focus on characterizing the *support* of the limiting stationary distribution, whose elements are referred to as the *stochastically stable states* [25]. More precisely, a state $z \in Z$ is stochastically stable if and only if $\lim_{\epsilon \to 0^+} \mu(z, \epsilon) > 0$ where $\mu(z, \epsilon)$ is a stationary distribution of the process P^ϵ for a fixed $\epsilon > 0$.

Theorem 1. *Let G be an interdependent n-person game on a finite joint action space $\mathcal{A}$. Under the dynamics defined above, a state $z = [a, u, m] \in Z$ is stochastically stable if and only if the following conditions are satisfied:*

(i) The action profile a optimizes $W(a) = \sum_{i \in N} U_i(a)$.

(ii) The benchmark actions and payoffs are aligned, i.e., $u_i = U_i(a)$ for all i.

(iii) The mood of each agent is content, i.e., $m_i = C$ for all i.

IV. Proof of Theorem 1

The proof relies on the theory of resistance trees for regular perturbed Markov decision processes [26], which we briefly review here. Let P^0 denote the probability transition matrix of a finite state Markov chain on the state space Z. Consider a "perturbed" process P^ϵ on Z where the "size" of the perturbations can be indexed by a scalar $\epsilon > 0$. The process P^ϵ is called a *regular perturbed Markov process* if P^ϵ is ergodic for all sufficiently small $\epsilon > 0$ and P^ϵ approaches P^0 at an exponentially smooth rate, that is,

$$\forall z, z' \in Z, \quad \lim_{\epsilon \to 0^+} P^\epsilon_{zz'} = P^0_{zz'},$$

and

$$\forall z, z' \in Z, \quad P^{\epsilon}_{zz'} > 0 \text{ for some } \epsilon > 0 \;\Rightarrow\; 0 < \lim_{\epsilon \to 0^+} \frac{P^{\epsilon}_{zz'}}{\epsilon^{r(z \to z')}} < \infty,$$

where $r(z \to z')$ is a nonnegative real number called the *resistance* of the transition $z \to z'$. (Note in particular that if $P^0_{zz'} > 0$ then $r(z \to z') = 0$.)

Let the recurrence classes of P^0 be denoted by $E_1, E_2, ..., E_M$. For each pair of distinct recurrence classes E_i and E_j, $i \neq j$, an *ij-path* is defined to be a sequence of distinct states $\zeta = (z_1 \to z_2 \to ... \to z_m)$ such that $z_1 \in E_i$ and $z_m \in E_j$. The *resistance* of this path is the sum of the resistances of its edges, that is,

$$r(\zeta) = r(z_1 \to z_2) + r(z_2 \to z_3) + ... + r(z_{m-1} \to z_m).$$

Let $\rho_{ij} = \min r(\zeta)$ be the least resistance over all ij-paths ζ. Note that ρ_{ij} must be positive for all distinct i and j, because there exists no path of zero resistance between distinct recurrence classes.

Now construct a complete directed graph with M vertices, one for each recurrence class. The vertex corresponding to class E_j will be called j. The weight on the directed edge $i \to j$ is ρ_{ij}. A *j-tree* T is a set of $M - 1$ directed edges such that, from every vertex different from j, there is a unique directed path in the tree to j. The resistance of such a tree is the sum of the resistances on the $M - 1$ edges that compose it. The *stochastic potential*, γ_j, of the recurrence class E_j is the minimum resistance over all trees rooted at j. The following result provides a simple criterion for determining the stochastically stable states ([26], Theorem 4).

Let P^{ϵ} be a regular perturbed Markov process, and for each $\epsilon > 0$ let μ^{ϵ} be the unique stationary distribution of P^{ϵ}. Then $\lim_{\epsilon \to 0} \mu^{\epsilon}$ exists and the limiting distribution μ^0 is a stationary distribution of P^0. The stochastically stable states (i.e., the support of μ^0) are precisely those states contained in the recurrence classes with minimum stochastic potential.

It can be verified that the dynamics introduced above define a regular perturbed Markov process. The proof of Theorem 1 proceeds by a series of lemmas. Let C^0 be the subset of states in which each agent is content and the benchmark action and utility are aligned. That is, if $[a, u, m] \in C^0$ then $u_i = U_i(a)$ and $m_i = C$ for each agent $i \in N$. Let D^0 represent the set of states in which everyone is discontent. That is, if $[a, u, m] \in D^0$ then $u_i = U_i(a)$ and $m_i = D$ for each agent $i \in N$.

The first lemma provides a characterization of the recurrence classes of the unperturbed process P^0.

Lemma 2. *The recurrence classes of the unperturbed process P^0 are D^0 and all singletons $z \in C^0$.*

Proof: The set of states D^0 represents a single recurrence class of the unperturbed process since the probability of transitioning between any two states $z_1, z_2 \in D^0$ is $O(1)$ and when $\epsilon = 0$ there is no possibility of exiting from D^0. Suppose now that a proper subset of agents $S \subset N$ is discontent and the benchmark actions and benchmark utilities of all other agents are a_{-S} and u_{-S} respectively. By interdependence, there exists an agent $j \notin S$ and an action tuple $a'_S \in \prod_{i \in S} \mathcal{A}_i$ such that $u_j \neq U_j(a'_S, a_{-S})$. This situation cannot be a recurrence class of the unperturbed process because the agent set S will eventually play action a'_S with probability 1, thereby causing agent j to become discontent. This process can be repeated to show that all agents will eventually become discontent with probability $O(1)$; hence any state that consists of a partial collection of discontent agents $S \subset N$ is not a recurrence class of the unperturbed process.

Lastly, consider a state $[a, u, C]$ where all agents are content but there exists at least one agent i whose benchmark action and benchmark utility are not aligned, i.e., $u_i \neq U_i(a)$. For the unperturbed process, at the ensuing time step the action profile a will be played and agent i will become discontent since $u_i \neq U_i(a)$. Since one agent is discontent, all agents will eventually become discontent. This completes the proof of Lemma 2. ■

We know from [26] that the computation of the stochastically stable states can be reduced to an analysis of rooted trees on the vertex set consisting solely of the recurrence classes. We denote the collection of states D^0 by a single variable D to represent this single recurrence class since the exit probabilities are the same for all states in D^0. By Lemma 2, the set of recurrence classes consists of the singleton states in C^0 and also the singleton state D. Accordingly, we represent a state $z \in C^0$ by just $[a, u]$ and drop the extra notation highlighting that the agents are content. We now reiterate the definition of edge resistance.

Definition 2 (Edge resistance)**.** *For every pair of distinct recurrence classes w and z, let $r(w \to z)$ denote the total resistance of the least-resistance path that starts in w and ends in z. We call*

$w \to z$ *an edge and* $r(w \to z)$ *the resistance of the edge.*

Let $z = [a, u]$ and $z' = [a', u']$ be any two distinct states in C^0. The following observations will be useful.

(i) The resistance of the transition $z \to D$ satisfies

$$r(z \to D) = c.$$

This holds because one experiment can cause all agents to become discontent.

(ii) The resistance of the transition $D \to z$ satisfies

$$r(D \to z) = \sum_{i \in N} (1 - u_i) = n - W(a).$$

This holds because each agent i needs to accept the benchmark payoff u_i, which has a resistance $(1 - u_i)$.

(iii) The resistance of the transition $z \to z'$ satisfies

$$c \leq r(z \to z') < 2c.$$

This holds because $r(z \to z') \leq r(z \to D) + r(D \to z')$ by the definition of edge resistance. Therefore, each transition of minimum resistance includes at most one agent who experiments.

The following lemma characterizes the stochastic potential of the states in C^0. Before stating this lemma, we define a *path* $\mathcal{P}$ over the states $D \cup C^0$ to be a sequence of edges of the form

$$\mathcal{P} = \{z^0 \to z^1 \to ... \to z^m\},$$

where each z^k for $k \in \{0, 1, ..., m\}$ is in $D \cup C^0$. The *resistance* of a path $\mathcal{P}$ is the sum of the resistance of each edge in the path, i.e.,

$$R(\mathcal{P}) = \sum_{k=1}^{m} r(z^{k-1} \to z^k).$$

Lemma 3. *The stochastic potential of any state* $z = [a, u]$ *in* C^0 *is*

$$\gamma(z) = c\left(\left|C^0\right| - 1\right) + \sum_{i \in N} (1 - u_i). \tag{5}$$

Proof: We first prove that (5) is an upper bound for the stochastic potential of z by constructing a tree rooted at z with the prescribed resistance. To that end, consider the tree T with the following properties:

P-1: The edge exiting each state $z' \in C^0 \setminus \{z\}$ is of the form $z' \to D$. The total resistance associated with these edges is $c\left(|C^0|-1\right)$.

P-2: The edge existing the state D is of the form $D \to z$. The resistance associated with this edge is $\sum_{i\in N}(1-u_i)$.

The tree T is rooted at z and has total resistance $c\left(|C^0|-1\right)+\sum_{i\in N}(1-u_i)$. It follows that $\gamma(z) \leq c\left(|C^0|-1\right)+\sum_{i\in N}(1-u_i)$, hence (5) holds as an inequality. It remains to be shown that the right-hand side of (5) is also a lower bound for the stochastic potential.

We argue this by contradiction. Suppose there exists a tree T rooted at z with resistance $R(T) < c\left(|C^0|-1\right)+\sum_{i\in N}(1-u_i)$. Since the tree T is rooted at z we know that there exists a path $\mathcal{P}$ from D to z of the form

$$\mathcal{P} = \{D \to z^1 \to z^2 \to ... \to z^m \to z\},$$

where $z^k \in C^0$ for each $k \in \{1,...,m\}$. We claim that the resistance associated with this path of $m+1$ transitions satisfies

$$R(\mathcal{P}) \geq mc + \sum_{i\in N}(1-u_i).$$

The term mc comes from applying observation (iii) to the last m transitions on the path $\mathcal{P}$. The term $\sum_{i\in N}(1-u_i)$ comes from the fact that each agent needs to accept u_i as the benchmark payoff at some point during the transitions.

Construct a new tree T' still rooted at z by removing the edges in $\mathcal{P}$ and adding the following edges:

- $D \to z$ which has resistance $\sum_{i\in N}(1-u_i)$.
- $z^k \to D$ for each $k \in \{1,...,m\}$ which has total resistance mc.

The new tree T' is still rooted at z and has a total resistance that satisfies $R(T') \leq R(T)$. Note that if the path $\mathcal{P}$ was of the form $D \to z$ then this augmentation does not alter the tree structure.

Now suppose that there exists an edge $z' \to z''$ in the tree T' for some states $z', z'' \in C^0$. By observation (iii) the resistance of this edge satisfies $r(z' \to z'') \geq c$. Construct a new tree T'' by removing the edge $z' \to z''$ and adding the edge $z' \to D$, which has a resistance c. This new

tree T'' is rooted at z, and its resistance satisfies

$$\begin{aligned} R(T'') &= R(T') + r(z' \to D) - r(z' \to z'') \\ &\leq R(T') \\ &\leq R(T). \end{aligned}$$

Repeat this process until we have constructed a tree T^* for which no such edges exist. Note that the tree T^* satisfies properties P-1 and P-2 and consequently has a total resistance $R(T^*) = c\left(|C^0| - 1\right) + \sum_{i \in N}\left(1 - u_i\right)$. Since by construction $R(T^*) \leq R(T)$ we have a contradiction. This completes the proof of Lemma 3. ■

We will now prove Theorem 1 by analyzing the minimum resistance trees using the above lemmas. We first show that the state D is not stochastically stable. Suppose, by way of contradiction, that there exists a minimum resistance tree T rooted at the state D. Then there exists an edge in the tree T of the form $z \to D$ for some state $z \in C^0$ and the resistance of this edge is c. Create a new tree T' rooted at z by removing the edge $z \to D$ from T and adding the edge $D \to z$. The latter has resistance $n < c$. Therefore

$$\begin{aligned} R(T') &= R(T) + r(D \to z) - r(z \to D) \\ &\leq R(T) + n - c \\ &< R(T). \end{aligned}$$

Hence T is not a minimum resistance tree. This contradiction shows that the state D is not stochastically stable. It follows that all the stochastically stable states are contained in the set C^0.

From Lemma 3 we know that a state $z = [a, u]$ in C^0 is stochastically stable if and only if

$$a \in \arg\min_{a^* \in \mathcal{A}} \left\{ c\left(\left|C^0\right| - 1\right) + \sum_{i \in N}\left(1 - U_i(a^*)\right) \right\},$$

equivalently

$$a \in \arg\max_{a^* \in \mathcal{A}} \left\{ \sum_{i \in N} U_i(a^*) \right\}.$$

Therefore, a state is stochastically stable if and only if the action profile is efficient. This completes the proof of Theorem 1. □

V. Relaxing Interdependence

In this section we focus on whether the interdependence condition in Definition 1 can be relaxed while ensuring that the stochastically stable states remain efficient. Recall that a game is interdependent if it is not possible to partition the agents into two distinct groups S and $N \setminus S$ that do not mutually interact with one another. One way that this condition can fail is that the game can be broken into two completely separate sub-games that can be analyzed independently. In this case our algorithm ensures that in each sub-game the only stochastically stable states are the efficient action profiles. Hence, this remains true in the full game.

In general, however, some version of interdependence is needed. To see why, consider the following two-player game:

	A	B
A	$1/2, 1/4$	$1/2, 0$
B	$1/4, 0$	$1/4, 3/4$

Here, the row agent affects the column agent but the reverse is not true. Consequently, the recurrence states of the unperturbed process are $\{AA, AB, BA, BB, A\emptyset, B\emptyset, \emptyset\emptyset\}$ where: $A\emptyset$ is the state where agent 1 is content with action profile A and agent 2 is discontent; $\emptyset\emptyset$ is the state where both agents are discontent. We claim that the action profile (A, A), which is not efficient, is stochastically stable. This can be deduced from Figure 1 (here we choose $c = n = 2$). The

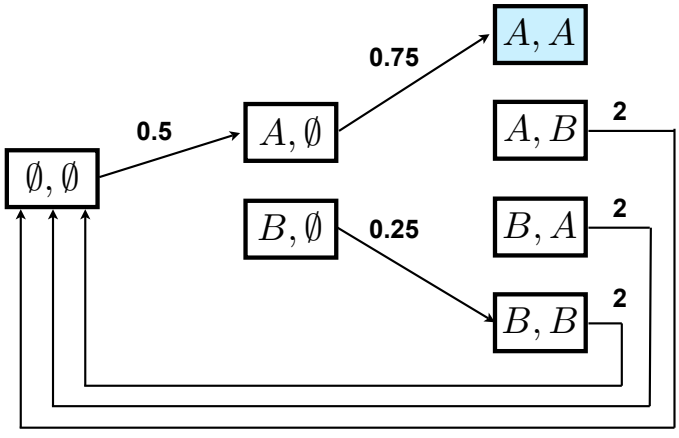

Fig. 1. Illustration of the minimum resistance tree rooted at the action profile (A, A).

illustrated resistance tree has minimum stochastic potential because each edge in the given tree has minimum resistance among the edges exiting from that vertex. Consequently, this inefficient action profile AA is stochastically stable.

VI. Illustrations

In this section we shall apply our results to several examples in order to illustrate how the algorithm works in concrete terms.

A. *Prisoner's dilemma*

Consider the following prisoner's dilemma game where all players' utilities are scaled between 0 and 1:

	C	D
C	$1/2, 1/2$	$0, 2/3$
D	$2/3, 0$	$1/3, 1/3$

It is easy to verify that these payoffs satisfy the interdependence condition. Consequently, our algorithm guarantees that the action profile (C, C) is the only stochastically stable state. We will now verify this by directly computing the resistances for of each of the transitions. The recurrence classes of the unperturbed process are $(CC, CD, DC, DD, \emptyset)$, where the agents are content for the given action profiles and $\emptyset$ corresponds to the scenario where both agents are discontent. (For notation simplicity we omit the baseline utilities for each of the four joint action profiles.)

Consider the transition $CC \to DD$. Its resistance is

$$r(CC \to DD) = c + (1 - 1/3) + (1 - 1/3) = c + 4/3.$$

The term c comes from the fact that we have only one experimenter and the term $2(1 - 1/3)$ results from the fact that both agents 1 and 2 need to accept the new benchmark payoff of $1/3$ in this transition. Let $c = n = 2$ for the remaining portion of this section. The resistances of all possible transitions are highlighted in Table 1. Each entry in this table represents the resistance going from the row-state to the column-state.

The stochastic potential of each of the five states can be evaluated by analyzing the trees rooted at each state. The minimum resistance tree rooted at each state is shown in Figure 2. Note that

	CC	CD	DC	DD	$\emptyset$
CC	$\cdot$	$2+(1-2/3)+(1-0)=10/3$	$2+(1-2/3)+(1-0)=10/3$	$2+2(1-1/3)=10/3$	2
CD	$2+2(1-1/2)=3$	$\cdot$	$2+(1-2/3)+(1-0)=10/3$	$2+2(1-1/3)=10/3$	2
DC	$2+2(1-1/2)=3$	$2+(1-2/3)+(1-0)=10/3$	$\cdot$	$2+2(1-1/3)=10/3$	2
DD	$2+2(1-1/2)=3$	$2+(1-2/3)+(1-0)=10/3$	$2+(1-2/3)+(1-0)=10/3$	$\cdot$	2
$\emptyset$	$2(1-1/2)=1$	$(1-2/3)+(1-0)=4/3$	$(1-2/3)+(1-0)=4/3$	$2(1-1/3)=4/3$	$\cdot$

TABLE I

EVALUATION OF RESISTANCES FOR PRISONER'S DILEMMA GAME.

each of the minimum resistance trees has the very simple structure identified in Lemma 3. It is evident that CC has minimum stochastic potential, hence is the unique stochastically stable state.

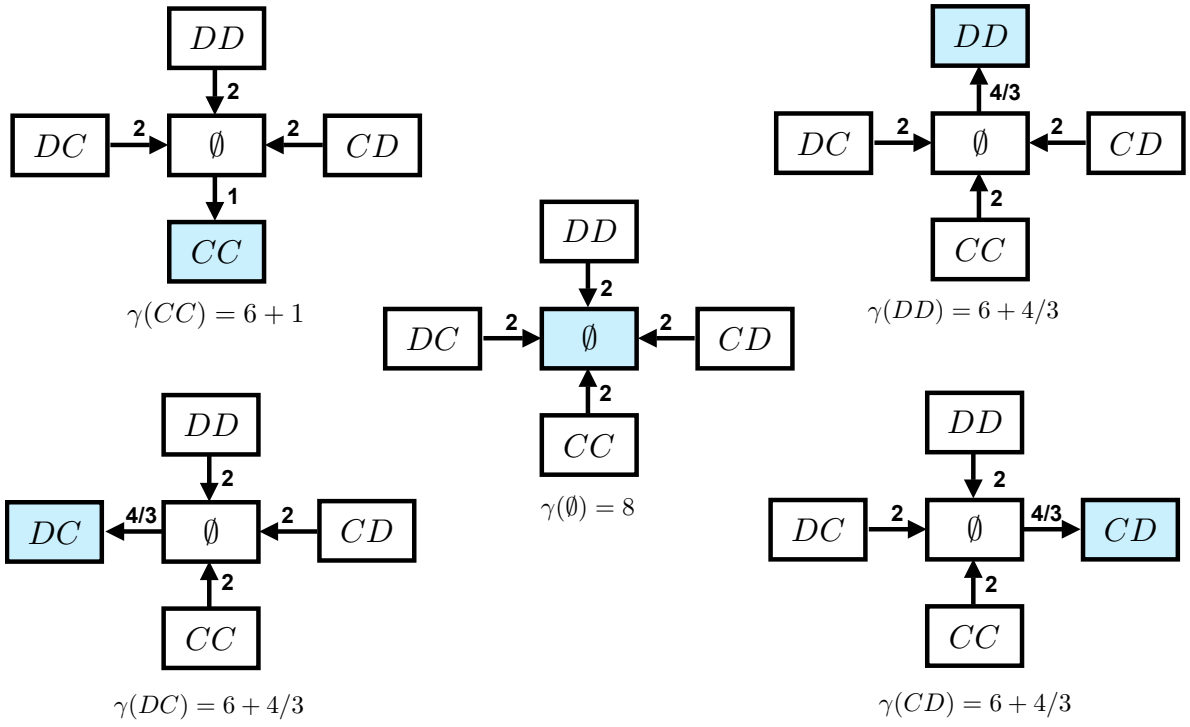

Fig. 2. Stochastic potential for each state in the prisoner's dilemma game.

B. Wind farms

In this section we focus on the control of a wind farm where the goal is to generate as much power as possible. The ingredients of the problem are the following:

- **Agents**: Individual wind turbines denoted by $i \in \{1, 2, ..., n\} = N$.
- **Decisions**: The action set for turbine i is the set of axial induction factors denoted by $\mathcal{A}_i$. The axial induction factor indicates the amount of power the turbine extracts from the wind given the current wind conditions.
- **Power production**: The power produced by turbine i is a function of the actions of all turbines. The power generated by turbine i given the decision of all turbines $a = (a_1, a_2, ..., a_n) = (a_i, a_{-i})$ is given by $P_i(a_i, a_{-i})$. We assume throughout that the exogenous wind conditions are fixed so we omit this in the power expression for each turbine.
- **System level objective:** The goal is to optimize the total power production in the wind farm, that is,

$$P(a) = \sum_{i \in N} P_i(a)$$

Most of the existing research on the control of wind turbines focuses on the single turbine setting [27]. Controlling an array of turbines in a wind farm is fundamentally more challenging because of the aerodynamic interaction between the turbines. In fact, these interactions render most of the single turbine control algorithms *highly inefficient* for optimizing wind farm productivity by introducing a degree of interconnectivity between the objective (or power) functions of the individual turbines [18], [28]. More specifically, the power generated by one turbine is dependent on the exogenous wind conditions coupled with the axial induction factors of other turbines. Lastly, these aerodynamic interactions are poorly characterized, hence the precise structural form of the power generated by the wind farm $P(a_1, ..., a_n)$ is not well characterized.

The results in this paper provide a method for optimizing power production that takes into the aerodynamic interactions between the turbines, but does not assume that these are known to the system designer. More generally the method described here provides a fully decentralized method for optimizing total system performance when little is known about how the individual components interact.

REFERENCES

[1] G. Chasparis and J. Shamma, "Distributed dynamic reinforcement of efficient outcomes in multiagent coordination and network formation," 2011, discussion paper, Department of Electrical Engineering, Georgia Tech.

[2] N. Li and J. R. Marden, "Decoupling coupled constraints through utility design," 2011, discussion paper, Department of ECEE, University of Colorado, Boulder.

[3] ——, "Designing games for distributed optimization," 2011, discussion paper, Department of ECEE, University of Colorado, Boulder.

[4] J. R. Marden, "State based potential games," 2011, discussion paper, Department of ECEE, University of Colorado, Boulder.

[5] R. Gopalakrishnan, J. R. Marden, and A. Wierman, "An architectural view of game theoretic control," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 3, pp. 31–36, 2011.

[6] J. R. Marden, G. Arslan, and J. S. Shamma, "Connections between cooperative control and potential games," *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, vol. 39, pp. 1393–1407, December 2009.

[7] G. Arslan, J. R. Marden, and J. S. Shamma, "Autonomous vehicle-target assignment: a game theoretical formulation," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 129, pp. 584–596, September 2007.

[8] R. Johari, "The price of anarchy and the design of scalable resource allocation mechanisms," in *Algorithmic Game Theory*, N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, Eds. Cambridge University Press, 2007.

[9] R. S. Komali and A. B. MacKenzie, "Distributed topology control in ad-hoc networks: A game theoretic perspective," in *Proceedings of IEEE Consumer Communication and Network Conference*, 2007.

[10] 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*, vol. 48, pp. 373–396, February 2009.

[11] J. R. Marden, G. Arslan, and J. S. Shamma, "Joint strategy fictitious play with inertia for potential games," *IEEE Transactions on Automatic Control*, vol. 54, pp. 208–220, February 2009.

[12] H. P. Young, "Learning by trial and error," *Games and Economic Behavior*, vol. 65, pp. 626–643, 2009.

[13] B. R. Pradelski and H. P. Young, "Learning efficient Nash equilibria in distributed systems," 2010, discussion paper, Department of Economics, University of Oxford.

[14] L. S. Shapley, "Stochastic games," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 39, no. 10, pp. 1095–1100, 1953.

[15] D. Wolpert and K. Tumor, "An overview of collective intelligence," in *Handbook of Agent Technology*, J. M. Bradshaw, Ed. AAAI Press/MIT Press, 1999.

[16] J. R. Marden and A. Wierman, "Distributed welfare games," 2008, discussion paper, Department of ECEE, University of Colorado, Boulder.

[17] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*. New York, NY, USA: Cambridge University Press, 2007.

[18] K. E. Johnson and N. Thomas, "Wind farm control: Addressing the aerodynamic interaction among wind turbines," in *Proceedings of the 2009 American Control Conference*, 2009.

[19] S. Hart and A. Mas-Colell, "Stochastic uncoupled dynamics and Nash equilibrium," *Games and Economic Behavior*, vol. 57, no. 2, pp. 286–303, 2006.

[20] ——, "Uncoupled dynamics do not lead to Nash equilibrium," *American Economic Review*, vol. 93, no. 5, pp. 1830–1836, 2003.

[21] D. Foster and H. Young, "Regret testing: Learning to play Nash equilibrium without knowing you have an opponent," *Theoretical Economics*, vol. **1**, pp. 341–367, 2006.

[22] I. Arieli and Y. Babichenko, "Average testing and the efficient boundary," 2011, discussion paper, Department of Economics, University of Oxford and Hebrew University.

[23] D. Fudenberg and E. Maskin, "The folk theorem in repeated games with discounting or with incomplete information," *Econometrica*, vol. **54**, pp. 533–554, 1986.

[24] M. Osborne and A. Rubinstein, *A Course in Game Theory*. Cambridge, MA: MIT Press, 1994.

[25] D. Foster and H. Young, "Stochastic evolutionary games dynamics," *Journal of Theoretical Population Biology*, vol. **38**, 219–232.

[26] H. P. Young, "The evolution of conventions," *Econometrica*, vol. 61, no. 1, pp. 57–84, January 1993.

[27] L. Pao and K. Johnson, "Control of wind turbines: Approaches, challenges, and recent developments," *Control Systems, IEEE*, vol. 31, no. 2, pp. 44 –62, 2011.

[28] R. J. Barthelmie and L. E. Jensen, "Evaluation of wind farm efficiency and wind turbine wakes at the nysted offshore wind farm," *Wind Energy*, vol. 13, no. 6, pp. 573–586, 2010.